

INSTITUT FÜR INFORMATIK
UND PRAKTISCHE MATHEMATIK

**The Price of Anarchy in Selfish Multicast
Routing**

Andreas Baltz, Lasse Kliemann, Anand Srivastav

Bericht Nr. 0510

Juni 2005



CHRISTIAN-ALBRECHTS-UNIVERSITÄT

KIEL

The Price of Anarchy in Selfish Multicast Routing

Andreas Baltz*, Lasse Kliemann†, Anand Srivastav
{aba,lki,asr}@numerik.uni-kiel.de

Institut für Informatik und Praktische Mathematik, CAU Kiel
Christian-Albrechts-Platz 4
24118 Kiel

June 2005

*Supported by DFG.

†Supported by DFG.

Abstract

We study the price of anarchy for selfish multicast routing games in directed multigraphs with latency functions on the edges, extending the known theory for the unicast situation, and exhibiting new phenomena not present in the unicast model. In the multicast model we have N commodities, where for each $i = 1, \dots, N$, a flow from a source s_i to a finite number of terminals $t_i^1, \dots, t_i^{k_i}$ has to be routed such that every terminal t_i^j receives flow $n_i \in \mathbb{R}_{\geq 0}$. The flow demand of n_i for each commodity is under the control of infinitely many users (or players), each of them controlling a negligible fraction of it and trying to selfishly find the fastest route possible. Each of the players can choose from a set of strategies, which describe different routing decisions. For commodity i , a strategy is a set of s_i - t_i^j -paths ($j = 1, \dots, k_i$). In the unicast model, $k_i = 1$ for all i . A *flow* (or *action distribution*) is a mapping that assigns to each strategy the amount of users choosing that strategy. A value $f(S)$ assigned to a strategy S in the multicast model can be realized in two different ways: either every s_i - t_i^j -path of the strategy carries flow $f(S)$, which is the usual notion of flows (inspired by physical transport), or we allow the network nodes to multiply duplicate incoming flows, which is characteristic for data transfer. Duplication can be useful in order to reduce the load of high-latency edges. We consider both kinds of realizations.

Nash flows (or *Nash equilibria*) are special flows in which no player has an incentive to unilaterally deviate from his current strategy. To evaluate the overall performance of a flow, we have a function called the *social cost* of the flow. The *price of anarchy* is the quotient between the social cost of a worst-case Nash equilibrium and an optimal flow.

We start the analysis by showing (via variational inequalities) that multicast games admit Nash equilibria. One of the significant results of this paper is the proof of an upper bound for the price of anarchy of $\frac{4}{3} \cdot \frac{\nu^2}{\nu^*}$ in some variants of multicast routing for linear edge latencies. We also prove a lower bound of ν , so we have upper and lower bounds that are tight up to a factor of $\frac{4}{3}\nu$. Here, ν and ν^* are network and strategy dependent parameters reflecting the maximum/minimum consumption of the network. Both are 1 in the unicast case. For edge-latencies being polynomials of degree p , we give examples with price of anarchy at least ν^p , which in the general situation with $\nu > 1$, is an exponential increase compared to the Roughgarden bound of $O(p/\ln p)$ for the unicast model. This exhibits the contrast to the unicast case, where we have Roughgarden's (2002) result that the price of anarchy is independent of the network topology. To our knowledge this paper is the first thorough study of the price of anarchy in the multicast scenario. The approach may lead to further research extending game-theoretic network analysis to models used in applications.

1. Introduction

Multicast routing in communication networks is a natural and practically relevant extension of the so far quite well studied unicast routing. Among the applications of multicast routing are the transmission of music, movies, conferences, or any other popular content, that is requested by several customers at a time. A formal description of our multicast routing model needs many technical definitions. We keep the introduction on a more informal level and refer the reader to Section 2 for all necessary details.

Problem Formulation. An instance of selfish multicast routing consists of a directed multigraph $G = (V, E)$, where the edges are also called links, a set of N player classes, called commodities, where commodity i is characterized by a source s_i and terminals (or sinks) $t_i^1, \dots, t_i^{k_i}$, and a (flow) demand of $n_i \in \mathbb{R}_{\geq 0}$. The links are each equipped with a latency function $l_e : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$. For commodity i , a set $S = \{P_1, \dots, P_{k_i}\}$ where P_j is an s_i - t_i^j -path, is called a strategy. The task is to realize for every commodity i , a flow in the network from s_i to all terminals $t_i^1, \dots, t_i^{k_i}$, satisfying the demand n_i for every terminal. We think of the demand as being under control of infinitely many players, each controlling a negligible fraction and selfishly trying to find the fastest route for it. This game-theoretic model is known as the Wardrop model.

In the unicast model, $k_i = 1$ for all i , so we have a collection of single source/single sink commodities, and every strategy S consists of one path only. In the multicast case there are two different ways to route the flow $f(S)$ assigned to a strategy S for commodity i : either, we route $f(S)$ on each path, which is the usual notion of flows satisfying the Kirchhoff conservation law (here shortly called *conservation flow*), or we allow multiple duplication of flow at certain nodes: a link which serves several, say r , terminals in a strategy, i.e., a link contained in r paths of that strategy, only needs to transmit the data once, not r times. That is because the data can later be duplicated to serve all terminals. In this way, the congestion on the links can be reduced. We call such a flow *duplication flow*.

The cost of a flow is defined by $\text{SC}(f) = \sum_{S \in \mathfrak{S}} l_S(f) f(S)$, where \mathfrak{S} is the set of all strategies of all commodities, $f(S) \leq n_i$ is the portion of the demand that by the decision of the selfish players has been allocated to strategy S , and $l_S(f)$ is the strategy latency for S . We study four different definitions for l_S , which all coincide in the unicast case. Together with the two types of flows (conservation and duplication), we thus have 8 variants of multicast. The price of anarchy for a multicast instance \mathcal{I} is $\rho(\mathcal{I}) = \sup_f \frac{\text{SC}(f)}{\text{SC}(f^*)}$, where f ranges over all Nash equilibria and f^* is an optimal flow. A Nash equilibrium is a flow in which no player (meaning: no portion of the flow, however small)

has an incentive to unilaterally deviate from his current strategy.

Previous and Related Work. By the pioneering work of Roughgarden [6] and Roughgarden and Tardos [8] we know that $\rho(\mathcal{I})$ in the unicast model for latency functions being polynomials of degree p , is bounded from above by $O(\frac{p}{\ln p})$ (and is $\frac{4}{3}$ for $p = 1$). As already an example of a 2-parallel links network has a price of anarchy of $\frac{4}{3}$ (for $p = 1$), the surprising conclusion is that it is *independent* of the network topology [6, Sec. 3.4].

Our Results. A solid foundation for the analysis of multicast routing games is given in Section 2. In Section 2.1 we introduce a concise model, and in Section 2.2 we show, using results on variational inequalities, the existence of Nash equilibria.

As a main result, we show in Section 3 that the price of anarchy in multicast routing may depend heavily on the network topology and the strategies. Certain edges of the graph may be utilized under certain strategies more than others, although the players on those strategies are not charged for this. On the other hand, some strategies may depend highly on some edges but only contribute a small amount to their utilization. See Remark 4.1 for a more detailed discussion of this. To capture the effects of this phenomenon, which does not occur in unicast routing, we introduce for each edge and strategy an integer called the *consumption*. Moreover we introduce two new invariants for a graph G and a set of strategies \mathfrak{S} , which we call *maximum (resp. minimum) consumption number*, $\nu = \nu(G, \mathfrak{S})$ resp. $\nu^* = \nu^*(G, \mathfrak{S})$. We have $\nu = 1 = \nu^*$ in unicast routing. We show in Section 3.3 that in two variants of multicast for linear latency functions, the price of anarchy is at most $\frac{4}{3} \frac{\nu^2}{\nu^*}$ and in Section 3.2 provide a lower bound for one of these variants of ν (with $\nu^* = 1$). So, we have here a gap of $\frac{4}{3}\nu$.

For latency functions being polynomials of degree p , we present (also in Section 3.2) a multicast instance with price of anarchy at least ν^p . As in general $\nu > 1$, the ν^p bound is exponentially larger than the corresponding unicast bound of $O(\frac{p}{\ln p})$. This is surprising (and disappointing from the point of view of a company running the network). For instances using the advantages of duplication flows in order to de-load high-latency links, the cost of the global optimum decreases drastically, but unfortunately, due to selfish behavior, the users grab (greedily) certain links without a lookahead and block them out, so that the cost of the Nash equilibrium still stands high. For other definitions of strategy latency, in Section 4 we are able to prove that results from non-atomic congestion games, i.e., bounds of the form $O(\frac{p}{\ln p})$, carry over.

Open Problems. A couple of interesting open problems arise from this paper. For example, can the $\frac{4}{3}\nu$ factor gap between upper and lower bound

be closed? Furthermore, is the price of anarchy for polynomials of degree p of order $O(\nu^p)$, matching the lower bound? It would also be interesting to consider polynomial time algorithms for the computation of equilibria.

An ambitious task would be to study multicast for information flows with duplication *and* coding facilities of the network. Such networks are the state-of-the-art in today's engineering designs. Our work can be considered as a first step in this direction.

2. Basics of Multicast Routing

2.1. Model and Instances

An instance of selfish multicast routing consists of the following.

- A directed multigraph $G = (V, E)$. The edges are also called *links*.
- A set of N *player classes* (or *user classes*). Sometimes, player classes are also called *commodities*. Each player class is characterized by a demand n_i and a vector of vertices $(s_i; t_i^1, \dots, t_i^{k_i})$, where s_i is the source, and the $t_i^1, \dots, t_i^{k_i}$ are the terminals.
- The demand n_i is supposed to be routed from s_i to each of the terminals $t_i^1, \dots, t_i^{k_i}$. We think of the demand as being under control of infinitely many players, each of them controlling a negligible amount of it. This is the well-known Wardrop model (see, e.g., [6, Sec. 2.2]), which will become clearer when we define flows and Nash equilibria below.
- Each link $e \in E$ in the graph is equipped with a latency function $l_e : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$. We always assume each latency function to be non-decreasing and *standard* [6]. This means that it is differentiable and $\xi \mapsto l_e(\xi)\xi$ is convex.

If an amount ξ of traffic is to be routed through the link e , each unit of flow will take $l_e(\xi)$ time to traverse e . Hence we have a total latency of $l_e(\xi)\xi$ on that link.

- For $i \in [N] = \{1, \dots, N\}$, we call a set of paths $S := \{P_1, \dots, P_{k_i}\}$ where P_j is a path connecting s_i with t_i^j for $j \in [k_i]$, a *strategy*. Note that for unicast routing $k_i = 1$ for all i . The set of all strategies we wish to allow for player class i is denoted by \mathfrak{S}_i . We assume¹ that $\mathfrak{S}_i \cap \mathfrak{S}_j = \emptyset$ for all $i, j \in [N]$. Let $\mathfrak{S} := \bigcup_{i \in [N]} \mathfrak{S}_i$.
- An *action distribution* (according to [8]), simply called *flow*, is a map $f : \mathfrak{S} \rightarrow \mathbb{R}_{\geq 0}$ such that all the demands are met, i.e., $\sum_{S \in \mathfrak{S}_i} f(S) =$

¹Otherwise we have to treat \mathfrak{S} as a multiset.

$n_i \quad \forall i \in [N]$. A flow can be understood as a partition of each of the real intervals $[0, n_i]$. Each of these intervals represents the continuum of infinitely many players of the corresponding player class. The quantity $f(S)$ gives, for each $S \in \mathfrak{S}_i$, the portion of demand that by the decision of the players from that class is routed according to that particular strategy S .

As described in the introduction, the routing of a flow in the multicast model can be done in two different ways: we can route the demand with flows in the usual sense (conservation flows) or with flows allowing duplication (duplication flows).

- Let $e \in E$ and $S \in \mathfrak{S}$. We define the *consumption* of e under S as $c(e, S) := |\{P \in S; e \in P\}|$, i.e., the consumption is the number of paths in S traversing e , or in other words, the number of terminals served via e in this strategy.

- The *congestion* f_e of a link e with respect to a flow f is the amount of traffic that link e has to process. The total latency of a link e hence is $l_e(f_e)f_e$. Each instance defines the congestion in one of the following ways, depending on whether we have conservation flows or duplication flows.

$$f_e := \begin{cases} \sum_{S \in \mathfrak{S}(e)} c(e, S)f(S) & \text{conserv. flow} \\ \sum_{S \in \mathfrak{S}(e)} f(S) & \text{dupl. flow} \end{cases} \quad (1)$$

Here, $\mathfrak{S}(e)$ denotes the set of all strategies that contain a path which in turn contains e .

- We denote by $l_S(f)$ the so-called latency of strategy S with respect to a flow f . In unicast $l_S(f)$ is simply the sum of the latencies in the single path of which the strategy S consists².

Let $S = \{P_1, \dots, P_{k_i}\} \in \mathfrak{S}$. As in [6], the latency of a path P under f is defined by

$$l_P(f) := \sum_{e \in P} l_e(f_e).$$

By $E(S)$ we denote the union of the edges in all the paths in S . Note that we consider $E(S)$ *not* as a multiset, so edges do not appear multiple times even if they lie in several paths.

We introduce the following four definitions of the latency of a strategy S .

$$\begin{aligned} l_S^{\text{edges}}(f) &:= \sum_{e \in E(S)} l_e(f_e), & l_S^{\text{paths}}(f) &:= \sum_{P \in S} l_P(f), \\ l_S^{\text{paths avg}}(f) &:= \frac{1}{|S|} l_S^{\text{paths}}(f), & l_S^{\text{max}}(f) &:= \max_{P \in S} l_P(f) \end{aligned} \quad (2)$$

²The maximum over all links in the path has also been studied [5].

An instance of selfish multicast routing includes one of these strategy latency functions.

Remark 2.1. a) For unicast routing, all four definitions coincide.

b) It is easy to see that $l_S^{\text{paths}}(f) = \sum_{e \in E(S)} c(e, S) l_e(f_e)$.

2.2. Nash Equilibria, Social Cost, Price of Anarchy

A flow f is called a *Nash equilibrium* (sometimes abbreviated *NE*), if

$$f(S_1) > 0 \implies l_{S_1}(f) \leq l_{S_2}(f) \quad \forall S_1, S_2 \in \mathfrak{S}_i \quad \forall i \in [N]. \quad (3)$$

Hence, in a Nash equilibrium, only minimum-latency strategies are used, since then no player has an incentive to choose a different strategy (provided the rest of the players keep their current decision). If each l_S is continuous (which will be the case during all our studies), then the game admits at least one Nash equilibrium. This follows from the characterization of Nash equilibria as the solutions to a certain variational inequality (see Theorem 2.4 and the discussion after that).

We define the *social cost* of a flow f as $\text{SC}(f) := \sum_{S \in \mathfrak{S}} l_S(f) f(S)$. The social cost captures the overall performance of the system for a given flow f . We will always assume that our instances admit a flow f^* with minimum social cost and that $\text{SC}(f^*) > 0$. Existence is guaranteed if all l_S are continuous (which will be the case in our studies), because the set of flows is compact. For an instance \mathcal{I} of selfish multicast routing with optimal flow f^* , define the price of anarchy by

$$\rho(\mathcal{I}) := \sup_{f \text{ is NE}} \frac{\text{SC}(f)}{\text{SC}(f^*)}.$$

Nash equilibria have a very simple structure, as seen in the following proposition. The proof for this is straightforward.

Proposition 2.2. *Let f be a Nash equilibrium. Then, for every $i \in [N]$, there exists a real number $l_i(f)$ such that $l_S(f) = l_i(f)$ for all $S \in \mathfrak{S}_i$, whenever $f(S) > 0$, and no strategy in \mathfrak{S}_i has latency less than $l_i(f)$.*

Corollary 2.3. *Let f be a Nash equilibrium. Then $\text{SC}(f) = \sum_{i \in [N]} l_i(f) n_i$.*

We now aim for further characterizations of Nash equilibria. Let f, \tilde{f} be flows. Define $\text{SC}^f(\tilde{f}) := \sum_{S \in \mathfrak{S}} l_S(f) \tilde{f}(S)$. The first part of the following

theorem is well-known for the unicast case, see, e.g., [6, Lem. 3.3.7] or [1] and the references therein. The whole theorem also holds in a more general context than multicast routing, for it (and its proof) does not require the notion of congestion.

Theorem 2.4. *a) Let f be a flow. Then f is a Nash equilibrium if and only if we have*

$$\text{SC}^f(\tilde{f}) \geq \text{SC}(f) \quad \text{for all flows } \tilde{f}. \quad (4)$$

b) Let each l_S be continuous. Then the multicast game admits at least one Nash equilibrium.

PROOF. We refer the reader to the appendix for the proof of a). For b) note that (4) is equivalent to $\sum_{S \in \mathfrak{S}} l_S(f)(\tilde{f}(S) - f(S)) \geq 0$ for all flows \tilde{f} . This is a well-studied variational inequality. It has been shown in [2] with deep results from the index theory of vector fields that it admits at least one solution. ■

Note that all strategies from (2) are continuous, because we only consider standard latency functions and because the congestion is a continuous mapping. Hence, all our multicast games admit at least one Nash equilibrium.

In the rest of the paper we investigate the price of anarchy for conservation flows resp. duplication flows and the four strategy latencies from (2). These are 8 cases. In Section 3 we show for 5 of them that the price of anarchy depends on the network topology, while in two other cases it does not (Section 4).

3. Price of Anarchy Dependent on the Network Topology

For a directed graph G and a set of strategies \mathfrak{S} we define the *minimum and maximum consumption number* as

$$\nu^*(G, \mathfrak{S}) := \min_{S \in \mathfrak{S}} \min_{e \in E(S)} c(e, S), \quad \nu(G, \mathfrak{S}) := \max_{S \in \mathfrak{S}} \max_{e \in E(S)} c(e, S).$$

We write just ν^* and ν , if we are dealing with only one instance at a time. In a unicast situation there is only one path in each strategy and hence $\nu^*(G, \mathfrak{S}) = 1 = \nu(G, \mathfrak{S})$.

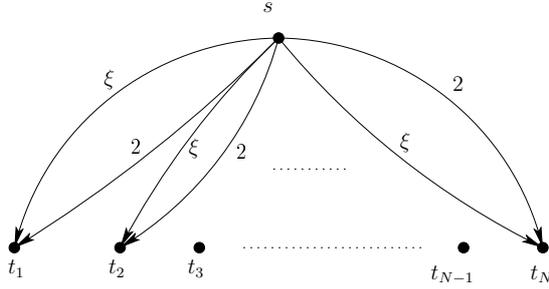


Figure 1: Example $\mathcal{I}^{r,R,N}$. The values r and R are non-negative real numbers. ξ denotes the latency function $l : \xi \mapsto \xi$.

3.1. Latency $l^{\text{paths avg}}$ and l^{max}

Consider the instance $\mathcal{I}^{r,R,N}$ in Figure 1. In this instance, we have $N + 1$ player classes, all sharing the source s .

- For each $i \in [N]$, player class i has one terminal t_i . The demand is r , for a fixed $r \in \mathbb{R}_{\geq 0}$.
- Player class $N + 1$ has N terminals t_1, \dots, t_N . This class has demand R , $R \in \mathbb{R}_{\geq 0}$ fixed.

There are two links between s and each t_i , one with constant latency $r + R$ and one with latency function $l : \xi \mapsto \xi$. Defining a strategy in this example for player class $N + 1$ means picking for each t_i either the ξ -link or the link with constant latency. Conservation flows and duplication flows coincide, since each strategy consists of edge disjoint paths.

Theorem 3.1. *By choosing r sufficiently small and R and N sufficiently large, the price of anarchy in the example in Figure 1 can be made arbitrarily high (although we still have linear latency functions). More precisely, we have for large N*

$$\rho(\mathcal{I}^{r,R,N}) \geq \left(1 + \frac{R}{r}\right)(1 - o(1)).$$

This holds for all four combinations of conservation flows and duplication flows on the one hand, and $l^{\text{paths avg}}$ and l^{max} on the other hand.

PROOF. Let f be the flow where all players from all classes choose the links with latency function $\xi \mapsto \xi$. We then have $l_S^{\text{paths avg}}(f) = r + R$ for all strategies S . The same holds for l^{max} . Hence, f is a Nash flow for $l^{\text{paths avg}}$ and l^{max} . By Corollary 2.3 we have $\text{SC}(f) = \left(\sum_{i \in [N]} (r + R)r\right) + (r + R)R =$

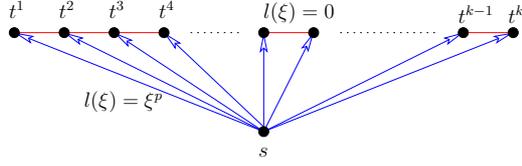


Figure 2: The links in the upper path can be used in both directions and have latency 0.

$Nr(r+R) + (r+R)R$. For comparison, take \tilde{f} as the flow in which all players from classes 1 to N stick to the ξ -links, but all players from class $N+1$ take the links with constant latency $r+R$. We then have (for $l^{\text{paths avg}}$ and l^{max}) $\text{SC}(\tilde{f}) = \left(\sum_{i \in [N]} r \cdot r\right) + (r+R)R = Nr^2 + (r+R)R$. Hence, the price of anarchy is at least

$$\rho(\mathcal{I}^{r,R,N}) \geq \frac{Nr(r+R) + (r+R)R}{Nr^2 + (r+R)R}. \quad (5)$$

Since the term from (5) tends to $1 + \frac{R}{r}$ as $N \rightarrow \infty$, the claim follows. \blacksquare

3.2. Duplication Flows and l^{paths}

Theorem 3.2. *There are examples of selfish multicast routing instances with duplication flows using l^{paths} and latency functions that are polynomials of degree at most p (with non-negative coefficients) where the price of anarchy is at least ν^p .*

PROOF. Consider the instance in Figure 2. We are given one player class with k terminals t^1, \dots, t^k . The demand is 1. Note that in this example, $\nu = k$. The latency functions on the links on the upper path are identically 0, and these links may be used in any direction. The links of the form (s, t^j) all have latency function $\xi \mapsto \xi^p$ for a fixed p . A Nash equilibrium f is achieved if all players use the tree consisting of the edges $(s, t^1), \dots, (s, t^k)$. This flow has social cost $\text{SC}(f) = k$.

Now, a better flow \tilde{f} is given as follows. For every $j \in [k]$ let a fraction of $\frac{1}{k}$ users ‘inject’ their flow into the upper path via the edge (s, t^j) . Each of these strategies has latency $k \frac{1}{k^p} = k^{1-p}$, because the edge (s, t^j) has congestion $\frac{1}{k}$ and hence a latency of $\frac{1}{k^p}$, and this edge is contained in k paths. Because there are k such strategies, each of them carrying $\frac{1}{k}$ units of flow, we have a social cost of $\text{SC}(\tilde{f}) = k k^{1-p} \frac{1}{k} = k^{1-p}$.

Hence the price of anarchy ρ of this instance is at least $\rho \geq \frac{\text{SC}(f)}{\text{SC}(\tilde{f})} = k k^{p-1} = k^p$. \blacksquare

In Theorem 4.4 we will see that the same instance for conservation flows (and l^{paths}) has a price of anarchy bounded by $O(p)$. So, we have an exponential increase in the price of anarchy when we switch to duplication. This is due to the much better optimum which takes advantage of duplication. However, the ν^p bound from Theorem 3.2 says that due to uncoordinated behavior of the players, the benefit of duplication is completely neglected.

Summarizing, we considered the cases of conservation flows and $l^{\text{paths avg}}$ or l^{max} , duplication flows and $l^{\text{paths avg}}$ or l^{max} , and duplication flows and l^{paths} . For these cases, the price of anarchy — even from a worst-case point of view — is dependent on the network topology and the set of strategies, i.e., we can construct instances with arbitrarily high prices of anarchy for some fixed class of latency functions.

3.3. Upper Bounds for Linear Latency Functions

We consider the following two cases: a) conservation flows and l^{edges} , b) duplication flows and l^{paths} . We also restrict ourselves to linear latency functions with non-negative coefficients, i.e.,

$$l_e(\xi) = \lambda_e \xi + \mu_e \quad \forall e \in E \quad \forall \xi \in \mathbb{R}_{\geq 0}, \quad (6)$$

where $\lambda_e, \mu_e \in \mathbb{R}_{\geq 0}$. The following holds for case a) as well as b) (see appendix).

Proposition 3.3. *Let f, \tilde{f} be flows. Then $\sum_{e \in E} l_e(f_e) \tilde{f}_e \leq \nu \sum_{e \in E} l_e(\tilde{f}_e) \tilde{f}_e + \frac{1}{4\nu} \sum_{e \in E} l_e(f_e) f_e$.*

3.3.1. Conservation Flows and l^{edges}

We first prove an upper bound on the total latency $\sum_{e \in E} l_e(f_e) f_e$. See the appendix for a proof.

Lemma 3.4. *For every flow f we have $\sum_{e \in E} l_e(f_e) f_e \leq \nu \cdot \text{SC}(f)$.*

The proof of the following theorem uses methods from [1, Th. 2.1.].

Theorem 3.5. *Let \mathcal{I} be an instance of selfish multicast routing using conservation flows and l^{edges} . Then $\rho(\mathcal{I}) \leq \frac{4\nu^2}{3\nu^*}$.*

PROOF. Let f be a Nash equilibrium and \tilde{f} some flow. By Theorem 2.4 we get

$$\begin{aligned} \text{SC}(f) &\leq \text{SC}^f(\tilde{f}) = \sum_{S \in \mathfrak{S}} l_S^{\text{edges}}(f) \tilde{f}(S) = \sum_{S \in \mathfrak{S}} \sum_{e \in E(S)} l_e(f_e) \tilde{f}(S) = \sum_{e \in E} \sum_{S \in \mathfrak{S}(e)} l_e(f_e) \tilde{f}(S) \\ &= \frac{1}{\nu^*} \sum_{e \in E} l_e(f_e) \sum_{S \in \mathfrak{S}(e)} \nu^* \tilde{f}(S) \leq \frac{1}{\nu^*} \sum_{e \in E} l_e(f_e) \sum_{S \in \mathfrak{S}(e)} c(e, S) \tilde{f}(S) = \frac{1}{\nu^*} \sum_{e \in E} l_e(f_e) \tilde{f}_e. \end{aligned}$$

By Proposition 3.3 and Lemma 3.4 we get

$$\begin{aligned} \frac{1}{\nu^*} \sum_{e \in E} l_e(f_e) \tilde{f}_e &\leq \frac{1}{\nu^*} \nu \sum_{e \in E} l_e(\tilde{f}_e) \tilde{f}_e + \frac{1}{\nu^*} \frac{1}{4\nu} \sum_{e \in E} l_e(f_e) f_e \\ &\leq \frac{1}{\nu^*} \nu \sum_{e \in E} l_e(\tilde{f}_e) \tilde{f}_e + \frac{1}{4\nu} \sum_{e \in E} l_e(f_e) f_e \leq \frac{\nu^2}{\nu^*} \text{SC}(\tilde{f}) + \frac{1}{4} \text{SC}(f). \end{aligned}$$

In total, it follows that $\frac{3}{4} \text{SC}(f) \leq \frac{\nu^2}{\nu^*} \text{SC}(\tilde{f})$, hence $\frac{\text{SC}(f)}{\text{SC}(\tilde{f})} \leq \frac{4\nu^2}{3\nu^*}$. The claim follows by using an optimal flow as \tilde{f} . \blacksquare

3.3.2. Duplication Flows and l^{paths}

In a similar way as in the previous section, 3.3.1, we can prove for duplication flows and l^{paths} :

Lemma 3.6. *For every flow f we have $\nu^* \sum_{e \in E} l_e(f_e) f_e \leq \text{SC}(f)$.*

Theorem 3.7. *Let \mathcal{I} be an instance of selfish multicast routing using duplication flows and l^{paths} . Then $\rho(\mathcal{I}) \leq \frac{4\nu^2}{3\nu^*}$.*

4. Price of Anarchy Independent of the Network Topology

The proofs in this section are based on identifying the given multicast game with a *non-atomic congestion game with separable latencies* (*separable NCG* for short). These games are well studied [8]. We first describe known results on such games, and their connection to our multicast game.

4.1. Non-atomic Congestion Games with Separable Latencies

A separable NCG is defined similar to our multicast game, however, there is no graph required. We are simply given a set E of elements. Each of the elements has a latency function l_e . A strategy is a subset of E . By $\mathfrak{S}(e)$ we denote all strategies containing element e . Crucial for a NCG are its definitions of congestion and strategy latency. For each $e \in E$ and $S \in \mathfrak{S}$ let a_{eS} be some non-negative real number. For a given flow, the congestion is defined by

$$f_e := \sum_{S \in \mathfrak{S}(e)} a_{eS} f(S), \quad (7)$$

and the latency of a strategy $S \in \mathfrak{S}$ is defined as

$$l_S(f) := \sum_{e \in S} a_{eS} l_e(f_e). \quad (8)$$

Nash equilibria, social cost and price of anarchy are defined as in our multicast games. An important point is that the factors a_{eS} appear both in congestion and in strategy latency. This allows us [8, Prop. 2.8] to write the social cost as $\text{SC}(f) = \sum_{e \in E} l_e(f_e) f_e$.

Remark 4.1. Comparing this to multicast games with duplication flows and l^{paths} , we note that they do not have this kind of symmetry. Instead, they carry factors like the a_{eS} (the consumptions, in fact), but only in the definition of strategy latency. These factors do not appear in the congestion. As we have seen in Section 3.2, the lack of such a symmetry can lead to high prices of anarchy, that may even depend on the network topology. Other variants, like conservation flows and l^{edges} have the consumption as a factor in the congestion, but not in the strategy. Having high factors in the strategy and relatively small factors in the congestion means that players choosing that strategy suffer highly under a high congestions on this edge, but do not contribute much to that congestion. This can happen in multicast with duplication flows and l^{paths} with respect to edges that serve many terminals (in the chosen strategy).

On the other hand, high factors in the congestion but only relatively small factors in the strategy latency means that although the players on this strategy induce a high utilization on a link, they are not charged accordingly. This can happen in multicast with conservation flows and l^{edges} with respect to edges that serve many terminals (in the chosen strategy).

These observations give rise to the question about a new, more general class of congestion games, in which we have different factors in the strategy latencies and the congestion.

The price of anarchy for a separable NCG with standard latency functions is well understood [8]. Let \mathcal{L} be a class of standard latency functions that contains at least one non-zero function. We call such a class also *standard*. Note, that we will always assume that our multicast games have at least one link with a non-zero latency function. Hence, if we consider the class of all latency functions for a given instance, this class is automatically standard. Now, it is easy to see that for each $l \in \mathcal{L}$ there exists $\beta_l : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ such that $l^*(\beta_l(\xi)) = l(\xi)$ for all $\xi \in \mathbb{R}_{\geq 0}$, where $l^*(\xi) := l'(\xi)\xi + l(\xi)$. The *anarchy value* of \mathcal{L} [8, Def. 4.3], see also [6, Def. 3.3.2], is

$$\alpha(\mathcal{L}) := \sup_{0 \neq l \in \mathcal{L}, \xi > 0, l(\xi) > 0} \frac{l(\xi)\xi}{l(\xi)\xi + l(\beta_l(\xi))\beta_l(\xi) - l(\xi)\beta_l(\xi)}.$$

The amazing property of separable NCGs with latency functions from a standard class \mathcal{L} is that their price of anarchy can be upper-bounded in terms $\alpha(\mathcal{L})$ — a parameter that is completely independent of the structure of \mathfrak{G} .

Theorem 4.2 (Roughgarden, Tardos [8]). *Let Γ be a separable NCG with latency functions from the standard class \mathcal{L} . Then $\rho(\Gamma) \leq \alpha(\mathcal{L})$.*

For polynomial latency functions the anarchy values are well-known.

Corollary 4.3 ([8]). *Let Γ be a separable NCG with latency functions that are polynomials of degree at most p with non-negative coefficients. Then $\rho(\Gamma) \leq (1 - p(p+1)^{-(p+1)/p})^{-1}$. For linear latency functions, i.e., if $p = 1$, the quantity on the right hand side is $\frac{4}{3}$. For $p \rightarrow \infty$, it is $\Theta(\frac{p}{\ln p})$, yielding a rough bound of $O(p)$.*

The bound in Theorem 4.2 is tight in the following sense: let \mathcal{L} be a standard class containing the constant functions³. Then we can find for every $\epsilon > 0$, a separable NCG Γ with latency functions from \mathcal{L} and $\rho(\Gamma) > \alpha(\mathcal{L}) - \epsilon$. This means that — from a worst-case point of view — the price of anarchy in separable NCGs is independent of the strategy structure and the demands.

4.2. Duplication Flows and l^{edges} , Conservation Flows and l^{paths}

Theorem 4.4. *Let \mathcal{L} be standard. Let \mathcal{I} be an instance of selfish multicast routing with latency functions from \mathcal{L} , then we have $\rho(\mathcal{I}) \leq \alpha(\mathcal{L})$ if we consider a) duplication flows and use l^{edges} , or b) conservation flows and use l^{paths} .*

³This requirement can even be relaxed; see [8] for details.

PROOF. We only consider case a), i.e., duplication flows and strategy latencies l^{edges} , and refer to the appendix for the second case. The first case can be modeled as a separable NCG as follows: we take the links in E as elements and as latency functions for the elements we take the latency functions of the links. A strategy S is identified with $E(S)$, so that it can be seen as a subset of the set of elements. For all $e \in E$ and $S \in \mathfrak{S}$ set

$$a_{eS} := \begin{cases} 1 & \text{if } e \in E(S) \\ 0 & \text{otherwise.} \end{cases}$$

It is straightforward to see that this NCG is exactly the multicast game. ■

4.3. Equal Consumptions

We know that in the unicast case $\nu = 1 = \nu^*$. We now relax this condition to the following.

$$c(e, S) = c(e, T) \quad \forall e \in E \forall S, T \in \mathfrak{S}(e). \quad (\#)$$

We call (#) the *equal consumption condition*. Under this condition, we have the following theorem. The proof of this is based on the same idea as the ones in the previous section (although it requires some more steps) and has hence been moved into the appendix.

Theorem 4.5. *Let \mathcal{L} be standard. Let \mathcal{I} be an instance of selfish multicast routing with latency functions from \mathcal{L} . Then $\rho(\mathcal{I}) \leq \alpha(\mathcal{L})$, if we have the equal consumption condition and either a) conservation flows and use l^{edges} , or b) duplication flows and use l^{paths} .*

An important observation is the following. Reconsider the example in Figure 2 with $k = 2$. Then, the equal consumption condition is violated for some links by the minimum possible amount, i.e., by 1. As we have seen in Section 3.2, this already may cause an exponential jump in the price of anarchy. If the equal consumption condition was true, we would have a price of anarchy of $O(p)$. But instead we have $\Omega(2^p)$.

5. Future Work

We considered the price of anarchy for several variants of selfish multicast routing and pointed out some cases where the price of anarchy depends on the network topology. For most of these cases, we have lower bounds. We have also upper bounds for linear latency functions in two cases. An ongoing effort

is to find upper bounds for other variants and more general latency functions and to tighten the existing bounds. We are also interested in establishing bicriteria bounds like in [6, Sec. 3.6]. Another aim is a further study of the generalized congestion games mentioned in Remark 4.1.

We have seen examples with a very high price of anarchy. This motivates the design of mechanisms to induce more efficient equilibria, e.g., through taxation [3].

Finally, we intend to leave the Wardrop model, which assumes an infinite number of players, and turn towards a finite number of players, as in the well-known KP-model [4]. For unicast routing on general topologies, approaches in that direction have been made recently [7].

References

- [1] José R. Correa, Andreas S. Schulz, and Nicolás E. Stier Moses. Selfish routing in capacitated networks. *Mathematics of Operations Research*, pages 961–976, November 2004. URL <http://columbia.edu/~ns2224/papers/css-capsoueMORpublished.pdf>.
- [2] Philip Hartman and Guido Stampacchia. On some non-linear elliptic differential-functional equations. *Acta Mathematica*, 115:271–310, 1966.
- [3] George Karakostas and Stavros G. Kolliopoulos. Edge pricing of multicommodity networks for heterogeneous selfish users. In *Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science (FOCS'04)*, 2004. URL <http://www.cas.mcmaster.ca/%7Egk/papers/focs04.pdf>.
- [4] Elias Koutsoupias and Christos H. Papadimitriou. Worst-case equilibria. In *Proceedings of the 16th International Symposium on Theoretical Aspects of Computer Science (STACS'99)*, volume 1563 of *Lecture Notes in Computer Science*, pages 404–413, March 1999. URL <http://www.cs.berkeley.edu/~christos/nash.ps>.
- [5] Lavy Libman and Ariel Orda. Atomic resource sharing in noncooperative networks. *Telecommunication Systems*, 17(4):385–409, 2001.
- [6] Tim Roughgarden. *Selfish Routing*. PhD thesis, Cornell University, May 2002. URL <http://theory.stanford.edu/~tim/papers/thesis.ps>.
- [7] Tim Roughgarden. Selfish routing with atomic players. In *Proceedings of the 15th Annual ACM–SIAM Symposium on Discrete Algorithms*

(SODA'04), 2004. URL <http://theory.stanford.edu/~tim/papers/split.ps>.

- [8] Tim Roughgarden and Éva Tardos. Bounding the inefficiency of equilibria in nonatomic congestion games. *Games and Economic Behavior*, 47(2): 389–403, 2004. URL <http://theory.stanford.edu/~tim/papers/ncg.ps>.

Appendix

A. Some Special Cases

Edge Disjoint Paths. Everything that holds for conservation flows carries over to duplication flows, provided we consider strategies with edge disjoint paths (i.e., strategies S such that for all paths $P, Q \in S$ we have $E(P) \cap E(Q) = \emptyset$). For then, duplication is never used.

Equal Number of Terminals. Consider conservation flows, $l^{\text{paths avg}}$, and the special case that $k_i = k_j$ for all $i, j \in [N]$, and call the common number of terminals k . We already know that conservation flows and l^{paths} can be modeled as a separable NCG, see Theorem 4.4. All we now have to do is replace each l_e by the function $\xi \mapsto \frac{1}{k}l_e(\xi)$. This does not change the anarchy value of the class of used latency functions. On the other hand, l^{paths} for the modified instance is the same as $l^{\text{paths avg}}$ for the original instance. Hence, we can apply Theorem 4.4.

B. Proofs

B.1. Proof of Theorem 2.4 a)

Let first f be a Nash equilibrium. Fix an arbitrary flow \tilde{f} . Then by Corollary 2.3 we have

$$\begin{aligned} & \text{SC}^f(\tilde{f}) - \text{SC}(f) \\ &= \sum_{i \in [N]} \sum_{S \in \mathfrak{G}_i} \underbrace{l_S(f)}_{\geq l_i(f) \text{ by Proposition 2.2}} \tilde{f}(S) - \sum_{i \in [N]} l_i(f)n_i \\ &\geq \sum_{i \in [N]} \left(\left(\sum_{S \in \mathfrak{G}_i} l_i(f)\tilde{f}(S) \right) - l_i(f)n_i \right) \end{aligned}$$

$$\begin{aligned}
&= \sum_{i \in [N]} \left(\left(l_i(f) \underbrace{\sum_{S \in \mathfrak{G}_i} \tilde{f}(S)}_{=n_i} \right) - l_i(f)n_i \right) \\
&= 0.
\end{aligned}$$

Now, let f be a flow such that (4) holds. Fix $i_0 \in [N]$ and pick $S_0 \in \mathfrak{G}_{i_0}$ such that

$$l_{S_0}(f) = \min_{S \in \mathfrak{G}_{i_0}} l_S(f) =: l_{i_0}(f).$$

Define \tilde{f} by

$$\tilde{f}(S) := \begin{cases} n_{i_0} & \text{if } S = S_0 \\ 0 & \text{if } S \in \mathfrak{G}_{i_0} \setminus \{S_0\} \\ f(S) & \text{otherwise} \end{cases}.$$

Then, by (4) we have

$$\begin{aligned}
0 &\leq \sum_{i \in [N]} \sum_{S \in \mathfrak{G}_i} l_S(f) \underbrace{(\tilde{f}(S) - f(S))}_{=0 \text{ if } i \neq i_0} \\
&= \sum_{S \in \mathfrak{G}_{i_0}} l_S(f) (\tilde{f}(S) - f(S)) \\
&= l_{S_0}(f)n_{i_0} - \sum_{S \in \mathfrak{G}_{i_0}} l_S(f)f(S) \\
&= l_{i_0}(f)n_{i_0} - \sum_{S \in \mathfrak{G}_{i_0}} \underbrace{l_S(f)}_{\geq l_{i_0}(f)} f(S).
\end{aligned}$$

Because $\sum_{S \in \mathfrak{G}_{i_0}} f(S) = n_{i_0}$, it follows that $l_S(f) = l_{i_0}(f)$ whenever $f(S) > 0$, $S \in \mathfrak{G}_{i_0}$. Because i_0 was taken arbitrarily, f is a Nash equilibrium. \blacksquare

B.2. Proof of Proposition 3.3

We have

$$\sum_{e \in E} l_e(f_e) \tilde{f}_e = \sum_{e \in E} (\lambda_e f_e + \mu_e) \tilde{f}_e = \frac{1}{\nu} \sum_{e \in E} (\nu \lambda_e f_e \tilde{f}_e + \nu \mu_e \tilde{f}_e).$$

Since $0 \leq (\nu \tilde{f}_e - \frac{1}{2} f_e)^2$, we have

$$\nu f_e \tilde{f}_e \leq \nu^2 \tilde{f}_e^2 + \frac{1}{4} f_e^2.$$

Hence

$$\begin{aligned}
& \frac{1}{\nu} \sum_{e \in E} (\nu \lambda_e f_e \tilde{f}_e + \nu \mu_e \tilde{f}_e) \leq \frac{1}{\nu} \sum_{e \in E} (\nu^2 \lambda_e \tilde{f}_e^2 + \frac{1}{4} \lambda_e f_e^2 + \nu \mu_e \tilde{f}_e) \\
& \leq \nu \sum_{e \in E} (\lambda_e \tilde{f}_e + \mu_e) \tilde{f}_e + \frac{1}{4\nu} \sum_{e \in E} \lambda_e f_e^2 \leq \nu \sum_{e \in E} l_e(\tilde{f}_e) \tilde{f}_e + \frac{1}{4\nu} \sum_{e \in E} l_e(f_e) f_e.
\end{aligned}$$

■

B.3. Proof of Lemma 3.4

$$\begin{aligned}
\sum_{e \in E} l_e(f_e) f_e &= \sum_{e \in E} l_e(f_e) \sum_{S \in \mathfrak{S}(e)} c(e, S) f(S) \leq \sum_{e \in E} l_e(f_e) \sum_{S \in \mathfrak{S}(e)} \nu \cdot f(S) \\
&= \nu \sum_{e \in E} \sum_{S \in \mathfrak{S}(e)} l_e(f_e) f(S) = \nu \sum_{S \in \mathfrak{S}} \sum_{e \in E(S)} l_e(f_e) f(S) = \nu \sum_{S \in \mathfrak{S}} l_S^{\text{edges}}(f) f(S) = \nu \cdot \text{SC}(f).
\end{aligned}$$

■

B.4. Proof of Theorem 4.4 b)

We consider conservation flows and strategy latencies l^{paths} . This can be modeled as a separable NCG in a similar way as before. We set

$$a_{eS} := c(e, S) \tag{9}$$

for each $e \in E$ and $S \in \mathfrak{S}$. When we compute the congestion using (7) we get the right expression for the congestion for conservation flows, see (1). We already stated in Remark 2.1 b) that $l_S^{\text{paths}}(f) = \sum_{e \in E(S)} c(e, S) f(S)$. By (8) and (9) we get exactly l^{paths} as the strategy latency. ■

B.5. Proof of Theorem 4.5

a) First consider conservation flows and l^{edges} . We will model this as a separable NCG. Again, as elements we take the links in E and as latency functions for the elements we take the latency functions of the links. A strategy S is identified with $E(S)$, so that it can be seen as a subset of the set of elements.

For each $e \in E$ which lies in a path included in some strategy, denote by c_e the common value $c(e, S)$ of all strategies $S \in \mathfrak{S}(e)$. Set c_e arbitrarily, e.g., $c_e := 0$, if e is not used in any strategy. Define new latency functions for all $e \in E$:

$$\tilde{l}_e : \mathbb{R}_{\geq 0} \longrightarrow \mathbb{R}_{\geq 0}, \xi \longmapsto l_e(c_e \xi), \tag{10}$$

and corresponding multipliers $\tilde{a}_{eS} := 1$ if $e \in E(S)$ and 0 otherwise.

The NCG $\tilde{\Gamma}$ defined by the $(\tilde{l}_e)_{e \in E}$ and the $(\tilde{a}_{eS})_{\substack{e \in E \\ S \in \mathfrak{S}}}$ models our multicast game. To see this, we observe that by (7) and (8)

$$\begin{aligned}
\tilde{l}_S(f) &= \sum_{e \in E(S)} \tilde{a}_{eS} \tilde{l}_e \left(\underbrace{\sum_{T \in \mathfrak{S}(e)} \tilde{a}_{eT} f(T)}_{\text{cong. in } \tilde{\Gamma}} \right) \\
&= \sum_{e \in E(S)} \tilde{l}_e \left(\sum_{T \in \mathfrak{S}(e)} f(T) \right) \\
&= \sum_{e \in E(S)} l_e(c_e \sum_{T \in \mathfrak{S}(e)} f(T)) \\
&= \sum_{e \in E(S)} l_e \left(\sum_{T \in \mathfrak{S}(e)} c(e, T) f(T) \right) \\
&= \sum_{e \in E(S)} l_e(f_e) = l_S^{\text{edges}}(f).
\end{aligned}$$

The transformation in (10) preserves the anarchy value of the class of used latency functions. Hence, the bound of Theorem 4.2 holds, if we take for \mathcal{L} the latency functions on the links of the network.

b) Now consider duplication flows and l^{paths} . We proceed as above, but define new latency functions for all $e \in E$:

$$\tilde{l}_e : \mathbb{R}_{\geq 0} \longrightarrow \mathbb{R}_{\geq 0}, \xi \longmapsto c_e l_e(\xi), \quad (11)$$

and corresponding multipliers $\tilde{a}_{eS} := 1$ if $e \in E(S)$ and 0 otherwise. The NCG defined by the $(\tilde{l}_e)_{e \in E}$ and the $(\tilde{a}_{eS})_{\substack{e \in E \\ S \in \mathfrak{S}}}$ models our multicast game. It is straightforward to see that in the NCG, we get exactly the same congestions as in our multicast game. We also have the right individual latencies, since

$$\begin{aligned}
\tilde{l}_S(f) &= \sum_{e \in E(S)} \tilde{l}_e(f_e) \\
&= \sum_{e \in E(S)} c_e l_e(f_e) \\
&= \sum_{e \in E(S)} c(e, S) l_e(f_e) \\
&= l_S^{\text{paths}}(f) \quad \text{by Remark 2.1 b).}
\end{aligned}$$

The transformation in (11) preserves the anarchy value of the class of used latency functions, so we can use Theorem 4.2. ■

C. Summary Table

We summarize some of our most important results on the price of anarchy in Table 1.

	conservation	duplication
l^{edges}	<ul style="list-style-type: none"> • $\frac{4}{3} \frac{\nu^2}{\nu^*}$ for $p = 1$ • $\Theta\left(\frac{p}{\ln p}\right)$ (top. cond.) 	<ul style="list-style-type: none"> • $\frac{4}{3}$ for $p = 1$ • $\Theta\left(\frac{p}{\ln p}\right)$
l^{paths}	<ul style="list-style-type: none"> • $\frac{4}{3}$ for $p = 1$ • $\Theta\left(\frac{p}{\ln p}\right)$ 	<ul style="list-style-type: none"> • $\frac{4}{3} \frac{\nu^2}{\nu^*}$ for $p = 1$ • $\Omega(\nu^p)$ • $\Theta\left(\frac{p}{\ln p}\right)$ (top. cond.)
$l^{\text{paths avg}}$	$\Omega\left(\frac{R}{r}\right)$	$\Omega\left(\frac{R}{r}\right)$
l^{max}	$\Omega\left(\frac{R}{r}\right)$	$\Omega\left(\frac{R}{r}\right)$

Table 1: Price of anarchy for different variants of multicast and latency functions being polynomials of degree p with non-negative coefficients. The ‘topological condition’ refers to the equal consumption condition.