On the Equilibria of Resource-Sharing Games

Josu Doncel, Urtzi Ayesta, Olivier Brun, Balakrishna Prabhu

On the Equilibria of Resource-Sharing Games

J. Doncel[a,c], U. Ayesta[a,b,c,d], O. Brun[a,c], B.J. Prabhu[a,c]

[a] CNRS, LAAS, 7 avenue du colonel Roche, F-31400 Toulouse, France
[b] IKERBASQUE — Basque Foundation for Science, 48011 Bilbao, Spain
[c] Univ. de Toulouse, INSA, LAAS, F-31400 Toulouse, France
[d] UPV/EHU, Univ. of the Basque Country, 20018 Donostia, Spain

***Abstract*—We investigate the strategic interaction between a fixed number of users sharing the capacity of a processor operating with relative priorities. Each user chooses a payment, which corresponds to his priority level, and submits jobs of variable sizes according to a stochastic process. These jobs have to be completed before some user-specific deadline. They are executed on the processor and receive a share of the capacity that is proportional to the priority level. The users' goal is to choose priority levels so as to minimize their own payment, while guaranteeing that their jobs meet their deadlines. Given the complexity of the underlying queueing model, we develop an approximation based on heavy-traffic. We characterize the solution of the game under the heavy-traffic assumption and we numerically investigate the accuracy of the approximation. Our results show that the approximate solution captures accurately the structure of the equilibrium in the original game.**

I. Introduction

We are interested in studying the equilibria that arises in queueing games where a common resource is shared among multiple concurrent users. The study of strategic behavior in queueing systems has a long history and there is by now a broad literature, *cf.* [10] and [16] for monographs. A particular problem who has received a lot of attention deals with the strategic behavior of users in parallel servers, see for example [11], [5], [3]. In recent years, motivated by the rise of paid resource sharing systems like in cloud computing, researchers have started to investigate pricing schemes, where capacity of the server is shared simultaneously by all jobs present in the system, see for example [12] or [19]. For the case in which the the underlying queueing model has no priorities we refer to [4] and [21].

We are interested in studying the equilibria in a more complex scenario, where users may arrive at random and leave the system after getting service, and when the capacity allocated to each user is a function of the prices. More precisely, we assume that the capacity is shared according to the Discriminatory Processor Sharing (DPS) discipline introduced in [15], which is a multi-class generalization of the *egalitarian* Processor Sharing queue. The DPS discipline is a versatile model that captures the essential features of a system that implements service differentiation, see [1] for a recent survey. As we will see later in Section II the analysis of time-sharing queueing models with relative priorities like DPS is extremely challenging, and as a consequence results are scarce. In the DPS model, each user chooses a priority level, and the processor capacity is shared in proportion to the priority level of all jobs being executed. The higher the priority level chosen by the user, the higher the cost he will have to pay for the execution of his job on the processor. Each users submits jobs of variable sizes according to a stochastic process. All jobs in the system receive simultaneous service with a share of the capacity, determined by the DPS rule, that is proportional to the priority level of the user. The users' goal is to choose priority levels so as to minimize their own payment, while guaranteeing that the probability of their jobs not meeting their deadlines is below some threshold.

A central difficulty in the analysis of the equilibria of this game comes from the absence of a closed-form expression for the mean processing times of the jobs in a DPS system. For example, the mean unconditional sojourn time in the system is only known in the case of two classes with exponentially distributed service requirements. It is thus not surprising that results on strategic behavior under DPS are scarce. For example, in [13] the authors consider two types of applications in a DPS queue that compete to be served and they analyze how optimal prices can be found. A more recent work is [20], where the authors define a game for the DPS queue where each user seeks to minimize the sum of the expected processing cost and payment. Given the difficulty in analyzing the model, the authors propose a heavy-traffic approximation of the problem.

We follow a similar approach to [20] and consider an approximate approach using results from heavy-traffic theory. More precisely, we use results from [9] to obtain tractable expressions for the mean response time in the system. Even though of approximate nature, we believe that the heavy-traffic approach allows to derive interesting insights into the performance of the system. Our main contribution is to provide a complete characterization of the solution to the problem using the heavy-traffic approximation. In particular we show that classes can be ordered in a decreasing order with respect to the ratio between the mean size requirement and their constraints on the response time. We characterize the sufficient and necessary condition for the game to have a Nash equilibrium, and then show that this equilibrium is unique and fully characterize it. Interestingly, we show that in equilibrium, the prices that users pay decrease as the ratio above mentioned decreases. We then explain how the heavy-traffic solution can be used to obtain an approximate solution to the original problem. The numerical experiments illustrate that when the various classes have a similar ratio between the mean size and response time constraint, then the heavy-traffic approximation predicts satisfactorily the outcome (both in terms of equilibrium weights and performance) of the original game. However, an interesting situation arises when the disparity of the users increases or when the original game becomes unfeasible far from the saturation point. In this case, the error in predicting the equilibrium weights can be very significant, but in spite of this, the heavy-traffic approximation captures very accurately the structure of the equilibrium. Despite its limitations, we consider that our paper will represent a step further in the difficult area of pricing with time-sharing systems, even though more research is needed in order to enhance the understanding of resource-sharing games.

The rest of the paper is organized as follows. In Section II we describe the model. We present the game with constraints on the mean response time in Section III. In Section IV we analyse the game for the heavy-traffic regime and in Section V we study the game for an arbitrary load of the system. We present the numerical experiments and discuss the accuracy of our approximation in Section VI.

II. Game description

Consider a game in which a single server of unit capacity is shared among R classes (or users). We assume that the arrival process of jobs of each class i is Poisson with rate λ_i and that the service requirements of jobs are i.i.d. and have an arbitrary distribution with mean $\mathbb{E}(B_i)$ and second moment $\mathbb{E}\left(B_i^2\right)$. For the case of exponential service time distribution, we will use the notation $\mathbb{E}(B_i) = \mu_i^{-1}$ and $\mathbb{E}\left(B_i^2\right) = 2/\mu_i^2$. We define the total incoming traffic of the system by $\lambda = \sum_{i=1}^R \lambda_i$. Let $\rho_i = \lambda_i \mathbb{E}(B_i)$ be the load of class i and the total load of the system be $\rho = \sum_{i=1}^R \rho_i$.

The processing capacity of the server is shared amongst jobs according to the DPS discipline, that is, all jobs present in the system are served simultaneously at rates controlled by a vector of weights $\{g_i > 0; i = 1, \ldots, R\}$. If there are N_i jobs of class i present in the system, then class-i jobs are served at rate

$$r_i(N_1, \ldots, N_R) = \frac{g_i}{\sum_{j=1}^R g_j N_j}. \tag{1}$$

When all the weights are equal, DPS is equivalent to the standard PS discipline. By changing the weights, one can effectively control the instantaneous service rates of different job classes. For example, by setting the weight of a class close to infinity, one can give preemptive priority to this class. The possibility of providing different service rates to users of various classes makes DPS an appropriate model to study the performance of heterogeneous time-sharing systems. We note that a direct consequence of (1) is that the service rate every class gets for a vector $\theta(g_1, \ldots, g_R)$ is independent of the common factor θ.

We describe our game formulation in Subsection II-B. Prior to that, we briefly mention the main results on DPS that we need in this paper.

A. Main results on DPS

We denote by $T_i(\boldsymbol{g}; \rho)$ the random variable corresponding to the response time of a class-i job in a DPS queue for the vector of weights of $\boldsymbol{g} = (g_1, \ldots, g_R)$ when the load in the system is $\rho < 1$. The mean response time is denoted by $\overline{T}_i(\boldsymbol{g}; \rho) = \mathbb{E}(T_i(\boldsymbol{g}; \rho))$.

In a seminal paper, Fayolle et al. proved that for exponential service time distributions, the mean response time is the solution of a system of equations. For completeness we state their result:

Proposition 1 ([7]): In the case of exponentially distributed required service times, the unconditional average response times satisfy the following linear system of equations:

$$\overline{T}_k(\boldsymbol{g}; \rho) \left(1 - \sum_{j=1}^R \frac{\lambda_j g_j}{\mu_j g_j + \mu_k g_k}\right) - \sum_{j=1}^R \frac{\lambda_j g_j \overline{T}_j(\boldsymbol{g}; \rho)}{\mu_j g_j + \mu_k g_k} = \frac{1}{\mu_k}, \tag{2}$$

with $k = 1, \ldots, R$.

A solution to this system of equations is only known for the case $R = 2$. In this case the solution is :

$$\overline{T}_1(\boldsymbol{g}; \rho) = \frac{1}{\mu_1(1-\rho)} \left(1 + \frac{\mu_1 \rho_2 (g_2 - g_1)}{\mu_1 g_1 (1-\rho_1) + \mu_2 g_2 (1-\rho_2)}\right), \tag{3}$$

and

$$\overline{T}_2(\boldsymbol{g}; \rho) = \frac{1}{\mu_2(1-\rho)} \left(1 + \frac{\mu_2 \rho_1 (g_1 - g_2)}{\mu_1 g_1 (1-\rho_1) + \mu_2 g_2 (1-\rho_2)}\right). \tag{4}$$

Unfortunately, for general service time distributions the results are scarce. In [7] the authors showed that the derivative of the mean conditional (on the service requirement) response time of the various classes satisfies a system of integro-differential equations. Unfortunately a closed-form solution of this system of equations has been obtained only in the case of exponential distributions. To the best of our knowledge, there is no known tractable results on the distribution of the response time $T_i(\boldsymbol{g}; \rho)$.

To overcome this difficulty, in our approach we will approximate $T_i(\boldsymbol{g}; \rho)$ using a heavy-traffic characterization. It turns out that the scaled response time $(1-\rho)T_i(\boldsymbol{g}; \rho)$ has a proper distribution as $\rho \to 1$. The DPS queue in heavy-traffic was first considered in [9], (see also [17] and [18]). The result we require reads:

Proposition 2 ([9]): When scaled with $1-\rho$, the response time of class-i jobs has a proper distribution as $\rho \to 1$.

$$(1-\rho)\ T_i(\boldsymbol{g}; \rho) \xrightarrow{d} T_i(\boldsymbol{g}; 1) = X \cdot \frac{\mathbb{E}(B_i)}{g_i}, i = 1, \ldots, R \tag{5}$$

where $\xrightarrow{d}$ denotes convergence in distribution and X is an exponentially distributed random variable with mean

$$E(X) = \frac{\sum_k \lambda_k \mathbb{E}\left(B_k^2\right)}{\sum_k \lambda_k \mathbb{E}\left(B_k^2\right) \frac{1}{g_k}}. \tag{6}$$

Proposition 2 implies that for sufficiently high load, the response time distribution in a DPS queue can be approximated by an exponential random variable, that is,

$$T_i(\boldsymbol{g};\rho) \approxeq \frac{T_i(\boldsymbol{g};1)}{1-\rho} \stackrel{d}{=} \frac{\mathbb{E}(B_i)}{g_i(1-\rho)}X, \tag{7}$$

and for the mean response time we obtain that

$$\overline{T}_i(\boldsymbol{g};\rho) \approx \frac{\mathbb{E}(B_i)}{g_i(1-\rho)} \frac{\sum_k \lambda_k \mathbb{E}\left(B_k^2\right)}{\sum_k \lambda_k \mathbb{E}\left(B_k^2\right)\frac{1}{g_k}}. \tag{8}$$

In the above derivation, we have ignored a technical subtlety. Indeed, in order for (8) to be valid, one needs to establish that the heavy-traffic limit and expectation can be interchanged, namely, $\lim_{\rho\to 1}\overline{T}_i(\boldsymbol{g};\rho) = \mathbb{E}(\lim_{\rho\to 1} T_i(\boldsymbol{g};\rho))$. In [18] the authors performed numerical experiments to validate the validity of this interchange. In the rest of the paper we will assume that the interchange is valid.

In the case of identical weights g_i, the DPS queue is equivalent to the well-known *egalitarian* PS, which has been thoroughly studied, see for example [14] or [6]. For PS, it holds that $\overline{T}_i(\boldsymbol{g};\rho) = \mathbb{E}(B_i)/(1-\rho)$. From (6) and (5) we get that $\overline{T}_i(\boldsymbol{g};1) = \mathbb{E}(B_i)$, and it follows that the approximation $\overline{T}_i(\boldsymbol{g};\rho) = \frac{\overline{T}_i(\boldsymbol{g};1)}{1-\rho}$ is exact.

We can now introduce the game formulation we study in the paper.

B. Game formulation

We assume that the service provider (or the server) proposes to each class i the choice of its weight g_i in exchange of a payment per-unit-of-work proportional to the chosen weight. The quality-of-service metric of class i is the probability of its jobs missing a given deadline d_i. Class i then wants to ensure that this probability is below a certain threshold $\alpha_i \in (0,1)$ while paying as little as possible for this service. Formally, class-i solves the problem

$$\begin{aligned} &\min_{g_i \geq \epsilon} \quad \rho_i g_i && \text{(OPT-P)} \\ &\text{subject to} \quad \mathbb{P}\left(T_i(\boldsymbol{g};\rho) > d_i\right) \leq \alpha_i. && (9) \end{aligned}$$

The quantity ϵ is the minimum price a class has to pay in order to get access to the service.

As explained in Subsection II-A the probability of jobs missing a deadline in a DPS queue has no easy-to-compute closed-form expression. One could then consider a game in which the constraints are based on the mean response time of tasks. The optimization problem above then gets modified as follows

$$\begin{aligned} &\min_{g_i \geq \epsilon} \quad \rho_i g_i && \text{(OPT-M)} \\ &\text{subject to} \quad \overline{T}_i(\boldsymbol{g};\rho) \leq c_i, && (10) \end{aligned}$$

for $i = 1, \dots, R$.

The modified game (OPT-M) is not completely unrelated to the original game (OPT-P) as we shall argue next. Assuming that the load is high enough, we invoke the heavy-traffic approximation so:

$$\begin{aligned} \mathbb{P}\left(T_i(\boldsymbol{g};\rho) > d_i\right) &= \mathbb{P}\left(T_i(\boldsymbol{g};1) > (1-\rho)d_i\right), \\ &= e^{-\frac{(1-\rho)d_i}{\overline{T}_i(\boldsymbol{g};1)}}, \end{aligned}$$

implying that

$$\mathbb{P}\left(T_i(\boldsymbol{g};\rho) > d_i\right) \leq \alpha_i \iff -\frac{(1-\rho)\,d_i}{\overline{T}_i(\boldsymbol{g};1)} \leq \log \alpha_i. \tag{11}$$

Since $\alpha_i \in (0,1)$, we have $\log \alpha_i < 0$ and, hence, we obtain the following equivalent constraint $\overline{T}_i(\boldsymbol{g};1) \leq \tilde{c}_i = -\frac{(1-\rho)d_i}{\log \alpha_i}$.

Thus, we propose to use the heavy-traffic result given in Proposition 2 as an approximation to (OPT-P) and (OPT-M). We obtain the problem

$$\begin{aligned} &\min_{g_i \geq \epsilon} \quad \rho_i g_i && \text{(OPT-HT)} \\ &\text{subject to} \quad \overline{T}_i(\boldsymbol{g};1) \leq \tilde{c}_i. && (12) \end{aligned}$$

In the case $\tilde{c}_i = -\frac{(1-\rho)d_i}{\log \alpha_i}$ we will be approximating (OPT-P), and if $\tilde{c}_i = (1-\rho)c_i$ we will be approximating (OPT-M).

This approximation has the advantage that one can compute the scaled mean response time of a DPS queue in heavy-traffic $\overline{T}_i(\boldsymbol{g};1)$ for any service time distribution with finite second moment.

Our hope is that the solution of the game (OPT-HT) will give useful insights into the equilibrium properties of the (OPT-M) and (OPT-P). We emphasize that the benefit of the heavy-traffic approximation is that the mean response time formulae have a nice closed-form expressions even for general service time distributions whereas (OPT-M) has a simple structure only in case of exponentially distributed service times, while (OPT-P) does not appear to be tractable even for that case. In Section VI we investigate the accuracy of the approximation, and show that it always gives us the structure of the equilibrium and our approach is accurate when the users have similar mean size and mean service time characteristics.

Before going further, we give a couple of definitions.

Definition 1 (Feasibility): For fixed traffic conditions, a vector $\boldsymbol{c}$ of deadlines is feasible if there is a vector $\boldsymbol{g}$ of weights such that $\overline{T}_i(\boldsymbol{g};\rho) \leq c_i$, for all $i = 1, \dots, R$.

A game is feasible if its vector of deadlines is feasible. A related definition is that of an achievable vector which is defined as a vector of mean response times for which there exists a vector of weights leading to the desired mean response times. In [8], it was shown that the achievable region of a DPS queue with Poisson inputs and general service time distribution is a polytope with one equality constraint and $2^R - 2 + R$

inequality constraints. For an achievable vector $\mathbf{t}$, it follows that the set of deadline vectors which are coordinatewise larger that $\mathbf{t}$ are feasible. One can then compute the set of feasible deadline vectors by taking the union over feasible sets of each achievable vector. This definition simplifies considerably in the heavy-traffic limit, where it will be shown that the feasible region can be characterized using $R+1$ inequalities.

Definition 2: A class i will be considered *fair* if $\mathbb{E}(B_i)/c_i \leq (1-\rho)$.

In other words, a class i is fair if the response time it would obtain under PS, $\mathbb{E}(B_i)/(1-\rho)$, would satisfy its own constraint on the mean performance c_i. As we will see in Proposition 7, a sufficient condition for the game to be feasible is that all classes be fair.

Without loss of generality, we assume that the classes are ordered in decreasing order of $\mathbb{E}(B_k)/c_k$, i.e., if $i<j$, then $\mathbb{E}(B_i)/c_i \geq \mathbb{E}(B_j)/c_j$. We observe that the ratio $\mathbb{E}(B_k)/c_k$ is the maximum throughput of a class-i job with a service requirement equal to the mean. In the case of exponential service time distribution, it becomes $c_1\mu_1 \leq c_2\mu_2 \leq \cdots \leq c_R\mu_R$.

In the following sections, we shall first give results related to the game (OPT-M), then do a more detailed analysis of the approximation (OPT-HT), and finally compare the two results using numerical experiments.

III. Solution of (OPT-M)

A vector of weights $\boldsymbol{g}^{NE} = (g_1^{NE}, \ldots, g_R^{NE})$ is a Nash equilibrium (NE) for the game (OPT-M) if each class is paying the least possible amount while ensuring that its mean response time does not exceed its deadline. Thus, we can say that a vector of weights $\boldsymbol{g}^{NE}$ is a Nash equilibrium if for all $i = 1,\ldots,R$

$$g_i^{NE} = \text{argmin}\left\{g_i \geq \epsilon \ : \ \overline{T}_i(\boldsymbol{g};\rho) \leq c_i\right\}.$$

In [2] the authors showed that $\overline{T}_i(\boldsymbol{g};\rho)$ is decreasing with g_i and increasing in g_j for $j \neq i$. It then follows that, for a given i,

$$\begin{aligned} g_i^{NE} > \epsilon, &\quad \Rightarrow \overline{T}_i(\boldsymbol{g}^{NE};\rho) = c_i, \\ g_i^{NE} = \epsilon, &\quad \Rightarrow \overline{T}_i(\boldsymbol{g}^{NE};\rho) \leq c_i. \end{aligned}$$

Since $\overline{T}_i(\boldsymbol{g};\rho)$ is decreasing in g_i, a class which is paying more than ϵ is necessarily satisfying its constraint with equality. Otherwise, if it were to be satisfying the constraint with strict inequality, then it would pay less and still satisfy its deadline. On the other hand, a class which is paying the least possible price could be satisfying its deadline with strict inequality.

We notice that the dynamics of best-response are given by increasing the weight of class i when $\overline{T}_i(\boldsymbol{g};\rho) > c_i$ and decreasing the weight of class i when $\overline{T}_i(\boldsymbol{g};\rho) < c_i$ and $g_i > \epsilon$. Thus, we observe that if we start the best-response dynamics from a feasible point $\boldsymbol{g}$, the weights of all the classes always decrease. Moreover, after each best-response, the current vector of weights remains feasible because by decreasing its weight a class can only improve the mean response times of the other classes. Thus, we can conclude that the equilibrium will be obtained in a finite number of best-response steps.

Proposition 3: The dynamics of best-response converge to the Nash Equilibrium when started from a feasible point.

From Proposition 3, we immediately obtain the following corollary.

Corollary 1: If the game is feasible, there exists a Nash equilibrium.

For more precise results for this game, we focus on the game (OPT-M) for two classes, i.e. $R=2$, and exponentially distributed service time requirements (see (3) and (4)). We aim to find the minimum values of the weights such that the time constraints $\overline{T}_1(\boldsymbol{g};\rho) \leq c_1$ and $\overline{T}_2(\boldsymbol{g};\rho) \leq c_2$ are satisfied. We present the condition for the existence of a Nash equilibrium:

Proposition 4: The game is feasible if and only if the *deadlines* c_1 and c_2 satisfy $a(c_1) \geq b(c_2)$, where $a(c_1) = \frac{-\mu_1\rho_2-\mu_1(1-\rho_1)[\mu_1 c_1(1-\rho)-1]}{-\mu_1\rho_2+\mu_2(1-\rho_2)[\mu_1 c_1(1-\rho)-1]}$ and $b(c_2) = \frac{\mu_2\rho_1-\mu_1(1-\rho_1)[\mu_2 c_2(1-\rho)-1]}{\mu_2\rho_1+\mu_2(1-\rho_2)[\mu_2 c_2(1-\rho)-1]}$.

We now present the unique equilibrium of the game:

Proposition 5: If the game is feasible, the unique equilibrium is a vector of weights $\mathbf{g^{NE}}$ such that:

- if class 1 is fair, i.e. $(1-\rho)^{-1} \leq c_1\mu_1$, then $\mathbf{g^{NE}} = (\epsilon,\epsilon)$,
- otherwise, $\mathbf{g^{NE}} = (\frac{\epsilon}{a(c_1)},\epsilon)$, where $a(c_1)$ is as defined in Proposition 4.

Proposition 5 characterizes the unique equilibrium of the game when the number of classes is two with exponential service time requirements. We explain briefly the solution of the problem. Assuming feasibility, at least class 2 is fair. If class 1 is also fair, then $(g_1,g_2) = (\epsilon,\epsilon)$ is the equilibrium; however, if the mean response time of class 1 for PS weights exceeds its deadline c_1, the class 1 must pay $g_1 > \epsilon$ per unit-of-work to ensure that its time constraint is satisfied.

IV. Solution of (OPT-HT)

In this section we will characterize the solution to (OPT-HT). In Subsection IV-A we first investigate the achievable vector of performances in heavy-traffic, in Subsection IV-B we characterize the sufficient and necessary condition under which the game has a feasible solution and in Subsection IV-C we present the unique Nash-Equilibrium of the game.

A. Achievable Performances

The following proposition characterizes the achievable region of performances in heavy-traffic:

Proposition 6: A vector of performances of DPS $(t_1, \ldots, t_R)$ is achievable in heavy-traffic if and only if

$$\sum_k \lambda_k \frac{\mathbb{E}(B_k^2)}{\mathbb{E}(B_k)} t_k = \sum_{j=1}^{R} \lambda_j \mathbb{E}\left(B_j^2\right). \qquad (13)$$

Proof: See appendix B-A ■

Remark 1: In the case $\rho < 1$, in [8] the authors characterized the achievable region of DPS under exponential service requirements. It can be easily checked, than the vector space obtained in [8] converges to the hyperplane (13) as the load increases to 1.

B. Feasibility

As introduced in Definition 1, we say that the game is feasible if there exists at least one vector of weights $\boldsymbol{g}$ such that all time constraints are satisfied. Using the same arguments that for Proposition 3 and Corollary 1, this is easily seen to imply the existence of a Nash equilibrium for (OPT-HT). In the following result, we give a sufficient and necessary condition for the game to be feasible.

Proposition 7: The game (OPT-HT) is feasible if and only if $\sum_i \lambda_i \mathbb{E}\left(B_i^2\right)\left(\frac{\tilde{c_i}}{\mathbb{E}(B_i)} - 1\right) \geq 0$.

Proof: See appendix B-A ■

Interestingly, we observe that a sufficient condition for the game to be feasible is that in heavy-traffic all classes be *fair*. Note that $\overline{T}_i(\boldsymbol{g}^{PS}; 1) = \mathbb{E}(B_i)$, thus from Proposition 7 if $\overline{T}_i(\boldsymbol{g}^{PS}; 1) \leq \tilde{c}_i$, $\forall i$, then the game is feasible.

C. Characterization of the Nash Equilibrium

We now show that if the game is feasible, the Nash equilibrium is unique and fully characterize it. The following theorem states our main result:

Theorem 1: If the problem is feasible, the unique Nash equiliblibrium is

$$\begin{aligned} g_i^{NE} &= \epsilon \frac{\tilde{t_m}/\mathbb{E}(B_m)}{\tilde{c_i}/\mathbb{E}(B_i)}, \text{ for all } i < m \\ g_i^{NE} &= \epsilon, \text{ for all } i \geq m, \end{aligned}$$

where $m = 1, \ldots, R$ is the minimum value such that there exists a value $\tilde{t_m} \leq \tilde{c_m}$ verifying

$$\frac{\tilde{t_m}}{\mathbb{E}(B_m)} = \frac{\sum_{k=1}^{R} \lambda_k \mathbb{E}\left(B_k^2\right) - \sum_{k=1}^{m-1} \lambda_k \frac{\mathbb{E}\left(B_k^2\right)}{\mathbb{E}(B_k)} \tilde{c_k}}{\sum_{k=m}^{R} \lambda_k \mathbb{E}\left(B_k^2\right)}. \qquad (14)$$

In the particular case where all classes are fair, then the solution is very simple:

Corollary 2: If all the users are fair in heavy-traffic, i.e., $\overline{T}_i(\boldsymbol{g}^{PS}; 1) \leq \tilde{c}_i$, then the equilibrium is $\mathbf{g^{NE}} = (\epsilon, \ldots, \epsilon)$.

The following corollary shows that the price paid by classes in the Nash equilibrium decreases as the ratio $\mathbb{E}(B_k)/c_k$ decreases, namely:

Corollary 3: Let $\mathbf{g^{NE}} = (g_1^{NE}, \ldots, g_R^{NE})$ be the vector of weights in the equilibrium. We have

$$g_1^{NE} \geq g_2^{NE} \geq \cdots \geq g_{R-1}^{NE} \geq \epsilon$$

Proof: It follows from the result of Theorem 1 and our assumption on the ordering of the classes. ■

It is interesting to observe that the ordering of classes in the equilibrium do not depend on the arrival or second moment of the distributions. Instead, the key parameter is the ratio $\mathbb{E}(B_k)/c_k$, which can be interpreted as the throughput of a class k. Thus, classes will deviate from the minimum weight in decreasing order with respect to the throughput they expect to obtain from the system.

V. Approximating (OPT-M)

In this section we explain how the results of Section IV can be used to obtain insights into the solution of games (OPT-P) and (OPT-M). As explained in Section II-B, provided that ρ is sufficiently large for the approximation $\overline{T}_i(\boldsymbol{g}; \rho) = \frac{\overline{T}_i(\boldsymbol{g}; 1)}{1-\rho}$ to be valid, the results established for game (OPT-HT) can be applied to approximate the solution of (OPT-P) by setting $\tilde{c}_i = -(1-\rho)d_i/\log \alpha_i$ and the solution of (OPT-M) by setting $\tilde{c}_i = (1-\rho)c_i$. We will focus on the case (OPT-M). This choice allows to evaluate numerically the accuracy of the approximation using the formulas presented in Section II-A.

A. Feasibility when $\rho < 1$

It follows directly from Proposition 7 that a necessary and sufficient condition for the (approximate) feasibility of (OPT-M) is

$$\sum_i \lambda_i \mathbb{E}\left(B_i^2\right)\left(\frac{c_i}{\mathbb{E}(B_i)/(1-\rho)} - 1\right) \geq 0. \qquad (15)$$

This implies that if all users are fair, then the game is feasible.

B. The Nash Equilibrium for $\rho < 1$

Extending Theorem 1 to the case $\rho < 1$ with $\tilde{c}_i = c_i(1-\rho)$, we obtain that the Nash-Equilibrium of (OPT-M) can be approximated by

$$g_i^{NE} = \epsilon \frac{t_m/\mathbb{E}(B_m)}{c_i/\mathbb{E}(B_i)}, \text{ for all } i < m$$

$$g_i^{NE} = \epsilon, \text{ for all } i \geq m,$$

where $m = 1, \ldots, R$ is the minimum value such that there exists a value $t_m \leq c_m$ verifying

$$\frac{t_m}{\mathbb{E}(B_m)} = \frac{\sum_{k=1}^{R} \frac{\lambda_k \mathbb{E}\left(B_k^2\right)}{(1-\rho)} - \sum_{k=1}^{m-1} \lambda_k \frac{\mathbb{E}\left(B_k^2\right)}{\mathbb{E}(B_k)} c_k}{\sum_{k=m}^{R} \lambda_k \mathbb{E}\left(B_k^2\right)}. \qquad (16)$$

Note that if class 1 is fair, then all users are fair. In this case, the right-hand side of (16) is upper-bounded by $(1-\rho)^{-1}$, implying that $c_1 \geq \frac{\mathbb{E}(B_1)}{1-\rho} \geq t_1$, so that $m = 1$. Thus, if class 1 is fair, the approximate equilibrium corresponds to the PS solution $g_i^{NE} = \epsilon$ for all i, which is clearly the exact equilibrium.

It is interresting to compare the above approximate characterization of the Nash equilibrium with the exact result given in Proposition 5 in the case of two users and exponential service time distributions. As discussed above, if class 1 is fair, then the approximate and exact equilibria coincide and correspond to the PS queue. Otherwise, the equilibrium in both instances have the same form, i.e., $\mathbf{g^{NE}} = (g_1^{NE}, \epsilon)$, with $g_1^{NE} > \epsilon$.

Observe that $t_m/\mathbb{E}(B_m)$ increases with ρ, and define ρ_{min} and ρ_{max} as the threshold values such that:

- if $\rho \leq \rho_{min}$, then all classes are paying the minimum price ϵ,
- if $\rho_{min} < \rho \leq \rho_{max}$ the game is feasible and there is at least one class paying more than ϵ,
- if $\rho > \rho_{max}$, the game is not feasible.

C. Characterization of ρ_{min}

As discussed above, if class 1 is fair, that is if $\frac{\mathbb{E}(B_1)}{c_1} \leq 1-\rho$, then all users are paying the minimum price at the equilibrium. As a consequence, the minimum value ρ_{min} such that at least one user pays more than ϵ is obtained when $\frac{\mathbb{E}(B_1)}{c_1} = 1 - \rho_{min}$, that is for

$$\rho_{min} = 1 - \frac{\mathbb{E}(B_1)}{c_1}. \tag{17}$$

We emphasize that, since we have not used heavy-traffic results to characterize ρ_{min}, the above expression of ρ_{min} is the exact threshold where class 1 starts paying more than ϵ. We also note from (17) that if the throughput $\mathbb{E}(B_1)/c_1$ of class 1 is close to 0, then ρ_{min} is close to 1, implying that the PS solution $(\epsilon, \dots, \epsilon)$ corresponds to the equilibrium for a large range of utilization rates.

D. Characterization of ρ_{max}

We obtain an approximate value for ρ_{max} using the heavy-traffic characterization. From (15) it follows that

$$\rho_{max} = \frac{\sum_{i=1}^{R} \lambda_i \mathbb{E}\left(B_i^2\right)\left(\frac{c_i}{\mathbb{E}(B_i)} - 1\right)}{\sum_{i=1}^{R} \lambda_i \frac{\mathbb{E}\left(B_i^2\right)}{\mathbb{E}(B_i)} c_i} \tag{18}$$

We emphasize that this approximation of ρ_{max} is only valid if the real value is sufficiently close to 1 for the approximation $\overline{T}_i(\boldsymbol{g};\rho) = \frac{\overline{T}_i(\boldsymbol{g};1)}{1-\rho}$ to be accurate.

E. Identical throughput expectations

A particular case of interest is obtained when all classes have the same throughput expectations. In this case, we can characterize exactly the value of ρ_{max}.

Proposition 8: If $\mathbb{E}(B_i)/c_i = k < 1$ for all i, then the unique equilibrium of the game is the PS solution $(\epsilon, \dots, \epsilon)$ for $\rho \leq 1-k$, and the game is not feasible for $\rho > 1-k$.

Proof: If all users had the same weights (so the equilibrium were PS), we would have that $\mathbb{E}(B_i)/c_i = 1-\rho$, for all i. Since $\mathbb{E}(B_i)/c_i = k < 1$, we conclude that if $\rho \leq 1-k$ then $(\epsilon, \dots, \epsilon)$ is the unique equilibrium. When $\rho = 1-k$ we have $c_i = \mathbb{E}(B_i)/(1-\rho)$, $\forall i$, that is, c_i, $\forall i$, is equal to the sojourn time in a PS queue. This means that the vector $(c_1, \dots, c_R)$ lies in the achievable region of the system, and as soon as ρ increases further the game becomes infeasible. ■

We thus have $\rho_{min} = \rho_{max} = 1-k$. From (18) we also conclude that in this case the approximation of ρ_{max} gives the exact value $1-k$.

VI. Numerical Experiments

In this section, we present the results of numerical experiments in order to compare the equilibrium of the game (OPT-M) (which we call the original problem) with that of the heavy-traffic approximation (OPT-HT). Our main observation from the experiments that we conducted is that while in certain cases the error in weights can be substantial, the proposed heavy-traffic approximation is good at predicting the set of classes that pay a higher than minimum price at the equilibrium, and the mean response times of the classes paying the minimum price.

Without loss of generality, the minimum weight ϵ is set to 1 in all the experiments.

A. Exponential service time distribution

First, we present the results for exponentially distributed service times. In the first set of experiments, there are two players with deadlines $c_1 = 5$ and $c_2 = 6$, and the mean service times $\mu_1 = 2$ and $\mu_2 = 3$. Note that $c_1\mu_1 = 10 < c_2\mu_2 = 18$. We now vary the total system load starting from 0.8 until the system becomes unfeasible while maintaining $\rho_1 = 0.3\rho$ and $\rho_2 = 0.7\rho$. For each value of load, the equilibrium is computed using the best-response algorithm. In order to compute the best-response of a class for the original problem, the mean response time is computed from the system of equations presented in Proposition 1.

In the bottom subfigure of Figure 1, we plot the equilibrium weights for both the original problem and the HT approximation as a function of the total system load. The percentage relative error[1] between the two is shown in the top subfigure

[1]The percentage relative error for class i is given by $\left|\frac{g_i^{SYS} - g_i^{HT}}{g_i^{SYS}}\right| \times 100$, where g_i^{SYS} (resp., g_i^{HT}) is its equilibrium weight for the original problem (resp. HT approximation).

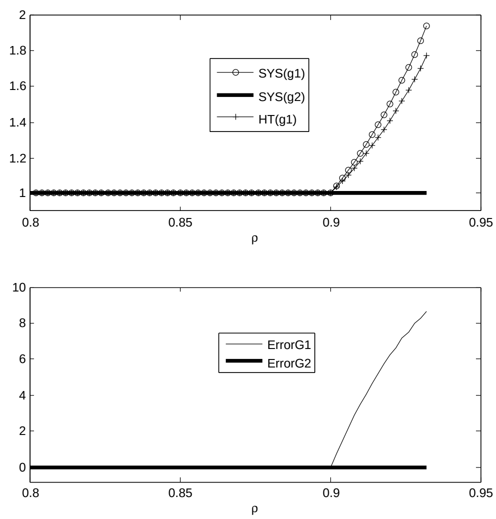

Fig. 1: Comparison of equilibrium weights (above) and the corresponding percentage relative error (below) as a function of the total system load. $R = 2$ and exponential service time distribution.

of the same figure.

Both problems become unfeasible for $\rho > 0.93$, so the data is restricted to $\rho \leq 0.93$. When the load of the system is between 0.9 and 0.93 we observe in Figure 1 (below) that the equilibrium of the heavy-traffic result approximates very well the equilibrium of the original problem. In particular, the heavy-traffic approximation follows the same increasing trend of the equilibrium weight of class 1 as that of the original problem. The error of class 1 users is small, while there is no error for the users of class 2. We see in Figure 1 (above) that the maximum percentage relative error is 9% and it is achieved when $\rho = 0.93$.

In the second set of experiments, we scale the deadlines by $(1-\rho)^{-1}$, that is, the deadline of user i, $c_i = \frac{\tilde{c}_i}{(1-\rho)}$ for some fixed $\tilde{c}_i$. This reflects that class i jobs is aware that the performance worsens as ρ increases, and is willing to adjust its deadline correspondingly. When the deadlines are scaled with $(1-\rho)^{-1}$, the constraint on the mean response time of player i for the original problem becomes $\overline{T}_i(\boldsymbol{g}; \rho) \leq \frac{\tilde{c}_i}{1-\rho}$, and that for the heavy-traffic approximation becomes $\overline{T}_i(\boldsymbol{g}; 1) \leq \tilde{c}_i$. Note that the latter constraint does not change with ρ.

We set the parameters to : $\mu_1 = 2$ and $\mu_2 = 3$, $\rho_1 = 0.3\rho$, and $\rho_2 = 0.7\rho$, with the scaled deadlines being $\tilde{c}_1 = 0.3$ and $\tilde{c}_2 = 0.7$. In Figure 2, we present the accuracy of the heavy-traffic approximation as $\rho \to 1$. As expected, the error in the weight of class 1 reduces as the load tends to 1.

In the next set of experiments, we look at a four-player game with parameters $\mathbf{c} = [10, 15, 25, 45]$ and $\boldsymbol{\mu} = [1, 2, 4, 9]$. The loads of individual classes are in the proportion $[1/3, 1/6, 1/4, 1/4]$, that is $\rho_1 = \rho/3$, $\rho_2 = \rho/6$, and so on. In Figure 3, the equilibrium weights are plotted in the top

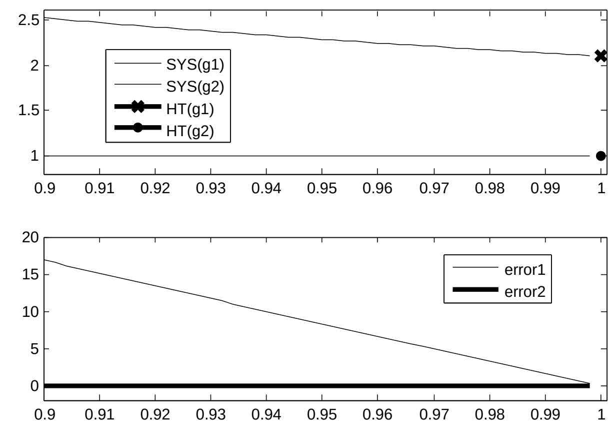

Fig. 2: Comparision of equilibrium weights (above) and the corresponding percentage relative error (below) as a function of the total load, and the deadlines of the two classes are scaled by $(1-\rho)^{-1}$. $R = 2$ exponential service time distribution.

subfigure, the corresponding error is plotted in the middle subfigure, and in the bottom subfigure we plot the error in the equilibrium mean response times of the classes. We did not plot the weights and the error for class 4 because its weight is always 1 in both the systems.

The trend in the four-player plots is similar to that of the two-player example in which the deadlines are not scaled. Until $\rho < 1 - \frac{1}{c_1\mu_1} = 0.9$, PS is a feasible solution and all the classes pay the minimum price. As the load increases further and moves closer to the ρ_{max} of the original system, one or more classes start to pay, and the error in the equilibrium starts to increase. The main observation here is that, even though the maximum error in the weights is around 19%, the maximum error in the mean response times is less than 2%.

It is rather surprising that mean response times in the heavy-traffic approximation are so close to that in the original game. As another example in support of this observation, we set the parameters to : $\mathbf{c} = [5/3, 5/4, 10, 100]$ and $\boldsymbol{\mu} = [1, 2, 8, 12]$, with the proportion of loads being the same as before. The main difference with the previous example is that there is much more heterogeneity in the deadlines and the $c\mu$ of the classes. The first two classes have a much smaller deadline and the range of values of $c_i\mu_i$ is now much larger as well compared to the previous example. The plots are shown In Figure 4. Note that the scale is logarithmic for the vertical axis in the top subfigure, and that in the top and middle subfigure, the data is plotted only for the first two classes because the other two classes are always paying the minimum price.

The error in the weight of class 1 is close to 60% at $\rho = 0.5$ and increases to almost 100% at $\rho = 0.9$. The error for class 2 is similarly large for loads close to $\rho = 0.9$ which means that the prediction is poor. For example, for $\rho = 0.9$ the weights are : $g_1^{SYS} = 7287.8$, $g_1^{HT} = 45.2$, $g_2^{SYS} = 2120.6$, and $g_2^{HT} = 30.14$. That is, the heavy-traffic approximation predicts a weight of 45.2 for class 1 whereas the weight in the original system is 7287.8. There is a similar disparity in

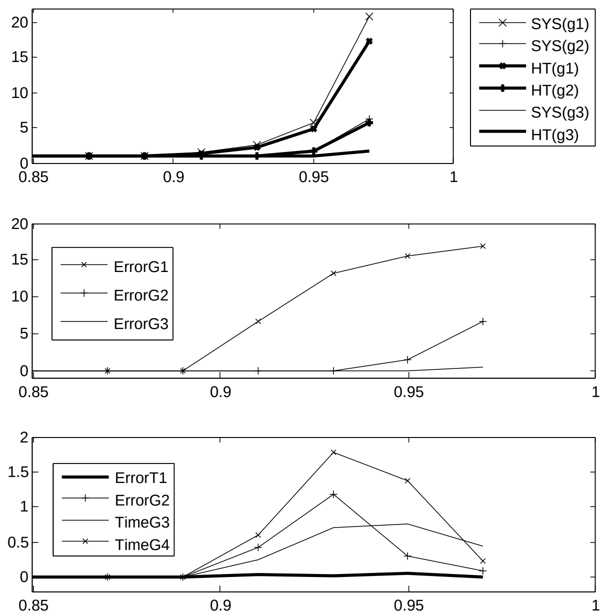

Fig. 3: Comparison of equilibrium weights (above) and the corresponding percentage relative error (below) as a function of the total system load. $R = 4$ and exponential service time distribution. $\mathbf{c} = [10, 15, 25, 45]$, $\boldsymbol{\mu} = [1, 2, 4, 9]$.

the weight of class 2. A similar observation on the negative impact of heterogeneity on the error was also made in [20]. On the other hand and in spite of the large disparity in the weights, the maximum error in the mean response times is negligible. For classes 1 and 2 it is not surprising that the error is small because their mean response times are equal to their constraint since they are paying more than the minimum price. For classes 3 and 4, the mean response times are strictly smaller than their constraint, and their values in the original system and as predicted by the heavy-traffic approximation are : $T_3^{SYS} = 9$, $T_3^{HT} = 9.41$, $T_4^{SYS} = 6.7$, and $T_4^{HT} = 6.28$, which are reasonably close.

B. Hyper-exponential service requirements

Finally, in this subsection, compare the approximation for a two-player game with hyper-exponentially distributed service times.

While there is no explicit expression for mean response time in DPS with service time distributions other than the exponential distribution, for the hyper-exponential distribution, a simple trick can be used to compute the mean response times using those of the exponential distribution. For example, consider a two-class DPS queue with hyper-exponential distribution of two phases each. The service rates of the phases are (μ_1, μ_2) for class 1 and (μ_3, μ_4) for class 2. and the arrival rates to these phases are (λ_1, λ_2) for class 1 and (λ_3, λ_4) for class 2. In order to compute the mean response time in this queue when the weights are $\mathbf{g} = (g_1, g_2)$, one first computes the mean response time in a four-class DPS queue

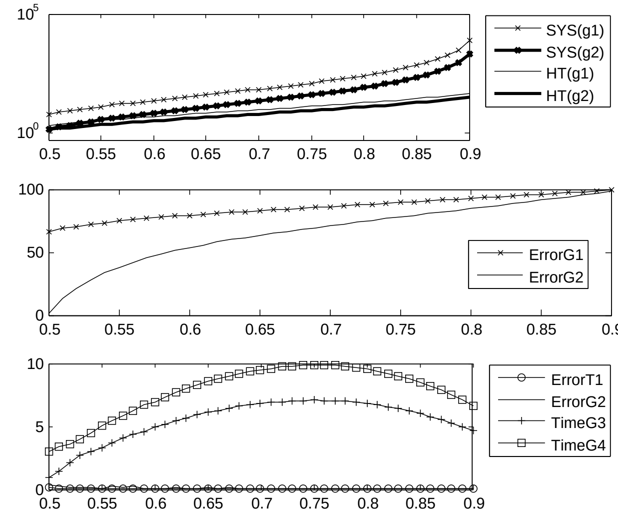

Fig. 4: Comparison of equilibrium weights (above) and the corresponding percentage relative error (below) as a function of the total system load. $R = 4$ and exponential service time distribution. $\mathbf{c} = [5/3, 5/4, 10, 100]$, $\boldsymbol{\mu} = [1, 2, 8, 12]$.

with exponential distribution and weights $\mathbf{g} = (g_1, g_1, g_2, g_2)$. The arrival rate of class i in this queue is λ_i, and the rates of the exponential distribution of class i is taken to be μ_i. The mean respone time of class i in the DPS queue with hyper-exponential distribution is then:

$$\overline{T}_1^{HEXP}(\boldsymbol{g}; \rho) = \frac{\lambda_1}{\lambda_1 + \lambda_2} \overline{T}_1(\boldsymbol{g}; \rho) + \frac{\lambda_2}{\lambda_1 + \lambda_2} \overline{T}_2(\boldsymbol{g}; \rho),$$
$$\overline{T}_2^{HEXP}(\boldsymbol{g}; \rho) = \frac{\lambda_3}{\lambda_3 + \lambda_4} \overline{T}_3(\boldsymbol{g}; \rho) + \frac{\lambda_2}{\lambda_3 + \lambda_4} \overline{T}_4(\boldsymbol{g}; \rho).$$

Using the above trick, the equilibrium weights were computed for the two-player DPS game with parameters: $\mu_1 = 1$, $\mu_2 = 3$, $\mu_3 = 5$, $\mu_4 = 7$, and deadlines $c_1 = 5$ and $c_2 = 7$. The fraction of the load of class 1 was $(\rho_1, \rho_2) = (\frac{\rho}{6}, \frac{\rho}{3})$, and for class 2 it was $(\rho_3, \rho_4) = (\frac{\rho}{4}, \frac{\rho}{4})$.

In Figure 5 we depict variation of the weights and the relative error when the total load of the system changes. Finally, we observe that the error on the equilibrium is similar to that of the exponentially distributed service times.

VII. Acknowledgementes

This work has been partially supported by grant ANR-11-INFR-001.

References

[1] S. Aalto, U. Ayesta, S. Borst, V. Misra, and R. N. nez Queija, "Beyond Processor Sharing," *ACM SIGMETRICS Performance Evaluation Review*, vol. 34, pp. 36–43, 2007.

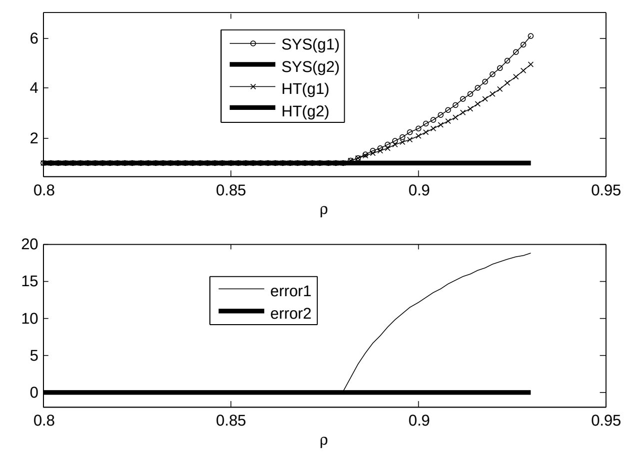

Fig. 5: Comparison of equilibrium weights (above) and the corresponding percentage relative error (below) as a function of the total system load. $R = 2$ and hyper-exponential service time requirements.

[2] K. Avrachenkov, U. Ayesta, P. Brown, and R. Núñez-Queija, "Discriminatory processor sharing revisited," in *Proceedings of IEEE INFOCOM*, 2005.

[3] U. Ayesta, O. Brun, and B. J. Prabhu, "Price of anarchy in non-cooperative load balancing," in *Proceedings of IEEE Infocom*, 2010.

[4] S. Birmiwal, R. R. Mazumdar, and S. Sundaram, "Predictable revenue under processor sharing," in *CISS*, 2012, pp. 1–6.

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

[6] J. Cohen, "The multiple phase service network with generalized processor sharing," *Acta Informatica*, vol. 12, pp. 245–284, 1979.

[7] G. Fayolle, I. Mitrani, and R. Iasnogorodski, "Sharing a processor among many job classes," *Journal of the ACM*, vol. 27, no. 3, pp. 519–532, 1980.

[8] E. Gelenbe and I. Mitrani, *Analysis and Synthesis of Computer Systems*. London: Academic Press, 1980.

[9] S. Grishechkin, "On a relationship between processor sharing queues and crump-mode-jagers branching processes," *Adv. Appl. Prob.*, vol. 24, no. 3, pp. 653–698, 1992.

[10] R. Hassin and M. Haviv, *To Queue or not to Queue: Equilibrium Behavior in Queueing Systems*. Boston etc.: Kluwer Academic Publishers, 2003.

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

[12] M. Haviv and J. van der Wal, "Equilibrium strategies for processor sharing and random queues with relative priorities," *Probability in the Engineering and Informational Sciences*, vol. 11, pp. 403–412, 1997.

[13] Y. Hayel and B. Tuffin, "Pricing for heterogeneous services at a discriminatory processor sharing queue," in *Proceedings of Networking*, 2005.

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

[15] L. Kleinrock, "Time-shared systems: A theoretical treatment," *Journal of the ACM*, vol. 14, no. 2, pp. 242–261, 1967.

[16] N. Nisan, T. Roughgarden, E. Tardos, and V. Vazirani, *Algorithmic Game Theory*. New York, NY, USA: Cambridge University Press, 2007.

[17] K. Rege and B. Sengupta, "Queue length distribution for the discriminatory processor sharing queue," *Operations Research*, vol. 44, no. 4, pp. 653–657, 1996.

[18] I. Verloop, U. Ayesta, and R. Núñez-Queija, "Heavy-traffic analysis of a multiple-phase network with discriminatory processor sharing," *Operations Research*, vol. 59, no. 3, pp. 648–660, 2011.

[19] M. Vojnovic and T. Nguyen, "The weighted proportional sharing mechanisms," in *ACM SIGMETRICS*, 2011.

[20] Y. Wu, L. Bui, and R. Johari, "Heavy traffic approximation of equilibria in resource sharing games," *Selected Areas in Communications, IEEE Journal on*, vol. 30, no. 11, pp. 2200–2209, 2012.

[21] Y. Yang, R. Ma, and J. Lui, "Price differentiation and control in the kelly mechanism," *Performance Evaluation*, vol. to appear, 2013.

APPENDIX A
PROOFS OF THE GAME WITH CONSTRAINTS ON THE MEAN RESPONSE TIME

A. Feasibility

First, we introduce this result we use to characterize the feasibility of the game with two classes of jobs with exponential service requirements:

Lemma 1: For any vector $\mathbf{g} = (g_1, g_2)$ there exists $a(c_1) = \frac{-\mu_1\rho_2-\mu_1(1-\rho_1)[\mu_1 c_1(1-\rho)-1]}{-\mu_1\rho_2+\mu_2(1-\rho_2)[\mu_1 c_1(1-\rho)-1]}$ and $b(c_2) = \frac{\mu_2\rho_1-\mu_1(1-\rho_1)[\mu_2 c_2(1-\rho)-1]}{\mu_2\rho_1+\mu_2(1-\rho_2)[\mu_2 c_2(1-\rho)-1]}$ such that

$$\begin{aligned} \overline{T}_1(\boldsymbol{g};\rho) = c_1 \quad &\Longleftrightarrow \quad \frac{g_2}{g_1} = a(c_1), \\ \overline{T}_2(\boldsymbol{g};\rho) = c_2 \quad &\Longleftrightarrow \quad \frac{g_2}{g_1} = b(c_2). \end{aligned}$$

Proof: It follows from equations (3) and (4). ∎

Since we know that when g_i increases the mean response time of class-i jobs decreases, then it follows from lemma 1 the following property:

Corollary 4: For any vector (g_1, g_2) the values of $a(c_1)$ and $b(c_2)$ defined in lemma 1 verify that

$$\begin{aligned} \overline{T}_1(\boldsymbol{g};\rho) \le c_1 \quad &\Longleftrightarrow \quad \frac{g_2}{g_1} \le a(c_1), \\ \overline{T}_2(\boldsymbol{g};\rho) \le c_2 \quad &\Longleftrightarrow \quad \frac{g_2}{g_1} \ge b(c_2). \end{aligned}$$

For a given values of the *deadlines* c_1 and c_2, we obtain $a(c_1)$ and $b(c_2)$, so that we can state that all the points (g_1, g_2) such that $\overline{T}_1(\boldsymbol{g};\rho) \le c_1$ and $\overline{T}_2(\boldsymbol{g};\rho) \le c_2$ must satisfy $a(c_1) \ge \frac{g_2}{g_1} \ge b(c_2)$. Thus, the desired result follows directly from this property because if $a(c_1) \le b(c_2)$ there are no weights (g_1, g_2) verifying both time constraints.

B. Nash Equilibrium

Proof of proposition 5:

To characterize the equilibrium, we need to assume that the solution of the game exist which means that c_1 and c_2 are such that $a(c_1) \ge b(c_2)$. It is obvious that if the game is not feasible there is no solution of the game.

If user of class 1 is fair, then $\frac{c_1}{\mathbb{E}(B_1)} \geq (1-\rho)^{-1}$ and hence we see that $\frac{c_2}{\mathbb{E}(B_2)} \geq (1-\rho)^{-1}$ because we have assumed that $\frac{c_1}{\mathbb{E}(B_1)} \leq \frac{c_2}{\mathbb{E}(B_2)}$. We notice that

$$\frac{c_i}{\mathbb{E}(B_i)} \geq (1-\rho)^{-1},\ i=1,2 \iff \overline{T}_i(\boldsymbol{g}^{PS};\rho) \leq c_i,\ i=1,2,$$

This means that if both classes are fair, then the Processor Sharing weights satisfy both the time constraints, and thus the equilibrium is $g_i^{NE} = \epsilon$, for $i=1,2$, since both classes have the minimum weight possible and the time constraints are satisfied.

If $\frac{c_1}{\mathbb{E}(B_1)} < (1-\rho)^{-1}$, then we can ensure that $(1-\rho)^{-1} \leq \frac{c_2}{\mathbb{E}(B_2)}$ since we assumed feasibility (if $\frac{c_i}{\mathbb{E}(B_i)} < (1-\rho)^{-1}$, for $i=1,2$, then $\overline{T}_i(\boldsymbol{g}^{PS};\rho) > c_i$, $i=1,2$, and according to corollary 4 and proposition 4, there is no solution of the game).

We observe that, if only users of class 2 are fair, then (ϵ,ϵ) is not the equilibrium because class 1 do not satisfy its time constraint with PS queue weights. In fact, the equilibrium is achieved in $\boldsymbol{g}=(g_1,\epsilon)$, where g_1 is such that $\overline{T}_1(\boldsymbol{g};\rho)=c_1$ and $\overline{T}_2(\boldsymbol{g};\rho) \leq c_2$ because g_1 is the minimum weight satisfying its time constraint and ϵ is the minimum weight possible for class 2 and its time constraint is verified.

According to lemma 1, we state that $\overline{T}_1(\boldsymbol{g};\rho) = c_1$ is obtained when $\frac{g_2}{g_1} = a(c_1)$ which yields to $g_1 = \frac{\epsilon}{a(c_1)}$. ■

Appendix B
Proofs of the heavy-traffic approximation game

A. Achievable Performances

Proof of Proposition 6: We observe that (13) holds when $\overline{T}_i(\boldsymbol{g};\rho) = t_i$ and using the expression of the scaled mean response time in the heavy-traffic regime of a class-i job of Proposition 2.

The other implication is proven if we show that any vector $t \in \mathbb{R}_+^R$ such that $\overline{T}_i(1) = t_i$ satisfying (13) can be obtained by a vector of DPS weights $\mathbf{g}$.

From Proposition 2 we observe that the scaled mean response time in heavy-traffic for all $i \neq j$ verifies that

$$\frac{\overline{T}_i(\boldsymbol{g};1)}{\overline{T}_j(\boldsymbol{g};1)} = \frac{g_j/\mathbb{E}(B_j)}{g_i/\mathbb{E}(B_i)}. \tag{19}$$

It then follows that all the components of the vector $\mathbf{g}$ must verify that $\frac{g_i}{g_j} = \frac{t_j/\mathbb{E}(B_j)}{t_i/\mathbb{E}(B_i)}$. In fact, we present that a vector satisfying the latter holds that $\overline{T}_i(\boldsymbol{g};1)=t_i$, for $i=1,\dots,R$. From the result of Proposition 2 and 19, we obtain

$$\overline{T}_i(\boldsymbol{g};1) = \mathbb{E}(B_i)\frac{\sum_k \lambda_k \mathbb{E}\left(B_k^2\right)}{\sum_k \lambda_k \mathbb{E}\left(B_k^2\right)\frac{t_k/\mathbb{E}(B_k)}{t_i/\mathbb{E}(B_i)}} = t_i \frac{\sum_k \lambda_k \mathbb{E}\left(B_k^2\right)}{\sum_k \lambda_k \frac{\mathbb{E}\left(B_k^2\right)}{\mathbb{E}(B_k)} t_k}$$

and it follows from (13) that $\overline{T}_i(\boldsymbol{g};1) = t_i$, $i=1,\dots,R$. ■

B. Feasibility

Proof of proposition 7:

If the problem is feasible in heavy-traffic there exists a vector $\mathbf{t}=(t_1,\dots,t_R)$ such that $\overline{T}_i(\boldsymbol{g};1)=t_i \leq c_i$ for all i and the condition of proposition 6 is verified. Then, since $t_i \leq c_i$ for all i, it follows that $\sum_i \lambda_i \frac{\mathbb{E}\left(B_i^2\right)}{\mathbb{E}(B_i)} c_i \geq \sum_k \lambda_k \mathbb{E}\left(B_k^2\right)$.

We now show the other implication of the proposition using that $\sum_k \lambda_k \frac{\mathbb{E}\left(B_k^2\right)}{\mathbb{E}(B_k)} c_k \geq \sum_k \lambda_k \mathbb{E}\left(B_k^2\right)$. We define a new vector $t=(t_1,\dots,t_R)$ such that $t_i = c_i \frac{\sum_k \lambda_k \mathbb{E}\left(B_k^2\right)}{\sum_k \lambda_k \frac{\mathbb{E}\left(B_k^2\right)}{\mathbb{E}(B_k)} c_k}$, for all i. We observe that t_i is positive for all i and $t_i \leq c_i$ because $\sum_k \lambda_k \frac{\mathbb{E}\left(B_k^2\right)}{\mathbb{E}(B_k)} c_k \geq \sum_k \lambda_k \mathbb{E}\left(B_k^2\right)$.

We now show that the vector $\mathbf{t}$ satisfies the condition of proposition 6

$$\sum_k \lambda_k \frac{\mathbb{E}\left(B_k^2\right)}{\mathbb{E}(B_k)} t_k = \sum_k \lambda_k \frac{\mathbb{E}\left(B_k^2\right)}{\mathbb{E}(B_k)} c_i \frac{\sum_i \lambda_i \mathbb{E}\left(B_i^2\right)}{\sum_i \lambda_i \frac{\mathbb{E}\left(B_i^2\right)}{\mathbb{E}(B_i)} c_i}$$

which equals $\sum_i \lambda_i \mathbb{E}\left(B_i^2\right)$ and this means that the vector $\mathbf{t}$ is achievable. Thus, we have shown that $\mathbf{t}$ is achievable and $t_i \leq c_i$ which means that the problem is feasible in heavy-traffic. ■

C. Nash Equilibrium

Let us first introduce some results that we will be used to prove theorem 1.

We define the vector $\mathbf{g}^m$ as a vector where its i-th coordinate, g_i^m, is $g_i^m = 1$, if $i \geq m$, and $g_i^m > 1$, otherwise.

Definition 3: For all $m = 1,\dots,R$, we define $\mathbf{g}^m = (g_1^m, g_2^m, \dots, g_{m-1}^m, 1, \dots, 1)$, where $g_i^m > 1$, if $i < m$.

We observe that when $m=1$, then $\mathbf{g}^m$ coincides with the all-ones vector.

We show that if a vector $\mathbf{g}^m$ satisfies the m-th time constraint, then all the constraints $m+1,\dots,R$ will be also satisfied.

Lemma 2: Let $\mathbf{g}^m$ be a vector as defined in definition 3. If $\overline{T}_m(\boldsymbol{g}^m;1) \leq \tilde{c_m}$, then, for all $j > m$, $\overline{T}_j(\boldsymbol{g}^m;1) \leq \tilde{c_j}$.

Proof: From the expression $\overline{T}_m(\boldsymbol{g}^m;1) \leq \tilde{c_m}$ we obtain

$$\frac{\sum_k \lambda_k \mathbb{E}\left(B_k^2\right)}{\sum_k \lambda_k \mathbb{E}\left(B_k^2\right)/g_k} \leq \tilde{c_m} g_m^m/\mathbb{E}(B_m) = \tilde{c_m}\ 1/\mathbb{E}(B_m)$$

Since for all $j > m$, we have that $\tilde{c_m}/\mathbb{E}(B_m) \leq \tilde{c_j}/\mathbb{E}(B_j)$ due to the order of classes and that $\tilde{c_i} = c_i(1-\rho)$ and using that $g_j = 1$ for all $j \geq m$, then we can state that

$$\frac{\sum_k \lambda_k \mathbb{E}\left(B_k^2\right)}{\sum_k \lambda_k \mathbb{E}\left(B_k^2\right)/g_k} \leq \tilde{c_m}/\mathbb{E}(B_m) \leq \tilde{c_j}/\mathbb{E}(B_j) \iff$$

$$\overline{T}_j(\boldsymbol{g};1) \leq \tilde{c_j},$$

for all $j > m$. ■

We are now in position to proof the main result of the equilibrium.

Proof: Proof of theorem 1.

To characterize the equilibrium, we need to assume that the feasible region is not empty. It is obvious that if the feasible region is the empty set there is no solution of the game.

First, we give the conditions that must verify the equilibrium. Then, we characterize the weights in the equilibrium. Thirdly, we see that this condition coincides with the condition given in (14). Finally, we see that the equilibrium is unique.

Let m be the minimum value such that $\overline{T}_m(\boldsymbol{g}^m;1) \leq \tilde{c_m}$, where $\boldsymbol{g}^m$ is as defined in definition 3. We prove that the equilibrium is obtained for the vector $\mathbf{g}^m$ verifying that m is the minimum such that $\overline{T}_m(\boldsymbol{g}^m;1) \leq \tilde{c_m}$ and, in addition, it is satisfied that $\overline{T}_i(\boldsymbol{g}^m;1) = \tilde{c_i}$ for all $i < m$.

Using the result of lemma 2, we can say that if $\overline{T}_m(\boldsymbol{g}^m;1) \leq \tilde{c_m}$, then $\overline{T}_k(\boldsymbol{g}^m;1) \leq \tilde{c_k}$, for $k = m+1,\ldots,R$. This means that all the if the m-th time constraint is satisfied then all the following constraints also hold. We show that the vector of weights $\mathbf{g^{NE}} = \epsilon\ \mathbf{g}^m$ that verifies $\overline{T}_i(\boldsymbol{g}^m;1) = \tilde{c_i}$, for all $i < m$ and $\overline{T}_i(\boldsymbol{g}^m;1) \leq \tilde{c_i}$ for all $i \leq m$ is the equilibrium. We know that $\mathbf{g^{NE}}$ is feasible because it satisfies all the time constraints. We now see that $\mathbf{g^{NE}}$ minimizes the weights. In case that one of the first $m-1$ coordinates of $\mathbf{g}^m$ diminishes its weight, its time constraints will not be satisfied since they are equalities. The rest of the coordinates of $\mathbf{g}^m$ are ϵ, so they can not be less and their time constraints are satisfied.

We now characterize the weights of the equilibrium. From the definition of the vector $\mathbf{g^m}$, we know that $g_i^{NE} = \epsilon$, for all $i \geq m$. Using the feasibility condition, we state that there exists $\tilde{t_i} \leq \tilde{c_i}$ for all i such that $\overline{T}_i(\boldsymbol{g}^m;1) = \tilde{t_i}$. From the result given in proposition 6, it follows that this performance is given when the weights satisfy $\frac{g_i^{NE}}{g_j^{NE}} = \frac{\tilde{t_j}/\mathbb{E}(B_j)}{\tilde{t_i}/\mathbb{E}(B_i)}$ for all $i \neq j$. Since $\tilde{t_i} = \tilde{c_i}$ for all $i < m$, we can state that for all $i < m$ we have $\frac{g_i^{NE}}{g_m^{NE}} = \frac{\tilde{t_m}/\mathbb{E}(B_m)}{\tilde{c_i}/\mathbb{E}(B_i)}$, i.e., $g_i^{NE} = \epsilon\ \frac{\tilde{t_m}/\mathbb{E}(B_m)}{\tilde{c_i}/\mathbb{E}(B_i)}$ for all $i < m$.

We prove that the condition $\overline{T}_m(\boldsymbol{g}^m;1) = \tilde{t_m} \leq \tilde{c_m}$ is equivalent to (14), from Proposition 2 and that the weights in the equilibrium are $g_i^{NE} = \epsilon$ for all $i \geq m$ and for all $i < m$, $g_i^{NE} = \epsilon\ \frac{\tilde{t_m}/\mathbb{E}(B_m)}{\tilde{c_i}/\mathbb{E}(B_i)}$. Using Proposition 2 for the vector $\mathbf{g^m}$ and that $\frac{g_i^{NE}}{g_j^{NE}} = \frac{g_i^m}{g_j^m}$, it follows that for the minimum m we have

$$\tilde{c_m} \geq \tilde{t_m} = \mathbb{E}(B_m)\frac{\sum_{k=1}^R \lambda_k \mathbb{E}\left(B_k^2\right)}{\sum_{k=1}^R \lambda_k \mathbb{E}\left(B_k^2\right)\frac{g_m^m}{g_k^m}}$$

Since for all $k < m$, $\frac{g_m^m}{g_k^m} = \frac{\tilde{c_k}/\mathbb{E}(B_k)}{\tilde{t_m}/\mathbb{E}(B_m)}$ and for all $k \geq m$, $\frac{g_m^m}{g_k^m} = 1$, then

$$\tilde{t_m} = \mathbb{E}(B_m)\frac{\sum_{k=1}^R \lambda_k \mathbb{E}\left(B_k^2\right)}{\sum_{k=1}^{m-1} \lambda_k \mathbb{E}\left(B_k^2\right)\frac{\tilde{c_k}/\mathbb{E}(B_k)}{\tilde{t_m}/\mathbb{E}(B_m)} + \sum_{k=m}^R \lambda_k \mathbb{E}\left(B_k^2\right)}$$

This is equivalent to say that

$$\frac{\tilde{t_m}}{\mathbb{E}(B_m)} = \frac{\sum_{k=1}^R \lambda_k \mathbb{E}\left(B_k^2\right)}{\sum_{k=1}^{m-1} \lambda_k \mathbb{E}\left(B_k^2\right)\frac{\tilde{c_k}/\mathbb{E}(B_k)}{\tilde{t_m}/\mathbb{E}(B_m)} + \sum_{k=m}^R \lambda_k \mathbb{E}\left(B_k^2\right)}$$

And rearranging both sides of the equation it yields to

$$\frac{\tilde{t_m}}{\mathbb{E}(B_m)} = \frac{\sum_{k=1}^R \lambda_k \mathbb{E}\left(B_k^2\right) - \sum_{k=1}^{m-1} \lambda_k \frac{\mathbb{E}\left(B_k^2\right)}{\mathbb{E}(B_k)} c_k}{\sum_{k=m}^R \lambda_k \mathbb{E}\left(B_k^2\right)}$$

that coincides with the given formula since $\tilde{t_m} \leq \tilde{c_m}$.

We show the uniqueness of the equilibrium proving that if the equilibrium is $\mathbf{g^m}$, then $\mathbf{g^{m+i}}$ is not the equilibrium, for $i = 1,\ldots,R-m$. To prove this, we consider that there exists a value m satisfying

$$\frac{\tilde{c_m}}{\mathbb{E}(B_m)} \geq \frac{t_m}{\mathbb{E}(B_m)} = \frac{\sum_{k=1}^R \lambda_k \mathbb{E}\left(B_k^2\right) - \sum_{k=1}^{m-1} \lambda_k \frac{\mathbb{E}\left(B_k^2\right)}{\mathbb{E}(B_k)} \tilde{c_k}}{\sum_{k=m}^R \lambda_k \mathbb{E}\left(B_k^2\right)} \tag{20}$$

and we will see that for any $i = 1,\ldots,R-m$, $\mathbf{g^{m+i}}$ that satisfies (14) is not the equilibrium. To prove that, we show that there is no vector $\mathbf{g^{m+i}}$ with weights as defined in theorem 1 that verifies

$$\frac{\tilde{c_{m+i}}}{\mathbb{E}(B_{m+i})} \geq \frac{\overline{t_{m+i}}}{\mathbb{E}(B_{m+i})} = \frac{\sum_{k=1}^R \lambda_k \mathbb{E}\left(B_k^2\right) - \sum_{k=1}^{m+i-1} \lambda_k \frac{\mathbb{E}\left(B_k^2\right)}{\mathbb{E}(B_k)} \tilde{c_k}}{\sum_{k=m+i}^R \lambda_k \mathbb{E}\left(B_k^2\right)} \tag{21}$$

We suppose that there exist a value $i = 1,\ldots,R-m$ such that (21) is verified and we will arrive to a contradiction because we observe that the m-th coordinate of $\mathbf{g^{m+i}}$ is less than ϵ, i.e., $\overline{t_{m+i}}/\mathbb{E}(B_{m+1}) \leq \tilde{c_m}/\mathbb{E}(B_m)$.

Using (20), we obtain the following equality:

$$\frac{t_m}{\mathbb{E}(B_m)}\sum_{k=m}^R \lambda_k \mathbb{E}(B_k^2) = \sum_{k=1}^R \lambda_k \mathbb{E}\left(B_k^2\right) - \sum_{k=1}^{m-1} \lambda_k \frac{\mathbb{E}\left(B_k^2\right)}{\mathbb{E}(B_k)}\tilde{c_k} \tag{22}$$

From (21) we observe that

$$\frac{\tilde{c_{m+i}}}{\mathbb{E}(B_{m+i})} \geq \frac{\overline{t_{m+i}}}{\mathbb{E}(B_{m+i})} = \frac{\sum_{k=1}^R \lambda_k \mathbb{E}\left(B_k^2\right) - \sum_{k=1}^{m+i-1} \lambda_k \frac{\mathbb{E}\left(B_k^2\right)}{\mathbb{E}(B_k)} \tilde{c_k}}{\sum_{k=m+i}^R \lambda_k \mathbb{E}\left(B_k^2\right)}$$

$$= \frac{\sum_{k=1}^R \lambda_k \mathbb{E}\left(B_k^2\right) - \sum_{k=1}^{m-1} \lambda_k \frac{\mathbb{E}\left(B_k^2\right)}{\mathbb{E}(B_k)} \tilde{c_k} - \sum_{k=m}^{m+i-1} \lambda_k \frac{\mathbb{E}\left(B_k^2\right)}{\mathbb{E}(B_k)} \tilde{c_k}}{\sum_{k=m+i}^R \lambda_k \mathbb{E}\left(B_k^2\right)}$$

Taking into account the equality of (22), we obtain

$$\frac{\overline{t_{m+i}}}{\mathbb{E}(B_{m+i})} = \frac{\frac{t_m}{\mathbb{E}(B_m)}\sum_{k=m}^R \lambda_k \mathbb{E}(B_k^2) - \sum_{k=m}^{m+i-1} \lambda_k \frac{\mathbb{E}\left(B_k^2\right)}{\mathbb{E}(B_k)} \tilde{c_k}}{\sum_{k=m+i}^R \lambda_k \mathbb{E}\left(B_k^2\right)}$$

Since we know that $\frac{\tilde{c_m}}{\mathbb{E}(B_m)} \leq \frac{\tilde{c_k}}{\mathbb{E}(B_k)}$ for all $k > m$, then

$$\frac{\overline{t_{m+i}}}{\mathbb{E}(B_{m+i})} \leq \frac{\frac{t_m}{\mathbb{E}(B_m)} \sum_{k=m}^{R} \lambda_k \mathbb{E}(B_k^2) - \frac{\tilde{c_m}}{\mathbb{E}(B_k)} \sum_{k=m}^{m+i-1} \lambda_k \mathbb{E}\left(B_k^2\right)}{\sum_{k=m+i}^{R} \lambda_k \mathbb{E}\left(B_k^2\right)}$$

We now use that $t_m \leq \tilde{c_m}$ since $\mathbf{g^m}$ is the equilibrium to state that

$$\begin{aligned}\frac{\overline{t_{m+i}}}{\mathbb{E}(B_{m+i})} &\leq \frac{\frac{t_m}{\mathbb{E}(B_m)} \sum_{k=m}^{R} \lambda_k \mathbb{E}(B_k^2) - \frac{c_m}{\mathbb{E}(B_k)} \sum_{k=m}^{m+i-1} \lambda_k \mathbb{E}\left(B_k^2\right)}{\sum_{k=m+i}^{R} \lambda_k \mathbb{E}\left(B_k^2\right)} \\ &= \frac{\frac{\tilde{c_m}}{\mathbb{E}(B_m)} \sum_{k=m+i}^{R} \lambda_k \mathbb{E}(B_k^2)}{\sum_{k=m+i}^{R} \lambda_k \mathbb{E}\left(B_k^2\right)} = \frac{c_m}{\mathbb{E}(B_m)}\end{aligned}$$

which means that the component m of the equilibrium $\mathbf{g^{m+i}}$ is less than the $m+i$-th component, which is in contradiction with proposition 3.

■