Optimal policy for multi-class scheduling in a single server queue

Natalia Osipova, Urtzi Ayesta, Konstantin Avrachenkov

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Optimal policy for multi-class scheduling in a single server queue

Natalia Osipova — Urtzi Ayesta — Konstantin Avrachenkov

N° ????

Mars 2009

Thème COM

Rapport de recherche

ISSN 0249-6399 ISRN INRIA/RR--????--FR+ENG

Optimal policy for multi-class scheduling in a single server queue

Natalia Osipova [*], Urtzi Ayesta [†], Konstantin Avrachenkov [‡]

Thème COM — Systèmes communicants
Projets MAESTRO

Rapport de recherche n° ???? — Mars 2009 — 30 pages

Résumé : Nous obtenons la politique optimale pour l'ordonnancement dans une file d'attente multi-classe avec un serveur unique. Nous appliquons les résultats de Gittins [Git89], où il avait trouvé la politique optimale qui minimise le temps moyen de sejour dans le système dans la file d'attente $M/G/1$ avec un serveur unique parmi toutes les politiques non-anticipatoires. Nous montrons que l'extension des résultats de Gittins permet de caractériser la politique d'ordonnancement optimale dans la file d'attente $M/G/1$ multi-classe. Nous appliquons le résultat général dans plusieurs cas, lorsque la distribution de temps de service a un taux de hasard décroissant, comme Pareto et hyper-exponentielle. Nous montrons que dans le cas de plusieurs classes, la politique optimale est la politique prioritaire, dans laquelle les tâches de classes différentes sont classifiés sur plusieurs niveaux de priorité en fonction de leur service obtenu. Nous obtenons pour chaque classe l'expression du temps moyen conditionnel de séjour en utilisant une approche de tâche marquées. Avec ça, nous comparons numériquement le temps moyen de séjour dans le système entre les politiques de Gittins et les politiques populaires comme PS, FCFS et LAS. Comme dans Internet, la distribution de la taille des fichiers est "heavy-tailed" et possède la propriété de DHR, la politique optimale de Gittins peut être appliquée dans les routeurs d'Internet, où les paquets générés par des applications différentes doivent être servis. Typiquement, le routeur n'a pas d'accès au temps exact de séjour requis (en paquets) de la connexion TCP, mais il peut avoir l'accès au service atteint de chaque connexion. Ainsi, nous implémentons l'algorithme optimal de Gittins en NS-2 et nous faisons des simulations numériques pour évaluer le gain de performance possible.

Mots-clés : M/G/1, file d'attente multi-classe, la politique de Gittins, NS-2

[*] INRIA Sophia Antipolis
[†] CNRS LAAS
[‡] INRIA Sophia Antipolis

Unité de recherche INRIA Sophia Antipolis
2004, route des Lucioles, BP 93, 06902 Sophia Antipolis Cedex (France)
Téléphone : +33 4 92 38 77 77 — Télécopie : +33 4 92 38 77 65

Optimal policy for multi-class scheduling in a single server queue

Abstract: In this chapter we apply the Gittins optimality result to characterize the optimal scheduling discipline in a multi-class $M/G/1$ queue. We apply the general result to several cases of practical interest where the service time distributions belong to the set of DHR distributions, like Pareto or hyper-exponential. When there is only one class it is known that in this case the LAS policy is optimal. We show that in the multi-class case the optimal policy is a priority discipline, where jobs of the various classes depending on their attained service are classified into several priority levels. Using a tagged-job approach we obtain, for every class, the mean conditional sojourn time. This allows us to compare numerically the mean sojourn time in the system between the Gittins optimal and popular policies like PS, FCFS and LAS.

Our results may be applicable for instance in an Internet router, where packets generated by different applications must be served or service is non-preemptive. Typically a router does not have access to the exact required service time (in packets) of the TCP connections, but it may have access to the attained service of each connection. Thus we implement the Gittins optimal algorithm in NS-2 and perform experiments to evaluate the achievable performance gain. We find that in the particular example with two classes and Pareto-type service time distribution the Gittins policy outperform LAS by nearly 10% under moderate load.

Key-words: M/G/1, multi-class queue, optimal scheduling, Gittins policy, NS-2 simulator

1 Introduction

We are interested to schedule jobs in the $M/G/1$ queue with the aim to minimize the mean sojourn time in the system as well as the mean number of jobs in the system. In our study we restrict ourselves to the non-anticipating scheduling policies. Let us recall that the policy is non-anticipating if it does not use information about the size of the arriving jobs. In [Git89], Gittins considered an $M/G/1$ queue and proved that the so-called Gittins index rule minimizes the mean delay. At every moment of time the Gittins rule calculates, depending on the attained service times of jobs, which job should be served. Gittins derived this result as a byproduct of his groundbreaking results on the multi-armed bandit problem. The literature on multi-armed bandit related papers that build on Gittins' result is huge (see for example [VWB, Whi88, Web92, Tsi93, DGNM96, FW99, BNM00]). However, the optimality result of the Gittins index in the context of an $M/G/1$ queue has not been fully exploited, and it has not received the attention it deserves.

In the present work we generalize the Gittins index approach to the scheduling in the multi-class $M/G/1$ queue. We emphasize that Gittins' optimality in a multi-class queue holds under much more general conditions than the condition required for the optimality of the well-known $c\mu$-rule. We recall that the $c\mu$-rule is the discipline that gives strict priority in descending order of $c_k\mu_k$, where c_k and μ_k refer to a cost and the inverse of the mean service requirement, respectively, of class k. Indeed it is known (see for example [BVW85, SY92, NT94]) that the $c\mu$-rule minimizes the weighted mean number of customers in the queue in two main settings : (i) generally distributed service requirements among all non-preemptive disciplines and (ii) exponentially distributed service requirements among all preemptive non-anticipating disciplines. In the preemptive case the $c\mu$-rule is only optimal if the service times are exponentially distributed. On the other hand, by applying Gittins' framework to the multi-class queue one can characterize the optimal policy for arbitrary service time distributions. We believe that our results open an interesting avenue for further research. For instance well-known optimality results in a single-class queue like the optimality of the LAS discipline when the service times are of type decreasing hazard rate or the optimality of FCFS when the service time distribution is of type New-Better-than-Used-in-Expectation can all be derived as corollaries of Gittins' result. The optimality of the $c\mu$-rule can also easily be derived from the Gittins' result.

In order to get insights into the structure of the optimal policy in the multi-class case we consider several relevant cases where the service time distributions are Pareto or hyper-exponential. We have used these distributions due to the evidence that the file size distributions in the Internet are well presented by the heavy-tailed distributions such as Pareto distributions with the infinite second moment. Also it was shown that job sizes in the Internet are well modelled with the distributions with the decreasing hazard rate. We refer to [NMM98, CB97, Wil01] for more details on this area, see also Subsection **??**. In particular, we study the optimal multi-class scheduling in the following cases of the service time distributions : two Pareto distributions, several Pareto distributions, one hyper-exponential and one exponential distributions. Using a tagged-job approach and the collective marks method we obtain, for every class, the mean conditional sojourn time. This allows us to compare numerically the mean sojourn time in the system between the Gittins optimal and popular policies like PS, FCFS and LAS. We find that in a particular example with two classes and Pareto-type service time distribution the Gittins policy outperforms LAS by nearly 25% under moderate load.

From an application point of view, our findings could be applied in Internet routers. Imagine that incoming packets are classified based on the application or the source that generated them. Then it is reasonable to expect that the service time distributions of the various classes may differ from each other. A router in the Internet does not typically have access to the exact required service time (in packets) of the TCP connections, but it may have access to the attained service of each connection. Thus we can apply our theoretical findings in order to obtain the optimal (from the connection-level performance point of view) scheduler at the packet level. We implement the Gittins scheduling policy in the NS-2 simulator and perform experiments to evaluate the achievable performance gain.

The structure of the chapter is as follows : In Section 2 we review the Gittins index policy for the single-class $M/G/1$ queue and then provide a general framework of the Gittins index policy for the multi-class $M/G/1$ queue. In Section 3, we study the Gittins index policy for the case of two Pareto distributed classes. In particular, we derive analytic expressions for the mean conditional sojourn times, study various properties of the optimal policy, provide numerical examples and NS-2 simulations. At the end of Section 3 we generalize the results to multiple Pareto classes. In Section 4 we study the case of two distributions : one distribution being exponential and the other distribution being hyper-exponential with two phases. For the case of exponential and hyper-exponential distributions, we also obtain analytical results and provide numerical examples. Section 5 concludes the chapter. Some additional profs are given in the Appendix.

2 Gittins policy in multi-class $M/G/1$ queue

Let us first recall the basic results related to the Gittins index policy in the context of a single-class $M/G/1$ queue.

Let Π denote the set of non-anticipating scheduling policies. Popular disciplines such as PS, FCFS and LAS, also called FB, belong to Π. Important disciplines that do not belong to Π are SRPT and Shortest Processing Time (SPT).

We consider a single-class $M/G/1$ queue. Let X denote the service time with distribution $P(X \leq x) = F(x)$. The density is denoted by $f(x)$, the complementary distribution by $\overline{F}(x) = 1 - F(x)$ and the hazard rate function by $h(x) = f(x)/\overline{F}(x)$. Let $\overline{T}^{\pi}(x)$, $\pi \in \Pi$ denote the mean conditional sojourn time for the job of size x in the system under the scheduling policy π, and $\overline{T}^{\pi}$, $\pi \in \Pi$ denote the mean sojourn time in the system under the scheduling policy π.

Let us give some definitions.

Definition 1. *For any $a, \Delta \geq 0$, let*

$$J(a, \Delta) = \frac{\int_0^{\Delta} f(a+t)dt}{\int_0^{\Delta} \overline{F}(a+t)dt} = \frac{\overline{F}(a) - \overline{F}(a+\Delta)}{\int_0^{\Delta} \overline{F}(a+t)dt}. \tag{1}$$

For a job that has attained service a and is assigned Δ units of service, equation (1) can be interpreted as the ratio between (i) the probability that the job will complete with a quota of Δ (interpreted as payoff) and (ii) the expected processor time that a job with attained service a and service quota Δ will require from the server (interpreted as investment). Note that for every $a > 0$

$$J(a,0) = \frac{f(a)}{\overline{F}(a)} = h(a),$$
$$J(a,\infty) = \frac{\overline{F}(a)}{\int_0^\infty \overline{F}(a+t)\,dt} = 1/E[X-a|X>a].$$

Note further that $J(a,\Delta)$ is continuous with respect to Δ.

Definition 2. *The Gittins index function is defined by*

$$G(a) = \sup_{\Delta \geq 0} J(a,\Delta), \tag{2}$$

for any $a \geq 0$.

We call $G(a)$ the *Gittins index* after the author of book [Git89], which handles various static and dynamic scheduling problems. Independently, Sevcik defined a corresponding index when considering scheduling problems without arrivals in [Sev74]. In addition, this index has been dealt with by Yashkov, see [Yas92] and references therein, in particular the works by Klimov [Kli74, Kli78].

Definition 3. *For any $a \geq 0$, let*

$$\Delta^*(a) = \sup\{\Delta \geq 0 \mid J(a,\Delta) = G(a)\}. \tag{3}$$

By definition, $G(a) = J(a,\Delta^*(a))$ for all a.

Definition 4. *The Gittins index policy π_g is the scheduling discipline that at every instant of time gives service to the job in the system with highest $G(a)$, where a is the job's attained service.*

Theorem 1. *The Gittins index policy minimizes the mean sojourn time in the system between all non-anticipating scheduling policies. In other words, in the $M/G/1$ queue for any $\pi \in \Pi$,*

$$\overline{T}^{\pi_g} \leq \overline{T}^{\pi}.$$

Démonstration. See [Git89]. ☐

Note that by Little's law the Gittins index policy also minimizes the mean number of jobs in the system.

We generalize the result of Theorem 1 to the case of the multi-class single server queue. Let us consider a multi-class $M/G/1$ queue. Let X_i denote the service time with distribution $P(X_i \leq x) = F_i(x)$ for every class $i = 1,\ldots,N$. The density is denoted by $f_i(x)$ and the complementary distribution by $\overline{F}_i(x) = 1 - F_i(x)$. The jobs of every class-i arrive with the Poisson process with rate λ_i, the total arrival rate is $\lambda = \sum_{i=1}^N \lambda_i$. For every class $i = 1,\ldots,N$ we define $J_i(a,\Delta) = \frac{\int_0^\Delta f_i(a+t)dt}{\int_0^\Delta \overline{F}_i(a+t)dt}$ and then the Gittins index of a class-i job is defined as $G_i(a) = \sup_{\Delta\geq 0} J_i(a,\Delta)$.

We define as $\overline{T}_i^{\pi}(x)$ the mean conditional sojourn time for the class-i job of size x, $i = 1,\ldots,N$ and as $\overline{T}^{\pi}$ the mean sojourn time in the system under the scheduling policy $\pi \in \Pi$.

Proposition 1. *In a multi-class $M/G/1$ queue the policy that schedules the job with highest Gittins index $G_i(a)$, $i = 1, \ldots, N$ in the system, where a is the job's attained service, is the optimal policy that minimizes the mean sojourn time.*

Démonstration. The result follows directly from the application of the Definition 2 and Theorem 1 to a multi-class $M/G/1$ queue. □

Let $h_i(x) = f_i(x)/\overline{F}_i(x)$ denote the hazard rate function of class $i = 1, \ldots, N$. Let the service time distribution of class-i have a decreasing hazard rate. It is possible to show, see [AA07], that if $h_i(x)$ is non-increasing, the function $J_i(a, \Delta)$ is non-increasing in Δ. Thus

$$G_i(a) = J_i(a, 0) = h_i(a). \tag{4}$$

As a consequence we obtain the following proposition.

Proposition 2. *In a multi-class $M/G/1$ queue with non-increasing hazard rate functions $h_i(x)$ for every class $i = 1, \ldots, N$, the policy that schedules the job with highest $h_i(a)$, $i = 1, \ldots, N$ in the system, where a is the job's attained service, is the optimal policy that minimizes the mean sojourn time.*

Démonstration. Follows immediately from the Gittins policy Definition 4, Proposition 1 and equation (4). □

The policy presented in Proposition 2 is an optimal policy for the multi-class single-server queue. Let us notice that for the single class single server queue the Gittins policy becomes a LAS policy, as the hazard rate function is the same for all jobs and so the job with the maximal value of the hazard rate function from attained service is the job with the least attained service. When we serve jobs with the Gittins policy in the multi-class queue to find a job which has to be served next we need to calculate the hazard rate of every job in the system. The job which has the maximal value of the hazard rate function is served the next. Later by the value of the hazard rate we mean the value of the hazard rate function of the job's attained service.

Now let us consider several subcases of the described general approach. Depending on the behavior of the hazard rate functions of the job classes the policy is different. We consider the case with two job classes in the system and two subcases : (a) both job classes are distributed with Pareto and the hazard rate functions do not cross and (b) job size distributions are hyper-exponential with one and two phases and they cross at one point. Then we extend the case of two Pareto job classes to the case of N Pareto job classes. We provide the analytical expressions for the mean conditional sojourn times in the system and numerical results. We implemented the algorithm for the case of two Pareto classes with the NS-2 simulator on the packet level.

3 Two Pareto classes

Let us first present the case when job sizes are distributed according to Pareto distribution.

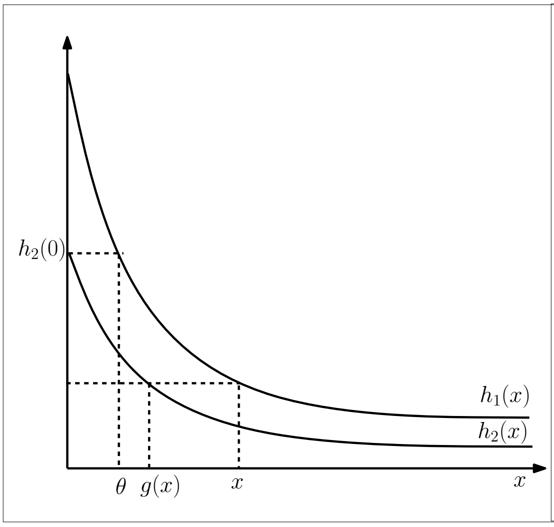

FIG. 1 – Two Pareto classes, hazard rates

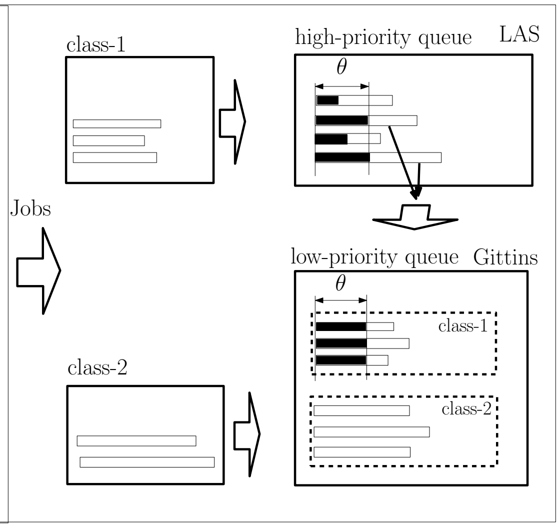

FIG. 2 – Two Pareto classes, policy scheme

3.1 Model description

We consider the case when the job size distribution functions are Pareto. We consider the two-class single server $M/G/1$ queue. Jobs of each class arrive to the server with Poisson process with rates λ_1 and λ_2. The job sizes are distributed according to the Pareto distributions, namely

$$F_i(x) = 1 - \frac{b_i^{c_i}}{(x+b_i)^{c_i}}, \quad i = 1, 2. \tag{5}$$

Here $b_i = m_i(c_i - 1)$, where m_i is the mean of class-i, $i = 1, 2$. Then $f_i(x) = b_i^{c_i} c_i/(x + b_i)^{c_i+1}$, $i = 1, 2$ and the hazard rate functions are

$$h_i(x) = \frac{c_i}{(x+b_i)}, \quad i = 1, 2.$$

These functions cross at the point

$$a^{**} \quad = \quad \frac{c_2 b_1 - c_1 b_2}{c_1 - c_2}.$$

Without loss of generality suppose that $c_1 > c_2$. Then the behavior of the hazard rate functions depends on the values of b_1 and b_2.

Let us first consider the case when the hazard rate function do not cross, so $a^{**} < 0$. This happens when $b_1/b_2 < c_1/c_2$. Then the hazard-rate functions are decreasing and never cross and $h_1(x) > h_2(x)$, for all $x \geq 0$.

Let us denote θ and function $g(x)$ in the following way that

$$h_1(x) = h_2(g(x)), \quad h_1(\theta) = h_2(0).$$

We can see that $g(\theta) = 0$. For given expressions of $h_i(x)$, $i = 1, 2$ we get

$$g(x) = \frac{c_2}{c_1}(x + b_1) - b_2, \quad \theta = \frac{c_1 b_2 - c_2 b_1}{c_2}.$$

According to the definition of function $g(x)$, the class-1 job of size x and the class-2 job of size $g(x)$ have the same value of the hazard rate when they are fully served, see Figure 1. Then the optimal policy structure is given on Figure 2.

3.2 Optimal policy

Jobs in the system are served in two queues, low and high priority queues. The class-1 jobs which have attained service $a < \theta$ are served in the high priority queue with LAS policy. When the class-1 job achieves θ amount of service it is moved to the second low priority queue. The class-2 jobs are moved immediately to the low priority queue. The low priority queue is served only when the high priority queue is empty. In the low priority queue jobs are served in the following way : the service is given to the job with the highest $h_i(a)$, where a is the job's attained service. So, for every class-1 job with a attained service the function $h_1(a)$ is calculated, for every class-2 job with a attained service the function $h_2(a)$ is calculated. After all values of $h_i(a)$ are compared, the job which has the highest $h_i(a)$ is served.

Now let us calculate the expressions of the mean conditional sojourn time for the class-1 and class-2 jobs.

3.3 Mean conditional sojourn times

Let us denote by indices $[]^{(1)}$ and $[]^{(2)}$ the values for class-1 and class-2 accordingly.

Let us define as $\overline{X^n_y}^{(i)}$ the n-th moment and $\rho^{(i)}_y$ be the utilization factor for the distribution $F_i(x)$ truncated at y for $i = 1, 2$. The distribution truncated at y equals $F(x)$ for $x \le y$ and equals 1 when $x > y$. Let us denote $W_{x,y}$ the mean workload in the system which consists only of class-1 jobs with service times truncated at x and of class-2 jobs with service times truncated at y. According to the Pollaczek-Khinchin formula

$$W_{x,y} = \frac{\lambda_1 \overline{X^2_x}^{(1)} + \lambda_2 \overline{X^2_y}^{(2)}}{2(1 - \rho^{(1)}_x - \rho^{(2)}_y)}.$$

Now let us formulate the following Theorem which we prove in the Appendix.

Theorem 2. *In the two-class $M/G/1$ queue where the job size distributions are Pareto, given by (5), and which is scheduled with the Gittins policy described in Subsection 3.2, the mean conditional* *sojourn times for class-*1 *and class-*2 *jobs are*

$$T_1(x) = \frac{x + W_{x,0}}{1 - \rho_x^{(1)}}, \quad x \leq \theta, \tag{6}$$

$$T_1(x) = \frac{x + W_{x,g(x)}}{1 - \rho_x^{(1)} - \rho_{g(x)}^{(2)}}, \quad x > \theta, \tag{7}$$

$$T_2(g(x)) = \frac{g(x) + W_{x,g(x)}}{1 - \rho_x^{(1)} - \rho_{g(x)}^{(2)}}, \quad x > \theta. \tag{8}$$

Démonstration. The proof is very technical and is given in the Appendix. Let us give a very general idea of the proof. To obtain expressions (7), (8) we use the fact that the second low priority queue is the queue with batch arrivals. To obtain expressions of the mean batch size with and without the tagged job we apply the Generating function analysis using the method of the collective marks. □

The obtained expressions (6), (7) and (8) can be interpreted using the tagged-job and mean value approach.

Let us consider class-1 jobs. The job of size $x \leq \theta$ is served in the high priority queue with the LAS policy, so for it the mean conditional sojourn time is known, [Kle76, Sec. 4.6], $T_1(x) = \frac{x+W_{x,0}}{1-\rho_x^{(1)}}$, $x \leq \theta$, where $W_{x,0}$ is the mean workload and $\rho_x^{(1)}$ is the mean load in the system for class-1 jobs with the service time distribution truncated at x. The mean workload $W_{x,0}$ and mean load $\rho_x^{(1)}$ consider only jobs of the high priority queue of class-1.

For jobs of size $x > \theta$ the expression (7) can be presented in the following way, $T_1(x) = x + W_{x,g(x)} + T_1(x)(\rho_x^{(1)} + \rho_{g(x)}^{(2)})$, where

- x is time which is actually spent to serve the job ;
- $W_{x,g(x)}$ is the mean workload which the tagged job finds in the system and which has to be processed before it ;
- $T_1(x)(\rho_x^{(1)} + \rho_{g(x)}^{(2)})$ is the mean time to serve jobs which arrive to the system during the sojourn time of the tagged job and which have to be served before it.

Let us provide more explanations. Let us find the expression for the mean workload in the system for the class-1 job of size x, which is the tagged job. According to the PASTA property of Poisson arrivals, all jobs arriving to the system see the system in the same steady state. So, class-1 and class-2 jobs see the same mean workload in the system when they arrive. As we need to take into account only the mean workload which is served before the tagged job, then for each job the mean workload $W_{x,g(x)}$ depends on the size of the tagged job, x. Jobs which have to be served before the tagged job of class-1 of size x are class-1 jobs of size less than x and class-2 jobs of size less than $g(x)$. Then using Pollaczek-Khinchin formula (6) for class-1 jobs of size less than x and class-2 jobs less than $g(x)$ we conclude that $W_{x,g(x)}$ gives the mean workload in the system for the class-1 job of size x, which has to be served before it. Let us notice that the mean workload in the system for the class-2 job of size $g(x)$ is the same, $W_{x,g(x)}$.

Now let us find the mean workload which arrives during the sojourn time of the tagged job. The sojourn time of the tagged job is $T_1(x)$. The mean load of jobs arriving to the system is :

for the class-1 of size less than x is $\lambda_1 \overline{X_x^1}^{(1)} = \rho_x^{(1)}$ and for the class-2 with size less than $g(x)$ is $\lambda_2 \overline{X_{g(x)}^1}^{(2)} = \rho_{g(x)}^{(2)}$. Then $T_1(x)(\rho_x^{(1)} + \rho_{g(x)}^{(2)})$ is the mean workload which arrives during the sojourn time of the tagged job of class-1 of size x.

Now we use the similar analysis to give an interpretation to the expression of $T_2(g(x))$ for the class-2 job of size $g(x)$. We can rewrite expression (8) in the following way $T_2(g(x)) = g(x) + W_{x,g(x)} + T_2(g(x))(\rho_x^{(1)} + \rho_{g(x)}^{(2)})$.

In the case of the tagged job of class-2 of size $g(x)$ jobs which have to be served before the tagged job are jobs of class-1 of size less than x and jobs of class-2 of size less than $g(x)$. Then in the previous expression $g(x)$ is the time to serve the class-2 job of size $g(x)$; $W_{x,g(x)}$ is the mean workload in the system for the class-2 job of size $g(x)$ which has to be served before it; $T_2(g(x))(\rho_x^{(1)} + \rho_{g(x)}^{(2)})$ is the mean work which arrives during the sojourn time $T_2(x)$ and which has to be served before class-2 job of size $g(x)$.

3.4 Properties of the optimal policy

Property 1. *When class-2 jobs arrive to the server they are not served immediately, but wait until the high priority queue is empty. The mean waiting time is the limit $\lim_{g(x)\to 0} T_2(g(x))$. As $\lim_{x\to\theta} g(x) = 0$, then*

$$\lim_{g(x)\to 0} T_2(g(x)) = \frac{W_{\theta,0}}{1-\rho_\theta^{(1)}} = \frac{\lambda_1 \overline{X_\theta^2}^{(1)}}{2(1-\rho_\theta^{(1)})^2}.$$

Let us notice that

$$\lim_{g(x)\to 0} T_2(g(x)) \neq T_1(\theta) = \frac{\theta + W_{\theta,0}}{1-\rho_\theta^{(1)}}.$$

Class-2 jobs wait in the system to be served in the low priority queue, the mean waiting time is $\lim_{g(x)\to 0} T_2(g(x))$. Class-$1$ jobs of size more then θ also wait in the system to be served in the low priority queue, the mean waiting time for them is $T_1(\theta)$. Property 1 shows that these two mean waiting times are not equal, so class-1 jobs and class-2 jobs wait different times to start to be served in the low priority queue.

Property 2. *Let us consider the condition of no new arrival. According to the optimal policy structure in the low priority queue jobs are served according to the LAS policy with different rates, which depend on the number of jobs in each class and hazard rate functions. For the case when there are no new arrivals in the low priority queue we can calculate the rates with which class-1 jobs and class-2 jobs are served in the system at every moment of time. We consider that all class-1 jobs and all class-2 jobs already received the same amount of service. Let n_1 and n_2 be the number of jobs in class-1 and class-2 and let x_1 and x_2 be the attained services of every job in these classes. Then at any moment*

$$h_1(x_1) = h_2(x_2).$$

If the total capacity of the server is Δ, then let Δ_1 and Δ_2 be the capacities which each job of class-1 and class-2 receives. Then

$$n_1\Delta_1 + n_2\Delta_2 = \Delta. \tag{9}$$

Also

$$h_1(x_1 + \Delta_1) = h_2(x_2 + \Delta_2).$$

As Δ is very small (and so as well Δ_1 and Δ_2) according to the LAS policy, then we can approximate

$$h_i(x + \Delta_i) = h_i(x) + \Delta_i h_i'(x), \quad i = 1, 2.$$

Then from the previous equations we have

$$\Delta_1 h_1'(x_1) = \Delta_2 h_2'(x_2).$$

Then using (9) we get

$$\frac{\Delta_1}{\Delta} = \frac{h_2'(x_2)}{n_1 h_2'(x_2) + n_2 h_1'(x_1)},$$
$$\frac{\Delta_2}{\Delta} = \frac{h_1'(x_1)}{n_1 h_2'(x_2) + n_2 h_1'(x_1)}.$$

This result is true for any two distributions for which the hazard rates are decreasing and never cross. For the case of two Pareto distributions given by (5) we have the following :

$$\frac{\Delta_1}{\Delta} = \frac{c_1}{n_1 c_1 + n_2 c_2}, \quad \frac{\Delta_2}{\Delta} = \frac{c_2}{n_1 c_1 + n_2 c_2}.$$

So, for the case of two Pareto distributions the service rates of class-1 and class-2 jobs do not depend on the current jobs' attained services.

Property 3. *As one can see from the optimal policy description, class-1 and class-2 jobs leave the system together if they have the same values of the hazard rate functions of their sizes and if they find each other in the system. According to the definition of the $g(x)$ function we can conclude that the class-1 job of size x and class-2 job of size $g(x)$, if they find each other in the system, leave the system together. But these jobs do not have the same conditional mean sojourn time,*

$$T_1(x) \neq T_2(g(x)).$$

This follows from expressions (7) and (8).

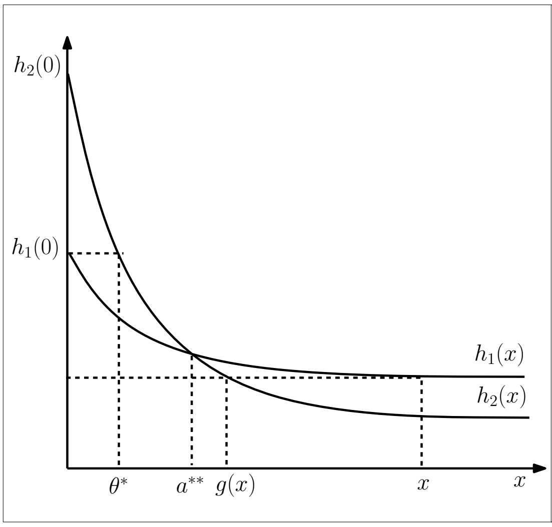

FIG. 3 – Two Pareto extension classes, hazard rates

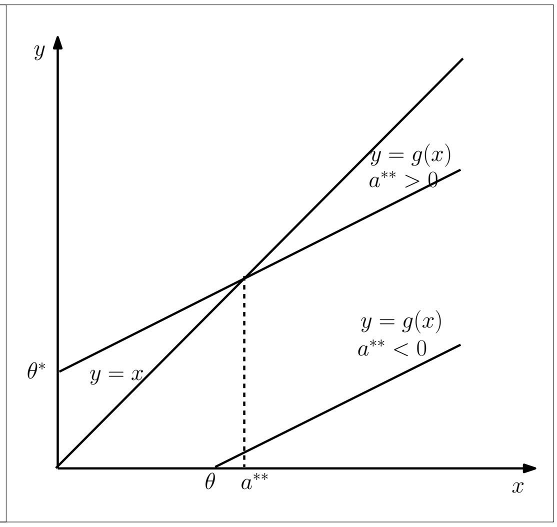

FIG. 4 – Two Pareto extension classes, $g(x)$ function behavior

3.5 Two Pareto classes with intersecting hazard rate functions

Now let us consider the case when the hazard rate function cross, then $a^{**} = (c_2 b_1 - c_1 b_2)/(c_1 - c_2) \geq 0$, see Figure 3. As we considered $c_1 > c_2$, then $h_1(0) < h_2(0)$ and then class-2 jobs are served in the high priority queue until they receive $\theta^* = (c_2 b_1 - c_1 b_2)/c_1$ amount of service. Here θ^* is such that $h_2(\theta^*) = h_1(0)$ and $g(\theta^*) = 0$. In this case the $g(x)$ function crosses the $y = x$ function at point a^{**}, see Figure 4, and so in the low priority queue class-2 jobs are served with higher priority with comparison to class-1 jobs until they receive a^{**} amount of service. After class-1 and class-2 jobs received a^{**} amount of service the priority changes and class-1 jobs receive more capacity of the server in the system. According to this analysis we can rewrite the expressions of mean conditional sojourn times of Section 3, Theorem 2 in the following way

Corollary 1. *In the two-class $M/G/1$ queue where the job size distributions are Pareto, given by (5) such that the hazard rate functions cross, and which is scheduled with the Gittins optimal policy, the mean conditional sojourn times for class-1 and class-2 jobs are*

$$T_1(x) = \frac{x + W_{x,g(x)}}{1 - \rho_x^{(1)} - \rho_{g(x)}^{(2)}}, \quad x \geq 0,$$

$$T_2(x) = \frac{x + W_{0,x}}{1 - \rho_x^{(2)}}, \quad x \leq \theta^*,$$

$$T_2(g(x)) = \frac{g(x) + W_{x,g(x)}}{1 - \rho_x^{(1)} - \rho_{g(x)}^{(2)}}, \quad x > \theta^*.$$

Démonstration. The proof follows from the previous discussion. □

3.6 Numerical results

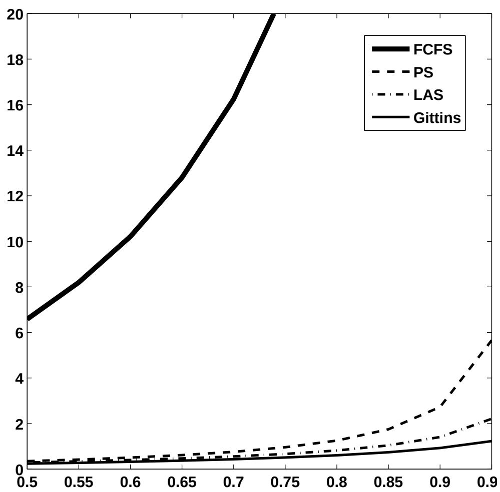

FIG. 5 – Two Pareto classes, mean sojourn times with respect to the load ρ, V_1

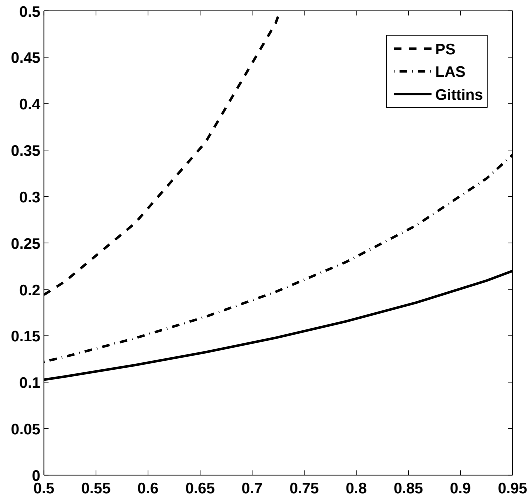

FIG. 6 – Two Pareto classes, mean sojourn times with respect to the load ρ, V_2

We consider two classes with parameters presented in Table 1 and we calculate the mean sojourn time in the system numerically, using the expressions of the mean conditional sojourn time (7), (8) and (6). We provide the results for two different parameters sets, which we call V_1 and V_2.

TAB. 1 – Two Pareto classes, parameters

V	c_1	c_2	m_1	m_2	ρ_1	ρ_2	ρ
V_1	25.0	2.12	0.04	0.89	0.1	0.4..0.85	0.5..0.95
V_2	10.0	1.25	0.05	1.35	0.25	0.25..0.74	0.5..0.99

It is known that in the Internet most of the traffic is generated by the large files (80%), while most of the files are very small (90%). This phenomenon is referred to as “mice-elephant” effect. Also it is known that the file sizes are well presented by the heavy-tailed distributions like Pareto. Here class-1 jobs represent "mice" class and class-2 jobs "elephants". We consider that the load of the small files is fixed and find the mean sojourn time in the system according to the different values of the "elephant" class arrival rate.

We compare the mean sojourn time for the Gittins policy, PS, FCFS and LAS policies. These policies can be applied either in the Internet routers or in the Web service. The expected sojourn times for these policies are, see [Kle76],

$$\overline{T}^{PS} = \frac{\rho/\lambda}{1-\rho},$$
$$\overline{T}^{FCFS} = \rho/\lambda + W_{\infty,\infty},$$

here $W_{\infty,\infty}$ means the total mean unfinished work in the system.

$$\overline{T}^{LAS} = \frac{1}{\lambda_1+\lambda_2}\int_0^{\infty} \overline{T}^{LAS}(x)(\lambda_1 f_1(x) + \lambda_2 f_2(x))dx,$$

where

$$\overline{T}^{LAS}(x) = \frac{x + W_{x,x}}{1-\rho_x^{(1)} - \rho_x^{(2)}}.$$

The mean sojourn times for the parameters sets V_1 and V_2 are presented in Figures 5,6. For the results of V_2 we do not plot the mean sojourn time for the FCFS policy as class-2 has an infinite second moment. The relative gains in mean sojourn time between the Gittins and LAS and Gittins and PS policies are the following. For the set of parameters V_1 :$\max \frac{\overline{T}^{FCFS}-\overline{T}^{Gitt}}{\overline{T}^{FCFS}} = 0.99$, $\max \frac{\overline{T}^{PS}-\overline{T}^{Gitt}}{\overline{T}^{PS}} = 0.78$ and $\max \frac{\overline{T}^{LAS}-\overline{T}^{Gitt}}{\overline{T}^{LAS}} = 0.45$. For the set of parameters V_2 : $\max \frac{\overline{T}^{PS}-\overline{T}^{Gitt}}{\overline{T}^{PS}} = 0.98$ and $\max \frac{\overline{T}^{LAS}-\overline{T}^{Gitt}}{\overline{T}^{LAS}} = 0.39$. The maximal gain is achieved when the system is loaded by around 90%. We note that the PS policy produces much worse results than LAS and Gittins policies.

3.7 Simulation results

We implement Gittins policy algorithm for the case of two Pareto distributed classes in NS-2 simulator. The algorithm is implemented in the router queue. In the router we keep the trace of the attained service (number of the transmitted packets) for every connection in the system. We use timer to detect the moment when there are no more packets from a connection in the queue. Then we stop to keep the trace of the attained service for this connection.

It is possible to select the packet with the minimal sequence number of the connections which has to be served instead of selecting the first packet in the queue. In the current simulation this parameter does not play a big role according to the selected model scheme and parameters. (There are no drops in the system, so there are no retransmitted packets. Then all the packets arrive in the same order as they were sent.)

The algorithm which is used for the simulations is as follows :

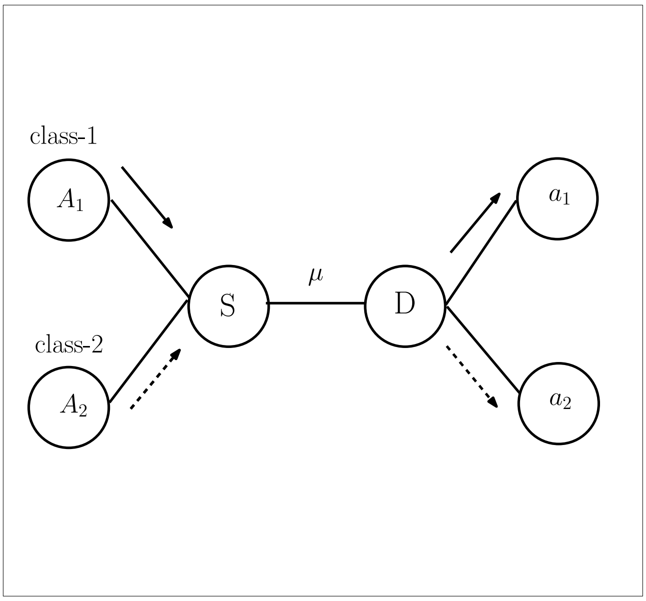

FIG. 7 – NS-2 simulation scheme.

Algorithm
on packet dequeue
select the connection f with the max $h_i(a_f)$, where
    a_f is the flow's attained service
select the first packet p_f of the connection f in the queue
dequeue selected packet p_f
set $a_f = a_f + 1$

To compare Gittins policy with the LAS policy we also implemented LAS algorithm in the router queue. According to the LAS discipline the packet to dequeue is the packet from the connection with the least attained service.

The simulation topology scheme is given in Figure 7. Jobs arrive to the bottleneck router in two classes, which represent mice and elephants in the network. Jobs are generated by FTP sources which are connected to TCP senders. File size distributions are Pareto, $F_i = 1 - b_i^{c_i}/(x + b_i)^{c_i}$, $i = 1, 2$. Jobs arrive according to Poisson arrivals with rates λ_1 and λ_2.

We consider that all connections have the same propagation delays. The bottleneck link capacity is $\mu = 100\,\mathrm{Mbit/s}$. All the connections have a Maximum Segment Size (MSS) of $540\,\mathrm{B}$. The simulation run time is 2000 seconds. We provide two different versions of parameters selection, which we call Vs_1 and Vs_2. In Vs_1 first class takes 25% of the total bottleneck capacity and in Vs_2

it takes 50%. Both scenarios correspond to the case when the hazard rate function do not cross, see Subsection 3.

The parameters we used are given in Table 2.

TAB. 2 – Two Pareto classes, simulation parameters

Ver.	c_1	c_2	m_1	m_2	ρ_1	ρ_2	ρ
Vs_1	10.0	1.25	0.5	6.8	0.25	0.50	0.75
Vs_2	10.0	2.25	0.5	4.5	0.50	0.37	0.87

The results are given in Table 3. We provide results for the NS-2 simulations and the values of the mean sojourn times provided by the analytical model with the same parameters. We calculate the related gain of the Gittins policy in comparison with DropTail and LAS policies, $g_1 = \frac{\overline{T}^{DT} - \overline{T}^{Gitt}}{\overline{T}^{DT}}$ and $g_2 = \frac{\overline{T}^{LAS} - \overline{T}^{Gitt}}{\overline{T}^{LAS}}$

TAB. 3 – Mean sojourn times

Ver.	$\overline{T}^{DT}$	$\overline{T}^{LAS}$	$\overline{T}^{Gitt}$	g_1	g_2
Vs_1 NS-2	18.72	2.10	2.08	88.89%	0.95%
Vs_1 theory	PS : 4.71	1.58	1.01	78.56%	36.08%
Vs_2 NS-2	6.23	2.03	1.83	70.63%	9.85%
Vs_2 theory	PS : 6.46	3.25	2.19	66.10%	32.62%

We found that with the NS-2 simulations the gain of the Gittins policy in comparison with the LAS policy is not so significant when the small jobs do not take a big part of the system load. As one can see in Vs_2 when the class-1 load is 50% the related gain of the Gittins policy in comparison with LAS policy is 10%. In both versions the relative gain for the corresponding analytical system is much higher and reaches up to 36%. We explain this results with the phenomena related to the TCP working scheme. Also we explain the low gain in Vs_1 by the fact that the load in the system is not high.

3.8 Multiple Pareto classes

We consider a multi-class single server $M/G/1$ queue. Jobs arrive to the system in N classes. Jobs of i-th class, $i = 1, \ldots, N$ arrive according to Poisson arrival processes with rates λ_i. Jobs size distributions are Pareto, namely

$$F_i(x) = 1 - \frac{1}{(x+1)^{c_i}}, \quad i = 1, \ldots, N.$$

Then, the hazard rates

$$h_i(x) = \frac{c_i}{(x+1)}, \quad i = 1, \ldots, N,$$

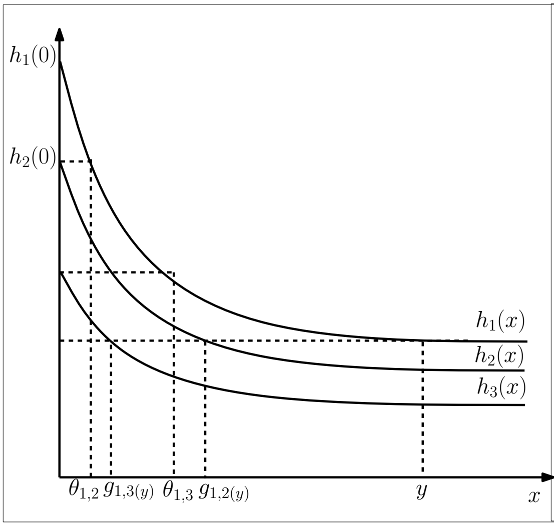

FIG. 8 – N Pareto classes, hazard rates

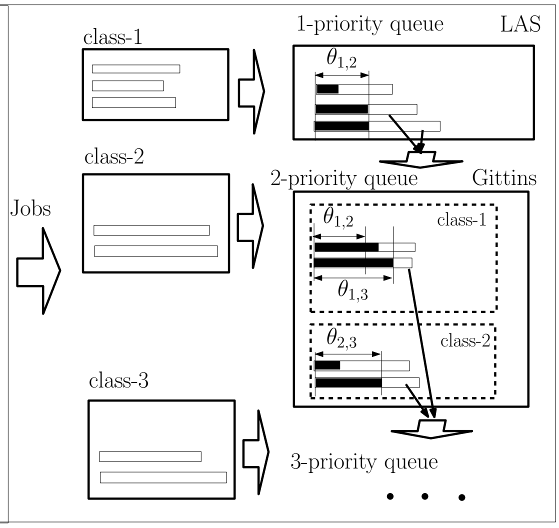

FIG. 9 – N Pareto classes, policy scheme

never cross. Without loss of generality, let us consider that $c_1 > c_2 > \ldots > c_N$. Let us define the values of $\theta_{i,j}$ and $g_{i,j}(x)$, $i,j = 1,\ldots,N$ in the following way

$$h_i(\theta_{i,j}) = h_j(0),$$
$$h_i(x) = h_j(g_{i,j}(x)).$$

Then we get

$$g_{i,j}(x) = \frac{c_j}{c_i}(x+1) - 1, \quad \theta_{i,j} = \frac{c_i}{c_j} - 1.$$

Let us notice that $\theta_{k,i} < \theta_{k,i+1}$ and $\theta_{i,k} > \theta_{i+1,k}$, $k = 1,\ldots,N$, $i = 1,\ldots,N-1$, $i \neq k$, $i \neq k+1$, see Figure 8. Let us denote that $\theta_{i,i} = 0$ for $i = 1,\ldots,N$.

The scheme of the optimal policy is given on Figure 9.

Optimal policy.

There are N queues in the system. Class-1 jobs arrive to the system and go to the first-priority queue-1. There they are served with the LAS policy until they get $\theta_{1,2}$ of service. Then they are moved to the queue-2, which is served only when the queue-1 is empty. In the queue-2 jobs of class-1 are served together with jobs of class-2. Every moment the service is given to the job with the highest $h_i(a)$, $i = 1, 2$, where a is a jobs attained service. When jobs of class-1 attain service $\theta_{1,3}$ they are moved to the queue-3. When jobs of class-2 attain service $\theta_{2,3}$ they are also moved to the queue-3. In queue-3 the jobs of class-1, class-2 and class-3 are served together. Every moment of time the service is given to the job with the highest $h_i(a)$, $i = 1, 2, 3$, where a is a jobs attained service. And so on.

To find the expressions for the mean conditional sojourn times in the system we use the analysis which we used in interpretation of the mean conditional sojourn times expressions in the case of two class system, see Section 3. The mean conditional sojourn time for the tagged job of class-k consists of the time to serve the tagged job when the system is empty, the mean workload in the system which has to be served before the tagged job and the mean workload which arrives during the sojourn time of the tagged job and has to be served before it.

Let the tagged job be from class-1 of size x. Jobs which have the same priority in the system and which have to be served before the tagged job are : class-1 jobs of size less than x, class-i jobs of size less than $g_{1,i}(x)$.

We denote $\overline{X_y^n}^{(i)}$ the n-th moment and $\rho_y^{(i)}$ the utilization factor for the distribution $F_i(x)$ of the class-i, $i = 1, \dots, N$ truncated at y. The mean workload in the system which has to be served before the tagged job is then found with Pollaczek-Khinchin formula and equals to

$$W_{x,g_{1,2}(x),\dots,g_{1,N}(x)} = \frac{\sum_{i=1}^N \lambda_i \overline{X^2_{g_{1,i}(x)}}}{2(1 - \sum_{i=1}^N \rho_{g_{1,i}(x)})}.$$

Then we formulate the theorem.

Theorem 3. *For class-*1 *jobs of size* x *such as* $\theta_{1,p} < x < \theta_{1,p+1}$, $p = 1, \dots, N$ *and corresponding class-*k *jobs with sizes* $g_{1,k}(x)$, $k = 2, \dots, p$ *the mean conditional sojourn times are given by*

$$T_1(x) = \frac{x + W(x, g_{1,2}(x), \dots, g_{1,p}(x))}{1 - \rho_1(x) - \rho_2(g_{1,2}(x)) - \dots - \rho_p(g_{1,p}(x))},$$
$$T_k(g_{1,k}(x)) = \frac{g_{1,k}(x) + W(x, g_{1,2}(x), \dots, g_{1,p}(x))}{1 - \rho_1(x) - \rho_2(g_{1,2}(x)) - \dots - \rho_p(g_{1,p}(x))}.$$

Here we consider that $\theta_{i,N+1} = \infty$, $i = 1, \dots, N$.

Démonstration. Similar to the proof of Theorem 2. □

4 Hyper-exponential and exponential classes

We consider a two class $M/G/1$ queue. Jobs of each class arrive with Poisson arrival process with rates λ_1 and λ_2. The job size distribution of class-1 is exponential with mean $1/\mu_1$, and hyper-exponential with two phases for class-2 with the mean $(\mu_3 p + (1-p)\mu_2)/(\mu_2\mu_3)$. Namely,

$$F_1(x) = 1 - e^{-\mu_1 x}, \ F_2(x) = 1 - pe^{-\mu_2 x} - (1-p)e^{-\mu_3 x}. \tag{10}$$

Note that the hazard rates are

$$h_1(x) = \mu_1, \ h_2(x) = \frac{p\mu_2 e^{-\mu_2 x} + (1-p)\mu_3 e^{-\mu_3 x}}{pe^{-\mu_2 x} + (1-p)e^{-\mu_3 x}}, \ x \geq 0.$$

The hazard rate function of class-1 is a constant and equals to $h_1 = \mu_1$. The hazard rate function $h_2(x)$ of the class-2 is decreasing in x. As both hazard rate functions are non-increasing the optimal

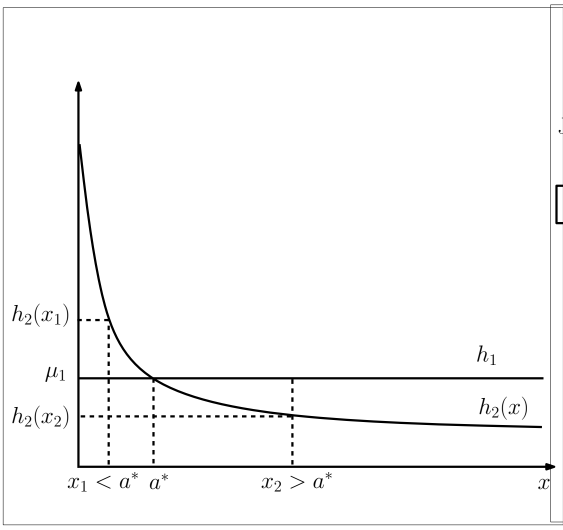

FIG. 10 – Exponential and HE classes, hazard rates.

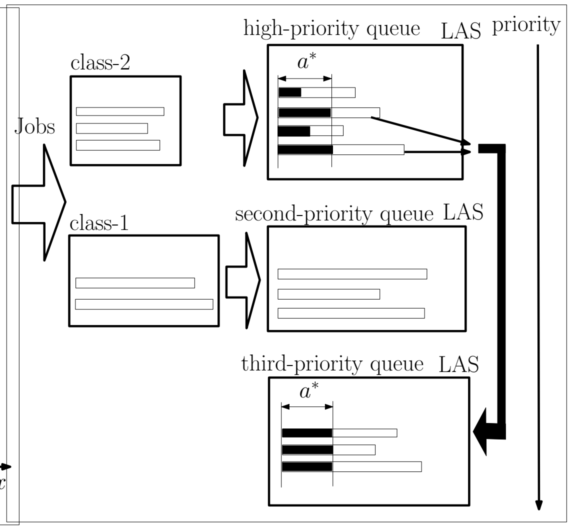

FIG. 11 – Exponential and HE classes, policy description.

policy which minimizes the mean sojourn time is Gittins policy based on the value of the hazard function, which gives service to jobs with the maximal hazard rate.

For the selected job size distributions the hazard rate functions behave in different ways depending on parameters μ_1, μ_2, μ_3 and p. The possible behaviors of the hazard rate functions determine the optimal policy in the system. If the hazard rate functions never cross, the hazard rate of class-1 is higher than the hazard rate of class-2, then class-1 jobs are served with priority to class-2 jobs. This happens when $h_1 > h_2(x)$, $x > 0$. As $h_2(x)$ is decreasing, then this happens when $\mu_1 > h_2(0)$. Let us consider that $\mu_2 > \mu_3$, then as $h_2(0) = p\mu_2 + (1-p)\mu_3$ and $\mu_1 > h_2(0)$ if $\mu_1 > \mu_2 > \mu_3$. For this case it is known that the optimal policy is a strict priority policy, which serves class-1 jobs with the strict priority with respect to class-2 jobs. From our discussion it follows that this policy is optimal even if $\mu_2 > \mu_1 > \mu_3$, but still $\mu_1 > p\mu_2 + (1-p)\mu_3$.

Let us consider the case when $\mu_2 > \mu_1 > \mu_3$ and $\mu_1 < p\mu_2 + (1-p)\mu_3$. Then it exists the unique point of intersection of $h_2(x)$ and h_1. Let us denote by a^* the point of this intersection. The value of a^* is the solution of $\frac{p\mu_2 e^{-\mu_2 x} + (1-p)\mu_3 e^{-\mu_3 x}}{p e^{-\mu_2 x} + (1-p) e^{-\mu_3 x}} = \mu_1$. Solving this equation, we get that

$$a^* = \frac{1}{\mu_2 - \mu_3} \ln\left(\frac{p}{1-p}\frac{\mu_2 - \mu_1}{\mu_1 - \mu_3}\right).$$

The hazard rate function scheme is given on Figure 10. Then, the optimal policy is the following.

4.1 Optimal policy.

There are three queues in the system, which are served with the strict priority between them. The second priority queue is served only when the first priority queue is empty and the third priority queue is served only when the first and second priority queues are empty. Class-2 jobs that arrive to the system are served in the first priority queue with the LAS policy until they get a^* amount of service. After they get a^* amount of service they are moved to the third priority queue, where they are served according to the LAS policy. Class-1 jobs arrive to the system and go to the second priority queue, where they are served with LAS policy. Since $h_1(x) = \mu_1$, class-1 jobs can be served with any non-anticipating scheduling policy. The scheme of the optimal policy is given on Figure 11.

According to this optimal policy we find the expressions of the expected sojourn times for the class-1 and class-2 jobs.

4.2 Expected sojourn times

Let us recall that the mean workload in the system for class-1 jobs of size less than x and class-2 jobs of size less than y is $W_{x,y}$ and is given by (6). We prove the following Theorem.

Theorem 4. *The mean conditional sojourn times in the $M/G/1$ queue with job size distribution given by (10) under Gittins optimal policy described in Subsection 4.1 are given by*

$$T_1(x) = \frac{x + W_{x,a^*}}{1 - \rho_x^{(1)} - \rho_{a^*}^{(2)}}, \quad x \geq 0, \tag{11}$$

$$T_2(x) = \frac{x + W_{0,x}}{1 - \rho_x^{(2)}}, \quad x \leq a^*, \tag{12}$$

$$T_2(x) = \frac{x + W_{\infty,x}}{1 - \rho_\infty^{(1)} - \rho_x^{(2)}}, \quad x > a^*. \tag{13}$$

Démonstration. To find expressions of the mean conditional sojourn times we use the mean-value analysis and tagged job approach. The mean conditional sojourn time for the class-1 job of size x consists of the following elements.

- x, time needed to serve the job itself.
- mean workload in the system which has to be served before the tagged job.
- mean time to serve jobs which arrive to the system during the sojourn time of the current job and which have to be served before the tagged job.

When the tagged job is a class-1 job of size x jobs which have to be served before it are all class-1 jobs of size x and all class-2 jobs of size less than a^*. Then the mean workload which the tagged job finds in the system and which has to be served before it is W_{x,a^*}. The mean work which arrives to the system during the sojourn time of the tagged job $T_1(x)$ and have to be done before it takes into account only class-1 jobs of size less than x and class-2 jobs of size less than a^*. So, it equals to $T_1(x)(\rho_x^{(1)} + \rho_{a^*}^{(2)})$.

For the tagged job of class-2 of size $x \leq a*$ the jobs which have to be served before it are class-2 jobs of size less than x. Then the mean workload which the tagged job finds in the system and which has to be served before it is $W_{0,x}$ and the mean time to serve jobs which arrive to the system during $\overline{T}_2(x)$ is $T_2(x)\rho_x^{(2)}$.

For the class-2 job of size $x > a^*$ the jobs which have to be served before it are all class-1 jobs and class-2 jobs of size less than x. Then the mean workload which has to be served before the tagged job is $W_{\infty,x}$ and the mean time spend to serve jobs which arrive during the sojourn time of the current job is $T_2(x)(\rho_\infty^{(1)} + \rho_x^{(2)})$.

Summarizing the results of the previous discussion we get

$$\begin{aligned} T_1(x) &= x + W_{x,a^*} + T_1(x)(\rho_x^{(1)} + \rho_{a^*}^{(2)}) \quad x \geq 0, \\ T_2(x) &= y + W_{0,x} + T_2(x)\rho_x^{(2)}, \quad x \leq a^*, \\ T_2(x) &= y + W_{\infty,x} + T_2(x)(\rho_\infty^{(1)} + \rho_x^{(2)}) \quad x > a^*. \end{aligned}$$

from here we get the proof of the Theorem. □

4.3 Numerical results

Let us calculate numerically for some examples the mean sojourn time in the system when the Gittins policy is used. We consider two classes with the parameters given in Table 4. Also here $p = 0.1$ and the threshold value is $a^* = 7.16$. We compare the obtained results with the mean sojourn times when the system is scheduled with FCFS, PS and LAS policies, the results are given on Figure 12.

TAB. 4 – Exponential and HE classes, simulation parameters

μ_1	μ_2	μ_3	m_1	m_2	ρ_1	ρ_2	ρ
0.6	1.0	0.5	1.6	1.1	0.1	0.4..0.85	0.5..0.95

4.4 Pareto and exponential classes

We can apply the same analysis for the case when class-1 job size distribution is exponential and class-2 job size distribution is Pareto. Let us consider the case when the hazard rate functions of class-1 and class-2 cross at one point.

Let $F_1(x) = 1 - e^{-\mu_1 x}$ and $F_2(x) = 1 - b_2^{c_2}/(x+b_2)^{c_2}$. Then $h_1 = \mu_1$ and $h_2(x) = c_2/(x+b_2)$. The crossing point is $a^* = c_2/\mu_1 - b_2$. When $a^* \leq 0$ the hazard rate functions do not cross and then the optimal policy is to give strict priority to class-1 jobs. If $a^* > 0$ then the hazard rate functions cross at one point and the optimal policy is the same as in the previous section. Then the expressions of the mean conditional sojourn timed of class-1 and class-2 are also (11), (12) and (13).

5 Conclusions

In [Git89], Gittins considered an $M/G/1$ queue and proved that the so-called Gittins index rule minimizes the mean delay. The Gittins rule determines, depending on the jobs attained service, which

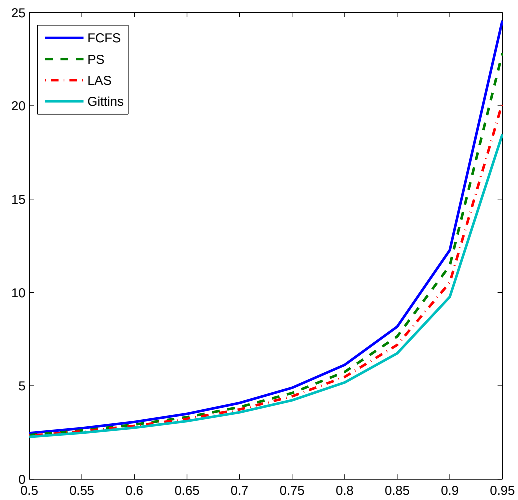

FIG. 12 – Exponential and HE classes, mean sojourn times with respect to the load ρ

job should be served next. Gittins derived this result as a by-product of his groundbreaking results on the multi-armed bandit problem. Gittins' results on the multi-armed bandit problem have had a profound impact and it is extremely highly cited. However, Gittins work in the $M/G/1$ context has not received much attention.

In [AA07], the authors showed that Gittins policy could be used to characterize the optimal scheduling policy when the hazard rate of the service time distribution is not monotone. In the current work we use the Gittins policy to characterize the optimal scheduling discipline in a multi-class queue. Our results show that, even though all service times have a decreasing hazard rate, the optimal policy can significantly differ from LAS, which is known to be optimal in the single-server case. We demonstrate that in particular cases PS has much worse performance than Gittins policy.

Using NS-2 simulator we implemented the Gittins optimal policy in the router queue and provided simulations for several particular schemes. With the simulation results we found that the Gittins policy can achieve 10% gain in comparison with the LAS policy and provides much better performance than the DropTail policy.

In future research we may consider other types of service time distributions. The applicability of our results in real systems like the Internet should also be more carefully evaluated. We also would like to investigate the conditions under which the Gittins policy gives significantly better performance than LAS policy.

6 Appendix : Proof of Theorem 2

We prove that the mean conditional sojourn times in the system described in Section 3 scheduled with the optimal Gittins policy given in Subsection 3.2 are given with (6), (7) and (8).

The class-1 jobs of size $x \leq \theta$ are served in the high priority queue with LAS policy, so the expression for the mean conditional sojourn time for this case is known, see [Kle76, sec. 4.6], as is given by (6).

Let us consider class-1 jobs with sizes $x > \theta$ and class-2 jobs, which are served in the low priority queue. There is a strict priority between the queues and the low priority queue is served only when the high priority queue is empty. Then the low priority queue is a queue with batch arrivals. To find the expressions of the mean conditional sojourn times in the system we use the analysis similar to the one of Kleinrock for Multi Level Processor Sharing queue in [Kle76, sec. 4.7].

In the following analysis we consider only the class-1 jobs of size less than x and class-2 jobs of size less than $g(x)$. So, we consider that the class-1 job size distribution is truncated at x and job size distribution of class-2 is truncated at $g(x)$.

We formulate the following Lemma.

Lemma 1. *The mean conditional sojourn times for class-*1 *job of size* $x > \theta$ *and for class-*2 *job of size* $g(x) > 0$ *equal to*

$$T_1(x) = \frac{\theta + W_{\theta,0}}{1-\rho_\theta^{(1)}} + \frac{\alpha_1(x-\theta, g(x))}{1-\rho_\theta^{(1)}}, \tag{14}$$

$$T_2(g(x)) = \frac{W_{\theta,0}}{1-\rho_\theta^{(1)}} + \frac{\alpha_2(x-\theta, g(x))}{1-\rho_\theta^{(1)}}, \tag{15}$$

where $\alpha_1(x-\theta, g(x))$ *and* $\alpha_2(x-\theta, g(x))$ *are the times spent in the low priority queue by class-*1 *and class-*2 *jobs respectively and equal to*

$$\alpha_1(x-\theta, g(x)) = \frac{x - \theta + A_1(x) + W_b}{1-\rho_b},$$

$$\alpha_2(x-\theta, g(x)) = \frac{g(x) + A_2(g(x)) + W_b}{1-\rho_b},$$

where W_b *is the mean workload in the low priority queue which the tagged batch sees when arrives to the low priority queue,* ρ_b *is the mean load in the low priority queue and* $A_i(x)$, $i = 1, 2$ *are the mean works which arrive to the low priority queue with the tagged job in the batch.*

Démonstration. Let us consider that the tagged job is from class-1 and has a size $x > \theta$. The time it spends in the system consists of the mean time it spends in the high priority queue. This time is $\frac{\theta+W_{\theta,0}}{1-\rho_\theta}$ as it has to be served only with class-1 jobs until it gets θ amount of service. After the tagged job is moved to the low priority queue after waiting while the high priority queue becomes empty. The time $\alpha_1(x-\theta)$ is the time spent by the tagged job in the low priority queue. This time consists of the time spent to serve the job itself, $x - \theta$, of the mean workload in the low priority queue which the tagged job finds, W_b, of the mean work which arrives in the batch with the tagged job, $A_1(x)$ and of the mean work which arrives during the sojourn time of the tagged job, $\alpha_1(x-\theta)\rho_b$.

We use the same analysis for the mean conditional sojourn time of the class-2 job of size $g(x)$. $\square$

Now let us find the expressions for the W_b, ρ_b, $A_1(x)$ and $A_2(x)$. Let us define the truncated distribution $F_{1,\theta,x}(y) = F_1(y), \theta < y < x$ and $F_{1,\theta,x}(y) = 0, y < \theta, y > x$. Let $\overline{X^n_{\theta,x}}^{(i)}$ be the n-th moment and $\rho^{(i)}_{\theta,x}$, $i = 1,2$ be the utilization factor for this truncated distribution. We use this notation because jobs of class-1 which find themselves in a batch are already served until θ.

Let N_i be the random variable which denotes the number of jobs in a batch of class-i, $i = 1,2$. We define $X^{(1)}_{\theta,x}$ as the random variable which denotes the size of class-1 job in a batch. Let $X^{(2)}_{g(x)}$ be the random variable which corresponds to the size of the class-2 job in a batch. Then

$$Y_b = \sum_{i=1}^{N_1} X^{(1)}_{i,\theta,x} + \sum_{i=1}^{N_2} X^{(2)}_{i,g(x)},$$

is the random variable which denotes the size of the batch. Let us denote as λ_b the batch arrival rate. We know that $\lambda_b = \lambda_1 + \lambda_2$. According to the previous notations we can write

$$\rho_b = \lambda_b E[Y_b],$$

here $E[Y_b]$ is the mean work that a batch brings and by Pollaczek-Khinchin

$$W_b = \frac{\lambda_b E[Y_b^2]}{2(1-\rho_b)}.$$

Let us note that W_b does not depend from which class the tagged job comes. As we know the first and the second moments of $X^{(1)}_{\theta,x}$, $X^{(2)}_{g(x)}$, to find ρ_b and W_b we need to know the first and the second moments of N_i, $i = 1,2$. To find this values we we use the method of the Generating functions, which is described in the following section.

6.1 Generating function calculation

We propose a two dimension generating function $G(z_1, z_2)$, which we obtain using collective marks method. The method of the collective marks is described in [Kle75, Ch. 7].

Definition 5. *Let us mark jobs in a batch in the following way. We mark a job of class-*1 *with a probability* $1 - z_1$*, then* z_1 *is a probability that a job of class-*1 *is not marked. The same is defined for jobs of class-*2 *as* z_2*. Let* p_{n_1,n_2} *be the probability that* n_1 *class-*1 *and* n_2 *class-*2 *jobs arrive in the batch. Then*

$$G(z_1, z_2) = \sum_{n1}\sum_{n2} z_1^{n_1} z_2^{n_2} p_{n_1,n_2}$$

is a generation function and it gives a probability that there are no marked jobs in the batch.

Let us define as a "starter" or S a tagged job. Let us distinguish the cases when the starter S belongs to class-1 or class-2 and denote by $G_1(z_1, z_2)$ and $G_2(z_1, z_2)$ the probabilities that there are no marked jobs in the batch if the starter is from the class-1 and class-2. When the $S \in$ class-1, we consider two cases depending on the size of the starter ($S \leq, > \theta$). Then

$$G(z_1, z_2) = \frac{\lambda_1}{\lambda_b}([G_1(z_1, z_2), S \leq \theta] + [G_1(z_1, z_2), S > \theta]) + \frac{\lambda_2}{\lambda_b} G_2(z_1, z_2).$$

Lemma 2. *The Generating function equals to*

$$\begin{aligned} G(z_1, z_2) &= \frac{\lambda_1}{\lambda_b}(\int_0^\theta e^{-\lambda_1 x(1-G_1(z_1,z_2))-\lambda_2 x(1-z_2)} dF_1(x) + \\ &+ z_1 e^{-\lambda_1 \theta(1-G_1(z_1,z_2))-\lambda_2 \theta(1-z_2)} \overline{F}_1(\theta)) + \frac{\lambda_2}{\lambda_b} z_2. \end{aligned} \tag{16}$$

Démonstration. Let us calculate $G_1(z_1, z_2)$. When the class-1 job arrives to the system it creates the busy period. Still this job does not receive θ amount of service the low priority queue is not served. So, jobs which arrive to the low priority queue and jobs which are already in the low priority queue are waiting and so they create a batch. The probability that there are no marked job in this batch is $G_1(z_1, z_2)$.

Let the class-1 job of size x arrives to the system. Let $x \leq \theta$. The probability that k_1 class-1 jobs arrive in the period $(0, x)$ is $P_1(x) = e^{-\lambda_1 x}(\lambda_1 x)^{k_1}/k_1!$. The probability that all the batches generated by this arrived k_1 jobs of class-1 is $G_1(z_1, z_2)^{k_1}$, because each of them generates the batch which does not have marked jobs with probability $G_1(z_1, z_2)$. During time $(0, x)$ the probability that k_2 class-2 jobs arrive to the system is $P_2(x) = e^{-\lambda_2 x}(\lambda_2 x)^{k_2}/k_2!$. The probability that this jobs are not marked is not included in $G_1(z_1, z_2)$ and equals to $z_2^{k_2}$. Then we summarize on k_1 and k_2, integrate on x in $(0, \theta)$ with $dF_1(x)$, as only the class-1 jobs generate busy periods. We get that the probability that there are no marked jobs in the batch is

$$\begin{aligned} [G_1(z_1, z_2), S \leq \theta] &= \int_0^\theta \left(\sum_{k_1=0}^{\infty} P_1(x) G_1(z_1, z_2)^{k_1} P_2(x) z_2^{k_1} \right) dF_1(x) = \\ &= \int_0^\theta e^{-\lambda_1 x(1-G_1(z_1,Z_2))-\lambda_2 x(1-z_2)} dF_1(x). \end{aligned}$$

Let class-1 job of size $x > \theta$ arrives to the system. The class-1 job is first served in the high priority queue until it gets θ of service. Then it is moved to the low priority queue. The probability that k_1 class-1 jobs arrive in the period $(0, \theta)$ is $P_1(\theta) = e^{-\lambda_1 \theta}(\lambda_1 \theta)^{k_1}/k_1!$. The probability that there are no marked jobs in all the batches generated by this arrived k_1 class-1 jobs is $G_1(z)^{k_1}$. The probability that k_2 class-2 jobs arrive to the system in the period $(0, \theta)$ is $P_2(\theta) = e^{-\lambda_2 \theta}(\lambda_2 \theta)^{k_2}/k_2!$. The probability that all this jobs are not marked is z^{k_2}.

We have to take into account the "starter" itself, as it has the size more than θ and it comes in the batch. The probability that the starter is not marked is z_1. Then we summarize on k_1 and k_2, integrate on x on (θ,∞) with $dF_1(x)$, as only the class-2 jobs generate busy periods. We get

$$\begin{aligned}[G_1(z_1,z_2),S>\theta] &= \int_\theta^\infty \left(\sum_{k_1=0}^{\infty} P_1(\theta)G_1(z_1,z_2)^{k_1} z_1 P_2(\theta) z_2^{k_1}\right) dF_1(x) = \\ &= z_1 e^{-\lambda_1\theta(1-G_1(z_1,z_2))-\lambda_2\theta(1-z_2)}\overline{F}_1(\theta).\end{aligned}$$

Let us find $G_2(z_1,z_2)$. When a job of the second class arrives to the system it generates the batch of size one, then the probability that jobs of this batch are not marked is z_2. Then $G_2(z_1,z_2)=z_2$.

$$[G_2(z_1,z_2)] = \int_0^\infty z_2 dF_2(x) = z_2.$$

Finally

$$G(z_1,z_2) = \frac{\lambda_1}{\lambda_b}G_1(z_1,z_2) + \frac{\lambda_2}{\lambda_b}G_2(z_1,z_2),$$

and we get (16). Let us notice that $G(1,1)=1$. □

Now we can calculate $E[N_1]$, $E[N_2]$ and so ρ_b and W_b. After some mathematical calculations we get the following result.

Lemma 3.

$$\begin{aligned}\rho_b &= 1 - \frac{1-\rho_x^{(1)}-\rho_{g(x)}^{(2)}}{1-\rho_\theta^{(1)}}, \\ W_b &= W_{x,g(x)} - W_\theta(1+\rho_b) - \theta\frac{\rho_x^{(1)}-\rho_\theta^{(1)}}{1-\rho_\theta^{(1)}}.\end{aligned}$$

Démonstration. We use the following equations. For $i=1,2$

$$\begin{aligned}E[N_i] &= \frac{\partial G(z_1,z_2)}{\partial z_i}|_{1,1}, \\ E[N_i(N_i-1)] &= E[N_i^2]-E[N_i] = \frac{\partial^2 G(z_1,z_2)}{\partial z_i^2}|_{1,1}, \\ E[N_1N_2] &= \frac{\partial^2 G(z_1,z_2)}{\partial z_1\partial z_2}|_{1,1}.\end{aligned}$$

Using $b_i = \frac{E[N_i^2]}{E[N_i]} - 1$ after some mathematical calculations we obtain the result of the current Lemma. □

Now let us find expressions for $A_1(x)$ and $A_2(x)$.

Lemma 4. *The mean workload which comes with the tagged job of class-*1 *of size* x *in the batch and has to be served before it equals to*

$$A_1(x) = 2(W_\theta + \theta)\rho_b - \theta \frac{\rho^{(2)}_{g(x)}}{1 - \rho^{(1)}_\theta}.$$

Démonstration. The term $A_1(x)$ is the work that arrives with the tagged job of class-1 of size x and that gets served before its departure. Since the tagged job arrives from class-1 only when the batch is started by a class-1 job, the calculations now will depend on $G_1(z_1, z_2)$. We denote $b_{1|1}$ and $b_{2|1}$ the mean number of jobs of class-1 and class-2 which arrive in the batch with the tagged job of class-1 when the batch is initiated by a class-1 job. Then

$$A_1(x) \quad = \quad b_{1|1} E[X^{(1)}_{\theta,x}] + b_{2|1} E[X^{(2)}_{g(x)}] - E[X^{(1)}_{\theta,x}].$$

Here

$$b_{1|1} = \sum_{n_1} n_1 \frac{n_1 P(n_1)}{E[N_{1|1}]} = \frac{E[N^2_{1|1}]}{E[N_{1|1}]},$$

where $N_{1|1}$ is the random variable which corresponds to the number of jobs of class-1 in the batch when the batch is initiated by the class-1 job. So the number of class-1 jobs that arrive in addition to the tagged job is $\left(\frac{E[N^2_{1|1}]}{E[N_{1|1}]} - 1\right)$. Note that since we condition on the fact that the starter is a class-1 job, $N_{1|1}$ is now calculated from $G_1(z_1, z_2)$ so :

$$E[N_{1|1}] = \frac{\partial G_1(z_1, z_2)}{\partial z_1}|_{1,1},$$
$$E[N_{1|1}(N_{1|1} - 1)] = \frac{\partial^2 G_1(z_1, z_2)}{\partial z_1 \partial z_1}|_{1,1}.$$

Then we can find $(b_{1|1} - 1)$. Now we need to calculate $b_{2|1}$, that is, the mean number of class-2 jobs that the tagged job of class-1 job see. We have that from the Generating function $G_1(z_1, z_2)$ by conditioning on the number of class-1 jobs :

$$G_1(z_1, z_2) = \sum_{n1} \sum_{n2} z_1^{n_1} z_2^{n_2} p_{n_1,n_2} = \sum_{n1} \sum_{n2} z_1^{n_1} z_2^{n_2} p_{n_2|n_1} p_{n_1},$$
$$\frac{\partial^2 G_1(z_1, z_2)}{\partial z_1 \partial z_2}|_{1,1} = E[N_1] \sum_{n1} \sum_{n2} n_2 p_{n_2|n_1} \frac{n_1 p_{n_1}}{E[N_1]} = E[N_1] b_{2|1}.$$

Then we can calculate $b_{2|1}$

$$b_{2|1} = \frac{1}{E[N_{1|1}]} \frac{\partial^2 G_1(z_1, z_2)}{\partial z_1 \partial z_2}|_{(1,1)}.$$

Finally we find the expression for $A_1(x)$. □

Lemma 5. *The mean workload which comes with the tagged job of class-*2 *of size* $g(x)$ *in the batch and has to be served before it equals to*

$$A_2(g(x)) = 2(W_\theta + \theta)\rho_b - \theta \frac{\rho^{(2)}_{g(x)}}{1 - \rho^{(1)}_\theta} - \theta\rho_b.$$

Démonstration. The term $A_2(g(x))$ is the work that arrives with the tagged job of size $g(x)$ of class-2 and that gets served before its departure. When the tagged job arrives from class-2 the batch can be started by a class-1 or by a class-2 job, so the calculations depend on $G(z_1, z_2)$. We denote $b_{1|2}$ and $b_{2|2}$ the mean number of jobs of class-1 and class-2 which arrive in the batch with the tagged job of class-2. Then

$$\begin{aligned} A_2(g(x)) &= b_{1|2}E[X^{(1)}_{\theta,x}] + b_{2|2}E[X^{(2)}_{g(x)}] - E[X^{(2)}_{g(x)}] = \\ &= b_{1|2}E[X^{(1)}_{\theta,x}] + (b_{2|2} - 1)E[X^{(2)}_{g(x)}]. \end{aligned}$$

As the tagged job is from class-2, then $b_{2|2} = b_2$. We need to find the value of $b_{1|2}$. We use the fact that jobs of class-1 and class-2 arrive independently from each other.

$$G(z_1, z_2) = \sum_{n1}\sum_{n2} z_1^{n_1} z_2^{n_2} p_{n_1,n_2} = \sum_{n1}\sum_{n2} z_1^{n_1} z_2^{n_2} p_{n_1|n_2} p_{n_2}$$

$$\frac{\partial^2 G(z_1, z_2)}{\partial z_1 \partial z_2}|_{1,1} = E[N_2] \sum_{n1}\sum_{n2} n_1 p_{n_1|n_2} \frac{n_2 p_{n_2}}{E[N_2]} = E[N_2] b_{1|2}.$$

Then

$$b_{1|2} = \frac{1}{E[N_2]} \frac{\partial^2 G(z_1, z_2)}{\partial z_1 \partial z_2}|_{1,1}.$$

From here we get the expression for $A_2(g(x))$. □

Now we can prove the result of Theorem 2.

Lemma 6. *Expressions (14), (15) and (7), (8) are equal.*

Démonstration. After simplification of the expressions (14), (15) we get equations (7), (8). □

Table des matières

Références

[AA07] S. Aalto and U. Ayesta. Mean delay optimization for the $M/G/1$ queue with Pareto type service times. In *Extended abstract in ACM SIGMETRICS 2007, San Diego, CA*, pages 383–384, 2007.

[BNM00] D. Bertsimas and J. Nino-Mora. Restless bandits, linear programming relaxations and a primal-dual index heuristic. *Operations Research*, 48 :80–90, 2000.

[BVW85] C. Buyukkoc, P. Varaya, and J. Walrand. The $c\mu$ rule revisited. *Adv. Appl. Prob.*, 17 :237–238, 1985.

[CB97] M. E. Crovella and A. Bestavros. Self-similarity in World Wide Web traffic : evidence and possible causes. *IEEE/ACM Transactions on Networking*, 5 :835–846, 1997.

[DGNM96] M. Dacre, K. Glazebrook, and J. Niño-Mora. The achievable region approach to the optimal control of stochastic systems. *Journal of the Royal Statistical Society. Series B, Methodological*, 61(4) :747–791, 1996.

[FW99] E. Frostig and G. Weiss. Four proofs of gittins' multiarmed bandit theorem. *Applied Probability Trust*, 1999.

[Git89] J. Gittins. Multi-armed Bandit Allocation Indices. *Wiley, Chichester*, 1989.

[Kle75] L. Kleinrock. *Queueing systems*, volume 1. John Wiley and Sons, 1975.

[Kle76] L. Kleinrock. *Queueing systems*, volume 2. John Wiley and Sons, 1976.

[Kli74] G. Klimov. Time-sharing service systems. i. *Theory of Probability and Its Applications*, 19 :532–551, 1974.

[Kli78] G. Klimov. Time-sharing service systems. ii. *Theory of Probability and Its Applications*, 23 :314–321, 1978.

[NMM98] M. Nabe, M. Murata, and H. Miyahara. Analysis and modeling of World Wide Web traffic for capacity dimensioning of Internet access lines. *Perform. Eval.*, 34(4) :249–271, 1998.

[NT94] P. Nain and D. Towsley. Optimal scheduling in a machine with stochastic varying processing rate. *IEEE/ACM Transactions on Automatic Control*, 39 :1853–1855, 1994.

[Sev74] K. Sevcik. Scheduling for minimum total loss using service time distributions. *Journal of the ACM*, 21 :66–75, 1974.

[SY92] J. Shanthikumar and D. Yao. Multiclass queueing systems : Polymatroidal structure and optimal scheduling control. *Operations Research*, 40(2) :293–299, 1992.

[Tsi93] J.N. Tsitsiklis. A short proof of the Gittins index theorem. In *IEEE CDC*, pages 389–390, 1993.

[VWB] P. Varaiya, J. Walrand, and C. Buyukkoc. Extensions of the multiarmed bandit problem : the discounted case, journal = IEEE Transactions on Automatic Control, volume = 30, number = , pages = 426–439, year = 1985.

[Web92] R. Weber. On the Gittins index for multiarmed bandits. *Annals of Appllied Probability*, 2(4) :1024–1033, 1992.

[Whi88] P. Whittle. Restless bandits : activity allocation in a changing world. *Journal of Applied Probability*, 25 :287–298, 1988.

[Wil01] C. Williamson. Internet traffic measurement. *IEEE Internet Computing*, 5 :70–74, 2001.

[Yas92] S. Yashkov. Mathematical problems in the theory of shared-processor systems. *Journal of Mathematical Sciences*, 58 :101–147, 1992.

Unité de recherche INRIA Sophia Antipolis
2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Unité de recherche INRIA Futurs : Parc Club Orsay Université - ZAC des Vignes
4, rue Jacques Monod - 91893 ORSAY Cedex (France)
Unité de recherche INRIA Lorraine : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)
Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)
Unité de recherche INRIA Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier (France)
Unité de recherche INRIA Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399