Adv Syst Sci Appl 2017; **4**:61–77
Published online at http://ijassa.ipu.ru/ojs/ijassa/article/view/518

Price of Anarchy for Maximizing the Minimum Machine Load

Yulia. V. Chirkova

Institute of Applied Mathematical Research, Karelian Research Centre RAS, Petrozavodsk, Russia

Abstract: The maximizing the minimum machine delay game (or cover game) with uniformly related machines is considered. Players choose machines with different speeds to run their jobs trying to minimize job's delay, i.e. the chosen machine's completion time. The social payoff is the minimal delay over all machines. For the general case of N machines we found the lower bound for Price of Anarchy (PoA), and for the case of 3 machines we found its exact value. We proved that the PoA does not change or increases when an additional third machine is included into the system with two machines. Also we propose a method of computation the PoA value and illustrate it for 3 machines.

Keywords: Nash equilibrium, maximizing the minimum load, cover game, price of anarchy

1. INTRODUCTION

Load balancing represents a major problem in networks and distributed computing systems, since load optimization guarantees efficient resource utilization. Modern systems such as telecommunication networks, cloud computing systems, GRID, etc. consist of independent components, in many cases without their centralized control. Particularly, users located at nodes and data transmission protocols do not interact with each other for maintaining a certain load level. Furthermore, in practice they demonstrate egoistic behavior with respect to free resources. And global optimization methods often become inapplicable due to the infeasibility of realizing optimal resource utilization plans in such systems (server request schedules, capacity norms of data transmission channels and so on). The game-theoretic approach allows treating load balancing as a game, where players have egoistic behavior and can reach some equilibrium state such that none of them benefits from unilateral deviation from a chosen strategy. System efficiency is assessed by comparing the above equilibria with the global optimum.

The present paper focuses on the maximizing the minimum machine delay game (or cover game) [1–3] also known as the scheduling problem [4] in the form of a game equivalent to the KP-model (see [5, 6]) with parallel different-capacity channels where system optimization is the maximizing the minimum machine delay [1–3] instead of the minimization the maximum machine delay (makespan). It is necessary to distribute several jobs of various volumes among machines of nonidentical speeds. The volume of a job is its completion time on a free unit-speed machine. Machine load is the total volume of jobs executed by a given machine. The ratio of machine load and speed defines its delay, i.e., the job completion time at this machine. Each player chooses a machine for its job striving to minimize job's delay. Players have egoistic behavior and reach a Nash equilibrium, *viz.*, a job distribution such that none of them benefits from unilateral change of a chosen machine. In the sequel, we study pure strategies Nash equilibria only; as is well-known [7, 8], such an equilibrium always exists

*Corresponding author: julia@krc.karelia.ru

in the described class of games. The system payoff (also called the social payoff) is the minimum delay over all machines for an obtained job distribution. The price of anarchy [1] (PoA) is defined as the maximum ratio of the optimal social payoff and the social payoff in the worst-case Nash equilibrium.

The problem where the system tries to maximize the minimum delay over all machines appears from the concept of a fair resource sharing and efficient routing of traffic. The paper [1] first in the equilibrium efficiency studying for such model gives motivations coming from issues of Quality of Service, fair resource allocation, and fair queuing. The base idea is that each system component must be loaded as much as possible and not to idle. Consider an example where each player pays the system a value which equals his delay for his job processing. Fair system should not have privileged players who pay rather less than others due to successful machine choose. Also such system should not have machines providing small or zero payoff.

According to the earlier publications, the PoA in the maximizing the minimum machine delay games with pure strategies can be estimated by

- for $N \geq 2$ machines with speeds $1 \leq \cdots \leq s$ [1] the price of anarchy is not limited if $s \geq 2$;
- the price of anarchy is closed to and no more than 1.7 for any number of homogeneous machines [1, 9];
- the price of anarchy equals
$$\begin{cases} \frac{2+s}{(1+s)(2-s)} & \text{for } 1 \leq s \leq \sqrt{2}, \\ \frac{2}{s(2-s)} & \text{for } \sqrt{2} < s < 2 \end{cases}$$
for two machines with speeds $1 \leq s$ [2];
- the price of anarchy equals
$$\frac{2+s}{2(2-s)} \text{ for } 1 \leq s < 2$$
for three machines with speeds $1 = 1 \leq s$ [2];
- the price of anarchy equals
$$\begin{cases} \frac{1+s}{s} & \text{for } 1 \leq s \leq s_0, \\ \frac{2+s}{(1+s)(2-s)} & \text{for } s_0 < s \leq \sqrt{2}, \\ \frac{2}{s(2-s)} & \text{for } \sqrt{2} < s < 2 \end{cases}$$
in the hierarchical model of two machines with speeds $1 \leq s$ and two types of jobs where first machine can process both types jobs and second machine can process only second type jobs [3]. Here s_0 is the largest root of the equation $\frac{1+s}{s} = \frac{2+s}{(1+s)(2-s)}$.

In what follows, we derive a lower estimate for the PoA in the case of $N \geq 3$ machines. Also we present the exact value of the PoA for 3 machines with speeds $1 \leq r \leq s < 2$:

$$\begin{cases} \frac{2+s}{(1+r)(2-s)} & \text{for } rs \leq 2, \\ \frac{2}{r(2-s)} & \text{for } rs > 2. \end{cases}$$

Moreover we show that the PoA increases or does not change under new machine inclusion into the system of two machines. In the case of N machines, a computing algorithm of the exact PoA value is proposed based on solving a series of linear programming problems. The algorithm is described for the case of 3 machines and is implemented numerically in the form of a program which draws the curves of the PoA as a function of the fastest machine and compares them with the curves of the corresponding estimates.

2. THE MODEL

Consider a system $S = S(N, v)$ composed of N machines operating with speeds $v_1 = 1 \leq \dots \leq v_N = s$. Note that such choose of machine speeds does not contradict with generality since one always can normalize speeds dividing them by the speed of the slowest machine. The system is used by a set of players $U = U(n, w)$: each of n players chooses an appropriate machine for its job execution. For player j, the volume of job equals w_j, $j = 1, \dots, n$. Denote by $W = \sum_{j=1}^{n} w_j$ the total volume of all jobs. Free machine i with speed v_i executes a job of volume w during the time w/v_i.

Study the following pure strategies game $\Gamma =< S(N, v), U(n, w), \lambda >$. Each player can choose any machine. The strategy of player j is machine l_j selected by this player for its job execution. Then the strategy profile in the game Γ represents the vector $L = (l_1, \dots, l_n)$. The load of machine i, i.e., the total volume of all jobs assigned to the machine is defined by $\delta_i(L) = \sum_{j=1,\dots,n: l_j = i} w_j$. The delay of machine i takes the form

$$\lambda_i(L) = \sum_{j=1,\dots,n: l_j = i} w_j / v_i = \frac{\delta_i(L)}{v_i}.$$

Actually, this quantity is the same for all players selecting a given machine.

We suppose that the goal of the system is minimizing of the least busy machine idling, that is maximizing of its working time or delay on it. The social payoff is described by the minimum delay over all machines:

$$SC(L) = \min_{i=1,\dots,N} \lambda_i(L).$$

Designate by

$$OPT = OPT(S, U) = \max_{L \text{ is a profile in } \Gamma(S,U,\lambda)} SC(L)$$

the optimal payoff (the social payoff in the optimal case) where maximization runs over all admissible strategy profiles in the game $\Gamma(S, U, \lambda)$.

A strategy profile L such that none player benefits from unilateral deviation (change of the machine chosen in L for its job execution) is a pure strategies Nash equilibrium. To provide a formal definition, let $L(j \to i) = (l_1, \dots, l_{j-1}, i, l_{j+1}, \dots, l_n)$ signify the profile obtained from a profile L if player j replaces machine l_j chosen by it in the profile L for another machine i, whereas the rest players use the same strategies as before (remain invariable).

Definition 2.1:
A strategy profile L is said to be a pure strategies Nash equilibrium iff each player chooses a machine with the minimum delay, i.e., for each player $j = 1, \dots, n$ we have the inequality $\lambda_{l_j}(L) \leq \lambda_i(L(j \to i))$ for all machines $i = 1, \dots, N$.

Definition 2.2:
The price of anarchy in the system S is the maximum ratio of the social payoff in the optimal case and the social payoff in the worst-case Nash equilibrium:

$$PoA(S) = \max_U \frac{OPT(S, U)}{\min_{L \text{ is a Nash equilibrium in } \Gamma(S,U,\lambda)} SC(L)}.$$

3. THE GENERAL CASE OF N MACHINES

In this section we give the following assumptions and results which will be employed in further analysis.

Consider a system composed of $N \geq 2$ machines operating with speeds $v_1 = 1 \leq \cdots \leq v_N = s$. If the number of jobs n is less than the number of machines N then obviously the social payoff is zero in any profile. In this case we assume by definition that the ratio of an optimal payoff to an equilibrium payoff is 1. Further we suppose that $n \geq N$.

If $s \geq 2$ then the price of anarchy is infinite [1]. Therefore, we assume 1 ¡ s ¡ 2 in the following. If the number of jobs n is more than or equals the number of machines N then obviously all machines are loaded in an optimal profile. Moreover in this case all machines are loaded in any equilibrium.

The optimal social payoff is not larger than the social payoff in the case when the whole volume of jobs is distributed among machines proportionally to their speeds so that all machines have an identical delay:

$$OPT \leq \frac{W}{\sum\limits_{i=1}^{N} v_i}. \tag{3.1}$$

Further we determine estimates for equilibrium delays and volumes for some jobs processed on machines. We also restate the proof for results taken from cited papers for the sake of completeness.

Lemma 3.1:
[2] If the number of jobs n is not less than the number of machines N then in any equilibrium the load of any machine is more than zero.

Proof
Consider an arbitrary equilibrium profile L. Suppose that some machine i has zero load. Then there is a machine k receiving not more than two jobs. Since $v_1 = 1 \leq \cdots \leq v_N = s < 2$ then $v_i > \frac{v_k}{2}$. Let w_k be the minimal job volume on k. If it moves to an idle machine i then its load becomes equal $\frac{w_k}{v_i} < \frac{2w_k}{v_k} \leq \lambda_k(L)$, that is less comparing with its load in the profile L. □

Denote the number of jobs on some machine k in a profile L by n_k.

Lemma 3.2:
[2] Suppose that L is a Nash equilibrium profile and $SC(L) = \lambda_i(L)$. If $n_k > \frac{v_k}{v_i}$ then $\lambda_k(L) \leq \frac{n_k v_i}{n_k v_i - v_k}\lambda_i(L)$ for any machine k.

Proof
Let w be the job with the smallest volume on some machine k. Then $w \leq \frac{v_k}{n_k}\lambda_k(L)$. Since L is an equilibrium then $\lambda_k(L) \leq \lambda_i(L) + \frac{w}{v_i} \leq \lambda_i(L) + \frac{v_k}{n_k v_i}$ and thus $\lambda_k(L) \leq \frac{n_k v_i}{n_k v_i - v_k}\lambda_i(L)$. □

Lemma 3.3:
Suppose that L is an equilibrium profile and $SC(L) = \lambda_i(L)$ and consider an arbitrary machine k. If $n_k \geq 2$ and $1 \leq \frac{v_k}{v_i} < 2$ then the volume w_j of any job j on the machine k is at most $\frac{v_i v_k}{2v_i - v_k}\lambda_i(L)$. Moreover the total volume of remaining jobs on k is also no more than $\frac{v_i v_k}{2v_i - v_k}\lambda_i(L)$.

Proof
Let the machine k receive two or more jobs and w be the minimal job volume on k. Then

the total volume of remaining jobs on k equals $v_k\lambda_k(L) - w$. Since L is an equilibrium then $\lambda_k(L) = \frac{v_k\lambda_k(L)}{v_k} \leq \lambda_i(L) + \frac{w}{v_i}$ and thus $v_k\lambda_k(L) - w \leq v_k\lambda_i(L) + \left(\frac{v_k}{v_i} - 1\right) w \leq v_k\lambda_i(L) + \left(\frac{v_k}{v_i} - 1\right) w_{j:l_j=k} \leq v_k\lambda_i(L) + \left(\frac{v_k}{v_i} - 1\right)(v_k\lambda_k(L) - w)$. Then $w \leq w_{j:l_j=k} \leq v_k\lambda_k(L) - w \leq \frac{v_i v_k}{2v_i - v_k}\lambda_i(L)$. □

The next theorem determines the lower estimate for the price of anarchy in the system of $N \geq 3$ machines. The estimate is determined by speeds of 3 machines in the system: the first one and the second one which are the slowest, and the last one which is the fastest.

Theorem 3.1:
For the system composed of $N \geq 3$ machines having speeds $v_1 = 1 \leq v_2 = r \leq v_3 \leq \cdots \leq v_N = s < 2$ the price of anarchy is at least

$$est(r, s) = \min\{\frac{2+s}{(1+r)(2-s)}, \frac{2}{r(2-s)}\}. \tag{3.2}$$

Proof
As far as we need to prove the lower estimate it suffices to present examples of systems providing ratios of the optimal payoff and the worst-case equilibrium payoff given in the theorem condition. Suppose that in the system each machine i has a speed v_i for each $i = 1, \ldots, N$.

1. Let first $rs \leq 2$. Then $est(r, s) = \frac{2+s}{(1+r)(2-s)}$. Consider the set of jobs: $w_1 = w_2 = (1+r)s$, $w_3^i = v_i(2+s)$, where $i = 3\ldots, N$, $w_4 = 2r - s$, $w_5 = 2 - rs$. In the equilibrium L machine N receives jobs w_1 and w_2, each machine $i-1$ receives each job w_3^i, $i = 3\ldots, N$, jobs w_4 and w_5 are assigned to machine 1. We need to show that L is really an equilibrium and find the system payoff.

The loads of machines N and 1 equal $2s(1+r)$ and $(1+r)(2-s)$ respectively. The load of each machine $i = 2, \ldots, N-1$ equals $v_{i+1}(2+s)$. Since $\lambda_N(L) = 2(1+r) > (1+r)(2-s) = \lambda_1(L)$ and $\lambda_i(L) = \frac{v_{i+1}(2+s)}{v_i} \geq (1+r)(2-s) = \lambda_1(L)$, $i = 2, \ldots, N-1$, due to $2+s > 1+r$, $v_{i+1} \geq v_i$ and $2-s \leq 1$, then machine 1 has the smallest delay which equals its load.

Denote the delay of machine i as $\lambda_i^j(L) = \lambda_i(L) + \frac{w_j}{v_i}$ in the case where some job j deviates from the profile L and moves to machine i from another machine. No one of jobs w_1 or w_2 moves to machine i, $i = 2, \ldots, N-1$, since $\lambda_N(L) = 2(1+r) \leq (2+s) + (1+r) \leq \frac{v_{i+1}(2+s)+s(1+r)}{v_i} = \lambda_i^1(L) = \lambda_i^2(L)$. Also no one of them moves to machine 1, since $\lambda_N(L) = 2(1+r) = (1+r)(2-s) + s(1+r) = \lambda_1^1(L) = \lambda_1^2(L)$. Each of jobs w_3^i, $i = 3, \ldots, N$, has not reason to move to machine N due to $\lambda_{i-1}(L) = \frac{v_i(2+s)}{v_{i-1}} \leq 2(1+r) + \frac{v_i(2+s)}{s} = \lambda_N^{i3}(L)$, that is equivalent to the inequality $(s - v_{i-1})v_i(2+s) \leq 2sv_{i-1}(1+r)$ which holds true since $s - v_{i-1} < 1$, $2+s < 4$ $2\frac{s}{v_i}v_{i-1}(1+r) \geq 4$. Also no one of jobs w_3^i, $i = 3, \ldots, N$, moves to machine $j > i-1$ since $\lambda_{i-1}(L) = \frac{v_i(2+s)}{v_{i-1}} < \frac{2v_i(2+s)}{v_j} \leq \frac{(v_i+v_j)(2+s)}{v_j} = \lambda_j^{i3}(L)$. Moreover, job w_3^i does not move to slower machine 1 or $j < i-1$, and no one job from machine 1 moves to another machine because the delay on machine 1 is minimal. Therefore, the given profile is an equilibrium with the social payoff $(1+r)(2-s)$.

Consider the profile where each job w_3^i, $i = 3, \ldots, N$ belongs to machine i, jobs w_1 and w_4 are assigned to machine 2 and machine 1 receives jobs w_2 and w_5. The social payoff equals $2+s$ in this profile, so, $OPT \geq 2+s$.

2. Let now $rs > 2$. Then $est(r, s) = \frac{2}{r(2-s)}$. Consider the set of jobs: $w_1 = w_2 = rs$, $w_3^i = 2v_i$, $i = 3, \ldots, N$, $w_4 = r(2-s)$. In the equilibrium L jobs w_1 and w_2 belong to

machine N, each job w_3^i, $i = 3, \dots, N$ is assigned to machine $i-1$, machine 1 receives job w_4. We show that it is an equilibrium indeed and find the system payoff.

Since $\lambda_N(L) = 2r > r(2-s) = \lambda_1(L)$ and $\lambda_i(L) = \frac{2v_{i+1}}{v_i} \geq r(2-s) = \lambda_1(L)$, $i = 2, \dots, N-1$, due to $\frac{2}{v_i} \geq 1$, $v_{i+1} \geq r$ and $2 - s < 1$, then machine 1 has the smallest delay which equals $r(2-s)$. Job w_1 or w_2 does not move to machine i, $i = 2, \dots, N-1$, since $\lambda_N(L) = 2r = r + r \leq \frac{2v_{i+1}+rs}{v_i} = \lambda_i^1(L) = \lambda_i^2(L)$, and also to machine 1 due to $\lambda_N(L) = 2r = r(2-s) + rs = \lambda_1^1(L) = \lambda_1^2(L)$. No one of jobs w_3^i, $i = 3, \dots, N$, moves to machine N since $\lambda_{i-1}(L) = \frac{2v_i}{v_{i-1}} \leq 2r + \frac{2v_i}{s} = \lambda_N^{i3}(L)$ due to $\frac{2v_i(s-v_{i-1})}{s} \leq 2rv_i$. Also no one of jobs w_3^i, $i = 3, \dots, N$, moves to machine $j > i-1$, since $\lambda_{i-1}(L) = \frac{2v_i}{v_{i-1}} \leq \frac{4v_i}{v_j} \leq \frac{2(v_i+v_j)}{v_j} = \lambda_j^{i3}(L)$. Moreover, job w_3^i does not move to slower machine 1 or $j < i-1$. No one job on machine 1 moves to other machines with not smaller delay. Hence, the given profile L is an equilibrium with the social payoff $r(2-s)$.

Consider the profile where each job w_3^i, $i = 3, \dots, N$, is assigned to machine i, jobs w_1 and w_4 belong to machine 2, and job w_2 to machine 1. The social payoff equals 2 for this profile, thus, $OPT \geq 2$.

In both considered cases the ratio of the optimal payoff and the equilibrium payoff equal $est(r, s)$, hence, the price of anarchy is not less than this estimate. □

From the obtained estimate (3.2) we see that when the speed of the fastest machine increases and comes closer to the value of 2, the lower estimate of the price of anarchy grows infinitely. Hence, we obtain the following corollary from the theorem 3.1.

Corollary 3.1:
For the system composed of $N \geq 3$ machines having speeds $v_1 = 1 \leq v_2 = r \leq v_3 \leq \dots \leq v_N = s < 2$ the price of anarchy tends to infinity as $s \to 2 - 0$.

According to the following result, for PoA evaluation it suffices to consider only games, where the optimal social payoff equals 1.

Theorem 3.2:
For the system S, the price of anarchy constitutes

$$PoA(S) = \max_{U_1 : OPT(S, U_1) = 1} \frac{1}{\min\limits_{L \text{ is a Nash equilibrium in } \Gamma(S, U_1, \lambda)} SC(L)}.$$

Proof
We show that one can normalize job volumes in any game $\Gamma(S, U, \lambda)$ such that the optimal social payoff becomes equal 1 and the ratio of the optimal social payoff and the worst-case equilibrium payoff does not change its value.

Assume that L is the worst-case equilibrium in the game $\Gamma(S, U, \lambda)$ with an arbitrary set of players $U(n, w)$. For each player j, the volume of its job equals w_j, and the vector L_{OPT} gives the optimal strategy profile in this game. Let SC and OPT be the social payoff in the profile L and the optimal social payoff, respectively. The ratio of the optimal and worst-case equilibrium social payoff is defined by $\frac{OPT}{SC}$. So long as L represents an equilibrium, then for any player j we obtain that $\frac{\sum\limits_{k=1,\dots,n: l_k = l_j} w_k}{v_{l_j}} \leq \frac{\sum\limits_{k=1,\dots,n: l_k = i} w_k + w_j}{v_i}$ for any machine i.

Now, explore the game with the same set of machines and players, where each player j has the job of volume $\frac{w_j}{OPT}$. The social payoff in the profiles L and L_{OPT} constitutes $\frac{SC}{OPT}$ and 1, respectively. By virtue of the linear homogeneity of machine delays in their loads, the profiles L and L_{OPT} form the worst-case equilibrium and optimal profiles, respectively, in the new game. Particularly, the profile L is an equilibrium in the new game, since for any player

j the inequality $\frac{\sum\limits_{k=1,\dots,n:l_k=l_j} w_k}{v_{l_j} OPT} \le \frac{\sum\limits_{k=1,\dots,n:l_k=i} w_k + w_j}{v_i OPT}$ holds true for any machine i. Imagine that L is any non-worst-case equilibrium in the new game. Then the game admits an equilibrium L' with social payoff $\frac{SC'}{OPT}$ such that the social payoff in the profile L' is less than that in the profile L, i.e., $\frac{SC'}{OPT} < \frac{SC}{OPT}$. However, in the initial game the profile L' corresponds to the social payoff $SC' < SC$, and the equilibrium L' is worse than its counterpart L. Similarly, L_{OPT} gives the optimal profile in the new game. Then the ratio of the optimal and the worst-case equilibrium social payoff in the new game also equals $\frac{OPT}{SC}$.

Consequently, any game $\Gamma(S, U, \lambda)$ corresponds to a game $\Gamma(S, U_1, \lambda)$ with normalized job volumes such that $OPT(S, U_1) = 1$. Moreover, the ratio of the optimal and the worst-case equilibrium social payoff is same in both games. Hence, for PoA evaluation it suffices to consider only games with unit optimal social payoff. □

4. THE CASE OF 3 MACHINES

As a matter of fact, the exact PoA value in the two-machine model was found in the paper [2]. Consider the case of 3 machines in the system S. Without loss of generality, throughout this section we believe that the machines have speeds $v_1 = 1 \le v_2 = r \le v_3 = s$, i.e., machine 1 is the slowest one, machine 2 has medium speed and machine 3 is the fastest one.

Lemma 4.1:
For the system S composed of 3 machines having speeds $v_1 = 1 \le v_2 = r \le v_3 = s$ the inequality $OPT \le \frac{W - w_k}{1+r}$ holds true for any job k with the volume w_k.

Proof
Suppose that there is such job of the volume w_k assigned to machine i in the optimal profile L, that $OPT > \frac{W-w_k}{1+r}$. Then all optimal delays on machines exceed $\frac{W-w_k}{1+r}$. Moreover, it is clear that $\lambda_i(L) \ge \frac{w_k}{v_i}$. Hence, $W = v_i\lambda_i(L) + v_j\lambda_j(L) + v_l\lambda_l(L) > w_k + (v_j + v_l)\frac{W-w_k}{1+r} \ge w_k + (1+r)\frac{W-w_k}{1+r} = W$. □

Lemma 4.2:
For the system S composed of 3 machines having speeds $v_1 = 1 \le v_2 = r \le v_3 = s$, if two jobs of volumes w_{k_1} and w_{k_2} are assigned to the same machine in the optimal profile, then $OPT \le \frac{W - w_{k_1} - w_{k_2}}{1+r}$.

Proof
Assume for the sake of contradiction that $OPT > \frac{W-w_{k_1}-w_{k_2}}{1+r}$ and jobs of volumes w_{k_1} and w_{k_2} are assigned to machine i in the optimal profile. Then all optimal delays on machines exceed $\frac{W-w_{k_1}-w_{k_2}}{1+r}$ and $\lambda_i(L) \ge \frac{w_{k_1}+w_{k_2}}{v_i}$. Then $W = v_i\lambda_i(L) + v_j\lambda_j(L) + v_l\lambda_l(L) > w_{k_1} + w_{k_2} + (v_j + v_l)\frac{W-w_{k_1}-w_{k_2}}{1+r} \ge w_{k_1} + w_{k_2} + (1+r)\frac{W-w_{k_1}-w_{k_2}}{1+r} = W$. □

Theorem 4.1:
For the system S composed of 3 machines having speeds $v_1 = 1 \le v_2 = r \le v_3 = s < 2$ the price of anarchy does not exceed $est(r, s) = \min\{\frac{2+s}{(1+r)(2-s)}, \frac{2}{r(2-s)}\}$.

Proof
In the proof we consider cases with a certain number of jobs assigned to each of two the most loaded machines having the largest delay. For each case we show that the price of anarchy estimate presented in the theorem condition is true. Suppose that L is an equilibrium profile

and $SC(L) = \lambda_i(L)$, that is machine i has the smallest delay. Explore different cases of an equilibrium L.

1. Each of machines j and l receives one job. In the optimal profile these two jobs occupy at most two machines. Thus, there is some machine k in the optimal profile, taking partially or wholly the equilibrium load of machine i and nothing else. That is $OPT \le \frac{v_i\lambda_i(L)}{v_k} \le s\lambda_i(L)$. By lemma 8.1 $s \le est(r,s)$.

2. Machine j receives $n_j \ge 2$ jobs, machine l receives $n_l = 1$ job. By lemma 3.2 $\lambda_j(L) \le \frac{2v_i}{2v_i - v_j}\lambda_i(L)$. By lemma 4.1 $OPT \le \frac{v_i\lambda_i(L) + \frac{2v_iv_j}{2v_i-v_j}\lambda_i(L)}{1+r} = \lambda_i(L)\frac{2v_i^2+v_iv_j}{(1+r)(2v_i-v_j)}$.

a) Assume first that $v_i \ge v_j$. Then $2v_i^2 + v_iv_j \le 3v_i^2$, since this expression increases by v_j. Also $2v_i - v_j \ge v_i$, so long as this expression decreases by v_j. Then $OPT \le \lambda_i(L)\frac{3v_i}{1+r} \le \lambda_i(L)\frac{3s}{1+r} \le \lambda_i(L)est(r,s)$ by lemma 8.2.

b) Suppose now that $v_i < v_j$. Then by lemma 8.3 $\frac{2v_i^2+v_iv_j}{(1+r)(2v_i-v_j)} < \frac{2+s}{(1+r)(2-s)}$.

Explore now two cases. In the first case assume that $v_i = r$ and $v_j = s$. Here by lemma 8.4 $\frac{2r^2+rs}{(1+r)(2r-s)} < \frac{2}{r(2-s)}$.

In the second case $v_i = 1$. By lemma 3.3 $w_k \le \frac{v_k}{2-v_k}\lambda_i(L) \le \frac{s}{2-s}\lambda_i(L)$ and $v_j\lambda_j(L) - w_k \le \frac{s}{2-s}\lambda_i(L)$ for any job of volume w_k assigned to machine j.

If all jobs assigned to machine j in the profile L remain there in the optimal profile, two cases are possible. If a single job assigned to machine l in the equilibrium keeps its position in the optimal profile, then the load of machine i can only decrease with system's transition from the equilibrium to the optimal profile. If this single job leaves machine l, then in the optimal profile machine l receives the load at most $\lambda_i(L)$ coming from machine i. In both cases $OPT \le \lambda_i(L)$.

If jobs move from machine j only to machine l with system's transition from the equilibrium to the optimal profile, similar two cases are possible. If a single job assigned to machine l in the equilibrium remains there in the optimal profile, then the load of machine i can only decrease with transition to the optimal profile. Then $OPT \le \lambda_i(L)$. If this single job leaves machine l, then in the optimal profile machine l can receive the load at most $\lambda_i(L) + \frac{s}{2-s}\lambda_i(L)$ coming from machines i and j. Then $OPT \le \lambda_i(L)\frac{1+\frac{s}{2-s}}{v_l} \le \lambda_i(L)\frac{1+\frac{s}{2-s}}{r} = \lambda_i(L)\frac{2}{r(2-s)}$.

If some jobs move from machine j to machine i with system's transition from the equilibrium to the optimal profile, we obtain the same two cases. If a single job assigned to machine l in the equilibrium remains there in the optimal profile, then machine j receives the load at most $\lambda_i(L) + \frac{s}{2-s}\lambda_i(L)$ consisting from the load remaining on j and possibly coming from i. Then $OPT \le \lambda_i(L)\frac{1+\frac{s}{2-s}}{v_j} \le \lambda_i(L)\frac{1+\frac{s}{2-s}}{r} = \lambda_i(L)\frac{2}{r(2-s)}$. If this single job leaves machine l, then in the optimal profile machine l can receive the load at most $\lambda_i(L) + \frac{s}{2-s}\lambda_i(L)$ coming from machines i and j.

3. Each of machines j and l receives exactly two jobs: $n_j = n_l = 2$. The total number of jobs assigned to machines j and l is four, the number of machines is three, so in the optimal profile at least two of these jobs (w_{k_1} and w_{k_2}) become assigned to the same machine. Then by lemma 4.2 $OPT \le \frac{W - w_{k_1} - w_{k_2}}{1+r} = \frac{v_i\lambda_i(L) + w_{k_3} + w_{k_4}}{1+r}$, where w_{k_3} w_{k_4} remaining two jobs assigned to machines j and l.

Consider machine $k \in \{j,l\}$. If $v_i \le v_k$, then by lemma 3.3 the volume of any of jobs assigned to machine k does not exceed $\lambda_i(L)\frac{v_iv_k}{2v_i-v_k}$.

Let now $v_i > v_k$. L is an equilibrium, therefore $\lambda_k(L) \le \lambda_i(L) + \frac{w}{v_i}$, where w is the smallest volume job assigned to machine k. Thus, $w \ge v_i\lambda_k(L) - v_i\lambda_i(L)$. By lemma 3.2

$\lambda_k(L) \le \lambda_i(L)\frac{2v_i}{2v_i-v_k} \le 2\lambda_i(L), \quad 2v_i - v_k \ge v_i, \quad w \le \lambda_i(L)$. Another job of the largest volume assigned to machine k has the volume $v_k\lambda_k(L) - w \le v_k\lambda_k(L) = v_i\lambda_i(L) - (v_i - v_k)\lambda_k(L) \le v_i\lambda_i(L) - (v_i - v_k)\lambda_i(L) = v_k\lambda_i(L) \le r\lambda_i(L) \le \lambda_i(L)\frac{rs}{2r-s} \le \lambda_i(L)\frac{s}{2-s}$.

a) Let $v_i = s$, then $OPT \le \lambda_i(L)\frac{s+2r}{1+r} \le \lambda_l(L)\frac{3s}{1+r} \le \lambda_i(L)est(r,s)$ by lemma 8.2.

b) Let $v_i = r$, then $OPT \le \lambda_i(L)\frac{r+2\frac{rs}{2r-s}}{1+r} = \lambda_i(L)\frac{2r^2+rs}{(1+r)(2r-s)} \le \lambda_i(L)est(r,s)$ by lemma 8.3.

c) Let $v_i = 1$, then $OPT \le \lambda_i(L)\frac{r+2\frac{s}{2-s}}{1+r} = \lambda_i(L)\frac{2+s}{(1+r)(2-s)}$. From the other side so long as the number of machines equals three there are surely two machines α and β receiving at most one job from considered four jobs and, perhaps, some part of the load of machine i. Then OPT does not exceed the minimal delay over these machines: $OPT \le \min\limits_{\alpha\ne\beta}\{\lambda_i(L)\frac{2}{v_\alpha(2-s)}, \lambda_i(L)\frac{2}{v_\beta(2-s)}\} \le \min\{\lambda_i(L)\frac{2}{1(2-s)}, \lambda_i(L)\frac{2}{r(2-s)}\} = \lambda_i(L)\frac{2}{r(2-s)}$.

4. Machines j and l receive the following job allocation: $n_j \ge 2$, $n_l \ge 3$. By lemma 3.2 $\lambda_j(L) \le \lambda_i(L)\frac{2v_i}{2v_i-v_j}$ and $\lambda_l(L) \le \lambda_i(L)\frac{3v_i}{3v_i-v_l}$. Thus in accordance with the estimate (3.1), $OPT \le \lambda_i(L)\frac{v_i+\frac{2v_iv_j}{2v_i-v_j}+\frac{3v_iv_l}{3v_i-v_l}}{1+r+s} \le \lambda_i(L)est(r,s)$ by lemma 8.5 and lemma 8.6. □

The next theorem is a special case of theorem 3.1 for the system of three machines.

Theorem 4.2:
For the system S composed of 3 machines having speeds $v_1 = 1 \le v_2 = r \le v_3 = s < 2$ is at least $est(r,s) = \min\{\frac{2+s}{(1+r)(2-s)}, \frac{2}{r(2-s)}\}$.

Then we obtain from theorems 4.1 and 4.2 an exact value of the price of anarchy for the tree-machine system.

Theorem 4.3:
For the system S composed of 3 machines having speeds $v_1 = 1 \le v_2 = r \le v_3 = s < 2$ the price of anarchy exactly equals

$$\begin{cases} \frac{2+s}{(1+r)(2-s)} & \textit{for } rs \le 2, \\ \frac{2}{r(2-s)} & \textit{for } rs > 2. \end{cases}$$

The exact PoA value allows establishing a possibility for PoA increase under new machine inclusion into the system, i.e., in a situation resembling the Braess paradox [10–13] when system's power increasing leads to its performance characteristic degradation. The next statement illustrates that the price of anarchy increases or does not change under new machine inclusion into the system of two machines.

Theorem 4.4:
For the system S composed of two machines having speeds $1 \le s$ the price of anarchy does not decrease with adding a new machine of speed $1 \le q < 2$.

Proof
1. Suppose that new machine has a speed of $q \le s$. If $qs \le s^2 < 2$ then the price of anarchy does not decrease since $\frac{2+s}{(1+s)(2-s)} \le \frac{2+s}{(1+q)(2-s)}$. If $s^2 > 2$ and $qs \le 2$ then that does not decrease due to $\frac{2}{s(2-s)} \le \frac{2+s}{(1+s)(2-s)} \le \frac{2+s}{(1+q)(2-s)}$. Consequently, we have the same if $s^2 > 2$ and $qs > 2$ since $\frac{2}{s(2-s)} \le \frac{2}{q(2-s)}$.

2. Suppose now that new machine is more powerful than existing in the system, $s < q < 2$. If $qs \le 2$, then $s^2 \le 2$, and the price of anarchy does not decrease since $\frac{2+s}{(1+s)(2-s)} \le \frac{2+q}{(1+s)(2-q)}$. If $qs > 2$ $s^2 \le 2$, then that does not decrease due to $\frac{2+s}{(1+s)(2-s)} \le \frac{2}{s(2-s)} \le \frac{2}{s(2-q)}$. Also if $qs > 2$ and $s^2 > 2$, we obtain the same because of $\frac{2}{s(2-s)} \le \frac{2}{s(2-q)}$. □

5. EVALUATING THE PRICE OF ANARCHY

In the previous section, we have derived an analytic expression for the price of anarchy in the three-machine model, where the fastest machine possesses a rather high speed. In what follows, we suggest a computing method for the price of anarchy on the example of the system of 3 machines which is similar to a corresponding method for the load balance game [14]. This method can be generalized to systems composed of more machines. But such generalization increases the number of linear programming problems to-be-solved and the number of associated variables and imposed constraints. Particularly the N-machine model requires $N!$ linear programming problems each of which includes $(2^N - 1)^{N-1}$ subproblems with N^2 variables.

Consider the following system of linear equations in the components of the vectors $a = (a_1, a_2, a_3), b = (b_1, b_2, b_3), c = (c_1, c_2, c_3)$.

$$\begin{cases} \frac{a_1+a_2+a_3}{v_i} \leq \frac{b_1+b_2+b_3+\min\limits_{k=1,2,3:a_k>0} a_k}{v_j} \\ \frac{a_1+a_2+a_3}{v_i} \leq \frac{c_1+c_2+c_3+\min\limits_{k=1,2,3:a_k>0} a_k}{v_l} \\ \frac{b_1+b_2+b_3}{v_j} \leq \frac{c_1+c_2+c_3+\min\limits_{k=1,2,3:b_k>0} b_k}{v_l} \\ \frac{a_1+a_2+a_3}{v_i} \geq \frac{b_1+b_2+b_3}{v_j} \geq \frac{c_1+c_2+c_3}{v_l} \\ \max\limits_{k=1,2,3} a_k > 0 \\ \max\limits_{k=1,2,3} b_k > 0 \\ a_k, b_k, c_k \geq 0, k = 1, 2, 3. \end{cases} \tag{5.3}$$

This system describes a set of hyperplanes passing through the point $(0, 0, 0, 0, 0, 0, 0, 0, 0)$ in the 9-dimensional space, and the solution set represents a domain in the space bounded by the hyperplanes. The above system is feasible, as far as, e.g., the triplet $a_1 = a_2 = a_3 = \alpha s_i$, $b_1 = b_2 = b_3 = \alpha s_j$ and $c_1 = c_2 = c_3 = \alpha s_l$ makes its solution for all $\alpha > 0$. Furthermore, the solution set is unbounded, since α can be arbitrarily large.

Study the system S composed of 3 machines having speeds $1 \leq r \leq s$ and n players. Let L indicate a Nash equilibrium in the system S such that machine i is slowest in this profile having the greatest delay, machine j has a medium delay and machine l is fastest. Suppose that in the equilibrium L machine i receives the total volume of jobs defined by $\sum\limits_{k=1,\dots,n:l_k=i} w_k = a_1 + a_2 + a_3$ and the corresponding volumes for machines j and l equal $\sum\limits_{k=1,\dots,n:l_k=j} w_k = b_1 + b_2 + b_3$ and $\sum\limits_{k=1,\dots,n:l_k=l} w_k = c_1 + c_2 + c_3$, respectively. The volume of jobs on each machine is somehow divided into three parts so that each component of the three-dimensional vectors a, b and c is either zero or positive and includes at least one job.

Lemma 5.1:
Let L be a Nash equilibrium in the game involving three machines i, j and l and n players such that

$$\lambda_i(L) \geq \lambda_j(L) \geq \lambda_l(L),$$
$$\sum_{k=1,\dots,n:l_k=i} w_k = a_1 + a_2 + a_3,$$
$$\sum_{k=1,\dots,n:l_k=j} w_k = b_1 + b_2 + b_3,$$
$$\sum_{k=1,\dots,n:l_k=l} w_k = c_1 + c_2 + c_3.$$

Here for all $k = 1, 2, 3$ component a_k equals zero or the volume of at least one job on machine i, component b_k equals zero or the volume of at least one job on machine j, and component

c_k equals zero or the volume of at least one job on machine l. Then the set of the vectors a, b and c is the solution of the system (5.3).

Proof
Suppose that L represents a Nash equilibrium and $\lambda_i(L) \geq \lambda_j(L) \geq \lambda_l(L)$. The lemma 3.1 claims that $\lambda_k(L) > 0$, $k = i, j, l$. In this case, the following inequalities take place:

$$\begin{cases} \dfrac{\sum\limits_{k=1,\ldots,n:l_k=i} w_k}{v_i} \leq \dfrac{\sum\limits_{k=1,\ldots,n:l_k=j} w_k + \min\limits_{k=1,\ldots,n:l_k=i,w_k>0} w_k}{v_j} \\ \dfrac{\sum\limits_{k=1,\ldots,n:l_k=i} w_k}{v_i} \leq \dfrac{\sum\limits_{k=1,\ldots,n:l_k=l} w_k + \min\limits_{k=1,\ldots,n:l_k=i,w_k>0} w_k}{v_l} \\ \dfrac{\sum\limits_{k=1,\ldots,n:l_k=j} w_k}{v_j} \leq \dfrac{\sum\limits_{k=1,\ldots,n:l_k=l} w_k + \min\limits_{k=1,\ldots,n:l_k=j,w_k>0} w_k}{v_l} \\ \dfrac{\sum\limits_{k=1,\ldots,n:l_k=i} w_k}{v_i} \geq \dfrac{\sum\limits_{k=1,\ldots,n:l_k=j} w_k}{v_j} \geq \dfrac{\sum\limits_{k=1,\ldots,n:l_k=l} w_k}{v_l}. \end{cases}$$

Since each nonzero quantity a_k ($k = 1, 2, 3$) equals the volume of at least one job on machine i, then we naturally have that $\min\limits_{k:a_k>0} a_k \geq \min\limits_{k:l_k=i,w_k>0} w_k$ that provides satisfaction of the first and the second inequality of the system (5.3). Similarly, $\min\limits_{k:b_k>0} b_k \geq \min\limits_{k:l_k=j,w_k>0} w_k$. This means satisfaction of the system (5.3). □

Lemma 5.2:
Any solution of the system (5.3) defines a Nash equilibrium L in the game involving the system S composed of 3 machines i, j and l and players whose jobs correspond to the nonzero components of the vectors a, b and c and the delays are sorted in the order $\lambda_i(L) \geq \lambda_j(L) \geq \lambda_l(L)$.

Proof
Assume that the set of the vectors a, b and c gives the solution of the system (5.3). Consider the game with 3 machines i, j and l. Let each nonzero component of the vectors a, b and c specify the job volume of a regular player. Consider a profile L such that the jobs of volumes $a_k > 0$, $b_k > 0$ and c_k are assigned to machines i, j and l, respectively. So long as all inequalities (5.3) hold true, the profile L gives the desired Nash equilibrium. □

The following result is immediate.

Theorem 5.1:
Any Nash equilibrium L in the game involving the system S composed of 3 machines i, j and l and n players corresponds to a Nash equilibrium L' in the game involving the same system S and at most 9 players, where each machine receives no more than 3 jobs and the delays on all machines in L and L' do coincide.

Proof
Consider a Nash equilibrium L in the game with the system S of 3 machines and n players. Number the machines so that $\lambda_i(L) \geq \lambda_j(L) \geq \lambda_l(L)$. According to Lemma 5.1, for any Nash equilibrium in the game involving the system S and any number of players there exist a corresponding solution a, b, c of the system (5.3). By virtue of Lemma 5.2, this solution determines a Nash equilibrium L' in the game with the system S such that the nonzero components of the vectors a, b and c specify the job volumes on machines i, j and l, respectively. By definition, the element sum of the vector a represents the load of machine i in a profile L. Hence, delays on machine i coincide in both equilibria L and L'. Similarly, for machines j and l the delays in the equilibrium L coincide with the corresponding delays in the equilibrium L'. □

This theorem claims that it is sufficient to consider only games, where in an equilibrium each machine receives at most three jobs and the equilibrium solves the system (5.3). And the domain of the social payoff coincides with the value domain of games with an arbitrary number of players.

Imagine that the components of the vectors a, b and c are chosen as follows. In the optimal profile yielding the maximum social payoff, machines i, j and l receive the total volumes of jobs $a_1 + b_1 + c_1$, $a_2 + b_2 + c_2$ and $a_3 + b_3 + c_3$, respectively, and the lowest delay can be on each of them. Furthermore, by Theorem 3.2, the volumes of jobs are assumed to be normalized so that in the optimal profile the maximum delay among all machines equals 1. In our case, this means that

$$\begin{aligned} a_1 + b_1 + c_1 \geq v_i, \\ a_2 + b_2 + c_2 \geq v_j, \\ a_3 + b_3 + c_3 \geq v_l, \end{aligned}$$

and at least one of these inequalities holds as an equality.

Lemma 5.3:
Solution of the linear programming problem

$$LPP(v_i, v_j, v_l): \begin{cases} c_1 + c_2 + c_3 \to \min \\ (r1) \quad \frac{a_1+a_2+a_3}{v_i} \leq \frac{b_1+b_2+b_3+\min\limits_{k:a_k>0} a_k}{v_j} \\ (r2) \quad \frac{a_1+a_2+a_3}{v_i} \leq \frac{c_1+c_2+c_3+\min\limits_{k:a_k>0} a_k}{v_l} \\ (r3) \quad \frac{b_1+b_2+b_3}{v_j} \leq \frac{c_1+c_2+c_3+\min\limits_{k:b_k>0} b_k}{v_l} \\ (r4) \quad \frac{a_1+a_2+a_3}{v_i} \geq \frac{b_1+b_2+b_3}{v_j} \geq \frac{c_1+c_2+c_3}{v_l} \\ (r5) \quad \max\limits_{k=1,2,3} a_k > 0 \\ (r6) \quad \max\limits_{k=1,2,3} b_k > 0 \\ (r7) \quad a_k, b_k, c_k \geq 0, k = 1, 2, 3 \\ (r8) \quad a_1 + b_1 + c_1 \geq v_i \\ (r9) \quad a_2 + b_2 + c_2 \geq v_j \\ (r10) \quad a_3 + b_3 + c_3 \geq v_l \end{cases} \tag{5.4}$$

with respect to the components of the vectors a, b and c provides the minimal social payoff in a Nash equilibrium among all games, where in an equilibrium at most 3 jobs are assigned to each machine, i, j and l indicate the numbers of the machines in the descending order of their delays and the optimal social payoff makes up 1.

Proof
Due to Lemma 5.2, any solution of inequalities $(r1) - (r7)$ in the problem $LPP(v_i, v_j, v_l)$ defines an equilibrium in the game with 3 machines, where each machine receives at most 3 jobs and i, j, and l are the numbers of machines in the descending order of their delays

The goal function in this game is bounded above only by the hyperplanes corresponding to inequalities $(r8) - (r10)$. Actually, inequalities $(r1) - (r7)$ admit arbitrarily small non-negative values of the goal function, including zero. Therefore, the minimum is reached on one of the boundaries answering to the last three inequalities. This guarantees that one of them is satisfied as an equality, *ergo* the optimal payoff in the game corresponding to the solution of the problem $LPP(v_i, v_j, v_l)$ equals 1. □

Consequently, exact PoA evaluation for the system S composed of 3 machines calls for solving a series of linear programming methods $LPP(v_i, v_j, v_l)$ for all permutations $(1, r, s)$. And the minimum solution among them yields the value of $PoA(S)$. In other words, it is possible to establish the following fact.

Theorem 5.2:
For the system S composed of 3 machines having speeds $v_1 = 1 \leq v_2 = r \leq v_3 = s < 2$, the price of anarchy constitutes $PoA(S)$, which is the inverse value of

$$\frac{1}{PoA(S)} = \min_{(v_i,v_j,v_l) \text{ are permutations of } (1,r,s)} \left\{ \frac{c_1+c_2+c_3}{v_l} | a, b, c \text{ is a solution of } LPP(v_i, v_j, v_l) \right\},$$

where $LPP(v_i, v_j, v_l)$ is the linear programming problem (5.4).

Proof
According to Lemma 5.3, the solution of the problem (5.4) gives the minimum social payoff in a Nash equilibrium, where i, j and l are the numbers of the machines in the descending order of their delays, among all games such that in an equilibrium each machine receives at most 3 jobs and the optimal payoff equals 1. The minimum solution among the problems for all admissible permutations $(1, r, s)$ as the values of (v_i, v_j, v_l) provides the minimum social payoff in a Nash equilibrium among all games, where in an equilibrium at most 3 jobs are assigned to each machine and the optimal payoff equal 1.

By Theorem 5.1, for any equilibrium in the game involving the system S of 3 machines and an arbitrary number of players, it is possible to construct a corresponding equilibrium in the game with the same machines and a set of at most 9 players, where each machine receives no more than 3 jobs and the social payoff coincides for both equilibria. Thus, for PoA evaluation it suffices to consider only games, where in an equilibrium each machine has at most 3 jobs.

Using Theorem 3.2, we finally obtain that for PoA evaluation it suffices to consider only games, where the social payoff in the optimal profile equal 1. □

6. COMPUTING EXPERIMENTS

To estimate the price of anarchy in the three-machine model, we have developed a program implementation of PoA evaluation method presented in the previous section. This program allows to compare visually the theoretic PoA value and its exact value constructed by solving a series of linear programming problems. Moreover the program provides the possibility to see the PoA dynamics for the machine number $N > 3$ where no any theoretic PoA estimates are obtained. The parameters of the system S act as the options in the program; by assumption, the speed of machine 1 equals 1, whereas an exact value and a certain range are specified for the speeds of machines 2 and 3, respectively. In this case, users can study the PoA dynamics under variations in the speed of one machine.

The figures 6.1 and 6.2 present examples of PoA estimates for different values of speeds of machine 2 and 3. At the fig. 6.1 the speed of machine 2 is $r = 1.1$, and the speed of machine 3 is $s \in [r, 2)$. At the fig. 6.2 the speed of the fastest machine 3 is $s = 1.7$ and the speed of machine 2 is $r \in [1, s]$. Here we can see that theoretical and computed values of PoA coincide.

The next example is more interesting. Consider the system of four machines with speeds $v_1 = 1 \leq v_2 = q \leq v_3 = r \leq v_4 = s < 2$. Figures 6.3 and 6.4 present PoA comparing with the lower PoA estimate (3.2), which in fact is PoA for the system composed of 3 machines with speeds $1 \leq r \leq s < 2$. Fig. 6.3 presents PoA for the following cases. At the area A the value of q changes in the range $[1, r]$, $r = 1.3$, $s = 1.5$. At the area B $q = 1.3$, the value of r changes in the range $[q, s]$, $s = 1.5$. At the area C $q = 1.3$, $r = 1.5$, s value changes in the range $[r, 2)$. In these cases PoA value for four-machine system coincide with its lower estimate (3.2).

Fig. 6.4 presents the PoA dynamics for those systems where machine speeds differ rather little, that is normalized speeds are closed to 1. In this case one can see that the PoA value

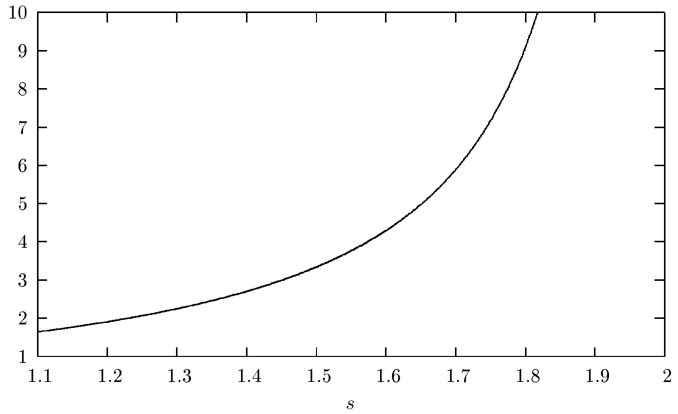

Fig. 6.1. PoA for the system S, where $r = 1.1, s \in [r, 2)$.

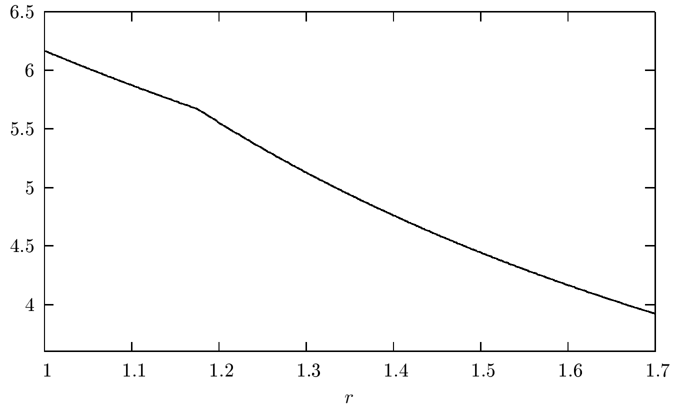

Fig. 6.2. PoA for the system S, where $s = 1.7, r \in [1, s]$.

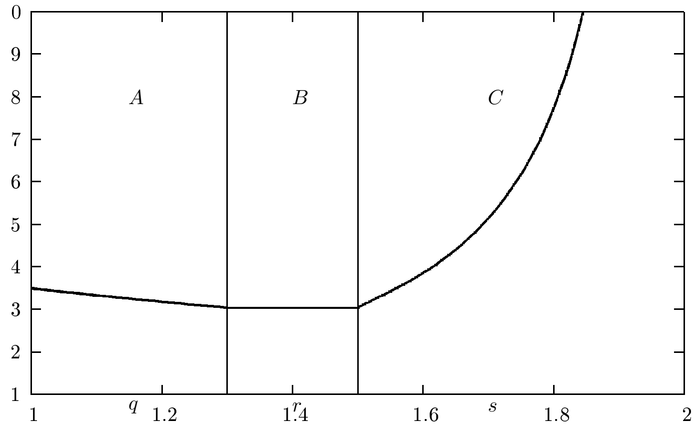

Fig. 6.3. PoA for the four-machine system S.

presented by the thin curve exceeds its estimate (3.2) presented by the bold curve. Both curves coincide under machine speeds increasing. At the area A the value of q changes in the range $[1, r]$, $r = 1.05$, $s = 1.1$. At the area B $q = 1.05$, the value of r changes in the range $[q, s]$, $s = 1.1$. At the area C $q = 1.05$, $r = 1.1$, the value of s changes in the range $[r, 1.3)$.

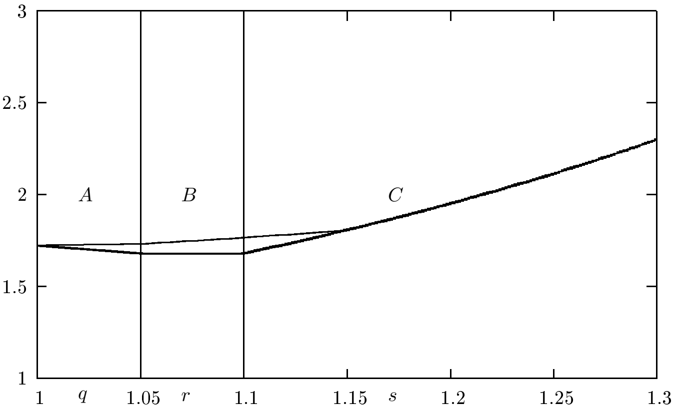

Fig. 6.4. PoA for the four-machine system S with small speeds

7. CONCLUSION

This paper has explored the service system composed of N machines and n players and derived the lower estimate for the price of anarchy in the maximizing the minimum machine delay game (or cover game). The three-machine model has been analyzed in detail. Here we have determined the exact value of the price of anarchy and showed that the PoA increases or does not change under new machine inclusion into the system of two machines. Also we have proposed a computing algorithm of the exact PoA value. The algorithm can be generalized to systems with more machines, but this increases the number of linear programming problems to-be-solved and the number of associated variables and imposed constraints. And finally, we have implemented the algorithm as a program and conducted numerical experiments for comparing the obtained estimates of the PoA with its exact value. The results of these experiments have demonstrated the correctness of the derived estimates. For the case of four-machine system computing experiments demonstrate partial PoA coinciding for three and four-machine systems, the analytic confirmation of this fact needs further investigations.

ACKNOWLEDGEMENTS

This work was supported by the Mathematical Sciences Branch of the Russian Academy of Sciences and by the Russian Foundation for Basic Research (project no. 16-51-55006), and by the Russian Humanitarian Scientific Foundation (project no. 15-02-00352_a).

8. APPENDIX

This appendix contains supporting lemmas which are used in proofs in section 4.

Lemma 8.1:
For any real r, s *such that* $1 \le r \le s < 2$ *we have* $s \le \min\{\frac{2+s}{(1+r)(2-s)}, \frac{2}{r(2-s)}\}$.

Proof
$\frac{2}{r(2-s)} \ge \frac{2}{s(2-s)} \ge s$, since $s^3 - 2s^2 + 2 = s(s-1)^2 + (2-s) > 0$.
$\frac{2+s}{(1+r)(2-s)} \ge \frac{2+s}{(1+s)(2-s)} \ge s$, since $s^3 - s^2 - s + 2 > s^3 - 2s^2 + 2 = s(s-1)^2 + (2-s) > 0$. □

Lemma 8.2:
For any real r, s *such that* $1 \le r \le s < 2$ *we have* $\frac{3s}{1+r} \le \min\{\frac{2+s}{(1+r)(2-s)}, \frac{2}{r(2-s)}\}$.

Proof
First, $3s \le \frac{2+s}{2-s}$, due to $3s^2 - 5s + 2 = (s-1)(3s-2) > 0$. Second, $\frac{3s}{1+r} \le \frac{2}{r(2-s)}$, since $6rs - 3rs^2 - 2 - 2r = r(6s - 3s^2 - 2) - 2 = r(1 - 3(s-1)^2) - 2 \le r - 2 < 0$. □

Lemma 8.3:
For $v_i < v_j$, $v_i, v_j \in \{1, r, s\}$, *where real* r, s *are such that* $1 \le r \le s < 2$, *we have* $\frac{2v_i^2 + v_i v_j}{2v_i - v_j} \le \frac{2+s}{2-s}$.

Proof
If $v_i < v_j$, then $\frac{2v_i^2 + v_i v_j}{2v_i - v_j}$ decreases by v_i and increases by v_j, since $4v_i^2 - 4v_i v_j - v_j^2 < 0$ and $v_i(2v_i - v_j) + 2v_i^2 + v_i v_j > 0$. □

Lemma 8.4:
For any real r, s *such that* $1 \le r \le s < 2$ *we have* $\frac{2r^2 + rs}{(1+r)(2r-s)} < \frac{2}{r(2-s)}$.

Proof
The inequality in the condition is equivalent to $f(r, s) = -r^2 s^2 - 2s(r^3 - r^2 - r - 1) + 4(r^3 - r^2 - r) < 0$, chech if it holds true. Show that $f'_r(r, s) = -2rs^2 + 2(2-s)(3r^2 - 2r - 1) < 0$, then $f(r, s) \le f(1, s) = -s^2 + 4s - 4 = -(2-s)^2 < 0$.

For each fixed s the function $f'_r(r, s)$ is a parabola with branches directed upwards. Thus, its largest value is achieved on one of two ends of the interval $r \in [1, s]$. At the left end $f'_r(1, s) = -2s^2 < 0$. At the right end $f'_r(s, s) = -8s^3 + 16s^2 - 6s - 4 = -8s(s-1)^2 - 2(2-s) < 0$. □

Lemma 8.5:
For $v_i \ne v_j \ne v_l$, $v_i, v_j, v_l \in \{1, r, s\}$, *where real* r, s *are such that* $1 \le r \le s < 2$, *we have* $f(v_i, v_j, v_l) = v_i + \frac{2v_i v_j}{2v_i - v_j} + \frac{3v_i v_l}{3v_i - v_l} \le 1 + \frac{2s}{2-s} + \frac{3s}{3-s}$.

Proof
The function $f(v_i, v_j, v_l)$ obviously increases by v_j and v_l, hence $f(v_i, v_j, v_l) \le v_i + \frac{2sv_i}{2v_i - s} + \frac{3sv_i}{3v_i - s} = g(v_i)$.

Show that $g(v_i)$ decreases by v_i. The derivative $g'_{v_i}(v_i) = 1 - \frac{2s^2}{(2v_i - s)^2} - \frac{3s^2}{(3v_i - s)^2}$ increases by v_i and, therefore, does not exceed $g'_{v_i}(s) = 1 - 2 - \frac{3}{4} < 0$.

Then $g(v_i) \le g(1) = 1 + \frac{2s}{2-s} + \frac{3s}{3-s}$. □

Lemma 8.6:
For any real r, s *such that* $1 \le r \le s < 2$ *we have* $\frac{1 + \frac{2s}{2-s} + \frac{3s}{3-s}}{1+r+s} \le \min\{\frac{2+s}{(1+r)(2-s)}, \frac{2}{r(2-s)}\}$.

Proof
We show first that $1 + \frac{2s}{2-s} + \frac{3s}{3-s} \le \frac{(1+r+s)(2+s)}{(1+r)(2-s)}$. The right part of the inequality decreases by r, thus it suffices to show that $1 + \frac{2s}{2-s} + \frac{3s}{3-s} \le \frac{(1+2s)(2+s)}{(1+s)(2-s)}$. This is equivalent to $s \le s^2$, that holds true under $s \ge 1$.

Show now that $1 + \frac{2s}{2-s} + \frac{3s}{3-s} \le \frac{2(1+r+s)}{r(2-s)}$. The right part of the inequality decreases by r, so it suffices to show that $1 + \frac{2s}{2-s} + \frac{3s}{3-s} \le \frac{2(1+2s)}{s(2-s)}$. This inequality is equivalent to $-4s^3 + 11s^2 - 4s - 6 = -s(2s-3)^2 - (2-s)(3-s) < 0$. □

REFERENCES

1. Epstein L., Kleiman E., & van Stee R. (2009) Maximizing the minimum load: the cost of selfishness. In *Proceedings of the 5th International Workshop on Internet and Network Economics*, LNCS 5929, 232-243.
2. Tan Z., Wan L., Zhang Q. & Ren W. (2012) Inefficiency of equilibria for the machine covering game on uniform machines. *Acta Informatica*, **49** (6), 361–379.
3. Wu Y., Cheng T. C. E. & Ji M. (2015) Inefficiency of the Nash equilibrium for selfish machine covering on two hierarchical uniform machines. *Information Processing Letters*, **115** (11), 838–844.
4. Andelman N., Feldman M. & Mansour Y. (2007) Strong price of anarchy. In *Proc. of the 18th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 189–198.
5. Koutsoupias E., Papadimitriou C. H. (1999) Worst-case Equilibria. In *Proc. of STACS 1999*, **1563**, 404–413.
6. Lücking T., Mavronicolas M., Monien B., Rode M., Spirakis P. & (2003) Vrto I. Which is the Worst-case Nash Equilibrium? In *Proc. of the 26th International Symposium on Mathematical Foundations of Computer Sciencem, LNCS 2747*, 551–561.
7. Even-Dar E., Kesselman A. & Mansour Y. (2003) Convergence time to Nash equilibria. In *Proc. of the 30th International Colloquium on Automata, Languages and Programming (ICALP2003)*, 502–513.
8. Fotakis D., Kontogiannis S. C., Koutsoupias E., Mavronicolas M. & Spirakis P. G. (2002) The structure and complexity of Nash equilibria for a selfish routing game. In *Proc. of the 29th International Colloquium on Automata, Languages and Programming (ICALP2002)*, 123–134.
9. Chen X., Epstein L., Kleiman E. & van Stee R. (2013) Maximizing the minimum load: The cost of selfishness. *Theoretical Computer Science*, **482**, 9–19.
10. Murchland J. D. (1970) Braess's paradox of traffic flow. *Transportation Research*, **4**, 391–394.
11. Roughgarden T. & Tardos É. (2002) How bad is selfish routing? *Journal of the ACM*, **49** (2), 236–259.
12. Korilis Y. A., Lazar A. A. & Orda A. (1999) Avoiding the Braess paradox in non-cooperative networks. *Journal of Apllied Probability*, **36** (1), 211–222.
13. Mazalov, V. V. (2014) *Mathematical Game Theory and Applications*. New York, NY: Wiley.
14. Chirkova Yu. V. (2015) Price of anarchy in machine load balancing game. *Automation and Remote Control*, **76** (10), 1849–1864.

