Price of Anarchy in Non-Cooperative Load Balancing Games

U. Ayesta[1,3], O. Brun[2,4,*], B.J. Prabhu[2,4]

[1] *BCAM – Basque Center for Applied Mathematics, 48170 Derio, Spain*
[2] *CNRS ; LAAS ; 7 avenue du colonel Roche, F-31077 Toulouse, France*
[3] *IKERBASQUE, Basque Foundation for Science, 48011 Bilbao, Spain*
[4] *Université de Toulouse ; UPS, INSA, INP, ISAE ; LAAS ; F-31077 Toulouse, France*

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. Independently of the state of the servers, each dispatcher seeks to determine the routing strategy that optimizes the performance for its own traffic. The interaction of the various dispatchers thus gives rise to a non-cooperative game.

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.

Keywords: atomic games, load balancing, processor sharing, price of anarchy.

1. 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

Email addresses: ayesta@bcamath.org (U. Ayesta[1,3]), brun@laas.fr (O. Brun[2,4,*]), bjprabhu@laas.fr (B.J. Prabhu[2,4])

Corresponding author: O. Brun

The work of U. Ayesta is partially supported by grant MTM2010-17405 of the Ministry of Science and Innovation (MICINN), Spain, and grant PI2010-2 of the Department of Educación and Research, Basque Government.

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.

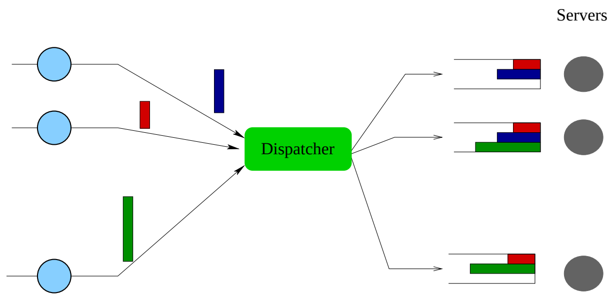

Figure 1: Centralized architecture for a server farm.

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 system. 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 [11, 16, 24].

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

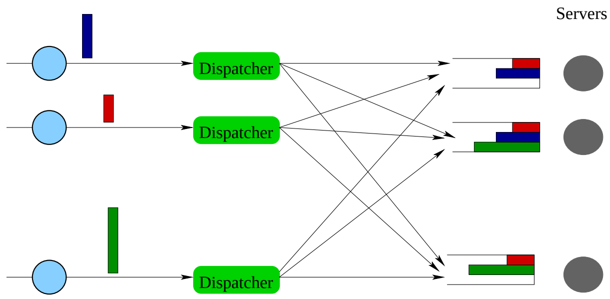

Figure 2: Decentralized architecture for a server farm.

arriving job is handled by a dispatcher and it takes its own routing decision) the game is referred as "non-atomic" and the corresponding equilibrium 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 2 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. We would like to emphasize that the problem we study finds applications not only in server-farms, but in fact in any distributed or parallel computer system where the workload must be balanced.

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 [21] 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 [19]. 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 3, we describe the model and state the problem. Section 4 explores the structure of the underlying Nash equilibria and prove their existence and uniqueness. It also establish several properties of these equilibria that form the foundation of the subsequent analysis. In Section 5, we analyze the global cost at the Nash equilibria and show that the maximum of this cost is achieved in the symmetric case. With this result at hand, we prove in Section 6 that the PoA is upper bounded by the square root of the number of dispatchers. Finally, some conclusions are drawn and possible extensions are discussed in Section 7.

2. 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 [16], 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, 13], while [22, 15] studied systems with exponential service time distributions and arbitrary scheduling discipline. In [14] 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 [10] 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 [15] the authors considered the 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 [15] that the PoA is upper bounded by the number of servers. We also refer to [25] for similar results. Another closely related work is [1]. The main difference between the models studied in [15] 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 [18, 23]. 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.

Kameda and co-authors have investigated a related load balancing problem in [12, 20, 11] where communication delays between servers are explicitly taken into account. In [12] they illustrate by simulations that Braess-like paradox occur, and that the worst-case degree of the paradox is obtained in the complete symmetric case.

3. Problem Formulation and Main Results

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

Server $j \in \mathcal{S}$ 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. $\frac{c_m}{r_m} \leq \frac{c_n}{r_n}$ if $m \leq n$. Let $\mathbf{r} = (r_j)_{j \in \mathcal{S}}$ and $\mathbf{c} = (c_j)_{j \in \mathcal{S}}$ denote the vectors of server capacities and server costs, respectively, and let $\overline{r} = \sum_{n \in \mathcal{S}} r_n$ denote the total capacity of the system.

Jobs of class $i \in \mathcal{C}$ 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 job size (see for example [17] or [8]).

Let λ_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 $\boldsymbol{\lambda}$ of traffic intensities belongs to the following set :

$$\Lambda = \left\{ \boldsymbol{\lambda} \in \mathbb{R}^K \ : \ \sum_{i \in \mathcal{C}} \lambda_i = \bar{\lambda}, \ \lambda_i \geq 0, \ i \in \mathcal{C} \right\},$$

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

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

$$\mathcal{X}_i = \left\{ \mathbf{x}_i \in \mathbb{R}^S \ : \ 0 \leq x_{i,j} \leq r_j \ , \forall j \in \mathcal{S}; \sum_{j \in \mathcal{S}} x_{i,j} = \lambda_i \right\}$$

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

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 :

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

where $y_j = \sum_{k \in \mathcal{C}} x_{k,j}$ is the traffic offered to server j. Note that, introducing $r_{i,j} = r_j - \sum_{k \neq i} x_{k,j}$, the available capacity of server j as seen by class i, the problem can alternatively be formulated as

$$\underset{\mathbf{x}_i \in \mathcal{X}_i}{\text{minimize}} \ \sum_{j \in \mathcal{S}} c_j \frac{x_{i,j}}{r_{i,j} - x_{i,j}}. \tag{1}$$

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

$$\mathbf{x}_i \in \arg\min_{\mathbf{z} \in \mathcal{X}_i} \ T_i(\mathbf{x}_1, \ldots, \mathbf{x}_{i-1}, \mathbf{z}, \mathbf{x}_{i+1}, \ldots, \mathbf{x}_K), \quad i \in \mathcal{C}.$$

Let $T_{ij}(\mathbf{x})$ denote the partial derivative of T_i with respect to $x_{i,j}$ at point $\mathbf{x}$:

$$T_{ij}(\mathbf{x}) = c_j \left[\frac{1}{r_j - y_j} + \frac{x_{i,j}}{(r_j - y_j)^2} \right]. \tag{2}$$

According to the Karush-Kuhn-Tucker optimality conditions, $\mathbf{x} \in \mathcal{X}$ is a NEP if and only if there exist multipliers μ_i such that

$$T_{ij}(\mathbf{x}) = \frac{c_j}{r_j - y_j} + \frac{c_j x_{i,j}}{(r_j - y_j)^2} \quad = \quad \mu_i \quad \text{if } x_{i,j} > 0, \tag{3}$$

$$T_{ij}(\mathbf{x}) = \frac{c_j}{r_j - y_j} \quad \geq \quad \mu_i \quad \text{if } x_{i,j} = 0. \tag{4}$$

For each server $j \in \mathcal{S}$, let $\mathcal{C}_j = \{i \in \mathcal{C} : x_{i,j} > 0\}$ be the set of classes which route traffic to server j. Similarly, let $\mathcal{S}_i = \{j \in \mathcal{S} : x_{i,j} > 0\}$ be the set of servers to which class i routes traffic. Note that $i \in \mathcal{C}_j \iff j \in \mathcal{S}_i$. We can now rewrite equations (3) and (4) as

$$\frac{c_j}{r_j - y_j} < \mu_i \iff i \in \mathcal{C}_j \iff j \in \mathcal{S}_i. \tag{5}$$

Let $\mathbf{x}$ be a NEP for the system with K dispatchers. The global performance of the system can be assessed using the global cost

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

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 $D_1(\bar{\lambda}, \mathbf{r}, \mathbf{c})$ in this case.

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

$$PoA(K) = \sup_{\boldsymbol{\lambda} \in \Lambda, \mathbf{r} \in \mathbb{R}^S, \mathbf{c} \in \mathbb{R}^S} \frac{D_K(\boldsymbol{\lambda}, \mathbf{r}, \mathbf{c})}{D_1(\bar{\lambda}, \mathbf{r}, \mathbf{c})}.$$

In the following section, we establish several important properties of the Nash equilibrium when the input parameters $\boldsymbol{\lambda}$, $\mathbf{r}$, and $\mathbf{c}$ are fixed. These properties will be used to prove the main results of this paper related to the PoA.

3.1. Main Results

Before getting into the technical details, we present here an overview of the most important results obtained in the paper. The first theorem, which is proved in Section 5, states that that the global cost $D_K(\boldsymbol{\lambda}, \mathbf{r}, \mathbf{c})$ achieves its maximum when $\boldsymbol{\lambda}$ is the symmetric vector $\boldsymbol{\lambda}^= = \left(\frac{\bar{\lambda}}{K}, \ldots, \frac{\bar{\lambda}}{K}\right)$.

Theorem 1.

$$\sup_{\boldsymbol{\lambda} \in \Lambda} D_K(\boldsymbol{\lambda}, \mathbf{r}, \mathbf{c}) = D_K(\boldsymbol{\lambda}^=, \mathbf{r}, \mathbf{c}), \quad \forall \mathbf{r} \in \mathbb{R}^S, \mathbf{c} \in \mathbb{R}^S.$$

To get some intuition behind Theorem 1 consider the particular case of two dispatchers. When one of the dispatchers routes all the traffic, the Nash Equilibrium coincides with the social optimum, and thus there is no inefficiency. The opposite extreme corresponds to the case when each dispatcher routes half of the traffic. In this case both dispatchers have the same "weight", and since they selfishly compete for resources, they end up operating in the most inefficient equilibrium.

Theorem 1 implies 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 6, gives these lower and upper bounds on the PoA.

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

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

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 only one server, $PoA(K) = 1$. Hence, we do not consider this case.*

4. Existence, Uniqueness and Monotonicity Properties of the NEP

In this section, we show the existence and uniqueness of the NEP and investigate properties of the traffic flow at this point. Some of these properties are generalization of those presented in [23] and [18] in the case of unit holding costs.

4.1. Existence and Uniqueness

We have assumed that $\overline{\lambda} < \overline{r}$. Since the cost function $T_i(\mathbf{x})$ of each user $i \in \mathcal{C}$ satisfies the conditions defining a type-A cost function in [23], we can apply Theorem 2.1 in [23] and conclude to the existence of a unique NEP.

4.2. Properties related to traffic intensities

We prove below that there is a monotonicity among classes in their use of servers: a class with a higher demand uses more of each and every server. Our main results are stated in Proposition 1 and Corollary 1.

Proposition 1. *The following statements are equivalent :*

1. $\mu_i < \mu_k$.
2. $\exists j \in \mathcal{S}_k : x_{i,j} < x_{k,j}$.
3. $x_{i,j} < x_{k,j}, \forall j \in \mathcal{S}_k$.
4. $\lambda_i < \lambda_k$.

PROOF. Please, see Appendix A.1. ■

The following corollary is an immediate consequence of Proposition 1.

Corollary 1. *The following statements hold :*

1. $\mu_i = \mu_k \iff x_{i,j} = x_{k,j}, \forall j \in \mathcal{S}_k \iff \lambda_i = \lambda_k$.
2. *If* $\lambda_i < \lambda_k$ *then* $\mathcal{S}_i \subset \mathcal{S}_k$.
3. *If* $\lambda_i = \lambda_k$ *then* $\mathcal{S}_i = \mathcal{S}_k$.

The above results show that a class with a higher demand uses more of each and every server. If two classes have the same traffic intensity, then they send the same amount of flow on each server. In particular, if all classes have the same demand, i.e. $\boldsymbol{\lambda} = \boldsymbol{\lambda}^{=}$, then, for all server $j \in \mathcal{S}$ and for all $i \in \mathcal{C}$ we have $x_{i,j} = y_j/K$.

Recall that we have assumed that $\lambda_i \leq \lambda_k$ for $i \leq k$. Therefore, according to the above results, if we consider two classes i and $k > i$, then we have $\mathcal{S}_i \subseteq \mathcal{S}_k$, $\mu_i \leq \mu_k$ and $x_{i,j} \leq x_{k,j}$ for all servers $j \in \mathcal{S}_k$, with the equalities holding if and only if $\lambda_i = \lambda_k$.

4.3. Properties related to server costs per unit capacity

The above results tell how an order on λ_i translates to an order on $x_{i,j}$, μ_i and $\mathcal{S}_i$, i.e., quantities of class i. We now give the analogous results for similar quantities of server j. Denote $\frac{r_j - y_j}{c_j} =: \kappa_j$. We can rewrite (5) as

$$\mu_i^{-1} < \kappa_j \iff i \in \mathcal{C}_j \iff j \in \mathcal{S}_i. \tag{6}$$

Note that κ_m is to server m what μ_i is to class i. The following proposition states another interesting monotonicity property regarding the order of preference of servers as seen by each class.

Proposition 2. *The following statements are equivalent :*

1. $\kappa_m < \kappa_n$.
2. $\exists i \in \mathcal{C}_n : \frac{x_{i,m}}{c_m} < \frac{x_{i,n}}{c_n}$.
3. $\frac{x_{i,m}}{c_m} < \frac{x_{i,n}}{c_n}, \forall i \in \mathcal{C}_n$.
4. $\frac{r_m}{c_m} < \frac{r_n}{c_n}$.

PROOF. Please, see Appendix A.2. ■

The following corollary is an immediate consequence of Proposition 2.

Corollary 2. *The following statements hold :*

1. $\kappa_m = \kappa_n \iff \frac{x_{i,m}}{c_m} = \frac{x_{i,n}}{c_n}, \forall i \in \mathcal{C}_n \iff \frac{r_m}{c_m} = \frac{r_n}{c_n}$.
2. *If* $\frac{r_m}{c_m} < \frac{r_n}{c_n}$ *then* $\mathcal{C}_m \subset \mathcal{C}_n$.
3. *If* $\frac{r_m}{c_m} = \frac{r_n}{c_n}$ *then* $\mathcal{C}_m = \mathcal{C}_n$.

The above corollary shows that we get a partition of classes among servers at the NEP : starting with a server m of maximal cost per unit capacity $\frac{c_m}{r_m}$ and moving towards servers n with lower cost per unit capacity $\frac{c_n}{r_n} < \frac{c_m}{r_m}$, we observe more and more classes joining the servers, i.e. $\mathcal{C}_m \subset \mathcal{C}_n$.

Recall that the servers were numbered in the following order: $c_1/r_1 \leq c_2/r_2 \leq \ldots \leq c_S/r_S$. According to the above properties, it implies that if we consider two servers n and $m > n$, then we have $\mathcal{C}_m \subseteq \mathcal{C}_n$, $\frac{r_n - y_n}{r_m - y_m} \geq \frac{c_n}{c_m}$ and $\frac{x_{i,n}}{x_{i,m}} \geq \frac{c_n}{c_m}$ for each class $i \in \mathcal{C}_n$, with the equalities holding if and only if $c_n/r_n = c_m/r_m$.

Before moving to the analysis of the set of servers used by each class at the equilibrium, we conclude this section with a last property related to the server costs per unit capacity. This technical result will play a key role when comparing the cost of two different equilibria.

Lemma 1.

$$\frac{c_j r_j}{(r_j - y_j)^2} \geq \frac{c_{j+1} r_{j+1}}{(r_{j+1} - y_{j+1})^2}, \ \forall j \in \mathcal{S},$$

with strict inequality if $\mathcal{C}_j \setminus \mathcal{C}_{j+1} \neq \emptyset$.

Proof. Please, see Appendix A.3. ■

The above lemma also leads to the following ordering on the ratio of the traffic offered to a server to its service capacity - also known as the load on that server.

Corollary 3.

$$\frac{y_j}{r_j} \geq \frac{y_{j+1}}{r_{j+1}}, \forall j.$$

Proof. We use the fact that $r_j/c_j \geq r_{j+1}/c_{j+1}$ together with Lemma 1 to deduce that

$$\left(\frac{r_j}{r_j - y_j}\right)^2 \geq \left(\frac{r_{j+1}}{r_{j+1} - y_{j+1}}\right)^2,$$

from which we can conclude the stated result. ■

4.4. Characterization of the set of servers used

The following proposition shows that the set of servers used by each class has the so-called water-filling structure. It is an extension of Proposition 1 in [18] to the case of unequal holding costs. The proof is based on the fact that dispatcher i solves the optimization problem (1). The proof closely parallels [18] and can be found in the technical report [2].

Proposition 3. *For each class $i \in \mathcal{C}$, there exist S_i such that the set $\mathcal{S}_i$ of servers used by class i is $\mathcal{S}_i = \{1, \ldots, S_i\}$. Moreover, the threshold S_i is such that $G_{i,S_i} < \lambda_i \leq G_{i,S_i+1}$, where*

$$G_{i,s} = \sum_{j=1}^{s-1} r_{i,j} - \sqrt{\frac{r_{i,s}}{c_s}} \sum_{j=1}^{s-1} \sqrt{c_j\, r_{i,j}} \quad s = 2, \ldots, S \tag{7}$$

with $G_{i,1} = 0$ and $G_{i,S+1} = \sum_{j \in \mathcal{S}} r_j - \overline{\lambda} + \lambda_i$. Note that $G_{i,S+1}$ is the system capacity as seen by class i jobs.

Remark 2. *In the special case $\lambda_i = G_{i,S_i+1}$, inequality (4) holds tight for $j = S_i + 1$. Therefore, in this case, we can define the set of servers used by class i as $\mathcal{S}_i = \{1, \ldots, S_i, S_i + 1\}$, where server $S_i + 1$ is "marginally" used, with $x_{i,S_i+1} = 0$.*

From Corollary 1, we can conclude that the thresholds $S_1, \ldots, S_K$ satisfy the order $S_1 \leq S_2 \leq \ldots \leq S_K$.

5. Analysis of the Global Cost

In this section, it will be assumed that the capacity vector $\mathbf{r}$ and the cost vector $\mathbf{c}$ are fixed. Our goal is to prove that the global cost $D_K(\boldsymbol{\lambda}, \mathbf{r}, \mathbf{c})$ achieves its maximum in the symmetric case, i.e. when $\boldsymbol{\lambda} = \boldsymbol{\lambda}^{=}$.

For each rate vector $\boldsymbol{\lambda} \in \Lambda$, we already know that there exists a unique NEP $\mathbf{x} \in \mathcal{X}$. Let us define the function $\mathcal{N} : \Lambda \to \mathcal{X}$ such that for each vector $\boldsymbol{\lambda} \in \Lambda$, $\mathcal{N}(\boldsymbol{\lambda}) \in \mathcal{X}$ is this unique NEP. In the sequel, the function $\mathcal{N}$ will be called the Nash mapping. We have the following result which is an adaption of Theorem 1 in [18](see [2] for the proof).

Theorem 3. *The Nash mapping $\mathcal{N}$ is a continuous function from Λ into $\mathcal{X}$.*

In order to prove that the global cost achieves its maximum in the symmetric case, we need to compare the equilibria $\mathcal{N}(\boldsymbol{\lambda})$ and $\mathcal{N}(\hat{\boldsymbol{\lambda}})$ that are induced by two different rate vectors $\boldsymbol{\lambda}$ and $\hat{\boldsymbol{\lambda}}$ in Λ. If the resulting equilibria are such that the set of servers over which each class sends its flow do not coincide at both equilibria, then the comparisons become extremely complex, if possible at all. To avoid this difficulty, we proceed as follows. In Section 5.1, we compare the equilibria induced by two different rate vectors $\boldsymbol{\lambda}$ and $\hat{\boldsymbol{\lambda}}$, assuming that (i) these equilibria are such that each class sends its flow to the same servers under both equilibria, and (ii) $\hat{\boldsymbol{\lambda}}$ is obtained from $\boldsymbol{\lambda}$ through a basic transformation (see below). In Section 5.2 we exploit the continuity of the Nash mapping to show that the global cost increases under this transformation even when the set of servers is different at the two equilibria. Finally, in Section 5.3, we show that the symmetric rate vector $\boldsymbol{\lambda}^{=}$ can be obtained from any rate vector $\boldsymbol{\lambda}$ with a finite number of such transformations.

5.1. Basic Transformation of a Rate Vector

For each rate vector $\lambda \in \Lambda$, recall that by convention $\lambda_1 = \min_{i\in\mathcal{C}} \lambda_i$ and $\lambda_K = \max_{i\in\mathcal{C}} \lambda_i$. Define the sets $\mathcal{C}_{min}$ and $\mathcal{C}_{max}$ as follows :

$$\mathcal{C}_{min} = \{i \in \mathcal{C} \;:\; \lambda_i = \lambda_1\} \quad \text{and} \quad \mathcal{C}_{max} = \{i \in \mathcal{C} \;:\; \lambda_i = \lambda_K\},$$

and let $n_{min} = |\mathcal{C}_{min}|$ and $n_{max} = |\mathcal{C}_{max}|$.

Definition 1. *For each vector $\boldsymbol{\lambda} \in \Lambda$, define the function $h_{\boldsymbol{\lambda}} : [0, n_{max}\,\lambda_K] \to \Lambda$ as follows :*

$$h_{\boldsymbol{\lambda}}(\epsilon) = \boldsymbol{\lambda} + \epsilon \left(\frac{1}{n_{min}} \sum_{i\in\mathcal{C}_{min}} \mathbf{e}_i \;-\; \frac{1}{n_{max}} \sum_{i\in\mathcal{C}_{max}} \mathbf{e}_i \right), \tag{8}$$

where $\mathbf{e}_i$ denotes the vector in $\mathbb{R}^K$ with the i-th component equal to 1 and all other components equal to 0. A rate vector $\hat{\boldsymbol{\lambda}} \in \Lambda$ is said to be obtained from $\boldsymbol{\lambda}$ under a basic transformation if and only if there exist $\epsilon \in [0, n_{max}\lambda_K]$ such that $\hat{\boldsymbol{\lambda}} = h_{\boldsymbol{\lambda}}(\epsilon)$. In this case, ϵ is called the step of the transformation.

Note that the above transformation increases the traffic of classes $i \in \mathcal{C}_{min}$ (the classes with the smallest amount of traffic) and decrease correspondingly the traffic of classes $i \in \mathcal{C}_{max}$ (the classes with the largest amount of traffic).

In the following, we will compare two rate vectors $\boldsymbol{\lambda}$ and $\hat{\boldsymbol{\lambda}}$. If z is a certain quantity related to the Nash equilibrium induced by the vector $\boldsymbol{\lambda}$ then we shall denote the corresponding quantity for vector $\hat{\boldsymbol{\lambda}}$ by $\hat{z}$. The comparison of equilibria carried out in this section is done under the following assumption.

Assumption 1. *The rate vectors $\boldsymbol{\lambda} \in \Lambda$ and $\hat{\boldsymbol{\lambda}} \in \Lambda$ are such that:*

1. *$\hat{\boldsymbol{\lambda}}$ is obtained from $\boldsymbol{\lambda}$ under a basic transformation,*
2. *$\mathcal{C}_j = \hat{\mathcal{C}}_j$, $\forall j \in \mathcal{S}$.*

In other words, we assume that the transformation of $\boldsymbol{\lambda}$ into $\hat{\boldsymbol{\lambda}}$ leaves unaffected the set of servers used by each class.

The key point here is that in order to determine the impact of a basic transformation of the rate vector $\boldsymbol{\lambda}$ on the global cost, we need to compare the server loads under the equilibria $\mathbf{x} = \mathcal{N}(\boldsymbol{\lambda})$ and $\hat{\mathbf{x}} = \mathcal{N}(\hat{\boldsymbol{\lambda}})$. To this end, let us define the sets $\mathcal{S}^+$ and $\mathcal{S}^-$ as follows :

$$\mathcal{S}^+ = \{j \in \mathcal{S} \ : \ \hat{y}_j > y_j\} \quad \text{and} \quad \mathcal{S}^- = \mathcal{S} \setminus \mathcal{S}^+,$$

i.e., $\mathcal{S}^+$ is the set of servers whose load increases under the transformation while $\mathcal{S}^-$ is the set of servers whose load is non-increasing under the transformation. Note that $\mathcal{S}^- \neq \emptyset$ due to flow conservation. Note also that $\mathcal{S}^+$ is empty if and only if the load of each and every server is constant under the transformation.

We now state several results regarding the impact of the transformation on server loads. In Proposition 4, we give two properties of the set $\mathcal{S}^+$. We show that (i) if there exists at least one server whose load increases under the transformation then the load of each and every server used by class 1 increases, and that (ii) the load of all servers is non-increasing under the transformation if and only if all traffic classes use the same set of servers.

Proposition 4. *The following statement hold :*

1. *If* $\mathcal{S}^+ \neq \emptyset$ *then* $\mathcal{S}_1 \subset \mathcal{S}^+$*,*
2. $\mathcal{S}^+ = \emptyset \iff \mathcal{S}_1 = \mathcal{S}_K$*.*

Proof. Please see Appendix B.2. ■

As a direct consequence of the above proposition, we get the following corollary that tells us that if at equilibria $\mathbf{x}$ and $\hat{\mathbf{x}}$ all classes use the same set of servers, then the server loads are constant under the transformation.

Corollary 4. $y_j = \hat{y}_j, \forall j \in \mathcal{S} \iff \mathcal{S}_1 = \mathcal{S}_K$*.*

We now turn our attention to the set $\mathcal{S}^-$.

Proposition 5. *For all* $j \in \mathcal{S}$*, if* $j \in \mathcal{S}^-$ *then* $j+1 \in \mathcal{S}^-$*.*

Proof. Please see Appendix B.3. ■

Proposition 5 says that the transformation induces a monotonic partition of servers: there exists a threshold $J < S$ such that for all servers $j > J$ the load is non-increasing under the transformation.

Using the above results regarding the impact of the transformation on the server loads, the following two theorems compare the costs $D(\boldsymbol{\lambda})$ and $D(\hat{\boldsymbol{\lambda}})$. The first theorem uses the following two lemmata.

Lemma 2. *If* $\mathcal{C}_m = \mathcal{C}_n$*, then* $\hat{y}_m \geq y_m \iff \hat{y}_n \geq y_n$*.*

Proof. From Lemma 17 in Appendix B.1, $\hat{y}_m \geq y_m$ is equivalent to $\sum_{i \in \mathcal{S}_m} \hat{\mu}_i \geq \sum_{i \in \mathcal{S}_m} \mu_i$, which, since $\mathcal{C}_m = \mathcal{C}_n$, is equivalent to $\sum_{i \in \mathcal{S}_n} \hat{\mu}_i \geq \sum_{i \in \mathcal{S}_n} \mu_i$. Again, from Lemma 17, we can conclude that $\hat{y}_n \geq y_n$. ■

Corollary 5. *For* $m, n \in \mathcal{S}_1$*,* $\hat{y}_m > y_m \iff \hat{y}_n > y_n$*.*

Lemma 3. *If b_i, $i = 1, 2, ...$ is such that*

- $b_1 > 0$*;* $b_i \leq 0 \Rightarrow b_{i+1} \leq 0$*; and* $\sum_i b_i = 0$,

and a_i, $i = 1, 2, ...,$ is such that

- $a_i \geq a_{i+1}$*; and* $a_I - a_{I+1} > 0$*, where* $I = \max\{i : b_i > 0\}$,

then $\sum_i a_i b_i > 0$.

PROOF. Please see Appendix B.4. ■

We are now in position to state our main results.

Theorem 4. $D(\boldsymbol{\lambda}) < D(\hat{\boldsymbol{\lambda}}) \iff \mathcal{S}_1 \subsetneq \mathcal{S}_K$.

PROOF. We first show that if $\mathcal{S}_1 \subsetneq \mathcal{S}_K$ then $D(\boldsymbol{\lambda}) < D(\hat{\boldsymbol{\lambda}})$. As a function of the loads, the global cost is given by

$$D(\boldsymbol{\lambda}) = \sum_{j \in \mathcal{S}} \frac{c_j r_j}{r_j - y_j} - \sum_{j \in \mathcal{S}} c_j. \tag{9}$$

Let $\Delta y_j = \hat{y}_j - y_j$. Note that $\Delta y_j > 0 \iff (r_j - \hat{y}_j)^{-1} > (r_j - y_j)^{-1}$, which leads to $\Delta y_j \neq 0 \iff \Delta y_j (r_j - \hat{y}_j)^{-1} > \Delta y_j (r_j - y_j)^{-1}$. Thus,

$$D(\hat{\boldsymbol{\lambda}}) - D(\boldsymbol{\lambda}) = \sum_{j \in \mathcal{S}} \frac{c_j r_j (\hat{y}_j - y_j)}{(r_j - \hat{y}_j)(r_j - y_j)} \geq \sum_{j \in \mathcal{S}} \frac{c_j r_j}{(r_j - y_j)^2} \Delta y_j. \tag{10}$$

We now show that the RHS in the above inequality is positive. Since $\mathcal{S}_1 \subsetneq \mathcal{S}_K$, from Proposition 4, we can infer that $\mathcal{S}^+ \neq \emptyset$ and $\mathcal{S}^- \neq \emptyset$. From Proposition 4, we can also infer that $\mathcal{S}_1 \subset \mathcal{S}^+$. Hence, $\Delta y_1 > 0$. From Proposition 5, if $j \in \mathcal{S}^-$ then $j + 1 \in \mathcal{S}^-$. Therefore, the sequence $\Delta y_j, j \in \mathcal{S}$ is such that

$$\Delta y_1 > 0, \quad \Delta y_j \leq 0 \Rightarrow \Delta y_{j+1} \leq 0 \quad \text{and} \quad \sum_{j \in \mathcal{S}} \Delta y_j = 0$$

Let $J = \max\{j : j \in \mathcal{S}^+\}$. Then, $J + 1 = \min\{j : j \in \mathcal{S}^-\}$. Note that $\mathcal{C}_J \neq \mathcal{C}_{J+1}$, otherwise from Lemma 2, either both J and $J + 1$ belong to $\mathcal{S}^+$ or both belong to $\mathcal{S}^-$. From Lemma 1, we can conclude that

$$\frac{c_j r_j}{(r_j - y_j)^2} \geq \frac{c_{j+1} r_{j+1}}{(r_{j+1} - y_{j+1})^2}, \ \forall j \quad \text{and} \quad \frac{c_J r_J}{(r_J - y_J)^2} > \frac{c_{J+1} r_{J+1}}{(r_{J+1} - y_{J+1})^2}$$

Since the sequences $c_j r_j / ((r_j - y_j)^2$ and Δy_j satisfy the conditions of Lemma 3, we can conclude that the RHS of (10) is strictly positive, and thus that $D(\boldsymbol{\lambda}) < D(\hat{\boldsymbol{\lambda}})$.
To show the converse, if $D(\boldsymbol{\lambda}) < D(\hat{\boldsymbol{\lambda}})$ then necessarily there exists an m such that $y_m \neq \hat{y}_m$. From Proposition 4, we obtain $\mathcal{S}_1 \neq \mathcal{S}_K$. Since $\mathcal{S}_1 \subset \mathcal{S}_K$, we can conclude that $\mathcal{S}_1 \subsetneq \mathcal{S}_K$. ■

Theorem 4 shows that if all the classes do not use the same set of servers at the equilibrium $\mathbf{x}$, then the transformation will strictly increase the cost. The following theorem proves that the cost is constant under the transformation if all classes use the same set of servers.

Theorem 5. $D(\boldsymbol{\lambda}) = D(\hat{\boldsymbol{\lambda}}) \iff \mathcal{S}_1 = \mathcal{S}_K$.

PROOF. From Proposition 4, if $\mathcal{S}_1 = \mathcal{S}_K$ then $y_j = \hat{y}_j, \forall j \in \mathcal{S}$ and therefore, $D(\boldsymbol{\lambda}) = D(\hat{\boldsymbol{\lambda}})$. To prove the inverse, if $\mathcal{S}_1 \neq \mathcal{S}_K$ then necessarily $\mathcal{S}_1 \subsetneq \mathcal{S}_K$. From Theorem 4, we can conclude that $D(\boldsymbol{\lambda}) \neq D(\hat{\boldsymbol{\lambda}})$. ■

5.2. Maximum Step of a Basic Transformation

Theorems 4 and 5 enable the comparison of the equilibria induced by two different rate vectors $\boldsymbol{\lambda}$ and $\hat{\boldsymbol{\lambda}}$, provided that $\hat{\boldsymbol{\lambda}}$ can be obtained from $\boldsymbol{\lambda}$ under a basic transformation which leaves unaffected the set of servers used by each class. The main limitation of these results comes from the latter assumption. However, as will be shown below, the continuity of the Nash mapping can be exploited to prove that, under certain conditions, the global cost is non-decreasing under the transformation even if some classes change the set of servers they use.

Definition 2. *For each rate vector $\boldsymbol{\lambda} \in \Lambda$, the maximum step of the transformation $h_{\boldsymbol{\lambda}}$ is*

$$\Delta = \min\left(n_{min}\,\Delta_{min}, n_{max}\,\Delta_{max}\right), \tag{11}$$

where $\Delta_{min} = -\lambda_1 + \min\left(\frac{\bar{\lambda}}{K}, \min_{i \in \mathcal{C}\setminus\mathcal{C}_{min}} \lambda_i\right)$ *and* $\Delta_{max} = \lambda_K - \max\left(\frac{\bar{\lambda}}{K}, \max_{i \in \mathcal{C}\setminus\mathcal{C}_{max}} \lambda_i\right)$.

For each rate vector $\boldsymbol{\lambda}$, let $\boldsymbol{\lambda}(\epsilon) = h_{\boldsymbol{\lambda}}(\epsilon)$ for $\epsilon \in [0, \Delta]$. All quantities of interest can be treated as functions of ϵ. Therefore, in the following, if z is a certain quantity related to the Nash equilibrium induced by the vector $\boldsymbol{\lambda}$ then we shall denote the corresponding quantity for vector $\boldsymbol{\lambda}(\epsilon)$ by $z(\epsilon)$.

The following lemma details how the sets $\mathcal{C}_{min}$ and $\mathcal{C}_{max}$ evolve under the transformation. The proof is tedious and long, but mathematically not involved, and can be found in the technical report [2].

Lemma 4. *For each $\epsilon \leq \Delta$,*

1. $\epsilon < \Delta \implies \mathcal{C}_{min} = \mathcal{C}_{min}(\epsilon)$ *and* $\mathcal{C}_{max} = \mathcal{C}_{max}(\epsilon)$,
2. $\mathcal{C}_{min} \cup \mathcal{C}_{max} \neq \mathcal{C} \implies \mathcal{C}_{min} \cup \mathcal{C}_{max} \subsetneq \mathcal{C}_{min}(\Delta) \cup \mathcal{C}_{max}(\Delta)$,
3. $\mathcal{C}_{min} \cup \mathcal{C}_{max} = \mathcal{C} \implies \boldsymbol{\lambda}(\Delta) = \boldsymbol{\lambda}^{=}$.

In other words, if the step ϵ of a basic transformation is lower than the maximum step Δ, then the sets $\mathcal{C}_{min}$ and $\mathcal{C}_{max}$ will be unaffected by the transformation. On the contrary, if $\epsilon = \Delta$, then, after the transformation, we will have either (i) one more class in the set $\mathcal{C}_{min}$ or $\mathcal{C}_{max}$, or (ii) $\boldsymbol{\lambda} = \boldsymbol{\lambda}^{=}$.

The following proposition states that if we consider two rate vectors obtained from $\boldsymbol{\lambda}$ under a basic transformation of steps lower than the maximum step, then one can be obtained from the other by a basic transformation.

Proposition 6. *Let $\epsilon_1, \epsilon_2 \in [0, \Delta]$, $\epsilon_1 < \epsilon_2$. Then $\boldsymbol{\lambda}(\epsilon_2)$ can be obtained from $\boldsymbol{\lambda}(\epsilon_1)$ under a basic transformation.*

Proof. Since $\epsilon_1 < \epsilon_2$ implies $\epsilon_1 < \Delta$, we have $\mathcal{C}_{min}(\epsilon_1) = \mathcal{C}_{min}$ and $\mathcal{C}_{max}(\epsilon_1) = \mathcal{C}_{max}$. Accordingly, $\boldsymbol{\lambda}(\epsilon_2)$ can be written as

$$h_{\boldsymbol{\lambda}}(\epsilon_1) + (\epsilon_2 - \epsilon_1) \left(\sum_{i \in \mathcal{C}_{min}(\epsilon_1)} \frac{\mathbf{e}_i}{n_{min}(\epsilon_1)} - \sum_{i \in \mathcal{C}_{max}(\epsilon_1)} \frac{\mathbf{e}_i}{n_{max}(\epsilon_1)} \right),$$

i.e., $\boldsymbol{\lambda}(\epsilon_2) = h_{\boldsymbol{\lambda}(\epsilon_1)}(\epsilon_2 - \epsilon_1)$. ■

We now show that even if some classes change the set of servers they use, the global cost is non-decreasing under the transformation $\boldsymbol{\lambda}(\epsilon) = h_{\boldsymbol{\lambda}}(\epsilon)$ provided that $\epsilon \leq \Delta$. The proof closely parallels the discussion in Section III.B of [18].

Theorem 6. *For $\epsilon \leq \Delta$, $D(\boldsymbol{\lambda}(\epsilon)) \geq D(\boldsymbol{\lambda})$.*

Proof. To prove the result, it suffices to show that D is a non-decreasing function of ϵ on $[0, \Delta]$. Let $A_{i,j} = \{\epsilon \in [0, \Delta] \ : \ G_{i,j}(\epsilon) \leq \lambda_i(\epsilon) \leq G_{i,j+1}(\epsilon)\}$, denote the set of $\epsilon \in [0, \Delta]$ for which class i sends flow to servers $\{1, \ldots, j\}$ under equilibrium $\mathcal{N}(\boldsymbol{\lambda}(\epsilon))$. Continuity of the Nash mapping then implies that the functions $G_{i,j}(\epsilon)$ and $\lambda_i(\epsilon)$ are continuous on $\epsilon \in [0, \Delta]$. Hence, $A_{i,j}$ is a closed set.

For each $\mathbf{S} \in \mathcal{S}^K$, define $A_{\mathbf{s}} = \cap_{i \in \mathcal{C}} A_{i,S_i}$, which is also a closed set. If $\epsilon_1, \epsilon_2 \in A_{\mathbf{S}}$, then each class sends its flow to the same set of servers under equilibria $\mathcal{N}(\boldsymbol{\lambda}(\epsilon_1))$ and $\mathcal{N}(\boldsymbol{\lambda}(\epsilon_2))$. Consider a vector $\mathbf{S} \in \mathcal{S}^K$ and assume that we can find $\epsilon_1, \epsilon_2 \in A_{\mathbf{S}}$ such that $\epsilon_1 < \epsilon_2$. According to Proposition 6, the vector $\boldsymbol{\lambda}(\epsilon_2)$ can be obtained from $\boldsymbol{\lambda}(\epsilon_1)$ under a basic transformation. Since $\epsilon_1, \epsilon_2 \in A_{\mathbf{S}}$, we can conclude using Theorems 4 and 5 that $D(\epsilon_2) \geq D(\epsilon_1)$.

Since $[0, \Delta] = \cup_{\mathbf{S} \in \mathcal{S}^K} A_{\mathbf{S}}$, all conditions of Theorem 5 in [18] are fulfilled, and we can conclude that D is a non-decreasing function of ϵ on $[0, \Delta]$. ■

5.3. Maximum of the Global Cost

The purpose of this section is to prove that the global cost achieves its maximum in the symmetric case, i.e., when $\boldsymbol{\lambda} = \boldsymbol{\lambda}^{=} = \left(\frac{\bar{\lambda}}{K}, \ldots, \frac{\bar{\lambda}}{K}\right)$. To this end, starting from a fixed rate vector $\boldsymbol{\lambda}$, we build a sequence $\left(\boldsymbol{\lambda}^k\right)_{k \in \mathbb{N}}$ of rate vectors such that $\boldsymbol{\lambda}^0 = \boldsymbol{\lambda}$ and $\boldsymbol{\lambda}^{k+1}$ is obtained from $\boldsymbol{\lambda}^k$ under a basic transformation of maximum step, i.e., $\boldsymbol{\lambda}^{k+1} = h_{\boldsymbol{\lambda}^k}(\Delta^k)$. The following proposition shows that the sequence $\left(\boldsymbol{\lambda}^k\right)_{k \in \mathbb{N}}$ converges to $\boldsymbol{\lambda}^{=}$ in a finite number of steps.

Proposition 7. *The sequence* $\left(\lambda^k\right)_{k\in\mathbb{N}}$ *converges to* $\lambda^=$ *in at most* K *steps.*

Proof. Let w^k be the number of classes in $\mathcal{C}^k_{min} \cup \mathcal{C}^k_{max}$. Note that $w^0 \geq 2$. According to Lemma 4.3, if $w^k = K$, then $\lambda^{k+1} = \lambda^=$. Otherwise, according to Lemma 4.2, we have $\mathcal{C}^k_{min} \cup \mathcal{C}^k_{max} \subsetneq \mathcal{C}^{k+1}_{min} \cup \mathcal{C}^{k+1}_{max}$, and thus $w^k < w^{k+1} \leq K$. This structure implies that in at most K steps we have $w^k = K$, and thus $\boldsymbol{\lambda}^{k+1} = \boldsymbol{\lambda}^=$. ■

We now prove Theorem 1.

Proof of Theorem 1. For each $\boldsymbol{\lambda} \in \Lambda$, the sequence $\left(\boldsymbol{\lambda}^k\right)_{k\in\mathbb{N}}$ converges to $\boldsymbol{\lambda}^=$ in a finite number of steps. According to Theorem 6, we have $D(\boldsymbol{\lambda}^{k+1}) \geq D(\boldsymbol{\lambda}^k)$. This implies that $D(\boldsymbol{\lambda}^=) \geq D(\boldsymbol{\lambda})$. ■

6. Price of Anarchy

According to Theorem 1, we have

$$PoA(K) = \sup_{\boldsymbol{\lambda},\mathbf{r},\mathbf{c}} \frac{D_K(\boldsymbol{\lambda},\mathbf{r},\mathbf{c})}{D_1(\bar{\lambda},\mathbf{r},\mathbf{c})} = \sup_{\mathbf{r},\mathbf{c}} \frac{D_K(\boldsymbol{\lambda}^=,\mathbf{r},\mathbf{c})}{D_1(\bar{\lambda},\mathbf{r},\mathbf{c})}. \tag{12}$$

Therefore, in order to analyze the PoA, we can focus on the symmetric case. We analyze the symmetric game in Section 6.1 and derive an explicit expression for the equilibrium flows. These results are then used in Section 6.2 to prove that the PoA is upper-bounded by the square root of the number of dispatchers. In Section 6.3 we prove the lower bound on the PoA by exhibiting an example for which the ratio $\frac{D_K(\boldsymbol{\lambda},\mathbf{r},\mathbf{c})}{D_1(\bar{\lambda},\mathbf{r},\mathbf{c})}$ is $K/(2\sqrt{K}-1)$. Finally, in Section 6.4, we summarize our result on the PoA and discuss its consequences.

6.1. 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 [9]). This is formally stated in the following proposition.

Proposition 8. *The multi-strategy* $\mathbf{x}$ *is a NEP of the symmetric game if and only the loads* $y_j = \sum_{i\in\mathcal{C}} x_{i,j}$, $j \in \mathcal{S}$, *are the global optima of the following convex optimization problem:*

$$\begin{aligned}
&\underset{\mathbf{y}}{\text{minimize}} && \sum_{j\in\mathcal{S}} \frac{c_j}{K}\left[\frac{y_j}{r_j - y_j} + (K-1)\log\left(\frac{r_j}{r_j - y_j}\right)\right] \\
&s.t. && \\
& && \textstyle\sum_{j\in\mathcal{S}} y_j = \bar{\lambda}, \\
& && 0 \leq y_j < r_j,\ \forall j \in \mathcal{S}.
\end{aligned}$$

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 $u_j = c_j/r_j$, $j \in \mathcal{S}$, and $u_{S+1} = \infty$. Note that, by definition, the sequence u_j is increasing in j. Let us also define the function

$$W_j(K,z) = \mathbb{1}_{\{z\in[u_j,u_{j+1})\}} \cdot \left(\sum_{s=1}^{j} \frac{2r_s}{\sqrt{(K-1)^2+4Ku_s^{-1}z} - (K-1)} - \sum_{s=1}^{j} r_s + \bar{\lambda} \right),$$

and let $W(K,z) = \sum_{j\in\mathcal{S}} W_j(K,z)$.

The following lemma states some properties of the function $W(K,z)$.

Lemma 5. *The function $W(K,z)$ is such that:*

1. *for a fixed K, the function $W : [u_1,\infty) \to \mathbb{R}$ is continuous and decreasing in z,*
2. *for a fixed z, $W(K,z)$ is decreasing in K,*
3. *for a fixed K, $W(K,z) = 0$ has a unique solution in the interval (u_1,∞).*

Proof. Please see Appendix C.1. ■

The following proposition gives the solution of the potential game.

Proposition 9. *The subset of servers that are used at the NEP is $\mathcal{S}^*(K) = \{1,2,\dots,j^*(K)\}$, where $j^*(K)$ is the greatest value of j such that $W(K,u_{j+1}) \leq 0 < W(K,u_j)$. The equilibrium flows are $x_{i,j} = \frac{y_j}{K}$, $i \in \mathcal{C}, j \in \mathcal{S}^*(K)$, where the offered traffic of server j is given by*

$$y_j = r_j \, \frac{\sqrt{(K-1)^2 + 4K\gamma(K)r_j/c_j} - (K+1)}{\sqrt{(K-1)^2 + 4K\gamma(K)r_j/c_j} - (K-1)}, \tag{13}$$

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

Proof. Please see Appendix C.2. ■

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 10. *The function $\gamma(K)$ is decreasing in K.*

PROOF. For $K_1 < K_2$, we have $0 = W(K_1, \gamma(K_1)) > W(K_2, \gamma(K_1))$, where the inequality follows from Lemma 5.2. Using $W(K_2, u_1) = \bar{\lambda} > 0$, and Lemma 5.3, we can conclude that $u_1 < \gamma(K_2) < \gamma(K_1)$. ■

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 6. *For $K \geq 1$, $\mathcal{S}^*(K+1) \subset \mathcal{S}^*(K)$.*

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

6.2. 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 (13).

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 6.

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

PROOF. Please see Appendix D. ■

The above lemma leads to the following upper bound on $PoA(K)$.

Proposition 11.

$$PoA(K) \leq \sqrt{K}.$$

PROOF. Since $\mathcal{S}^*(K) \subset \mathcal{S}^*(1)$,

$$D_K(\boldsymbol{\lambda}^=, \mathbf{r}, \mathbf{c}) = \sum_{j \in \mathcal{S}^*(K)} c_j \frac{y_j(K)}{r_j - y_j(K)} \leq \sum_{j \in \mathcal{S}^*(1)} c_j \frac{y_j(K)}{r_j - y_j(K)}.$$

which, on substituting from Lemma 6, gives

$$\frac{D_K(\boldsymbol{\lambda}^=, \mathbf{r}, \mathbf{c})}{D_1(\bar{\lambda}, \mathbf{r}, \mathbf{c})} \leq \sqrt{K}.$$

Since this bound in independent of $\mathbf{r}$ and $\mathbf{c}$, we can conclude that $PoA(K) \leq \sqrt{K}$. ■

6.3. 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 12.

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

PROOF. To prove this statement, we give a particular choice of the $\mathbf{r}$ and $\mathbf{c}$ for which $\frac{D_K(\boldsymbol{\lambda},\mathbf{r},\mathbf{c})}{D_1(\bar{\lambda},\mathbf{r},\mathbf{c})} = \frac{K}{2\sqrt{K}-1}$, 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 $j > 1$. Using Proposition 3, one can verify that if

$$\frac{(r_1-\bar{\lambda})^2}{r_1} < c_1 < \frac{(r_1-\bar{\lambda})^2}{r_1-\bar{\lambda}+\frac{1}{K}\bar{\lambda}} \tag{14}$$

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 (14) is always satisfied we set $c_1 = (r_1-\bar{\lambda})^2\,\alpha(r_1)$ for $\alpha(r_1)$ such that $r_1^{-1} < \alpha(r_1) < \left(r_1-\bar{\lambda}+\frac{\bar{\lambda}}{K}\right)^{-1}$. Note that $\frac{c_1}{r_1} < 1 = \frac{c_2}{r_2}$. Taking the limit as $r_1 \downarrow \bar{\lambda}$, we get (the details are in [2])

$$\frac{D_K(\boldsymbol{\lambda}^{=},\mathbf{r},\mathbf{c})}{D_1(\bar{\lambda},\mathbf{r},\mathbf{c})} = \frac{\bar{\lambda}\alpha(\bar{\lambda})}{2\left(\bar{\lambda}\alpha(\bar{\lambda})\right)^{1/2}-1}.$$

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

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

which proves the inequality (12). ■

6.4. Discussion on the PoA

We first give the proof of Theorem 2.

PROOF OF THEOREM 2. From Propositions 11 and 12 we can conclude that

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

■

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 $PoA(1) = 1$, 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 ([15], [25]) 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 $\sqrt{K}$.

7. Conclusions and future work

We investigated the performance of non-cooperative load-balancing in processor-sharing server-farms. We have first shown that 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, and therefore using the potential function method. We have then proved that, 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

[1] E. Altman, U. Ayesta, and B. Prabhu. Load balancing in processor sharing systems. *Telecommunication Systems*, 47(1–2):35–48, 2011.

[2] U. Ayesta, O. Brun, and B. J. Prabhu. Price of anarchy in non-cooperative server-farms. LAAS Research Report. Available at `http://hal.archives-ouvertes.fr/hal-00416123`.

[3] C. H. Bell and S. Stidham. Individual versus social optimization in the allocation of customers to alternative servers. *Management Science*, 29:831–839, 1983.

[4] S. C. Borst. Optimal probabilistic allocation of customer types to servers. In *Proceedings of ACM SIGMETRICS*, pages 116–125, Sept. 1995.

[5] V. Cardellini, E. Casalicchio, M. Colajanni, and P. S. Yu. The state of the art in locally distributed web-server systems. *ACM Computing Surveys*, 34(2):263–311, 2001.

[6] H. L. Chen, J. Marden, and A. Wierman. The effect of local scheduling in load balancing designs. In *Proceedings of IEEE Infocom*, 2009.

[7] Y.-C. Chow and W. H. Kohler. Models for dynamic load balancing in a heterogeneous multiple processor system. *IEEE Transactions on Computers*, 28(5):354–361, 1979.

[8] J. Cohen. The multiple phase service network with generalized processor sharing. *Acta Informatica*, 12:245–284, 1979.

[9] R. Cominetti, J. R. Correa, and N. E. Stier-Moses. The impact of oligopolistic competition in networks. *Oper. Res.*, 57:1421–1437, 2009.

[10] A. Czumaj, P. Krysta, and B. Vocking. Selfish traffic allocation for server farms. In *Proceedings of STOC*, 2002.

[11] S. El-Zoghdy, H. Kameda, and J. Li. Comparison of dynamic vs. static load-balancing policies in a mainframe-personal computer network model. *Information*, 5(4):431–446, 2002.

[12] S. El-Zoghdy, H. Kameda, and J. Li. Numerical studies on a paradox for non-cooperative static load balancing in distributed computer systems. *Computers and Operations Research*, (33):345–355, 2006.

[13] H. Feng, V. Misra, and D. Rubenstein. Optimal state-free, size-aware dispatching for heterogeneous M/G/-type systems. *Performance Evaluation*, 62(1–4):36–39, 2005.

[14] V. Gupta, M. Harchol-Balter, K. Sigman, and W. Whitt. Analysis of join-the-shortest-queue routing for web server farms. In *Proceedings of Performance*, 2007.

[15] M. Haviv and T. Roughgarden. The price of anarchy in an exponential multi-server. *Operations Research Letters*, 35:421–426, 2007.

[16] H. Kameda, J. Li, C. Kim, and Y. Zhang. *Optimal load balancing in distributed computer systems*. Springer-Verlag, 1997.

[17] F. Kelly. *Stochastic Networks and Reversibility*. Wiley, Chichester, 1979.

[18] Y. Korilis, A. Lazar, and A. Orda. Capacity allocation under noncooperative routing. *IEEE Transactions on Automatic Control*, 42(3):309–325, March 1997.

[19] E. Koutsoupias and C. H. Papadimitriou. Worst-case equilibria. In *STACS 1999*, 1999.

[20] J. Li and H. Kameda. Load balancing problems for multiclass jobs in distributed/parallel computer systems. *IEEE Transactions on Computers*, 47(3):322–332, 1998.

[21] D. Monderer and L. S. Shapley. Potential games. *Games and Econ. Behavior*, 14:124–143, 1996.

[22] L. M. Ni and K. Hwang. Optimal load balancing in a multiple processor with many job classes. *IEEE Trans. Software Eng.*, 11(5):491–496, 1985.

[23] A. Orda, R. Rom, and N. Shimkin. Competitive routing in multi-user communication networks. *IEEE/ACM Transactions on Networking*, 1:510–521, October 1993.

[24] B. Shirazi, A. Hurson, and K. Kavi. *Scheduling and load-balancing in parallel and distributed systems.* Silver Spring, MD: IEEE Computer Society Press, 1995.

[25] T. Wu and D. Starobinski. A comparative analysis of server selection in content replication networks. *IEEE/ACM Trans. Netw.*, 16(6):1461–1474, 2008.

Appendix A. Proof of results in Section 4

Appendix A.1. Proof of Proposition 1

We first prove a series of technical lemmata before proving the main statement.

Lemma 7. $\mathcal{S}_i \cap \mathcal{S}_k \neq \emptyset$.

Proof. Assume the contrary, i.e., if $m \in \mathcal{S}_i$ then $m \notin \mathcal{S}_k$, and if $n \in \mathcal{S}_k$ then $n \notin \mathcal{S}_i$. For one such pair m and n, from (5), we can conclude that $\mu_i > \frac{c_m}{r_m - y_m} \geq \mu_k$ and $\mu_k > \frac{c_n}{r_n - y_n} \geq \mu_i$, which is a contradiction. ∎

Since $\mathcal{S}_i \cap \mathcal{S}_k \neq \emptyset$, from (3), we have

$$\mu_i - \mu_k = \frac{c_j}{(r_j - y_j)^2}(x_{i,j} - x_{k,j}), \quad \forall j \in \mathcal{S}_i \cap \mathcal{S}_k. \tag{A.1}$$

Lemma 8. $\mu_i < \mu_k \iff \exists j \in \mathcal{S}_k : x_{i,j} < x_{k,j}$.

Proof. *Straight part:* From Lemma 7, $\mathcal{S}_i \cap \mathcal{S}_k \neq \emptyset$. If $\mu_i < \mu_k$, then, from (A.1), $\exists j \in \mathcal{S}_k : x_{i,j} < x_{k,j}$.

Converse part: $\exists j \in \mathcal{S}_k : x_{i,j} < x_{k,j}$. Either $j \in \mathcal{S}_i$ or $j \notin \mathcal{S}_i$. If $j \in \mathcal{S}_i$ then, from (A.1), $\mu_i < \mu_k$. If $j \notin \mathcal{S}_i$, then, from (5), $\mu_i \leq \frac{c_j}{r_j - y_j} < \mu_k$. ∎

Lemma 9. *If $\mu_i < \mu_k$, then $\mathcal{S}_i \subset \mathcal{S}_k$.*

Proof. If $j \in \mathcal{S}_i$, then, from (5), $\frac{c_j}{r_j - y_j} < \mu_i$. If $\mu_i < \mu_k$ then $\frac{c_j}{r_j - y_j} < \mu_k$. Hence, from (5) we can conclude that $j \in \mathcal{S}_k$. Therefore, $\mathcal{S}_i \subset \mathcal{S}_k$. ∎

Lemma 10. $\exists m \in \mathcal{S}_k : x_{i,m} < x_{k,m} \iff x_{i,j} < x_{k,j}, \quad \forall j \in \mathcal{S}_k.$

PROOF. *Straight part:* If $\exists m \in \mathcal{S}_k : x_{i,m} < x_{k,m}$, then, from Lemmata 8 and 9, we have $\mu_i < \mu_k$ and $\mathcal{S}_i \subset \mathcal{S}_k$. For $j \in \mathcal{S}_i$, from (A.1), we have $x_{i,j} < x_{k,j}$. For $j \in \mathcal{S}_k \setminus \mathcal{S}_i$, $x_{i,j} = 0$ and $0 < x_{k,j}$. Hence, $x_{i,j} < x_{k,j}$, $\forall j \in \mathcal{S}_k$.

Converse part: It is true from the statement. ■

We are now in position of proving Proposition 1: PROOF. *1* $\iff$ *2* $\iff$ *3* follows from Lemmata 8 and 10. Now, we show *3* $\iff$ *4*.

Straight part: If $x_{i,j} < x_{k,j}$, $\forall j \in \mathcal{S}_k$, then, from the fact that *3* $\iff$ *1* and Lemma 9, we can conclude that $\lambda_i = \sum_{j \in \mathcal{S}_i} x_{i,j} = \sum_{j \in \mathcal{S}_k} x_{i,j} < \sum_{j \in \mathcal{S}_k} x_{k,j} = \lambda_k$.

Converse part: Since $\lambda_k = \sum_{j \in \mathcal{S}_k} x_{k,j}$, if $\lambda_i < \lambda_k$, then $\exists j \in \mathcal{S}_k : x_{i,j} < x_{k,j}$. Since *2* $\iff$ *3*, if $\lambda_i < \lambda_k$, then $x_{i,j} < x_{k,j}$, $\forall j \in \mathcal{S}_k$. ■

Appendix A.2. Proof of Proposition 2

We first prove a series of technical lemmata before proving the main statement.

We recall that κ_m is to class m what μ_i is to class i. Also, for $i \in \mathcal{C}_m$, we can rewrite (3) as

$$\begin{aligned} \mu_i &= \frac{c_m}{r_m - y_m}\left(1 + \frac{c_m}{r_j - y_m}\frac{x_{i,m}}{c_m}\right) \\ &= \kappa_m^{-1}\left(1 + \kappa_m^{-1}\frac{x_{i,m}}{c_m}\right). \end{aligned} \tag{A.2}$$

Lemma 11. $\mathcal{C}_m \cap \mathcal{C}_n \neq \emptyset$.

PROOF. Assume the contrary, i.e., if $i \in \mathcal{C}_m$, then $i \notin \mathcal{C}_n$, and if $k \in \mathcal{C}_n$, then $k \notin \mathcal{C}_m$. For one such pair i and k, from (6), we can conclude that $\kappa_m > \mu_i^{-1} \geq \kappa_n$ and $\kappa_n > \mu_k^{-1} \geq \kappa_m$, which is a contradiction. ■

Since $\mathcal{C}_m \cap \mathcal{C}_n \neq \emptyset$, from (A.2), we have

$$\kappa_m^{-1}\left(1 + \kappa_m^{-1}\frac{x_{i,m}}{c_m}\right) = \kappa_n^{-1}\left(1 + \kappa_n^{-1}\frac{x_{i,n}}{c_n}\right), \quad \forall i \in \mathcal{C}_m \cap \mathcal{C}_n. \tag{A.3}$$

Lemma 12. $\kappa_m < \kappa_n \iff \exists i \in \mathcal{C}_n : \frac{x_{i,m}}{c_m} < \frac{x_{i,n}}{c_n}$.

PROOF. *Straight part:* From Lemma 11, $\mathcal{C}_m \cap \mathcal{C}_n \neq \emptyset$. If $\kappa_m < \kappa_n$, then, from (A.3), $\exists i : \frac{x_{i,m}}{c_m} < \frac{x_{i,n}}{c_n}$.

Converse part: $\exists i \in \mathcal{C}_n : \frac{x_{i,m}}{c_m} < \frac{x_{i,n}}{c_n}$. Either $i \in \mathcal{C}_m$ or $i \notin \mathcal{C}_m$. If $i \in \mathcal{C}_m$, then, from (A.3), $\kappa_m < \kappa_n$. If $i \notin \mathcal{C}_m$, then, from (6), $\kappa_m \leq \mu_i^{-1} < \kappa_n$. ■

Lemma 13. *If* $\kappa_m < \kappa_n$*, then* $\mathcal{C}_m \subset \mathcal{C}_n$.

PROOF. If $i \in \mathcal{C}_m$, then, from (6), $\mu_i < \kappa_m$. If $\kappa_m < \kappa_n$, then $\mu_i < \kappa_n$. Hence, from (6) we can conclude that $i \in \mathcal{C}_n$. Therefore, $\mathcal{C}_m \subset \mathcal{C}_n$. ■

Lemma 14. $\exists m \in \mathcal{C}_i : \frac{x_{i,m}}{c_m} < \frac{x_{i,n}}{c_n} \iff \frac{x_{i,m}}{c_m} < \frac{x_{i,n}}{c_n}, \forall i \in \mathcal{C}_n$.

PROOF. *Straight part:* If $\exists i \in \mathcal{C}_n : \frac{x_{i,m}}{c_m} < \frac{x_{i,n}}{c_n}$, then, from Lemmata 12 and 13, we have $\kappa_m < \kappa_n$ and $\mathcal{C}_m \subset \mathcal{C}_n$. For $i \in \mathcal{C}_m$, from (A.3), $\frac{x_{i,m}}{c_m} < \frac{x_{i,n}}{c_n}$. For $i \in \mathcal{C}_n \setminus \mathcal{C}_m$, $x_{i,m} = 0$ and $0 < x_{i,n}$. Hence, $\frac{x_{i,m}}{c_m} < \frac{x_{i,n}}{c_n}$, $\forall i \in \mathcal{C}_n$.

Converse part: It is true from the statement. ■

We are now in position to show Proposition 2: PROOF. *1* $\iff$ *2* $\iff$ *3* follows from Lemmata 12 and 14. Next, we show *3* $\iff$ *4*.

Straight part: If $\frac{x_{i,m}}{c_m} < \frac{x_{i,n}}{c_n}$, $\forall i \in \mathcal{C}_n$, then from the fact that *3* $\iff$ *1* and Lemma 13, we can conclude that $\frac{r_m}{c_m} = \kappa_m + \sum_{i\in\mathcal{C}_m} \frac{x_{i,m}}{c_m} < \kappa_n + \sum_{i\in\mathcal{C}_n} \frac{x_{i,n}}{c_n} = \frac{r_n}{c_n}$.

Converse part: Since $\frac{r_n}{c_n} = \kappa_n + \sum_{i\in\mathcal{C}_n} \frac{x_{i,n}}{c_n}$, if $\frac{r_m}{c_m} < \frac{r_n}{c_n}$, then either $\kappa_m < \kappa_n$ or $\exists i \in \mathcal{C}_n : \frac{x_{i,m}}{c_m} < \frac{x_{i,n}}{c_n}$. Since *1* $\iff$ *2* $\iff$ *3*, we can conclude that if $\frac{r_m}{c_m} < \frac{c_n}{r_n}$, then $\frac{x_{i,m}}{c_m} < \frac{x_{i,n}}{c_n}$, $\forall i \in \mathcal{C}_n$. ■

Appendix A.3. Proof of Lemma 1

PROOF. From (3), if $x_{i,j} > 0$, then

$$\mu_i = \frac{c_j}{r_j - y_j} + \frac{c_j x_{i,j}}{(r_j - y_j)^2},$$

from which we obtain

$$\sum_{i\in\mathcal{C}_j} \mu_i = \frac{c_j N_j}{r_j - y_j} + \frac{c_j y_j}{(r_j - y_j)^2} = \frac{c_j(N_j - 1)}{r_j - y_j} + \frac{c_j r_j}{(r_j - y_j)^2}.$$

Since $\sum_{i\in\mathcal{C}_{j+1}} \mu_i = \sum_{i\in\mathcal{C}_j} \mu_i - \sum_{i\in\mathcal{C}_j\setminus\mathcal{C}_{j+1}} \mu_i$,

$$\frac{c_{j+1}(N_{j+1} - 1)}{r_{j+1} - y_{j+1}} + \frac{c_{j+1} r_{j+1}}{(r_{j+1} - y_{j+1})^2} = \frac{c_j(N_{j+1} - 1)}{r_j - y_j} + \frac{c_j r_j}{(r_j - y_j)^2} - \sum_{i\in\mathcal{C}_j\setminus\mathcal{C}_{j+1}} \frac{c_j x_{i,j}}{(r_j - y_j)^2}.$$

Thus,

$$\frac{c_j r_j}{(r_j - y_j)^2} - \frac{c_{j+1} r_{j+1}}{(r_{j+1} - y_{j+1})^2} = \left(\frac{c_{j+1}}{r_{j+1} - y_{j+1}} - \frac{c_j}{r_j - y_j}\right)(N_{j+1} - 1) + \sum_{i\in\mathcal{C}_j\setminus\mathcal{C}_{j+1}} \frac{c_j x_{i,j}}{(r_j - y_j)^2}.$$

From Proposition 2, $\frac{c_j}{r_j} \leq \frac{c_{j+1}}{r_{j+1}}$ implies $\kappa_j \geq \kappa_{j+1}$, i.e. $\frac{c_{j+1}}{r_{j+1} - y_{j+1}} \geq \frac{c_j}{r_j - y_j}$. Since $\sum_{i\in\mathcal{C}_j\setminus\mathcal{C}_{j+1}} \frac{c_j x_{i,j}}{(r_j - y_j)^2}$ is strictly positive if $\mathcal{C}_j \setminus \mathcal{C}_{j+1} \neq \emptyset$, we can conclude that

$$\frac{c_j r_j}{(r_j - y_j)^2} - \frac{c_{j+1} r_{j+1}}{(r_{j+1} - y_{j+1})^2} \geq 0,$$

with strict inequality if $\mathcal{C}_j \setminus \mathcal{C}_{j+1} \neq \emptyset$. ■

Appendix B. Proofs of results in Section 5.1

In this appendix, we present the proofs of Propositions 4 and 5. These proofs use some preliminary results which are stated below.

Appendix B.1. Preliminary Results

The following two lemmata compare the Nash equilibria induced by two different rate vectors $\boldsymbol{\lambda}$ and $\hat{\boldsymbol{\lambda}}$. They do not rely on Assumption 1.

Lemma 15. *For $i \in \mathcal{C}_j$,*

1. *if $\hat{y}_j < y_j$ and $\hat{x}_{i,j} \leq x_{i,j}$, then $\hat{\mu}_i < \mu_i$.*
2. *if $\hat{y}_j \leq y_j$ and $\hat{x}_{i,j} \leq x_{i,j}$, then $\hat{\mu}_i \leq \mu_i$.*
3. *if $\hat{y}_j \leq y_j$ and $\hat{x}_{i,j} < x_{i,j}$, then $\hat{\mu}_i < \mu_i$.*
4. *if $\hat{y}_j = y_j$ and $\hat{\mu}_i < \mu_i$, then $\hat{x}_{i,j} < x_{i,j}$.*

Proof. *Proof of part* 1 : for $i \in \mathcal{C}_j$, we rewrite (3) as

$$x_{i,j} = (r_j - y_j)\left(\frac{r_j - y_j}{c_j}\mu_i - 1\right).$$

Therefore, $\hat{x}_{i,j} \leq x_{i,j}$ is equivalent to

$$(r_j - \hat{y}_j)\left(\frac{r_j - \hat{y}_j}{c_j}\hat{\mu}_i - 1\right) \leq (r_j - y_j)\left(\frac{r_j - y_j}{c_j}\mu_i - 1\right),$$

Since $r_j - y_j < r_j - \hat{y}_j$, we can conclude that $\hat{\mu}_i < \mu_i$. The proofs of parts 2, 3, and 4 follow similarly. ■

Lemma 16. *For m and n in $\mathcal{S}$, and $i \in \mathcal{C}_m \cap \mathcal{C}_n$,*
if $\hat{y}_m > y_m$, $\hat{x}_{i,m} \geq x_{i,m}$, and $\hat{y}_n \leq y_n$, then $\hat{x}_{i,n} > x_{i,n}$.

Proof. Assume the contrary, that is, $\exists\, n, m \in \mathcal{S}$ and $i \in \mathcal{C}_m \cap \mathcal{C}_n$ such that $\hat{y}_m > y_m$, $\hat{x}_{i,m} \geq x_{i,m}$, $\hat{y}_n \leq y_n$ and $\hat{x}_{i,n} \leq x_{i,n}$. From Lemma 15.1, $\hat{y}_m > y_m$ and $\hat{x}_{i,m} \geq x_{i,m}$ implies $\hat{\mu}_i > \mu_i$. However, from Lemma 15.2, $\hat{y}_n \leq y_n$ and $\hat{x}_{i,n} \leq x_{i,n}$ implies $\hat{\mu}_i \leq \mu_i$, which is a contradiction. ■

We will also use the following result which is valid under Assumption 1.

Lemma 17. *Under Assumption 1, for any $j \in \mathcal{S}$,*
$\hat{y}_j \geq y_j \iff \sum_{i\in\mathcal{C}_j} \hat{\mu}_i \geq \sum_{i\in\mathcal{C}_j} \mu_i$.

Proof. From (2) and (3),

$$\sum_{i\in\mathcal{C}_j} \mu_i = N_j c_j \frac{1}{r_j - y_j} + c_j \frac{y_j}{(r_j - y_j)^2}.$$

Since $N_j = \hat{N}_j$ (from Assumption 1), we can conclude that $\sum_{i\in\mathcal{C}_j} \mu_i$ is an increasing function of y_j. ■

Appendix B.2. Proof of Proposition 4

PROOF. *Proof of part* 1 *:* Assume by contradiction that we can find a server $s \in \mathcal{S}_1$ such that $s \in \mathcal{S}^-$. Then, according to Corollary 5, $\mathcal{S}_1 \subset \mathcal{S}^-$. Since $\mathcal{S}^+ \neq \emptyset$ and $\hat{y}_j > y_j$ for all $j \in \mathcal{S}^+$, we have $\sum_{j\in\mathcal{S}^+} \hat{y}_j > \sum_{j\in\mathcal{S}^+} y_j$, i.e.,

$$\sum_{i\in\mathcal{C}} \left(\sum_{j\in\mathcal{S}^+} \hat{x}_{i,j} \right) > \sum_{i\in\mathcal{C}} \left(\sum_{j\in\mathcal{S}^+} x_{i,j} \right),$$

from which we conclude that there exists i such that $\sum_{j\in\mathcal{S}^+} \hat{x}_{i,j} > \sum_{j\in\mathcal{S}^+} x_{i,j}$. Since $\mathcal{S}_k = \mathcal{S}_1 \subset \mathcal{S}^-$ for all $k \in \mathcal{C}_{min}$, we necessarily have $i \notin \mathcal{C}_{min}$ and thus $\hat{\lambda}_i \leq \lambda_i$. Therefore,

$$\hat{\lambda}_i = \sum_{j\in\mathcal{S}^-} \hat{x}_{i,j} + \sum_{j\in\mathcal{S}^+} \hat{x}_{i,j} \leq \sum_{j\in\mathcal{S}^-} x_{i,j} + \sum_{j\in\mathcal{S}^+} x_{i,j} = \lambda_i.$$

Thus,

$$\sum_{j\in\mathcal{S}^-} \hat{x}_{i,j} \leq \sum_{j\in\mathcal{S}^-} x_{i,j} + \left(\sum_{j\in\mathcal{S}^+} x_{i,j} - \sum_{j\in\mathcal{S}^+} \hat{x}_{i,j} \right) < \sum_{j\in\mathcal{S}^-} x_{i,j}.$$

We therefore conclude that class i is such that $\sum_{j\in\mathcal{S}^+} \hat{x}_{i,j} > \sum_{j\in\mathcal{S}^+} x_{i,j}$ and $\sum_{j\in\mathcal{S}^-} \hat{x}_{i,j} < \sum_{j\in\mathcal{S}^-} x_{i,j}$. Therefore, we can find a server $m \in \mathcal{S}^+$ and a server $n \in \mathcal{S}^-$ such that $\hat{x}_{i,m} > x_{i,m}$ and $\hat{x}_{i,n} < x_{i,n}$. But according to Lemma 16, this is impossible. We therefore conclude that $\mathcal{S}_1 \subset \mathcal{S}^+$.

Proof of part 2 *:* We first prove that if $S^+ = \emptyset$ then $\mathcal{S}_1 = \mathcal{S}_K$. This is equivalent to proving that if $y_j = \hat{y}_j$, $\forall j \in \mathcal{S}$ then $\mathcal{S}_1 = \mathcal{S}_K$. Assume the contrary, that is $\mathcal{S}_1 \subsetneq \mathcal{S}_K$. Then, $\exists m : m \in S_K, m \notin S_1$. Since $y_m = \hat{y}_m$, from Lemma 17, we get $\sum_{i\in\mathcal{C}_m} \mu_i = \sum_{i\in\mathcal{C}_m} \hat{\mu}_i$, which we can rewrite as

$$\sum_{i\in\mathcal{C}_{max}} \mu_i + \sum_{i\in\mathcal{C}_m\setminus\mathcal{C}_{max}} \mu_i = \sum_{i\in\mathcal{C}_{max}} \hat{\mu}_i + \sum_{i\in\mathcal{C}_m\setminus\mathcal{C}_{max}} \hat{\mu}_i. \tag{B.1}$$

We shall show that the above equality is not possible, which then proves the claim. For $i \in \mathcal{C}_{max}$, since $\lambda_i > \hat{\lambda}_i$, $\sum_{j\in\mathcal{S}_i} x_{i,j} > \sum_{j\in\mathcal{S}_i} \hat{x}_{i,j}$. Thus, there exists an $n \in \mathcal{S}_i$ such that $x_{i,n} > \hat{x}_{i,n}$. Since $y_n = \hat{y}_n$, from Lemma 15.3, we can conclude that $\mu_i > \hat{\mu}_i$, and that $\sum_{i\in\mathcal{C}_{max}} \mu_i > \sum_{i\in\mathcal{C}_{min}} \hat{\mu}_i$, which, upon substitution in (B.1), leads to

$$\sum_{i\in\mathcal{C}_m\setminus\mathcal{C}_{max}} \mu_i < \sum_{i\in\mathcal{C}_m\setminus\mathcal{C}_{max}} \hat{\mu}_i.$$

If $\mathcal{C}_m \setminus \mathcal{C}_{max} = \emptyset$, then the above inequality cannot be possible which then proves the claim. So, assume $\mathcal{C}_m \setminus \mathcal{C}_{max} \neq \emptyset$. Then the above inequality implies that $\exists i \notin C_{min} \cup C_{max} : \mu_i < \hat{\mu}_i$. Since $y_j = \hat{y}_j, \forall j \in \mathcal{S}_i$, application of Lemma 15.4 leads to $x_{i,j} < \hat{x}_{i,j}, \forall j \in \mathcal{S}_i$, and consequently to $\lambda_i = \sum_{j\in\mathcal{S}_i} x_{i,j} < \sum_{j\in\mathcal{S}_i} \hat{x}_{i,j} = \hat{\lambda}_i$. However, $\lambda_i = \hat{\lambda}_i$ for $i \notin \mathcal{C}_{min} \cup \mathcal{C}_{max}$. Hence, there is a contradiction, and we can conclude that $\mathcal{S}_1 = \mathcal{S}_K$. ■

Appendix B.3. Proof of Proposition 5

PROOF. If $\mathcal{S}^+ = \emptyset$ then the proposition is true. So, assume $\mathcal{S}^+ \neq \emptyset$. Then, from Proposition 4, $\mathcal{S}_1 \subset \mathcal{S}^+$. In order to prove the proposition, assume by contradiction that there exists a server $j \in \{S_1 + 1, \ldots, S_K - 1\}$ such that $j \in \mathcal{S}^-$ and $j + 1 \in \mathcal{S}^+$. Again, if $S_1 + 1 = S_K$ then the proposition is true. So, assume that $S_1 + 1 < S_K$.

Since $j \in \mathcal{S}^-$ and $j + 1 \in \mathcal{S}^+$, from Lemma 2,

$$\sum_{i \in \mathcal{C}_j} \hat{\mu}_i \leq \sum_{i \in \mathcal{C}_j} \mu_i, \tag{B.2}$$

and

$$\sum_{i \in \mathcal{C}_{j+1}} \hat{\mu}_i > \sum_{i \in \mathcal{C}_{j+1}} \mu_i, \tag{B.3}$$

Moreover, from the contrapositive of Lemma 2, we can conclude that $\mathcal{C}_j \setminus \mathcal{C}_{j+1} \neq \emptyset$. Note that since $j < S_K$, classes $i \in \mathcal{C}_{max}$ do not belong to $\mathcal{C}_j \setminus \mathcal{C}_{j+1}$. Similarly, since $j > S_1$, classes $i \in \mathcal{C}_{min}$ do not belong to $\mathcal{C}_j \setminus \mathcal{C}_{j+1}$.

Since $\mathcal{C}_{j+1} \subset \mathcal{C}_j$, $\sum_{i \in \mathcal{C}_{j+1}} \hat{\mu}_i > \sum_{i \in \mathcal{C}_{j+1}} \mu_i$ is equivalent to

$$N_{j+1} \frac{c_j}{r_j - \hat{y}_j} + \frac{c_j}{(r_j - \hat{y}_j)^2} \sum_{i \in \mathcal{C}_{j+1}} \hat{x}_{i,j} > N_{j+1} \frac{c_j}{r_j - y_j} + \frac{c_j}{(r_j - y_j)^2} \sum_{i \in \mathcal{C}_{j+1}} x_{i,j},$$

and since $\hat{y}_j \leq y_j$, this implies that $\sum_{i \in \mathcal{C}_{j+1}} \hat{x}_{i,j} > \sum_{i \in \mathcal{C}_{j+1}} x_{i,j}$. Since $\hat{y}_j \leq y_j$, necessarily $\sum_{i \in \mathcal{C}_j \setminus \mathcal{C}_{j+1}} \hat{x}_{i,j} < \sum_{i \in \mathcal{C}_j \setminus \mathcal{C}_{j+1}} x_{i,j}$. However, since all classes $k \in \mathcal{C}_{min} \cup \mathcal{C}_{max}$ do not belong to $\mathcal{C}_j \setminus \mathcal{C}_{j+1}$, we know that $\hat{\lambda}_i = \lambda_i$ for all $i \in \mathcal{C}_j \setminus \mathcal{C}_{j+1}$, and thus

$$\sum_{l=1}^{j} \sum_{i \in \mathcal{C}_j \setminus \mathcal{C}_{j+1}} x_{i,l} = \sum_{l=1}^{j} \sum_{i \in \mathcal{C}_j \setminus \mathcal{C}_{j+1}} \hat{x}_{i,l},$$

from which we obtain

$$\sum_{l<j} \sum_{i \in \mathcal{C}_j \setminus \mathcal{C}_{j+1}} x_{i,l} = \sum_{l<j} \sum_{i \in \mathcal{C}_j \setminus \mathcal{C}_{j+1}} \hat{x}_{i,l} + \left(\sum_{i \in \mathcal{C}_j \setminus \mathcal{C}_{j+1}} \hat{x}_{i,j} - \sum_{i \in \mathcal{C}_j \setminus \mathcal{C}_{j+1}} x_{i,j} \right),$$

and therefore

$$\sum_{l<j} \sum_{i \in \mathcal{C}_j \setminus \mathcal{C}_{j+1}} x_{i,l} < \sum_{l<j} \sum_{i \in \mathcal{C}_j \setminus \mathcal{C}_{j+1}} \hat{x}_{i,l}. \tag{B.4}$$

Substracting (B.3) from (B.2), we obtain $\sum_{i \in \mathcal{C}_j \setminus \mathcal{C}_{j+1}} \hat{\mu}_i < \sum_{i \in \mathcal{C}_j \setminus \mathcal{C}_{j+1}} \mu_i$. Hence, for each server $l < j$,

$$(N_j - N_{j+1}) \frac{c_l}{r_l - \hat{y}_l} + \frac{c_l}{(r_l - \hat{y}_l)^2} \sum_{i \in \mathcal{C}_j \setminus \mathcal{C}_{j+1}} \hat{x}_{i,l} < (N_j - N_{j+1}) \frac{c_l}{r_l - y_l} + \frac{c_l}{(r_l - y_l)^2} \sum_{i \in \mathcal{C}_j \setminus \mathcal{C}_{j+1}} x_{i,l}.$$

But, for $l < j$ and $l \in \mathcal{S}^+$, it implies that $\sum_{i\in\mathcal{C}_j\setminus\mathcal{C}_{j+1}} \hat{x}_{i,l} < \sum_{i\in\mathcal{C}_j\setminus\mathcal{C}_{j+1}} x_{i,l}$, and thus

$$\sum_{l<j,l\in\mathcal{S}^+} \sum_{i\in\mathcal{C}_j\setminus\mathcal{C}_{j+1}} \hat{x}_{i,l} < \sum_{l<j,l\in\mathcal{S}^+} \sum_{i\in\mathcal{C}_j\setminus\mathcal{C}_{j+1}} x_{i,l}. \tag{B.5}$$

From (B.4), we have

$$\sum_{l<j,l\in\mathcal{S}^-} \sum_{i\in\mathcal{C}_j\setminus\mathcal{C}_{j+1}} \hat{x}_{i,l} > \sum_{l<j,l\in\mathcal{S}^-} \sum_{i\in\mathcal{C}_j\setminus\mathcal{C}_{j+1}} x_{i,l} + \left(\sum_{l<j,l\in\mathcal{S}^+} \sum_{i\in\mathcal{C}_j\setminus\mathcal{C}_{j+1}} x_{i,l} - \sum_{l<j,l\in\mathcal{S}^+} \sum_{i\in\mathcal{C}_j\setminus\mathcal{C}_{j+1}} \hat{x}_{i,l} \right),$$

and using (B.5) it leads to

$$\sum_{l<j,l\in\mathcal{S}^-} \sum_{i\in\mathcal{C}_j\setminus\mathcal{C}_{j+1}} \hat{x}_{i,l} > \sum_{l<j,l\in\mathcal{S}^-} \sum_{i\in\mathcal{C}_j\setminus\mathcal{C}_{j+1}} x_{i,l}. \tag{B.6}$$

According to (B.3), for each server $l < j$,

$$N_{j+1} \frac{c_l}{r_l - \hat{y}_l} + \frac{c_l}{(r_l - \hat{y}_l)^2} \sum_{i\in\mathcal{C}_{j+1}} \hat{x}_{i,l} > N_{j+1} \frac{c_l}{r_l - y_l} + \frac{c_l}{(r_l - y_l)^2} \sum_{i\in\mathcal{C}_{j+1}} x_{i,l}.$$

But, for $l < j$, $l \in \mathcal{S}^-$, it implies that $\sum_{i\in\mathcal{C}_{j+1}} \hat{x}_{i,l} > \sum_{i\in\mathcal{C}_{j+1}} x_{i,l}$, and thus

$$\sum_{l<j,l\in\mathcal{S}^-} \sum_{i\in\mathcal{C}_{j+1}} \hat{x}_{i,l} > \sum_{l<j,l\in\mathcal{S}^-} \sum_{i\in\mathcal{C}_{j+1}} x_{i,l} \tag{B.7}$$

Now, summing (B.7) and (B.6) gives

$$\sum_{l<j,l\in\mathcal{S}^-} \sum_{i\in\mathcal{C}_j} \hat{x}_{i,l} > \sum_{l<j,l\in\mathcal{S}^-} \sum_{i\in\mathcal{C}_j} x_{i,l}. \tag{B.8}$$

However, for each server $l \in \mathcal{S}^-$, we have $\hat{y}_l \leq y_l$ and thus $\sum_{l<j,l\in\mathcal{S}^-} \hat{y}_l \leq \sum_{l<j,l\in\mathcal{S}^-} y_l$. Since, for $l < j$, y_l can also be written as $y_l = \sum_{i\in\mathcal{C}_j} x_{i,l} + \sum_{i\notin\mathcal{C}_j} x_{i,l}$, it yields

$$\sum_{l<j,l\in\mathcal{S}^-} \sum_{i\notin\mathcal{C}_j} \hat{x}_{i,l} \leq \sum_{l<j,l\in\mathcal{S}^-} \sum_{i\notin\mathcal{C}_j} x_{i,l} + \left(\sum_{l<j,l\in\mathcal{S}^-} \sum_{i\in\mathcal{C}_j} x_{i,l} - \sum_{l<j,l\in\mathcal{S}^-} \sum_{i\in\mathcal{C}_j} \hat{x}_{i,l} \right),$$

and using (B.8),

$$\sum_{l<j,l\in\mathcal{S}^-} \sum_{i\notin\mathcal{C}_j} \hat{x}_{i,l} < \sum_{l<j,l\in\mathcal{S}^-} \sum_{i\notin\mathcal{C}_j} x_{i,l}. \tag{B.9}$$

Therefore, there exists a class $i \notin \mathcal{C}_j$ such that

$$\sum_{l<j,l\in\mathcal{S}^-} \hat{x}_{i,l} < \sum_{l<j,l\in\mathcal{S}^-} x_{i,l}. \tag{B.10}$$

It implies that, for this class i, we can find a server $n \notin \mathcal{S}_i$ and $n \in \mathcal{S}^-$ such that $\hat{x}_{i,n} < x_{i,n}$. Since $\mathcal{C}_{max} \subsetneq \mathcal{C}_j$, we know that $i \notin \mathcal{C}_{max}$. Moreover, since $\mathcal{S}_k = \mathcal{S}_1 \subset \mathcal{S}^+$ for all $k \in \mathcal{C}_{min}$, $i \notin \mathcal{C}_{min}$. We therefore have $\hat{\lambda}_i = \lambda_i$. Thus,

$$\sum_{l \in \mathcal{S}^-} \hat{x}_{i,l} + \sum_{l \in \mathcal{S}^+} \hat{x}_{i,l} = \sum_{l \in \mathcal{S}^-} x_{i,l} + \sum_{l \in \mathcal{S}^+} x_{i,l},$$

which implies

$$\sum_{l \in \mathcal{S}^+} \hat{x}_{i,l} = \sum_{l \in \mathcal{S}^+} x_{i,l} + \left(\sum_{l \in \mathcal{S}^-} x_{i,l} - \sum_{l \in \mathcal{S}^-} \hat{x}_{i,l} \right),$$

and with (B.10), it yields $\sum_{l \in \mathcal{S}^+} \hat{x}_{i,l} > \sum_{l \in \mathcal{S}^+} x_{i,l}$. This implies that there exists a server $m < j$, $m \in \mathcal{S}^+$ such that $\hat{x}_{i,m} > x_{i,m}$. But, according to Lemma 16, there cannot be two servers $m, n \in \mathcal{S}$ such that $\hat{y}_m > y_m$, $\hat{y}_n \leq y_n$, $\hat{x}_{i,m} > x_{i,m}$ and $\hat{x}_{i,n} < x_{i,n}$. This is a contradiction. Therefore, if $j \in \mathcal{S}^-$, then $j+1 \in \mathcal{S}^-$ for all servers $j \in \mathcal{S}$. ■

Appendix B.4. Proof of Lemma 3

Proof. Writting $\sum_i a_i b_i$ as $\sum_{i \leq I} a_i b_i + \sum_{i > I} a_i b_i$, we get

$$\sum_{i \leq I} a_i b_i + \sum_{i > I} a_i b_i \geq a_I \sum_{i \leq I} b_i - a_{I+1} \sum_{i > I} |b_i| \geq (a_I - a_{I+1}) \sum_{i \leq I} b_i > 0.$$

■

Appendix C. Proofs of results in section 6.1

Appendix C.1. Proof of Lemma 5

Proof. Let us first prove property 1. By definition $W(k, x) = W_j(K, x)$ in the interval $[u_j, u_{j+1})$. Since W_j is continuous and decreasing in (u_j, u_{j+1}) so is W. To conclude the proof, we need to verify that W is continuous at $u_j, j = 2, 3, \ldots, S$. We have

$$\begin{aligned}\lim_{x \to u_j^+} W(K,x) - \lim_{x \to u_j^-} W(K,x) &= \lim_{x \to u_j^+} W_j(K,x) - \lim_{x \to u_j^-} W_{j-1}(K,x) \\ &= \frac{2r_j}{\sqrt{(K-1)^2 + 4Ku_j^{-1}u_j} - (K-1)} - r_j \quad = 0,\end{aligned}$$

which shows that the function $W(K, x)$ is also continuous at the points u_j, $j = 2, 3, ..., S$.

To prove property 2 it is sufficient to show that $\sqrt{(K-1)^2 + 4Ku_i^{-1}u_j} - (K-1)$ is increasing in K, for which we show below that its derivative with respect to K is positive.

$$
\begin{array}{rcl}
\frac{1}{2}\frac{2(K-1)+4u_i^{-1}u_j}{\sqrt{(K-1)^2+4Ku_i^{-1}u_j}}-1 & > & 0 \\
\Leftrightarrow \qquad (K-1)+2u_i^{-1}u_j & > & \sqrt{(K-1)^2+4Ku_i^{-1}u_j} \\
\Leftrightarrow \qquad 4(u_i^{-1}u_j)^2-4u_i^{-1}u_j & > & 0.
\end{array}
$$

Since $u_i^{-1}u_j > 1$ the above inequality holds.

Finally, let us now prove property 3. First, we note that $W(K, u_1) = \bar{\lambda}$ and $W(K, \infty) = -\bar{r} + \bar{\lambda}$ which is negative (by assumption). Also according to property 1, $W(K, x)$ is continuous and decreasing in the interval $[u_1, \infty)$. Hence, there is a unique value of x for which $W(K, x) = 0$. ■

Appendix C.2. Proof of Proposition 9

PROOF. Let $\mathbf{y}$ be an optimal solution of the equivalent problem stated in Proposition 8. For ease of notation, let $\phi_j = r_j/(r_j - y_j)$, $j \in \mathcal{S}$. According to the KKT conditions, there exist γ such that for each $j \in \mathcal{S}$,

$$
\begin{array}{rcll}
K\, u_j^{-1}\, \gamma & \leq & \phi_j(\phi_j + K - 1), \quad \forall j \in \mathcal{S}, & \text{(C.1)} \\
y_j\, \left[\phi_j(\phi_j + K - 1)\; - \; K\, u_j^{-1}\, \gamma\right] & = & 0, \quad \forall j \in \mathcal{S}, & \text{(C.2)}
\end{array}
$$

with equality in (C.1) if and only if $y_j > 0$.

Let us now consider a server j. Let us first assume that $u_j < \gamma$. In this case, a necessary condition for (C.1) to hold is $\phi_j(\phi_j + K - 1) > K$, which implies $\phi_j > 1$ and hence $y_j > 0$. We therefore obtain from (C.2) that

$$\phi_j^2 + (K-1)\phi_j - Ku_j^{-1}\gamma = 0.$$

The above equation has a single positive root given by

$$\phi_j = \frac{1}{2}\left[\sqrt{(K-1)^2 + 4Ku_j^{-1}\gamma} - (K-1)\right].$$

We thus conclude that if $u_j < \gamma$, then the load $y_j = r_j\,(\phi_j - 1)/\phi_j$ of the server j is given by

$$y_j = r_j\,\frac{\sqrt{(K-1)^2 + 4Ku_j^{-1}\gamma} - (K+1)}{\sqrt{(K-1)^2 + 4Ku_j^{-1}\gamma} - (K-1)}.$$

Let us now assume on the contrary that $u_j > \gamma$. If $y_j > 0$, then $\phi_j > 1$, which implies that $\phi_j(\phi_j + K - 1) > K$. However, according to the complementary slackness condition (C.2), the left hand side is just $Ku_j^{-1}\gamma$, and we therefore obtain that $\gamma > u_j$, i.e., a contradiction. As a consequence, if $u_j > \gamma$, then $y_j = 0$. Finally, we conclude from the above analysis that

$$y_j = \begin{cases} r_j\,\frac{\sqrt{(K-1)^2+4Ku_j^{-1}\gamma}-(K+1)}{\sqrt{(K-1)^2+4Ku_j^{-1}\gamma}-(K-1)} & \text{if } u_j < \gamma, \\ 0 & \text{otherwise.} \end{cases} \qquad \text{(C.3)}$$

Let $j^*(K)$ be such that $u_{j^*(K)} < \gamma \le u_{j^*(K)+1}$. Then the subset of servers used at the Nash equilibrium is $\mathcal{S}^*(K) = \{1, \ldots, j^*(K)\}$. Using (C.3), we deduce from $\sum_{j\in\mathcal{S}^*(K)} y_j = \bar\lambda$ that

$$\sum_{j\in\mathcal{S}^*(K)} r_j - \bar\lambda = \sum_{k\in\mathcal{S}^*(K)} \frac{2r_k}{\sqrt{(K-1)^2 + 4Ku_k^{-1}\gamma} - (K-1)},$$

i.e., $W(K,\gamma) = 0$, which implies that $\gamma = \gamma(K)$ according to Lemma 5.3. Moreover, since for a fixed K the function $W : [u_1, \infty) \to \mathbb{R}$ is decreasing in z, we deduce from $u_{j^*(K)} < \gamma(K) \le u_{j^*(K)+1}$ that $W(K, u_{j^*(K)+1}) \le 0 < W(K, u_{j^*(K)})$. ■

Appendix D. Proof of Lemma 6

Proof. From Corollary 6, we have $\mathcal{S}^*(K) \subset \mathcal{S}^*(1)$. For $j \in \mathcal{S}^*(1) \setminus \mathcal{S}^*(K)$, $\rho_j(K) = 0$. Hence (6) holds. It now remains to be shown that (6) holds for every $j \in \mathcal{S}^*(K)$. From (13),

$$y_j(K) = r_j \frac{\sqrt{(K-1)^2 + 4K\gamma(K)r_j/c_j} - (K+1)}{\sqrt{(K-1)^2 + 4K\gamma(K)r_j/c_j} - (K-1)}, \tag{D.1}$$

from which it follows that

$$\frac{y_j(K)}{r_j - y_j(K)} = \frac{\sqrt{(K-1)^2 + 4K\gamma(K)r_j/c_j} - (K+1)}{2}. \tag{D.2}$$

We shall now use the fact that $y_j(K)/(r_j - y_j(K))$ is increasing in $\gamma(K)$. Since $\gamma(K) \le \gamma(1)$, from (D.2),

$$\begin{aligned}
\frac{y_j(K)}{r_j - y_j(K)} &\le \frac{\sqrt{(K-1)^2 + 4K\gamma(1)r_j/c_j} - (K+1)}{2} \\
&= \frac{\sqrt{(K-1)^2 + 4K\gamma(1)r_j/c_j} - (K+1)}{2} \frac{\sqrt{(K-1)^2 + 4K\gamma(1)r_j/c_j} + (K+1)}{\sqrt{(K-1)^2 + 4K\gamma(1)r_j/c_j} + (K+1)} \\
&= 2K \frac{\gamma(1)r_j/c_j - 1}{\sqrt{(K-1)^2 + 4K\gamma(1)r_j/c_j} + (K+1)}.
\end{aligned}$$

Since $K - 1 \ge 0$ and $K + 1 \ge 2\sqrt{K}$, it yields

$$\frac{y_j(K)}{r_j - y_j(K)} \le \sqrt{K} \frac{\gamma(1)r_j/c_j - 1}{\sqrt{\gamma(1)r_j/c_j} + 1} = \sqrt{K}\left(\sqrt{\gamma(1)r_j/c_j} - 1\right). \tag{D.3}$$

From (D.2),

$$\frac{y_j(1)}{r_j - y_j(1)} = \frac{\sqrt{4\gamma(1)r_j/c_j} - 2}{2} = \sqrt{\gamma(1)r_j/c_j} - 1, \tag{D.4}$$

which upon substitution in (D.3) gives the desired result. ■