Efficiency of Restricted Tolls in Non-atomic Network Routing Games

Vincenzo Bonifaci[1⋆], Mahyar Salek[2⋆⋆], and Guido Schäfer[3]

[1] Università Telematica Internazionale Uninettuno, Rome, Italy
vincenzo.bonifaci@uninettunouniversity.net
[2] Department of Computer Science, University of Southern California, USA
salek@usc.edu
[3] Center for Mathematics and Computer Science (CWI) and VU University Amsterdam, The Netherlands
g.schaefer@cwi.nl

Abstract. An effective means to reduce the inefficiency of Nash flows in non-atomic network routing games is to impose tolls on the arcs of the network. It is a well-known fact that marginal cost tolls induce a Nash flow that corresponds to a minimum cost flow. However, despite their effectiveness, marginal cost tolls suffer from two major drawbacks, namely (i) that potentially every arc of the network is tolled, and (ii) that the imposed tolls can be arbitrarily large.

In this paper, we study the *restricted network toll problem* in which tolls can be imposed on the arcs of the network but are restricted to not exceed a predefined threshold for every arc. We show that optimal restricted tolls can be computed efficiently for parallel-arc networks and affine latency functions. This generalizes a previous work on taxing subnetworks to arbitrary restrictions. Our algorithm is quite simple, but relies on solving several convex programs. The key to our approach is a characterization of the flows that are inducible by restricted tolls for single-commodity networks. We also derive bounds on the efficiency of restricted tolls for multi-commodity networks and polynomial latency functions. These bounds are tight even for parallel-arc networks. Our bounds show that restricted tolls can significantly reduce the price of anarchy if the restrictions imposed on arcs with high-degree polynomials are not too severe. Our proof is constructive. We define tolls respecting the given thresholds and show that these tolls lead to a reduced price of anarchy by using a (λ, μ)-smoothness approach.

⋆ Part of this work has been done while the author was at MPI Informatics, Saarbrücken, Germany.
⋆⋆ Part of this work has been done while the author spent a summer internship at CWI.

1   Introduction

Congestion in traffic networks has several negative effects as it causes, e.g., environmental pollution, waste of natural resources and time, stress on the traffic participants, etc. With the increase in traffic in recent years, it becomes an increasingly important issue to implement regulation means that efficiently reduce congestion in networks. In this context, road pricing has long been recognized as being one of the most effective regulation means. The idea is to charge traffic participants for the usage of roads by imposing tolls. Such negative incentives usually lead to a change in behavior in that traffic participants, for example, travel along longer (but less congested) routes, avoid certain parts of the network (at certain times), or do not travel at all during peak-times, etc. Recent technological advances, in particular, in satellite technology, facilitate the realization of such pricing schemes, e.g., by enabling to collect tolls electronically. Furthermore, they open up the possibility to implement *dynamic* pricing schemes, in which tolls may vary over time or depend on congestion.

In this paper, we study the problem of computing efficient pricing schemes to reduce congestion in network applications caused by selfish behavior. In our studies we use a well-established model of traffic routing in networks, also known as the *Wardrop model*. In this model, we are given a directed graph $G = (V, A)$ with latency functions $\ell := (\ell_a)_{a \in A}$ on the arcs, k commodities $(s_1, t_1), \dots, (s_k, t_k) \in V \times V$, and a non-negative demand r_i for every commodity $i \in [k]$. The latency functions are used to model the (flow-dependent) congestion on the arcs and are assumed to be non-negative and non-decrasing. The demand r_i of commodity $i \in [k]$ specifies the amount of flow that needs to be routed from s_i to t_i. A common interpretation is that the r_i units of flow represent an (infinitely) large population of players, each controlling an infinitesimal amount of the r_i flow units. The goal of every player is to send his flow along a shortest latency path from its source s_i to its destination t_i. The resulting game is also called a *non-atomic network routing game*. A flow f in which no player has an incentive to unilaterally deviate from its path is called a *Nash flow* (or *Wardrop flow*).

In general, a Nash flow can be inefficient in the sense that it does not correspond to an optimal flow that minimizes *social cost*, i.e., the total average latency. The *price of anarchy* [12] is a measure to quantify the efficiency loss caused by selfish behavior. In the context of network routing games, it is defined as the worst-case ratio over all instances between the cost of a Nash flow and the cost of an optimal flow. In a seminal work, Roughgarden and Tardos [17] show that the price of anarchy of non-atomic network routing games is unbounded in general and provide bounds for specific classes of latency functions, e.g., polynomial latency functions.

An effective means to reduce the price of anarchy in network routing games is to impose non-negative tolls $\tau := (\tau_a)_{a \in A}$ on the arcs. We consider both *dynamic* and *static* tolls in this paper. In the dynamic case, the toll that is imposed on arc $a \in A$ is defined by a (flow-dependent) toll function τ_a which maps every flow value x to a non-negative toll $\tau_a(x)$. In the static case, the toll on arc $a \in A$ is specified by a non-negative constant τ_a. By traversing an arc $a \in A$ with flow value x, a player now experiences a delay of $\ell_a(x)$ and additionally has to pay a toll of $\tau_a(x)$. We let $\alpha > 0$ be a parameter that specifies how players value time over money. That is, the combined cost of an arc $a \in A$ with flow value x is defined as $\phi_a(x) := \ell_a(x) + \alpha\tau_a(x)$. We assume that every players' goal is to choose a path that minimizes his total combined cost. A stable outcome of this game is a Nash flow with respect to the combined cost functions $\phi := (\phi_a)_{a \in A}$.

A fundamental result due to Beckman, McGuire and Winsten [1] states that *marginal cost tolls* induce a Nash flow that is socially optimal. That is, if we define $\tau_a(x) := \frac{1}{\alpha} x \cdot \ell_a'(x)$ for every arc $a \in A$ then a Nash flow with respect to ϕ is an optimal flow with respect to ℓ. Even

though marginal cost tolls are theoretically appealing, they have two major drawbacks: (i) It is assumed that tolls can be imposed on *every* arc of the network. (ii) The tolls imposed on the arcs can be arbitrarily large. These are severe drawbacks that rule out the applicability of marginal cost tolls in several situations.

In this paper, we overcome these drawbacks by restricting the set of feasible tolls. These restrictions are assumed to be given exogenously by means of threshold functions on the arcs. That is, in a ***restricted network toll problem*** we are given an instance of the network routing game together with some threshold functions $\theta := (\theta_a)_{a \in A}$ that specify an upper bound on the maximum toll chargeable on each arc. As for tolls, we call threshold functions θ *dynamic* if they are flow-dependent and *static* otherwise. We call the tolls $\tau = (\tau_a)_{a \in A}$ θ-*restricted* if for every $a \in A$, $0 \le \tau_a(x) \le \theta_a(x)$ for all flow values $x \ge 0$. Given θ-restricted tolls τ, let f^τ denote a Nash flow that is induced by τ, i.e., f^τ is a Nash flow with respect to $\phi = \ell + \alpha\tau$.

Our model incorporates several interesting special cases. For example, we can enforce that tolls are only imposed on a subnetwork induced by a subset $T \subseteq A$ of the arcs by setting $\theta_a = \infty$ for every $a \in T$ and $\theta_a = 0$ otherwise. Another example is that we can restrict the toll on each arc $a \in A$ by a (flow-independent) threshold value θ_a. Yet another example is that we can require that the toll on each arc $a \in A$ does not exceed a certain fraction of the latency of that arc, e.g., $\theta_a(x) = \varepsilon\ell_a(x)$ for some $\varepsilon > 0$.

Given the restrictions imposed on the set of feasible tolls, the following two natural questions arise and will be studied in this paper:

1. Can one quantify the efficiency of θ-restricted tolls?

   We are interested in studying the efficiency of θ-restricted tolls in relation to the cost of a socially optimal flow. To this aim, we define the *efficiency* of θ-restricted tolls as the minimum ratio of the cost of a Nash flow f^τ inducible by θ-restricted tolls τ and the cost of an optimal flow. We also address the problem of computing θ-restricted tolls that guarantee a certain efficiency.

2. Can one compute (approximately) optimal θ-restricted tolls?

   We consider the roblem of computing (approximately) optimal θ-restricted tolls. We call θ-restricted tolls τ *optimal* if the Nash flow f^τ induced by τ has cost less than or equal to any other Nash flow that is inducible by θ-restricted tolls. Similarly, θ-restricted tolls τ are said to be λ-*approximate* for some $\lambda \ge 1$ if the cost of f^τ is at most λ times the cost of any other Nash flow inducible by θ-restricted tolls.

Clearly, from the discussion above it follows that we obtain an efficiency of one if $\theta_a = \infty$ for every $a \in A$. On the other hand, the efficiency coincides with the price of anarchy if $\theta_a = 0$ for every $a \in A$.

The special case that tolls can only be imposed on a subset $T \subseteq A$ of the arcs has recently been studied by Hoefer, Olbrich and Skopalik [10]. For this case, the authors derive an algorithm to compute optimal T-restricted tolls for parallel-arc networks with affine latency functions. They also prove that the problem of computing optimal tolls is NP-hard, even for two-commodity networks and affine latency functions. Note that the restricted network toll problem that we consider here is more general, and thus this hardness result extends to our setting.

Our Results. The main contributions presented in this paper are as follows:

In Section 3 we show that optimal θ-restricted tolls can be computed efficiently in parallel-arc networks with affine latency functions. This extends the result of Hoefer et al. [10] to

arbitrary dynamic threshold functions on the arcs. Our approach is different from the one described in [10]. Despite its generality, our algorithm is quite simple. The key to our approach is a characterization of the flows that are inducible by θ-restricted tolls. Our characterization applies to single-commodity networks in general. It allows us to determine whether a given flow is inducible by θ-restricted tolls by verifying whether there is a negative cycle in a properly constructed graph (which can be done in polynomial time). Based on this characterization, we derive an algorithm to compute optimal θ-restricted tolls for parallel-arc networks. Our algorithm works for general latency functions; however, we can only guarantee polynomial running time if all latency and threshold functions are affine (in which case we need to solve a series of convex programs).

In Section 4 we derive upper bounds on the efficiency of dynamic θ-restricted tolls for multi-commodity networks and polynomial latency functions of degree p. Our pricing scheme is a simple and natural adaptation of marginal cost tolls to the restricted setting: for every arc $a \in A$ we charge marginal cost tolls if this does not exceed the threshold θ_a, and we charge θ_a otherwise. Essentially, we show that these tolls achieve an efficiency that depends on the degree of the polynomial and the smallest ratio between the threshold value and the latency of an arc (see Section 4 for precise statements). The technique that we use to establish these bounds rests on a (λ, μ)-*smoothness* approach [16] that has previously been used successfully to bound the price of anarchy of network routing games [2, 8] and in a more general context in [16]. We also prove that our bounds are tight, even for parallel-arc networks. Our pricing scheme also provides a way to compute θ-restricted tolls for multi-commodity networks and polynomial latency functions that are λ-approximate, where λ is equal to the established efficiency.

Our findings support the intuition that, in order to achieve a good efficiency, it is more important to be able to impose tolls on the arcs that are sensitive to flow changes (high degree polynomials) than on the arcs that are relatively insensitive to flow changes (low degree polynomials).

For the special case that all restrictions are of the form $\theta_a(x) = \varepsilon \ell_a(x)$, our bound matches exactly the *price of stability* of ε-Nash flows shown by Christodoulou, Koutsoupias and Spirakis [3]:

$$
\begin{cases}
\left((1+\varepsilon) \left(1 - \frac{p}{p+1} \left(\frac{1+\varepsilon}{p+1} \right)^{1/p} \right) \right)^{-1} & \text{if } \varepsilon < p \\
1 & \text{if } \varepsilon \geq p.
\end{cases}
$$

Our result therefore shows that such tolls allow us to reduce the (generally large) inefficiency of Nash flows to at least the price of stability of ε-Nash flows; the actual instance-dependent efficiency of such tolls might be better than that.

All our results mentioned above hold for dynamic threshold functions (and thus also for static ones).

Related Work. As mentioned above, most related to our work is the recent article [10] by Hoefer et al. who study the problem of taxing subnetworks, a special case of the restricted toll problem that we consider here. The authors focus on the problem of computing optimal tolls. They show that this problem is NP-hard for two-commodity networks and affine latency functions by a reduction from partition. They also derive an algorithm to compute optimal tolls for parallel-arc networks and affine latency functions. Their algorithm is sophisticated and crucially exploits that the restrictions are of the form $\theta_a \in \{0, \infty\}$ for every arc $a \in A$.

The classic result that marginal cost pricing induces optimal flows is due to Beckmann, McGuire and Winsten [1]. More recently, it has been shown that optimal-inducing tolls exist

even when users are heterogeneous, i.e., have different latency/toll trade-offs: this was first shown for single-commodity networks by Cole, Dodis and Roughgarden [4] and then extended to the multi-commodity case by Fleischer, Jain and Mahdian [7] and independently by Karakostas and Kolliopoulos [11].

Cole et al. [5] study the setting in which the cost of each user is defined as the latency plus the taxes paid by the user. For heterogeneous users, Fleischer [6] shows that if there is a single commodity, then tolls that are linear in the maximum latency of the optimal flow are sufficient to force the users to the system optimum. The question of computing tolls that enforce particular flows has been studied in [9]. The above papers all study the non-atomic model; tolls for heterogeneous users in the context of atomic routing games have been considered by Swamy [18].

Bounds on the price of anarchy and the price of stability of ε-Nash flows in non-atomic and atomic congestion games, including network congestion games, have been derived recently by Christodoulou et al. [3].

2 Preliminaries

We provide formal definitions of the concepts introduced in the Introduction. Suppose we are given an instance $I = (G, (\ell_a)_{a \in A}, (s_i, t_i)_{i \in [k]}, (r_i)_{i \in [k]})$ of the non-atomic network routing game. Let $\mathcal{P}_i$ denote the set of all simple directed s_i, t_i-paths in G and define $\mathcal{P} := \cup_{i \in [k]} \mathcal{P}_i$. An outcome of the game is a flow $f : \mathcal{P} \to \mathbb{R}_+$ that is feasible, i.e., $\sum_{P \in \mathcal{P}_i} f_P = r_i$ for every $i \in [k]$. Given a flow f, the total flow on arc $a \in A$ is defined as $f_a := \sum_{P \in \mathcal{P} : a \in P} f_P$. We define the latency of a path $P \in \mathcal{P}$ with respect to f as $\ell_P(f) := \sum_{a \in P} \ell_a(f_a)$. The total cost $C(f)$ of f is given by its average latency, i.e., $C(f) := \sum_{P \in \mathcal{P}} f_P \ell_P(f)$. A flow that minimizes $C(\cdot)$ is called optimal and denoted by f^*. A feasible flow f is called a *Nash flow* (or *Wardrop flow*) with respect to $\ell := (\ell_a)_{a \in A}$ if and only if

$$\forall i \in [k], \ \forall P \in \mathcal{P}_i, \ f_P > 0 : \qquad \ell_P(f) \leq \ell_{P'}(f) \quad \forall P' \in \mathcal{P}_i. \tag{1}$$

Throughout this paper, we assume that the latency functions are non-negative, monotone non-decreasing, differentiable and semi-convex, i.e., $x \cdot \ell_a(x)$ is convex for every arc $a \in A$; such latency functions are also called *standard* [14]. The cost of a Nash flow is unique if the latency functions are standard.

In a *restricted network toll problem* we are given an instance I of the network routing game and threshold functions $\theta := (\theta_a)_{a \in A}$ on the arcs. In this setting, non-negative tolls $\tau := (\tau_a)_{a \in A}$ can be imposed on the arcs that have to obey the bounds defined by the threshold functions $(\theta_a)_{a \in A}$. In the most general setting, both tolls and threshold functions are flow-dependent. Given a feasible flow f, we define the combined cost that a player experiences by traversing arc $a \in A$ as $\phi_a(f_a) = \ell_a(f_a) + \alpha \tau_a(f_a)$. We assume that every players' goal is to choose a path P that minimizes the combined cost $\ell_P(f) + \alpha \tau_P(f)$, where $\tau_P(f) := \sum_{a \in P} \tau_a(f_a)$. For notational convenience, we assume that α is normalized to 1. This is without loss of generality because we can always divide all toll functions by α.

The tolls $\tau = (\tau_a)_{a \in A}$ are called θ-*restricted* if for every arc $a \in A$, $0 \leq \tau_a(x) \leq \theta_a(x)$ for all flow values $x \geq 0$. We define $\mathcal{T}(\theta)$ as the set of all θ-restricted tolls, i.e.,

$$\mathcal{T}(\theta) := \{(\tau_a)_{a \in A} \mid \forall a \in A \ : \ 0 \leq \tau_a(x) \leq \theta_a(x) \ \forall x \geq 0\}.$$

Given θ-restricted tolls τ, let f^τ denote a Nash flow that is induced by τ, i.e., f^τ is a Nash flow with respect to $\phi = \ell + \tau$. The *efficiency* of θ-restricted tolls for a given instance of the

restricted network toll problem is defined as

$$\min_{\tau \in \mathcal{T}(\theta)} \frac{C(f^{\tau})}{C(f^*)}. \tag{2}$$

That is, we relate the cost of the best Nash flow f^{τ} that is inducible by θ-restricted tolls τ to the cost of an optimal flow. Note that we account for the average latency of the network here rather than the total disutility (latency plus toll) of the players. The reason for that is that we are interested in characterizing the effect of tolls on the performance (measured in terms of average latency) of the network.

Given the restrictions $\theta = (\theta_a)_{a \in A}$ on the arcs, θ-restricted tolls τ are *optimal* if the Nash flow f^{τ} induced by τ satisfies $C(f^{\tau}) \leq C(f^{\bar{\tau}})$ for all Nash flows $f^{\bar{\tau}}$ induced by θ-restricted tolls $\bar{\tau}$. Similarly, θ-restricted tolls τ are λ-*approximate* for some $\lambda \geq 1$ if $C(f^{\tau}) \leq \lambda C(f^{\bar{\tau}})$ for all Nash flows $f^{\bar{\tau}}$ induced by θ-restricted tolls $\bar{\tau}$.

3 Computing optimal θ-restricted tolls

We first give a characterization of the flows that are inducible by θ-restricted tolls for single-commodity networks. This characterization will be the key to derive an algorithm that computes optimal θ-restricted tolls for parallel-arc networks. All results presented in this section hold for flow-dependent threshold functions θ.

3.1 Characterization of inducible flows for single-commodity networks

We consider the problem of determining whether a given flow f is inducible by θ-restricted tolls. We focus on the single-commodity case. As we will see, this problem reduces to verifying whether there is a negative cycle in a properly constructed graph.

Suppose we are given a flow f. Recall that f is a Nash flow with respect to $\ell + \tau$ iff for every two s,t-paths $P, P' \in \mathcal{P}$ with $f_P > 0$ it holds $\ell_P(f) + \tau_P \leq \ell_{P'}(f) + \tau_{P'}$. Said differently, every flow-carrying path must be a shortest path with respect to the combined cost $\phi := \ell + \tau$. Subsequently, let ℓ_a, τ_a and θ_a refer to $\ell_a(f_a)$, $\tau_a(f_a)$ and $\theta_a(f_a)$, respectively. (In the discussion below, several definitions will depend on the flow f; however, for notational convenience we often do not state this dependence explicitly.)

We use the following alternative characterization of Nash flows (see, e.g., [15]). For every vertex $u \in V$, let δ_u be the length of a shortest path from s to u with respect to $\ell + \tau$. Define A^+ as the set of arcs with positive flow, i.e., $A^+ := \{a \in A : f_a > 0\}$. Then f is a Nash flow with respect to $\phi = \ell + \tau$ if and only if (i) $\delta_v \leq \delta_u + \ell_a + \tau_a$ for every arc $a = (u,v) \in A$, and (ii) $\delta_v = \delta_u + \ell_a + \tau_a$ for every arc $a = (u,v) \in A^+$.

We can thus express the set $\mathcal{F}(\theta)$ of θ-restricted tolls that induce f as follows:

$$\mathcal{F}(\theta) := \{(\tau_a)_{a \in A} \mid \begin{array}{ll} \delta_v - \delta_u \leq \ell_a + \tau_a & \forall a = (u,v) \in A \setminus A^+ \\ \delta_v - \delta_u = \ell_a + \tau_a & \forall a = (u,v) \in A^+ \\ \delta_u \text{ free} & \forall u \in V \\ 0 \leq \tau_a \leq \theta_a & \forall a \in A \}. \end{array} \tag{3}$$

Note that the $(\delta_u)_{u \in V}$ are unrestricted in this formulation. Alternatively, we could have required that $\delta_s = 0$ and $\delta_u \geq 0$ for every $u \in V$. However, this is equivalent to the formulation (3) stated above.

We define a graph $\hat{G} = \hat{G}(f) = (V, \hat{A})$ with arc-costs $c : \hat{A} \to \mathbb{R}$ as follows: $\hat{G}$ contains all arcs $a = (u, v) \in A$ and, additionally, for every arc $a = (u, v) \in A^+$ the reversed arc (v, u). We call the former type of arcs *forward arcs* and the latter type of arcs *backward arcs*. The cost of each forward arc $a = (u, v) \in \hat{A}$ is equal to $c_a := \ell_a + \theta_a$. Every backward arc $a = (v, u) \in \hat{A}$ has a cost equal to the negative of the latency of its reversed arc $(u, v) \in A$, i.e., $c_a := -\ell_{(u,v)}$.

Given some subset X of arcs and functions $(g_a)_{a \in X}$, we define $g(X)$ as a short for $\sum_{a \in X} g_a$.

Theorem 1. *Let f be an arbitrary feasible flow. Then f is inducible by θ-restricted tolls if and only if $\hat{G}(f)$ does not contain a cycle of negative cost.*

Proof. Suppose $\hat{G} = \hat{G}(f)$ contains a cycle $C \subseteq \hat{A}$ of negative cost. Since only backward arcs have negative cost, at least one backward arc is part of C. Partition C into the set F of forward arcs and the set B of backward arcs, respectively. Let $\bar{B}$ denote the set of reversed arcs in B. Note that $\bar{B} \subseteq A^+$. We have

$$c(C) = c(F) + c(B) = \ell(F) + \theta(F) - \ell(\bar{B}) < 0. \tag{4}$$

Suppose for the sake of contradiction that $\tau = (\tau_a)_{a \in A} \in \mathcal{F}(\theta)$ are feasible tolls that induce f. By the feasibility of τ, we have for every forward arc $a = (u, v) \in F$, $\delta_v - \delta_u \leq \ell_{(u,v)} + \tau_{(u,v)}$ and for every backward arc $a = (u, v) \in B$, $\delta_u - \delta_v = \ell_{(v,u)} + \tau_{(v,u)}$, or equivalently, $\delta_v - \delta_u = -\ell_{(v,u)} - \tau_{(v,u)}$. Summing over all arcs in C, we obtain:

$$
\begin{aligned}
0 = \sum_{(u,v) \in C} \delta_v - \delta_u &= \sum_{(u,v) \in F} \delta_v - \delta_u + \sum_{(u,v) \in B} \delta_v - \delta_u \\
&\leq \sum_{(u,v) \in F} \ell_{(u,v)} + \tau_{(u,v)} - \sum_{(u,v) \in B} \ell_{(v,u)} + \tau_{(v,u)} \\
&\leq \ell(F) + \theta(F) - \ell(\bar{B}) - \tau(\bar{B}) < -\tau(\bar{B}),
\end{aligned}
$$

where the last inequality follows from (4). Thus $\tau(\bar{B}) < 0$ which is a contradiction since $\tau_a \geq 0$ for every arc $a \in A$.

Next suppose that $\hat{G}$ does not contain a negative cycle. We can then determine the shortest path distance δ_u from s to every node $u \in V$ in $\hat{G}$ with respect to c. (These distances are well-defined because $\hat{G}$ does not contain a negative cycle.) Note that for every arc $a = (u, v) \in \hat{A}$ we have $\delta_v \leq \delta_u + c_{(u,v)}$. Based on these distances, we extract tolls $\tau := (\tau_a)_{a \in A}$ as follows: For every arc $a = (u, v) \in A$, we define

$$\tau_a := \max\{0, \delta_v - \delta_u - \ell_a\}. \tag{5}$$

We show that τ induces f. By definition, we have for every arc $a = (u, v) \in A$: $\delta_v - \delta_u - \tau_a \leq \ell_a$. Consider an arc $a = (u, v) \in A^+$. Then $\delta_u - \delta_v \leq -\ell_a$, or equivalently, $\delta_v - \delta_u - \ell_a \geq 0$. Thus, $\delta_v - \delta_u - \tau_a = \ell_a$. Clearly, $\tau_a \geq 0$ for every $a \in A$. Moreover, for every arc $a = (u, v) \in A$ we have $\delta_v - \delta_u \leq \ell_a + \theta_a$ and thus $\delta_v - \delta_u - \ell_a \leq \theta_a$. We can infer that $\tau_a \leq \theta_a$ for every $a = (u, v) \in A$. $\square$

Note that the proof of the theorem also provides a way to extract the respective tolls if f is inducible by θ-restricted tolls: Given f, we compute the shortest path distance δ_u with respect to c from s to u for every $u \in V$ and define the toll τ_a for every arc $a = (u, v) \in A$ as in (5).

The following corollary is an immediate consequence of the above theorem and the fact that negative cycles can be detected efficiently (e.g., by the Bellman-Ford algorithm).

Corollary 1. *Given a flow f, we can determine in polynomial time whether f is inducible by θ-restricted tolls.*

3.2 Computing optimal tolls in paralle-arc networks

In light of the above characterization, the problem of computing θ-restricted tolls such that the cost $C(f^{\tau})$ of the induced Nash flow f^{τ} is minimized is equivalent to the problem of computing a minimum cost flow f such that $\hat{G}(f)$ does not contain a negative cost cycle. Once we have determined f, we can extract the optimal θ-restricted tolls τ as defined in (5). This equivalence constitutes the basis of our algorithm to compute optimal tolls in parallel-arc networks.

Let $G = (V,A)$ be a parallel-arc network and let f be a feasible flow. The condition of Theorem 1 then reduces to the following property: f is inducible by θ-restricted tolls if and only if

$$\forall a \in A, \ f_a > 0: \qquad \ell_a(f_a) \le \ell_{a'}(f_{a'}) + \theta_{a'}(f_{a'}) \quad \forall a' \in A. \tag{6}$$

Note that these conditions are similar to the Nash flow conditions in (1) (specialized to parallel-arc networks) with the difference that we allow some additional slack $\theta_{a'}(f_{a'})$ on the right-hand side. Thus, our goal is to determine a minimum cost flow f among all flows that satisfy (6).

Corollary 2. *The problem of computing optimal θ-restricted tolls for the parallel-arc restricted network toll problem is equivalent to computing a minimum cost flow f satisfying* (6).

Computing a minimum cost flow can be done efficiently by solving a convex program. However, here we need to ensure (6) additionally and it is a-priori not clear how to encode these constraints. Note that for Nash flows the corresponding conditions are ensured by applying the Karush-Kuhn-Tucker conditions to a convex program with an appropriately chosen objective function. A similar approach does not work here because we cannot deliberately choose an objective function and because of the asymmetry in (6) (due to the slack).

Our approach exploits the following key insight. Fix some minimum cost flow f^* satisfying (6) and suppose we knew the minimum value $z = \min\{\ell_a(0) + \theta_a(0) \mid a \in A, \ f_a^* = 0\}$ among all zero-flow arcs in f^*. Let $z = \infty$ if all arcs have positive flow in f^*. We can then compute an minimum cost flow $f^z = (f_a^z)_{a \in A}$ satisfying (6) as follows. From (6) we infer that $f_a^z = 0$ for every arc $a \in A$ with $\ell_a(0) > z$. Let $A^z = \{a \in A \mid \ell_a(0) \le z\}$ be the remaining arcs. On the arcs in A^z, we compute a feasible flow $(f_a^z)_{a \in A^z}$ of minimum cost satisfying $\ell_a(f_a^z) \le z$ for every $a \in A^z$ and $\ell_a(f_a^z) \le \ell_{a'}(f_{a'}^z) + \theta_{a'}(f_{a'}^z)$ for every $a,a' \in A^z$. The latter can be done by solving the program:

$$
\begin{aligned}
C^z = \min \ & \textstyle\sum_{a \in A^z} f_a^z \ell_a(f_a^z) \\
\text{s.t.} \ & \textstyle\sum_{a \in A^z} f_a^z = r \\
& f_a^z \ge 0 && \forall a \in A^z \\
& \ell_a(f_a^z) \le z && \forall a \in A^z \\
& \ell_a(f_a^z) \le \ell_{a'}(f_{a'}^z) + \theta_{a'}(f_{a'}^z) && \forall a,a' \in A^z.
\end{aligned}
\tag{7}
$$

The only remaining problem is that we do not know z. However, because there are at most $|A| + 1$ different possibilities (including the case $z = \infty$), we can simply compute a flow f^z for each possible value z and finally return the best flow f that has been encountered. The complete algorithm is summarized in Algorithm 1.

Theorem 2. *Algorithm 1 computes a minimum cost flow f satisfying* (6).

Proof. Let $f = f^z$ be the flow returned by Algorithm 1. Clearly, f is a feasible flow by construction. We argue that f satisfies (6). Consider some $a \in A^+$. Note that $f_{a'} = 0$ for every arc $a' \notin A^z$ and thus $a \in A^z$. Because $(f_a)_{a \in A^z}$ is a feasible solution to (7), we have

---

Algorithm 1: Algorithm to compute minimum cost flow satisfying (6)

1   Let $Z = \{\ell_a(0) + \theta_a(0) \mid a \in A\}$.
2   **for** *every $z \in Z \cup \{\infty\}$* **do**
3   Define $A^z := \{a \in A \mid \ell_a(0) \leq z\}$.
4   Set $f_a^z = 0$ for every $a \notin A^z$.
5   Let $(f_a^z)_{a \in A^z}$ be an optimal solution of cost C^z to the program in (7).
       (Remark: $(f_a^z)_{a \in A^z}$ is undefined and $C^z = \infty$ if (7) is infeasible.)
6   **end**
7   Return $f = f^z$ with C^z minimum among all $z \in Z \cup \{\infty\}$.

$\ell_a(f_a) \leq z \leq \ell_{a'}(0) = \ell_{a'}(f_{a'})$ for every $a' \notin A^z$. Moreover, $\ell_a(f_a) \leq \ell_{a'}(f_{a'}) + \theta_{a'}(f_{a'})$ for every $a' \in A^z$. Thus, f satisfies (6).

Let f^* be an optimal flow. We show that $C(f) \leq C(f^*)$. Define z as the minimum value $\ell_a(f_a^*) + \theta_a(f_a^*)$ of a zero-flow arc $a \in A$, i.e., $z = \min\{\ell_a(0) + \theta_a(0) \mid a \in A, \ f_a^* = 0\}$. Let $z = \infty$ if all arcs have positive flow. Note that $f_a^* = 0$ for every $a \notin A^z$ and thus $C(f^*) = \sum_{a \in A^z} f_a^* \ell_a(f_a^*)$. Observe that $(f_a^*)_{a \in A^z}$ is a feasible solution for the program in (7) with respect to z. Thus $C(f^z) \leq C(f^*)$. Because $C(f) \leq C(f^z)$, this concludes the proof. $\qquad\square$

Finally, observe that the program in (7) is convex if all latency and threshold functions are affine, i.e., of the form $q_1 x + q_0$ with $q_1, q_0 \geq 0$. In particular, the constraints of (7) are linear and the objective function is convex quadratic in this case, so the program can be solved exactly in polynomial time [13].

Corollary 3. *Algorithm 1 computes a minimum cost flow f satisfying* (6) *in polynomial time if all latency and threshold functions are affine.*

Hoefer et al. [10] derived a similar result for the special case that $\theta_a \in \{0, \infty\}$ for every arc $a \in A$.

4   General efficiency of θ-restricted tolls

We provide bounds on the efficiency of θ-restricted tolls for multi-commodity networks with polynomial latency functions of degree p. Our approach is algorithmic: We show how to compute θ-restricted tolls for a given instance of the restricted network toll problem that guarantee the claimed efficiency bound. The results given in this section hold for dynamic threshold functions.

Let $\mathcal{L}_p$ be defined as the set of all polynomial functions g of the form $g(x) = \sum_{d=0}^{p} q_d x^d$ with non-negative coefficients q_d, $d = 0, \ldots, p$. Moreover, let $\mathcal{M}_d$ refer to the set of all monomial functions of the form $\ell_a(x) = q_d x^d$ with non-negative coefficient q_d. Suppose we are given an arc $a \in A$ with $\ell_a \in \mathcal{L}_p$. We can replace a by a sequence of $p + 1$ arcs with latency functions in $\mathcal{M}_p, \ldots, \mathcal{M}_0$, respectively, in the obvious way. We can therefore assume without loss of generality that all latency functions $(\ell_a)_{a \in A}$ of the given instance are monomials. Subsequently, we use $d(a)$ to refer to the degree of the (monomial) latency function ℓ_a of $a \in A$, i.e., $d(a) = d$ iff $\ell_a \in \mathcal{M}_d$.

The basic idea is very simple. We define toll functions $(\tau_a)_{a \in A}$ as follows:

$$\tau_a(x) = \min\{x \cdot \ell_a'(x), \ \theta_a(x)\}. \tag{8}$$

That is, on each arc $a \in A$, we impose marginal cost tolls $x \cdot \ell'_a(x)$ if this does not exceed the threshold $\theta_a(x)$ and otherwise charge the maximum possible toll $\theta_a(x)$. Clearly, these tolls are θ-restricted. Note that these tolls are dynamic. We show at the end of the section how to derive static tolls that achieve the same efficiency.

Let $\phi := (\phi_a)_{a \in A}$ be the combined cost, i.e., for every $a \in A$, $\phi_a(x) := \ell_a(x) + \tau_a(x)$ for every $x \geq 0$, and let $f = f^\tau$ be a Nash flow with respect to ϕ. We next derive a bound on the ratio $C(f)/C(f^*)$, where f^* is an optimal flow. We adapt the (λ, μ)-smoothness approach [16] which was previously used successfully to bound the price of anarchy of network routing games [2, 8] and in a more general context in [16].

Because f is a Nash flow with respect to ϕ, it satisfies the following *variational inequality*, i.e., for every feasible flow x

$$\sum_{a \in A} \phi_a(f_a) f_a \leq \sum_{a \in A} \phi_a(f_a) x_a. \tag{9}$$

By the definition of ϕ,

$$\begin{aligned} C(f) = \sum_{a \in A} \ell_a(f_a) f_a &\leq \sum_{a \in A} \ell_a(f_a) x_a + \tau_a(f_a)(x_a - f_a) \\ &\leq \sum_{a \in A} \omega(\ell_a, \lambda) \ell_a(f_a) f_a + \lambda \ell_a(x_a) x_a, \end{aligned} \tag{10}$$

where we define

$$\omega(\ell_a, \lambda) := \sup_{f_a, x_a \geq 0} \frac{(\ell_a(f_a) + \tau_a(f_a) - \lambda \ell_a(x_a)) x_a - \tau_a(f_a) f_a}{\ell_a(f_a) f_a}.$$

We assume by convention that $0/0 = 0$. Finally, let $\omega(\lambda) := \sup_{a \in A} \omega(\ell_a, \lambda)$. With this definition, (10) implies

$$C(f) \leq \omega(\lambda) C(f) + \lambda C(x). \tag{11}$$

Because $\omega(\lambda)$ depends on λ, let Λ refer to the values of λ such that $\omega(\lambda) < 1$. Then for every $\lambda \in \Lambda$, (11) is equivalent to

$$C(f) \leq \lambda (1 - \omega(\lambda))^{-1} C(x). \tag{12}$$

The goal is to find $\lambda \in \Lambda$ that provides the best upper bound.

Lemma 1. *Let $\ell_a \in \mathcal{M}_d$ and define $\varepsilon_a := \tau_a(f_a)/\ell_a(f_a)$. We have*

$$\omega(\ell_a, \lambda) = \left(\frac{d(1 + \varepsilon_a)}{d + 1} \right) \left(\frac{1 + \varepsilon_a}{(d+1)\lambda} \right)^{1/d} - \varepsilon_a.$$

Moreover, $\omega(\ell_a, \lambda) < 1$ for

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

Proof. Define $\mu := x_a/f_a$. Then

$$\begin{aligned} \omega(\ell_a, \lambda) &= \sup_{f_a, x_a \geq 0} \frac{(\ell_a(f_a) + \tau_a(f_a) - \lambda \ell_a(\mu f_a)) \mu f_a - \tau_a(f_a) f_a}{\ell_a(f_a) f_a} \\ &= \sup_{\mu \geq 0} (1 - \lambda \mu^d) \mu - (1 - \mu) \varepsilon_a. \end{aligned}$$

9

The last expression attains its maximum at

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

Substituting yields the claim. Note that the restriction $\omega(\ell_a, \lambda) < 1$ implies that

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

$\square$

We continue to study the values for $\omega(\ell_a, \lambda)$ and λ. Observe that for every arc $a \in A$ with $\ell_a \in \mathcal{M}_d$ there are two possibilities for $\varepsilon_a = \tau_a(f_a)/\ell_a(f_a)$: If $\tau_a(f_a) = f_a \cdot \ell'_a(f_a)$ then $\varepsilon_a = d$; otherwise, $\tau_a(f_a) = \theta_a(f_a) < f_a \cdot \ell'_a(f_a)$ and thus $\varepsilon_a = \theta_a(f_a)/\ell_a(f_a) < d$.

We thus obtain

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

Choosing $\lambda = 1$ therefore satisfies the restrictions imposed on λ in the above lemma (and is tight for $d = 0$). Subsequently, we fix $\lambda := 1$.

We need to derive an upper bound on $\omega(\ell_a, 1)$:

$$\omega(\ell_a, 1) = \left(\frac{d(1+\varepsilon_a)}{d+1} \right) \left(\frac{1+\varepsilon_a}{d+1} \right)^{1/d} - \varepsilon_a.$$

Note that $\omega(\ell_a, 1)$ decreases as ε_a increases. This motivates the following definitions:

$$\bar{\varepsilon}_d = \min\{\varepsilon_a \mid a \in A, \ell_a \in \mathcal{M}_d\}$$

and

$$\omega(d, 1) = d \left(\frac{1+\bar{\varepsilon}_d}{1+d} \right)^{1+1/d} - \bar{\varepsilon}_d. \tag{13}$$

With these definitions, we obtain

$$\omega(1) = \max_{d=0,\ldots,p} \omega(d, 1). \tag{14}$$

Corollary 4. *Suppose $\bar{\varepsilon}_d = d$. Then $\omega(d, 1) = 0$.*

Observe that if we have $\bar{\varepsilon}_d = d$ for every $d = 0, \ldots, p$ then the above corollary in combination with (12) implies that $C(f) \leq C(x)$ (which actually follows readily from the observation that in this case marginal cost tolls are θ-restricted and induce an optimal flow).

We obtain the following theorem.

Theorem 3. *Given an instance of the restricted network toll problem with polynomial latency functions of degree p, the efficiency of θ-restricted tolls as defined in (8) is no worse than*

$$\max_{d=0,\ldots,p} \left((1+\bar{\varepsilon}_d) \left(1 - \frac{d}{d+1} \left(\frac{1+\bar{\varepsilon}_d}{d+1} \right)^{1/d} \right) \right)^{-1}$$

10

Proof. The proof follows from (12) with $\lambda = 1$ ($\lambda \in \Lambda$ as argued above) and the definition of $\omega(1)$ in (14). $\qquad\square$

We give some interpretation of the above theorem. Our result suggests that it is more important to impose large tolls on arcs with high degree latency functions than on the ones with low degree functions. As an example, consider the following extreme situation: Suppose the restrictions $(\theta_a)_{a \in A}$ are such that we can impose marginal cost tolls on all arcs $a \in A$ with latency functions of degree larger than t, and no tolls on all other arcs. The above bound then proves that the tolls in (8) achieve an efficiency no worse than the price of anarchy for degree t polynomials (see [17]), i.e.,

$$\left(1 - \frac{t}{t+1}\left(\frac{1}{t+1}\right)^{1/t}\right)^{-1}.$$

We next show that the bound in Theorem 3 is tight.

Theorem 4. *For every p and every choice of δ with $0 \leq \delta \leq p$ there is a parallel-arc instance of the restricted network toll problem with polynomial latency functions of degree p such that the efficiency of the tolls defined in (8) is equal to*

$$\left((1+\delta)\left(1 - \frac{p}{p+1}\left(\frac{1+\delta}{p+1}\right)^{1/p}\right)\right)^{-1}$$

Proof. Consider a network consisting of two parallel arcs $a, \bar{a}$ connecting s and t. Suppose we want to route $r = 1$ units of flow from s to t. Let the respective latency functions be $\ell_a(x) = 1$ and $\ell_{\bar{a}}(x) = x^p$. Define threshold functions $\theta_a(x) = 0$ and $\theta_{\bar{a}}(x) = \delta\ell_{\bar{a}}(x)$. Let the tolls $(\tau_a)_{a \in A}$ be defined as in (8). Note that $\bar{\varepsilon}_p = \delta$.

Consider a Nash flow $f = f^\tau$ induced by τ. If $\bar{\varepsilon}_p \geq p$ then f is a Nash flow with respect to marginal cost functions and thus an optimal flow. The claim follows.

Otherwise $\bar{\varepsilon}_p < p$. It is not hard to verify that $f_{\bar{a}} = (\bar{\varepsilon}_p + 1)^{-1/p}$ and $f_a = 1 - f_{\bar{a}}$. The total cost of f is

$$C(f) = \left(\frac{1}{\bar{\varepsilon}_p + 1}\right)^{1+1/p} + 1 - \left(\frac{1}{\bar{\varepsilon}_p + 1}\right)^{1/p} = 1 - \frac{\bar{\varepsilon}_p}{\bar{\varepsilon}_p + 1}\left(\frac{1}{\bar{\varepsilon}_p + 1}\right)^{1/p}.$$

Consider an optimal flow f^*. We have $f_{\bar{a}}^* = (p+1)^{-1/p}$ and $f_a^* = 1 - f_{\bar{a}}^*$. The total cost of f^* is

$$C(f^*) = \left(\frac{1}{p+1}\right)^{1+1/p} + 1 - \left(\frac{1}{p+1}\right)^{1/p} = 1 - \frac{p}{p+1}\left(\frac{1}{p+1}\right)^{1/p}.$$

Taking the ratio $C(f)/C(f^*)$ yields the claim. $\qquad\square$

The next corollary characterizes the efficiency of θ-restricted tolls with $\theta_a(x) = \varepsilon\ell_a(x)$ for every arc $a \in A$ and a fixed $\varepsilon \geq 0$. Intuitively, such restrictions reflect the requirement that the toll on each arc does not exceed an ε-fraction of the travel time along that arc.

Corollary 5. *Given an instance of the restricted network toll problem with polynomial latency functions of degree p and threshold functions of the form $\theta_a(x) = \varepsilon\ell_a(x)$, the efficiency of θ-restricted tolls as defined in (8) is no worse than 1 if $\varepsilon \geq p$ and no worse than*

$$\left((1+\varepsilon)\left(1 - \frac{p}{p+1}\left(\frac{1+\varepsilon}{p+1}\right)^{1/p}\right)\right)^{-1}$$

11

otherwise.

Proof. If $\varepsilon \geq p$ we have $\bar{\varepsilon}_d = d$ for every $d = 0, \ldots, p$ and thus $C(f) \leq C(x)$. (This result actually follows readily from the observation that in this case marginal cost tolls are ε-restricted and induce an optimal flow.) Otherwise, $\varepsilon < p$. Observe that for $d \leq \varepsilon$, $\bar{\varepsilon}_d = d$ and thus $\omega(d, 1) = 0$ by Corollary 4. For $\varepsilon < d \leq p$, $\bar{\varepsilon}_d = \varepsilon$. Note that $\omega(d, 1)$ increases with d (for fixed $\varepsilon_d = \varepsilon$). Thus, in this case

$$\omega(1) = p \left(\frac{1+\varepsilon}{1+p} \right)^{1+1/p} - \varepsilon.$$

Substituting yields the claimed bound. □

This bound matches exactly the *price of stability* of ε-Nash flows shown by Christodoulou, Koutsoupias and Spirakis [3].

Computing static tolls: The tolls defined in (8) are dynamic. It is not hard to see, that we can also define static tolls achieving the same efficiency: Simply define the dynamic tolls τ as in (8) and compute a Nash flow $f = f^\tau$ with respect to the combined cost functions $\phi = \ell + \tau$. Note that this can be done by solving the following convex program:

$$
\begin{aligned}
\min \ & \sum_{a \in A} \int_0^{f_a} \phi_a(x) dx \\
\text{s.t.} \ & \sum_{P \in \mathcal{P}_i} f_P = r_i && \forall i \in [k] \\
& f_a = \sum_{i \in [k]} \sum_{P \in \mathcal{P}_i : a \in P} f_P && \forall a \in A \\
& f_P \geq 0 && \forall P \in \mathcal{P}_i, \ \forall i \in [k].
\end{aligned}
\tag{15}
$$

In order to being able to solve this program we need that each function $h_a(y) = \int_0^y \phi_a(x) dx$ is continuously differentiable and convex (see, e.g., [17]). This is the case because $\phi_a(x) = \ell_a(x) + \tau_a(x)$ is continuous and non-decreasing. Once f has been computed, we can extract the respective static tolls inducing f by defining $\tau_a = \tau_a(f_a)$ for every arc $a \in A$.

Computing approximate tolls: Observe that our results also yield an approach to compute θ-restricted tolls that are λ-approximate, where λ is the efficiency guarantee stated in Theorem 3: Simply define the dynamic tolls τ as in (8) and compute a Nash flow $f = f^\tau$ with respect to the combined cost functions $\phi = \ell + \tau$ and the respective static tolls τ as described above. By Theorem 3, the Nash flow f^τ induced by τ satisfies

$$C(f^\tau) \leq \lambda C(f^*) \leq \lambda C(f^{\bar{\tau}}),$$

for every Nash flow $f^{\bar{\tau}}$ induced by θ-restricted tolls $\bar{\tau}$. Thus, τ is λ-approximate.

Corollary 6. *There is an algorithm that computes in polynomial time λ-approximate θ-restricted tolls for the multi-commodity network toll problem with monomial latency functions, where λ is the efficiency guarantee stated in Theorem 3.*

References

1. M. Beckmann, B. McGuire, and C. Winsten. *Studies in the Economics of Transportation*. Yale University Press, New Haven, 1956.
2. V. Bonifaci, T. Harks, and G. Schäfer. Stackelberg routing in arbitrary networks. *Mathematics of Operations Research*, 35(2):1–17, 2010.
3. G. Christodoulou, E. Koutsoupias, and P. G. Spirakis. On the performance of approximate equilibria in congestion games. In *Proc. 17th European Symposium on Algorithms*, pages 251–262, 2009.
4. R. Cole, Y. Dodis, and T. Roughgarden. Pricing network edges for heterogeneous selfish users. In *Proc. 35th Symp. on Theory of Computing*, pages 521–530, 2003.
5. R. Cole, Y. Dodis, and T. Roughgarden. How much can taxes help selfish routing? *Journal of Computer and System Sciences*, 72(3):444–467, 2006.
6. L. Fleischer. Linear tolls suffice: New bounds and algorithms for tolls in single source networks. *Theoretical Computer Science*, 348(2-3):217–225, 2005.
7. L. Fleischer, K. Jain, and M. Mahdian. Tolls for heterogeneous selfish users in multicommodity networks and generalized congestion games. In *Proc. 45th Symp. on Foundations of Computer Science*, pages 277–285, 2004.
8. T. Harks. Stackelberg strategies and collusion in network games with splittable flow. In *Proc. 6th Workshop on Approximation and Online Algorithms*, pages 133–146, 2008.
9. T. Harks, G. Schäfer, and M. Sieg. Computing flow-inducing network tolls. Technical Report 36-2008, Institut für Mathematik, Technische Universität Berlin, Germany, 2008.
10. M. Hoefer, L. Olbrich, and A. Skopalik. Taxing subnetworks. In *Proc. 4th Workshop on Internet and Network Economics*, pages 286–294, 2008.
11. G. Karakostas and S. G. Kolliopoulos. Edge pricing of multicommodity networks for heterogeneous selfish users. In *Proc. of 45th Symp. on Foundations of Computer Science*, pages 268–276, 2004.
12. E. Koutsoupias and C. H. Papadimitriou. Worst-case equilibria. In *Proc. 16th Symp. on Theoretical Aspects of Computer Science*, pages 404–413, 1999.
13. M. K. Kozlov, S. P. Tarasov, and L. G. Khachiyan. The polynomial solvability of convex quadratic programming. *USSR Computational Mathematics and Mathematical Physics*, 20(5):223–228, 1980.
14. T. Roughgarden. The price of anarchy is independent of the network topology. *J. Comput. Syst. Sci.*, 67(2):341–364, 2003.
15. T. Roughgarden. On the severity of Braess's paradox: Designing networks for selfish users is hard. *J. Comput. Syst. Sci.*, 72(5):922–953, 2006.
16. T. Roughgarden. Intrinsic robustness of the price of anarchy. In *Proc. 41st ACM Symposium on Theory of Computing*, pages 513–522, 2009.
17. T. Roughgarden and É. Tardos. How bad is selfish routing? *Journal of the ACM*, 49(2):236–259, 2002.
18. C. Swamy. The effectiveness of Stackelberg strategies and tolls for network congestion games. In *Proc. 18th Symp. on Discrete Algorithms*, 2007.