Computing the Price of Anarchy in Processor Load Balancing Game with Linear Delays

Julia V. Chirkova

Institute of Applied Mathematical Research,
Karelian Research Centre of RAS,
Pushkinskaya str., 11, Petrozavodsk, Karelia, 185910, Russia
E-mail: `julia@krc.karelia.ru`
WWW home page: `http://www.krc.karelia.ru/HP/julia`

Abstract This paper considers a generalization of the processor load balancing game also known as KP-model. A linear delay of a processor may depend on not only its load but on loads of other processors. Players choose processors of different speeds to run their jobs striving to minimize job's delay, i.e., the job completion time on a chosen processor. The social cost is the maximum delay over all processors. We propose a computing algorithm of the exact PoA value which can be applied to estimate the POA visually if its exact analytical expression is not obtained yet or it is rather complicated to figure out its formula.

Keywords: processor load balancing game, Nash equilibrium, price of anarchy, linear functional, computation.

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. 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 a generalization of the machine load balancing game in the form of the KP-model (see Koutsoupias and Papadimitriou, 1999) with parallel different-capacity channels. It is necessary to distribute several jobs of various volumes among processors of nonidentical speeds. The volume of a job is its completion time on a free unit-speed processor when all other processors are idle. Processor load is the total volume of jobs executed by a given processor. A linear delay of a processor, i.e. completion time for all jobs on this processor, may depend on not only its load but on loads of other processors. Each player chooses a processor 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 processor. In the sequel, we study pure strategies Nash equilibria only; as is well-known (Fotakis et al., 2002) for the original KP-model such an equilibrium always exists. The system cost (also called the social

cost) is the maximum delay over all processors for an obtained job distribution. The price of anarchy (Koutsoupias and Papadimitriou, 1999) (PoA) is defined as the maximum ratio of the social cost in the worst-case Nash equilibrium and the optimal social cost.

The POA evaluating is an important problem, since this value is an indicator of the system quality. It shows how a non-cooperative players behavior cold be worse than cooperative. Sometimes in special cases it is possible to find an exact analytical expression for the POA value, as for two-processors system in the KP-model (Epstein, 2010), or some its upper estimation, as for such three-processors system (Chirkova, 2015).

Numerical computing of the POA value and its graphical presentation allows estimating the POA visually if its exact analytical expression is not obtained yet or it is rather complicated to figure out its formula. We propose a computing algorithm of the exact PoA value. This is a generalization of the computing method for the original KP-model, presented in Chirkova, 2015. Here a linear delay of a processor may depend on not only its load but on loads of other processors.

2. The model

Consider a system of N processors of different performances. The system is used by a set of players $U = U(n, w)$: each of n players chooses an appropriate processor for its job execution. For player j, the volume of job equals w_j, $j = 1, \ldots, n$. Denote by $W = \sum_{j=1}^{n} w_j$ the total volume of all jobs.

We study the following pure strategies game. Each player can choose any processor. The strategy of player j is processor l_j selected by this player for its job execution. Then the strategy profile in the game Γ represents the vector $L = (l_1, \ldots, l_n)$. The load of processor i, i.e., the total volume of all jobs assigned to the processor is defined by $\delta_i(L) = \sum_{j=1,\ldots,n: l_j=i} w_j$. The delay of processor i takes the form of a linear functional

$$\lambda_i(L) = (\alpha_{i1}, \ldots, \alpha_{iN})(\delta_1(L), \ldots, \delta_N(L))^T$$

for the vector of processor loads and a vector $\alpha_i \in R^N$. Actually, this quantity is the same for all players selecting a given processor. Denote also $S = S(N, \lambda)$ the system of N processors with given delays λ. Therefore the game is described by $\Gamma(S, U) = < S(N, \lambda), U(n, w) >$.

The social cost is described by the maximum delay over all processors:

$$SC(L) = \max_{i=1,\ldots,N} \lambda_i(L).$$

Designate by

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

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

A strategy profile L such that none player benefits from unilateral deviation (change of the processor chosen in L for its job execution) is a pure strategies Nash

equilibrium. To provide a formal definition, let $L(j \to i) = (l_1, \ldots, l_{j-1}, i, l_{j+1}, \ldots, l_n)$ signify the profile obtained from a profile L if player j replaces processor l_j chosen by it in the profile L for another processor i, whereas the rest players keep the same strategies as before.

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

Note that while the original KP-model always has a pure Nash equilibrium (Fotakis et al., 2002), in our generalized case the game may not to possess any pure Nash equilibrium.

Example 1. Here is an example of simple game which does not have any pure NE. Consider the game with two jobs of volumes 1 and three processors of speeds 1. The following delays are given: $\lambda_1(L) = \delta_1(L) + \epsilon\delta_2(L)$, $\lambda_2(L) = \delta_2(L) + \epsilon\delta_3(L)$, $\lambda_3(L) = \delta_3(L) + \epsilon\delta_1(L)$. Consider an arbitrary profile, e.g. $L = (1, 2)$. The first job obtains a delay $1 + \epsilon$, the second one obtains 1. The first job migrates to the free processor 3 and obtains a delay 1. Then the second job delay increases to $1 + \epsilon$ and forces it to migrate to the first processor which becomes idle. It increases the first job delay, which needs to migrate to the processor 2. Repeating the process we obtain an infinite profile loop $(1, 2) \to (3, 2) \to (3, 1) \to (2, 1) \to (2, 3) \to (1, 3) \to (1, 2) \to \ldots$.

Further we assume that the system and job sets are such that the game possesses a pure Nash equilibrium. We define Price of Anarchy of the corresponding system as follows.

Definition 2. The price of anarchy in the system S is the maximum ratio of the social cost in the worst-case Nash equilibrium and the social cost in the optimal case:
$$PoA(S) = \max_U \frac{\max\limits_{L \text{ is a Nash equilibrium in } \Gamma(S,U)} SC(L)}{OPT(S,U)}.$$

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

Theorem 1. *For the system S, the price of anarchy constitutes*
$$PoA(S) = \max_{U_1 : OPT(S,U_1)=1} \max_{L \text{ is a Nash equilibrium in the game } \Gamma(S,U_1)} SC(L).$$

Proof. Assume that L is the worst-case equilibrium in the game $\Gamma(S, U)$ 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. Denote a load $\delta_i' = \delta_i(L)$ for each processor i in the equilibrium L in this game. Let SC and OPT be the social cost in the profile L and the optimal social cost, respectively. The ratio of the worst-case and optimal social cost is defined by $\frac{SC}{OPT}$. So long as L represents an equilibrium, then for any player j we obtain that $\sum\limits_{k=1,\ldots,N} \alpha_{l_j k}\delta_k' \leq \sum\limits_{k=1,\ldots,N} \alpha_{ik}\delta_k' + w_j(\alpha_{ii} - \alpha_{il_j})$ for any processor i.

Now, explore the game with the same set of processors and players, where each player j has the job of volume $\frac{w_j}{OPT}$. Then for any processor i its load is $\frac{\delta'_i}{OPT}$. The social cost in the profiles L and L_{OPT} constitutes $\frac{SC}{OPT}$ and 1, respectively. Now let's show that L and L_{OPT} form the worst-case 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 $\frac{\sum\limits_{k=1,\ldots,N} \alpha_{l_j k}\delta'_k}{OPT} \leq \frac{\sum\limits_{k=1,\ldots,N} \alpha_{ik}\delta'_k + w_j(\alpha_{ii}-\alpha_{il_j})}{OPT}$ holds true for any processor i. Imagine that L is any non-worst-case equilibrium in the new game. Then the game admits an equilibrium L' with social cost $\frac{SC'}{OPT}$ such that the social cost in the profile L' exceeds 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 cost $SC' > SC$, 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 worst-case and optimal social cost in the new game also equals $\frac{SC}{OPT}$.

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

3. Evaluating the POA in the 3-processor model

We suggest a computing method for the price of anarchy in the system of 3 processors. This method can be generalized to systems composed of more processors. But such generalization increases the number of linear programming problems to-be-solved and the number of associated variables and imposed constraints.

Consider the following system of linear equations in the components of the vectors $a = (a_1, a_2, a_3), b = (b_1, b_2, b_3), c = (c_1, c_2, c_3)$. Denote $A(a) = a_1 + a_2 + a_3$, $B(b) = b_1 + b_2 + b_3$, $C(c) = c_1 + c_2 + c_3$, $\Delta(a, b, c) = (A(a), B(b), C(c))^T$, $\alpha_k = (\alpha_{k1}, \alpha_{k2}, \alpha_{k3})$. Numbers i, j, l are components of the set of processor numbers $1, 2, 3$ so that $i \neq j \neq l \neq i$.

$$\begin{cases} \alpha_i \Delta(a,b,c) \leq \alpha_j \Delta(a,b,c) + \min\limits_{k=1,2,3:a_k>0} a_k(\alpha_{jj} - \alpha_{ji}) \\ \alpha_i \Delta(a,b,c) \leq \alpha_l \Delta(a,b,c) + \min\limits_{k=1,2,3:a_k>0} a_k(\alpha_{ll} - \alpha_{li}) \\ \alpha_j \Delta(a,b,c) \leq \alpha_l \Delta(a,b,c) + \min\limits_{k=1,2,3:b_k>0} b_k(\alpha_{ll} - \alpha_{lj}) \text{ or } \max\limits_{k=1,2,3} b_k = 0 \\ \alpha_i \Delta(a,b,c) \geq \alpha_j \Delta(a,b,c) \geq \alpha_l \Delta(a,b,c) \\ a_k, b_k, c_k \geq 0, k = 1, 2, 3. \end{cases} \quad (1)$$

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. If the above system is feasible and (a, b, c) is its solution, then for any real number $\beta > 0$, a set $(\beta a, \beta b, \beta c)$ is a solution too. Furthermore, the solution set is unbounded, since β can be arbitrarily large.

Study the system S composed of 3 processors with numbers i, j, l and n players. Let L indicate a Nash equilibrium in the system S such that processor i has a maximal delay in this profile, processor j has a medium delay and processor l has a minimal, otherwise we may re-enumerate processors. Suppose that in the equilibrium L processor i receives the total volume of jobs defined by $\sum\limits_{k=1,\ldots,n:l_k=i} w_k = a_1 + a_2 + a_3$

and the corresponding volumes for processors j and l equal $\sum\limits_{k=1,\ldots,n:l_k=j} w_k = b_1 + b_2 + b_3$ and $\sum\limits_{k=1,\ldots,n:l_k=l} w_k = c_1 + c_2 + c_3$, respectively. The volume of jobs on each processor 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 1. *Let L be a Nash equilibrium in the game involving three processors i, j and l, and n players such that*

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

Here for all $k = i, j, l$ component a_k equals zero or the volume of at least one job on processor i, component b_k equals zero or the volume of at least one job on processor j, and component c_k equals zero or the volume of at least one job on processor l. Then the set of the vectors a, b and c is the solution of the system (1).

Proof. Suppose that L represents a Nash equilibrium and $\lambda_i(L) \geq \lambda_j(L) \geq \lambda_l(L)$. Denote $\delta(L) = (\delta_i(L), \delta_j(L), \delta_l(L))^T$ a vector-column of the processors loads. In this case, the following inequalities take place:

$$\begin{cases} \alpha_i \delta_i(L) \leq \alpha_j \delta_j(L) + \min\limits_{k=1,\ldots,n:l_k=i,w_k>0} w_k(\alpha_{jj} - \alpha_{ji}) \\ \alpha_i \delta_i(L) \leq \alpha_l \delta_l(L) + \min\limits_{k=1,\ldots,n:l_k=i,w_k>0} w_k(\alpha_{ll} - \alpha_{li}) \\ \alpha_j \delta_j(L) \leq \alpha_l \delta_l(L) + \min\limits_{k=1,\ldots,n:l_k=j,w_k>0} w_k(\alpha_{ll} - \alpha_{lj}) \text{ or } \max\limits_{k=1,\ldots,n:l_k=j} w_k = 0 \\ \alpha_i \delta_i(L) \geq \alpha_j \delta_j(L) \geq \alpha_l \delta_l(L). \end{cases}$$

It's clear that $\delta(L) = \Delta(a, b, c)$. Since each nonzero quantity a_k ($k = 1, 2, 3$) equals the volume of at least one job on processor i, then we naturally have that $\min\limits_{k:a_k>0} a_k \geq \min\limits_{k:l_k=i,w_k>0} w_k$. Similarly, $\min\limits_{k:a_k>0} a_k \geq \min\limits_{k:l_k=i,w_k>0} w_k$ and $\min\limits_{k:b_k>0} b_k \geq \min\limits_{k:l_k=j,w_k>0} w_k$. This means satisfaction of the system (1).

Lemma 2. *Any nonzero solution of the system (1) defines a Nash equilibrium L in the game involving the system S composed of 3 processors 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 (1). Consider the game with 3 processors 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 $c_k > 0$ are assigned to processors i, j and l, respectively. So long as all inequalities (1) hold true, the profile L gives the desired Nash equilibrium.

The following result is immediate.

Theorem 2. *Any Nash equilibrium L in the game involving the system S composed of 3 processors 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 processor receives no more than 3 jobs and the delays on all processors in L and L' do coincide.*

Proof. Consider a Nash equilibrium L in the game with the system S of 3 processors and n players. Number the processors so that $\lambda_i(L) \geq \lambda_j(L) \geq \lambda_l(L)$. According to Lemma 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 (1). By virtue of Lemma 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 processors i, j and l, respectively. By definition, the element sum of the vector a represents the load of processor i in a profile L. Hence, delays on processor i coincide in both equilibria L and L'. Similarly, for processors 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 processor receives at most three jobs and the equilibrium solves the system (1). And the domain of the social cost 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 minimum social cost, processors 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 highest delay can be on each of them. Furthermore, by Theorem 1, the volumes of jobs are assumed to be normalized so that in the optimal profile the maximum delay among all processors equals 1. In our case, this means that

$$\alpha_{ii}(a_1 + b_1 + c_1) + \alpha_{ij}(a_2 + b_2 + c_2) + \alpha_{il}(a_3 + b_3 + c_3) \leq 1,$$
$$\alpha_{ji}(a_1 + b_1 + c_1) + \alpha_{jj}(a_2 + b_2 + c_2) + \alpha_{jl}(a_3 + b_3 + c_3) \leq 1,$$
$$\alpha_{li}(a_1 + b_1 + c_1) + \alpha_{lj}(a_2 + b_2 + c_2) + \alpha_{ll}(a_3 + b_3 + c_3) \leq 1,$$

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

Denote now the linear programming problem $LPP(i, j, l)$ where i, j and l are numbers assigned with processors.

Lemma 3. *A nonzero solution of the linear programming problem $LPP(i, j, l)$:*

$$\begin{cases} \alpha_i \Delta(a, b, c) \to \max \\ (r1) \quad \alpha_i \Delta(a, b, c) \leq \alpha_j \Delta(a, b, c) + \min_{k=1,2,3:a_k>0} a_k(\alpha_{jj} - \alpha_{ji}) \\ (r2) \quad \alpha_i \Delta(a, b, c) \leq \alpha_l \Delta(a, b, c) + \min_{k=1,2,3:a_k>0} a_k(\alpha_{ll} - \alpha_{li}) \\ (r3) \quad \alpha_j \Delta(a, b, c) \leq \alpha_l \Delta(a, b, c) + \min_{k=1,2,3:b_k>0} b_k(\alpha_{ll} - \alpha_{lj}) \ or \ \max_{k=1,2,3} b_k = 0 \\ (r4) \quad \alpha_i \Delta(a, b, c) \geq \alpha_j \Delta(a, b, c) \geq \alpha_l \Delta(a, b, c) \\ (r5) \quad a_k, b_k, c_k \geq 0, k = 1, 2, 3 \\ (r6) \quad \alpha_{ii}(a_1 + b_1 + c_1) + \alpha_{ij}(a_2 + b_2 + c_2) + \alpha_{il}(a_3 + b_3 + c_3) \leq 1, \\ (r7) \quad \alpha_{ji}(a_1 + b_1 + c_1) + \alpha_{jj}(a_2 + b_2 + c_2) + \alpha_{jl}(a_3 + b_3 + c_3) \leq 1, \\ (r8) \quad \alpha_{li}(a_1 + b_1 + c_1) + \alpha_{lj}(a_2 + b_2 + c_2) + \alpha_{ll}(a_3 + b_3 + c_3) \leq 1, \end{cases}$$

$$(2)$$

with respect to the components of the vectors a, b and c provides the maximum social cost in a Nash equilibrium among all games, where in an equilibrium at most 3 jobs

are assigned to each processor, i, j and l indicate the numbers of the processors in the descending order of their delays and the optimal social cost makes up 1.

Proof. Due to Lemma 2, any solution of inequalities $(r1) - (r5)$ in the problem $LPP(i, j, l)$ defines an equilibrium in the game with 3 processors, where each processor receives at most 3 jobs and i, j, and l are the numbers of processors in the descending order of their delays.

The goal function in this game is bounded above only by the hyperplanes corresponding to inequalities $(r6) - (r8)$. Actually, inequalities $(r1) - (r5)$ admit arbitrarily large values of the goal function, since if (a, b, c) is their solution, then for any real number $\beta > 0$, a set $(\beta a, \beta b, \beta c)$ is a solution too. Therefore, the maximum 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 cost in the game corresponding to the solution of the problem $LPP(i, j, l)$ equals 1.

Consequently, exact PoA evaluation for the system S composed of 3 processors calls for solving a series of linear programming methods $LPP(i, j, l)$ for all permutations of processors. And the maximum solution among them yields the value of $PoA(S)$. In other words, it is possible to establish the following fact.

Theorem 3. *For the system S composed of 3 processors, the price of anarchy constitutes*

$$PoA(S) = \max_{\substack{(i,j,l) \, are \, permutations \, of \, (1,2,3)}} \left\{ \alpha_i \Delta(a, b, c) \mid a, b, c \text{ is a solution of } LPP(i, j, l) \right\},$$

where $LPP(i, j, l)$ is the linear programming problem (2).

Proof. According to Lemma 3, the solution of the problem (2) gives the maximum social cost in a Nash equilibrium, where i, j and l are the numbers of the processors in the descending order of their delays, among all games such that in an equilibrium each processor receives at most 3 jobs and the optimal cost equals 1. The maximum solution among the problems for all admissible permutations of processors $(1, 2, 3)$ provides the maximum social cost in a Nash equilibrium among all games, where in an equilibrium at most 3 jobs are assigned to each processor and the optimal cost equal 1.

By Theorem 2, for any equilibrium in the game involving the system S of 3 processors and an arbitrary number of players, it is possible to construct a corresponding equilibrium in the game with the same processors and a set of at most 9 players, where each processor receives no more than 3 jobs and the social cost coincides for both equilibria. Thus, for PoA evaluation it suffices to consider only games, where in an equilibrium each processor has at most 3 jobs.

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

4. The POA for the game with linear externalities

Consider the game with linear externalities (Chirkova, 2021a) as a special case of our model. In such game the delay of processor i takes the form

$$\lambda_i(L) = \sum_{j \in M: l_j = i} \frac{w_j}{v_i} + \sum_{k \neq i} e_{ik} \sum_{j \in M: l_j = k} w_j = \frac{\delta_i(L)}{v_i} + \sum_{k \neq i} e_{ik} \delta_k(L).$$

Here v_i is the speed of the processor i, and each externality coefficient e_{ik} reflects the impact of the load of the processor k into the delay of the processor i.

Also we assume the following:

1. For each processor pairs $i \neq k$ the inequality $e_{ik} < \frac{1}{v_i}$ holds.
2. For each processor pairs $i \neq k$ it holds that $e_{ki} < \frac{1}{v_i}$.
3. For each processor pair $i \neq k$, such that $v_i \geq v_k$, we have the inequality $\sum_{l \neq i} e_{il} \leq \sum_{l \neq k} e_{kl}$.

It is established (Chirkova, 2021a; Chirkova, 2021b) that there is at least one a pure Nash equilibrium in the game with linear externalities for the system of two processors. Moreover an expression for an exact value of the POA for the system of two processors with speeds $1 \leq s$ is found (Chirkova, 2021a). It equals to

$$Est(S) = \min\{Est^{max}(S), \frac{\eta}{s\zeta}\}, \tag{3}$$

where

$$Est^{max}(S) = \max\{\frac{\eta(\zeta + 1 - se_{21})}{\zeta(\eta + 1 - se_{21})}, \qquad \frac{s(1 - e_{12})}{\eta - s + s^2 e_{21}}, \\ \frac{s - se_{21} + e_{12}(1 - se_{12})}{\zeta}\},$$

and $\eta = 1 + s - s(e_{12} + e_{21})$, $\zeta = 1 - se_{12}e_{21}$.

But first the values of the POA for this model were found numerically to provide a visual estimation of the possibility to obtain an analytical expression and confirm its correctness. As the considered model is a special case of the generalized model with linear delays, similarly we obtain the linear programming problem $LPP(i,j)$:

$$\begin{cases} \frac{a_1 + a_2}{v_i} + e_{ij}(b_1 + b_2) \to \max \\ \frac{a_1 + a_2}{v_i} + e_{ij}(b_1 + b_2) \leq \frac{b_1 + b_2}{v_j} + e_{ji}(a_1 + a_2) + \min_{k=1,2:a_k > 0} a_k(\frac{1}{v_j} - e_{ji}) \\ \frac{a_1 + a_2}{v_i} + e_{ij}(b_1 + b_2) \geq \frac{b_1 + b_2}{v_j} + e_{ji}(a_1 + a_2) \\ a_k, b_k \geq 0, k = 1, 2 \\ \frac{a_1 + b_1}{v_i} + e_{ij}(a_2 + b_2) \leq 1 \\ \frac{a_2 + b_2}{v_j} + e_{ji}(a_1 + b_1) \leq 1. \end{cases} \tag{4}$$

The following theorem is a special case of the Theorem 3 reduced to two processors.

Theorem 4. *For the system S composed of 2 processors, the price of anarchy constitutes*

$$PoA(S) = \max_{(i,j) \, are \; permutations \; of \; (1,2)} \left\{ \frac{a_1 + a_2}{v_i} + e_{ij}(b_1 + b_2) | a, b \; is \; a \; solution \; of \; LPP(i,j) \right\},$$

where $LPP(i,j)$ is the linear programming problem (4).

4.1. Numerical examples

To estimate the price of anarchy in the two-machine model with linear externalities, we have developed a program allowing visual estimation of the the POA

exact value constructed by solving a series of linear programming problems. The parameters of the system S act as the options in the program; by assumption, the speed of machine 1 equals 1, whereas a certain range is for the speed of a machine 2. In this case, users can study the PoA dynamics under variations in the speed of one machine.

The Figure 1 presents examples of evaluated POA values in the model with two processors for different values of s. The Fig. 1 a) shows the POA for the externality values $e_{12} = 0.11$ and $e_{21} = 0.1$. The Fig. 1 b) presents the case when externalities are $e_{12} = 0.3$ and $e_{21} = 0.1$.

All numerically evaluated POA values coincide with corresponding values given by the analytical expression (3). In addition, the program randomly generates various sets of jobs U, finds for each resulting game $\Gamma(S, U)$ the ratio of the system costs in the worst case Nash equilibrium to the optimal costs and displays the corresponding points on the figure. All these points are located below the obtained POA values, that confirms the correctness of calculated POA values.
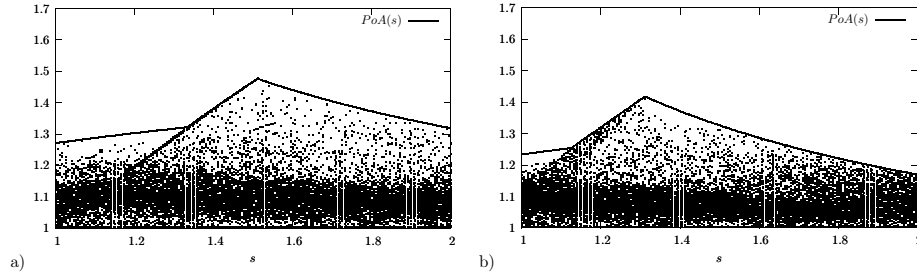

Fig. 1. The POA for the system S with a) $e_{12} = 0.11$, $e_{21} = 0.1$, b) $e_{12} = 0.3$, $e_{21} = 0.1$.

5. Conclusion

This paper has explored the service system composed of N machines and n players and derived the algorithm to compute an exact value for the price of anarchy in the load balancing game with linear delays. The three-machine model has been analyzed in detail. 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. Also similarly we can compute the POA value for the game of maximizing the minimal processor delay (Epstein et al., 2009; Chirkova, 2017; Chirkova, 2021b), considering the minimal delay among processors as a target maximized function. Finally, we have implemented the algorithm as a program and conducted numerical experiments for visual estimates of the PoA in the game with two processors and linear externalities, always possessing a pure Nash equilibrium. The results of these experiments have demonstrated the correctness of the derived algorithm.

References

Chirkova, J. V. (2021). *Load Balancing Game with Linear Externalities.* In: Mathematical Game Theory and its Applications, Vol. 13, Iss. 2, pp. 62–79 (in Russian).

Chirkova, J. V. (2021). *Maximizing the Minimum Processor Load with Linear Externalities.* In: Mathematical Optimization Theory and Operations Research: Recent Trends. MOTOR 2021. Communications in Computer and Information Science (Strekalovsky, A., Kochetov, Y., Gruzdeva, T., Orlov, A., eds), Vol. 1476, pp. 147–162. Springer, Cham.

Chirkova, Yu. V. (2015). *Price of anarchy in machine load balancing game.* Automation and Remote Control, **76(10)**, 1849–1864.

Chirkova, Yu. V. (2017). *Price of Anarchy for Maximizing the Minimum Machine Load.* Advances in Systems Science and Applications, **17(4)**, 61–77.

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.* Proc. of the 29th International Colloquium on Automata, Languages and Programming (ICALP2002), 123–134.

Epstein, L. (2010). *Equilibria for Two Parallel Links: The Strong Price of Anarchy Versus the Price of Anarchy.* Acta Inform. **47(7–8)**, 375–389.

Epstein L., Kleiman E., van Stee R. (2009). *Maximizing the minimum load: the cost of selfishness.* Proceedings of the 5th International Workshop on Internet and Network Economics, LNCS, **5929**, 232–243.

Koutsoupias, E. and Papadimitriou, C. H. (1999). *Worst-Case Equilibria.* Proc. STACS, **1563**, 404–413.

Lücking, T., Mavronicolas, M., Monien, B., Rode, M., Spirakis, P., Vrto, I. (2003). *Which is the Worst-case Nash Equilibrium?* Proc. of the 26th International Symposium on Mathematical Foundations of Computer Science, LNCS **2747**, 551–561.