

INSTITUT FÜR INFORMATIK

**The Price of Anarchy and  
Computation of Equilibria in  
Non-atomic Consumption-Relevance  
Congestion Games**

Lasse Kliemann

Bericht Nr. 0814

Dezember 2008

CHRISTIAN-ALBRECHTS-UNIVERSITÄT  
ZU KIEL

# The Price of Anarchy and Computation of Equilibria in Non-atomic Consumption-Relevance Congestion Games

**Keywords:** non-atomic congestion games, Wardrop model, price of anarchy, multicast routing, computation of equilibria.

Lasse Kliemann\*

lki@informatik.uni-kiel.de

December 2008

## Abstract

We present an extension to the model of non-atomic congestion games (NCG). NCGs enforce a symmetry between the consumption of elements (e.g., network links) and their relevance for the players: players utilizing an element  $e$  through a strategy  $S$  (e.g., a multicast tree) with rate of consumption  $C_{eS} > 0$  experience the element's latency amplified by that same factor  $C_{eS}$ . Our extension instead allows a factor  $R_{eS}$ , independent of  $C_{eS}$ , to express the amplification of the element's latency from the players' perspective, or, in other words, the *relevance* of element  $e$  for strategy  $S$ . We therefore call the extended model *non-atomic consumption-relevance congestion games* (NCRCG).

NCRCGs exhibit new phenomena, including multiple Nash equilibria of different social cost and – even from a worst-case point of view – a dependence of the price of anarchy on structural parameters not limited to the class of element latency functions used. It poses new computational challenges.

We prove almost tight lower, upper, and bicriteria bounds for the price of anarchy for super-homogeneous classes of element latency functions. We show a positive computational result for affine element latency functions. A summary of experimental results is given, which suggest that our lower bound is the best possible.

---

\*Supported by the Deutsche Forschungsgemeinschaft, Priority Program 1126, “Algorithmics of Large and Complex Networks”, Grant Sr7-3.

# Part I.

## Model and Known Results

### 1. Informal Introduction

We give an informal introduction to congestion games and the motivation behind the NCRCG model, as well as an overview of its new phenomena and our contribution. We refer to Sec. 2 for all mathematical definitions. Congestion games model competitive situations where individuals, called *players*, need to occupy elementary resources, called *elements*, in specific combinations in order to perform their tasks. The eligible combinations of elements are called *strategies*. For instance, elements can be links in a network, and strategies can be paths or trees that connect certain terminal nodes. A distribution of all players across available strategies is called an *action distribution*. Given an action distribution, then for each element, a quantity called the *congestion* captures how severely that element is stressed. The congestion depends on the amount of players utilizing the element and may also depend on the strategies through which this utilization takes place. The latter is expressed through element-strategy-dependent factors called *rates of consumption*, or, as we will prefer, *consumption numbers*. The congestion does *not* depend on any kind of ‘identity’ of the players.

Each of the elements performs its part of the task with a latency, and that latency is obtained by applying an element-dependent non-decreasing real function, called the *element latency function*, to the congestion. The latency experienced by a player depends on the latencies of the elements contained in the strategy chosen by that player. In the NCG model this dependence is again influenced by the elements’ consumption numbers: the latency experienced by a player using a specific strategy  $S$  is the weighted sum over the element latencies in  $S$ , where the elements’ consumption numbers are used as weights. In contrast, in the NCRCG model, the weights are given by factors called *relevance numbers*, which are *independent* of consumption numbers.

A notion of ‘total latency’, called the *social cost*, is used to measure the overall performance of the system under a given action distribution. Large interest exists in comparing the social cost and the structure of different particular kinds of action distributions, most notably those with optimal social cost and those which constitute Nash equilibria, i.e., action distributions in which no player has an incentive to change his currently chosen strategy. The ratio of the social cost of a worst-case Nash equilibrium and an optimal action distribution is called the *price of anarchy*. It is a measure of performance-loss due to non-cooperative behavior.

Congestion games can be used to model non-cooperative routing in networks. Network links take the role of elements, and paths or trees take the role of strategies. However, there are limitations as to what routing scenarios can be modeled by NCGs, and these limitations gave the original motivation to extend the model to NCRCG. Most notably, modeling multicast routing requires to express the role of a link inside a multicast tree, which may require the rate of consumption to differ from the relevance. For instance, a

link close to the source has high relevance for the players since it serves many terminals – compared to a link that only serves a few. However, if a duplication technique for realizing the multicast flow is in use, data has to be sent down this link only *once*, no matter how many terminals are served. So, the relevance of a link varies under different multicast trees, depending on where in the tree the link is located – whereas the consumption, i.e., how steep the congestion increases with the amount of players utilizing that link, thanks to the duplication technique shows no such dependence and might be the same for all multicast trees which contain the link.

Allowing consumption numbers to differ from relevance numbers leads to structural and quantitative changes. Whereas in the NCG model all Nash equilibria have the same social cost, the NCRCG model allows Nash equilibria with different social cost, and there can be a large gap between those. With only mild assumptions on the element latency functions, the social cost is a separable convex function in NCGs. Not so in NCRCGs, where the social cost function in general is not convex. On the quantitative side, NCRCGs can exhibit a much higher price of anarchy than NCGs, even if restricted to the same class of element latency functions. For an NCG with element latency function drawn from a class  $\mathcal{L}$ , it was proved by Roughgarden and Tardos [31] that – under some additional assumptions on  $\mathcal{L}$  – the price of anarchy is bounded by the *anarchy value*  $\alpha(\mathcal{L})$ , which only depends on  $\mathcal{L}$ , and is  $\Theta(\frac{p}{\ln p})$  for polynomials with non-negative coefficients and maximum degree  $p$ . An extension to that concept was later given by Correa, Schulz, and Stier Moses [11] using the  $\beta(\mathcal{L})$  parameter. For NCRCGs, these bounds no longer hold; even for a fixed class of element latency functions, the price of anarchy can be arbitrarily high.

This article is organized as follows. The rest of Part I gives all mathematical definitions of the NCRCG model (Sec. 2) and summarizes previous work (Sec. 3). Most importantly, Sec. 3.2 introduces the  $\beta$  parameter, which is needed later. Part II contains our new results, summarized in the following paragraphs.

**Our Contribution.** We identify a new scalar parameter  $\gamma$ , which captures parts of an NCRCG’s characteristics regarding the price of anarchy (Sec. 5). We establish bounds containing that new parameter along with the known  $\beta := \beta(\mathcal{L})$ . If element latency functions are polynomials of degree at most  $p$  and non-negative coefficients, the upper bounds (Sec. 8) are  $\gamma^{p+1}$  and  $\frac{1}{1-\gamma\beta}\gamma$ , depending on the range of  $\gamma$ . We show corresponding lower bounds for both ranges of  $\gamma$ , which are only one factor of  $\gamma$  away (Sec. 7).

We discuss different types of multicast and show how they can be modelled as NCRCGs (Sec. 6). We show that even with strictly increasing element latency functions, Nash equilibria in multicast are non-unique in general.

On the computational side, we show how mathematical programming can be used and point out cases that lead to *convex* programming (Sec. 10). We briefly present experimental results (Sec. 11) suggesting that our lower bounds are, in fact, also upper bounds. This poses the challenging open task to prove this experimentally gained conjecture.

## 2. The NCRCG Model

### 2.1. Instances

**Definition 1.** A continuous and non-decreasing function  $\ell : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$  is called an *element latency function*.

The following definition lists all items that characterize a particular instance in the NCRCG model. It is not a definition of game-theoretic concepts; these will be introduced in Sec. 2.3.

**Definition 2.** An instance in the non-atomic consumption-relevance congestion game model (called shortly “a non-atomic consumption-relevance congestions game”, or “an NCRCG”) is defined by the following items.

- (i) A number  $m \in \mathbb{N}$ . We denote  $E := [m]$  ( $= \{1, \dots, m\}$ ), and each number in  $E$  stands for an *element*.
- (ii) For each element  $e \in E$  an element latency function  $\ell_e$ .
- (iii) A number  $N \in \mathbb{N}$ . Each number in  $[N]$  stands for a *player class*.
- (iv) For each  $i \in [N]$  a number  $d_i \in \mathbb{R}_{> 0}$ , called the *demand* of player class  $i$ . The demand is to represent a large number of non-cooperative players. The game-theoretic concept behind this will be explained in Sec. 2.3.
- (v) For each  $i \in [N]$  a finite set  $\mathfrak{S}_i$ , chosen such that  $\mathfrak{S}_1, \dots, \mathfrak{S}_N$  are pairwise disjoint and each  $\mathfrak{S}_i$  has at least cardinality 2. We say that each element in  $\mathfrak{S}_i$  stands for a *strategy* for player class  $i$ . Denote  $\mathfrak{S} := \bigcup_{i \in [N]} \mathfrak{S}_i$  and  $n := |\mathfrak{S}|$ . We will often identify  $\mathfrak{S} = [n]$  to simplify notation.
- (vi) A matrix  $C \in \mathbb{R}_{\geq 0}^{m \times n}$  – its entries are called *consumption numbers* – and a matrix  $R \in \mathbb{R}_{\geq 0}^{m \times n}$  – its entries are called *relevance numbers*. We say that for  $e \in E$  and  $S \in \mathfrak{S}$  the number  $C_{eS}$  is the consumption of element  $e$  under strategy  $S$ , and the number  $R_{eS}$  is the relevance of element  $e$  for strategy  $S$ . We further demand that neither of the two matrices has a row or a column of only zeros, and that for all  $e \in E, S \in \mathfrak{S}$  it holds

$$C_{eS} = 0 \text{ if and only if } R_{eS} = 0 .$$

### 2.2. Action Distributions, Congestions, Latencies, Social Cost

Define the set of *action distributions* by

$$\mathcal{A} := \left\{ a \in \mathbb{R}_{\geq 0}^n; \sum_{S \in \mathfrak{S}_i} a_S = d_i \quad \forall i \in [N] \right\} .$$

An action distribution describes a way of how the demand  $d_i$  of each player class  $i$  is distributed across the strategies in  $\mathfrak{S}_i$ . All following notions are relative to a given

action distribution  $a$ . We define the *vector of congestions* as  $\vec{g}(a) := C \cdot a$ , where the dot denotes the usual matrix-vector product. Hence for each  $e \in E$  we have its congestion

$$g_e(a) = \sum_{S \in \mathfrak{S}} C_{eS} a_S .$$

The congestion  $g_e(a)$  of an element  $e$  is a measure of how heavily that element  $e$  is stressed. As can be noticed from the definition, the model allows that an element may be stressed differently by the same amount of players *depending on the strategy through which these players occupy the element*. Recall now that each element  $e$  has an element latency function  $\ell_e$ . This function tells us how the element reacts to congestion. We call  $\ell_e(g_e(a))$  the *element latency* of  $e$ .

We have to define how these element latencies are experienced by the players. The total of players in a player class  $i$  can be understood as the interval  $[0, d_i]$ . Each action distribution describes a possible way of how this interval can be partitioned and the parts assigned to strategies. All the players which by this are assigned to a strategy  $S$  experience the latency

$$L_S(a) := \sum_{e \in E} R_{eS} \ell_e(g_e(a)) .$$

We call this the *strategy latency* of  $S$ . The vector of all strategy latencies is denoted<sup>1</sup> by  $\vec{L}(a) := (L_S(a))_{S \in \mathfrak{S}}$ . The following two quantities capture overall characteristics of the system relative to two action distributions  $a$  and  $b$ , or just  $a$ , respectively. They are called *mixed social cost* and *social cost*:

$$\text{SC}^a(b) := \sum_{S \in \mathfrak{S}} L_S(a) b_S \quad \text{and} \quad \text{SC}(a) := \sum_{S \in \mathfrak{S}} L_S(a) a_S .$$

### 2.3. Game Theory and the Price of Anarchy

We study the non-atomic case of congestion games. That means that the decision of a ‘single player’ which strategy to take does not change the system (i.e., congestions, element latencies, or strategy latencies). We therefore do not have to distinguish players from each other but can subsume them to a real interval  $[0, d_i]$ , where  $d_i$  is a measure of how many players exist. In fact, we never rigorously defined what a ‘single player’ is.

An illustrative example for this is road traffic. The decision of a single driver which route to take will have *virtually no effect* on the congestions and latencies of the road network. However, each driver will only choose such routes that promise minimum latency. Hence, this system can be considered stable once that all drivers have chosen minimum-latency routes. Another example is a communication network with a large

---

<sup>1</sup>Using simply  $L(a)$  instead of  $\vec{L}(a)$  and referring to its  $S$ th component by  $L_S(a)$  would have been more systematic, but carries the risk for confusion that  $L$  might refer to the latency function of a single strategy – just as we sometimes denote a single element latency function by  $\ell$  and not by  $\ell_e$  for some  $e \in E$ . For the same reason, we write  $\vec{\ell}(a)$  for the vector of all element latencies. We also write  $\vec{g}(a)$  for the vector of all congestions.

number of users where each user has only got a small amount of data to send or to receive.

Action distributions in which no ‘single player’ has an incentive to switch strategies can now be very concisely characterized. The following is the standard definition for Nash equilibria in non-atomic games, also known as *Wardrop equilibria* [35].

**Definition 3.** Let an NCRCG  $\Gamma$  be given. An action distribution  $a$  with the following property is called a *Nash equilibrium* (abbreviated “N.E.”) for  $\Gamma$ :

$$\forall i \in [N] \quad \forall S, T \in \mathfrak{S}_i : \quad (a_S > 0 \implies L_S(a) \leq L_T(a)) .$$

The definition says that in a Nash equilibrium, only minimum-latency strategies are used. A deeper discussion of the non-atomic model itself can be found in [18].

Existence of Nash equilibria is guaranteed under our assumptions; see Thm. 2 in Sec. 3.1. Since  $\mathbf{SC}$  is continuous and  $\mathcal{A}$  is compact, optimal action distributions, i.e., with minimal social cost, exist. Their value is denoted by  $\mathbf{OPT}$ ; we always assume that  $\mathbf{OPT} > 0$ . We can now define the price of anarchy.

**Definition 4.** Given an NCRCG, its *price of anarchy* or *coordination ratio* is defined by  $\sup_{a \text{ is N.E.}} \mathbf{SC}(a)/\mathbf{OPT}$ . It is sometimes denoted by  $\rho$ . By continuity of  $L_S(\cdot)$ ,  $S \in \mathfrak{S}$ , we can replace the supremum by a maximum.

We have now defined the NCRCG model. The case  $C = R$  coincides with the previously studied NCG model [31]. The next subsection will make the model clearer. The reader might also want to take an excursion to Sec. 6 and Sec. 7 for more complex examples.

## 2.4. Modeling Unicast Routing

We show how non-cooperative unicast routing can be modeled as an NCRCG, in fact even as an NCG, since we will only need the case  $C = R$ . Let  $G = (V, E)$  be a directed multi-graph modeling a communication network with a continuous, non-decreasing latency function  $\ell_e : \mathbb{R}_{\geq 0} \longrightarrow \mathbb{R}_{\geq 0}$  for each edge  $e$  and  $N$  source-sink pairs  $(s_i, t_i)$ ,  $i = 1, \dots, N$ . Each of these  $N$  pairs has got a total demand of  $d_i$  units of flow to route from  $s_i$  to  $t_i$ , which represents a continuum  $[0, d_i]$  of players. The latency for transmitting one unit of flow along any path in the graph is the sum of the latencies of the edges in the path.

We can model this as an NCRCG in a straight-forward way. The edges of the graph together with their latency functions become the elements and element latency functions of the NCRCG, respectively. The  $N$  source-sink pairs correspond to  $N$  player classes. The set of strategies for each player class is a set of  $s_i$ - $t_i$ -paths and hence we can take the corresponding edge-path incidence matrix, which is a 0/1-valued matrix, as the consumption matrix  $C$  as well as the relevance matrix  $R$ . A Nash equilibrium now means that all flow travels along minimum-latency paths.

The set of *all*  $s_i$ - $t_i$ -paths can be exponentially large in the number of nodes  $|V|$ . If all or too large a subset of them shall be eligible for the users’ routing, the matrices  $C$  and  $R$  will be too large to be stored explicitly. This can be a problem for the computation of

optimal or Nash flows. For the scenario at hand, i.e., unicast routing, we can resort to a description on edge level, using flow conservation rules. The corresponding challenge for the general case is yet open.

### 3. Previous Work

#### 3.1. Existence of Nash Equilibria

The following characterization of Nash equilibria does not only allow a short argument for their existence, but will also be essential later when proving bounds on the price of anarchy.

**Theorem 1.** *An action distribution  $a$  is a Nash equilibrium if and only if*

$$\text{SC}(a) \leq \text{SC}^a(b) \quad \forall b \in \mathcal{A} . \quad (\text{VAR})$$

By the theory of variational inequalities, we get the existence guarantee.

**Theorem 2.** *Every NCRCG admits a Nash equilibrium.*

*Proof.* Equation (VAR) is a well-studied variational inequality, and we know that solutions exist since  $L_S(\cdot)$  is continuous for each  $S \in \mathfrak{S}$  and  $\mathcal{A}$  is compact and convex, see, e.g., [23, Thm. 1.4] or [16, Thm. 3.1].  $\square$

#### 3.2. Price of Anarchy: $\alpha$ and $\beta$ Parameters

The price of anarchy in NCGs, i.e., when  $C = R$ , was studied by Roughgarden and Tardos [31]. We briefly sketch one of the main results. An element latency function  $\ell$  is called *standard* if it is differentiable and  $x \mapsto \ell(x)x$  is convex. A class of element latency functions  $\mathcal{L}$  is called *standard* if each  $\ell \in \mathcal{L}$  is standard and  $\mathcal{L}$  contains at least one non-zero function.

Let  $\mathcal{L}$  be a standard class of element latency functions. For each  $\ell \in \mathcal{L}$  define the *marginal cost function*  $\ell^* : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}, x \mapsto \ell'(x)x + \ell(x)$ , which is the derivative of  $x \mapsto \ell(x)x$ . Then choose a function  $\mu_\ell : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$  such that  $\ell^*(\mu_\ell(v)) = \ell(v)$  and  $\mu(v) \leq v$  for all  $v \in \mathbb{R}_{\geq 0}$ . Such a function exists due to a simple argument using the given convexity of element latency functions. Now define, using the convention  $0/0 := 0$ , the *anarchy value* [31, Def. 4.3] of  $\mathcal{L}$  as

$$\alpha(\mathcal{L}) := \sup_{\substack{\ell \in \mathcal{L} \\ v \geq 0}} \frac{\ell(v)v}{(\ell(\mu_\ell(v)) - \ell(v))\mu_\ell(v) + \ell(v)v} . \quad (1)$$

It can be shown that this is a well-defined quantity, since it is independent of any particular choice for  $(\mu_\ell)_{\ell \in \mathcal{L}}$ . Roughgarden and Tardos [31] show that the price of anarchy in an NCG with element latency functions drawn from a standard class  $\mathcal{L}$  is no more than  $\alpha(\mathcal{L})$ . That bound is tight from a worst-case point of view.



Although the expression defining the anarchy value may look complicated at first, anarchy values for many interesting classes of functions can be derived relatively easily, see [30, Sec. 3.5.2]. For instance, let  $\text{Poly}^+(p)$  be the class of polynomials of degree at most  $p$  and non-negative coefficients. Then  $\alpha(\text{Poly}^+(p)) = (1 - p(p+1)^{-1 - \frac{1}{p}})^{-1} = \Theta(\frac{p}{\ln p})$ .

Correa, Schulz, and Stier Moses [11] show that the concept of anarchy value can be simplified and generalized at the same time. Let  $\mathcal{L}$  be a class of element latency functions, not necessarily standard. Using the convention  $0/0 := 0$ , define

$$\beta(\mathcal{L}) := \sup_{\substack{\ell \in \mathcal{L} \\ v \geq 0}} \frac{1}{\ell(v)v} \max_{x \geq 0} (\ell(v) - \ell(x))x . \quad (2)$$

Monotonicity ensures that there is no division by zero, except  $0/0$ , which is defined to 0. It follows that

$$(\ell(v) - \ell(x))x \leq \beta(\mathcal{L}) \ell(v)v \quad \forall v, x \in \mathbb{R}_{\geq 0} \quad \forall \ell \in \mathcal{L} . \quad (3)$$

It is easy to see that always  $\beta(\mathcal{L}) \leq 1$ . In case that  $\mathcal{L}$  happens to be standard and  $\alpha(\mathcal{L})$  is finite, we have  $\beta(\mathcal{L}) < 1$  and  $\frac{1}{1 - \beta(\mathcal{L})} = \alpha(\mathcal{L})$ . Moreover, for *any* class  $\mathcal{L}$  of element latency functions such that  $\beta(\mathcal{L}) < 1$ , it is shown in [11] that we can bound the price of anarchy of an NCG with element latency function drawn from  $\mathcal{L}$  by  $1/(1 - \beta(\mathcal{L}))$ . The  $\beta$  parameter is easy to derive for many classes of element latency functions, often easier than the anarchy value directly. We demonstrate that in Lem. 2 and Cor. 1.

### 3.3. Price of Anarchy: When $J\vec{L}$ is Positive (Semi-)Definite

Perakis [27] studies a setting in some respect more general than ours:  $\vec{L}$  may be any continuously differentiable function  $\mathbb{R}^n \rightarrow \mathbb{R}^n$ . On the other hand, she requires that the Jacobian  $J\vec{L}(a)$  is positive definite (for some results only positive semidefinite) for all  $a \in \mathcal{A}$ , and, if  $\vec{L}$  is non-affine, that a condition called the *Jacobian similarity property* is also given. Under these conditions, bounds on the price of anarchy are derived that are tight from a worst-case point of view. The condition on the definiteness of the Jacobian may or may not be given in the NCRCG model, and hence these bounds apply to it only in part. In [17] we compare these bounds experimentally with ours, where applicable.

### 3.4. Bicriteria Bounds

Bounds of this type have the following form: “a Nash equilibrium is (in terms of social cost) not worse than an optimum for  $t$  times the demand.” For NCGs and  $t = 2$  this was proved by Roughgarden and Tardos [31]. The statement was later refined to  $t = 1 + \beta(\mathcal{L})$  for NCGs with element latency functions from a class  $\mathcal{L}$ , see [12] for a short proof.

### 3.5. Uniqueness of Nash Equilibrium

In an NCG, all Nash equilibria have the same social cost. A simple proof works by exploiting the characterization in Thm. 1 to show that element latencies are unique under Nash equilibria and then to use Prop. 1 from page 11; for details see, e.g., [18].

### 3.6. Computation

If  $C = R$ , then Nash equilibria can be characterized as the optimal solutions to a convex program. This has been noted by Beckmann, McGuire, and Winsten [3], Dafermos and Sparrow [14], and Braess [7, 8]; a proof is also given in [18]. The proof is based on the Karush-Kuhn-Tucker theorem (“KKT theorem” for short) and seems to depend crucially on  $C = R$ . Still in the case  $C = R$ , a characterization of optima by a convex program is possible if the real functions  $x \mapsto \ell_e(x)x$  are convex for each  $e \in E$ , which is the case for element latency functions from  $\text{Poly}^+(p)$ .

### 3.7. Bibliographic Notes

We give a short and by far not exhaustive list of related work in roughly chronological order. Some have already been cited in the preceding outline of known results.

Pigou [28] in 1920 informally introduced what is nowadays known as “Pigou’s Example”. It is a simple unicast network exhibiting a high price of anarchy. See, e.g., [18] for a formal treatment. Nash [24] in 1951 introduced the concept of non-cooperative games and their equilibria in an atomic model. The term “Nash equilibrium” was subsequently also used in other models to describe configurations in which no player has an incentive to deviate from the chosen strategy. Wardrop [35] in 1952 studied a non-atomic model in the context of road traffic. He introduced two principles, which are the base of our understanding of Nash equilibria and optimal action distributions, respectively. We quote [35, p. 345]:

- (1) The journey times on all the routes actually used are equal, and less than those that would be experienced by a single vehicle on any unused route.
- (2) The average journey time is a minimum.

The characterization of Nash equilibria as optimal solutions to a convex program was noted by Beckmann, McGuire, and Winsten [3] in 1956 in the context of road traffic, Dafermos and Sparrow [14] in 1969, and Braess [7, 8] in 1968. Smith [33] in 1979 studied existence and uniqueness issues in the Wardrop model and gave the variational inequality formulation (VAR), which is frequently used in our work.

Rosenthal [29] in 1973 introduced congestion games in an atomic setting. Existence of equilibria was proved by the use of a potential function. Schmeidler [32] in 1973 introduced non-atomic games with an extended notion of action distributions.

In 1999, Koutsoupias and Papadimitriou in an atomic setting of routing in a simple network studied the quantity that would soon later be named the “price of anarchy” [26]. Their model, also known as the “KP model”, and variations of it were subsequently

addressed, among others, by Czumaj and Vöcking [13] in 2002, Lücking, Mavronicolas, Monien, Rode, Spirakis, and Vrto [21] in 2003, and Lücking, Mavronicolas, Monien, Rode [20] in 2004.

On the non-atomic side, the dissertation of Roughgarden [30] in 2002 and the work of Roughgarden and Tardos [31] introduced the anarchy value, which has shown to be a strong concept for bounding the price of anarchy. Subsequently, in 2004, that concept was generalized (and simplified) by Correa, Schulz, and Stier Moses [11]. They recently gave a nice graphical proof [12] for a bound on the price of anarchy. Chau and Sim [9] in 2003 and Perakis [27] in 2004 studied the price of anarchy in generalized settings.

Results on related non-atomic models were published by, among others, Friedman [15] in 1996, Blonski [5] in 1999, and Milchtaich [22] in 2004.

# Part II.

## Our Results

### 4. Preliminaries

This section collects some notions and simple results that will be used later. Most importantly, for action distributions  $a$  and  $b$ , we introduce two more global quantities, similar to the social cost, called *mixed total element cost* and *total element cost*:

$$\text{EC}^a(b) := \sum_{e \in E} \ell_e(g_e(a))g_e(b) \quad \text{and} \quad \text{EC}(a) := \sum_{e \in E} \ell_e(g_e(a))g_e(a) .$$

If  $C = R$ , then we have  $\text{EC}^a(b) = \text{SC}^a(b)$ , and so  $\text{EC}(a) = \text{SC}(a)$ , but not in the general case.

The next proposition says that Nash equilibria have a simple structure. Let

$$\Lambda_i(a) := \min\{L_S(a); S \in \mathfrak{S}_i\}$$

for  $a \in \mathcal{A}$  and  $i \in [N]$ . It follows directly from the definition:

**Proposition 1.** *An action distribution  $a$  is a Nash equilibrium if and only if for all  $i \in [N]$  and all  $S \in \mathfrak{S}_i$  we have that  $a_S > 0$  implies  $L_S(a) = \Lambda_i(a)$ . Moreover, the social cost of a Nash equilibrium  $a$  is*

$$\text{SC}(a) = \sum_{i \in [N]} \Lambda_i(a)d_i . \tag{4}$$

We conclude with some technical and notational conventions and remarks:

- The NCRCG model is feasibly defined and interesting even with one player class, i.e.,  $N = 1$ . That case is notationally simpler and the reader is encouraged to think of  $N = 1$  on a first read. In fact, all examples given in this article only have one player class.
- Let  $S \in \mathfrak{S}$ . The set of those elements that have non-zero entries in column  $S$  of matrix  $C$  (or, equivalently,  $R$ ), will – abusing notation – sometimes also be denoted by  $S$ , and we also speak of an element being ‘contained’ in  $S$ . Keep in mind that two strategies with different consumption or relevance numbers can constitute the same set of elements in this sense. This will be no problem; we will always treat such strategies as distinct. Moreover, it is sometimes convenient to consider the set of all strategies which in that sense ‘contain’ a particular element  $e$ . We will denote that as  $\mathfrak{S}(e) := \{S \in \mathfrak{S}; C_{e,S} \neq 0\}$ , which is the same as  $\{S \in \mathfrak{S}; R_{e,S} \neq 0\}$ .
- The congestion function  $\vec{g}$  is in fact (by a straight-forward extension) defined for all vectors from  $\mathbb{R}_{\geq 0}^n$  and not only for action distributions. Since element latency

functions are defined on  $\mathbb{R}_{\geq 0}$ , also the functions  $\vec{\ell}$ ,  $\vec{L}$ , EC, and SC are defined for vectors from  $\mathbb{R}_{\geq 0}^n$ . We will make use of that in a few places; in particular, we will use it later when scaling the demands and speaking of actions distributions w.r.t. those new demands.

## 5. The New Parameters $\gamma_1$ , $\gamma_2$ , and $\gamma$

We introduce new parameters for instances of the NCRCG model. They will play an important role in bounding the price of anarchy.

$$\gamma_1 := \max\left\{\frac{C_{eS}}{R_{eS}}; S \in \mathfrak{S}, e \in S\right\},$$

$$\gamma_2 := \max\left\{\frac{R_{eS}}{C_{eS}}; S \in \mathfrak{S}, e \in S\right\}.$$

Their product is denoted  $\gamma := \gamma_1\gamma_2$ .

Recall that we interpret  $S \in \mathfrak{S}$  as the set of elements which have non-zero consumption number in strategy  $S$ , or, equivalently, which have non-zero relevance number in strategy  $S$ . Hence there is no division by zero and so these parameters are well-defined and finite. We have  $\gamma_1 = \gamma_2 = 1$  if  $C = R$ , i.e., in the case of an NCG. We always have the product  $\gamma = \gamma_1\gamma_2 \geq 1$ .

**Scaling.** Let  $\varepsilon \in \mathbb{R}_{>0}$  and  $e \in E$ . If we scale the row of  $R$  corresponding to  $e$  with  $\varepsilon$  and then replace  $\ell_e$  by  $\tilde{\ell}_e(x) := \frac{1}{\varepsilon}\ell_e(x)$ , essentially nothing changes for the game. The functions  $\vec{L}$  and SC are unchanged, and hence so is the set of equilibria, optima, and their social costs. Also the  $\beta$  parameter does not change; if  $\tilde{\ell}_e$  extends the class of used element latency functions from  $\mathcal{L}$  to  $\tilde{\mathcal{L}}$ , we have  $\beta(\mathcal{L}) = \beta(\tilde{\mathcal{L}})$ . (The function EC changes, however.) Many of the upper bounds to be established later will be non-decreasing in the term  $\gamma = \gamma_1\gamma_2$  and hence it is desirable to reduce that quantity, while preserving other important properties of the game. We explain in the following how to do so by scaling. These elaborations are not necessary for understanding most of the rest of this work and hence can be skipped on a first read.

We call a vector  $\vec{\varepsilon} = (\varepsilon_e)_{e \in E} \in \mathbb{R}_{>0}^m$  a *relevance scaling* (or here just *scaling*) and define  $R_{\vec{\varepsilon}}$  to be the matrix  $R$  where row  $e$  was multiplied by  $\varepsilon_e$  for each  $e \in E$ . We would like to find an optimal scaling, i.e., one for that  $\gamma$ , taken w.r.t.  $R_{\vec{\varepsilon}}$ , is minimal over all possible scalings. It would be no extension to also consider series of scalings, since the sequential application of two scalings  $\vec{\varepsilon}$  and  $\vec{\delta}$  can as well be expressed by the one scaling  $(\varepsilon_e\delta_e)_{e \in E}$ . We call an instance *scalable* if it has a scaling that strictly reduces  $\gamma$ . So, if an instance  $\tilde{\Gamma}$  is *not* scalable, then there is no way to strictly reduce its  $\gamma$  by means of a scaling. If  $\tilde{\Gamma}$  is the result of scaling an instance  $\Gamma$  with  $\vec{\varepsilon}$ , then that scaling is optimal for  $\Gamma$ . For if  $\vec{\delta}$  was a better scaling for  $\Gamma$ , then the scaling  $(\varepsilon_e^{-1}\delta_e)_{e \in E}$  would reduce the  $\gamma$  of the non-scalable  $\tilde{\Gamma}$ , a contradiction.

Denote for each  $e \in E$

$$\begin{aligned}\gamma_1(e) &:= \max\left\{\frac{C_{eS}}{R_{eS}}; S \in \mathfrak{S}(e)\right\} , \\ \gamma_2(e) &:= \max\left\{\frac{R_{eS}}{C_{eS}}; S \in \mathfrak{S}(e)\right\} .\end{aligned}$$

Then, clearly, we have  $\gamma_1 = \max_{e \in E} \gamma_1(e)$  and  $\gamma_2 = \max_{e \in E} \gamma_2(e)$ .

**Proposition 2.** *If the maximum for  $\gamma_1$  and the maximum for  $\gamma_2$  are both attained for the same element, then the instance is not scalable.*

*Proof.* Let  $e \in E$  such that  $\gamma_i = \gamma_i(e)$ ,  $i = 1, 2$ . Let  $\vec{\varepsilon} \in \mathbb{R}_{>0}^m$  be a relevance scaling and  $\tilde{\gamma}_i(\cdot), \tilde{\gamma}_i, \tilde{\gamma}$ ,  $i = 1, 2$ , the ‘ $\gamma$ ’-values w.r.t.  $R_{\vec{\varepsilon}}$ . We have  $\tilde{\gamma}_1(e) = \frac{1}{\varepsilon_e} \gamma_1(e)$  and  $\tilde{\gamma}_2(e) = \varepsilon_e \gamma_2(e)$ . Since for  $\tilde{\gamma}_1$  and  $\tilde{\gamma}_2$ , the maximum is taken over all elements, these values cannot be smaller than  $\tilde{\gamma}_1(e)$  and  $\tilde{\gamma}_2(e)$ , respectively. It follows  $\tilde{\gamma} = \tilde{\gamma}_1 \tilde{\gamma}_2 \geq \frac{1}{\varepsilon_e} \gamma_1(e) \varepsilon_e \gamma_2(e) = \gamma_1(e) \gamma_2(e) = \gamma$ .  $\square$

The following procedure constructs an optimal relevance scaling.

- 1.) Initialize  $\vec{\varepsilon} := (1, \dots, 1)$  and  $R' := R$ . All ‘ $\gamma$ ’-quantities in the following are taken w.r.t.  $R'$ .
- 2.) Determine whether there is  $e \in E$  such that  $\gamma_1 = \gamma_1(e)$  and  $\gamma_2 = \gamma_2(e)$ . If so, then stop.
- 3.)   a) Find  $e \in E$  and  $S \in \mathfrak{S}(e)$  with  $\gamma_2 = \frac{R'_{eS}}{C_{eS}}$ .  
      b) Find  $T \in \mathfrak{S}(e)$  with  $\gamma_1(e) = \frac{C_{eT}}{R'_{eT}}$ .
- 4.) Set  $\varepsilon_e := \frac{C_{eT}}{R'_{eT}} \frac{1}{\gamma_1}$ .
- 5.) Multiply row  $e$  of  $R'$  by  $\varepsilon_e$  and replace  $R'$  with the resulting matrix.
- 6.) Goto 2.).

**Proposition 3.** *The above procedure constructs an optimal relevance scaling in at most  $m$  iterations.*

*Proof.* By Prop. 2, the stopping criterion ensures that upon termination, the given relevance scaling is optimal. Hence it is left to show that the procedure stops after at most  $m$  iterations.

In each iteration, the row  $e$  responsible for  $\gamma_2$  is scaled down as much as possible, i.e., such that the new  $\gamma_1(e)$  will just not exceed (the old)  $\gamma_1$ ; note that  $\gamma_1$  does not change during the whole procedure. This makes the new  $\gamma_1(e)$  equal to  $\gamma_1$ , thus  $\gamma_1$  is now attained for row  $e$ .

No row will be treated twice, for if a row  $e$  was chosen a second time, we would have  $\gamma_1(e) = \gamma_1$  and, since  $e$  was chosen again, also  $\gamma_2(e) = \gamma_2$ , so that the procedure would have stopped before that. Hence, the procedure stops after at most  $m$  iterations.  $\square$

For illustration, we note that rows are indeed scaled *down*, since  $\varepsilon_e < 1$  in each iteration by the properties of  $\gamma_1$ . Moreover, we note that in each iteration, there are two possible outcomes of scaling the row  $e$ . Either is  $\gamma_2$  still attained for  $e$  and the procedure terminates. Or the row in which  $\gamma_2$  is attained has changed to some  $f \neq e$ . Since only row  $e$  was modified,  $\gamma_2(f)$  is lesser or equal to the  $\gamma_2$  from before the iteration.

Scaling rows of the consumption matrix  $C$  is also possible. If a row  $e$  of  $C$  is scaled by  $\delta \in \mathbb{R}_{>0}$  then we have to replace  $\ell_e$  by  $x \mapsto \ell_e(\frac{1}{\delta}x)$ . However, for the sake of reducing  $\gamma = \gamma_1\gamma_2$  it does not matter whether we scale rows of  $R$ , or of  $C$ , or of both. For the combination of a relevance scaling  $\vec{\varepsilon}$  and consumption scaling  $\vec{\delta}$  can be transformed into an equivalent relevance scaling: set  $\lambda_e := \varepsilon_e/\delta_e$  for each  $e \in E$ . Then scaling  $R$  with  $\vec{\lambda}$  has the same effect on  $\gamma$  as the combination of  $\vec{\varepsilon}$  used on  $R$  and  $\vec{\delta}$  used on  $C$ .

## 6. Multicast Routing, a First Lower Bound, and Non-Uniqueness of Nash Equilibrium

The initial motivation for the NCRCG model comes from multicast routing. In multicast routing, we are given  $N$  sources  $s_i$ ,  $i \in [N]$ , each of them associated with  $k_i \in \mathbb{N}_{\geq 1}$  sinks  $t_i^1, \dots, t_i^{k_i}$ . For each  $i \in [N]$ , a demand of  $d_i$  has to be routed from  $s_i$  to each sink  $t_i^1, \dots, t_i^{k_i}$  simultaneously. Again, we assume the demand to represent a continuum  $[0, d_i]$  of players. The underlying structure, as in unicast routing, is a directed multigraph  $(V, E)$  where each edge  $e \in E$  has a latency function  $\ell_e$ . Let us fix one  $i \in N$  for a moment. To realize the desired routing for some fraction of the demand, we have to choose  $k_i$  paths  $\mathcal{P} := \{P_1, \dots, P_{k_i}\}$ , where  $P_j$  connects  $s_i$  with  $t_i^j$  for each  $j \in [k_i]$ . Such a set  $\mathcal{P}$  will constitute a strategy. Denote  $\mathcal{P}(e) := \{P \in \mathcal{P}; e \in P\}$ . Hence, to model this as an NCRCG, matrices  $C$  and  $R$  have to somehow represent all eligible choices of such collections of paths.

Before we can write out  $C$  and  $R$ , further decisions have to be made. There are at least two ways to *realize* a flow using a strategy  $\mathcal{P}$ . One is to realize a flow in the usual sense, obeying the Kirchhoff rule: the amount of flow entering a node is exactly the amount of flow leaving the node. This will be called a *conservation flow*. When using conservation flows, the consumption number of an edge  $e$  regarding strategy  $\mathcal{P}$  is  $|\mathcal{P}(e)|$ , i.e., the number of sinks served via  $e$ . A smarter way to realize  $\mathcal{P}$  is to exploit the fact that we deal with *data* to be routed. Unlike physical flows, data can be duplicated virtually without cost, provided that the nodes in the network offer such a feature. Thus the same data has to be sent down an edge only *once*, no matter how many sinks are served by this. We call this *duplication flows*. When using duplication flows, the consumption number of an edge regarding  $\mathcal{P}$  is 1 if it is contained in some path from  $\mathcal{P}$ , and 0 otherwise.

We now consider relevance numbers. There are many different reasonable ways in which the latency of a collection of paths  $\mathcal{P}$ , i.e., the strategy latency, could be understood; we point out four of them in [2]. Two interesting ones can be modeled by assigning relevance numbers to edges. In the first model, we would simply sum up the edge latencies of all edges involved, hence the relevance number of an edge is 1 if that

edge is contained in some path from  $\mathcal{P}$ , and 0 otherwise. We call this *edge-based strategy latency*. In the other model, we consider an edge more relevant if it serves many sinks. We will set its relevance number to  $|\mathcal{P}(e)|$  for that model<sup>2</sup> and refer to this model by the term *path-based strategy latency*, since it is equivalent to taking the sum of the latencies of all paths in  $\mathcal{P}$ , where the latency of a path is the sum of the latencies of the edges (exactly as in unicast routing). For a motivation, think of the source as a service provider and the terminals as customers which pay according to the quality of service. A higher latency means a loss of income for the service provider, which is proportional to the number of customers that experience that higher latency. The provider will hence consider an edge more important if it serves many customers.

Combining conservation flows with path-based strategy latency, or combining duplication flows with edge-based strategy latency, yields  $C = R$ , and is hence covered by previous analysis. However, combining conservation flows with edge-based latency, or combining duplication flows with path-based strategy latency, yields  $C \neq R$  in general. We will now focus on the latter combination, which we consider most important for applications.

### 6.1. Lower Bound and Non-Uniqueness of Nash Equilibrium

Consider the multicast instance from Fig. 1. We have one source, two sinks, and four strategies. Each strategy consists of two paths. For instance, strategy 1 consists of the paths (1, 3) and (1, 6); numbers denoting edges. Edges 1 and 2 have element latency function  $x \mapsto x^p$  and edges 3 to 6 have element latency function  $x \mapsto \theta x$ , where  $p \in \mathbb{N}_{\geq 1}$  and  $\theta \in \mathbb{R}_{\geq 0}$  are fixed parameters. Matrices  $C$  and  $R$  are as follows. Recall that strategies correspond to columns and elements correspond to rows. We have

$$C = \begin{pmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 \end{pmatrix} \quad \text{and} \quad R = \begin{pmatrix} 2 & 0 & 1 & 1 \\ 0 & 2 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 \end{pmatrix}.$$

It follows that  $\gamma_1 = 1$  and  $\gamma_2 = 2$ . We write out the strategy latencies for all four strategies for some arbitrary action distribution  $a$ :

$$\begin{aligned} L_1(a) &= 2\ell_1(g_1(a)) + \ell_3(g_3(a)) + \ell_6(g_6(a)) = 2g_1(a)^p + \theta g_3(a) + \theta g_6(a) \\ L_2(a) &= 2\ell_2(g_2(a)) + \ell_5(g_5(a)) + \ell_4(g_4(a)) = 2g_2(a)^p + \theta g_5(a) + \theta g_4(a) \\ L_3(a) &= \ell_1(g_1(a)) + \ell_3(g_3(a)) + \ell_2(g_2(a)) + \ell_4(g_4(a)) \\ &= g_1(a)^p + g_2(a)^p + \theta g_3(a) + \theta g_4(a) \\ L_4(a) &= \ell_1(g_1(a)) + \ell_6(g_6(a)) + \ell_2(g_2(a)) + \ell_5(g_5(a)) \\ &= g_1(a)^p + g_2(a)^p + \theta g_5(a) + \theta g_6(a) \end{aligned}$$

---

<sup>2</sup>Many other expressions involving  $|\mathcal{P}(e)|$  would also be reasonable, e.g.,  $|\mathcal{P}(e)|^c$  for some  $c > 0$ . All such variations can be expressed in the NCRCG model.



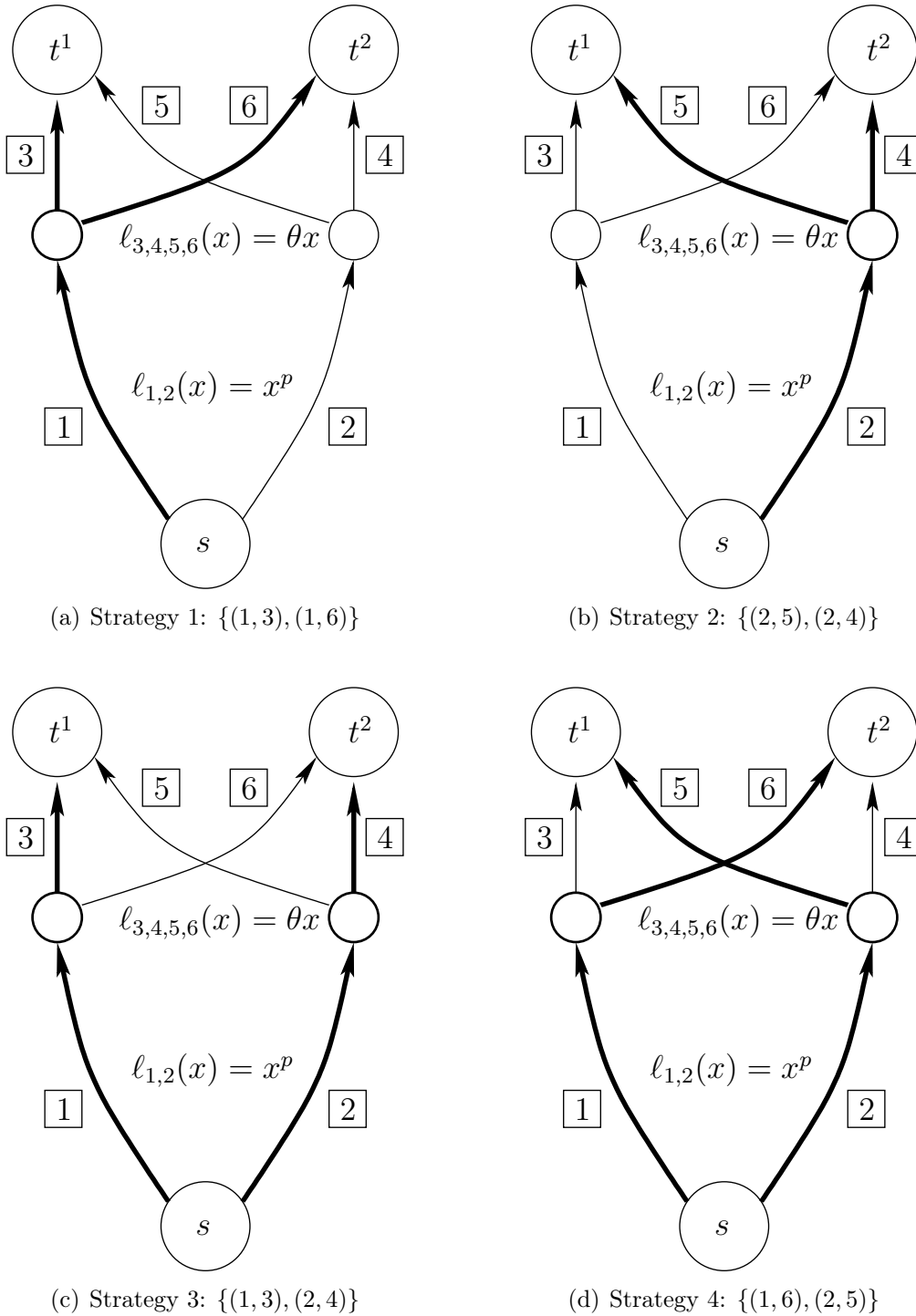


Figure 1: Multicast instance admitting multiple Nash equilibria with different congestions and different social cost. Link numbers are given in boxes,  $p \in \mathbb{N}_{\geq 1}$  and  $\theta \in \mathbb{R}_{\geq 0}$  are fixed parameters. The instance is shown four times to display all four strategies.

Let the demand be  $d = 1$ . It follows that  $a^* := (1/2, 1/2, 0, 0)$  and  $a' := (0, 0, 1/2, 1/2)$  are both Nash equilibria, since they induce the following congestions:

$$\begin{aligned} g_e(a^*) &= 1/2 \quad \forall e \in \{1, \dots, 6\} \ , \\ g_e(a') &= 1 \quad \forall e \in \{1, 2\} \quad \text{and} \quad g_e(a') = 1/2 \quad \forall e \in \{3, 4, 5, 6\} \ , \end{aligned}$$

and so  $L_S(a^*) = (1/2)^{p-1} + \theta$  and  $L_S(a') = 2 + \theta$  for all  $S \in \{1, \dots, 4\}$ . They have different social cost, namely  $\text{SC}(a^*) = (1/2)^{p-1} + \theta$  and  $\text{SC}(a') = 2 + \theta$ , and also induce different element latencies. This is an example that in the case  $C \neq R$ , there may be Nash equilibria of different social cost and different congestions, even though all element latency functions are strictly increasing (consider case  $\theta > 0$ ).

Regarding the price of anarchy, we note that it is at least

$$\frac{\text{SC}(a')}{\text{SC}(a^*)} = 2^{p-1} \frac{2 + \theta}{1 + \theta 2^{p-1}} = \Omega(2^p) = \Omega(\gamma_2^p) \quad \text{for } \theta = O(2^{1-p}).$$

This not only breaks the anarchy value bound, which would have been  $O(\frac{p}{\ln p})$ , but also shows a huge gap between the social costs of different Nash equilibria.

A similar example can be constructed with  $k$  sinks, for any number  $k \in \mathbb{N}$ , yielding  $\gamma_2 = k$  and a lower bound of  $\Omega(\gamma_2^p) = \Omega(k^p)$ . Details for that construction are given in [2].

## 7. General Lower Bound on the Price of Anarchy

Using the construction from [2], we can have multicast instances with a price of anarchy at least  $\Omega(\gamma_2^p)$  for any desired  $\gamma_2$ , using element latency functions from  $\text{Poly}^+(p)$ . We can extend that using the flexibility of general NCRCGs and also give a special bound in case of small  $\gamma$ .

**Theorem 3.** (i) *Let  $p \in \mathbb{N}_{\geq 1}$  and  $c, r \in \mathbb{R}_{\geq 1}$ . There exist NCRCGs with element latency functions only from  $\text{Poly}^+(p)$  with  $\gamma_1 = c$  and  $\gamma_2 = r$  such that the price of anarchy is at least  $\gamma^p = (\gamma_1 \gamma_2)^p$ .*

(ii) *Let  $p \in \mathbb{N}_{\geq 1}$  and  $c, r \in \mathbb{R}_{\geq 1}$  such that  $cr < (p + 1)^{\frac{1}{p}}$ . There exist NCRCGs with element latency functions only from  $\text{Poly}^+(p)$  with  $\gamma_1 = c$  and  $\gamma_2 = r$  such that the price of anarchy is at least*

$$\frac{1}{1 - \gamma p (p + 1)^{-1 - \frac{1}{p}}} .$$

The significance of the second lower bound will later be recognized when we notice that  $\beta(\text{Poly}^+(p)) = p(p + 1)^{-1 - \frac{1}{p}}$ , and hence the lower bound is  $1/(1 - \gamma\beta(\text{Poly}^+(p)))$ , being a natural extension to the known anarchy value bound for NCGs.

Both lower bounds from Thm. 3 have an upper-bound counterpart which is no more than a factor of  $\gamma$  away and will be given in Thm. 4.

*Proof of Thm. 3.* Part (i). Different kinds of instances can be used to show this worst-case lower bound. We give one that, provided  $c, r \in \mathbb{N}$ , only uses integral numbers in  $C$  and  $R$ , and only integral coefficients in the polynomials. Let  $N := 1$ ,  $d := 1$ ,  $m := r$ , and  $n := m + 1$  and define:

$$\begin{aligned} C &:= \begin{pmatrix} 1 & & c \\ & \ddots & \vdots \\ & & 1 & c \end{pmatrix} \in \mathbb{N}^{m \times (m+1)} \\ R &:= \begin{pmatrix} r & & 1 \\ & \ddots & \vdots \\ & & r & 1 \end{pmatrix} \in \mathbb{N}^{m \times (m+1)} \end{aligned} \quad (5)$$

Empty entries are 0. Then  $\gamma_1 = c$  and  $\gamma_2 = r$ , and the instance is not scalable. Let each element latency function be  $x \mapsto x^p$  and  $a := (0, \dots, 0, 1)$  and  $a^* := (\frac{1}{m}, \dots, \frac{1}{m}, 0)$ . Then  $a$  is a Nash equilibrium, since

$$\begin{aligned} \forall S \in [m] \quad L_S(a) &= rc^p \quad \text{and} \\ L_{m+1}(a) &= \sum_{e=1}^m c^p = mc^p = rc^p . \end{aligned}$$

We have its social cost

$$\text{SC}(a) = rc^p d = rc^p . \quad (6)$$

For  $a^*$  we have the following (showing as a side-product that it happens to be a Nash equilibrium as well):

$$\begin{aligned} \forall S \in [m] \quad L_S(a^*) &= r \left(\frac{1}{m}\right)^p = \left(\frac{1}{r}\right)^{p-1} \quad \text{and} \\ L_{m+1}(a^*) &= \sum_{e=1}^m \left(\frac{1}{m}\right)^p = m \left(\frac{1}{m}\right)^p = \left(\frac{1}{r}\right)^{p-1} . \end{aligned}$$

We have its social cost

$$\text{SC}(a^*) = \left(\frac{1}{r}\right)^{p-1} d = \left(\frac{1}{r}\right)^{p-1} .$$

It follows that  $\text{SC}(a)/\text{SC}(a^*) = rc^p r^{p-1} = (cr)^p = (\gamma_1 \gamma_2)^p = \gamma^p$ .

Part (ii). We extend the example from part (i) by a strategy with index  $r + 2$  with constant latency  $L_{r+2}(\cdot) = rc^p$ . By essentially the same argument as before,  $a := (0, \dots, 0, 1, 0)$  is a Nash equilibrium with social cost  $\text{SC}(a) = rc^p$ , just note that the newly introduced strategy does not offer better latency than the other ones under  $a$ . Define furthermore

$$a^* := \left( \frac{c}{(p+1)^{1/p}}, \dots, \frac{c}{(p+1)^{1/p}}, 0, 1 - r \frac{c}{(p+1)^{1/p}} \right) \in \mathbb{R}_{\geq 0}^{r+2} .$$

Then by the assumption on  $cr$ , we have an action distribution. So, we follow the idea of the first lower bound, but additionally take advantage of the fact that we can move some of the demand to the new strategy with constant latency.<sup>3</sup> The social cost of  $a^*$  is

$$\begin{aligned}
\text{SC}(a^*) &= \sum_{S=1}^r L_S(a^*)a_S^* + L_{r+2}(a^*)a_{r+2}^* \\
&= \sum_{S=1}^r r \frac{c^p}{p+1} \frac{c}{(p+1)^{1/p}} + rc^p \left(1 - r \frac{c}{(p+1)^{1/p}}\right) \\
&= rc^p \left( r \frac{1}{p+1} \frac{c}{(p+1)^{1/p}} + \left(1 - r \frac{c}{(p+1)^{1/p}}\right) \right) \\
&= rc^p \left(1 - rc \left((p+1)^{-\frac{1}{p}} - (p+1)^{-1} (p+1)^{-\frac{1}{p}}\right)\right) \\
&= rc^p \left(1 - rc \left((p+1)(p+1)^{-1-\frac{1}{p}} - (p+1)^{-1-\frac{1}{p}}\right)\right) \\
&= rc^p \left(1 - rc (p+1)^{-1-\frac{1}{p}} ((p+1) - 1)\right) \\
&= rc^p \left(1 - rc p (p+1)^{-1-\frac{1}{p}}\right) .
\end{aligned}$$

Recall that the Nash equilibrium  $a$  has  $\text{SC}(a) = rc^p$ . Hence

$$\frac{\text{SC}(a)}{\text{SC}(a^*)} = \frac{1}{1 - rc p (p+1)^{-1-\frac{1}{p}}} = \frac{1}{1 - \gamma p (p+1)^{-1-\frac{1}{p}}} . \quad \square$$

## 8. Upper Bound on the Price of Anarchy

Recall the characterization of Nash equilibria given in Thm.1. We can apply that to prove upper bounds on the price of anarchy, inspired by a basic idea from [11]. We first relate (mixed) total element cost to (mixed) social cost.

**Proposition 4.** *Let  $a, b \in \mathcal{A}$ . Then we have*

$$\frac{1}{\gamma_2} \text{SC}^a(b) \leq \text{EC}^a(b) \leq \gamma_1 \text{SC}^a(b) .$$

*Proof.* We start with the first inequality.

$$\begin{aligned}
\text{SC}^a(b) &= \sum_{S \in \mathfrak{S}} L_S(a) b_S = \sum_{S \in \mathfrak{S}} \sum_{e \in S} R_{eS} \ell_e(g_e(a)) b_S \\
&= \sum_{S \in \mathfrak{S}} \sum_{e \in S} \frac{R_{eS}}{C_{eS}} C_{eS} \ell_e(g_e(a)) b_S \leq \gamma_2 \sum_{S \in \mathfrak{S}} \sum_{e \in S} C_{eS} \ell_e(g_e(a)) b_S
\end{aligned}$$

---

<sup>3</sup>The exact setting was obtained by introducing a parameter  $x$ , and setting  $a^* = (x, \dots, x, 0, 1 - rx)$ , and then choosing an  $x$  with  $\text{SC}(a^*)$  minimal under the constraint that  $a^*$  is an action distribution.

$$= \gamma_2 \sum_{e \in E} \ell_e(g_e(a)) \underbrace{\sum_{S \in \mathfrak{G}(e)} C_{eS} b_S}_{=g_e(b)} = \gamma_2 \mathbf{EC}^a(b) .$$

The second inequality follows likewise:

$$\begin{aligned} \mathbf{EC}^a(b) &= \sum_{e \in E} \ell_e(g_e(a)) g_e(b) = \sum_{e \in E} \ell_e(g_e(a)) \sum_{S \in \mathfrak{G}(e)} C_{eS} b_S \\ &= \sum_{S \in \mathfrak{G}} \sum_{e \in S} \frac{C_{eS}}{R_{eS}} R_{eS} \ell_e(g_e(a)) b_S \\ &\leq \gamma_1 \sum_{S \in \mathfrak{G}} b_S \underbrace{\sum_{e \in S} R_{eS} \ell_e(g_e(a))}_{=L_S(a)} = \gamma_1 \mathbf{SC}^a(b) . \end{aligned} \quad \square$$

For the rest of this section, we will treat NCRCGs with super-homogeneous element latency functions.

**Definition 5.** Let  $s : (0, 1] \longrightarrow (0, \infty)$  such that  $s(1) = 1$ .

- (i) An element latency function  $\ell$  is called *s-super-homogeneous* if  $\ell(\varepsilon x) \geq s(\varepsilon)\ell(x)$  for all  $\varepsilon \in (0, 1]$  and all  $x \in \mathbb{R}_{\geq 0}$ .
- (ii) An element latency function class  $\mathcal{L}$  is called *s-super-homogeneous* if each  $\ell \in \mathcal{L}$  is *s-super-homogeneous*.
- (iii) Define  $\bar{s} : [1, \infty) \longrightarrow (0, \infty), t \mapsto s(t^{-1})^{-1}$ .

*Remark 1.* The element latency function  $\ell$  is *s-super-homogeneous* if and only if it is  $\bar{s}$ -sub-homogeneous, i.e.,

$$\ell(tx) \leq \bar{s}(t)\ell(x) \quad \forall t \geq 1 \quad \forall x \in \mathbb{R}_{\geq 0} . \quad (7)$$

*Proof.* If  $\ell$  is *s-super-homogeneous*, we have  $\ell(x) = \ell(t^{-1}tx) \geq s(t^{-1})\ell(tx) = \bar{s}(t)^{-1}\ell(tx)$  for all  $t \geq 1$  and all  $x \in \mathbb{R}_{\geq 0}$ , so  $\ell$  is  $\bar{s}$ -sub-homogeneous. If, on the other hand,  $\ell$  fulfills (7) then  $\ell(x) = \ell(\varepsilon^{-1}\varepsilon x) \leq \bar{s}(\varepsilon^{-1})\ell(\varepsilon x) = s(\varepsilon)^{-1}\ell(\varepsilon x)$  for all  $\varepsilon \in (0, 1]$  and all  $x \in \mathbb{R}_{\geq 0}$ , so  $\ell$  is *s-super-homogeneous*.  $\square$

For instance, the classes  $\text{Poly}^+(p)$  are *s-super-homogeneous* for  $s(\varepsilon) = \varepsilon^p$  (and also *s-sub-homogeneous*).

**Lemma 1.** Let  $\mathcal{L}$  be an *s-super-homogeneous* class of element latency functions. Let an NCRCG be given with element latency functions drawn from  $\mathcal{L}$ . Let  $a, b \in \mathbb{R}_{\geq 0}^n$  and  $t \geq 1$ . Then

$$\mathbf{EC}^a(b) \leq \frac{1}{t} \beta(\mathcal{L}) \mathbf{EC}(a) + \bar{s}(t) \mathbf{EC}(b) .$$

*Proof.*

$$\begin{aligned}
t \text{EC}^a(b) &= \sum_{e \in E} \ell_e(g_e(a)) t g_e(b) \\
&= \sum_{e \in E} (\ell_e(g_e(a)) - \ell_e(t g_e(b))) t g_e(b) + t \ell_e(t g_e(b)) g_e(b) \\
&\leq \sum_{e \in E} (\beta(\mathcal{L}) \ell_e(g_e(a)) g_e(a) + t \ell_e(t g_e(b)) g_e(b)) \quad \text{def. of } \beta(\mathcal{L}) \text{ or (3) on p. 8} \\
&\leq \sum_{e \in E} (\beta(\mathcal{L}) \ell_e(g_e(a)) g_e(a) + t \bar{s}(t) \ell_e(g_e(b)) g_e(b)) \quad \bar{s}\text{-sub-homogeneity} \\
&= \beta(\mathcal{L}) \text{EC}(a) + t \bar{s}(t) \text{EC}(b) .
\end{aligned}$$

Dividing by  $t$  yields the claimed inequality.  $\square$

Following [11] we derive a bound on  $\beta(\mathcal{L})$  for super-homogeneous classes.

**Lemma 2** ([11, Lem. 4.1]). *Let  $\mathcal{L}$  be an  $s$ -super-homogeneous class of element latency functions. Then  $\beta(\mathcal{L}) \leq \sup_{0 < \varepsilon < 1} \varepsilon(1 - s(\varepsilon))$ .*

*Proof.*

$$\begin{aligned}
\beta(\mathcal{L}) &= \sup_{\substack{\ell \in \mathcal{L} \\ 0 \leq v}} \frac{1}{\ell(v)v} \max_{0 \leq x} (\ell(v) - \ell(x))x \\
&= \sup_{\substack{\ell \in \mathcal{L} \\ 0 \leq v}} \frac{1}{\ell(v)v} \max_{0 \leq x \leq v} (\ell(v) - \ell(x))x && \text{by monotonicity} \\
&= \sup_{\substack{\ell \in \mathcal{L} \\ 0 < v}} \frac{1}{\ell(v)v} \max_{0 < x < v} (\ell(v) - \ell(x))x \\
&= \sup_{\substack{\ell \in \mathcal{L} \\ 0 < v}} \frac{1}{\ell(v)v} \max_{0 < x < v} (\ell(v) - \ell(\frac{x}{v}v))x \\
&\leq \sup_{\substack{\ell \in \mathcal{L} \\ 0 < v}} \frac{1}{\ell(v)v} \max_{0 < x < v} (\ell(v) - s(\frac{x}{v})\ell(v))x && \text{since } 0 < \frac{x}{v} < 1 \\
&= \sup_{0 < \varepsilon < 1} \varepsilon(1 - s(\varepsilon)) . && \square
\end{aligned}$$

A bound follows for  $(\varepsilon \mapsto \varepsilon^p)$ -super-homogeneous classes, e.g.,  $\text{Poly}^+(p)$ .

**Corollary 1** (cf. [11, Cor. 4.4]). *Let  $\mathcal{L}$  be an  $(\varepsilon \mapsto \varepsilon^p)$ -super-homogeneous element latency function class. Then  $\beta(\mathcal{L}) \leq p(p+1)^{-1-\frac{1}{p}}$ .*

*Proof.*

$$\beta(\mathcal{L}) \leq \sup_{0 < \varepsilon < 1} \varepsilon - \varepsilon^{p+1} \quad \text{by Lem. 2}$$

$$\begin{aligned}
&= (p+1)^{-\frac{1}{p}} - (p+1)^{-1-\frac{1}{p}} && \text{real calculus} \\
&= (p+1)(p+1)^{-1-\frac{1}{p}} - (p+1)^{-1-\frac{1}{p}} \\
&= ((p+1) - 1)(p+1)^{-1-\frac{1}{p}} = p(p+1)^{-1-\frac{1}{p}} . && \square
\end{aligned}$$

*Remark 2.* The bound in Cor. 1 is the best possible as shown by Pigou's example, see, e.g., [18] and the references therein.

The following is our main result in this section.

**Theorem 4.** (i) *The price of anarchy in an NCRCG with element latency functions drawn from an  $s$ -super-homogeneous class  $\mathcal{L}$  for which  $\beta := \beta(\mathcal{L}) < 1$  holds is no more than*

$$\begin{cases} \frac{1}{1-\beta}\gamma\bar{s}(\gamma) & \text{for all } \gamma \\ \min\{\frac{1}{1-\gamma\beta}\gamma, \frac{1}{1-\beta}\gamma\bar{s}(\gamma)\} & \text{for } \gamma < 1/\beta \end{cases} .$$

(ii) *The price of anarchy in an NCRCG with element latency functions drawn from  $\text{Poly}^+(p)$  is no more than*

$$\begin{cases} \gamma^{p+1} & \text{if } \gamma \geq (1+p)^{\frac{1}{p}} \\ \frac{1}{1-\gamma\beta}\gamma & \text{if } \gamma \leq (1+p)^{\frac{1}{p}} \end{cases} , \quad \text{where } \beta = p(p+1)^{-1-\frac{1}{p}} .$$

Let us briefly discuss these bounds before starting the proof. The bound  $\frac{1}{1-\beta}\gamma\bar{s}(\gamma)$  is universal; it holds for all  $\gamma$  and all classes  $\mathcal{L}$  of element latency functions, provided only that  $\beta = \beta(\mathcal{L}) < 1$ . The bound  $\frac{1}{1-\gamma\beta}\gamma$  is only proved for small  $\gamma$  and is sometimes better than the universal one, depending on the function  $\bar{s}$ . The second part of the theorem treats the case of polynomial element latency functions, i.e., from  $\text{Poly}^+(p)$  for a fixed  $p$ . We distinguish between large and small  $\gamma$ . Large  $\gamma$  start at  $(1+p)^{\frac{1}{p}}$ , and it will become clear from the proof why we chose that threshold. The bound for large  $\gamma$  improves on the universal one by a factor of  $1/(1-\beta)$ , where  $\beta = p(p+1)^{-1-\frac{1}{p}}$ , as we know well by now. The bound for small  $\gamma$  is a corollary from the bounds in part (i) of the theorem. In fact we get a bound of  $\min\{\frac{1}{1-\gamma\beta}\gamma, \frac{1}{1-\beta}\gamma^{p+1}\}$ . However, real calculus shows that the first expression is always upper-bounded by the second one and hence the first is the better bound.

*Proof of Thm. 4.* Let  $a$  be a Nash equilibrium and  $b$  be any action distribution. For all  $t \geq 1$  we have

$$\begin{aligned}
\text{SC}(a) &\leq \text{SC}^a(b) && \text{by Thm. 1} \\
&\leq \gamma_2 \text{EC}^a(b) && \text{by Prop. 4} \\
&\leq \gamma_2 \left( \frac{1}{t} \beta \text{EC}(a) + \bar{s}(t) \text{EC}(b) \right) && \text{by Lem. 1} \\
&\leq \gamma_2 \gamma_1 \left( \frac{1}{t} \beta \text{SC}(a) + \bar{s}(t) \text{SC}(b) \right) && \text{by Prop. 4}
\end{aligned}$$

$$= \gamma \left( \frac{1}{t} \beta \text{SC}(a) + \bar{s}(t) \text{SC}(b) \right) .$$

*Key observation:* if  $t \geq 1$  is chosen such that  $1 - \gamma \frac{1}{t} \beta > 0$ , the above yields

$$\frac{\text{SC}(a)}{\text{SC}(b)} \leq \frac{1}{1 - \gamma \frac{1}{t} \beta} \gamma \bar{s}(t).$$

All bounds follow from the key observation by an appropriate choice of  $t$ .

- For the first bound of part (i), i.e., the universal bound, choose  $t := \gamma$  and note that  $t \geq 1$  holds then, as well as  $1 - \gamma \frac{1}{t} \beta = 1 - \beta > 0$  since  $\beta < 1$ .
- To prove the second bound of part (i), let  $\gamma < 1/\beta$ , hence  $\gamma\beta < 1$ . We only have to show the  $\frac{1}{1-\gamma\beta}\gamma$  bound, since the other one is the universal bound. Choose  $t := 1$ . Then  $1 - \gamma \frac{1}{t} \beta = 1 - \gamma\beta > 0$ , and the bound follows from the key observation, since  $\bar{s}(1) = 1$ .
- To prove the first bound from part (ii), choose  $t := \gamma(p+1)^{-\frac{1}{p}}$ , which is  $\geq 1$  by assumption on  $\gamma$ , and also implies

$$1 - \gamma \frac{1}{t} \beta = 1 - (p+1)^{\frac{1}{p}} p (p+1)^{-1-\frac{1}{p}} = 1 - \frac{p}{p+1} > 0 .$$

The key observation yields

$$\frac{\text{SC}(a)}{\text{SC}(b)} \leq \frac{\gamma t^p}{1 - \frac{p}{p+1}} = \frac{\gamma \gamma^p (p+1)^{-1}}{1 - \frac{p}{p+1}} = \gamma^{p+1} .$$

- For the second bound from part (ii) we just have to note that

$$\gamma\beta \leq (p+1)^{\frac{1}{p}} p (p+1)^{-1-\frac{1}{p}} = \frac{p}{p+1} < 1 ,$$

hence  $\gamma < 1/\beta$ , and invoke the second bound from part (i).  $\square$

*Remark 3.* The bounds in Thm. 4(ii) are tight up to a factor of  $\gamma$  by Thm. 3.

When establishing the upper bound from Thm. 4 for a concrete instance with non-decreasing  $\bar{s}$  (as it is the case for polynomials), we can first scale that instance in order to minimize  $\gamma$ , as described in Sec. 5. As mentioned before, it can be shown easily that if that scaling extends the class of element latency functions used from  $\mathcal{L}$  to some  $\tilde{\mathcal{L}}$ , we have  $\beta(\mathcal{L}) = \beta(\tilde{\mathcal{L}})$ . Since all bounds are non-increasing in  $\gamma$  and, for part (ii), we have  $\frac{1}{1-\gamma\beta}\gamma = \gamma^{p+1}$  if  $\gamma = (1+p)^{\frac{1}{p}}$ , the bounds can only improve by the scaling.

The bounds for the scaled instance also hold for the original one. So, in particular, it is not necessary to ‘really’ scale the instance – if, e.g., further algorithmic tasks shall be carried out with the instance, we can well work with the original one. This is how we proceeded in our experimental studies.



## 9. Bicriteria Bound

The next theorem says that a Nash equilibrium is not worse in terms of social cost than an optimum for  $(1 + \beta(\mathcal{L}))\gamma$  times the demand. It is hence a natural extension to the known bicriteria bound for NCGs, which use a scaling of the demand by  $1 + \beta(\mathcal{L})$ .

**Theorem 5.** *Let  $\Gamma$  be an NCRCG with element latency functions drawn from the  $s$ -super-homogeneous class  $\mathcal{L}$ . Let  $a$  be a Nash equilibrium for  $\Gamma$  and  $b$  an action distribution for the NCRCG resulting from  $\Gamma$  by a multiplication of each demand  $d_i$  by  $(1 + \beta(\mathcal{L}))\gamma$ . Then  $\text{SC}(a) \leq \text{SC}(b)$ .*

*Proof.* Let  $t, u \geq 1$  to be specified more precisely later. Let  $b$  be an action distribution for the demands scaled up by  $u$ . Then  $\frac{1}{u}b$  is an action distribution for  $\Gamma$  and hence

$$\begin{aligned} \text{SC}(a) &\leq \text{SC}^a\left(\frac{1}{u}b\right) \leq \gamma_2 \text{EC}^a\left(\frac{1}{u}b\right) && \text{by Thm. 1 and Prop. 4} \\ &= \gamma_2 \frac{1}{u} \text{EC}^a(b) \\ &\leq \gamma_2 \frac{1}{u} \left( \frac{1}{t} \beta(\mathcal{L}) \text{EC}(a) + \bar{s}(t) \text{EC}(b) \right) && \text{by Lem. 1} \\ &\leq \gamma \frac{1}{u} \left( \frac{1}{t} \beta(\mathcal{L}) \text{SC}(a) + \bar{s}(t) \text{SC}(b) \right) . && \text{by Prop. 4} \end{aligned}$$

It follows that

$$\text{SC}(a) \underbrace{\left( \frac{u}{\gamma} - \frac{\beta(\mathcal{L})}{t} \right) \frac{1}{\bar{s}(t)}}_{\phi(u,t):=} \leq \text{SC}(b) .$$

To receive ' $\text{SC}(a) \leq \text{SC}(b)$ ', we need the  $\phi(u, t) \geq 1$ . Since  $\phi(\cdot, t)$  is strictly increasing and we wish to have  $u$  as small as possible, we aim for  $\phi(u, t) = 1$ , i.e.,

$$u = \left( \bar{s}(t) + \frac{\beta(\mathcal{L})}{t} \right) \gamma . \quad (8)$$

Choosing  $t := 1$ , we are forced to choose  $u := (1 + \beta(\mathcal{L}))\gamma$  and get (8); recall that  $\bar{s}(1) = 1$ . Note that for the case  $\bar{s}(t) = t^p$ , this choice of  $t$  is optimal, since for such  $\bar{s}$  the right-hand side in (8), increases with  $t$ .  $\square$

**Theorem 6.** *Consider element latency functions from  $\text{Poly}^+(p)$ . The scaling factor  $(1 + \beta(\mathcal{L}))\gamma$  in the previous theorem is the best possible up to a factor of  $(1 + \beta(\mathcal{L}))\gamma^{\frac{1}{p+1}}$ .*

*Proof.* Recall the first example from the proof of Thm. 3, displayed in (5) on page 18. We have parameters  $c, r, p \in \mathbb{N}_{\geq 1}$ ,  $m = r$ ,  $n = m + 1$ , element latency functions  $x \mapsto x^p$ , and as shown in that proof, we can construct a Nash equilibrium  $a$  with social cost  $\text{SC}(a) = rc^p$ , see (6). On the other hand, if we scale up the demand of 1 by factor

$$u := \frac{(1 + \beta(\mathcal{L}))\gamma}{(1 + \beta(\mathcal{L}))\gamma^{\frac{1}{p+1}}} = \gamma^{1 - \frac{1}{p+1}} = (cr)^{1 - \frac{1}{p+1}} ,$$

there is an action distribution  $b := (u/m, \dots, u/m, 0) \in \mathbb{R}_{\geq 0}^{m+1}$ . Let us compute  $\text{SC}(b)$ . We have

$$\begin{aligned} \forall S \in [m] \quad L_S(b) &= r \left(\frac{u}{m}\right)^p = r \left(\frac{u}{r}\right)^p \quad \text{and} \\ L_{m+1}(b) &= \sum_{e=1}^m \left(\frac{u}{m}\right)^p = m \left(\frac{u}{m}\right)^p = r \left(\frac{u}{r}\right)^p . \end{aligned}$$

It follows that

$$\text{SC}(b) = r \left(\frac{u}{r}\right)^p u = r^{1-p} u^{p+1} = r^{1-p} (cr) \left(1 - \frac{1}{p+1}\right)^{(p+1)} = r^{1-p} (cr)^p = r c^p .$$

Hence we have to scale the demands up by at least  $u = \gamma^{1-\frac{1}{p+1}}$  in order to have  $\text{SC}(a) \leq \text{SC}(b')$  for all action distributions  $b'$  that is admissible for the scaled demands.  $\square$

## 10. Computation

Convex programs play an important role for the computation of optima and Nash equilibria. Convex programs can be solved in polynomial time up to an arbitrarily small error, given some mild additional properties. Some of the practical methods require certain boundedness conditions, smoothness and efficiently computable first and second order derivatives of the functions involved. See, e.g., [4, 6, 25] for comprehensive treatments of convex optimization. The case of a convex quadratic function and linear constraints is polynomially solvable with the ellipsoid algorithm [19] or, up to an arbitrarily small error, by interior-point algorithms [4, 6, 25]. We will encounter that special case later with affine element latency functions, since then  $\text{SC}$  is a quadratic function.

Computation of Nash equilibria and optima is hindered by two changes that occur when we switch from NCGs to NCRCGs:

- $\text{SC}$  is not necessarily a convex function anymore;
- there seems to be no way to use the KKT theorem to recognize Nash equilibria as optima of a convex program (cf. Sec. 3.6).

We will address these concerns separately below. Before, we state a fact about  $\text{EC}$ . By Prop. 4, an algorithm that minimizes  $\text{EC}$  yields a  $\gamma$ -approximation for  $\text{OPT}$ . This is especially interesting if  $x \mapsto \ell_e(x)x$  is a convex function for all  $e \in E$ , since then  $\text{EC}$  is a convex function, even a separable one if considered a function of the congestions. However, it is clear that such an approach can only be of limited use since the computation does not involve the matrix  $R$ .

### 10.1. Convexity and Non-Convexity of $\text{SC}$ ; Computation of Optima

Let all element latency functions be twice continuously differentiable on  $\mathbb{R}_{>0}$ , which implies that  $\text{SC}$  is a twice continuously differentiable function on  $\mathbb{R}_{>0}^n$  with its Hessian

at  $v \in \mathbb{R}_{>0}^n$  being

$$\nabla^2 \text{SC}(v) = \left( \sum_{e \in E} \left\{ (R_{eS_1} C_{eS_2} + R_{eS_2} C_{eS_1}) \ell'_e(g_e(v)) + \sum_{S \in \mathfrak{S}} R_{eS} C_{eS_1} C_{eS_2} \ell''_e(g_e(v)) v_S \right\} \right)_{\substack{S_1 \in \mathfrak{S} \\ S_2 \in \mathfrak{S}}}.$$

If all element latency functions are affine, say,  $\ell_e(x) = \theta_e x + \tau_e$  with  $\theta_e, \tau_e \in \mathbb{R}_{\geq 0}$ , then  $\nabla^2 \text{SC}(v)$  is independent of  $v$ , namely

$$\nabla^2 \text{SC}(v) = \left( \sum_{e \in E} \theta_e (R_{eS_1} C_{eS_2} + R_{eS_2} C_{eS_1}) \right)_{\substack{S_1 \in \mathfrak{S} \\ S_2 \in \mathfrak{S}}}. \quad (9)$$

A real-valued, twice differentiable function of multiple real variables is convex on a convex open set if and only if its Hessian is positive semidefinite in every point of that set. We cannot apply that directly to  $\text{SC}$ , since the relevant set, namely  $\mathcal{A}$ , is not open (it is convex, though). If  $\nabla^2 \text{SC}$  is positive semidefinite in every point of some open convex set  $U \subseteq \mathbb{R}_{>0}^n$  such that  $\mathcal{A} \subseteq \bar{U}$ , then  $\text{SC}$  is convex on  $U$  and hence, by continuity, also on  $\mathcal{A}$ . The converse, however, does not hold, and so a test based on this can deliver false positives, i.e., categorize  $\text{SC}$  as non-convex when, in fact, it is convex (on  $\mathcal{A}$ ). As an example, consider  $C := \begin{pmatrix} 1 & 1 \end{pmatrix}$  and  $R := \begin{pmatrix} 3 & 1 \end{pmatrix}$ , let the one element have latency function  $x \mapsto x$ , and a demand  $d = 1$  be given. Then,  $\text{SC}$  is convex on  $\mathcal{A} = \{(a_1, a_2); a_1 + a_2 = 1\}$ . However  $\nabla^2 \text{SC}(\cdot) = \begin{pmatrix} 6 & 4 \\ 4 & 2 \end{pmatrix}$  is not positive semidefinite; it has got the negative eigenvalue  $4 - \sqrt{20}$ .

For an exact test, we have to use the *projected Hessian* instead. Details are explained in the following. Let the strategies be ordered such that strategies from one player class are grouped together and these groups arranged in the order  $i = 1, \dots, N$ . Set  $n_i := |\mathfrak{S}_i|$  for each  $i \in [N]$ . So, in our ordering the first  $n_1$  strategies are from player class 1, the next  $n_2$  strategies are from player class 2, etc. Let  $\mathfrak{S}'_i$  be the set  $\mathfrak{S}_i$  without the last strategy, so  $|\mathfrak{S}'_i| = n_i - 1$ . We will use an ordering on  $\mathfrak{S}' := \bigcup_{i \in [N]} \mathfrak{S}'_i$  similar to that on  $\mathfrak{S}$ , i.e., the first  $n_1 - 1$  strategies in  $\mathfrak{S}'$  are from player class 1, the next  $n_2 - 1$  strategies are from player class 2, etc. Define the affine mapping  $T$  and its matrix  $M_T \in \mathbb{R}^{n \times (n-N)}$  as shown in Sec. A in the appendix. The version for  $N = 1$  is as follows:

$$T : \mathbb{R}^{n-1} \longrightarrow \mathbb{R}^n, (v_1, \dots, v_{n-1}) \mapsto \underbrace{\begin{pmatrix} 1 & & & \\ & 1 & & \\ & & \ddots & \\ & & & 1 \\ -1 & -1 & \dots & -1 \end{pmatrix}}_{M_T :=} v + \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ d \end{pmatrix}.$$

However, in the following we will always refer to the general version found in the appendix. For each  $i \in [N]$  we have the following:  $n_i - 1$  real numbers are mapped by  $T$

to  $n_i$  real numbers such that the sum of the latter is always  $d_i$ . Define the set of *reduced action distributions*

$$\mathcal{A}' := \{v \in \mathbb{R}_{>0}^{n-N}; \sum_{S \in \mathcal{E}'_i} v_S < d_i \quad \forall i \in [N]\} .$$

This is an open convex set in  $\mathbb{R}^{n-N}$ . The image of  $\mathcal{A}'$  under  $T$  is contained in  $\mathbb{R}_{>0}^n$  and it consists of all action distributions save those which have one or more zero entries. It follows that  $\overline{T(\mathcal{A}')} = \mathcal{A}$ . By continuity of  $\text{SC}$ , we hence know that  $\text{SC}$  is convex on  $T(\mathcal{A}')$  if and only if it is convex on  $\mathcal{A}$ . Since  $T$  is an affine mapping,  $\text{SC}$  is convex on  $T(\mathcal{A}')$  if and only if  $\text{SC} \circ T$  is convex on  $\mathcal{A}'$ . The latter is the case if and only if  $\nabla^2(\text{SC} \circ T)(v)$  is positive semidefinite for all  $v \in \mathcal{A}'$ , since  $\mathcal{A}'$  is open.

We know from calculus that for twice continuously differentiable functions  $h : U \rightarrow \mathbb{R}^n$  and  $f : V \rightarrow \mathbb{R}$ , with  $h(U) \subseteq V$ ,  $U \subseteq \mathbb{R}^r$ , that for all  $v \in U$  we have

$$\nabla^2(f \circ h)(v) = \text{J}h(v)^\perp \cdot \nabla^2 f(h(v)) \cdot \text{J}h(v) + (\nabla f(h(v)) \cdot h_{ji}^*(v))_{\substack{j=1,\dots,r \\ i=1,\dots,r}}$$

where  $h_{ji}^*(v) = (\partial_j \partial_i h_k(v))_{k=1,\dots,n}$  for each  $i, j \in [n]$  is the vector of second partial derivatives  $\partial_j \partial_i$  of all component functions of  $h$  at  $v$ , and  $\text{J}h(v)$  denotes the Jacobian of  $h$  at  $v$ . Applying that to  $f := \text{SC}$  and  $h := T$  yields

$$\nabla^2(\text{SC} \circ T)(v) = M_T^\perp \cdot \nabla^2 \text{SC}(T(v)) \cdot M_T , \quad (10)$$

since all second-order partial derivatives of  $T$  vanish. The matrix given by (10) is also known as the *projected Hessian*. If element latency functions are affine, we have constant Hessian for  $\text{SC}$  as shown in (9), making it easy to check the projected Hessian for being positive semidefinite on  $\mathcal{A}'$ . We summarize our findings in the following theorem.

**Theorem 7.** *Let all element latency functions be twice continuously differentiable (on  $\mathbb{R}_{>0}$ ). Then  $\text{SC}$  is convex on  $\mathcal{A}$  if and only if the projected Hessian (10) is positive semidefinite for all  $v \in \mathcal{A}'$ . If all element latency functions are affine, the projected Hessian is independent of  $v$  and hence can be checked efficiently<sup>4</sup> for being positive semidefinite on  $\mathcal{A}'$ .*

Finally, if  $\text{SC}$  is convex, optima are characterized by a convex program:

$$\begin{aligned} \min \quad & \text{SC}(a) \\ \text{s.t.} \quad & a \in \mathcal{A} \end{aligned} \quad (\text{OPT NLP})$$

## 10.2. Computation of Nash Equilibria

Recall that in the case  $C = R$ , Nash equilibria can be characterized by a convex program, see Sec. 3.6 for references. A technically different approach can be taken in the

<sup>4</sup>Up to the numerical inaccuracies involved when, e.g., computing eigenvalues.

general case, also leading to a characterization of Nash equilibria as optimal solutions to minimization problems, which are not necessarily *convex*, however.

**Theorem 8.** *A vector  $a \in \mathbb{R}_{\geq 0}^n$  is a Nash equilibrium if and only if there exists  $\lambda \in \mathbb{R}^N$  such that  $(a, \lambda)$  is an optimal solution to the following mathematical program.<sup>5</sup>*

$$\begin{aligned} \min \quad & \text{SC}(a) - \lambda \cdot d \\ \text{s.t.} \quad & a \in \mathcal{A} \\ & \lambda \in \mathbb{R}^N \\ & \lambda_i \leq L_S(a) \quad \forall S \in \mathfrak{S}_i \quad \forall i \in [N] \end{aligned} \tag{Nash NLP}$$

*Proof.* The objective function is always non-negative, since for all feasible  $(a, \lambda)$  we have

$$\text{SC}(a) = \sum_{S \in \mathfrak{S}} L_S(a) a_S = \sum_{i \in N} \sum_{S \in \mathfrak{S}_i} \underbrace{L_S(a)}_{\geq \lambda_i} a_S \geq \sum_{i \in N} \lambda_i \sum_{S \in \mathfrak{S}_i} a_S = \sum_{i \in N} \lambda_i d_i = \lambda \cdot d .$$

From this inequality we also immediately deduce that for a feasible  $(a, \lambda)$  we have objective function value 0 if and only if

- $L_S(a) = \lambda_i$  for all  $S \in \mathfrak{S}_i$  and all  $i \in [N]$  which have  $a_S > 0$ ,
- and, trivially (due to feasibility),  $L_S(a) \geq \lambda_i$  for all  $S \in \mathfrak{S}_i$  and all  $i \in [N]$ .

The claim now follows from Prop. 1. □

If  $L_S(\cdot)$  is concave for all  $S \in \mathfrak{S}$  and  $\text{SC}$  is convex, then (Nash NLP) is a convex program. If we have affine element latency functions, then the constraints of (Nash NLP) are linear and the objective function is quadratic. If element latency functions are affine and  $\text{SC}$  is convex, then we essentially<sup>6</sup> have a *linearly constrained convex quadratic* program, which is solvable in polynomial time by the ellipsoid algorithm [19] or, up to an arbitrarily small error, by interior-point algorithms [4, 6, 25]. But even in cases where we do not have such a nice method available, the approach via (Nash NLP) is not entirely hopeless; a vector returned by some algorithm (e.g., some solver for general non-linear programs) can be checked easily for being a Nash equilibrium by considering

<sup>5</sup>By Thm. 1 and Thm. 8 we have that the set of solutions to (VAR) coincides with the set of optimal solutions to (Nash NLP). A more general result than this was proved earlier by Aghassi, Bertsimas, and Perakis [1], using stronger tools, including LP-duality.

<sup>6</sup>The objective function of a convex quadratic program is usually specified as

$$v \mapsto v^\top A v + b^\top v, \text{ with } A \text{ being a symmetric, positive semidefinite matrix, and } b \text{ a vector.} \tag{11}$$

If  $\ell_e(x) = \theta_e x + \tau_e$  with  $\theta_e, \tau_e \in \mathbb{R}_{\geq 0}$ , for each  $e \in E$ , and  $\theta := \text{diag}(\theta_e)_{e \in E} \in \mathbb{R}^{m \times m}$ ,  $\tau := (\tau_e)_{e \in E}$ , then  $\text{SC}(a) = a^\top (R^\top \theta C)^\perp a + (R^\top \tau)^\perp a$ . As we have seen before,  $\nabla^2 \text{SC} = R^\top \theta C + (R^\top \theta C)^\perp$ , and so also  $R^\top \theta C$ , needs not to be positive semidefinite, even if  $\text{SC}$  is convex. However, if  $\text{SC}$  is convex, we can always bring it into the form of (11) using the affine mapping  $T$ . The projected Hessian times  $\frac{1}{2}$  becomes the matrix  $A$ , the vector  $b$  has to be chosen accordingly, and instead of  $\mathcal{A}$  we use  $\{v \in \mathbb{R}_{\geq 0}^{n-N}; \sum_{S \in \mathfrak{S}_i} v_S \leq d_i \quad \forall i \in [N]\}$ .

the corresponding objective value of (Nash NLP) or by checking directly whether the solution fulfills the definition of Nash equilibrium. So we can at least always determine whether some proposed solution is in fact correct.

**Extreme Nash Equilibria.** Nash equilibria need not to be unique, and two different equilibria may have different social cost, as we have seen before. We add, for some numbers  $c_0 \leq c_1$ , the additional linear constraint  $c_0 \leq \lambda \cdot d \leq c_1$  resulting in the following mathematical program.

$$\begin{aligned}
& \min \text{SC}(a) - \lambda \cdot d \\
& \text{s.t. } a \in \mathcal{A} \\
& \lambda \in \mathbb{R}^N \qquad \qquad \qquad \text{(Nash NLP')} \\
& \lambda_i \leq L_S(a) \quad \forall S \in \mathfrak{S}_i \quad \forall i \in [N] \\
& c_0 \leq \lambda \cdot d \leq c_1
\end{aligned}$$

If we can check feasibility of (Nash NLP'), we can check whether there exists a Nash equilibrium with social cost between  $c_0$  and  $c_1$ . If we can, in case of feasibility, solve (Nash NLP'), we can find a Nash equilibrium with social cost between  $c_0$  and  $c_1$ . For affine element latency functions and convex  $\text{SC}$ , this is both possible: the constraints are linear then and the objective function is convex quadratic.

Let  $\tilde{\rho}$  denote an upper bound on the price of anarchy, and assume that we know  $\text{OPT}$ . Then we know that all Nash equilibria have social cost in the interval  $[\text{OPT}, \tilde{\rho} \text{OPT}]$ . Using binary search, we can hence compute a worst or a best equilibrium with an absolute error<sup>7</sup> of  $\varepsilon$  by solving at most  $\lceil \log_2(\frac{1}{\varepsilon} (\tilde{\rho} - 1) \text{OPT}) \rceil$  instances of (Nash NLP'), provided the latter is practical. We summarize our findings in the following theorem.

**Theorem 9.** *(i) Optima and Nash equilibria are characterized by the mathematical programs (OPT NLP) and (Nash NLP), respectively.*

*(ii) Let  $c_0 \leq c_1$ . If (Nash NLP') is feasible and has optimal value 0, it characterizes all Nash equilibria with social cost between  $c_0$  and  $c_1$ . If (Nash NLP') is infeasible or has optimal value  $> 0$ , no Nash equilibria with social cost between  $c_0$  and  $c_1$  exist.*

*(iii) Let  $\tilde{\rho}$  be an upper bound on the price of anarchy. We can compute<sup>8</sup> a worst or best Nash equilibrium up to an error of  $\varepsilon$  (and possibly an additional error introduced when solving the involved mathematical programs) by solving one instance*

---

<sup>7</sup>Meaning that the Nash equilibrium found shall have social cost at least  $\text{SC}(a) - \varepsilon$  or at most  $\text{SC}(a) + \varepsilon$ , where  $a$  is a worst or a best Nash equilibrium, respectively, depending on whether we wish to approximate a worst or a best one.

<sup>8</sup>This statement is to be understood relative to the practicability of solving (OPT NLP) and (Nash NLP'). It is not intended to make any claim that we actually can solve these programs, it merely states the number of different instances that would be needed to be solved. See also statements (iv) and (v).

of (OPT NLP), and by solving or showing to be infeasible at most  $\lceil \log_2(\frac{1}{\epsilon}(\bar{\rho} - 1)\text{OPT}) \rceil$  instances of (Nash NLP').

(iv) If  $L_S(\cdot)$  is concave for all  $S \in \mathfrak{S}$  and  $\text{SC}$  is convex, then both (OPT NLP) and (Nash NLP) are convex programs.

(v) If element latency functions are affine and  $\text{SC}$  is convex, then both (OPT NLP) and (Nash NLP) are linearly constrained convex quadratic programs.

In total, we have a satisfactory computational result for affine element latency functions: by Thm. 7 we can check whether  $\text{SC}$  is convex or not, and if so, we can compute optima and extreme Nash equilibria (up to a small error) by convex quadratic programming.

Practicability may, however, depend on the number of variables – and so for our programs in particular on the number of strategies. If, e.g., the set of strategies was to enumerate *all* possible multicast path collections (or trees) in a network, this could be exponential in the number of links. Efficiently handling such applications is future work.

## 11. Brief Summary of Experimental Results and Conjecture

We conducted extensive experimental studies of the price of anarchy in NCRCGs using the computational results from the previous section. We used our own implementation, which utilizes existing NLP solvers [10, 34] as back-ends. It is beyond the scope of this article to describe the experiments or the deployed random model in detail, and so we refer to [17] for all details instead. Experiments were done on randomly generated instances with polynomial element latency functions. Effort was put into creating a *variety* of different kinds of instances. Although we treated several million instances with  $p \in \{1, 2, 3\}$ , for none of them we could prove a price of anarchy exceeding that of Thm. 3(ii) by more than 1% for  $p \in \{1, 2\}$  or by 4% for  $p = 3$ . Attributing the 1% and 4% to numerical inaccuracies, we make the following conjecture.

**Conjecture.** *The price of anarchy in an NCRCG with element latency functions drawn from  $\text{Poly}^+(p)$  is no more than*

$$\begin{cases} \gamma^p & \text{if } \gamma \geq (1+p)^{\frac{1}{p}} \\ \frac{1}{1-\gamma\beta} & \text{if } \gamma \leq (1+p)^{\frac{1}{p}} \end{cases}, \quad \text{where } \beta = p(p+1)^{-1-\frac{1}{p}}.$$

It is worth noting that the finer facets of this conjecture were obtained *during* the experiments. The starting point was a conjecture inspired by the universal bound from Thm. 4 directly applied to polynomial element latency functions, namely we conjectured an upper bound of simply  $\frac{1}{1-\beta}\gamma^p$ . Most of the subsequent refinements that lead to Thm. 3, Thm. 4, and finally to the above conjecture were inspired by experimental results.

Fig. 2 shows results for random instances and affine latency functions. A dot is drawn for each instance and positioned horizontally according to its bound given by the conjecture (using a  $\gamma$  obtained from an optimal scaling) and positioned vertically according to

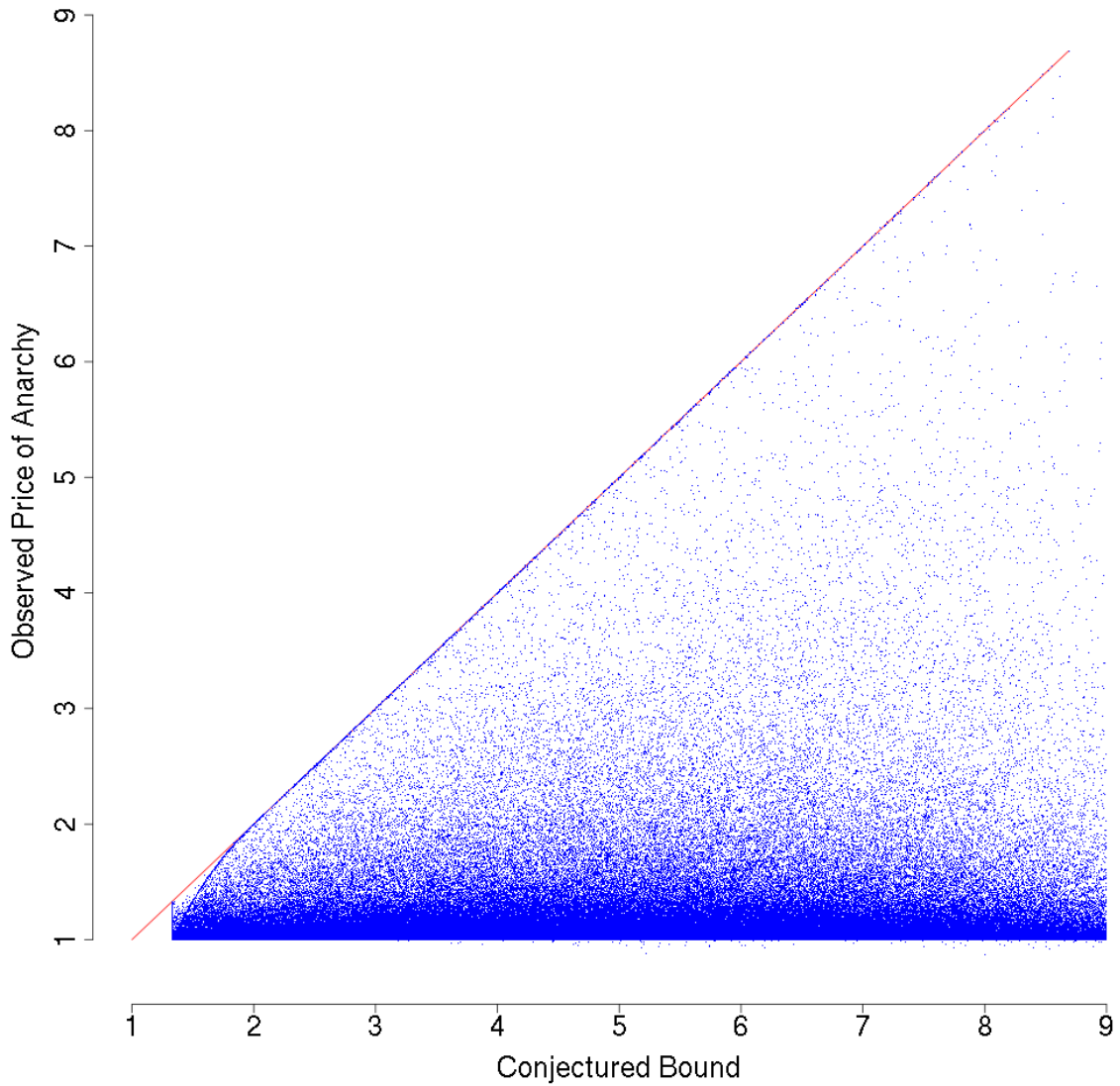


Figure 2: Scatter plot for random instances with affine element latency functions, i.e.,  $p = 1$ . A dot is drawn for each instance and positioned horizontally according to its bound given by the conjecture and positioned vertically according to the observed price of anarchy. A line marks the conjectured bound. Only those with  $\gamma$  smaller than the maximum observed price of anarchy were selected from the data set, for the sake of a better horizontal scale, resulting in approximately 650,000 instances shown out of a total of 864,000. Note that several dots are close to or on the line, but none above it. Plots and tables for larger data sets are given in [17].



the observed price of anarchy. A red line marks the conjectured bound. A dot that was clearly positioned to the upper left of that line would have constituted a counter-example to the conjecture. Some dots are positioned vertically below 1, which means that the optimum was not found and a Nash equilibrium had lower social cost than the false optimum. These dropout dots disappear when we remove instances with non-convex SC from the data set.

We speak of *observed price of anarchy* here meaning the value  $SC(a)/SC(b)$ , where  $a$  is the Nash equilibrium with highest social cost which we could find by means of trying to solve (Nash NLP') with different  $c_0$ , either in a binary search scheme or by a successively increasing  $c_0$  by a small increment, and  $b$  is the action distribution with the smallest social cost which we could find by means of trying to solve (OPT NLP). The observed price of anarchy is a lower bound on the price of anarchy; if we can treat the involved NLPs optimally or close to optimality, it equals the price of anarchy or is close to it depending on the accuracy of the NLP solutions.

The random instances presented in Fig. 2 were generated as follows: The number of elements was fixed to  $m = 4$ , the demand to  $d = 1$  (we used only one player class), the number of strategies ranges in  $n \in \{2, 3, 4, 5, 6, 9\}$ , consumption numbers were drawn from the interval  $[1.0, 9.0]$ , and relevance numbers were restricted to different intervals in different series of the experiment:  $[1.0, 1.0]$ ,  $[1.0, 5.0]$ , or  $[1.0, 9.0]$ . Choosing the most restrictive range for relevance numbers, i.e.,  $[1.0, 1.0]$  has shown to generate the most instances with an observed price of anarchy close to or on the conjectured bound. In half of the experiments, element latency functions of the form  $\ell(x) = \sum_{k=0}^p \theta_k x^k$  with random coefficients  $\theta_0, \dots, \theta_p$  were used, and for the other half, simply  $\ell(x) = x^p$  was used.

## 12. Open Problems

Challenges for future work mainly point in three directions from here.

1. Prove or disprove the conjectured upper bound on the price of anarchy.
2. Study the *price of stability*, i.e.,  $\inf_{a \text{ is N.E.}} SC(a)/OPT$ . Recall that compared to the NCG model, the NCRCG model introduces the novelty of Nash equilibria of *different social cost* and hence the price of stability may differ from the price of anarchy. As seen in Sec. 6.1 the gap between price of anarchy and price of stability may be exponential in  $p$  for element latency functions from  $\text{Poly}^+(p)$ . It appears that the basic tool, the variational inequality (VAR), which characterizes Nash equilibria, is insufficient for bounding the price of stability. New methods will be required, which hopefully might also yield a proof for the conjectured bound on the price of anarchy.
3. Develop algorithms to compute optima and Nash equilibria in certain non-quadratic or non-convex cases. Our experimental results (given in detail in [17]) can serve as a benchmark for future experimental studies: will a new algorithm be able to deliver a substantially larger fraction of instances with observed price of anarchy

near to the conjectured bound for the same random model? Will a new algorithm succeed in disproving the conjecture by the discovery of a counter-example?

Moreover, there is an additional computational challenge in the case that the number of strategies is too large in order to store  $C$  or  $R$  explicitly in memory or to treat mathematical programs that contain one variable for each strategy. In unicast routing, if all paths shall be eligible for routing, everything of interest can be expressed on edge level using flow conservation constraints. A representation as an action distribution can be reconstructed from this, if desired. It is unclear how to do something similar for general NCRCGs, however.

## Acknowledgments

I thank Prof. Srivastav for introducing me to the subject, his support, and for giving me the opportunity to participate in the DFG Priority Program 1126. I thank our student Ole Kliemann for his work on the implementation, his support while conducting the experiments, and helpful discussions. I thank Andreas Baltz and our student Sandro Esquivel for contributions in early stages of the project. And I thank the Deutsche Forschungsgemeinschaft for their financial support through Grant Sr7-3 in the Priority Program 1126 “Algorithmics of Large and Complex Networks”.

## References

- [1] Michele Aghassi, Dimitris Bertsimas, and Georgia Perakis, *Solving asymmetric variational inequalities via convex optimization.*, Operations Research Letters **34** (2006), no. 5, 481–490.
- [2] Andreas Baltz, Sandro Esquivel, Lasse Kliemann, and Anand Srivastav, *The price of anarchy in selfish multicast routing*, CAAN (Thomas Erlebach, ed.), Lecture Notes in Computer Science, vol. 4235, Springer, 2006, pp. 5–18.
- [3] Martin Beckmann, Charles B. McGuire, and Christopher B. Winsten, *Studies in the economics and transportation*, Yale University Press, 1956.
- [4] Aharon Ben-Tal and Arkadi Nemirovski, *Lectures on modern convex optimization: Analysis, algorithms, and engineering applications*, SIAM, 2001.
- [5] Matthias Blonski, *Anonymous games with binary actions*, Games and Economic Behavior **28** (1999), no. 2, 171–180.
- [6] Stephen Boyd and Lieven Vandenberghe, *Convex optimization*, Cambridge University Press, March 2004.
- [7] Dietrich Braess, *Über ein Paradoxon aus der Verkehrsplanung*, Unternehmensforschung **12** (1968), 258–268.
- [8] Dietrich Braess, Anna Nagurney, and Tina Wakolbinger, *On a paradox of traffic planning*, Transportation Science **39** (2005), 446–450, Translation from the original German.

- [9] Chi Kin Chau and Kwang Mong Sim, *The price of anarchy for non-atomic congestion games with symmetric cost maps and elastic demands*, Operations Research Letters **31** (2003), no. 3, 327–334.
- [10] Andrew R. Conn, Nicholas I.M. Gould, and Philippe L. Toint, *Lancelot: A FORTRAN package for large-scale nonlinear optimization (Release A)*, Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1992.
- [11] José R. Correa, Andreas S. Schulz, and Nicolás E. Stier Moses, *Selfish routing in capacitated networks*, Mathematics of Operations Research **29** (2004), no. 4, 961–976.
- [12] José R. Correa, Andreas S. Schulz, and Nicolás E. Stier Moses, *A geometric approach to the price of anarchy in nonatomic congestion games*, Games and Economic Behavior (2008), 457–469.
- [13] Artur Czumaj and Berthold Vöcking, *Tight bounds for worst-case equilibria*, ACM Transactions on Algorithms **3** (2007), no. 1, 1–17, Preliminary version at SODA 2002.
- [14] Stella C. Dafermos and Frederick T. Sparrow, *The traffic assignment problem for a general network*, Journal of Research of the National Bureau of Standards **73B** (1969), no. 2, 91–118.
- [15] Eric J. Friedman, *Dynamics and rationality in ordered externality games*, Games and Economic Behavior **16** (1996), no. 1, 65–76.
- [16] David Kinderlehrer and Guido Stampacchia, *An introduction to variational inequalities and applications*, Academic Press, New York, NY, 1980.
- [17] Lasse Kliemann, *Experimental studies of the price of anarchy in consumption-relevance congestion games*, Tech. Report 0815, Institut für Informatik, Christian-Albrechts-Universität Kiel, 2008.
- [18] Lasse Kliemann and Anand Srivastav, *Models of non-atomic congestion games – from unicast to multicast routing*, Abschlussband DFG-SPP 1126 “Algorithmik großer und komplexer Netzwerke”, Springer-Verlag, 2008, In press.
- [19] Mikhail K. Kozlov, Sergei P. Tarasov, and Leonid G. Khachiyan, *Polynomial solvability of convex quadratic programming*, Soviet Mathematics Doklady **20** (1979), 1108–1111.
- [20] Thomas Lücking, Marios Mavronicolas, Burkhard Monien, and Manuel Rode, *A new model for selfish routing*, Theoretical Computer Science **406** (2008), no. 3, 187–206, Preliminary version at STACS 2004.

- [21] Thomas Lücking, Marios Mavronicolas, Burkhard Monien, Manuel Rode, Paul G. Spirakis, and Imrich Vrto, *Which is the worst-case Nash equilibrium?*, MFCS (Branislav Rován and Peter Vojtás, eds.), Lecture Notes in Computer Science, vol. 2747, Springer, 2003, pp. 551–561.
- [22] Igal Milchtaich, *Social optimality and cooperation in nonatomic congestion games*, Journal of Economic Theory **114** (2004), no. 1, 56–87.
- [23] Anna Nagurney, *Network economics: A variational inequality approach*, Springer, 1999.
- [24] John F. Nash, *Non-cooperative games*, Annals of Mathematics **54** (1951), 289–295.
- [25] Yuri Nesterov and Arkadi Nemirovski, *Interior-point polynomial algorithms in convex programming*, third printing ed., Studies in Applied Mathematics, SIAM, 2001.
- [26] Christos H. Papadimitriou, *Algorithms, games, and the Internet*, STOC, ACM, 2001, Extended abstract at ICALP 2001, pp. 749–753.
- [27] Georgia Perakis, *The price of anarchy when costs are non-separable and asymmetric*, Mathematics of Operations Research **32** (2007), 614–628, Preliminary version at ICALP 2004.
- [28] Arthur C. Pigou, *The economics of welfare*, Macmillan, London, 1920.
- [29] Robert W. Rosenthal, *A class of games possessing pure-strategy Nash equilibria*, International Journal of Game Theory **2** (1973), no. 1, 65–67.
- [30] Tim Roughgarden, *Selfish routing*, Ph.D. thesis, Cornell University, Ithaca, NY, USA, 2002.
- [31] Tim Roughgarden and Éva Tardos, *Bounding the inefficiency of equilibria in nonatomic congestion games*, Games and Economic Behavior **47** (2004), no. 2, 389–403.
- [32] David Schmeidler, *Equilibrium points of nonatomic games*, Journal of Statistical Physics **7** (1973), no. 4, 295–300.
- [33] M.J. Smith, *The existence, uniqueness and stability of traffic equilibria*, Transportation Research Part B: Methodological **13** (1979), no. 4, 295–304.
- [34] Andreas Wächter and Lorenz T. Biegler, *On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming*, Mathematical Programming **106** (2006), no. 1, 25–57.
- [35] John G. Wardrop, *Some theoretical aspects of road traffic research*, Proceedings of the Institution of Civil Engineers, Part II **1** (1952), 325–362.

