SIAM J. Comput.
Vol. 39, No. 5, pp. 1957–1987

SELFISH TRAFFIC ALLOCATION FOR SERVER FARMS*

ARTUR CZUMAJ[†], PIOTR KRYSTA[‡], AND BERTHOLD VÖCKING[§]

Abstract. We study the price of selfish routing in noncooperative networks like the Internet. In particular, we investigate the price of selfish routing using the price of anarchy (a.k.a. the coordination ratio) and other (e.g., bicriteria) measures in the recently introduced game theoretic parallel links network model of Koutsoupias and Papadimitriou. We generalize this model toward general, monotone families of cost functions and cost functions from queueing theory. A summary of our main results for general, monotone cost functions is as follows: 1. We give an exact characterization of all cost functions having a bounded/unbounded price of anarchy. For example, the price of anarchy for cost functions describing the expected delay in queueing systems is unbounded. 2. We show that an unbounded price of anarchy implies an extremely high performance degradation under bicriteria measures. In fact, the price of selfish routing can be as high as a bandwidth degradation by a factor that is linear in the network size. 3. We separate the game theoretic (integral) allocation model from the (fractional) flow model by demonstrating that even a very small or negligible amount of integrality can lead to a dramatic performance degradation. 4. We unify recent results on selfish routing under different objectives by showing that an unbounded price of anarchy under the min-max objective implies an unbounded price of anarchy under the average cost objective and vice versa. Our special focus lies on cost functions describing the behavior of Web servers that can open only a limited number of Transmission Control Protocol (TCP) connections. In particular, we compare the performance of queueing systems that serve all incoming requests with servers that reject requests in case of overload. Our analysis indicates that all queueing systems without rejection cannot give any reasonable guarantee on the expected delay of requests under selfish routing even when the injected load is far away from the capacity of the system. In contrast, Web server farms that are allowed to reject requests can guarantee a high quality of service for every individual request stream even under relatively high injection rates.

Key words. traffic allocation, selfish routing, price of anarchy, game theory, server farms

AMS subject classifications. 60K30, 68M20, 68M10, 90B18, 90B20, 90B22, 91A10

DOI. 10.1137/070693862

1. Introduction.

The *price of anarchy* (also known as the coordination ratio) is a game theoretic measure introduced by Koutsoupias and Papadimitriou in [16] to reflect the price of selfish routing in *noncooperative network systems* like the *Internet.* In large-scale communication networks it is typically impossible to regulate network traffic to achieve an optimal performance of the system. In such networks, a common strategy of individual users is to follow the most rational approach, that is, to behave selfishly to maximize its own profits. This motivates investigations in game theoretical flavor of the relationship between systems in which each user is aware of the situation facing all other users and trying to optimize its own strategy (that is, being in a *Nash*

[*] Received by the editors June 7, 2007; accepted for publication (in revised form) November 16, 2009; published electronically February 3, 2010. A preliminary version of this paper appeared in *Proceedings of the 34th ACM Symposium on Theory of Computing (STOC'02)*, Montreal, Canada, 2002, pp. 287–296. This research was partially supported by NSF grant CCR-0105701, DFG grants Kr 2332/1-1 and Vo 889/1-1, IST program of the EU under contract IST-1999-14186 (ALCOM-FT), and EPSRC award EP/D063191/1 (Centre for Discrete Mathematics and Its Applications).

http://www.siam.org/journals/sicomp/39-5/69386.html

[†] Department of Computer Science and the Centre for Discrete Mathematics and Its Applications (DIMAP), University of Warwick, Coventry CV4 7AL, United Kingdom (A.Czumaj@warwick.ac.uk).

[‡] Department of Computer Science, University of Liverpool, Liverpool L69 3BX, United Kingdom (p.krysta@csc.liv.ac.uk).

[§] Department of Computer Science, RWTH Aachen, Ahornstr. 55, 52074 Aachen, Germany (voecking@cs.rwth-aachen.de).

equilibrium) and optimal strategies in such systems.

It is well known that Nash equilibria do not always optimize the overall performance. Therefore, in order to understand the phenomenon of noncooperative systems, Koutsoupias and Papadimitriou [16] initiated investigations of the behavior of the *price of anarchy* (or the worst-case coordination ratio), which is the ratio between the worst possible Nash equilibrium and the social (i.e., overall) optimum. In other words, this analysis seeks the price of uncoordinated individual utility-maximizing decisions (hence "the price of anarchy"). Koutsoupias and Papadimitriou [16] proposed that model to study routing problems in which a set of several agents want to send traffic from the sources to destinations along parallel links with linear cost functions.

In this paper, we generalize the results for the *routing problems for the model with parallel links* and *provide the first thorough study of the price of anarchy under more realistic cost functions*. Our goal is to investigate families of cost functions that are defined as generally as possible and our main motivation comes from the problem of allocating traffic in the model of a *Web server farm*, for which parallel links are the natural model. We consider agents maintaining Web server farms and offering the service of data storage for content providers (e.g., to store large images and other embedded files, which usually make up most of the traffic on the Internet). The request data streams, which would normally go to the content providers, must now be redirected to the new Web servers. This defines a load balancing problem in which streams must be mapped to the servers such that a high quality of service can be guaranteed for every stream. An important aspect that has to be taken into account is that different streams might have different characteristics, e.g., caused by different file lengths. For practical studies that investigate the reasons for and impacts of this variable traffic, see, e.g., [4, 6, 7, 22, 23].

Our special focus lies on cost functions describing the behavior of Web servers that can open only a limited number of Transmission Control Protocol (TCP) connections. In particular, we compare the performance of queueing systems that serve all incoming requests with servers that reject requests in case of overload. Our analysis indicates that all queueing systems without rejection cannot give any reasonable guarantee on the expected delay of requests under selfish routing even when the injected load is far away from the capacity of the system. In contrast, the systems that are allowed to reject requests can guarantee a high quality of service for every individual request stream, even under relatively high injection rates.

We present the first exhaustive study of the price of anarchy for the parallel links model under more realistic cost functions, with the main emphasis on the applications for Web server farms. In most of our studies, we assume that data streams can be described by a single weight (depending on the rate of requests and the expected length of the requested files). In this case, the load on a server is defined to be the sum of the weights of the streams mapped to the server. However, the cost occurring at a server can depend in a nontrivial way on this load; e.g., we consider general, continuous, and nondecreasing cost functions.

Furthermore, we generalize our analysis to the study of selfish routing for parallel links under more complex cost functions that depend not only on the sum of the injected rates but also on distributions of session length that might be different for different streams. We point out that our approach is not completely general in the sense that it still assumes Poisson arrivals of requests in every individual stream. However, we are able to derive the first rigorous analysis for selfish routing taking into account the heterogeneous nature of Internet traffic in the form of general session length distributions.

Outline of the paper. Section 2 contains the description of a part of the model that is not related to game theory, whereas section 3 introduces concepts related to game theory. Previous and related work is outlined in section 4. Section 5 presents our results concerning general monotone cost functions, and section 6 contains our results about queueing theoretical cost functions. Finally, section 7 has some concluding remarks.

We hold in this paper the policy that in order to follow the discussion of our new results and their consequences, the reader does not need to go into the proofs of these results presented in sections 5 and 6. Thus, reading the content of sections 5 and 6, skipping the proofs therein, gives a precise description of our results.

2. Description of the routing problem.

The routing problem described above can be formally defined as a scheduling problem with m servers and n data streams. The set of servers and streams is denoted by $[m]$ and $[n]$, respectively, where $[N]$ stands for the set $\{1, 2, \ldots, N\}$. The data streams shall be mapped to the servers such that a cost function (describing, for example, waiting time or denial of service) is minimized. The particular cost functions that we consider will depend on the traffic and server model that will be defined next in this section. In general, we will compare the solution (i.e., a mapping of the streams to servers) achieved by selfish agents with the optimal solution to a min-max optimization problem based on the notion of the price of anarchy that will be defined later in this section.

2.1. Traffic model.

Traffic streams are assumed to be sequences of requests for service. We consider sequences of stochastic nature. We assume that requests are issued by independent customers and are Poisson. Let r_i denote the *injection rate* of data stream i. The *lengths* of the sessions of stream i are determined by a probability distribution $\mathcal{D}_i$, i.e., the *service time* or *session length distribution* of stream i. We define the *weight* of stream i to be $\lambda_i = r_i \cdot \mathbf{E}[\text{ session length with respect to } \mathcal{D}_i\ ]$.

A *server farm* is a set of m servers, all using the same policy to serve requests. Different servers, however, may have different performance depending on their *bandwidths.* We denote the bandwidth of server j by b_j. We assume that $b_1 \geq b_2 \geq \cdots \geq b_m$, unless we explicitly state a precise assumption on the values of b_j's. Also, unless otherwise stated, we will assume that streams are *unsplittable*; that is, every stream must be mapped to exactly one server.

2.2. Server models: Sequential and parallel servers.

We distinguish between two general server farm models, one consisting of m *sequential servers*, each of which has a single service channel that can serve requests one after the other, and another of m *parallel servers*, each of which has multiple service channels so that it can handle a certain number of requests in parallel.

For a *sequential server* the bandwidth corresponds to the speed or service rate of its service channel. That is, the service channel of server j needs time ℓ/b_j to serve a session of length ℓ. If a session arrives when serving another session, then this session is stored in a first-come-first-served (FCFS) queue associated with the server. As soon as the service channel finishes its session, it starts serving the next session in the queue. The *waiting time* of a session is the time a session has to wait in the queue. The *system time* of a session is its waiting time plus its service time. If the mapping of the streams to the servers is fixed, then queueing theory allows us to derive formulas for the expected waiting time and the expected system time of a server as a function of the injection rates and session length distributions of the streams mapped to the server. We assume that the social objective is to minimize the maximum expected waiting or, alternatively, system time over all servers.

A *parallel server* has multiple service channels, and we assume that all of them have the same, uniform service rate. On each of its channels it can serve a session independently from other channels. Each channel serves requests with service rate 1; that is, the time a channel needs to serve a session is equal to the length of the session. The number of channels of server j corresponds to its bandwidth, and it is denoted by b_j. For example, it may correspond to the number of TCP connections that can be opened simultaneously on a Web server. In the *queueing model*, requests arriving at a server that cannot be handled immediately as all channels of the server are blocked are put into an FCFS queue. This queue is shared among all channels of the server. Like in the case of sequential servers, we aim at minimizing the maximum expected waiting or system time over all servers. Alternatively, in the *rejection model*, blocked requests are rejected and disappear from the system. A natural objective in this model is to minimize the number or the fraction of rejected requests.

In queueing theoretical notation, a server with k channels that delays blocked requests corresponds to a so-called M/D/k/∞ or, for short, M/D/k queue, where D corresponds to the service time (session length) distribution of the injected request stream. When blocked customers are rejected, then the corresponding queue is denoted by M/D/k/k. For a more detailed discussion about queueing theory background, we refer the reader to standard literature; see, e.g., [9, 11, 12].

2.3. Cost functions and families of cost functions. The cost occurring at the servers under some fixed *allocation* (i.e., a mapping of streams to servers) is defined by *families of cost functions* $\mathcal{F}_B = \{f_b | b \in B\}$, where B denotes the domain of possible bandwidth values and f_b describes the cost function for servers with bandwidth $b \in B$. Typically, we will assume $B = \mathbb{R}_{>0}$ or $B = \mathbb{N}_{>0}$, but, additionally, sometimes we will study finite domains of bandwidth. For example, a collection of identical servers with some specified bandwidth b can be described by a family of cost functions $\mathcal{F}_B = \{f_b\}$.

A cost function f is called *simple* if it depends only on the injected load, that is, if the cost of a server is a function of the sum of the streams' weights mapped to the server (but does not depend on other characteristics such as, e.g., the session length distribution). A simple cost function is called *monotone* if it is nonnegative, continuous, and nondecreasing. For an ordered set B, a family of simple cost functions $\mathcal{F}_B$ is called *monotone* if (i) f_b is monotone for every $b \in B$ and (ii) the cost functions are nonincreasing with b, i.e., $f_b(\lambda) \geq f_{b'}(\lambda)$ for every $\lambda \geq 0$ and $0 < b \leq b'$. A typical example of a monotone family of cost functions we will consider is derived from the formula for the expected system time (delay) on an M/M/1 server with injection rate λ and service rate b, namely,

$$\frac{1}{b - \min\{b, \lambda\}}.$$

The question of finding the price of selfish routing under cost functions of this form was already posed in the seminal paper by Koutsoupias and Papadimitriou [16]. This cost function is, however, only one particular example of cost functions that we will investigate; our analysis is applicable to a very general class of cost functions.

3. Preliminaries in game theory. In this section we present game theoretic aspects of our model in more detail. We distinguish between the *integral* allocation model and the *fractional* flow model.

3.1. Integral allocation model. The *integral* (or atomic) *allocation model* distinguishes between mixed and pure strategies. The set of *pure strategies* for agent $i \in [n]$ is $[m]$; that is, a pure strategy maps every stream to exactly one server. The mapping is described by a matrix $\mathcal{X} = (x_i^j)_{i\in[n],j\in[m]}$, where x_i^j is an indicator variable with $x_i^j = 1$ if stream (agent) i chooses strategy (server) j and 0 otherwise. A *mixed strategy* is a probability distribution over pure strategies. Let p_i^j denote the probability that agent i maps its stream to server j. We define the *load* of server j under an allocation $(x_1^j, \ldots, x_n^j)$ by $w_j = \sum_{i=1}^n \lambda_i \cdot x_i^j$ and the *cost* of server j by

$$C_j \;=\; f_{b_j}(x_1^j, \ldots, x_n^j).$$

Observe that w_j and C_j are random variables depending on the probabilities p_i^j. Let ℓ_j denote the *expected cost* on server j, that is, $\ell_j = \mathbf{E}\,[C_j]$. For a stream i, the *expected cost of stream i on server j* is defined by $c_i^j = \mathbf{E}\,[C_j \,|\, x_i^j = 1]$.

In this paper, we focus our attention on the problem of minimization of the maximum cost of the servers. In this model, we define the *social cost C* by

$$C \;=\; \mathbf{E}\left[\max_{j\in[m]} C_j\right]$$

and the *optimal cost (social optimum) opt* to be the minimum cost of a pure strategy,

$$opt \;=\; \min_{\mathcal{X}} \;\max_{j\in[m]} \;\; f_{b_j}(x_1^j, \ldots, x_n^j),$$

where the minimum is taken over all matrices $\mathcal{X} = (x_i^j)_{i\in[n],j\in[m]}$ such that $x_i^j \in \{0,1\}$ for $i \in [n], j \in [m]$ and $\sum_{j=1}^m x_i^j = 1$ for $i \in [n]$.

In order to formalize the notion of rational and selfish behavior of the system users, we introduce Nash equilibria of server farms. We say the probabilities $(p_i^j)_{i\in[n],j\in[m]}$ define a (system in a) *Nash equilibrium* if and only if $p_i^j > 0$ implies $c_i^j \le c_i^q$ for every $i \in [n]$ and $j, q \in [m]$. Thus, a Nash equilibrium is characterized by the property that there is no incentive for any agent to change its strategy. Then, in such a model, the *price of anarchy*, for a fixed set of servers and streams, is defined by $\max \frac{C}{opt}$, where the maximum is over all Nash equilibria. In other words, the price of anarchy specifies by which factor the cost of the system can increase due to selfish behavior. For a fixed family of cost functions $\mathcal{F}_B$, the *price of anarchy R* determines the maximum price of anarchy over all possible server farms, typically described as an asymptotic function in m.

In our study, we will identify several instances of cost functions for which R is unbounded. In the case when R is unbounded, we will investigate the *bicriteria price of anarchy $\overline{R}$*. Let opt_Γ denote the value of an optimal solution assuming that all injection rates r_i are increased by a factor of Γ. Then the *bicriteria price of anarchy* $\overline{R}$ is defined to be the smallest Γ satisfying $C \le opt_\Gamma$ over all Nash equilibria. In other words, we ask by which factor the injected load must be decreased so that the worst-case cost in Nash equilibrium cannot exceed the optimal cost for the original rates.

3.2. Fractional flow model. The motivation behind the *fractional* (nonatomic) *flow model* is to assume that every stream consists of infinitely many units, each carrying an infinitesimal (and thus negligible) amount of flow. Each such unit behaves in a selfish way. Intuitively, we expect each such unit to be allocated (selfishly) to a server promising minimum cost, taking into account the behavior of other units of flow. Assuming infinitesimal small units of flow, we obtain the *fractional* variant of the integral allocation model (this model was also considered in [10, 24, 25, 26, 29]).

To define the model formally, in the fractional model the variables x_i^j that describe the mapping of streams to servers can take arbitrary *real* values from $[0,1]$ subject to the constraint $\sum_{j\in[m]} x_i^j = 1$ for every $i \in [n]$. The model does not distinguish between mixed and pure strategies, and, in particular, the load on server j is simply defined by $w_j = \sum_{i\in[n]} \lambda_i \cdot x_i^j$. The *social cost* C in this case is defined by

$$C \;=\; \max_{j\in[m]} C_j,$$

and the *optimal cost (social optimum) opt* is

$$opt \;=\; \min_{\mathcal{X}} \max_{j\in[m]} \;\; f_{b_j}(x_1^j,\ldots,x_n^j),$$

where the minimum is taken over all matrices $\mathcal{X} = (x_i^j)_{i\in[n],j\in[m]}$ such that $x_i^j \in [0,1]$ for $i \in [n], j \in [m]$, and for each $i \in [n]$, we have $\sum_{j=1}^m x_i^j = 1$.

There are several equivalent ways to define a Nash equilibrium in this model. We use the characterization of Wardrop [33]; see also [28, 29]. A fractional allocation is in *Nash equilibrium* if $x_i^j > 0$ implies $C_j \le C_q$, for every $i \in [n]$ and $j, q \in [m]$. The *price of anarchy* is defined analogously to the integral allocation model and is denoted by R^*.

3.3. Integral allocations with negligible weights. The fractional flow model is a simplification of the integral allocation model that aims to model the situation in which each stream carries only a negligible fraction of the total load. We will investigate in detail the relationship between fractional flow and integral allocations of streams with tiny weights. For this purpose we define the notion of "ϵ-small streams." Assuming a server farm with identical servers and homogeneous traffic, we have the following definition. Stream i is called *ϵ-small* if

$$\lambda_i \;\le\; \frac{\epsilon}{m} \sum_{i'\in[n]} \lambda_{i'},$$

that is, the stream has at most an $\frac{\epsilon}{m}$-fraction of the overall weight.

In the case of heterogeneous servers or heterogeneous traffic we need a slightly more technical definition. Let us fix a server farm and a set of streams with positive weights. Let opt^* denote the minimum of maximum cost over all fractional flow allocations. For a stream $i \in [n]$, define the *scaled stream i* to be a stream with rate $r_i' = r_i/\epsilon$ and session length distribution $\mathcal{D}_i' = \mathcal{D}_i$. (Observe that this implies that the weight of the scaled stream is $\lambda_i' = \lambda_i/\epsilon$.) Then stream i is called *ϵ-small* if, for every $j \in [m]$, the cost of server j is at most opt^*, assuming this server gets assigned only the scaled stream i and no other stream.

We define R_ϵ to be the price of anarchy under the restriction that all streams are ϵ-small.

3.4. Average-cost objective functions. Besides the min-max objective introduced above, we shall also briefly consider another objective function that has been investigated by Roughgarden and Tardos [29]. The *average-cost objective function* aims at minimizing the expected weighted average cost over all streams (see also, e.g., [8, 31] for related results). Formally, the cost under this objective function is defined by

$$C_{ave} \;=\; \frac{1}{\lambda} \sum_{j \in [m]} \mathbf{E}\left[w_j \cdot C_j\right],$$

and the social optimum is defined by

$$opt_{ave} \;=\; \min_{\mathcal{X}} \left(\frac{1}{\lambda} \sum_{j \in [m]} w_j \cdot f_{b_j}(x_1^j, \ldots, x_n^j) \right),$$

where $\lambda = \sum_{i \in [n]} \lambda_i$ is the total injected weight and the minimum is taken over the same matrices $\mathcal{X}$ as in the definitions of the social optimum under the min-max objective in sections 3.1–3.3.

We shall consider the price of anarchy for various types of average-cost objective functions similarly as for the min-max objective function. In particular, the price of anarchy for average-cost objective functions is defined in the same way as for the min-max model considered before; the only difference is that now one compares average social cost with average social optimum. The price of anarchy for integral, fractional, ϵ-small, and bicriteria integral allocation under this objective function is denoted by Σ, Σ^*, Σ_ϵ, and $\overline{\Sigma}$, respectively.

4. Previous research.

The model investigated in this paper has been introduced in a seminal paper of Koutsoupias and Papadimitriou [16]. In that paper, only the integral allocation model is studied and only for linear cost functions (that is, functions of the form $f_b(x_1, \ldots, x_n) = \sum_{i=1}^n \lambda_i x_i / b$). The motivation behind studying these cost functions was to investigate the price of selfish routing under simple cost functions. Koutsoupias and Papadimitriou [16] give some basic results for the price of anarchy in this model (e.g., tight bounds for the model with two links, i.e., two servers). These results were later extended in [5, 15, 18], and, in particular, tight bounds in this model were established by Czumaj and Vöcking in [5]. Even though Koutsoupias and Papadimitriou [16] explicitly mention the interest in and the need of investigating more general cost functions, in particular, cost functions corresponding to the performance in queueing systems, no prior works for nonlinear cost functions are known in this model. For an up-to-date survey on the price of anarchy in the Koutsoupias–Papadimitriou (parallel links) model and related models, see Vöcking [32].

Roughgarden and Tardos [29] (see also, e.g., [2, 8, 31]) considered a related general network model, where the streams may be required to be routed in a network from a source to a destination, and the cost of allocating the stream to any edge in the routing path is taken into account. They focus mainly on the fractional flow model, making the assumption that each stream consists of infinitely many units, each of which behaves in a selfish way (see the fractional model described in section 3.2). The network parameter (to be minimized) is the *average latency* (or, equivalently, the total latency) experienced by the system. They showed that when the latency functions of the edges are linear, then the average latency in a Nash equilibrium is at most $\frac{4}{3}$ times the average latency in an optimal routing. For arbitrary nondecreasing and continuous latency functions, they show the existence of Nash equilibria whose average latency may be arbitrarily greater than the average latency in an optimal routing. Roughgarden [27] extends this study and gives some more precise bounds for various classes of latency cost functions, including some queueing systems cost functions. Roughgarden and Tardos [29] also give a bicriteria result that the average latency in Nash equilibrium is at most the average latency in an optimal routing of double the amount of flow. A similar model has also been investigated by Friedman [10], who studied how the amount of flow in the system influences the price of anarchy under the average-cost objective. Friedman also studies the queueing model M/M/1 in this context. Finally, Roughgarden and Tardos [30] show that some of the results from [29] do not require the combinatorial structure of networks and can be extended to nonatomic congestion games.

Roughgarden and Tardos [29] also consider integral allocations. They give an example of a network in which a bicriteria price of anarchy is unbounded. (This is in contrast to the fractional flow, where the identically defined price of anarchy is upper bounded by 2.) In addition, they prove a sufficient condition under which the bicriteria price of anarchy is bounded. The sufficient condition is that the streams are so small that adding any stream to any server increases the average latency by at most a factor of α for any fixed $\alpha > 0$. They remark that this condition is restricted to pure allocations, and its application requires families of cost functions with $f_b(0) > \gamma$ for any $b > 0$ and fixed $\gamma > 0$. Bounds on the price of anarchy in general atomic (integral allocation) network and congestion games were proved by Awerbuch, Azar, and Epstein [1] and by Christodoulou and Koutsoupias [3]. For further references on the price of anarchy in routing games, see the recent survey by Roughgarden [28].

5. Results for monotone cost functions.

5.1. Fractional allocation.

Our first theorem shows that all monotone cost functions behave nicely under fractional allocation.

THEOREM 5.1. *For every server farm whose servers are described by monotone cost functions, the price of anarchy R^* in the fractional flow model is* 1.

This theorem follows almost directly from the definition of Nash equilibria specifying that all servers with positive flow at Nash equilibrium must have the same cost, which under monotone cost functions implies that all Nash equilibria have the same social cost $C = opt$ (see, e.g., [29, Lemma 2.5]). As a consequence of this theorem, let us observe that it immediately separates (in the fractional flow model) the two different notions of the price of anarchy investigated by Koutsoupias and Papadimitriou [16] (the price of anarchy R^*) and by Roughgarden and Tardos [29] (the average-cost price of anarchy Σ^*). Both notions of the price of anarchy compare the cost of a Nash equilibrium with the optimal cost but use different definitions of cost. In sharp contrast to Theorem 5.1, Roughgarden and Tardos [29] prove that there exist instances of fractional flow for monotone families of cost functions in which the average-cost price of anarchy is unbounded. This shows that minimum average latency and minimum maximum costs can differ arbitrarily under general, monotone cost functions. In the context we consider here, that of server farms, the maximum cost seems to be the natural choice as it guarantees fairness and efficiency for every customer (agent) simultaneously.

5.2. Integral allocation.

Now let us turn to the integral allocation model. We say a price of anarchy R over a family of cost functions $\mathcal{F}_B$ is *bounded* if, for every m and every server farm with m servers with cost functions from $\mathcal{F}_B$, there exists $\Gamma > 0$ such that for every set of streams the value of the worst-case Nash equilibrium is at most $\Gamma \cdot opt$. (Observe that Γ might depend on m.) Otherwise, the price of anarchy is *unbounded*. Our first result is a complete characterization of monotone families of class functions for which the price of anarchy is bounded versus unbounded.

THEOREM 5.2. *The price of anarchy R over a monotone family $\mathcal{F}_B$ of cost functions is bounded if and only if*

$$\exists \alpha \geq 1, \quad \forall b \in B, \quad \forall \lambda > 0, \quad f_b(2\,\lambda) \leq \alpha \cdot f_b(\lambda).$$

The proof of this theorem is presented in section 5.4. Notice that this characterization of bounded versus unbounded price of anarchy can be applied also to server farms with identical servers. (Recall that such farms are described by families of cost functions consisting only of a single cost function.)

Observe that for every family of monotone cost functions we can identify a minimum $\alpha \in \mathcal{R}_{\geq 1} \cup \{\infty\}$ that fulfills the conditions specified in the theorem. A natural question is "How does the price of anarchy depend on α?" In fact, our analysis (see the proof of Theorem 5.2 and Lemma 5.5) shows that the price of anarchy is at most $m^{O(\log \alpha)}$. Furthermore, observe that α is constant if the family of cost functions is assumed to be fixed. Thus, we can conclude that for every fixed family of cost functions the price of anarchy is either unbounded or polynomially bounded in the number of servers, m.

Let us illustrate the power of the above theorem by investigating some examples. First, we consider families over *polynomial cost functions*, i.e., functions of the form $\sum_{r=0}^{k} a_r \cdot \lambda^r$ for a fixed $k \geq 0$. For these families we can pick $\alpha = 2^k$ to conclude that here the price of anarchy is bounded. In contrast, there is no such α for *exponential cost functions*, i.e., cost functions for which an additive increase in the load leads to a multiplicative increase in the cost. The same is true for *linear threshold functions* of the form $\max\{0, \lambda - 1\}$.

COROLLARY 5.3. *The price of anarchy R for server farms with polynomial cost functions of bounded degree is bounded, whereas the price of anarchy for server farms of (identical) servers with exponential cost functions or linear threshold cost functions is unbounded.*

In the next sections we shall discuss several other, practically motivated examples of families of cost functions with unbounded price of anarchy based on well-known formulas from queueing theory that describe, for example, the expected delay or the expected fraction of rejections.

5.3. Integral allocations with negligible weights. It is known that the fractional flow model and the integral allocations model are very closely related, and in some cases an almost identical behavior of these two models can be observed (see, e.g., Remark 2.3 in [29]). In this section, we study the relationship between these two models. Recall that R_ϵ is the price of anarchy for Nash equilibria over ϵ-small streams. Then we can show the following result, whose proof can be found in section 5.4.

THEOREM 5.4. *Let $\mathcal{F}_B$ be any monotone family of cost functions. For every $\epsilon > 0$, the price of anarchy R_ϵ over ϵ-small streams is bounded if and only if the price of anarchy R is bounded.*

Interestingly, this theorem *separates fractional flow from integral allocations with negligible weights*. On one hand, there are several practical examples for families of cost functions (e.g., M/M/1 waiting time or Erlang loss formulas) for which our results imply that for arbitrarily small ϵ the price of anarchy R_ϵ (and hence R) is unbounded. On the other hand, Theorem 5.1 implies that R^*, the price of anarchy in the fractional model, is 1. Moreover, the instances proving the characterization of unbounded price of anarchy use only pure strategies. Thus, even *pure allocations with* *negligible weights are different from fractional flow.*

Finally, we point out that unbounded price of anarchy is not only a special phenomenon of cost functions having a pole or an unbounded first derivative. Later in the paper, we will see a practical example of a family of cost functions (based on the Erlang loss formula) that have an unbounded price of anarchy, although these functions as well as their first derivatives are upper bounded by one.

5.4. Proof of Theorems 5.2 and 5.4. We prove Theorems 5.2 and 5.4 in a single proof using two lemmas. The first lemma proves the sufficient conditions for a bounded price of anarchy, while the second lemma proves the necessary conditions.

LEMMA 5.5. *Suppose we are given a server farm with m servers having cost functions from a fixed, monotone family $\mathcal{F}_B$ satisfying*

$$\exists \alpha \geq 1, \quad \forall b \in B, \quad \forall \lambda > 0, \quad f_b(2\,\lambda) \;\leq\; \alpha f_b(\lambda).$$

Then, for every set of streams the worst-case cost over all Nash equilibria is upper bounded by $opt \cdot m^{O(1)}$.

Proof. Fix an arbitrary allocation in Nash equilibrium. Let C denote the cost of this allocation. We will show that $C \leq m \cdot \alpha^{\lceil \log m \rceil} \cdot opt = m^{O(1)} \cdot opt$.

We first observe that

$$f_b(s\,\lambda) \leq \alpha^{\lceil \log s \rceil} \cdot f_b(\lambda)$$

for every $s \geq 1$. Let $X = \sum_{i \in [n]} \lambda_i$ denote the total injected load. Let x denote the load on server 1 (which has the greatest bandwidth) under an *optimal fractional allocation*, i.e., an allocation with minimum maximum cost over all servers assuming that streams can be split arbitrarily. Without loss of generality, we assume that server 1 has the maximum load over all servers, and hence $X \leq m\,x$. (Recall that since server 1 is the server with the greatest bandwidth and each function $f_b \in \mathcal{F}_B$ is nondecreasing in λ and nonincreasing in b, there exists an optimal fractional allocation with maximum load for server 1.) In this way,

$$f_{b_1}(X) \;\leq\; f_{b_1}(m\,x) \;\leq\; \alpha^{\lceil \log m \rceil} \cdot f_{b_1}(x) \;\leq\; \alpha^{\lceil \log m \rceil} \cdot opt.$$

Let M denote the set of servers $j \in [m]$ with $\sum_{i \in [n]} p_i^j > 0$. Pick any server j in M. Let $i \in [n]$ denote a stream with $p_i^j > 0$. Then, the Nash equilibrium property guarantees that $c_i^j \leq c_i^1$. Hence, for every $j \in M$,

$$\mathbf{E}\left[C_j\right] \;\leq\; \mathbf{E}\left[C_j | x_i^j = 1\right] \;=\; c_i^j \;\leq\; c_i^1 \;\leq\; f_{b_1}(X) \;\leq\; \alpha^{\lceil \log m \rceil} \cdot opt.$$

Furthermore, for $j \in [m] \setminus M$, we have $\mathbf{E}\left[C_j\right] = f_{b_j}(0) \leq opt \leq \alpha^{\lceil \log m \rceil} \cdot opt$. As a consequence,

$$C \;=\; \mathbf{E}\left[\max_{j \in [m]} C_j\right] \;\leq\; m \cdot \max_{j \in [m]} \mathbf{E}\left[C_j\right] \;\leq\; m \cdot \alpha^{\lceil \log m \rceil} \cdot opt \;=\; m^{O(1)} \cdot opt,$$

which completes the proof of Lemma 5.5. □

Now, we prove a sufficient condition for an unbounded price of anarchy. Observe that the negation of the property gives the sufficient condition for a bounded price of anarchy as specified in Theorem 5.2.

LEMMA 5.6. *Let $\epsilon > 0$ be chosen arbitrarily. Suppose we are given a server farm with only two identical servers, each with the same monotone cost function f satisfying*

$$\forall \alpha \geq 1, \quad \exists \lambda > 0, \quad f(2\,\lambda) > \alpha \cdot f(\lambda).$$

Then, for every $\Gamma \geq 1$, there exists a pure Nash equilibrium over ϵ-small streams with cost $C > \Gamma \cdot opt$.

Proof. First, let us show that the above property of the function f implies that

$$\forall \alpha \geq 1, \quad \beta > 0, \quad \exists \lambda > 0, \quad f((1+\beta)\cdot\lambda) > \alpha \cdot f(\lambda). \tag{5.1}$$

Indeed, if we consider the negations of the two statements, then it suffices to show that $\exists \alpha \geq 1, \beta > 0,\ \forall \lambda > 0\ f((1+\beta)\cdot\lambda) \leq \alpha \cdot f(\lambda)$ implies $\exists \alpha' \geq 1,\ \forall \lambda > 0,\ f(2\,\lambda) \leq \alpha' \cdot f(\lambda)$. Assume that indeed $\exists \alpha \geq 1, \beta > 0,\ \forall \lambda > 0\ f((1+\beta)\cdot\lambda) \leq \alpha \cdot f(\lambda)$ holds. Now, if $\beta \geq 1$, $f(2\,\lambda) \leq \alpha' \cdot f(\lambda)$ follows by monotonicity if we set $\alpha' = \alpha$. If $\beta < 1$, then $f(2\,\lambda) \leq \alpha' \cdot f(\lambda)$ follows by setting $\alpha' = \alpha^{\lceil 1/\log(1+\beta)\rceil}$.

Now, consider a server farm with two identical servers, each with the same monotone cost function f satisfying condition (5.1). For the purpose of contradiction, assume there exists Γ such that $C \leq \Gamma \cdot opt$ for every Nash equilibrium over ϵ-small streams. Therefore, by (5.1), there exists $\lambda > 0$ such that $f((1+\epsilon/10)\,\lambda) > \Gamma \cdot f(\lambda)$. Using this assumption, we will now define a Nash equilibrium over ϵ-small streams with cost $C > \Gamma \cdot opt$.

First, let us consider streams of identical weight $w \in [\lambda\epsilon/2, \lambda\epsilon]$ and assign them to the servers so that the cost on each server is exactly $f(\lambda)$. Let τ be the number of streams per server in this allocation. Now, let us slightly change the instance by taking two streams, one from each server, and "breaking" each of them into two smaller streams, one of weight $\frac{3}{5}\,w$ and the other of weight $\frac{2}{5}\,w$. It is easy to see that the optimal allocation for this instance has cost $opt = f(\lambda)$.

Let us consider a different allocation of the streams to the servers. We assign $\tau - 1$ streams of weight w and two streams of weight $\frac{3}{5}\,w$ to the first server and the remaining streams to the second server. In this way, the first server has cost $f(\lambda + \frac{1}{5}\,w)$, whereas the second server has cost $f(\lambda - \frac{1}{5}\,w)$. This allocation defines a Nash equilibrium, because the streams have minimum weight $\frac{2}{5}\,w$, and therefore there is no incentive for any of them to change its strategy. The cost of this Nash equilibrium is

$$C \;=\; f\left(\lambda + \frac{w}{5}\right) \;\geq\; f\left(\lambda + \frac{\lambda\cdot\epsilon}{10}\right) \;>\; \Gamma \cdot f(\lambda) \;=\; \Gamma \cdot opt.$$

Clearly this contradicts our initial assumption that $C \leq \Gamma \cdot opt$ for any Nash equilibrium over ϵ-small streams. This completes the proof of Lemma 5.6. ▯

Now, Theorems 5.2 and 5.4 follow immediately from Lemmas 5.5 and 5.6.

5.5. Bicriteria price of anarchy. It is not surprising that selfish routing can lead to a dramatic cost increase when the cost function has a pole. In principle, bicriteria measures can be much more informative as they filter out the extreme behavior of cost functions and look directly at the quality of the load balancing obtained by selfish routing. The following theorem, however, shows that an unbounded price of anarchy R implies a very poor worst-case behavior under bicriteria measures as well.

THEOREM 5.7. *Consider a server farm with m servers, with cost functions drawn from a monotone family $\mathcal{F}_B$. If the price of anarchy R is unbounded, then the bicriteria price of anarchy $\overline{R}$ has value at least m, even over ϵ-small streams.*

Proof. Let n denote, as usual, the number of streams, and let $\alpha = m^{n-1}$. Assume that n is such that $\frac{m}{n} \le \epsilon$. Since R is unbounded, Theorem 5.2 implies that there exist $f_b \in \mathcal{F}_B$ and $\lambda > 0$, such that $f_b(2\lambda) > \alpha f_b(\lambda)$.

Assume that we have m identical servers, each with bandwidth $b_j = b$ ($j \in [m]$), and a set of n identical data streams, each having weight $\lambda_i = \frac{\lambda m}{\Gamma n}$ ($i \in [n]$), where $\Gamma > 0$ is chosen such that $\frac{\lambda m}{b} \le \Gamma$. Define the probabilities p^i_j as $p^i_j = \frac{1}{m}$ for each $i, j \in [m]$. These probabilities define a Nash equilibrium, since all the expected costs c^i_j have the same value.

Let us fix a server j. The probability that all the data streams are assigned to server j is $\Pi_i p^i_j = m^{-n}$. In this case, the cost on server j is $f_b\left(\frac{\lambda m}{\Gamma}\right)$, since we have n streams, each of weight $\frac{\lambda m}{\Gamma n}$, and so their total weight is $\frac{\lambda m}{\Gamma n} n = \frac{\lambda m}{\Gamma}$.

Therefore, with probability m^{-n}, the cost on a particular server j is at least $f_b\left(\frac{\lambda m}{\Gamma}\right)$, and thus also $\max_{j\in[m]} C_j \ge f_b\left(\frac{\lambda m}{\Gamma}\right)$. Additionally, these events corresponding to different servers are pairwise disjoint. Using these observations, we can estimate $C = \mathbf{E}\left[\max_j C_j\right]$ as

$$\mathbf{E}\left[\max_j C_j\right] \ge m \cdot f_b\left(\frac{\lambda m}{\Gamma}\right) \cdot m^{-n} \;=\; \frac{f_b\left(\frac{\lambda m}{\Gamma}\right)}{m^{n-1}}.$$

We want to show that if $C \le opt_\Gamma$, then $\Gamma \ge m$. We consider two cases, $\lambda \ge b$ and $\lambda < b$. In the first case we have $\lambda \ge b$, which, by our assumption $\frac{\lambda m}{b} \le \Gamma$, implies that $\Gamma \ge m$, and so we are done in this case.

Suppose now that $\lambda < b$, and assume toward a contradiction that there is Γ such that $\frac{\lambda m}{b} \le \Gamma$, $C \le opt_\Gamma$, and $\Gamma < m$. Then, we obtain

$$\frac{f_b\left(\frac{\lambda m}{\Gamma}\right)}{m^{n-1}} \;\le\; opt_\Gamma \;\le\; f_b\left(\frac{\lambda m}{\Gamma n} \cdot \frac{n}{m} \cdot \Gamma\right) \;=\; f_b\left(\lambda\right),$$

where the last inequality follows by observing that the value of opt_Γ is at most the value of a solution in which we assign $\frac{n}{m}$ (we can assume that $\frac{n}{m}$ is a positive integer) data streams to each server, after blowing up each data stream by Γ. Then, we obtain

$$f_b\left(\frac{\lambda m}{\Gamma}\right) \;\le\; m^{n-1} \cdot f_b\left(\lambda\right). \tag{5.2}$$

By our assumption, $\forall \alpha \ge 1,\ \exists \lambda > 0,\ f_b(2\lambda) > \alpha f_b(\lambda)$. This, by the arguments from the proof of Lemma 5.6, implies that $\forall \alpha \ge 1, \beta > 0,\ \ \exists \lambda > 0,\ \ f_b((1+\beta)\lambda) > \alpha f_b(\lambda)$. Now, we plug $\alpha = m^{n-1}$, $1 + \beta = \frac{m}{\Gamma}$ to obtain a contradiction with inequality (5.2).

Since all the servers are identical, a data stream of weight λ_i is ϵ-small if $\lambda_i \le \epsilon \frac{\sum_j \lambda_j}{m}$. Since in our case $\lambda_i = \frac{\lambda m}{\Gamma n}$ for all $i \in [n]$, the last condition is equivalent to $\frac{\lambda m}{\Gamma n} \le \epsilon \frac{\lambda}{\Gamma}$, which gives $\frac{m}{n} \le \epsilon$. The last inequality is true by the choice of n, and so our data streams are ϵ-small. Also, observe, that we have chosen Γ such that $\sum_i \lambda_i = \frac{\lambda m}{\Gamma} \le b$; i.e., the constructed instance is extremely lightly loaded. □

The example proving the bad price of anarchy from Theorem 5.7 is a server farm of identical servers. For this case one can easily show that a bicriteria price of anarchy of m is the worst possible, as a Nash equilibrium cannot be worse than mapping all streams to the same server, and this, of course, is not worse than an optimal allocation with all rates blown up by a factor of m.

Even worse, our proof of this negative result gives a construction of a Nash equilibrium in which also all streams are identical and the total injected load $\sum_{i\in[n]} \lambda_i$ is less than the bandwidth of a single server. Thus, the bicriteria price of anarchy can be very poor, even in extremely lightly loaded cases.

5.6. Average-cost objective functions. There has been a lot of prior research that (unlike in our analysis above considers an objective function) aims at minimizing average cost (and hence also the total cost); see, e.g., [29]. In this section we investigate the relationship between different objective functions under integral allocation. Recall that the prices of anarchy for the average-cost objective function are denoted by Σ, Σ^*, Σ_ϵ, and $\overline{\overline{\Sigma}}$.

THEOREM 5.8. *For any monotone family of cost functions, Σ is bounded if and only if R is bounded. For every $\epsilon > 0$, Σ_ϵ is bounded if and only if R is bounded as well.*

Furthermore, for a server farm with m servers with cost functions from a monotone family of cost functions, if the price of anarchy Σ is unbounded, then $\overline{\overline{\Sigma}} \geq m$, and this holds even over ϵ-small streams.

Proof. Fix a monotone family $\mathcal{F}_B = \{f_b | b \in B\}$ of cost functions. First let us show that Σ for this family is bounded if the price of anarchy R is bounded. Clearly, if R is bounded, then we obtain from Theorem 5.2

$$\exists \alpha \geq 1, \quad \forall b \in B, \quad \forall \lambda > 0, \quad f_b(2 \cdot \lambda) \leq \alpha \cdot f_b(\lambda).$$

We will show that Σ is bounded provided this property is given. Fix any Nash equilibrium with probabilities p_i^j, $i \in [n]$, $j \in [m]$. We have to give an upper bound on the ratio between the expected average latency given by the probabilities p_i^j, on one hand, and the optimal average latency, on the other hand. Let $w = w_1, \dots, w_m$ denote a vector of random variables with w_j describing the injected load of server j. Let w^* denote a corresponding load vector of an optimal allocation that minimizes the average latency. We have to show that there exists $\Gamma \geq 1$ such that

$$\mathbf{E}\left[\sum_{j \in [m]} w_j \cdot f_{b_j}(w_j)\right] \;\leq\; \Gamma \cdot \sum_{j \in [m]} w_j^* \cdot f_{b_j}(w_j^*).$$

Define $X = \sum_{j \in [m]} w_j = \sum_{j \in [m]} w_j^*$. Then $f_{b_1}(X)$ corresponds to the cost that is obtained by assigning all of the load to the fastest server. Using the same arguments as in the proof of Lemma 5.5, we obtain $\mathbf{E}\left[f_{b_j}(w_j)\right] \leq f_{b_1}(X)$ for every $j \in [m]$, and thus

$$\begin{aligned}\mathbf{E}\left[\sum_{j \in [m]} w_j \cdot f_{b_j}(w_j)\right] &\leq \sum_{j \in [m]} X \cdot f_{b_1}(X) \;\leq\; m^2 \cdot \alpha^{\lceil 1+\log m \rceil} \cdot \frac{X}{m} \cdot f_{b_1}\!\left(\frac{X}{m}\right) \\ &\leq m^2 \cdot \alpha^{\lceil 1+\log m \rceil} \cdot \sum_{j \in [m]} w_j^* \cdot f_{b_j}(w_j^*).\end{aligned}$$

The second inequality follows from our assumptions on the family of cost functions. The third inequality needs some more explanation. Observe that there exists $j \in [m]$ with $w_j^* \geq X/m$. Applying monotonicity of our cost function family, we obtain $\frac{X}{m} f_{b_1}(\frac{X}{m}) \leq w_j^* f_{b_j}(w_j^*)$, which yields the inequality. As a consequence, $\Sigma \leq m^2 \, \alpha^{\lceil 1+\log m \rceil}$; that is, Σ is bounded.

It remains to show that an unbounded price of anarchy R implies unbounded Σ. If R is unbounded, then the family of cost functions satisfies

$$\forall \alpha \geq 1, \quad \exists b \in B, \quad \exists \lambda > 0, \quad f_b(2\lambda) > \alpha f_b(\lambda).$$

In Lemma 5.6 we describe an instance with two identical servers and a set of ϵ-small streams such that both servers have identical loads in an optimal allocation but there is a Nash equilibrium in which one of the servers receives more load than the other server. Using the above property we show that if one server has load at least $(1+\epsilon/10)$ times the average load, then the cost on this server can deviate by an arbitrarily large factor from the optimal cost, which then proves that R is unbounded. A straightforward adaptation of this argument shows also that Σ is unbounded if the above condition is fulfilled.

The proof of Theorem 5.7 in the average-cost case is basically the same as in the min-max case, since the values of the two objectives are the same in the lower bound instance. This completes the proof of Theorem 5.8. $\square$

We conclude that in the case of integral assignments, the average-cost objective leads to exactly the same characterizations for the price of anarchy and bicriteria price of anarchy as those given in the Theorems 5.2, 5.4, and 5.7 for the min-max objective.

We also point out that the property proven in Theorem 5.8 is nontrivial. In general, the behavior of the two objective functions can be quite different. For example, for the fractional case, Roughgarden and Tardos [29] give examples with unbounded price of anarchy Σ^*, whereas Theorem 5.1 shows that $R^* = 1$.

6. Results for queueing theoretical cost functions. Koutsoupias and Papadimitriou [16] asked as an open problem to investigate the price of anarchy of realistic cost functions of the form $\frac{1}{b-\min\{\lambda,b\}}$. This particular cost function describes the expected waiting time (delay) under an M/M/1 queueing process with injection rate λ and service rate b. Our characterization of bounded and unbounded price of anarchy given in Theorem 5.2 immediately implies that the integral price of anarchy for this family of functions is unbounded. This resolves an open problem from [16]. In the following, we consider the behavior of cost functions motivated by queueing theory in more detail. In particular, in section 6.3.3 we give a simple greedy exact algorithm for solving a related fractional scheduling problem with Pollaczek–Khinchin (P-K) cost functions under heterogeneous traffic.

6.1. Queueing systems without rejection. Several other studies are concerned with similar cost functions motivated by queueing systems; see, e.g., [13, 14, 17, 20]. In order to avoid discussions about what exactly is the right queueing model (e.g., M/M/1, M/D/1, M/G/1, M/G/b, ...) and what is the right cost model (e.g., expected waiting time or expected system time), let us introduce a unifying concept of "monotone queueing functions." A monotone cost function f_b is called a *monotone queueing function* if it satisfies $\lim_{\lambda\to b^-} f_b(\lambda) = \infty$. (Having the monotonicity, we may therefore think that the value of $f_b(\lambda)$ is unbounded and thus undefined for every $\lambda \geq b$.)

This assumption is motivated by the fact that expected waiting as well as expected system time in every queueing process without rejection goes to infinity when the injection rate approaches the service rate (or bandwidth) of the server. An immediate consequence of the pole of these functions is that parameter α from Theorem 5.2 is ∞. Thus, we have the following corollary to Theorems 5.1, 5.2, and 5.7.

COROLLARY 6.1. *For every family $\mathcal{F}_{\mathbb{R}_{>0}}$ of monotone queueing functions, $R^* = 1$, $R = \infty$, and $\overline{R} \geq m$, even under the restriction that all streams are ϵ-small.*

The proof for this negative result in Theorem 5.7 uses a Nash equilibrium in which all streams are identical and the total injected load $\sum_{i\in[n]} \lambda_i$ is less than the bandwidth of a single server. Thus, selfish routing can lead to a catastrophic performance degradation, even under bicriteria measures in extremely lightly loaded cases.

Recall that the instances proving the unbounded price of anarchy R are constructed from pure strategies only. However, the bad instances for the bicriteria price of anarchy $\overline{R}$ that we have seen until now use mixed strategies. This motivates us to investigate whether the randomness introduced due to the choice of mixed strategies is the only source of bad bicriteria price of anarchy. The following theorem demonstrates that bicriteria price of anarchy can also be poor when we restrict ourselves to pure strategies only.

THEOREM 6.2. *Let $\mathcal{F}_{\mathbb{R}_{>0}}$ be any family of monotone queueing functions. Suppose m is the number of servers and there exists $b > \sqrt{m}$ such that $f_b((1-\frac{b}{m})\cdot b) < f_1(\frac{b}{m})$, where $f_1, f_b \in \mathcal{F}_{\mathbb{R}_{>0}}$. Then, the bicriteria price of anarchy over pure strategies is at least $\frac{m}{2\,b}$.*

Before we prove Theorem 6.2, we give some further explanation of this theorem. For example, the cost function for M/M/1 waiting time is $\frac{\lambda}{b(b-\min\{b,\lambda\})}$, and the cost function for M/M/1 system time is $\frac{1}{b-\min\{b,\lambda\}}$. If we assume the cost function for waiting time, then the theorem implies a bicriteria price of anarchy over pure strategies of $\Omega(m^{1/3})$. Similarly, for system time the bicriteria price of anarchy is $\Omega(m^{1/2})$. In both cases, the total injected load in the example that gives these bad results is very small.

We now return to the proof of Theorem 6.2. The following lemma immediately yields Theorem 6.2.

LEMMA 6.3. *Let $\mathcal{F}_{\mathbb{R}_{>0}}$ be a family of monotone queueing functions. Suppose we are given $m+2$ servers with bandwidths $1 = b_1 = b_2 = \cdots = b_m \le b_{m+1} = \frac{b^2}{m} \le b_{m+2} = b$, such that $f_{b_j} \in \mathcal{F}_{\mathbb{R}_{>0}}$ for each $j = 1, 2, \ldots, m+2$. Assume, moreover, that there exists an ε with $0 < \varepsilon < \min\{1/\Gamma, 1\}$, such that the functions f_1 and f_b fulfill*

$$f_b\left(\left(1-\frac{1}{2\Gamma}\right)b+\varepsilon\right) \;\le\; f_1\left(\frac{1}{2\Gamma}\right), \tag{6.1}$$

where $\Gamma = \frac{m}{2b}$ and $\frac{b}{2\Gamma} - \varepsilon/2 > 1$. Then, in this system, $C > opt_\Gamma$, where C is the maximum value over all Nash equilibria assuming only pure strategies. Thus, for the bicriteria price of anarchy $\overline{R}$ over pure strategies we have $\overline{R} \ge \frac{m}{2b}$.

Proof. Before we define an appropriate instance of the problem, we will need some technical observations which use the properties of monotone queueing functions. First, let us define the following useful notation:

$$cost(\varepsilon) \;=\; \max\left\{f_1\left(\frac{1}{2}\right), f_b\left(\frac{b}{2}-\frac{\varepsilon\Gamma}{2}\right)\right\}.$$

Observe that $cost(0)$ is a finite nonnegative number. Now, by the properties of monotone queueing functions, observe that $\lim_{\varepsilon'\to 0} f_{b/(2\Gamma)}\left(\frac{b}{2\Gamma}-\varepsilon'/2\right) = \infty$. This implies that there exists an $\varepsilon' > 0$, such that

$$f_{b/(2\Gamma)}\left(\frac{b}{2\Gamma}-\varepsilon'/2\right) > cost(0). \tag{6.2}$$

Observe, furthermore, that replacing ε by $\min\{\varepsilon, \varepsilon'\}$ still maintains assumption (6.1) and $\frac{b}{2\Gamma}-\varepsilon/2 > 1$. Thus, from now on we assume that ε is such that (6.1), $\frac{b}{2\Gamma}-\varepsilon/2 > 1$, and (6.2) (with $\varepsilon' = \varepsilon$) hold.

We define the following instance of the problem. Let the servers $1, 2, \ldots, m$ be called slow. We also have a fast server with $b_{m+2} = b$ and an additional server with bandwidth $b_{m+1} = \frac{b^2}{m} = \frac{b}{2\Gamma}$. Assume that each slow server holds one small data stream with weight $\frac{1}{2\Gamma}$, and let the fast server have one large data stream with weight $\frac{b}{2\Gamma} - \varepsilon/2$. The additional server does not hold any data stream.

In the solution we have just defined, each slow server has cost (after blowing the streams up by Γ) $f_1\left(\frac{1}{2}\right)$, and the fast server has cost $f_b\left(\frac{b}{2} - \frac{\varepsilon\Gamma}{2}\right)$. Recall that

$$cost(\varepsilon) \; = \; \max\left\{f_1\left(\frac{1}{2}\right), f_b\left(\frac{b}{2} - \frac{\varepsilon\Gamma}{2}\right)\right\},$$

and observe that by the definition of monotone queueing functions, the value of $cost(\varepsilon)$ is finite even if $\varepsilon \to 0$, and obviously $opt_\Gamma \leq cost(\varepsilon)$.

Now, we will define a Nash equilibrium for our instance. We put as many small data streams on the fast server as possible, to fill this server almost to its bandwidth b. More precisely, we first assign to the fast server a total amount of $\left(1 - \frac{1}{2\Gamma}\right)b + \varepsilon$ of small streams. Observe that we have enough small streams to achieve this. Suppose toward a contradiction that there is not enough small streams, i.e., their total size $\frac{m}{2\Gamma} < \left(1 - \frac{1}{2\Gamma}\right)b + \varepsilon$. Because $\frac{m}{2\Gamma} = b$, this last inequality is same as $b < \left(1 - \frac{1}{2\Gamma}\right)b + \varepsilon$; thus $\frac{b}{2\Gamma} < \varepsilon$. By our assumption $1 \leq b_{m+1} = \frac{b}{2\Gamma}$, we obtain that $1 < \varepsilon$, which is a contradiction to our assumption on ε. Second, if there are any remaining small streams, then we additionally assign to the fast server a maximum number ℓ of small streams such that

$$f_b\left(\left(1 - \frac{1}{2\Gamma}\right)b + \varepsilon + \frac{\ell}{2\Gamma}\right) \leq f_1\left(\frac{1}{2\Gamma}\right) \tag{6.3}$$

(if $\ell = 0$, then (6.3) simply reduces to (6.1)). If, after these assignments of small streams to the fast server, there still are remaining small streams, we assign them to slow servers—one small stream per one slow server (some of the small servers will thus remain empty).

The large stream cannot now be assigned to the fast server (it would exceed its bandwidth), and by $\frac{b}{2\Gamma} - \varepsilon/2 > 1$, the large stream also cannot be put on any slow server. Therefore, the large data stream must be assigned to the additional server $m + 1$.

We argue that we have defined a Nash equilibrium. First, none of the small streams would go from the fast server to a slow server, since by (6.3) we have $f_b\left(\left(1 - \frac{1}{2\Gamma}\right)b + \varepsilon + \frac{\ell}{2\Gamma}\right) \leq f_1\left(\frac{1}{2\Gamma}\right)$. Also, none of the small streams would go from the fast server to the additional server, since the remaining space on the additional server is $\varepsilon/2$, which, in particular, is smaller than the size of a small stream, $\varepsilon/2 < \frac{1}{2\Gamma}$. It is also easy to see that none of the small streams, if any, would want to change from a slow server to the fast or to the additional server.

Finally, the large stream cannot go from the additional server to any slow server (since $\frac{b}{2\Gamma} - \varepsilon/2 > 1$), or to the fast server (since it would exceed the capacity on the fast server).

The cost on the additional server in this Nash equilibrium is $f_{b/(2\Gamma)}\left(\frac{b}{2\Gamma} - \varepsilon/2\right)$, and hence, by the properties of monotone queueing functions, $\lim_{\varepsilon\to 0} f_{b/(2\Gamma)}\left(\frac{b}{2\Gamma} - \varepsilon/2\right) = \infty$. Recall that we have chosen $\varepsilon > 0$ such that (6.2) holds with $\varepsilon' = \varepsilon$. Thus, we obtain that C, which is the maximum value over all Nash equilibria, is larger than $cost(0)$. By monotonicity, we have that $cost(0) \geq cost(\varepsilon)$. Finally, we obtain that $opt_\Gamma \leq cost(\varepsilon) \leq cost(0) < C$. □

To summarize, we have shown that even for pure strategies and under a small total injection rate the slowdown due to the lack of coordination can be significant.

6.2. An alternative bicriteria measure.

For the family of monotone queueing functions there is another interesting bicriteria measure. It is a natural question to ask by how much one has to decrease the bandwidths of the servers such that an optimal allocation under the decreased bandwidths is at least as expensive as a Nash equilibrium for the original system. Let $\overline{R}_{\text{bw}}$ denote the corresponding worst-case bicriteria price of anarchy.

It turns out that for most functions from queueing theory the effect of changing the bandwidths is larger than the effect of changing the injection rate. In fact, most of these functions show *superlinear scaling*, i.e., $f_b(\lambda) \leq \frac{1}{\alpha} \cdot f_{b/\alpha}(\lambda/\alpha)$ for every $\lambda \in [0, b)$ and $\alpha \geq 1$. Applying this property, we can determine the bicriteria bandwidth price of anarchy very precisely.

THEOREM 6.4. *For every family $\mathcal{F}_B$ of monotone queueing functions with superlinear scaling, $\overline{R}_{\text{bw}} = m$, where m is the number of servers.*

Let OPT_Γ denote the value of the optimum solution in a system where bandwidths of all servers are slowed down by a factor of Γ. We first prove the following lemma.

LEMMA 6.5. *Fix an arbitrary monotone queueing function $f = f_b$. Consider a server farm with m identical servers with cost function f. Then for every $\epsilon > 0$ and $\Gamma < m$ there exists a Nash equilibrium over ϵ-small streams such that $C > OPT_\Gamma$.*

Proof. Assume that we have m identical servers, each with bandwidth $b_j = b$ ($j \in [m]$), and a set of n identical data streams, each having weight of $\lambda_i = \frac{b-\delta}{n}$ ($i \in [n]$), where n is such that $\frac{m}{n} \leq \epsilon$, and $\delta \in [0, b)$ will be specified later. Define the probabilities p_i^j by setting $p_i^j = \frac{1}{m}$ for each $i \in [n]$ and $j \in [m]$. These probabilities define a Nash equilibrium, since all the expected costs c_i^j have the same value.

Let us fix a server j. The probability that all the data streams are assigned to server j is $\Pi_i \, p_i^j = m^{-n}$. In this case, the cost on server j is $f_b(b - \delta)$, since we have n streams, each of weight $\frac{b-\delta}{n}$, and so their total weight is $\frac{b-\delta}{n} n = b - \delta$.

Therefore, with probability m^{-n}, the cost on a particular server j is at least $f_b(b - \delta)$, and thus also $\max_j C_j$ is at least $f_b(b - \delta)$ for $j \in [m]$. Additionally, these events corresponding to different servers are pairwise disjoint. Using these observations, we can estimate $C = \mathbf{E}[\max_j C_j]$ as

$$\mathbf{E}\left[\max_j C_j\right] \;\geq\; m \cdot f_b(b - \delta) \cdot m^{-n} \;=\; \frac{f_b(b - \delta)}{m^{n-1}}.$$

We want to show that if $C \leq OPT_\Gamma$, then $\Gamma \geq m$. Assume toward a contradiction that $C \leq OPT_\Gamma$ and $\Gamma < m$. Then, we obtain

$$\frac{f_b(b - \delta)}{m^{n-1}} \;\leq\; OPT_\Gamma \;\leq\; f_{b/\Gamma}\left(\frac{b - \delta}{n} \cdot \frac{n}{m}\right) \;=\; f_{b/\Gamma}\left(\frac{b - \delta}{m}\right),$$

where the last inequality follows by observing that the value of OPT_Γ is at most the value of a solution in which we assign $\frac{n}{m}$ (we can assume that $\frac{n}{m}$ is a positive integer) data streams to each server, after slowing the server down by Γ. Then, by the last inequality and by continuity of functions f_b and $f_{b/\Gamma}$, we have

$$\lim_{\delta \to 0} \frac{f_b(b - \delta)}{m^{n-1}} \;\leq\; \lim_{\delta \to 0} f_{b/\Gamma}\left(\frac{b - \delta}{m}\right) \;=\; f_{b/\Gamma}\left(\frac{b}{m}\right).$$

By our assumption the right-hand side is finite but the left-hand side is infinite by the properties of the function f_b. Therefore, there exists a small enough $\delta > 0$ such that $\frac{f_b(b-\delta)}{m^{n-1}} > f_{b/\Gamma}\left(\frac{b}{m}\right) \geq f_{b/\Gamma}\left(\frac{b-\delta}{m}\right)$ (by monotonicity), which is a contradiction. ∎

Proof of Theorem 6.4. Lemma 6.5 implies $\overline{R}_{\mathrm{bw}} \geq m$. Therefore, we have to show that $C \leq OPT_m$. Assume that b_j, $j \in [m]$, are the bandwidths of the servers. Let $\Lambda = \sum_i \lambda_i$, and observe that we can assume $\Lambda \leq \frac{1}{m}\sum_j b_j$. Otherwise, there is a server, say, j, in the OPT_m solution with load strictly greater than $\frac{b_j}{m}$, so $OPT_m = \infty$, and we are done. By this assumption, we see that $\sum_i \lambda_i \leq \max_j b_j$. Therefore, we can assign all the data streams (deterministically) to the fastest server, say, server 1, and let $B = b_1 = \max_j b_j$.

Using linearity of expectation, we have

$$C \;=\; \mathbf{E}\left[\max_j C_j\right] \;\leq\; \mathbf{E}\left[\sum_j C_j\right] \;=\; \sum_j \mathbf{E}\left[C_j\right] \;\leq\; m \cdot \max_j \mathbf{E}\left[C_j\right].$$

We now argue that $\max_j \mathbf{E}\left[C_j\right] \leq f_B(\Lambda)$. Assume toward a contradiction that there is a server, say, j_0, such that $\mathbf{E}\left[C_{j_0}\right] > f_B(\Lambda)$. Then there is a data stream, say, i_0, with $p_{i_0}^{j_0} > 0$. Also, $c_{i_0}^{j_0} = \mathbf{E}[C_{j_0}|x_{i_0}^{j_0} = 1] \geq \mathbf{E}\left[C_{j_0}\right]$, which follows from the fact that the random variable $(C_{j_0}|x_{i_0}^{j_0} = 1)$ cannot assume smaller values than the random variable C_{j_0}. Similarly, $f_B(\Lambda) \geq c_{i_0}^1 = \mathbf{E}\left[C_1|x_{i_0}^1 = 1\right]$, since the value $f_B(\Lambda)$ corresponds to the situation where all probabilities $p_i^1 = 1$ for all i and $p_i^j = 0$ for all i and $j \neq 1$ (recall that 1 is the fastest server). Thus, $p_{i_0}^{j_0} > 0$ and $c_{i_0}^{j_0} > c_{i_0}^1$, which is a contradiction with the Nash equilibrium property. Therefore, we have

$$C \;\leq\; m \cdot \max_j \mathbf{E}\left[C_j\right] \;\leq\; m \cdot f_B(\Lambda).$$

By the superlinear scaling we obtain

$$C \;\leq\; m \cdot f_B(\Lambda) \;\leq\; f_{B/m}\left(\frac{\Lambda}{m}\right).$$

Let us now consider $OPT_m^{\mathtt{frac}}$, which is the cost of an optimum solution for the fractional flow model in the system, where the bandwidths of all servers are reduced by a factor of m. We claim that $f_{B/m}\left(\frac{\Lambda}{m}\right) \leq OPT_m^{\mathtt{frac}}$. To show this, we first argue that in the optimal fractional solution, the fastest server has the largest load. Let us fix two servers with bandwidths b_1, b_2 such that $b_1 < b_2$, and let x_1, x_2 be the loads on these servers in the optimal fractional solution. It is easy to see that $f_{b_1}(x_1) = f_{b_2}(x_2)$, and by the superlinear scaling, we obtain $f_{b_2}(\frac{b_2}{b_1}x_1) \leq \frac{b_1}{b_2} f_{b_1}(x_1) < f_{b_2}(x_2)$, which by the monotonicity property yields $x_1 \leq x_2$.

Now, by a simple averaging argument, there is a server in the optimal fractional solution with load at least $\frac{\Lambda}{m}$. Using the fact we have just shown, we see that the fastest server (with bandwidth B) must have load at least $\frac{\Lambda}{m}$. This shows $f_{B/m}\left(\frac{\Lambda}{m}\right) \leq OPT_m^{\mathtt{frac}}$. We can finish the proof by observing the following inequality:

$$C \;\leq\; f_{B/m}\left(\frac{\Lambda}{m}\right) \;\leq\; OPT_m^{\mathtt{frac}} \;\leq\; OPT_m. \qquad \square$$

Let us emphasize that Theorem 6.4 gives tight results, among others, for expected waiting time or system time in the queueing systems M/M/1, M/D/1, or for expected waiting time in the general M/M/b system. This follows by the fact that one can show that the respective cost functions of these queueing systems obey the superlinear scaling.

The cost function for M/M/1 expected system time is $\frac{1}{b-\lambda}$, and the cost function for M/M/1 waiting time is $\frac{\lambda}{b(b-\lambda)}$; see, e.g., formulas (2.19) and (2.20) in [11] (note that μ in those formulas corresponds to our b). Considering the M/D/1 queue, the expected waiting time is given by $\frac{\lambda}{2b(b-\lambda)}$ (see (3.16) in [11] with $k \to \infty$), and the system time is $\frac{2b-\lambda}{2b(b-\lambda)}$; see [11]. The superlinear scaling property is straightforward to show for those four cost functions, and in fact it holds with equality. Thus, Theorem 6.4 implies the following result.

Corollary 6.6. *Suppose we are given a server farm with m servers with a cost function family being either the expected system time or the expected waiting time in either the M/M/1 or M/D/1 queueing system. Then, $\overline{R}_{\text{bw}} = m$.*

We provide a proof of the superlinear scaling property for the complicated case, i.e., the expected waiting time in the general M/M/b system. A server in this queueing system has $b \in \mathbb{N}$ parallel channels, each operating at a fixed speed 1; therefore, μ, as used in [11], is 1. The expected waiting time in the M/M/b system is given by formula (2.27) in [11]:

$$\frac{\lambda^b}{b! \cdot b \cdot (1-\lambda/b)^2} \cdot \left(\sum_{i=0}^{b-1} \frac{\lambda^i}{i!} + \frac{\lambda^b}{b! \cdot (1-\lambda/b)} \right)^{-1} . \tag{6.4}$$

We defer a somehow technical proof of Lemma 6.7 to section 6.2.1.

Lemma 6.7. *The cost function describing the expected waiting time in the M/M/b system possesses the superlinear scaling property.*

Thus, by applying Lemma 6.7 and Theorem 6.4, we obtain the following result.

Corollary 6.8. *Consider a server farm with m servers with a cost function being the expected waiting time in the M/M/b queueing system. Then, $\overline{R}_{\text{bw}} = m$.*

6.2.1. Superlinear scaling for the M/M/b queue.

This section is devoted to the proof of Lemma 6.7. Recall that the expected waiting time in the M/M/b queue is given by formula (6.4). Since b in (6.4) denotes the number of channels, it is sufficient to consider $\alpha \geq 1$ in the superlinear scaling property such that $b/\alpha \in \mathbb{N}$. Let $f_b(\lambda)$ be given by (6.4), i.e.,

$$f_b(\lambda) = \frac{\lambda^b}{b! \cdot b \cdot (1-\lambda/b)^2} \cdot \left(\sum_{i=0}^{b-1} \frac{\lambda^i}{i!} + \frac{\lambda^b}{b! \cdot (1-\lambda/b)} \right)^{-1} .$$

We start with the following result.

Lemma 6.9. $f_{b+1}(\lambda) \leq \frac{b}{b+1} \cdot f_b(\frac{b}{b+1} \cdot \lambda)$ *for any* $b \in \mathbb{N}_{\geq 1}$ *and* $\lambda \in [0, b+1)$.

Before we prove Lemma 6.9, we show how it implies the superlinear scaling property, that is, Lemma 6.7 (Lemma 6.10 below is the precise statement of Lemma 6.7).

Lemma 6.10. *Given any $b \in \mathbb{N}_{\geq 1}$, $f_b(\lambda)$ defined by formula* (6.4), *and $\alpha \geq 1$ such that $b/\alpha \in \mathbb{N}$, we have that $f_b(\lambda) \leq \frac{1}{\alpha} f_{b/\alpha}(\lambda/\alpha)$ for any $\lambda \in [0, b)$.*

Proof. Since we want that $b/\alpha \in \mathbb{N}$, α must be rational, i.e., there is $a \in \mathbb{N}_{\geq 1}$, $b \geq a$, such that $\alpha = b/a$. Then, by using Lemma 6.9 multiple times we obtain

$$\begin{aligned} f_b(\lambda) &\leq \frac{b-1}{b} \cdot f_{b-1}\left(\frac{b-1}{b} \cdot \lambda\right) \leq \frac{b-1}{b} \cdot \frac{b-2}{b-1} \cdot f_{b-2}\left(\frac{b-1}{b} \cdot \frac{b-2}{b-1} \cdot \lambda\right) \leq \cdots \\ &\leq \frac{b-1}{b} \cdot \frac{b-2}{b-1} \cdot \cdots \cdot \frac{a}{a+1} \cdot f_a\left(\frac{b-1}{b} \cdot \frac{b-2}{b-1} \cdot \cdots \cdot \frac{a}{a+1} \cdot \lambda\right) = \frac{a}{b} \cdot f_a\left(\frac{a}{b} \cdot \lambda\right). \end{aligned}$$

Thus, we have $f_b(\lambda) \leq \frac{a}{b} \cdot f_{\frac{a}{b} \cdot b}\left(\frac{a}{b} \cdot \lambda\right)$. □

Proof of Lemma 6.9. We first rewrite formula (6.4) to get the following form which we will use throughout the proof:

$$f_b(\lambda) = \frac{\lambda^b}{b! \cdot b \cdot (1-\lambda/b)^2} \left(\sum_{i=0}^{b} \frac{\lambda^i}{i!} + \frac{\lambda^{b+1}}{b! \cdot b \cdot (1-\lambda/b)} \right)^{-1}.$$

Observe now that we can equivalently replace $\lambda \in [0, b+1)$ with $\frac{b+1}{b}\lambda$ in the formula $f_{b+1}(\lambda) \leq \frac{b}{b+1} f_b(\frac{b}{b+1}\lambda)$ to obtain $\frac{b+1}{b} f_{b+1}(\frac{b+1}{b}\lambda) \leq f_b(\lambda)$, where now $\lambda \in [0, b)$. In what follows, we will prove this latter, equivalent, formula.

Since $f_b(0) = 0$, and $f_b(\lambda) > 0$ if $\lambda \in (0, b)$, when we assume $\lambda > 0$, the formula we want to prove can be rewritten as

$$\frac{b+1}{b} \cdot \frac{1}{f_b(\lambda)} \leq \frac{1}{f_{b+1}\left(\frac{b+1}{b}\lambda\right)},$$

which is equivalent to

$$\begin{aligned} &\frac{b+1}{b} \cdot \frac{b! \cdot b \cdot (1-\lambda/b)^2}{\lambda^b} \cdot \left(\sum_{i=0}^{b} \frac{\lambda^i}{i!} \right) + \frac{b+1}{b} \cdot \lambda \left(1 - \frac{\lambda}{b}\right) \\ &\leq \frac{(b+1)! \cdot (b+1) \cdot (1-\lambda/b)^2}{\left(\frac{b+1}{b} \cdot \lambda\right)^{b+1}} \cdot \left(\sum_{i=0}^{b+1} \frac{\left(\frac{b+1}{b} \cdot \lambda\right)^i}{i!} \right) + \frac{b+1}{b} \cdot \lambda \left(1 - \frac{\lambda}{b}\right), \end{aligned}$$

and thus we have to prove the following:

$$\frac{b+1}{b} \cdot \frac{b! \cdot b \cdot (1-\lambda/b)^2}{\lambda^b} \cdot \left(\sum_{i=0}^{b} \frac{\lambda^i}{i!} \right) \leq \frac{(b+1)! \cdot (b+1) \cdot (1-\lambda/b)^2}{\left(\frac{b+1}{b} \cdot \lambda\right)^{b+1}} \cdot \left(\sum_{i=0}^{b+1} \frac{\left(\frac{b+1}{b} \cdot \lambda\right)^i}{i!} \right).$$

Since $\lambda < b$, we can obtain the following equivalent inequality:

$$\sum_{i=0}^{b} \frac{\left(\frac{b+1}{b}\right)^b \cdot \lambda^{i+1}}{i!} \leq b \cdot \sum_{i=0}^{b+1} \frac{\left(\frac{b+1}{b}\right)^i \cdot \lambda^i}{i!}.$$

Define now $p_i = \frac{\left(\frac{b+1}{b}\right)^b \cdot \lambda^{i+1}}{i!}$ for $i = 0, 1, \ldots, b$ and $q_i = \frac{\left(\frac{b+1}{b}\right)^i \cdot \lambda^i}{i!}$ for $i = 0, 1, \ldots, b+1$. We have to prove that $\sum_{i=0}^{b} p_i \leq b \cdot \sum_{i=0}^{b+1} q_i$. We will prove a stronger claim, namely, the following.

Claim. $p_i \leq b \cdot q_{i+1}$ *for each* $i = 0, 1, \ldots, b$. This claim is equivalent to

$$\left(\frac{b+1}{b}\right)^b \leq \left(\frac{b+1}{b}\right)^{x+1} \cdot \frac{b}{x+1} \quad \text{for} \quad x = 0, 1, \ldots, b.$$

Let us consider a function $h(x) = (\frac{b+1}{b})^{x+1} \cdot \frac{b}{x+1}$ for $x \in [0, b]$. We will argue that function h cannot assume values below $(\frac{b+1}{b})^b$.

Observe first that $h(b) = h(b-1) = (\frac{b+1}{b})^b$. Suppose that $b = 1$; then $h(1) = h(0) = (\frac{b+1}{b})^b$, and so the claim holds. Assume that $b \geq 2$, and consider the first derivative of h,

$$h'(x) = \left(\frac{b+1}{b}\right)^{x+1} \cdot \frac{b}{x+1} \cdot \left(\ln\left(\frac{b+1}{b}\right) - \frac{1}{x+1} \right) = h(x) \cdot \left(\ln\left(\frac{b+1}{b}\right) - \frac{1}{x+1} \right).$$

We will argue that the first derivative is negative when $x \in [0, b-1]$. Since $h(x) > 0$ for any $x \geq 0$, it suffices to show that $\ln\left(\frac{b+1}{b}\right) - \frac{1}{x+1} < 0$ when $x \in [0, b-1]$. This last inequality will certainly be true if it is true when we replace x by x's largest value, i.e., $b-1$. Then the inequality becomes

$$\ln\left(\frac{b+1}{b}\right) - \frac{1}{b} < 0 \Longleftrightarrow \left(1 + \frac{1}{b}\right)^b < \mathrm{e},$$

where the last inequality is known to hold for any $b \in \mathbb{N}_{\geq 1}$. We conclude that $h'(x) < 0$ when $x \in [0, b-1]$, and thus function h is decreasing in interval $[0, b-1]$. Since $b \geq 2$ and $h(0) = b+1 > \mathrm{e} > (\frac{b+1}{b})^b$, function h has its minimum at $b-1$, and, as we know, $h(b-1) = h(b) = (\frac{b+1}{b})^b$, which concludes the proof of the claim and at the same time the proof of Lemma 6.9. □

6.3. Queueing systems under heterogeneous traffic. Until now we implicitly assumed homogeneous traffic, that is, scenarios in which all streams have the same (general) session length distribution. However, several practical studies show that Internet traffic is far from being homogeneous [4, 7, 22]. Following these studies, one has to take into account different session length distributions that are characteristic of the heterogeneous traffic.

Let us investigate the consequences of heterogeneous traffic using a particular instance of a cost function. The *Pollaczek–Khinchin* (P-K) formula (see, e.g., formula (5.113) in [12]) describes the expected waiting time in M/G/1 queues. Observe that this is a very general class of queueing systems that allows arbitrary service time distributions. Before we proceed with our analysis, we first transform the P-K formula into a family of cost functions describing the expected time on servers with bandwidth b using only two parameters that specify the injected mix of streams into the server, namely, the expectation λ and the variance V of the injected load.

6.3.1. Derivation of the P-K cost function family in aggregate form. An M/G/1 server is a sequential server with a single service channel and an FCFS queue for storing blocked requests as described in section 2.2. Requests for sessions arrive at the server as a memoryless (i.e., Poisson) stream. Let r denote the injection rate, i.e., the expected number of requests arriving per unit of time at the server. The length of the session is described by a general probability distribution $\mathcal{D}$. If T is the length of a session (which is a random variable chosen according to $\mathcal{D}$) and b denotes the bandwidth of the server, then the server needs T/b units of time to process the session. The P-K equation (see, e.g., formula (5.113) in [12]) states that the expected waiting time (i.e., the time in the queue) at such server is described by the formula

$$W = \frac{r \cdot \mathbf{E}\left[T^2\right]}{2 \cdot b \cdot (b - r \cdot \mathbf{E}\left[T\right])}. \tag{6.5}$$

In the following, we rewrite this formula using different notation that will enable us to aggregate streams in a linear fashion.

Let X be a random variable describing the *load* arriving at a server in a time unit, i.e., the sum of arriving requests weighted with the corresponding session lengths. Let us investigate how the expected load, $\mathbf{E}\left[X\right]$, and the variance of the load, $\mathbf{Var}\left[X\right]$, are related to the injection rate r and the session length distribution $\mathcal{D}$. Consider any time interval of unit length. Let R denote the number of streams arriving during this interval and $T_1, \dots, T_R$ the lengths of these sessions. Then

$$\begin{aligned}\mathbf{E}[X] = \mathbf{E}\left[\sum_{i=1}^{R} T_i\right] &= \sum_{k\geq 0} \mathbf{Pr}[R=k] \cdot \mathbf{E}\left[\sum_{i=1}^{R} T_i \,\middle|\, R=k\right] \\ &= \sum_{k\geq 0} \mathbf{Pr}[R=k]\cdot k \cdot \mathbf{E}[T] \;=\; r\mathbf{E}[T],\end{aligned}$$

and

$$\begin{aligned}\mathbf{Var}[X] = \mathbf{E}\left[X^2\right] - \mathbf{E}[X]^2 &= \mathbf{E}\left[\left(\sum_{i=1}^{R} T_i\right)^2\right] - r^2\mathbf{E}[T]^2 \\ &= \mathbf{E}\left[\sum_{i=1}^{R} \mathbf{E}\left[T_i^2\right] + 2\sum_{i=1}^{R}\sum_{j=1}^{i-1} T_i T_j\right] - r^2\mathbf{E}[T]^2 \\ &= r\mathbf{E}\left[T^2\right] + \mathbf{E}[R(R-1)]\cdot\mathbf{E}[T]^2 - r^2\mathbf{E}[T]^2 \\ &= r\mathbf{E}\left[T^2\right],\end{aligned}$$

where the last equation follows from $\mathbf{E}[R(R-1)] = \mathbf{E}\left[R^2\right] - \mathbf{E}[R] = \mathbf{Var}\left[R^2\right] + \mathbf{E}[R]^2 - \mathbf{E}[R] = r + r^2 - r = r^2$. Hence, writing the P-K formula (6.5) in terms of expectation and variance of load arriving per unit of time gives

$$W = \frac{\mathbf{Var}[X]}{2\cdot b\cdot(b-\mathbf{E}[X])}.$$

Next we describe how different streams arriving at a server can be aggregated to a single stream. For every stream $i \in [n]$, λ_i is the expected load and V_i is the variance of the load. Let $x_i \in [0,1]$ be another parameter describing the fraction of stream i that is directed to the considered server, where we assume that every single request is directed to the considered server with probability x_i. This way, the expected load of stream i arriving at the considered server is $x_i\lambda_i$, and the variance of the load is x_iV_i because our above bounds show that the expectation and the variance of the load are proportional to the injection rate. Furthermore, the stream obtained by aggregating all streams in this way has expected load $\mathbf{E}[X] = \sum_{i\in[n]} x_i\lambda_i$ and variance $\mathbf{Var}[X] = \sum_{i\in[n]} x_iV_i$, where the latter equation follows from the independence of the load variables of different streams. Thus, the expected waiting time on the server is

$$W = \frac{\sum_{i\in[n]} x_iV_i}{2\cdot b\cdot(b-\sum_{i\in[n]} x_i\lambda_i)}. \tag{6.6}$$

For simplicity of notation, we drop the factor $\frac{1}{2}$ and define the cost of the considered server under the allocation $x_1, \dots, x_n$ as $2\,W$.

6.3.2. Fractional price of anarchy for the P-K cost function family. We now study the heterogeneous traffic using the P-K formula describing the expected waiting time in M/G/1 queues. Using our analysis from section 6.3.1, if every stream i is characterized by two weights λ_i and V_i, then by (6.6), the P-K family of cost functions can be defined as follows:

$$f_b(x_1,\dots,x_n) = \frac{\sum_{i=1}^{n} V_i\cdot x_i}{b\cdot(b-\sum_{i=1}^{n}\lambda_i\cdot x_i)}.$$

The remarkable fact here is that both parameters, the expected load and the variance, can be aggregated independently in a simple linear fashion. That is, the expected load injected into the server is $\lambda = \sum_{i=1}^{n} \lambda_i \cdot x_i$, and the variance of this load is $V = \sum_{i=1}^{n} V_i \cdot x_i$.

Observe that if we assume $\lambda_i = V_i$, then we are back in the homogeneous model with identical session length distribution, and we obtain a monotone queueing function with only one parameter, λ. Consequently, $R = \infty$, $R_\epsilon = \infty$, and $\overline{R} \geq m$ for the P-K cost function family. In the fractional flow model, however, we obtain different results. Recall that $R^* = 1$ under homogeneous traffic.

THEOREM 6.11. *The fractional price of anarchy R^* for the P-K cost function family is unbounded. If the ratio between the largest and smallest server bandwidth is bounded by S, then the price of anarchy is $R^* = S$.*

We need some technical tools before we present the proof of Theorem 6.11.

6.3.3. The local exchange lemma. Consider the following problem. We are given m servers, each with two positive parameters p_j and q_j, $j \in [m]$. We are also given n data streams, each with two positive parameters v_i and e_i, $i \in [n]$. We study the following linear program (without objective function), which we call LP:

$$\begin{aligned} &\sum_{i=1}^{n} (v_i p_j + e_i q_j) x_{ij} \leq 1 \qquad && \forall j \in [m], \\ &\sum_{j=1}^{m} x_{ij} = 1 && \forall i \in [n], \\ &x_{ij} \geq 0 && \forall i, j. \end{aligned} \tag{6.7}$$

The value of variable x_{ij} represents a fraction of data stream i assigned to server j. Let us assume that the LP is feasible and *tight*; i.e., every feasible solution satisfies all the constraints (6.7) with equality. We analyze the structure of feasible solutions of this LP.

LEMMA 6.12. *Let $j, j' \in [m]$ such that $\frac{p_j}{q_j} > \frac{p_{j'}}{q_{j'}}$. In any feasible solution of LP, if $x_{ij} > 0$ and $x_{i'j'} > 0$, then $\frac{v_i}{e_i} \leq \frac{v_{i'}}{e_{i'}}$.*

Proof. Toward a contradiction, we assume that there is a feasible solution with $x_{ij} > 0$, $x_{i'j'} > 0$, and $\frac{v_i}{e_i} > \frac{v_{i'}}{e_{i'}}$. Let ϵ be sufficiently small such that it is possible to exchange pieces of streams as follows. Suppose we decrease x_{ij} by ϵ and increase $x_{ij'}$ by the same amount. Next we increase $x_{i'j}$ by some amount ϵ' such that constraint (6.7) for j becomes tight again. Besides, we decrease $x_{i'j'}$ by ϵ'. We claim that this transformation yields a feasible solution, but this solution is not tight with respect to constraint (6.7) for server j', contradicting our assumption that the LP is tight, which implies $\frac{v_i}{e_i} \leq \frac{v_{i'}}{e_{i'}}$.

We need to show that constraint (6.7) for j' is not tight after the transformation. Since constraint (6.7) for j is tight after the transformation, it holds that

$$\epsilon(v_i p_j + e_i q_j) = \epsilon'(v_{i'} p_j + e_{i'} q_j),$$

which gives

$$\epsilon' = \epsilon \frac{v_i p_j + e_i q_j}{v_{i'} p_j + e_{i'} q_j}.$$

The change in the assignment of i increases the left-hand side of constraint (6.7) by

$$\epsilon \cdot (v_i p_{j'} + e_i q_{j'}),$$

while the change in the assignment of i' decreases this expression by

$$\epsilon' \cdot (v_{i'} p_{j'} + e_{i'} q_{j'}) = \epsilon \cdot \frac{(v_{i'} p_{j'} + e_{i'} q_{j'})(v_i p_j + e_i q_j)}{v_{i'} p_j + e_{i'} q_j}.$$

Hence, we need to show

$$(v_{i'} p_{j'} + e_{i'} q_{j'})(v_i p_j + e_i q_j) > (v_i p_{j'} + e_i q_{j'})(v_{i'} p_j + e_{i'} q_j),$$

which rewrites as

$$e_i v_{i'} p_{j'} q_j + e_{i'} v_i p_j q_{j'} > e_i v_{i'} p_j q_{j'} + e_{i'} v_i p_{j'} q_j.$$

The last inequality can be written as $(p_j q_{j'} - p_{j'} q_j)(e_{i'} v_i - e_i v_{i'}) > 0$, which by our assumptions that $\frac{v_i}{e_i} > \frac{v_{i'}}{e_{i'}}$ and $\frac{p_j}{q_j} > \frac{p_{j'}}{q_{j'}}$ is obviously true. $\square$

Observe that the structure of the feasible solution allows us to compute such a solution by a greedy algorithm: Sort the servers such that $\frac{p_1}{q_1} \geq \frac{p_2}{q_2} \geq \cdots \geq \frac{p_m}{q_m}$. Sort the data streams such that $\frac{v_1}{e_1} \leq \frac{v_2}{e_2} \leq \cdots \leq \frac{v_n}{e_n}$. For $i = 1$ to n, until stream i is completely assigned, assign as much as possible of this stream to the machine with smallest index not yet satisfying constraint (6.7) with equality.

6.3.4. LP characterization. Consider the heterogeneous model under the P-K cost function. Then, the optimum solution in the fractional model can be characterized by the following program:

$$\begin{array}{lll}
\text{minimize} & z & \\
\text{subject to} & \dfrac{\sum_{i=1}^{n} V_i x_i^j}{b_j (b_j - \sum_{i=1}^{n} \lambda_i x_i^j)} \leq z & \forall j \in [m], \\
 & \displaystyle\sum_{j=1}^{m} x_{ij} = 1 & \forall i \in [n], \\
 & x_{ij} \geq 0 & \forall i, j.
\end{array}$$

This program can be equivalently rewritten as the following program, called P_1:

$$\begin{array}{lll}
\text{minimize} & z & \\
\text{subject to} & \displaystyle\sum_{i=1}^{n} \left(\frac{V_i}{z\, b_j^2} + \frac{\lambda_i}{b_j} \right) x_i^j \leq 1 & \forall j \in [m], \\
 & \displaystyle\sum_{j=1}^{m} x_{ij} = 1 & \forall i \in [n], \\
 & x_{ij} \geq 0 & \forall i, j.
\end{array}$$

Now assume that z is fixed to the smallest possible value such that P_1 is feasible; then we obtain a linear program that is very similar to linear program LP. This similarity can be captured by defining $v_i = V_i$ and $e_i = \lambda_i$ for $i \in [n]$, and $p_j = 1/(z b_j^2)$, $q_j = 1/b_j$ for $j \in [m]$. We observe that for any fixed value of z we can test feasibility of the program P_1 using the greedy assignment algorithm above. Finally, it can be shown that we can use binary search for z to solve program P_1. The order of the servers is given by decreasing values of $\frac{p_j}{q_j} = \frac{1}{z b_j}$, which means by increasing values of bandwidths b_j. The order of data streams is given by increasing values of $\frac{v_i}{e_i} = \frac{V_i}{\lambda_i}$. As we have chosen z to be the smallest possible value such that the constraints are feasible, all feasible solutions of LP are tight.

Similar arguments show that in the worst-case Nash equilibrium, if we keep the order of the servers by increasing values of bandwidths b_j, then the order of data streams is now given by decreasing values of $\frac{v_i}{e_i} = \frac{V_i}{\lambda_i}$. To show this property, we observe that the costs on all servers in a fractional Nash equilibrium must be equal. Using this observation, we can characterize the worst-case Nash fractional equilibrium by the following program:

$$\begin{array}{lll} \text{maximize} & z & \\ \text{subject to} & \dfrac{\sum_{i=1}^{n} V_i x_i^j}{b_j(b_j - \sum_{i=1}^{n} \lambda_i x_i^j)} \geq z & \forall j \in [m], \\ & \displaystyle\sum_{j=1}^{m} x_{ij} = 1 & \forall i \in [n], \\ & x_{ij} \geq 0 & \forall i, j. \end{array}$$

This program is similar to program P_1 above, and it can be turned in the same way into a linear program almost equal to LP. The only difference is that the inequality $\leq$ in constraint (6.7) changes into $\geq$. Starting from this, one can prove the claim about the structure of the worst-case Nash equilibrium by following the arguments in the proof of Lemma 6.12 with reversed inequalities.

6.3.5. Proof of Theorem 6.11.

Now we will use the LP characterization of optimal solutions and the worst-case Nash equilibria from section 6.3.4 to prove Theorem 6.11.

Proof of Theorem 6.11. *An upper bound.* Assume that the bandwidths of the servers are such that $b_1 \geq b_2 \geq \cdots \geq b_m$. We are given a set of data streams with weights and variances λ_i, V_i, respectively, $i \in [n]$. We first consider a relaxation of the problem by replacing each data stream λ_i, V_i by two new data streams 0, V_i and λ_i, 0. We also delete from the new problem instance all data streams of the form 0, 0.

The value C of the worst-case Nash equilibrium for the relaxed problem is obviously not smaller than that for the original problem. Also, the value *opt* of the optimum solution for the relaxed problem is not greater than that for the original problem. Therefore, an upper bound on the price of anarchy R^* for the relaxed problem implies the same upper bound for the original price of anarchy. From now on, we work with the relaxed problem.

Define $\Lambda = \sum_{i \in [n]} \lambda_i$, $V = \sum_{i \in [n]} V_i$, $\tilde{S} = \sum_{j \in [m]} b_j$, and $G = \tilde{S} - \Lambda$. We will show that $opt \geq \frac{V}{b_1 G}$ and $C \leq \frac{V}{b_m G}$, which implies $R^* \leq b_1/b_m = S$.

We use now the order of data streams in the *opt* solution. (Note that we can extend the local exchange lemma for cases where $\lambda_i = 0$ for some data streams.) This order implies that there exists a j' such that *opt* maps all streams of the form λ_i, 0 to a subset of servers $\{j', j'+1, \ldots, m\}$. Assume that each server $j \geq j'$ is filled with $\lambda(j)$ amount of data stream weights, and, therefore, server $j \geq j'$ is filled up to a "gap" of

$$g_j = b_j - \lambda(j).$$

Furthermore, the *opt* solution maps all streams of the form $0, V_i$ to the servers

$\{1, 2, \ldots, j''\}$ for some j''. Let $V(j)$ denote the "amount of variance" received by server $j \in \{1, \ldots, m\}$.

We first observe that $j'' = m$. Assume this is false; i.e., there is a server $j_1 \in \{j''+1, \ldots, m\}$ with $V(j_1) = 0$. Then, we can just move a fraction of some type 0, V_i stream from a server $j_2 \in \{1, 2, \ldots, j''\}$ to server j_1. The cost on server j_1 was zero before and after this operation increases marginally. But then, we can also reassign type 0, V_i streams more evenly on the servers $\{1, 2, \ldots, j''\}$ using the free space on server j_2. This decreases the cost of the solution and thus yields a contradiction with the *opt* solution.

The cost on all servers $\{1, 2, \ldots, m\}$ must be identical; that is,

$$\frac{V(1)}{(b_1)^2} = \frac{V(2)}{(b_2)^2} = \cdots = \frac{V(j'-1)}{(b_{j'-1})^2} = \frac{V(j')}{b_{j'} \cdot g_{j'}} = \frac{V(j'+1)}{b_{j'+1} \cdot g_{j'+1}} = \cdots = \frac{V(m)}{b_m \cdot g_m}.$$

Consequently,

$$V(m) = \frac{V \cdot b_m \cdot g_m}{\sum_{j=1}^{j'-1} (b_j)^2 + \sum_{j=j'}^{m} b_j \cdot g_j}.$$

Therefore, the waiting time at server m and, hence, the value of *opt* is

$$\frac{V(m)}{b_m \cdot g_m} = \frac{V}{\sum_{j=1}^{j'-1} (b_j)^2 + \sum_{j=j'}^{m} b_j \cdot g_j} \geq \frac{V}{b_1 \cdot \left(\sum_{j=1}^{j'-1} b_j + \sum_{j=j'}^{m} g_j \right)} = \frac{V}{b_1 \cdot G}.$$

Analogously, we can use the reverse order of the streams in the worst-case Nash equilibrium, and we can argue that the variances must be split on all the servers (otherwise, the Nash property would be violated). This, with an appropriate definition of a "gap" g_j and j', shows that the value of Nash is

$$\frac{V}{\sum_{j=1}^{j'} b_j \cdot g_j + \sum_{j=j'+1}^{m} (b_j)^2} \leq \frac{V}{b_m \cdot \left(\sum_{j=1}^{j'} g_j + \sum_{j=j'+1}^{m} b_j \right)} = \frac{V}{b_m \cdot G}.$$

This completes the proof of the upper bound.

A lower bound. We show here a lower bound in Theorem 6.11, that is, $R^* \geq S$. Assume we are given two servers with $b_1 = 1$ and $b_2 = S$ and two data streams with $\lambda_1 = 0, V_1 = 1$ and $\lambda_2 = S, V_2 = 0$. We can use now the local exchange lemma to show that $R^* \geq S$, but we will show an analytical argument here.

Let $\lambda(j)$ be the amount of stream 2 assigned to server $j \in \{1, 2\}$, and let $V(j)$ be the amount of stream 1 assigned to server $j \in \{1, 2\}$. We have that $\lambda(1) + \lambda(2) = S$ and $V(1) + V(2) = 1$. We know that in *opt* and in the Nash solutions, the costs are the same on each server. Thus let us write that

$$\frac{V(1)}{1 - \lambda(1)} = \frac{V(2)}{S \cdot (S - \lambda(2))}.$$

Now, we have that $\lambda(2) = S - \lambda(1)$ and $V(2) = 1 - V(1)$. After plugging these values in the condition above, we obtain $V(1) = \frac{1-\lambda(1)}{(S-1)\cdot\lambda(1)+1}$. Consider now a function

$$f(\lambda(1)) = \frac{V(1)}{1 - \lambda(1)} = \frac{\frac{1-\lambda(1)}{(S-1)\cdot\lambda(1)+1}}{1 - \lambda(1)}.$$

It can be shown that f has a minimum when $\lambda(1) \to 1$, equal to $1/S$, and so $opt \leq 1/S$ (we use here the l'Hôpital rule from Calculus to prove this fact). We can also easily show that f has a maximum when $\lambda(1) = 0$, which gives that $C \geq 1$. Therefore, we obtain finally that $R^* = C/opt \geq S$, which completes the proof of Theorem 6.11. □

In light of Theorem 6.11, we conclude that the optimality of fractional flow in Nash equilibrium is a special property of homogeneous traffic on server farms, and hence one must take into account the heterogeneous nature of Web traffic when studying the price of selfish routing on the Internet.

6.4. Servers with parallel channels and rejection. Until now we assumed that all requests are served, regardless of how long they have to wait for service. In practice, however, Web servers can and have to reject requests when they are overloaded. For simplicity, let us assume that a server rejects requests whenever all service channels are occupied, and then these requests disappear from the system. In this case, the fraction of rejected requests is completely independent of the service time distribution. In other words, there is no difference between homogeneous and heterogeneous traffic under this service model. In fact, the fraction of rejected requests can be derived from the *Erlang loss formula*; see, e.g., [10, 11]. We obtain the following cost function family $\mathcal{F}_{\mathbb{N}_{>0}}$ for servers that can open up to b channels simultaneously:

$$f_b(x_1, \ldots, x_n) = \frac{\lambda^b/b!}{\sum_{k=0}^{b} \lambda^k/k!},$$

describing the fraction of rejected requests, where $\lambda = \sum_{i=1}^{n} \lambda_i \, x_i$. To simplify notation, let us simply consider f_b as a function of λ. Then it is easy to check that the family of Erlang loss functions is monotone. Thus, Theorem 5.1 immediately implies the following on the fractional price of anarchy.

COROLLARY 6.13. *For the family $\mathcal{F}_{\mathbb{N}_{>0}}$ of the Erlang loss cost functions, $R^* = 1$.*

On first glance, the family of Erlang loss functions makes an innocent impression. Indeed, these functions are continuous, convex, and monotonically increasing in λ and $f_b'(\lambda) \leq 1/b$ for every $\lambda \geq 0$. Nevertheless, the criterion from Theorem 5.2 can be used to show that the integral price of anarchy of the Erlang loss family is unbounded.

THEOREM 6.14. *For the family $\mathcal{F}_{\mathbb{N}_{>0}}$ of Erlang loss cost functions, $R = \infty$ and $\overline{R} \geq m$.*

Proof. We show that the cost increases by an unbounded factor around $\lambda = b$. To see this, let us consider the family of functions $\mathcal{F}_{\mathbb{N}_{>0}}$ with $F_b(x) = f_b(b\,x)$, i.e., the Erlang loss functions in terms of relative load. We claim that, for every $x \geq 0$,

$$\lim_{b\to\infty} F_b(x) = \max\left\{0, \frac{x-1}{x}\right\}.$$

As a consequence of this claim, for every $x \in (0.5, 1]$ and every $\alpha \geq 1$, there exists a sufficiently large $b \geq 1$ such that $F_b(2x)/F_b(x) > \alpha$. Consequently, there does not exist $\alpha \geq 1$ satisfying the criterion from Theorem 5.2, which implies $R = \infty$. Applying Theorem 5.7 yields $\overline{R} \geq m$.

It remains to prove the claim on $\lim_{b\to\infty} F_b(x)$. Let us first consider the case $x \leq 1$. We have to show $\lim_{b\to\infty} F_b(x) = 0$. We have

$$\frac{1}{f_b(\lambda)} = \sum_{k=0}^{b} \frac{b!}{\lambda^{b-k} k!} \geq \sum_{k=b-\lfloor\sqrt{b}\rfloor}^{b} \left(\frac{b-\lfloor\sqrt{b}\rfloor+1}{\lambda}\right)^{b-k}.$$

Applying $\lambda = xb \leq b$, we obtain

$$\frac{1}{f_b(\lambda)} \geq \sum_{k=b-\lfloor\sqrt{b}\rfloor}^{b} \left(\frac{b-\sqrt{b}}{b}\right)^{\sqrt{b}} \geq \frac{\sqrt{b}}{4},$$

where the last inequality holds for $b \geq 4$, as then $((b-\sqrt{b})/b)^{\sqrt{b}} \geq \frac{1}{4}$. Consequently, $F_b(x) = f_b(\lambda) \leq 4/\sqrt{b}$, and, hence, $\lim_{b\to\infty} F_b(x) = 0$.

Now assume $x = \lambda/b > 1$. On one hand, we obtain

$$\frac{1}{f_b(\lambda)} = \sum_{k=0}^{b} \frac{b!}{\lambda^{b-k}k!} = \sum_{k=0}^{b} x^{k-b}\frac{b!}{b^{b-k}k!} \leq \sum_{k\leq b} x^{k-b} = \frac{x}{x-1}.$$

On the other hand, we get

$$\frac{1}{f_b(\lambda)} = \sum_{k=0}^{b} x^{k-b}\frac{b!}{b^{b-k}k!} \geq \sum_{k=b-\lfloor b^{1/3}\rfloor}^{b} x^{k-b}\left(1-\frac{1}{b^{2/3}}\right)^{b^{1/3}} \overset{b\to\infty}{=} \frac{x}{x-1}.$$

Consequently, $\lim_{b\to\infty} F_b(x) = \lim_{b\to\infty} f_b(xb) = \frac{x-1}{x}$. □

In contrast to the monotone queueing functions from section 6.1, the source of the troubles for the Erlang loss cost functions is not an ∞-pole, but the rapid increase from tiny to small cost in the region around $\lambda = b$. One might thus hope that the absolute cost of selfish routing under the Erlang loss cost function family is small. In fact, this is confirmed by the following theorem.

THEOREM 6.15. *Let $\delta \geq 2/\log_2 m$. Consider a server farm of m servers with bandwidths $b_1 \geq \cdots \geq b_m$ and cost functions from the family $\mathcal{F}_{\mathbb{N}_{>0}}$ of Erlang loss cost functions. Suppose $\sum_{i\in[n]} \lambda_i \leq \frac{1}{6e}\sum_{j\in[m]} b_j$ and $\max_{i\in[n]} \lambda_i \leq \frac{b_m}{3\delta\log_2 m}$. Then, any Nash equilibrium has social cost at most $m^{-\delta+1} + m\cdot 2^{-\lfloor b_m/4\rfloor}$.*

Proof. Suppose that $\sum_{i\in[n]} \lambda_i \leq \sum_{j\in[m]} b_j/6\,e$. Suppose the maximum weight over all streams is $W = b_m/(3\,\delta\,\log m)$ for some given $\delta > 0$ with $\delta\log m \geq 2$. We have to show that for every Nash equilibrium $C \leq m^{-\delta+1} + m\cdot 2^{-\lfloor b_m/4\rfloor}$.

For $j \in [m]$, let X_j be the random variable describing the injected load on server j. A simple averaging argument shows the existence of a server $q \in [m]$ with

$$\mathbf{E}[X_q] \;=\; \sum_{i\in[n]} p_i^q\,\lambda_i \;\leq\; \frac{b_q}{6\,e}.$$

This server has expected cost $\mathbf{E}[C_q] = \mathbf{E}\left[f_{b_q}(X_q)\right]$. Consider an arbitrary server $j \in [m]$. Suppose there exists $i \in [n]$ with $p_i^j > 0$. Then the Nash equilibrium property gives $c_i^j \leq c_i^q$ so that

$$\mathbf{E}[C_j] \leq \mathbf{E}\left[C_j|x_i^j = 1\right] \;=\; c_i^j \;\leq\; c_i^q \;=\; \mathbf{E}[C_q|x_i^q = 1] \;\leq\; \mathbf{E}\left[f_{b_q}(X_q+W)\right].$$

If $p_i^j = 0$ for every $i \in [n]$, then $\mathbf{E}[C_j] = 0 \leq \mathbf{E}\left[f_{b_q}(X_q+W)\right]$ as well. Furthermore, observe that $f_{b_q}(X_q+W) \leq 1$ as the Erlang loss cost function describes the fraction of rejected requests. Hence, for any server $j \in [m]$ we have

$$\begin{aligned}\mathbf{E}[C_j] &\leq \mathbf{Pr}\left[X_q \leq \frac{b_q}{3}\right]\cdot f_{b_q}\left(\frac{b_q}{3}+W\right) + \mathbf{Pr}\left[X_q > \frac{b_q}{3}\right]\cdot 1\\ &\leq f_{b_q}\left(\frac{b_q}{3}+W\right) + \mathbf{Pr}\left[X_q > \frac{b_q}{3}\right].\end{aligned}$$

Let us estimate these two terms as follows. Observe that $W \leq b_m/6 \leq b_q/6$. Consequently,

$$f_{b_q}\left(\frac{b_q}{3}+W\right) \leq f_{b_q}\left(\frac{b_q}{2}\right) \leq \frac{\left(\frac{b_q}{2}\right)^{b_q}/b_q!}{\sum_{k=0}^{b_q}\left(\frac{b_q}{2}\right)^{k}/k!} \leq \frac{\left(\frac{b_q}{2}\right)^{\lceil b_q/2\rceil}\left\lfloor\frac{b_q}{2}\right\rfloor!}{b_q!},$$

where the last inequality follows by ignoring all additive terms in the denominator except the one with index $k=\lfloor b_q/2\rfloor$. Now observe that the last term can be rewritten as

$$\prod_{i=0}^{\lceil b_q/2\rceil-1}\frac{\frac{b_q}{2}}{b_q-i} \text{ or, alternatively, } \prod_{i=0}^{\lceil b_q/2\rceil-1}\frac{\frac{b_q}{2}}{\left\lfloor\frac{b_q}{2}\right\rfloor+1+i}.$$

Observe that the *ith* term in the first product corresponds to the $(\lceil b_q/2\rceil-1-i)th$ term in the second one. Combining these products by selecting the first $\lfloor\lceil b_q/2\rceil/2\rfloor$ terms from each of them, we get

$$f_{b_q}\left(\frac{b_q}{3}+W\right) \leq \prod_{i=0}^{\lfloor\lceil b_q/2\rceil/2\rfloor-1}\frac{\left(\frac{b_q}{2}\right)^2}{(b_q-i)\left(\left\lfloor\frac{b_q}{2}\right\rfloor+1+i\right)} \leq \prod_{i=0}^{\lfloor\lceil b_q/2\rceil/2\rfloor-1}\frac{\left(\frac{b_q}{2}\right)^2}{\frac{(b_q)^2}{2}+\frac{ib_q}{2}-i^2}.$$

As $i \leq b_q/2$, this gives

$$f_{b_q}\left(\frac{b_q}{3}+W\right) \leq \prod_{i=0}^{\lfloor\lceil b_q/2\rceil/2\rfloor-1}\frac{\left(\frac{b_q}{2}\right)^2}{\frac{(b_q)^2}{2}} \leq 2^{-\lfloor b_q/4\rfloor} \leq 2^{-\lfloor b_m/4\rfloor}.$$

Furthermore, using $\mathbf{E}\left[X_q\right] \leq b_q/(6e)$ and the upper bound for the maximum weight over all streams of W, we can apply the standard Hoeffding bound (see, e.g., [5]) to obtain

$$\mathbf{Pr}\left[X_q > \frac{b_q}{3}\right] \;\leq\; 2^{-b_q/3W} \;\leq\; m^{-\delta}$$

because $b_q \geq b_m$ and $W = b_m/(3\,\delta\,\log m)$. Hence, we can conclude

$$C \;=\; \mathbf{E}\left[\max_{j\in[m]} C_j\right] \;\leq\; m\cdot\max_{j\in[m]}\left(\mathbf{E}\left[C_j\right]\right) \;\leq\; m^{-\delta+1}+m\,2^{-\lfloor b_m/4\rfloor}.$$

This completes the proof of Theorem 6.15. □

In words, Theorem 6.15 says that if the total injected load is at most a constant fraction of the total bandwidth and every stream has not too large a weight, that is, streams are $O(\frac{1}{\log m})$-small, then the fraction of rejected requests is at most $m^{-\delta+1}+m\,2^{-\lfloor b_m/4\rfloor}$, where δ denotes an arbitrary constant. Under the same conditions, an optimal allocation would reject a fraction of at least $2^{-\Omega(b_1)}$ packets. Taking into account that typical Web servers can open several hundred TCP connections simultaneously, so that b_m can be assumed to be quite large, we conclude that the cost of selfish routing is very small in absolute terms even though the price of anarchy comparing these costs with the optimal cost is unbounded.

7. Conclusions. Our results have important algorithmic consequences in that the choice of the queueing discipline should take into account the possible performance degradation due to selfish and uncoordinated behavior of network users.

We have shown that the price of anarchy for closed queueing systems is unbounded. The same is true for server farms that reject requests in case of overload. However, there is a fundamental difference between these two kinds of queueing policies. Because of the infinity pole the delay under selfish routing in closed queueing systems is in general unbounded. In fact, we have explicitly shown that selfish routing in closed queueing systems can lead to an arbitrary large delay even when the total injected load can potentially be served by a single server. In contrast, the fraction of rejected requests under selfish routing can be bounded above by a function that is exponentially small in the number of TCP connections that can be opened simultaneously.

We conclude that server farms that serve all requests, regardless of how long requests have to wait, cannot give any reasonable guarantee on the quality of service when selfish agents manage the traffic. However, if requests are allowed to be rejected, then it is possible to guarantee a high quality of service for every individual request stream. Thus, the typical practice of rejecting requests in case of overload is a necessary condition to ensure efficient service under game theoretic measures.

REFERENCES

[1] B. Awerbuch, Y. Azar, and A. Epstein, *The price of routing unsplittable flow*, in Proceedings of the 37th Annual ACM Symposium on Theory of Computing (STOC), ACM, New York, 2005, pp. 57–66.

[2] M. Beckmann, C. B. McGuire, and C. B. Winston, *Studies in the Economics of Transportation*, Yale University Press, New Haven, CT, 1956.

[3] G. Christodoulou and E. Koutsoupias, *The price of anarchy of finite congestion games*, in Proceedings of the 37th Annual ACM Symposium on Theory of Computing (STOC), ACM, New York, 2005, pp. 67–73.

[4] M. E. Crovella and A. Bestavros, *Self-similarity in World Wide Web traffic: Evidence and possible causes*, IEEE/ACM Trans. Networking, 5 (1997), pp. 835–846.

[5] A. Czumaj and B. Vöcking, *Tight bounds for the worst-case equilibria*, ACM Trans. Algorithms, 3 (1) (2007), article 4.

[6] F. Douglis, A. Feldmann, B. Krishnamurthy, and J. Mogul, *Rate of change and other metrics: A live study of the World Wide Web*, in Proceedings of the 1st USENIX Symposium on Internet Technologies and Systems, Anaheim, CA, 1997, pp. 147–158.

[7] A. Feldmann, A. C. Gilbert, P. Huang, and W. Willinger, *Dynamics of IP traffic: A study of the role of variability and the impact of control*, in Proceedings of the ACM SIGCOMM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication, ACM, New York, 1999, pp. 301–313.

[8] M. Frank, *The Braess paradox*, Math. Programming, 20 (1981), pp. 283–302.

[9] P. Franken, D. Konig, U. Arndt, and V. Schmidt, *Queues and Point Processes*, Akademie-Verlag, Berlin; Wiley, Chichester, UK, 1982.

[10] E. Friedman, *Genericity and congestion control in selfish routing*, in Proceedings of the 43rd IEEE Conference on Decision and Control, IEEE Control Systems Society, Piscataway, NJ, 2004, pp. 4667–4672.

[11] D. Gross and C. M. Harris, *Fundamentals of Queueing Theory*, 3rd ed., John Wiley and Sons, New York, 1998.

[12] L. Kleinrock, *Queueing Systems. Volume* I: *Theory*, John Wiley and Sons, New York, 1975.

[13] Y. A. Korilis, A. A. Lazar, and A. Orda, *Capacity allocation under noncooperative routing*, IEEE Trans. Automat. Control, 42 (1997), pp. 309–325.

[14] Y. A. Korilis, A. A. Lazar, and A. Orda, *Avoiding the Braess paradox in noncooperative networks*, J. Appl. Probab., 36 (1999), pp. 211–212.

[15] E. Koutsoupias, M. Mavronicolas, and P. Spirakis, *Approximate equilibria and ball fusion*, Theory Comput. Syst., 36 (2003), pp. 683–693.

[16] E. Koutsoupias and C. H. Papadimitriou, *Worst-case equilibria*, in Proceedings of the 16th Annual Symposium on Theoretical Aspects of Computer Science (STACS), Trier, Germany, 1999, pp. 404–413.

[17] A. A. Lazar, A. Orda, and D. E. Penderakis, *Virtual path bandwidth allocation in multiuser networks*, IEEE/ACM Trans. Networking, 5 (1997), pp. 861–871.

[18] M. Mavronicolas and P. Spirakis, *The price of selfish routing*, Algorithmica, 48 (2007), pp. 91–126.

[19] R. Motwani and P. Raghavan, *Randomized Algorithms*, Cambridge University Press, New York, 1995.

[20] A. Orda, R. Rom, and N. Shimkin, *Competitive routing in multi-user communication networks*, IEEE/ACM Trans. Networking, 1 (1993), pp. 510–521.

[21] C. H. Papadimitriou and M. Yannakakis, *On complexity as bounded rationality*, in Proceedings of the 26th Annual ACM Symposium on Theory of Computing (STOC), ACM, New York, 1994, pp. 726–733.

[22] K. Park, G. Kim, and M. E. Crovella, *On the relationship between file sizes, transport protocols, and self-similar network traffic*, in Proceedings of the 4th IEEE International Conference on Network Protocols, IEEE, Washington, DC, 1996, pp. 171–180.

[23] V. Paxson and S. Floyd, *Wide area traffic: The failure of Poisson modeling*, IEEE/ACM Trans. Networking, 3 (1995), pp. 226–244.

[24] T. Roughgarden, *Stackelberg scheduling strategies*, SIAM J. Comput., 33 (2004), pp. 332–350.

[25] T. Roughgarden, *On the severity of Braess's Paradox: Designing networks for selfish users is hard*, J. Comput. System Sci., 72 (2006), pp. 922–953.

[26] T. Roughgarden, *How unfair is optimal routing?*, in Proceedings of the 13th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), ACM, New York, SIAM, Philadelphia, 2002, pp. 203–204.

[27] T. Roughgarden, *The price of anarchy is independent of the network topology*, J. Comput. System Sci., 67 (2003), pp. 341–364.

[28] T. Roughgarden, *Routing games*, in Algorithmic Game Theory, N. Nisan, T. Roughgarden, É. Tardos, and V. V. Vazirani, eds., Cambridge University Press, Cambridge, UK, 2007, pp. 461–486.

[29] T. Roughgarden and É. Tardos, *How bad is selfish routing?*, J. ACM, 49 (2002), pp. 236–259.

[30] T. Roughgarden and É. Tardos, *Bounding the inefficiency of equilibria in nonatomic congestion games*, Games Econom. Behav., 47 (2004), pp. 389–403.

[31] R. Steinberg and W. I. Zangwill, *The prevalence of Braess' paradox*, Transportation Science, 17 (1983), pp. 301–318.

[32] B. Vöcking, *Selfish load balancing*, in Algorithmic Game Theory, N. Nisan, T. Roughgarden, É. Tardos, and V. V. Vazirani, eds., Cambridge University Press, Cambridge, UK, 2007, pp. 517–542.

[33] J. G. Wardrop, *Some theoretic aspects of road traffic research*, Proc. Inst. Civil Engineers Part II, 1 (1952), pp. 325–378.