Course: Combinatorial Optimization

# A Glimpse of Algorithmic Game Theory: Selfish Routing

VU University Amsterdam
Spring 2013

Prof. dr. Guido Schäfer

Centre for Mathematics and Computer Science (CWI)
Networks and Optimization Group
Science Park 123, 1098 XG Amsterdam, The Netherlands

VU University Amsterdam
Department of Econometrics and Operations Research
De Boelelaan 1105, 1081 HV Amsterdam, The Netherlands

Website: http://www.cwi.nl/~schaefer
Email: g.schaefer@cwi.nl

Document last modified:
March 6, 2013

***Disclaimer***: Note that the lecture notes have undergone some rough proof-reading only. Please feel free to report any typos, mistakes, inconsistencies, etc. that you observe by sending me an email (g.schaefer@cwi.nl).

# Contents

# 1 Selfish Routing

We consider network routing problems in which users choose their routes so as to minimize their own travel time. Our main focus will be to study the inefficiency of Nash equilibria and to identify effective means to decrease the inefficiency caused by selfish behavior.

## 1.1 Motivating Example: Pigou Example

We first consider an example:

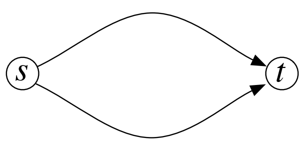


Figure 1: Pigou instance

**Example 1.1** (Pigou's example)**.** Consider the parallel-arc network in Figure 1. For every arc $a$, we have a latency function $\ell_a : \mathbb{R}_+ \to \mathbb{R}_+$, representing the load-dependent travel time or latency for traversing this arc. In the above example, we have for the upper arc $\ell_a(x) = 1$, i.e., the latency is one independently of the amount of flow on that arc. The lower arc has latency function $\ell_a(x) = x$, i.e., the latency grows linearly with the amount of flow on that arc. Suppose we want to send one unit of flow from $s$ to $t$ and that this one unit of flow corresponds to infinitely many users that want to travel from $s$ to $t$.

Every selfish user will reason as follows: The latency of the upper arc is one (independently of the flow) while the latency of the lower arc is at most one (and even strictly less than one if some users are not using this arc). Thus, every user chooses the lower arc. The resulting flow is a Nash flow. Since every user experiences a latency of one, the total average latency of this Nash flow is one.

We next compute an optimal flow that minimizes the total average latency of the users. Assume we send $p \in [0,1]$ units of flow along the lower arc and $1-p$ units of flow along the upper arc. The total average latency is $(1-p) \cdot 1 + p \cdot p = 1 - p + p^2$. This function is minimized for $p = \frac{1}{2}$. Thus, the optimal flow sends one-half units of flow along the upper and one-half units of flow along the lower arc. Its total average latency is $\frac{3}{4}$.

This example shows that selfish user behavior may lead to outcomes that are inefficient: The resulting Nash flow is suboptimal with a total average latency that is $\frac{4}{3}$ times larger than the total average latency of an optimal flow. This raises the following natural questions: How large can this inefficiency ratio be in general networks? Does it depend on the topology of the network?

## 1.2 Model

We formalize the setting introduced above. An instance of a *selfish routing game* is given as follows:

- directed graph $G=(V,A)$ with vertex set $V$ and arc set $A$;
- nondecreasing and continuous latency function $\ell_a : \mathbb{R}_+ \to \mathbb{R}_+$ for every arc $a \in A$.
- set of $k$ commodities $[k] := \{1,\dots,k\}$, specifying for each commodity $i \in [k]$ a source vertex $s_i$ and a target vertex $t_i$;
- for each commodity $i \in [k]$, a demand $r_i > 0$ that represents the amount of flow that has to be sent from $s_i$ to $t_i$;

We use $(G,r,\ell)$ to refer to an instance for short.

Let $\mathscr{P}_i$ be the set of all simple paths from $s_i$ to $t_i$ in $G$ and let $\mathscr{P} := \cup_i \mathscr{P}_i$. A *flow* is a function $f : \mathscr{P} \to \mathbb{R}_+$. The flow $f$ is *feasible* (with respect to $r$) if for all $i \in [k]$, $\sum_{P \in \mathscr{P}_i} f_P = r_i$, i.e., the total flow send from $s_i$ to $t_i$ meets the demand $r_i$. For a given flow $f$, we define the aggregated flow on arc $a \in A$ as $f_a := \sum_{P \in \mathscr{P}: a \in P} f_P$.

The *total travel time* of a path $P \in \mathscr{P}$ with respect to $f$ is defined as the sum of the latencies of the arcs on that path:

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

We assess the overall quality of a given flow $f$ by means of a global cost function $C$. Though there are potentially many different cost functions that one may want to consider (depending on the application), we focus on the *total average latency* as cost function here.

**Definition 1.1.** The *total cost* of a flow $f$ is defined as:

$$C(f) := \sum_{P \in \mathscr{P}} \ell_P(f) f_P. \tag{1}$$

Note that the total cost can equivalently be expressed as the sum of the average latencies on the arcs:

$$C(f) = \sum_{P \in \mathscr{P}} \ell_P(f) f_P = \sum_{P \in \mathscr{P}} \Big( \sum_{a \in P} \ell_a(f_a) \Big) f_P = \sum_{a \in A} \Big( \sum_{P \in \mathscr{P}: a \in P} f_P \Big) \ell_a(f_a) = \sum_{a \in A} \ell_a(f_a) f_a.$$

## 1.3 Nash Flows and their Existence

The basic viewpoint that we adopt here is that players act selfishly in that they attempt to minimize their own individual travel time. A standard solution concept to predict outcomes of selfish behavior is the one of an equilibrium outcome in which no player has an incentive to unilaterally deviate from its current strategy. In the context of nonatomic selfish routing games, this viewpoint translates to the following definition:

**Definition 1.2.** A feasible flow $f$ for the instance $(G,r,\ell)$ is a *Nash flow* if for every commodity $i \in [k]$ and two paths $P,\ Q \in \mathscr{P}_i$ with $f_P > 0$ and for every $\delta \in (0, f_P]$, we have $\ell_P(f) \leq \ell_Q(\tilde{f})$, where

$$\tilde{f}_P := \begin{cases} f_P - \delta & \text{if } P = P \\ f_P + \delta & \text{if } P = Q \\ f_P & \text{otherwise.} \end{cases}$$

Intuitively, the above definition states that for every commodity $i \in [k]$, shifting $\delta \in (0, f_P]$ units of flow from a flow carrying path $P \in \mathscr{P}_i$ to an arbitrary path $Q \in \mathscr{P}_i$ does not lead to a smaller latency.

A similar concept was introduced by Wardrop (1952) in his first principle: A flow for the nonatomic selfish routing game is a *Wardrop equilibrium* if for every source-target pair the latencies of the used routes are less than or equal to those of the unused routes.

**Definition 1.3.** A feasible flow $f$ for the instance $(G,r,\ell)$ is a *Wardrop equilibrium* (or *Wardrop flow*) if

$$\forall i \in [k],\ \forall P,\ Q \in \mathscr{P}_i,\ f_P > 0: \quad \ell_P(f) \leq \ell_Q(f). \tag{2}$$

For $\delta \to 0$ the definition of a Nash flow corresponds to the one of a Wardrop flow. Subsequently, we use the Wardrop flow definition; we slightly abuse naming here and will also refer to such flows as Nash flows.

**Corollary 1.1.** *Let $f$ be a Nash flow for $(G,r,\ell)$ and define for every $i \in [k]$, $c_i(f) := \min_{P \in \mathscr{P}_i} \ell_P(f)$. Then $\ell_P(f) = c_i(f)$ for every $P \in \mathscr{P}_i$ with $f_P > 0$.*

*Proof.* By the definition of $c_i(f)$, we have that for every $P \in \mathscr{P}_i$: $\ell_P(f) \geq c_i(f)$. Using (2), we conclude that for every $P \in \mathscr{P}_i$ with $f_P > 0$: $\ell_P(f) \leq c_i(f)$. ☐

Note that the above corollary states that for each commodity all flow carrying paths have the same latency and all other paths cannot have a smaller latency. The flow carrying paths are thus shortest paths with respect to the total latency.

We next argue that Nash flows always exist and that their cost is unique. In order to do so, we use a powerful result from convex optimization. Consider the following program (CP):

$$\begin{array}{llcll} \min & \sum_{a \in A} h_a(f_a) & & & \\ \text{s.t.} & \sum_{P \in \mathscr{P}_i} f_P & = & r_i & \forall i \in [k] \\ & f_a & = & \sum_{P \in \mathscr{P}: a \in P} f_P & \forall a \in A \\ & f_P & \geq & 0 & \forall P \in \mathscr{P}. \end{array}$$

Note that the set of all feasible solutions for (CP) corresponds exactly to the set of all flows that are feasible for our selfish routing instance $(G,r,\ell)$. The above program is a linear program if the functions $(h_a)_{a\in A}$ are linear. (CP) is a convex program if the functions $(h_a)_{a\in A}$ are convex. A convex program can be solved efficiently by using, e.g., the ellipsoid method. The following is a fundamental theorem in convex (or, more generally, non-linear) optimization:

**Theorem 1.1** (Karush–Kuhn–Tucker (KKT) Optimality Conditions)**.** *Consider the program (CP) with continuously differentiable and convex functions $(h_a)_{a\in A}$. A feasible flow $f$ is an optimal solution for (CP) if and only if*

$$\forall i \in [k],\ \forall P,\ Q \in \mathscr{P}_i,\ f_P > 0: \quad h'_P(f) := \sum_{a\in P} h'_a(f_a) \leq \sum_{a\in Q} h'_a(f_a) =: h'_Q(f), \qquad (3)$$

*where $h'_a(x)$ refers to the first derivative of $h_a(x)$.*

Observe that (3) is very similar to the Wardrop equilibrium conditions (2). In fact, these two conditions coincide if we define for every $a \in A$:

$$h_a(f_a) := \int_0^{f_a} \ell_a(x)dx. \qquad (4)$$

**Corollary 1.2.** *Let $(G,r,\ell)$ be a selfish routing instance with nondecreasing and continuous latency functions $(\ell_a)_{a\in A}$. A feasible flow $f$ is a Nash flow if and only if it is an optimal solution to (CP) with functions $(h_a)_{a\in A}$ as defined in (4).*

*Proof.* For every arc $a \in A$, the function $h_a$ is convex (since $\ell_a$ is nondecreasing) and continuously differentiable (since $\ell_a$ is continuous). The proof now follows from Theorem 1.1. □

We will also need the following theorem:

**Theorem 1.2** (Extreme Value Theorem)**.** *Let $X$ be a compact set and $f : X \to \mathbb{R}$ a continuous function. Then $f$ attains both a maximum and a minimum on $X$.*

**Corollary 1.3.** *Let $(G,r,\ell)$ be a selfish routing instance with nondecreasing and continuous latency functions $(\ell_a)_{a\in A}$. Then a Nash flow $f$ always exists. Moreover, its cost $C(f)$ is unique.*

*Proof.* The set of all feasible flows for (CP) is compact (closed and bounded). Moreover, the objective function of (CP) with (4) is continuous (since $\ell_a$ is continuous for every $a \in A$). Thus, the minimum of (CP) must exist (by the Extreme Value Theorem). Since the objective function of (CP) is convex, the optimal value of (CP) is unique. It is not hard to conclude that the cost $C(f)$ of a Nash flow is unique. □

Note that, in particular, the above observations imply that we can compute a Nash flow for a given nonatomic selfish routing instance $(G,r,\ell)$ efficiently by solving the convex program (CP) with (4).

## 1.4 Optimal Flows

We define an optimal flow as follows:

**Definition 1.4.** A feasible flow $f^*$ for the instance $(G,r,\ell)$ is an *optimal flow* if $C(f^*) \leq C(x)$ for every feasible flow $x$.

The set of optimal flows corresponds to the set of all optimal solutions to (CP) if we define for every arc $a \in A$:

$$h_a(f_a) := \ell_a(f_a) f_a. \tag{5}$$

Since the cost function $C$ is continuous (because $\ell_a$ is continuous for every $a \in A$), we conclude that an optimal flow always exists (again using the Extreme Value Theorem). Moreover, we will assume that $h_a$ is convex and continuously differentiable for each arc $a \in A$; latency functions $(\ell_a)_{a \in A}$ that satisfy these conditions are called *standard*. Using Theorem 1.1, we obtain the following characterization of optimal flows:

**Corollary 1.4.** *Let the latency functions $(\ell_a)_{a\in A}$ be standard. A feasible flow $f^*$ for the instance $(G,r,\ell)$ is an optimal flow if and only if:*

$$\forall i \in [k],\ \forall P,\ Q \in \mathscr{P}_i,\ f_P^* > 0: \quad \sum_{a\in P} \ell_a(f_a^*) + \ell_a'(f_a^*) f_a^* \leq \sum_{a \in Q} \ell_a(f_a^*) + \ell_a'(f_a^*) f_a^*.$$

That is, an optimal flow is a Nash flow with respect to so-called *marginal latency functions* $(\ell_a^*)_{a\in A}$, which are defined as

$$\ell_a^*(x) := \ell_a(x) + \ell_a'(x) x.$$

## 1.5 Price of Anarchy

We study the inefficiency of Nash flows in comparison to an optimal flow. A common measure of the inefficiency of equilibrium outcomes is the *price of anarchy*.

**Definition 1.5.** Let $(G,r,\ell)$ be an instance of the selfish routing game and let $f$ and $f^*$ be a Nash flow and an optimal flow, respectively. The *price of anarchy* $\rho(G,r,\ell)$ of the instance $(G,r,\ell)$ is defined as:

$$\rho(G,r,\ell) = \frac{C(f)}{C(f^*)}. \tag{6}$$

(Note that (6) is well-defined since the cost of Nash flows is unique.) The price of anarchy of a set of instances $\mathscr{I}$ is defined as

$$\rho(\mathscr{I}) = \sup_{(G,r,\ell)\in\mathscr{I}} \rho(G,r,\ell).$$

## 1.6 Upper Bounds on the Price of Anarchy

Subsequently, we derive upper bounds on the price of anarchy for selfish routing games. The following variational inequality will turn out to be very useful.

**Lemma 1.1** (Variational inequality)**.** *A feasible flow $f$ for the instance $(G,r,\ell)$ is a Nash flow if and only if it satisfies that for every feasible flow $x$:*

$$\sum_{a\in A} \ell_a(f_a)(f_a - x_a) \leq 0. \tag{7}$$

*Proof.* Given a flow $f$ satisfying (7), we first show that condition (2) of Definition 1.3 holds. Let $P,\ Q \in \mathscr{P}_i$ be two paths for some commodity $i \in [k]$ such that $\delta := f_P > 0$. Define a flow $x$ as follows:

$$x_a := \begin{cases} f_a & \text{if } a \in P\cap Q \text{ or } a \notin P\cup Q \\ f_a - \delta & \text{if } a \in P \\ f_a + \delta & \text{if } a \in Q. \end{cases}$$

By construction $x$ is feasible. Hence, from (7) we obtain:

$$\sum_{a\in A} \ell_a(f_a)(f_a - x_a) = \sum_{a\in P} \ell_a(f_a)(f_a - (f_a - \delta)) + \sum_{a\in Q} \ell_a(f_a)(f_a - (f_a + \delta)) \leq 0.$$

We divide the inequality by $\delta > 0$, which yields the Wardrop conditions (2).

Now assume that $f$ is a Nash flow. By Corollary 1.1, we have for every $i \in [k]$ and $P \in \mathscr{P}_i$ with $f_P > 0$: $\ell_P(f) = c_i(f)$. Furthermore, for $Q \in \mathscr{P}_i$ with $f_Q = 0$, we have $\ell_Q(f) \geq c_i(f)$. It follows that for every feasible flow $x$:

$$\begin{aligned} \sum_{a\in A} \ell_a(f_a) f_a &= \sum_{i\in[k]} \sum_{P\in\mathscr{P}_i} c_i(f) f_P = \sum_{i\in[k]} c_i(f) \left(\sum_{P\in\mathscr{P}_i} f_P\right) = \sum_{i\in[k]} c_i(f) \left(\sum_{P\in\mathscr{P}_i} x_P\right) \\ &= \sum_{i\in[k]} \sum_{P\in\mathscr{P}_i} c_i(f) x_P \leq \sum_{i\in[k]} \sum_{P\in\mathscr{P}_i} \ell_P(f) x_P = \sum_{a\in A} \ell_a(f_a) x_a. \end{aligned}$$

□

We derive an upper bound on the price of anarchy for affine linear latency functions with nonnegative coefficients:

$$\mathscr{L}_1 := \{g : \mathbb{R}_+ \to \mathbb{R}_+ \ : \ g(x) = q_1 x + q_0 \text{ with } q_0, q_1 \in \mathbb{R}_+\}.$$

**Theorem 1.3.** *Let $(G,r,\ell)$ be an instance of a nonatomic routing game with affine linear latency functions $(\ell_a)_{a\in A} \in \mathscr{L}_1^A$. The price of anarchy $\rho(G,r,\ell)$ is at most $\frac{4}{3}$.*

*Proof.* Let $f$ be a Nash flow and let $x$ be an arbitrary feasible flow for $(G,r,\ell)$. Using

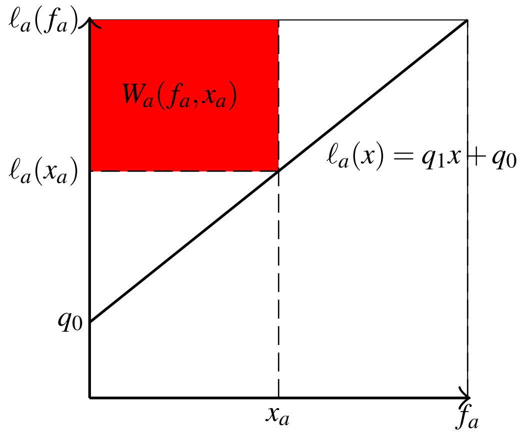


Figure 2: Illustration of the worst case ratio of $W_a(f_a,x_a)$ and $\ell_a(f_a)\, f_a$.

the variational inequality (7), we obtain

$$\begin{aligned} C(f) &= \sum_{a\in A} \ell_a(f_a) f_a \le \sum_{a\in A} \ell_a(f_a) x_a = \sum_{a\in A} \ell_a(f_a) x_a + \ell_a(x_a) x_a - \ell_a(x_a) x_a \\ &= \sum_{a\in A} \ell_a(x_a) x_a + \underbrace{[\ell_a(f_a) - \ell_a(x_a)] x_a}_{=:W_a(f_a,x_a)} = \sum_{a\in A} \ell_a(x_a) x_a + \sum_{a\in A} W_a(f_a,x_a). \end{aligned}$$

We next bound the function $W_a(f_a,x_a)$ in terms of $\omega \cdot \ell_a(f_a) f_a$ for some $0 \le \omega < 1$, where

$$\omega := \max_{f_a,x_a \ge 0} \frac{(\ell_a(f_a) - \ell_a(x_a))x_a}{\ell_a(f_a) f_a} = \max_{f_a,x_a \ge 0} \frac{W_a(f_a,x_a)}{\ell_a(f_a) f_a}.$$

Note that for $x_a \ge f_a$ we have $\omega \le 0$ (because latency functions are non-decreasing). Hence, we can assume $x_a \le f_a$. See Figure 2 for a geometric interpretation. Since latency functions are affine linear, $\omega$ is upper bounded by $\frac{1}{4}$. We obtain

$$C(f) \le C(x) + \sum_{a\in A} \frac{1}{4} \ell_a(f_a) f_a = C(x) + \frac{1}{4} C(f).$$

Rearranging terms and letting $x$ be an optimal flow concludes the proof. □

We can extend the above proof to more general classes of latency functions. For the latency function $\ell_a$ of an arc $a \in A$, define

$$\omega(\ell_a) := \sup_{f_a,x_a \ge 0} \frac{(\ell_a(f_a) - \ell_a(x_a))x_a}{\ell_a(f_a)\, f_a}. \tag{8}$$

We assume by convention $0/0 = 0$. See Figure 3 for a graphical illustration of this value. For a given class $\mathscr{L}$ of non-decreasing latency functions, we define

$$\omega(\mathscr{L}) := \sup_{\ell_a \in \mathscr{L}} \omega(\ell_a).$$

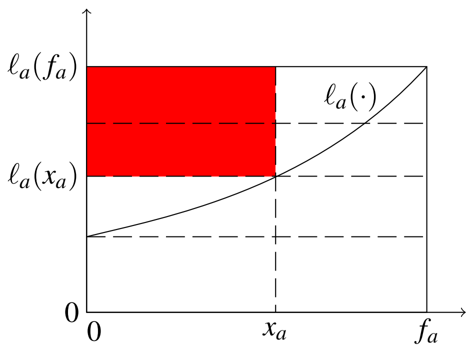


Figure 3: Illustration of $\omega(\ell_a)$.

**Theorem 1.4.** *Let $(G,r,\ell)$ be an instance of the nonatomic selfish routing game with latency functions $(\ell_a)_{a\in A} \in \mathscr{L}^A$. Let $0 \le \omega(\mathscr{L}) < 1$ be defined as above. The price of anarchy $\rho(G,r,\ell)$ is at most $(1-\omega(\mathscr{L}))^{-1}$.*

*Proof.* Let $f$ be a Nash flow and let $x$ be an arbitrary feasible flow. We have

$$\begin{aligned} C(f) &= \sum_{a\in A} \ell_a(f_a) f_a \le \sum_{a\in A} \ell_a(f_a)x_a = \sum_{a\in A} \ell_a(f_a)x_a + \ell_a(x_a)x_a - \ell_a(x_a)x_a \\ &= \sum_{a\in A} \ell_a(x_a)x_a + [\ell_a(f_a) - \ell_a(x_a)]x_a \le C(x) + \omega(\mathscr{L})C(f). \end{aligned}$$

Here, the first inequality follows from the variational inequality (7). The last inequality follows from the definition of $\omega(\mathscr{L})$. Since $\omega(\mathscr{L}) < 1$, the claim follows. □

In general, we define $\mathscr{L}_d$ as the set of latency functions $g : \mathbb{R}_+ \to \mathbb{R}_+$ that satisfy

$$g(\mu x) \ge \mu^d g(x) \quad \forall \mu \in [0,1].$$

Note that $\mathscr{L}_d$ contains polynomial latency functions with nonnegative coefficients and degree at most $d$.

**Lemma 1.2.** *Consider latency functions in $\mathscr{L}_d$. Then*

$$\omega(\mathscr{L}_d) \le \frac{d}{(d+1)^{(d+1)/d}}.$$

*Proof.* Recall the definition of $\omega(\ell_a)$:

$$\omega(\ell_a) = \sup_{f_a, x_a \ge 0} \frac{\big(\ell_a(f_a) - \ell_a(x_a)\big)x_a}{\ell_a(f_a)\, f_a}. \tag{9}$$

We can assume that $x_a \le f_a$ since otherwise $\omega(\ell_a) \le 0$. Let $\mu := \frac{x_a}{f_a} \in [0,1]$. Then

$$\omega(\ell_a) = \max_{\mu\in[0,1], f_a \ge 0} \left( \frac{\big(\ell_a(f_a) - \ell_a(\mu f_a)\big)\mu f_a}{\ell_a(f_a) f_a} \right) \le \max_{\mu\in[0,1], f_a \ge 0} \left( \frac{\big(\ell_a(f_a) - \mu^d \ell_a(f_a)\big)\mu f_a}{\ell_a(f_a) f_a} \right)$$

| $d$ | 1 | 2 | 3 | ... |
|---|---|---|---|---|
| $\rho(G,r,\ell)$ | $\approx 1.333$ | $\approx 1.626$ | $\approx 1.896$ | |

Table 1: The price of anarchy for polynomial latency functions of degree $d$.

$$= \max_{\mu\in[0,1]} (1-\mu^d)\mu. \tag{10}$$

Here, the first inequality holds since $\ell_a \in \mathscr{L}_d$. Since this is a strictly convex program, the unique global optimum is given by

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

Replacing $\mu^*$ in (10) yields the claim. □

**Theorem 1.5.** *Let $(G,r,\ell)$ be an instance of a nonatomic routing game with latency functions $(\ell_a)_{a\in A} \in \mathscr{L}_d^A$. The price of anarchy $\rho(G,r,\ell)$ is at most*

$$\rho(G,r,\ell) \le \left(1 - \frac{d}{(d+1)^{(d+1)/d}}\right)^{-1}.$$

*Proof.* The theorem follows immediately from Theorem 1.4 and Lemma 1.2. □

The price of anarchy for polynomial latency functions with nonnegative coefficients and degree $d$ is given in Table 1 for small values of $d$.

## 1.7 Lower Bounds on the Price of Anarchy

We can show that the bound that we have derived in the previous section is actually tight.

**Theorem 1.6.** *Consider nonatomic selfish routing games with latency functions in $\mathscr{L}_d$. There exist instances such that the price of anarchy is at least*

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

*Proof.* See assignments. □

## 1.8 Motivating Example: Braess's Paradox

**Example 1.2** (Braess's paradox)**.** Consider the network in Figure 4 (left). Assume that we want to send one unit of flow from $s$ to $t$. It is not hard to verify that the Nash flow splits evenly and sends one-half units of flow along the upper and lower arc, respectively. This flow is also optimal having a total average latency of $\frac{3}{2}$.

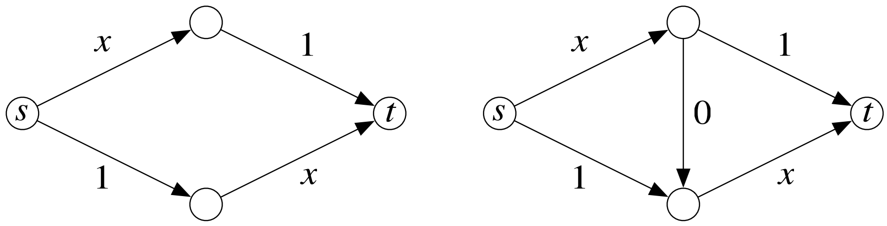


Figure 4: Braess Paradox

Now, suppose there is a global authority that wants to improve the overall traffic situation by building new roads. The network in Figure 4 (right) depicts an augmented network where an additional arc with constant latency zero has been introduced. How do selfish users react to this change? What happens is that every user chooses the zig-zag path, first traversing the upper left arc, then the newly introduced zero latency arc and then the lower right arc. The resulting Nash flow has a total average latency of 2.

The Braess Paradox shows that extending the network infrastructure does not necessarily lead to an improvement with respect to the total average latency if users choose their routes selfishly. In the above case, the total average latency degrades by a factor of $\frac{4}{3}$. In general, one may ask the following questions: How large can this degradation be? Can we develop efficient methods to detect such phenomena?

### 1.9 Detecting Braess's Paradox

Suppose we are given a single-commodity instance $(G,r,\ell)$ of the nonatomic selfish routing game. Let $f$ be a Nash flow for $(G,r,\ell)$ and define $d(G,r,\ell) := c_1(f)$ as the common latency of all flow-carrying paths (see Corollary 1.1). We study the following optimization problem: Given $(G,r,\ell)$, find a subgraph $H \subseteq G$ that minimizes $d(H,r,\ell)$. We call this problem the NETWORK DESIGN problem.

**Corollary 1.5.** *Let $(G,r,\ell)$ be a single-commodity instance of the nonatomic selfish routing game with linear latency functions. Then for every subgragph $H \subseteq G$:*

$$d(G,r,\ell) \leq \frac{4}{3} d(H,r,\ell).$$

*Proof.* Let $h$ and $f$ be the Nash flows for the instances $(H,r,\ell)$ and $(G,r,\ell)$, respectively. By Corollary 1.1, the latency of every flow-carrying path in a Nash flow is equal. Thus, the costs of the Nash flows $f$ and $h$, respectively, are $rd(G,r,\ell)$ and $rd(H,r,\ell)$. Using that $h$ is a feasible flow for $(G,r,\ell)$ and the upper bound of $4/3$ on the price of anarchy for linear latencies, we obtain

$$C(f) = rd(G,r,\ell) \leq \frac{4}{3} C(h) = \frac{4}{3} rd(H,r,\ell).$$

□

We can generalize the above proof to obtain:

**Corollary 1.6.** *Let $(G,r,\ell)$ be a single-commodity instance of the nonatomic selfish routing game with polynomial latency functions in $\mathscr{L}_d$. Then for every subgraph $H \subseteq G$:*

$$d(G,r,\ell) \leq \left(1 - \frac{d}{(d+1)^{(d+1)/d}}\right)^{-1} d(H,r,\ell).$$

We next turn to designing approximation algorithms that compute a "good" subgraph $H$ of $G$ with a provable approximation guarantee. We review some basics from computational theory first. Readers that are familiar with this topic can continue with Section **??**.

A trivial approximation algorithm (called TRIVIAL subsequently) for the NETWORK DESIGN problem is to simply return the original graph as a solution. Using the above corollaries, it follows that TRIVIAL has an approximation guarantee of

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

for latency functions in $\mathscr{L}_d$.

We will show that the performance guarantee of TRIVIAL is best possible, unless $P = NP$.

**Theorem 1.7.** *Assuming $P \neq NP$, for every $\varepsilon > 0$ there is no $(\frac{4}{3} - \varepsilon)$-approximation algorithm for the* NETWORK DESIGN *problem.*

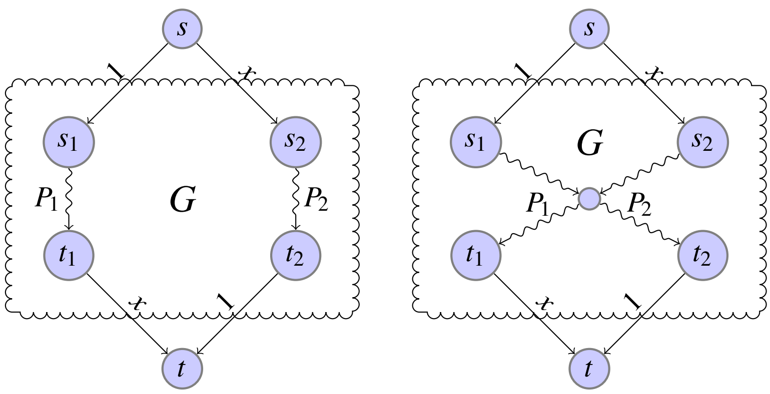


Figure 5: (a) Yes-instance of 2DDP. (b) No-instance of 2DDP.

*Proof.* We reduce from the *2-directed vertex-disjoint paths problem (*2DDP*)*, which is *NP*-complete. An instance of this problem is given by a directed graph $G = (V,A)$ and two vertex pairs $(s_1,t_1)$, $(s_2,t_2)$. The question is whether there exist a path $P_1$ from $s_1$ to $t_1$ and a path $P_2$ from $s_2$ to $t_2$ in $G$ such that $P_1$ and $P_2$ are vertex disjoint. We will show that a $(\frac{4}{3}-\varepsilon)$-approximation algorithm could be used to differentiate between yes- and no-instances of 2DDP in polynomial time.

Suppose we are given an instance $\mathscr{I}$ of 2DDP. We construct a graph $G'$ by adding a super source $s$ and a super sink $t$ to the network. We connect $s$ to $s_1$ and $s_2$ and $t_1$ and $t_2$ to $t$, respectively. The latency functions of the added arcs are given as indicated in Figure 1.9, where we assume that all latency functions in the original graph $G$ are set to zero. This can be done in polynomial time.

We will prove the following two statements:

(i) If $\mathscr{I}$ is a yes-instance of 2DDP then $d(H,1,\ell) = 3/2$ for some subgraph $H \subseteq G'$.

(ii) If $\mathscr{I}$ is a no-instance of 2DDP then $d(H,1,\ell) \geq 2$ for every subgraph $H \subseteq G'$.

Suppose for the sake of a contradiction that a $(\frac{4}{3}-\varepsilon)$-approximation algorithm ALG for the NETWORK DESIGN problem exists. ALG then computes in polynomial time a subnetwork $H \subseteq G'$ such that the cost of a Nash flow in $H$ is at most $(\frac{4}{3}-\varepsilon)\mathsf{opt}$, where $\mathsf{opt} = \min_{H\subseteq G'} d(H,r,\ell)$. That is, the cost of a Nash flow for the subnetwork $H$ computed by ALG is less than 2 for instances in (i) and it is at least 2 for instances in (ii). Thus, using ALG we can determine in polynomial time whether $\mathscr{I}$ is a yes- or no-instance, which is a contradiction to the assumption that $P \neq NP$. It remains to show the above two statements.

For (i), we simply delete all arcs in $G$ that are not contained in $P_1$ and $P_2$. Then, splitting the flow evenly along these paths yields a Nash equilibrium with cost $d(H,1,\ell) = 3/2$.

For (ii), we can assume without loss of generality that any subgraph $H$ contains an $s,t$-path. If $H$ has an $(s,s_2,t_1,t)$ path then routing the flow along this path yields a Nash flow with cost $d(H,1,\ell) = 2$. Suppose $H$ does not contain an $(s,s_2,t_1,t)$ path. Because $\mathscr{I}$ is a no-instance, we have three possibilities:

1. $H$ contains an $(s,s_1,t_1,t)$ path but no $(s,s_2,t_2,t)$ paths (otherwise two such paths must share a vertex and $H$ would contain an $(s,s_2,t_1,t)$ path);
2. $H$ contains an $(s,s_2,t_2,t)$ path but no $(s,s_1,t_1,t)$ path (otherwise two such paths must share a vertex and $H$ would contain an $(s,s_2,t_1,t)$ path);
3. every $s,t$-path in $H$ is an $(s,s_1,t_2,t)$ path.

It is not hard to verify that in either case, the cost of a Nash flow is $d(H,1,\ell) = 2$. □