The Influence of Link Restrictions on (Random) Selfish Routing

Martin Hoefer[1] * and Alexander Souza[2]

[1] Department of Computer Science, RWTH Aachen University, Germany.
mhoefer@informatik.rwth-aachen.de
[2] Department of Computer Science, University of Freiburg, Germany.
souza@informatik.uni-freiburg.de

Abstract. In this paper we consider the influence of link restrictions on the price of anarchy for several social cost functions in the following model of selfish routing. Each of n players in a network game seeks to send a message with a certain length by choosing one of m parallel links. Each player is restricted to transmit over a certain subset of links and desires to minimize his own transmission-time (latency). We study Nash equilibria of the game, in which no player can decrease his latency by unilaterally changing his link. Our analysis of this game captures two important aspects of network traffic: the dependency of the overall network performance on the total traffic t and fluctuations in the length of the respective message-lengths. For the latter we use a probabilistic model in which message lengths are random variables.

We evaluate the (expected) price of anarchy of the game for two social cost functions. For total latency cost, we show the tight result that the price of anarchy is essentially $\Theta\left(n\sqrt{m}/t\right)$. Hence, even for congested networks, when the traffic is linear in the number of players, Nash equilibria approximate the social optimum only by a factor of $\Theta\left(\sqrt{m}\right)$. This efficiency loss is caused by link restrictions and remains stable even under message fluctuations, which contrasts the unrestricted case where Nash equilibria achieve a constant factor approximation. For maximum latency the price of anarchy is at most $1 + m^2/t$. In this case Nash equilibria can be (almost) optimal solutions for congested networks depending on the values for m and t. In addition, our analyses yield average-case analyses of a polynomial time algorithm for computing Nash equilibria in this model.

1 Introduction

Recently, there has been a lot of interest in considering network users as non-cooperative selfish players that unilaterally seek to optimize their experienced network latency. This serves to quantify the deterioration

* Supported by DFG Research Training Group “AlgoSyn” and by DFG Research Cluster “UMIC” within Excellence Initiative of the German government.

of the total system performance, and it builds a foundation to derive protocols taking possible selfish defection into account. In their seminal work [13], Koutsoupias and Papadimitriou initiated this research direction by introducing the KP-model for selfish routing. Each of n players seeks to send a message with respective length t_j across a network consisting of m parallel capacitated links. The cost of a player j, called his *latency* ℓ_j, is the total length of messages on his chosen link i, scaled with the respective capacity. The latency corresponds to the duration of the transmission when the channel is shared by a set of players. Now each player strives to optimize his personally experienced latency by changing the chosen link for his message. He is satisfied with his link choice (also referred to as his strategy) if by unilaterally changing his link he cannot decrease his cost. If all players are satisfied, then the system is said to be in a stable state, called a *Nash equilibrium*.

In order to relate selfishly obtained stable solutions with those of an (imaginary) central authority, it is necessary to distinguish between the cost of the individual players and the *social cost* of the whole system caused by the community of all players. Naturally, depending on the choice of a social cost function selfish behavior is not always optimal. Consequently, the question arises how much worse a Nash equilibrium can be than the optimum. Koutsoupias and Papadimitriou [13] introduced the *price of anarchy* which is the ratio of the social cost of the worst Nash equilibrium and the optimum social cost, and proved initial bounds for special cases of the KP-model with maximum latency cost. Subsequently, generalized models with different latency functions, social cost functions and network topologies were considered for instance in [6, 5, 8, 9, 1, 4]. For a recent survey on results related to network congestion games see [12].

In this paper we treat a generalization of the KP-model in which the link set a player i can choose from is a restricted subset of all links available. This model was treated before with maximum latency and polynomial load social cost. For maximum latency computing the social optimum solution is a special case of generalized assignment problems and the single source unsplittable flow problem [14, 11]. Gairing et al. [7] gave a $(2 - 1/t_{\max})$-approximation algorithm for optimizing the social cost. They also showed how to compute in polynomial time a Nash equilibrium from any given starting solution without deteriorating the social cost, so the price of stability is 1. In [2] the price of anarchy for maximum latency was shown to be $O(\log m/ \log\log m)$ and to decrease with the ratio $r = \text{cost}(s^*)/t_{\max}$. In particular, for $r = \Omega(\log m/\epsilon^2)$ it is $1 + \epsilon$. Quadratic load social cost was recently studied in [3, 18]. Suri et al. [18]

show that the price of anarchy for identical machines is at least 2.012067, and Caragiannis et al. [3] provide a matching upper bound.

In contrast to previous work, we capture two important aspects of network traffic: the dependency of the overall network performance on the *total traffic* $t = \sum_j t_j$ and *fluctuations* in the length of the respective message-lengths t_j. In our model of fluctuation, the message-lengths are random variables T_j and the quality of equilibria is judged with the expected value of the price of anarchy, respectively stability. This idea of an *expected price of anarchy* was recently introduced by Mavronicolas et al. [15] (under the name diffuse price of anarchy) in the context the unrestricted KP-model with a cost-sharing mechanism. We considered the expected price of anarchy in [10], in which we were mostly concerned with the pure Nash equilibria of the unrestricted KP-model and the total latency $\sum_j \ell_j$ of all players. One main conclusion therein was that for highly congested networks, i.e., t being linear in n, Nash equilibria approximate the optimum solution within a constant factor.

In this paper, we characterize the loss of performance of Nash equilibria due to the presence of link restrictions. We show that the prices of stability and anarchy are essentially $\Theta\left(n\sqrt{m}/t\right)$ for total latency. Perhaps surprisingly this behaviour remains stable even in the stochastic counterpart. This means that – in contrast to other related average-case analyses, e.g., [17, 16] – the averaging effects of fluctuations do not necessarily yield improved expected prices of the game.

Our results foster an interesting new research direction connecting game theory and average-case analysis in the context of traffic allocation and scheduling. We consider efficiency measures including randomness by presenting *tight* bounds on the (expected) price of anarchy. By capturing a notion of fluctuation, we bring a network game closer to practice. Secondly, our analysis yields an average-case analysis on the expected performance of a generic approximation algorithm for various scheduling problems. Most notably, our analysis holds under weak probabilistic assumptions. This extends previous work, e.g. [10, 17, 16] on average-case analyses of scheduling on identical *unrestricted* machines.

1.1 Model and Notation

We formulate the KP-model with scheduling terminology, where each link corresponds to one of m identical parallel machines. There are n players in the game, and each player seeks to assign a task to one of the machines. Each task j has a certain finite *length* t_j. We scale all task lengths by a positive factor without changing the approximation factors, i.e., we assume normalization $t_j \in [0,1]$, and w.l.o.g. $n \geq m$ throughout. With each player j we associate a set $A_j \neq \emptyset$ of *allowed machines*, and each player is restricted to assignment only to machines in the set A_j. The strategy of a player is the choice of one of the allowed machines. A *schedule* is a function s that maps each task j to a machine i obeying the restrictions A_j. The total length on machine i is its *load* $w_i = \sum_{k \text{ on } i} t_k$. Each machine i executes its assigned tasks in parallel and hence the finishing-time of a task j is proportional to the total length on the chosen link i, i.e., its *latency* is $\ell_j = \sum_{k \text{ on } i} t_k = w_i$. The disutility of each player is the latency of its task, i.e., the selfish incentive of every player is to minimize the individual latency.

A schedule s is said to be in a (pure) Nash equilibrium if no player can decrease his latency by unilaterally changing the machine his task is processed on. More formally, the schedule s has the property that for each task j

$$w_i + t_j \geq w_{s(j)} \quad \text{holds for every } i \in A_j. \tag{1}$$

This game is known to always admit pure Nash equilibria, see e.g. [7].

Schedules are valued with a certain *(social) cost* function cost : $\Sigma \to \mathbb{R}_+$, where Σ denotes the set of all schedules. A Nash equilibrium is simply a schedule that satisfies the stability criterion (1), whereas an *optimum* schedule minimizes the cost function over all possible schedules. Hence, it is natural to consider how much worse Nash equilibria can be compared to the optimum. The *price of anarchy* [13] relates the Nash equilibrium with highest social cost to the optimum, i.e, it compares the "worst" Nash equilibrium with the best possible solution. In contrast, the *price of stability* relates the Nash equilibrium with lowest social cost to the optimum, i.e, it compares the "best" Nash equilibrium with the best possible solution.

1.2 Our Concepts and Results

The main matter of this paper is to investigate the influence of link restrictions on Nash equilibria. We consider two different social cost functions: total latency $\sum_{j \in J} \ell_j$ and maximum latency $\max_{j \in J} \ell_j$.

Our focus lies on two important aspects of network traffic: the influence of the total traffic upon the quality of Nash equilibria and the question if fluctuations in the task-lengths have an positive averaging effect. In terms of fluctuations, we consider the following natural stochastic model. Throughout, upper-case letters denote random variables, lower-case letters their realisations, respectively constants.

Let the task-length T_j of a task j be a random variable over a bounded interval with expectation $\mathbb{E}[T_j]$. As before, a schedule is a *Nash equilibrium* if (1) holds, i.e., if the concrete *realisations* t_j of the random variables T_j satisfy the stability criterion. Consequently, the set of schedules that are Nash equilibria is a random variable itself. We define the *expected price of anarchy*

$$\mathrm{EPoA}(\Sigma) = \mathbb{E}\left[\max\left\{\frac{\mathrm{cost}(S)}{\mathrm{cost}(S^*)} : \ S \in \Sigma \text{ is a Nash equilibrium}\right\}\right].$$

The *expected price of stability* is obtained by replacing the maximum by the minimum in straightforward manner. Notice that each expected value is taken with respect to the random task-lengths T_j. This means that the expectation is accumulated by evaluating the prices for each outcome t_j of the random variables T_j and weighting with the respective probability.

In Section 2 we consider total latency $\sum_j \ell_j$, for which [10] shows that prices of stability and anarchy are $\Theta(n/t)$, i.e., they are both decreasing with t. Theorem 1, respectively Theorem 2, provide *tight* lower and upper bounds for the case with link restrictions: we show that the prices of anarchy, respectively stability are $\Theta(n\sqrt{m}/t)$. The question arises whether fluctuations in task-lengths help reducing this bound. Unfortunately, we show that the bounds remains stable. The expected prices of anarchy and stability are $\Theta(n\sqrt{m}/\mathbb{E}[T])$ under relatively weak assumptions on the probability distributions of the T_j.

For maximum latency $\max_j \ell_j$, it is already known (see [2]) that the price of stability is 1 and that the price of anarchy follows a tradeoff depending on the largest task length and the cost of the social optimum. We show a similar tradeoff in Theorem 4: the price of anarchy is at most $1 + m^2/t$ and even in expectation it is at most $1 + m^2/\mathbb{E}[T]$. Hence, Nash equilibria are almost optimal for congested networks even with link restrictions.

Moreover, there is an algorithm due to Gairing et al. [7] which computes pure Nash equilibria for our game in polynomial time. Our analyses of the expected prices of anarchy of these social cost functions provide average-case analyses of that algorithm, see, e.g. Theorem 3.

2 Total Latency Cost

In this section, we consider the social cost function *total latency* $\mathrm{cost}(s) = \sum_j \ell_j$. Throughout, let p_i denote the number of *players* that use machine i, let $w_i = \sum_{j \text{ on } i} t_j$ be the *load* of machine i. Observe that we have the equality $\text{cost}(s) = \sum_j \ell_j = \sum_i p_i w_i$ for every feasible solution s. It will be convenient to denote $t = \sum_j t_j$ and $n = \sum_i p_i$ throughout. Recall that we normalize to $t_j \in [0, 1]$. Before considering the general case, we restrict ourselves to games with so-called clustered restrictions.

Clustered Restrictions. We speak of *clustered restrictions* in the game if the set A of allowed machines can is characterized as follows. Let J denote the set of tasks and let $J_1, \ldots, J_k$ be a disjoint partition of the tasks in non-empty sets. Let M denote the set of machines and let $M_1, \ldots, M_k$ be a disjoint partition of the machines in non-empty sets. Let $j \in J_i$ for some $i \in \{1, \ldots, k\}$ then, the set of allowed machines for task j is $A_j = M_i$. This means that j is allowed to use exactly those machines in the class M_i, but no others.

Theorem 1. *For clustered restrictions $A = \{A_1, A_2, \ldots, A_n\}$ with task-partition $J_1, \ldots, J_k$ and machine-partition $M_1, \ldots, M_k$ we have:*

(1) Define $\varepsilon_1 = \varepsilon_1(n, m, t) = 2nm/t^2$. The prices of stability and anarchy of the game are

$$\frac{n\sqrt{m}}{4t}(1 - o(1)) \leq \text{PoS}(\varSigma) \leq \text{PoA}(\varSigma) \leq \frac{n\sqrt{m}}{t} + \varepsilon_1. \qquad (2)$$

The lower bound holds for $t \geq m$; the upper for $t \geq 2$.

(2) Define $\varepsilon_2 = \varepsilon_2(n, m, \mathbb{E}[T]) = 2nm/\mathbb{E}[T]^2$. Suppose $T = \sum_j T_j$ with $\mathbb{E}[T] = \omega\left(\sqrt{n \log n}\right)$, where the T_j are independent. Then the expected prices of stabilty and anarchy of the game are

$$\frac{n\sqrt{m}}{4\mathbb{E}[T]}(1 - o(1)) \leq \text{EPoS}(\varSigma) \leq \text{EPoA}(\varSigma) \leq \left(\frac{n\sqrt{m}}{\mathbb{E}[T]} + \varepsilon_2\right)(1 + o(1)). \qquad (3)$$

The lower bound holds with the additional assumption that $\mathbb{E}[T] \geq m$.

For the proof of an upper bound notice that the clustered restrictions divide the problem into a set of unrestricted problems corresponding to the aforementioned partition into task sets $J_1, \ldots, J_k$ and the machine sets $M_1, \ldots, M_k$. Define $c_i = \sum_{j \in J_i} t_j$ as the *load* of a cluster. Further let $m_i = |M_i|$ and $n_i = |J_i|$. For the next lemma define the vectors $\boldsymbol{n} = (n_1, \ldots, n_k)$, $\boldsymbol{c} = (c_1, \ldots, c_k)$ and $\boldsymbol{m} = (m_1, \ldots, m_k)$. Furthermore, let $\mathcal{F}(n, t, m) \subset \mathbb{N}^k \times \mathbb{R}^k \times \mathbb{N}^k$ denote the subspace of feasible $(\boldsymbol{n}, \boldsymbol{c}, \boldsymbol{m})$, which simultaneously satisfy all the following constraints:

$$n_i \geq c_i \quad c_i > 0 \quad m_i \geq 1 \quad \textstyle\sum_i n_i = n \quad \sum_i c_i = t \quad \sum_i m_i = m.$$

Lemma 1. *Define the function* $f(\boldsymbol{n},\boldsymbol{c},\boldsymbol{m}) = (\sum_{i=1}^{k} \frac{n_i c_i}{m_i})/(\sum_{i=1}^{k} \frac{c_i^2}{m_i})$. *We have that* $f(\boldsymbol{n},\boldsymbol{c},\boldsymbol{m}) \le n\sqrt{m}/t$ *for* $(\boldsymbol{n},\boldsymbol{c},\boldsymbol{m}) \in \mathcal{F}(n,t,m)$.

Proof. For a geometric interpretation and intuition of the function f notice that for fixed n_i, the numerator is a hyperplane and the denominator is an elliptic paraboloid in the c_i. Therefore, f has a unique maximum, which can not be "very far" from the extremum of the elliptic paraboloid.

Without loss of generality, let $\frac{c_1}{m_1} \ge \frac{c_i}{m_i}$. Then for the numerator it is easy to see $\sum_{i=1}^{k} \frac{n_i c_i}{m_i} \le n\frac{c_1}{m_1}$. This gives $f(\boldsymbol{n},\boldsymbol{c},\boldsymbol{m}) \le (\frac{nc_1}{m_1})/(\sum_i \frac{c_i^2}{m_i})$. We strive to find the maximum value that this upper bound can attain. Hence, we try to maximize $f_1(\boldsymbol{c},\boldsymbol{m}) = (\frac{c_1}{m_1})/(\sum_i \frac{c_i^2}{m_i})$ subject to $\frac{c_1}{m_1} \ge \frac{c_i}{m_i}$, $c_i > 0$, $t = \sum_i c_i$, $m_i \ge 1$ and $m = \sum_i m_i$ for all $i \le k$.

How large can f_1 be? Let us fix values for m_1 and c_1. Then the denominator is minimized with the choice of $c_i = m_i(t-c_1)/(\sum_{\ell \ge 2} m_\ell)$ for the variables $c_2, \ldots, c_k$. Thus, we incorporate this assumption and get the remaining problem depending only on c_1 and m_1, which is to maximize $f_2(c_1,m_1) = (\frac{c_1}{m_1})/(\frac{c_1^2}{m_1} + \frac{(t-c_1)^2}{m-m_1})$ subject to $0 \le c_1 \le t$ and $1 \le m_1 < m$

Now assuming a fixed value for m_1, the best choice for c_1 is $c_1 = t\sqrt{\frac{m_1}{m}}$. Substitution and simplification reduces the problem to optimize only w.r.t. m_1, i.e. to maximize $f_3(m_1) = (\frac{1}{\sqrt{mm_1}})/(\frac{t}{m} + \frac{(1-\sqrt{m_1/m})^2 t}{m-m_1})$ subject to $1 \le m_1 < m$. It is a technical, but straightforward, exercise to show that for the first derivative $f_3'(m_1) \le 0$ for all $1 \le m_1 < m$. Hence, $f_3(m_1)$ is monotonic decreasing and the maximum obtained with $m_1 = 1$:

$$f_3(m_1) \le \frac{1/\sqrt{m}}{t/m + (1-\sqrt{1/m})^2 t/(m-1)} \le \frac{1/\sqrt{m}}{t/m} = \frac{\sqrt{m}}{t}.$$

We independently reduced the number of variables and finally derived $m_1 = 1$. A retrospective inspection shows that with our choices the constraints for $f_1(\boldsymbol{c},\boldsymbol{m})$ and $\frac{c_1}{m_1} \ge \frac{c_i}{m_i}$ are satisfied. Thus, the upper bound for f_3 results in an upper bound for f_1, and finally in $f(\boldsymbol{n},\boldsymbol{c},\boldsymbol{m}) \le n\sqrt{m}/t$. This proves the lemma. □

Finally, we need the following simple lemma, which is an adjustment from [10] to identical machines.

Lemma 2. *For every Nash equilibrium s for the selfish scheduling game without restrictions on identical machines* $\mathrm{cost}(s) \le n(t+2m)/m$. *For an optimum schedule* s^* *for such a game we have that* $\mathrm{cost}(s^*) \ge t^2/m$.

Proof (Proof of Theorem 1.). For the upper bound in (2) we may apply Lemma 2 to the unrestricted problems given by task sets $J_1, \ldots, J_k$ and the machine sets $M_1, \ldots, M_k$. With Lemma 1 we obtain

$$\frac{\text{cost}(s)}{\text{cost}(s^*)} \le \frac{\sum_{i=1}^{k} \frac{n_i(c_i+2m_i)}{m_i}}{\sum_{i=1}^{k} \frac{c_i^2}{m_i}} \le \frac{n\sqrt{m}}{t} + \frac{2n}{\sum_{i=1}^{k} \frac{c_i^2}{m_i}} \le \frac{n\sqrt{m}}{t} + \frac{2mn}{t^2}$$

To prove (3) we consider the probability that T deviates "much" from its expected value. Recall that $T = \sum_j T_j$ is a random variable. Let the random variables $S_0 = \mathbb{E}[T_1] + \cdots + \mathbb{E}[T_n]$ and $S_i = T_1 + \cdots + T_i + \mathbb{E}[T_{i+1}] + \cdots + \mathbb{E}[T_n]$ for $i = 1, \ldots, n$. The sequence $S_0, S_1, \ldots, S_n$ is a martingale, and differences are bounded by one: $|S_i - S_{i-1}| \le 1$. Therefore we may apply the Azuma-Hoeffding inequality: $\Pr[|S_n - S_0| \ge \lambda] \le 2\exp(-\lambda^2/2n)$. With the choice $\lambda = \sqrt{4n\log n}$ we have $\Pr\left[|T - \mathbb{E}[T]| \ge \sqrt{4n\log n}\right] \le 2/n^2$. Clearly $\text{PoA}(\Sigma) \le n$ always holds because each task is counted at least once but at most n times. With $\mathbb{E}[T] = \omega\left(\sqrt{n\log n}\right)$ we find

$$\begin{aligned}
\text{EPoA}(\Sigma) &\le \mathbb{E}\left[\min\left\{n, \frac{n\sqrt{m}}{T} + \frac{nm}{T^2}\right\}\right] \\
&\le \frac{n\sqrt{m}}{\mathbb{E}[T] - \sqrt{4n\log n}} + \frac{nm}{(\mathbb{E}[T] - \sqrt{4n\log n})^2} + n\frac{2}{n^2} \\
&= \left(\frac{n\sqrt{m}}{\mathbb{E}[T]} + \frac{nm}{\mathbb{E}[T]^2}\right)(1 + o(1)).
\end{aligned}$$

This proves the upper bounds. For the lower bounds we construct a deterministic task distribution and restrict the tasks to two sets of machines. We restrict the majority of tasks to a set of 2 machines, which creates a high price of stability similarly to the unrestricted case [10]. The remaining tasks on the remaining $m - 2$ machines are used to account for the total load, and their presence reduces the price of stability to essentially $\Theta(n\sqrt{m}/t)$. Details appear in the full version. □

General Restrictions. We continue with general restrictions, i.e., the sets $A_j \neq \emptyset$ are not constrained in any further way. Our main result states that the price of anarchy for general restrictions behaves similarly as for clustered restrictions.

Theorem 2. *Under the assumptions of Theorem 1, the bounds stated therein remain valid if* ε_1 *and* ε_2 *are replaced by* $\varepsilon_1 = \frac{2nm^2}{t^2}$ *and* $\varepsilon_2 = \frac{2nm^2}{\mathbb{E}[T]^2}$.

We relate the price of anarchy with clustered restrictions to general restrictions. This requires an additional concept, which is closely related to clusters. Thus we use similar notation.

Definition 1. *For a Nash equilibrium, label machines in order of their loads* $w_1 \geq w_2 \geq \cdots \geq w_m$*. A partition of the set of machines into* groups $M_1, \ldots, M_k$ *has the property that for every group* $M_i = \{r_{i-1}+1, \ldots, r_i\}$ *the loads of machines* $w_{r_i} - w_{r_{i+1}} > t_{max}$ *and* $w_\ell - w_{\ell+1} \leq t_{\max}$ *for all* $\ell \in \{r_{i-1}+1, \ldots, r_i - 1\}$.

We denote by J_i the set of tasks that are on any of the machines in M_i, and by $c_i = \sum_{\ell \in M_i} w_\ell = \sum_{j \in J_i} t_j$ the *load* of group M_i. Intuitively, the groups have the shape of stairs. The load difference between two consecutive steps is at most $t_{\max}$, but the step between two consecutive groups is more than $t_{\max}$ high. In the chosen Nash equilibrium every task in a group would like to switch to a group with lower load. The reason it does not do so must be that the restrictions forbid the change.

Now consider the machines with their optimum loads $w_1^*, \ldots, w_m^*$. Let $M_1, \ldots, M_k$ be the groups induced by any chosen Nash equilibrium. The above observation implies that also in the optimum solution no task on any of the machines M_i can be on any of the machines in $M_{i+1}, \ldots, M_k$, because the restrictions forbid it. However, it is possible that certain tasks of M_i change to the groups $M_1, \ldots, M_{i-1}$. The following lemma quantifies the effect of such changes.

Lemma 3. *Let* s^* *be an optimum schedule for an instance of the restricted selfish scheduling game with arbitrary restrictions. Let* s *be a Nash equilibrium that induces groups* $M_1, \ldots, M_k$ *with* $m_1, \ldots, m_k$ *machines and loads* $c_1, \ldots, c_k$*. Then* $\operatorname{cost}(s^*) \geq \sum_i c_i^2/m_i$*.*

Proof. Consider the optimum solution s^* with p_i^* players and w_i^* load on machine $i = 1, \ldots, m$. Group the machines into $M_1, \ldots, M_k$ as in the Nash equilibrium s. Define $c_i^* = \sum_{\ell \in M_i} w_\ell^*$ as the optimum load of the group M_i. Notice that $p_i^* \geq w_i^*$ because $t_j \leq 1$ for every task j. Clearly, the optimum cost of the group M_i is $\sum_{\ell \in M_i} p_\ell^* w_\ell^* \geq \sum_{\ell \in M_i} (w_\ell^*)^2 \geq (c_i^*)^2/m_i$. In order to prove the lower bound $\operatorname{cost}(s^*) \geq \sum_i c_i^2/m_i$, we transform the profile of the Nash equilibrium $c_1, \ldots, c_k$ into the optimum profile $c_1^*, \ldots, c_k^*$ without decreasing its cost. Let $x_1, \ldots, x_k$ denote the *current load*, which is initially $x_1 = c_1, \ldots, x_k = c_k$ and finally $x_1^* = c_1^*, \ldots, x_k^* = c_k^*$. We say that a group i is currently underloaded if $x_i < c_i^*$, overloaded if $x_i > c_i^*$, and saturated if $x_i = c_i^*$.

Observe that – by the restrictions – load is only allowed to move from a group M_ℓ with index ℓ to a group M_j with smaller index j. Hence, if there is an overloaded group (at all), then there must be an underloaded group with smaller index. Conversely, if there is an underloaded group (at all), there must be an overloaded group with larger index, due to the same reason. This property suggests an intuitive algorithm to transform the load profiles with the invariant that whenever there is an overloaded group, there is also an underloaded group with smaller index (and vice versa).

We repeatedly find the overloaded group with largest index (denoted ℓ) and the underloaded group with largest index (denoted j). Due to the invariant we know that $j < \ell$, i.e., there is an overloaded machine with larger index than any underloaded machine. We decrease x_ℓ and increase x_j by the same amount until one of the groups becomes saturated. This transformation preserves the invariant. The procedure eventually terminates, since we saturate at least one group in each iteration.

We determine the change in cost in one iteration as follows. Consider the initial situation, i.e., the Nash equilibrium with machine-loads $w_1 \geq \cdots \geq w_m$ and group-loads $c_1 \geq \cdots \geq c_k$. Let $w_i^{\min}$, respectively $w_i^{\max}$ denote the minimum, respectively maximum load of any machine in group M_i. Let $j < \ell$, note that $w_j^{\min} > w_\ell^{\max}$, and observe that

$$\frac{c_j}{m_j} \geq \frac{m_j w_j^{\min}}{m_j} = w_j^{\min} > w_\ell^{\max} = \frac{m_\ell w_\ell^{\max}}{m_\ell} \geq \frac{c_\ell}{m_\ell}.$$

Thus, initially, not only the group-loads c_i are in decreasing order, but also the relative loads c_i/m_i. Now consider a transformation in which load is moved from group M_ℓ to M_j. As every iteration increases x_j over c_j and decreases x_ℓ under c_ℓ, the inequality continues to hold for the values of x during the execution of our algorithm: $\frac{x_j}{m_j} \geq \frac{c_j}{m_j} > \frac{c_\ell}{m_\ell} \geq \frac{x_\ell}{m_\ell}$. Now suppose that M_j receives load $\delta > 0$ from M_ℓ. The change of the cost is $\delta^2(\frac{1}{m_j} + \frac{1}{m_\ell}) + 2\delta(\frac{x_j}{m_j} - \frac{x_\ell}{m_\ell})$. Since $\frac{x_j}{m_j} > \frac{x_\ell}{m_\ell}$, in every iteration our algorithm increases the cost. Hence, it transforms the Nash profile $c_1, \ldots, c_k$ into the optimum profile $c_1^*, \ldots, c_k^*$ without decreasing the cost. This yields $\mathrm{cost}(s^*) \geq \sum_i \frac{c_i^2}{m_i}$ and the proof is complete. □

For the proof of Theorem 2 we assemble the lower bound for s^* and a simple upper bound for any Nash equilibrium s. Then with a similar Azuma-Hoeffding argument as in the proof of Theorem 1 the result follows.

2.1 Average-Case Analysis of an Optimization Algorithm

In this short section, we point out that Theorem 2 also has an algorithmic perspective. By proving upper bounds on the expected price of anarchy of restricted selfish scheduling, we obtain an average-case analysis for an algorithm for the non-economical latency optimization problem (e.g. the standard scheduling variant of the game) as a byproduct. We consider the algorithm, which we call NASHIFY, due to Gairing et al. [7] introduced for maximum latency social cost. The algorithm begins with an arbitrary assignment and uses the idea of blocking flows to compute a Nash equilibrium. It has running time $O\left(nmA(\log t + m^2)\right)$ with $A = \sum_i |A_i|$. It is remarkable that the algorithm also performs well for total latency minimization for restricted scheduling, see Theorem 3 below. In the scheduling problem, the objective is to minimize $\sum_j \ell_j$, regardless if it is a Nash equilibrium or not. Let $\text{cost}(s)$ and $\text{cost}(s^*)$ denote the objective values of a schedule obtained by NASHIFY and by an (potentially exponential time) optimum algorithm OPT. While $\text{cost}(s)/\text{cost}(s^*)$ is called the *performance ratio*, for random task-lengths T_j the expectation $\mathbb{E}\left[\text{cost}(S)/\text{cost}(S^*)\right]$ is called the *expected performance ratio* of the algorithm NASHIFY. Here S and S^* are the associated random variables of s and s^*. The result below follows directly from [7] and Theorem 2.

Theorem 3. *Under the assumptions of Theorem 2, the bounds stated therein are upper bounds for the expected performance ratio of the algorithm* NASHIFY.

3 Maximum Latency Cost

Here we consider the social cost function $\text{cost}(s) = \ell_{\max} = \max_j \ell_j$. Define the parameter $r = \text{cost}(s^*)/t_{\max}$ of the game. Awerbuch et al. [2] showed a bound of $\Theta\left(\log m/(r\log(1+\frac{\log m}{r}))\right)$ on the price of anarchy. This gives $1+\epsilon$ price of anarchy for $r = \Omega(\log m/\epsilon^2)$. We contribute the following alternative bound, where the total traffic t is the parameter.

Theorem 4. *For our game with general restrictions and maximum latency social cost, we have the following:*

(1) For every $t = \sum_j t_j > 0$*, it holds that* $\text{PoA}(\Sigma) \le 1 + \frac{m^2}{t}$.
(2) For $T = \sum_j T_j$ *with* $\mathbb{E}\left[T\right] = \omega\left(\sqrt{n\log n}\right)$ *and independent* T_j *we have* $\text{EPoA}(\Sigma) \le 1 + \frac{m^2}{\mathbb{E}[T]}(1 + o(1))$.

Both bounds also hold for the (expected) performance of the algorithm NASHIFY.

References

1. Baruch Awerbuch, Yossi Azar, and Amir Epstein. The price of routing unsplittable flow. *Proc 37th STOC*, pages 57 – 66, 2005.
2. Baruch Awerbuch, Yossi Azar, Yossi Richter, and Dekel Tsur. Tradeoffs in worst-case equilibria. *Proc 1st WAOA*, pages 41 – 52, 2003.
3. Ioannis Caragiannis, Michele Flammini, Christos Kaklamanis, Panagiotis Kanellopoulos, and Luca Moscardelli. Tight bounds for selfish and greedy load balancing. *Proc 33rd ICALP*, pages 311–322, 2006.
4. George Christodoulou and Elias Koutsoupias. The price of anarchy of finite congestion games. *Proc 37th STOC*, pages 67 – 73, 2005.
5. Artur Czumaj, Piotr Krysta, and Berthold Vöcking. Selfish traffic allocation for server farms. *Proc 34th STOC*, pages 287 – 296, 2002.
6. Artur Czumaj and Berthold Vöcking. Tight bounds for worst-case equilibria. *Proc 13th SODA*, pages 413 – 420, 2002.
7. Martin Gairing, Thomas Lücking, Marios Mavronicolas, and Burkhard Monien. Computing nash equilibria for scheduling on restricted parallel links. *Proc 36th STOC*, pages 613 – 622, 2004.
8. Martin Gairing, Thomas Lücking, Marios Mavronicolas, Burkhard Monien, and Manuel Rode. Nash equilibria in discrete routing games with convex latency functions. *Proc 31st ICALP*, pages 645 – 657, 2004.
9. Martin Gairing, Thomas Lücking, Marios Mavronicolas, Burkhard Monien, and Paul Spirakis. Structure and complexity of extreme Nash equilibria. *Theoretical Computer Science*, 343(1-2):133 – 157, 2005.
10. Martin Hoefer and Alexander Souza. Tradeoffs and average-case equilibria in selfish routing. *Proc 15th ESA*, pages 63–74, 2007.
11. Stavros Kolliopoulos and Clifford Stein. Approximation algorithms for single-source unsplittable flow. *SIAM Journal on Computing*, 31:919–946, 2002.
12. Spyros Kontogiannis and Paul Spirakis. Atomic selfish routing in networks: A survey. *Proc 1st WINE*, pages 989 – 1002, 2005.
13. Elias Koutsoupias and Christos Papadimitriou. Worst-case equilibria. *Proc 17th STACS*, pages 404 – 413, 1999.
14. Jan Karel Lenstra, David Shmoys, and Eva Tardos. Approximation algorithms for scheduling unrelated parallel machines. *Mathematical Programming*, 46:259–271, 1990.
15. Marios Mavronicolas, Panagiota Panagopoulou, and Paul Spirakis. A cost mechanism for fair pricing of resource usage. *Proc 1st WINE*, pages 210–224, 2005.
16. Marc Scharbrodt, Thomas Schickinger, and Angelika Steger. A new average case analysis for completion time scheduling. *Proc 34th STOC*, pages 170 – 178, 2002.
17. Alexander Souza and Angelika Steger. The expected competitive ratio for weighted completion time scheduling. In *Proc 21st STACS*, pages 620 – 631. Springer Verlag, 2004.
18. Subhash Suri, Csaba Toth, and Yunhong Zhou. Selfish load balancing and atomic congestion games. *Algorithmica*, 47(1):79–96, 2007.