Price of Anarchy in Non-Cooperative Load Balancing

U. Ayesta, O. Brun, B.J. Prabhu
LAAS-CNRS
Université de Toulouse
7 Avenue du Colonel Roche
31077 Toulouse, France.
email: \{urtzi, brun, bjprabhu\}@laas.fr

Abstract—We investigate the price of anarchy of a load balancing game with $K$ dispatchers. The service rates and holding costs are assumed to depend on the server, and the service discipline is assumed to be processor-sharing at each server. The performance criterion is taken to be the weighted mean number of jobs in the system, or equivalently, the weighted mean sojourn time in the system.

For this game, we first show that, for a fixed amount of total incoming traffic, the worst-case Nash equilibrium occurs when each player routes exactly the same amount of traffic, i.e., when the game is symmetric. For this symmetric game, we provide the expression for the loads on the servers at the Nash equilibrium. Using this result we then show that, for a system with two or more servers, the price of anarchy, which is the worst-case ratio of the global cost of the Nash equilibrium to the global cost of the centralized setting, is lower bounded by $K/(2\sqrt{K} - 1)$ and upper bounded by $\sqrt{K}$, independently of the number of servers.

Index Terms—atomic games, load balancing, processor sharing, price of anarchy.

I. INTRODUCTION

Server farms are used nowadays in as diverse areas as e-service industry, database systems and grid computing clusters. Figure 1 depicts the typical architecture of a server farm with a single centralized dispatcher who receives jobs from different sources and routes them to a set of servers. Server farms have become a popular architecture in computing centers, used for example in the Cisco Local Director, IBM Network Dispatcher and Microsoft Sharepoint (see [5] for a recent survey). This configuration can also be used to model a web server farm, where requests for files (or HTTP pages) arrive to a dispatcher and are dispatched immediately to one of the servers in the farm for processing.

One of the fundamental issue in this context is to characterize the optimal routing strategy. The problem amounts to find the routing strategy of the dispatcher that will optimize a certain performance objective, such as the mean processing time (or sojourn time) of jobs for instance. By Little’s law, this performance objective is equivalent to the mean number of jobs in the systems. Such a routing strategy is known as social optimum or social welfare since it minimizes the mean processing time of jobs (we will also talk of a global optimum). This load balancing problem is perhaps one of the most studied one in the operations research community, and many works have been devoted to the analysis of the optimal routing in various static and dynamic scenarios [10], [14], [20].

In practice, it may however happen that a single centralized dispatcher is simply not feasible due to scalability or complexity reasons. In this case, the system designer will certainly have to resort to a distributed scheme in which several dispatchers are used as shown in Figure 2. In this case, each dispatcher will independently seek to minimize the processing time perceived by the traffic it routes. Thus the shift from a centralized to a distributed scheme will give rise to a non-cooperative game between the dispatchers.

Game theory provides the systematic framework to study and understand such problems. We can distinguish two different settings depending on the number of dispatchers. If
the number of dispatchers is finite, then it is said that the game is “atomic” and a well-known equilibrium strategy is given by the so-called Nash equilibrium, that is, a routing strategy from which unilateral deviation does not help any dispatcher in improving the performance perceived by the traffic it routes. When the number of dispatcher grows to infinity (every arriving job is handled by a dispatcher and it takes its own routing decision) the game is referred as “non-atomic” and the corresponding equilibria is given by the notion of Wardrop Equilibrium. In this case, the equilibrium point is characterized by the fact that the performance in every (used) server is the same. In the present article we are mostly interested in the “atomic” setting, and we refer to Section II for related work in the “non-atomic” setting.

From the system’s designer perspective a very important question pertains to the loss of performance incurred when shifting to a decentralized architecture. Indeed, in the decentralized architecture each dispatcher performs an individual optimization for its own jobs and thus, it can be expected that the overall performance of the decentralized scheme will be worse than in the centralized scheme. The system designer is probably ready to accept a distributed routing scheme provided the gain in scalability is not achieved at the expense of a significant loss in performance. In this context, the question turns out to be: can we provide performance guarantees for these decentralized routing schemes? This is the main question addressed in this paper.

Our objectives are two fold. Firstly we investigate the properties of the non-cooperative game. We show that there exists always a unique Nash Equilibrium, that is, a routing strategy from which no dispatcher has any incentive to deviate. We also show that the worst Nash Equilibrium occurs when the amount of traffic that every dispatcher routes is exactly the same. To the best of our knowledge this property has not been shown previously, and it may find applications in other games. For this particular case, we show that the game belongs to a particular class of games known as Potential Games [17] which is known to have several desirable properties. For instance, for a Potential Game, the best response algorithm converges to the equilibrium. Secondly we compare the performance of the global optimum with that given by the Nash Equilibrium, or in other words, the performance when there is only one dispatcher which routes all the traffic, and the performance when there are several dispatchers each one seeking to optimize its own performance. In order to do so we look at the Price of Anarchy (PoA) which was introduced by Koutsoupias and Papadimitriou [16]. The PoA is a measure of the inefficiency of a decentralized scheme. It is defined as the ratio between the performance obtained by the worst Nash equilibrium and the global optimal solution. Thus, the PoA lies in the interval $[1, \infty)$. We show that the PoA is of the order of the square root of the number of dispatchers. This result indicates that when the number of dispatchers is finite, so is the loss of efficiency. However as the number of dispatchers increases, the loss of efficiency may grow unboundedly. Thus, we recover the result in [1] where it was shown that when the number of dispatchers is infinite (the “non-atomic” as pointed out above) the PoA is infinite.

The rest of the paper is organized as follows. In Section III, we describe the model, state the problem, present the main results of the paper. In Section IV we give details of the derivation of the bounds on the PoA. Finally, some conclusions are drawn and possible extensions are discussed in Section V.

Due to the lack of space, we omit most of the proofs, which are contained in [2].

II. RELATED WORK

Load balancing in multi-server systems has been widely studied in the literature. Global and Individual optimality in load balancing are considered in the monograph [14], which does not consider decisions based on knowledge of the amount of load. Systems with general service time distribution and FCFS scheduling discipline were studied in [7], [3], [4], [11], while [18], [13] studied systems with exponential service time distributions and arbitrary scheduling discipline. In [12] the authors analyzed a multi-server system where requests join the server that has the smallest number of requests. In a recent work [6] the authors investigate the performance of a server farm where the scheduling discipline in each server is SRPT (Shortest Remaining Processing Time First). In [9] the authors studied the performance of selfish routing in a server farm with a min-max objective, that is, when the objective is to minimize the maximum sojourn time in the servers.

In recent years the study of PoA in multi-server queues has started to receive attention. In [13] the authors considered the non-atomic scenario where every arriving job can select the server in which it will be served. An important assumption is that the holding cost is the same in every server. Building upon results from [3], it is shown in [13] that the PoA is upper bounded by the number of servers. We also refer to [21] for similar results. Another closely related work is [1]. The main difference between the models studied in [13] and [1] was that in the latter the holding costs in every server could be arbitrarily chosen. Using potential game theory, it is shown in [1] that the PoA is unbounded in the non-atomic setting, i.e., it can be arbitrarily close to infinity. This was a surprising result since it indicated that unequal holding costs may have a profound impact on the system’s performance.

Our present work is closely related to work by Orda and co-authors [15], [19]. In these references the atomic non-cooperative setting was studied, but the focus was on existence, uniqueness and the properties of the Nash equilibrium rather than on the PoA. Moreover, it was also assumed that the holding cost per unit of time is the same in every server, which as we have mentioned can have a profound impact on the performance. Several of the arguments used in the present work are directly inspired from those references, but we emphasize that our main results and characterizations are new.
III. PROBLEM FORMULATION AND MAIN RESULTS

We consider a non-cooperative routing game with $K$ dispatchers and $S$ Processor-Sharing servers. Denote $S = \{1, \ldots, S\}$ to be the set of servers and $C = \{1, \ldots, K\}$ to be the set of dispatchers. Jobs received by dispatcher $i = 1, \ldots, K$ are said to be jobs of class $i$.

Server $j \in S$ has capacity $r_j$ and a holding cost $c_j$ per unit time is incurred for each job sent to this server. It is assumed that servers are numbered in the order of increasing cost per unit capacity, i.e., $\frac{c_j}{r_j} \leq \frac{c_k}{r_k}$ if $m \leq n$. Let $r = (r_j)_{j \in S}$ and $c = (c_j)_{j \in S}$ denote the vectors of server capacities and server costs, respectively, and let $\mathbf{r} = \sum_{n \in S} r_n$ denote the total capacity of the system.

Jobs of class $i \in C$ arrive to the system according to a Poisson process and have generally distributed service times. We do not specify the arrival rate and the characteristics of the service times distribution due to the fact that in an $M/G/1$-PS queue the mean number of jobs depends on the arrival process and service time distribution only through the traffic intensity, i.e., the product of the arrival rate and the mean service time.

Let $\lambda_i$ be the traffic intensity of class $i$. It is assumed that $\lambda_i \leq \lambda_j$ for $i \leq j$. Moreover, it will also be assumed that the vector $\lambda$ of traffic intensities belongs to the following set :

$$\Lambda = \left\{ \lambda \in \mathbb{R}^K : \sum_{i \in C} \lambda_i = \bar{\lambda} \right\},$$

where $\bar{\lambda}$ denotes the total incoming traffic intensity. It will be assumed throughout the paper that $\lambda < \bar{\tau}$, which is the necessary and sufficient condition to guarantee the stability of the system.

Let $x_{i,j} = (x_{i,j})_{j \in S}$ denote the routing strategy of dispatcher $i$, with $x_{i,j}$ being the amount of traffic that sends towards server $j$. Let

$$X_i = \left\{ x_i \in \mathbb{R}^S : 0 \leq x_{i,j} \leq r_j, \forall j \in S; \sum_{j \in S} x_{i,j} = \lambda_i \right\}$$

denote the set feasible routing strategies for dispatcher $i$. The vector $x = (x_i)_{i \in C}$ will be called a multi-strategy. The multi-strategies belong to the product strategy space $\mathcal{X} = \bigotimes_{i \in C} X_i$.

Dispatcher $i$ seeks to find a routing strategy that minimizes the mean weighted sojourn times of its jobs, which, by Little’s law, is equivalent to minimizing the mean weighted number of jobs in the system as seen by this class. This optimization problem, which depends on the routing decisions of the other classes, can be formulated as follows :

$$\text{minimize } T_i(x) = \sum_{j \in S} c_j \frac{x_{i,j}}{r_j - y_j},$$

where $y_j = \sum_{k \in C} x_{k,j}$ is the traffic offered to server $j$.

A Nash equilibrium of the routing game is a multi-strategy from which no class finds it beneficial to unilaterally deviate. Hence, $x \in \mathcal{X}$ is a Nash Equilibrium Point (NEP) if

$$x_i = \arg \min_{x \in X_i} T_i(x), \quad i \in C.$$

We have assumed that $\bar{\lambda} < \bar{\tau}$. Since the cost function $T_i(x)$ of each user $i \in C$ satisfies the conditions defining a type-A cost function in [19], we can apply Theorem 2.1 in [19] and conclude to the existence of a unique NEP.

The global performance of the system can be assessed using the global cost

$$D_K(\lambda, r, c) = \sum_{i \in C} T_i(x) = \sum_{j \in S} c_j \frac{y_j}{r_j - y_j},$$

where the offered traffic $y_j$ are those at the NEP. The above cost represents the mean weighted number of jobs in the system. Note that when there is a single dispatcher, we have a single class whose traffic intensity is $\lambda_1 = \bar{\lambda}$. The global cost can therefore be written as $D_1(\bar{\lambda}, r, c)$ in this case.

We shall use the price of anarchy as a metric in order to assess the inefficiency of a decentralized scheme with $K$ dispatchers. For our problem, it is defined as

$$\text{PoA}(K) = \sup_{\lambda, r, c} \frac{D_K(\lambda, r, c)}{D_1(\lambda, r, c)}.$$

In the following section we present the two main results of this paper.

A. Main Results

The first theorem states that that the global cost $D_K(\lambda, r, c)$ achieves its maximum when $\lambda$ is the symmetric vector $\lambda^* = (\frac{\lambda}{K}, \ldots, \frac{\lambda}{K})$.

**Theorem 1:**

$$\sup_{\lambda, r, c} D_K(\lambda, r, c) = \sup_{r, c} D_K(\lambda^*, r, c).$$

This result states that, for the calculation of the PoA, we can restrict ourselves to the symmetric game. This, coupled with the fact that in our setting the symmetric game is also a potential game, makes it more tractable for the analytic computation of the NEP and the global cost, thereby greatly simplifying the derivation of the lower and upper bounds on the PoA.

The second theorem, which is proved in Section IV, gives these lower and upper bounds on the PoA.

**Theorem 2:** For a system with two or more servers, 

$$\frac{K}{2\sqrt{K} - 1} \leq \text{PoA}(K) \leq \sqrt{K}. $$

This result states that the PoA is of the order of $\sqrt{K}$ independently of the number of servers, and thus remains bounded for a finite number of dispatchers.
**Remark 1:** For a system with the only one server, \( \text{PoA}(K) = 1 \). Hence, we do not consider this case.

In the following section, we give the details of the proof of Theorem 2. The details of the proof of Theorem 1 are given in [2].

### IV. PRICE OF ANARCHY

According to Theorem 1, we have

\[
\text{PoA}(K) = \sup_{\lambda, r, c} \frac{D_K(\lambda, r, c)}{D_1(\lambda, r, c)} = \sup_{r, c} \frac{D_K(\lambda^\infty, r, c)}{D_1(\lambda, r, c)}. \tag{1}
\]

Therefore, in order to analyze the PoA, we can focus on the symmetric case. We analyze the symmetric game in Section IV-A and derive an explicit expression for the equilibrium flows. These results are then used in Section IV-B to prove that the PoA is upper-bounded by the square root of the number of dispatchers. In Section IV-C we prove the lower bound on the PoA by exhibiting an example for which the ratio \( \frac{D_K(\lambda, r, c)}{D_1(\lambda, r, c)} \) is \( K/(2\sqrt{K} - 1) \). Finally, in Section IV-D, we discuss the consequences of bounds on PoA.

#### A. Analysis of the Symmetric Game

It is well known that in this case the non-cooperative routing game is a potential game, i.e., the equilibrium flows are the global minima of a standard convex optimization problem (see e.g. Theorem 4.1 in [8]). This is formally stated in the following proposition.

**Proposition 1:** The multi-strategy \( x \) is a NEP of the symmetric game if and only the loads \( y_j = \sum_{i \in C} x_{i,j}, j \in S \), are the global optima of the following convex optimization problem:

\[
\begin{align*}
\text{minimize} & \quad \sum_{j \in S} c_j \left( \frac{y_j}{r_j - y_j} + (K - 1) \log \left( \frac{r_j}{r_j - y_j} \right) \right) \\
\text{subject to} & \quad \sum_{j \in S} y_j = \lambda, \\
& \quad 0 \leq y_j < r_j, \forall j \in S.
\end{align*}
\]

Note that when \( K = 1 \), the above problem reduces to the global optimization problem solved by the centralized scheme, whereas when \( K \to \infty \), the above problem reduces to the problem stated in Proposition 4 of [1]. In the latter case, the equivalent problem states the common function optimized jointly by an infinite number of players and is characteristic of the Wardrop equilibrium.

In order to describe the solution of the above equivalent problem, let us define \( u_j = c_j / r_j, j \in S \), and \( u_{S+1} = \infty \). Note that, by definition, the sequence \( u_j \) is increasing in \( j \). Let us also define the function \( W_j(K, z) = \sum_{i \in [u_j, u_{j+1}]} \frac{2e}{(K-1)^2 + 4K(z - (K-1))} \), and let \( W_j(K, z) = \sum_{j \in S} W_j(K, z) \).

The following proposition gives the solution of the potential game.

\[
\begin{align*}
\text{Proposition 2:} \quad & \text{The subset of servers that are used at the NEP is } S^*(K) = \{1, 2, \ldots, j^*(K)\}, \text{ where } j^*(K) \text{ is the greatest value of } j \text{ such that } W(K, u_{j+1}) \leq 0 < W(K, u_j). \\
& \text{The equilibrium flows are } x_{i,j} = \frac{y_j}{u_j}, i \in C, j \in S^*(K), \text{ where } y_j \text{ is the offered traffic of server } j. \\
& \text{Then, } y_j = r_j \sqrt{(K - 1)^2 + 4K\gamma(K)r_j/c_j - (K - 1)} / (K - 1) + 1, \tag{2}
\end{align*}
\]

with \( \gamma(K) \) the unique root of \( W(K, z) = 0 \) in \([u_1, \infty)\).

We now prove that the distributed scheme with \( K \) dispatchers uses only a subset of the servers used by the centralized scheme. The proof is based on the following proposition.

**Proposition 3:** The function \( \gamma(K) \) is decreasing in \( K \).

The fact that \( \gamma(K) \) is decreasing in \( K \) implies that \( j^*(K) \) is non-increasing in \( K \). We therefore have the following important corollary.

**Corollary 1:** For \( K \geq 1 \), \( S^*(K + 1) \subset S^*(K) \).

As an immediate consequence, we can conclude that \( S^*(K) \subset S^*(1) \), i.e., the distributed scheme with \( K \) dispatchers uses only a subset of the servers used by the centralized scheme.

#### B. Upper Bound on the PoA

In order to distinguish between the offered traffic in server \( j \) for different values of \( K \), we denote by \( y_j(K) \) the offered traffic in equilibrium in the \( K \) player symmetric game, where \( y_j(K) \) is given by (2).

The following lemma gives a bound on the mean number of jobs in a server in the decentralized case in terms of the mean number of jobs in the same server in the centralized case.

**Lemma 1:**

\[
\frac{y_j(K)}{r_j - y_j(K)} \leq \sqrt{K} \frac{y_j(1)}{r_j - y_j(1)}, \forall j \in S^*(1).
\]

The above lemma leads to the following upper bound on \( \text{PoA}(K) \).

**Proposition 4:**

\[
\text{PoA}(K) \leq \sqrt{K}.
\]

**Proof:** Since \( S^*(K) \subset S^*(1) \),

\[
D_K(\lambda^\infty, r, c) = \sum_{j \in S^*(K)} c_j \frac{y_j(K)}{r_j - y_j(K)} \leq \sum_{j \in S^*(1)} c_j \frac{y_j(1)}{r_j - y_j(1)},
\]

which, on substituting from Lemma 1, gives

\[
\frac{D_K(\lambda^\infty, r, c)}{D_1(\lambda, r, c)} \leq \sqrt{K}.
\]

Since this bound is independent of \( r \) and \( c \), we can conclude that \( \text{PoA}(K) \leq \sqrt{K} \).
C. Lower Bound on the PoA

We now give an example which shows that the PoA is bounded below by $K/(2\sqrt{K} - 1)$.

**Proposition 5:**

$$\text{PoA}(K) \geq \frac{K}{2\sqrt{K} - 1}.$$  

**Proof:** To prove this statement, we give a particular choice of the $r$ and $c$ for which $\frac{D_K(\lambda, r, c)}{D_1(\lambda, r, c)} = \frac{K}{2\sqrt{K} - 1}$, independently of the number of servers $S \geq 2$. It follows closely the example in Theorem 5 in [1]. We take $c_j = r_j = 1$, for $j > 1$. If verify that if

$$\frac{(r_1 - \lambda)^2}{r_1} < c_1 < \frac{(r_1 - \lambda)^2}{r_1 - \lambda + \frac{1}{K} \lambda} \quad (3)$$

then the centralized scheme will use all servers whereas, at the NEP, the distributed scheme with $K$ dispatchers will only use the first server. In order to ensure that (3) is always satisfied we set $c_1 = (r_1 - \lambda)^2 \alpha(r_1)$ for $\alpha(r_1)$ such that $r_1^{-1} < \alpha(r_1) < \left(\frac{r_1 - \lambda + \frac{1}{K} \lambda}{r_1 - \lambda}\right)^{-1}$. Note that $\frac{c_1}{r_1} < 1 = \frac{c_2}{r_2}$. Taking the limit as $r_1 \downarrow \lambda$, we get (the details are in [2])

$$\frac{D_K(\lambda^\infty, r, c)}{D_1(\lambda, r, c)} = \frac{\lambda \alpha(\lambda)}{2(\lambda \alpha(\lambda))^{1/2} - 1}.$$  

Note that the RHS in the above equation is increasing in $\lambda \alpha(\lambda)$, and that $\lambda \alpha(\lambda)$ has to be chosen in the interval $(1, K)$. Choosing the larger value, we obtain

$$\frac{D_K(\lambda^\infty, r, c)}{D_1(\lambda, r, c)} = \frac{K}{2\sqrt{K} - 1},$$

which proves the inequality (5).

D. Discussion on the PoA

We first note that the bounds on the PoA are valid for all values of $K$ and not just asymptotically. From these bounds, we can infer that the PoA grows as $\sqrt{K}$ as $K$ grows to infinity. Thus, the PoA can be made arbitrarily large in the limit $K \to \infty$, which is an alternative proof of Theorem 5 in [1] for the Wardrop equilibrium. In the other extreme case of $K = 1$, the bounds lead to $\text{PoA}(1) = 1$, which is consistent with the fact that the case $K = 1$ corresponds to the centralized setting.

We also observe that the PoA is independent of the number of servers — the bounds are valid as long as there are at least two servers. This result is in contrast to the corresponding one for the case when server costs are equal, for which the PoA was shown to be bounded by the number of servers ([13], [21]) in the non-atomic game. Thus, we infer that the inclusion of unequal server costs has a non-negligible impact on the PoA in the sense that, even in a system with two servers, the PoA can be of the order of $\sqrt{K}$.

V. Conclusions and Future Work

For the load balancing game considered in the paper, the worst global performance is obtained when all $K$ dispatchers route exactly the same amount of traffic. This result implies that the analysis of the PoA can be done by focusing on the symmetric case. For a system with two or more servers, the PoA is lower bounded by $K/(2\sqrt{K} - 1)$ and upper bounded by $\sqrt{K}$, independently of the number of servers.

We believe that this methodology can be generalized to other network topologies than the parallel link scenario considered in this paper. We therefore plan to investigate under which conditions the symmetry of traffic demands leads to a maximum global cost for general network topologies.

**References**


