



HAL
open science

Batch Processor Sharing with Hyper-Exponential Service Time

Natalia Osipova

► **To cite this version:**

Natalia Osipova. Batch Processor Sharing with Hyper-Exponential Service Time. [Research Report] RR-6180, 2007, pp.20. inria-00144389v2

HAL Id: inria-00144389

<https://inria.hal.science/inria-00144389v2>

Submitted on 3 May 2007 (v2), last revised 21 Jun 2007 (v4)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

*Batch Processor Sharing with
Hyper-Exponential Service Time*

Natalia Osipova

N° 6180

April 2007

Thème COM

 *rapport
de recherche*



Batch Processor Sharing with Hyper-Exponential Service Time

Natalia Osipova

Thème COM — Systèmes communicants
Projet MAESTRO

Rapport de recherche n° 6180 — April 2007 — 20 pages

Abstract: We study Batch Processor-Sharing (BPS) queuing model with hyper-exponential service time distribution and Poisson batch arrival process. One of the main goals to study BPS is the possibility of its application in size-based scheduling, which is used in differentiation between Short and Long flows in the Internet. In the case of hyper-exponential service time distribution we find an analytical expression of the expected conditional response time for the BPS queue. We show, that the expected conditional response time is a concave function of the service time. We apply the received results to the Two Level Processor-Sharing (TLPS) model with hyper-exponential service time distribution and find the expression of the expected response time for the TLPS model. TLPS scheduling discipline can be applied to size-based differentiation in TCP/IP networks and Web server request handling.

Key-words: Batch Processor Sharing, Two Level Processor sharing, Hyper-Exponential distribution, Laplace transform, Cauchy matrices.

La File d'Attente Avec Service à Temps Partagé Avec Arrivées en Rafale

Résumé : Nous étudions la file d'attente avec service à temps partagé avec arrivées en rafale, BPS ("Batch Processor Sharing"), lorsque les temps de service ont une distribution hyper-exponentielle et que le processus d'arrivée des rafales est Poisson. Une des motivations de cette étude est que la file BPS apparaît naturellement lorsque l'on étudie les procédures d'ordonnancement favorisant les connexions courtes sur un réseau internet. Dans le cas d'une distribution hyper-exponentielle des temps de service nous trouvons une expression analytique pour le temps de réponse moyen conditionné sur le temps de service pour une file d'attente BPS. Nous montrons que ce temps de réponse moyen conditionné est une fonction concave du temps de service. Nous appliquons les résultats obtenus à la file d'attente munie d'une politique à Temps Partagées avec Deux-Niveaux "Two Level Processor Sharing" (TLPS) avec distribution hyper-exponentielle des temps de service. Nous trouvons l'expression du temps de réponse moyen pour la file TLPS. La discipline de service TLPS peut être utilisée pour ordonner l'accès aux ressources en fonction de la taille dans un réseau TCP/IP ou sur un serveur Web.

Mots-clés : La file d'attente avec service à temps partagé avec arrivées en rafale, la file d'attente munie d'une politique à Temps Partagées avec Deux-Niveaux, réseau TCP/IP, distribution hyper-exponentielle, Laplace transformant, Matrices de Cauchy.

1 Introduction

The Processor-Sharing (PS) queuing systems are now often used to model communication and computer systems. The PS systems were first introduced by Kleinrock (see [6] and references therein). Under the PS policy each job receives an equal share of the processor. The most important feature of PS is no requirement of knowledge of job sizes.

PS with batch arrivals (BPS) is not yet characterized fully. Kleinrock *et al.* [5] first studied BPS. They found, that the derivative of the expected response time satisfies the integral equation and found the analytical solution in the case when the job size distribution function¹ has the type $F(x) = 1 - p(x)e^{-\mu x}$ where $p(x)$ is a polynomial.

Bansal [4], using Kleinrock's integral equation, obtained the solution for the Laplace transform of the expected conditional service time as a solution of the system of linear equations, when the job size distribution is a hyper-exponential distribution. Also he considers distributions with a rational Laplace transform. Rege and Sengupta [12] obtained the expression for the response time in condition upon the number of customers in the system. Feng and Mishra [8] provided bounds for the expected conditional response time, the bounds depend on the second moment of the service time distribution. Avrachenkov *et al.* [2] prove existence and uniqueness of the Kleinrock's integral equation and provide asymptotic analysis and bounds of the expected conditional response time.

One of the main goals to study BPS is the possibility of its application in size-based scheduling and the possibility to take into account the burstiness of the arrival process. Bursty arrivals often occur in such modern systems as web server. Size-based scheduling is used in differentiation of Short and Long flows in the Internet. A general set of size-based scheduling mechanisms was introduced by Kleinrock and termed as Multy-Level PS (MLPS).

In MLPS jobs are classified into different classes depending on their attained amount of service. Jobs within the same class are served according to FCFC, PS or FB policy. The classes themselves are served according to the FB policy, so the priority is given to the jobs with small sizes.

We study the particular case of MLPS, Two-Level PS (TLPS) scheduling mechanism. It is based on the differentiation of jobs according to some threshold and gives priority to jobs with small sizes. TLPS scheduling mechanism could be used to model such applications as size based differentiation in TCP/IP networks and Web server request differentiation.

It was shown in [1], that when a job size distribution has a decreasing hazard rate, then with the selection of the threshold the expected sojourn time of the TLPS system could be

¹We shall use terms "job size" and "service time" interchangeably.

reduced in comparison to pure PS system.

The distribution of file sizes in the Internet has a decreasing hazard rate and often modelled with heavy-tailed distributions. In [9] it is shown, that heavy-tailed distributions such as Pareto distribution could be approximated with hyper-exponential distributions with a significant number of phases.

So, we study the TLPS model with hyper-exponential service time distribution. We apply the results of BPS queuing model for the TLPS model and find the expression of expected conditional sojourn time for TLPS model. We prove the concavity of the expected conditional sojourn time function for both BPS and TLPS models. We use this results in [3]. We note, that the concavity of the expected conditional sojourn time for BPS was proven by another method in [10].

The paper is organized as follows. In the First section the BPS scheduling mechanism is considered in the case when the job service time distribution is a hyper-exponential distribution. For this case we provide the solution of the Kleinrok's integral equation according to the expected conditional service time. In the Second Section the results received for the BPS model are applied to the TLPS model, where the job size distribution is also hyper-exponential. The analytical expression of the expected sojourn time is found. We put some technical proofs in the Appendix.

2 Batch Arrival PS queue analysis

2.1 Main definitions

Let us consider an M/G/1 system with batch arrivals and Processor-Sharing (PS) queuing discipline. The batches arrive according to a Poisson process with arrival rate λ . Let $E[N] = \bar{n}$ be the average size of a batch. Let b be the average number of jobs that arrive with (and in addition to) the tagged job, $b + 1 = E[N^2]/E[N]$.

$B(x)$ is the required job size (service time) distribution and $\bar{B}(x) = 1 - B(x)$ is its complementary distribution function.

The load is given by $\rho = \lambda \bar{n} m$, with $m = \int_0^\infty B(x) dx = \int_0^\infty x dB(x)$. We consider that the system is stable, $\rho < 1$.

Let $\alpha(x)$ be the expected conditional response time for a job with service time x and $\alpha'(x)$ be its derivative. Kleinrock showed in [6] that $\alpha'(x)$ satisfies the following integro-differential equation

$$\alpha'(x) = \lambda \bar{n} \int_0^\infty \alpha'(y) \bar{B}(x+y) dy + \lambda \bar{n} \int_0^x \alpha'(y) \bar{B}(x-y) dy + b \bar{B}(x) + 1. \quad (1)$$

We are interested in the case, when the job size distribution is a hyper-exponential function

$$B(x) = 1 - \sum_{i=1}^N p_i e^{-\mu_i x}, \quad \sum_i p_i = 1, \quad p_i > 0, \quad \mu_i > 0, \quad \forall i = 1, \dots, N, \quad 1 < N \leq \infty. \quad (2)$$

By \sum_i or \prod_i we will mean $\sum_{i=1}^N$ and $\prod_{i=1}^N$. By $\sum_{i \neq j}$ or $\prod_{i \neq j}$ we will mean $\sum_{i=1, \dots, N, i \neq j}$ and $\prod_{i=1, \dots, N, i \neq j}$. By $\forall i$ we will mean $\forall i = 1, \dots, N$.

Without loss of generality, we can assume, that

$$0 < \mu_N < \mu_{N-1} < \dots < \mu_2 < \mu_1 < \infty. \quad (3)$$

2.2 The expected conditional sojourn time for BPS

Let us first prove additional Lemma.

Lemma 1 . *The zeros b_i of the rational function*

$$\Psi(s) \stackrel{def}{=} 1 - \lambda \bar{n} \sum_i \frac{p_i}{s + \mu_i} = \frac{\prod_i (s + b_i)}{\prod_i (s + \mu_i)} \quad (4)$$

are all real, distinct, positive and satisfy the following inequalities:

$$0 < b_N < \mu_N, \quad \mu_{i+1} < b_i < \mu_i, \quad \forall i = 1, \dots, N-1.$$

Proof. Let us study the following equation

$$\Psi(s) = 0$$

Repeating the considerations of [7] we get, that it has N_1 roots $-b_i$, $i = 1, \dots, N_1$, where N_1 is the number of distinct elements within μ_i . As μ_i , $i = 1, \dots, N$ are all different, $\mu_i \neq \mu_j$, $i \neq j$, then $N_1 = N$ and there are N different roots $-b_i$. All $-b_i$ are real, distinct, negative, satisfy the following inequalities: $0 > -b_N > -\mu_N$, $-\mu_{i+1} > -b_i > -\mu_i$, $\forall i = 1, \dots, N-1$. With this we have the statement of Lemma 1. So, then it is possible to present $\Psi(s)$ in the following form:

$$\Psi(s) = \frac{\prod_i (s + b_i)}{\prod_i (s + \mu_i)},$$

and from here we have (4). □

Now we can prove our main result.

Theorem 2 . *The derivative of the expected conditional response time for BPS system with hyper-exponential job size distribution function given by (2) is given by:*

$$\alpha'(x) = c_0 + \sum_k c_k e^{-b_k x},$$

where

$$c_k = \frac{b}{2\lambda\bar{n}} \left(\frac{\prod_q (\mu_q^2 - b_k^2)}{b_k \prod_{q \neq k} (b_q^2 - b_k^2)} \right),$$

and where b_k are the solutions of the following equation:

$$1 - \lambda\bar{n} \sum_i \frac{p_i}{s + \mu_i} = 0,$$

and are all positive, distinct, real, satisfy the following inequalities

$$0 < b_N < \mu_N, \quad \mu_{i+1} < b_i < \mu_i, \quad \forall i = 1, \dots, N-1.$$

Proof. We can rewrite integral equation (1) in the following way:

$$\begin{aligned} \alpha'(x) &= \lambda\bar{n} \int_0^\infty \alpha'(y) \sum_i p_i e^{-\mu_i(x+y)} dy + \lambda\bar{n} \int_0^x \alpha'(y) \bar{B}(x-y) dy + b\bar{B}(x) + 1, \\ \alpha'(x) &= \lambda\bar{n} \sum_i p_i e^{-\mu_i x} \int_0^\infty \alpha'(y) e^{-\mu_i y} dy + \lambda\bar{n} \int_0^x \alpha'(y) \bar{B}(x-y) dy + b\bar{B}(x) + 1. \end{aligned}$$

We note that in the latter equation $\int_0^\infty \alpha'(y)e^{-\mu_i y} dy$, $i = 1, \dots, N$ are the Laplace transforms of $\alpha'(y)$ evaluated at μ_i , $i = 1, \dots, N$. Denote

$$L_i = \int_0^\infty \alpha'(y)e^{-\mu_i y} dy, \quad i = 1, \dots, N.$$

Then, we have

$$\alpha'(x) = \lambda \bar{n} \sum_i p_i L_i e^{-\mu_i x} + \lambda \bar{n} \int_0^x \alpha'(y) \bar{B}(x-y) dy + b \bar{B}(x) + 1.$$

Now taking the Laplace transform of the above equation and using the convolution property, we get the following equation. Here we denote $L_{\alpha'}(s)$ the Laplace transform of $\alpha'(x)$.

$$\begin{aligned} L_{\alpha'}(s) &= \lambda \bar{n} \sum_i \frac{p_i L_i}{s + \mu_i} + \lambda \bar{n} \sum_i \frac{p_i L_{\alpha'}(s)}{s + \mu_i} + b \sum_i \frac{p_i}{s + \mu_i} + \frac{1}{s}, \\ \implies L_{\alpha'}(s) \left(1 - \lambda \bar{n} \sum_i \frac{p_i}{s + \mu_i} \right) &= \lambda \bar{n} \sum_i \frac{p_i L_i}{s + \mu_i} + b \sum_i \frac{p_i}{s + \mu_i} + \frac{1}{s}. \end{aligned}$$

Let us note that $L_{\alpha'}(\mu_i) = L_i$, $i = 1, \dots, N$. According to Lemma 1 and (4), the following equation holds:

$$L_{\alpha'}(s) \frac{\prod_i (s + b_i)}{\prod_i (s + \mu_i)} = \lambda \bar{n} \sum_i \frac{p_i L_i}{s + \mu_i} + b \sum_i \frac{p_i}{s + \mu_i} + \frac{1}{s}, \quad (5)$$

$$\implies L_{\alpha'}(s) = \lambda \bar{n} \sum_i p_i L_i \frac{\prod_{k \neq i} (s + \mu_k)}{\prod_i (s + b_i)} + b \sum_i p_i \frac{\prod_{k \neq i} (s + \mu_k)}{\prod_i (s + b_i)} + \frac{1}{s} \frac{\prod_i (s + \mu_i)}{\prod_i (s + b_i)}. \quad (6)$$

From here we see, that there exist such c_k , that:

$$L_{\alpha'}(s) = \frac{c_0}{s} + \sum_k \frac{c_k}{s + b_k}. \quad (7)$$

Then, taking the inversion of the Laplace transform, we have the expression for $\alpha'(x)$:

$$\alpha'(x) = c_0 + \sum_k c_k e^{-b_k x}.$$

Now let us find c_k . To find c_0 let us multiply both parts of the equation (7) by s and find the value at the point $s = 0$. Then, we have that

$$c_0 = L_{\alpha'}(s) s|_{s=0} = \frac{\prod_i \mu_i}{\prod_i b_i}. \quad (8)$$

From (4) we have the following:

$$\frac{\prod_i b_i}{\prod_i \mu_i} = \Psi(s)|_{s=0} = 1 - \lambda \bar{n} \sum_i \frac{p_i}{\mu_i} = 1 - \lambda \bar{n} m = 1 - \rho.$$

So, then

$$c_0 = \frac{1}{1 - \rho}.$$

Let us find the other coefficients c_i .

Let us denote:

$$\begin{aligned} L_{\alpha'}^*(s) &= \sum_i \frac{c_i}{s + b_i}, \\ \implies L_{\alpha'}(s) &= \frac{c_0}{s} + \sum_i \frac{c_i}{s + b_i} = \frac{c_0}{s} + L_{\alpha'}^*(s). \end{aligned}$$

According to the system (5), (6) we have the following:

$$\begin{aligned} L_j^* &= \sum_i \frac{c_i}{\mu_j + b_i}, \quad \forall j, \\ (5) \implies L_{\alpha'}^*(s) \frac{\prod_i (s + b_i)}{\prod_i (s + \mu_i)} &= \lambda \bar{n} \sum_i \frac{p_i L_i^*}{s + \mu_i} + b \sum_i \frac{p_i}{s + \mu_i}, \\ \implies L_{\alpha'}^*(s) \frac{\prod_i (s + b_i)}{\prod_i (s + \mu_i)} (s + \mu_q)|_{s=-\mu_q} &= \lambda \bar{n} \sum_i \frac{p_i L_i^*}{s + \mu_i} (s + \mu_q)|_{s=-\mu_q} + \\ &\quad + b \sum_i \frac{p_i}{s + \mu_i} (s + \mu_q)|_{s=-\mu_q}, \quad \forall q, \\ \implies \sum_j \frac{c_j}{b_j - \mu_q} \frac{\prod_i (b_i - \mu_q)}{\prod_{i \neq q} (\mu_i - \mu_q)} &= \lambda \bar{n} p_q L_q^* + b p_q, \quad \forall q, \\ \implies \sum_j \frac{c_j}{b_j - \mu_q} \frac{\prod_i (b_i - \mu_q)}{\prod_{i \neq q} (\mu_i - \mu_q)} &= \lambda \bar{n} p_q \sum_j \frac{c_j}{b_j + \mu_q} + b p_q, \quad \forall q. \end{aligned}$$

Let us notice, that from (4) we have the following:

$$\begin{aligned} \frac{\prod_i (b_i - \mu_q)}{\prod_{i \neq q} (\mu_i - \mu_q)} &= \frac{\prod_i (s + b_i)}{\prod_i (s + \mu_i)} (s + \mu_q)|_{s=-\mu_q} = \Psi(s) (s + \mu_q)|_{s=-\mu_q} = \\ &= \left(1 - \lambda \bar{n} \sum_i \frac{p_i}{s + \mu_i} \right) (s + \mu_q)|_{s=-\mu_q} = -\lambda \bar{n} p_q, \quad \forall q, \end{aligned}$$

then

$$\begin{aligned} \Rightarrow & \sum_j \frac{c_j}{b_j - \mu_q} (-\lambda \bar{n} p_q) = \lambda \bar{n} p_q \sum_j \frac{c_j}{b_j + \mu_q} + b p_q, \quad \forall q, \\ \Rightarrow & \sum_j \frac{c_j}{\mu_q - b_j} - \sum_j \frac{c_j}{\mu_q + b_j} = \frac{b}{\lambda \bar{n}}, \quad \forall q, \\ \Rightarrow & \sum_j \frac{c_j b_j}{\mu_q^2 - b_j^2} = \frac{b}{2\lambda \bar{n}}, \quad \forall q. \end{aligned}$$

So, c_j are solutions of the following linear system:

$$\sum_j \frac{c_j b_j}{\mu_q^2 - b_j^2} = \frac{b}{2\lambda \bar{n}}, \quad q = 1, \dots, N. \quad (9)$$

Let us give the additional Lemma.

Lemma 3 . *The solution of the following system of linear equations:*

$$\sum_j \frac{x_j}{\mu_q^2 - b_j^2} = 1, \quad q = 1, \dots, N, \quad (10)$$

is given by

$$x_k = \frac{\prod_{q=1, \dots, N} (\mu_q^2 - b_k^2)}{\prod_{q \neq k} (b_q^2 - b_k^2)}, \quad k = 1, \dots, N.$$

Proof. The proof is given in the appendix. □

Corollary 4 . *The solution of equation (10) is positive. Namely, $x_k > 0$ for $k = 1, \dots, N$.*

Proof. The proof is given in the appendix.

If we denote

$$x_k = \frac{c_k b_k}{\frac{b}{2\lambda \bar{n}}}, \quad j = 1, \dots, N,$$

the system (9) will take the form (10) and for c_j the final result is as follows:

$$c_k = \frac{b}{2\lambda \bar{n}} \left(\frac{x_k}{b_k} \right) = \frac{b}{2\lambda \bar{n}} \left(\frac{\prod_q (\mu_q^2 - b_k^2)}{b_k \prod_{q \neq k} (b_q^2 - b_k^2)} \right), \quad \forall k = 1, \dots, N.$$

This completes the proof of Theorem 2.

□

Corollary 5 . *The expected conditional sojourn time function in the BPS system with hyper-exponential job size distribution given by (2) is a concave function.*

Proof. The function

$$\alpha(x) = c_0 x - \sum_k \frac{c_k}{b_k} e^{-b_k x} + \sum_k \frac{c_k}{b_k}, \quad \alpha(0) = 0$$

is a concave function if $\alpha''(x) \leq 0$.

$$\alpha''(x) = - \sum_k c_k b_k e^{-b_k x} \leq 0$$

as $c_k \geq 0, b_k > 0, k = 1, \dots, N$ which follows from Corollary 4 and Lemma 1.

□

3 Application to TLPS model

3.1 Main definitions

We study the Two Level Processor Sharing (TLPS) scheduling discipline with the hyper-exponential job size distribution. The model description is the following.

Jobs arrive to the system according to a Poisson process with rate λ .

Let θ be a given threshold. The jobs in the system that attained a service less than θ are assigned to the high priority queue. If in addition there are jobs with attained service greater than θ , such a job is separated into two parts. The first part of size θ is assigned to the high priority queue and the second part of size $x - \theta$ waits in the lower priority queue. The low priority queue is served when the high priority queue is empty. Both queues are served according to the Processor Sharing (PS) discipline.

Let us denote the job size distribution by $F(x)$. By $\bar{F}(x) = 1 - F(x)$ we denote the complementary distribution function. We consider the case, when $F(x)$ is a hyper-exponential distribution function, so

$$F(x) = 1 - \sum_{i=1}^N \tilde{p}_i e^{-\mu_i x}, \quad \sum_i \tilde{p}_i = 1, \quad \tilde{p}_i > 0, \quad \mu_i > 0, \quad \forall i. \quad (11)$$

The mean job size is given by $m = \int_0^\infty x dF(x)$ and the system load is $\rho = \lambda m$. We assume that the system is stable ($\rho < 1$) and is in steady state.

3.2 Expected conditional sojourn time in TLPS system

Let us denote by $\bar{T}^{TLPS}(x)$ the expected conditional sojourn time in the TLPS system for a job of size x and by $\bar{T}(\theta)$ the expected sojourn time of the system.

According to [6] the expected conditional sojourn time of the system is given by:

$$\bar{T}^{TLPS}(x) = \begin{cases} \frac{x}{1 - \rho_\theta}, & x \in [0, \theta], \\ \frac{\bar{W}(\theta) + \theta + \alpha(x - \theta)}{1 - \rho_\theta}, & x \in (\theta, \infty), \end{cases}$$

where ρ_θ is the utilization factor for the truncated distribution $\rho_\theta = \lambda \bar{X}_\theta^1$, the n -th moment of the distribution truncated at θ is

$$\bar{X}_\theta^n = \int_0^\theta n y^{n-1} \bar{F}(y) dy,$$

$(\overline{W}(\theta) + \theta)/(1 - \rho_\theta)$ expresses the time needed to reach the low priority queue. This time consists of the time $\theta/(1 - \rho_\theta)$ spent in the high priority queue, where the flow is served up to the threshold θ , plus the time $\overline{W}(\theta)/(1 - \rho_\theta)$ which is spent waiting for the high priority queue to empty. Here $\overline{W}(\theta) = \frac{\lambda X_\theta^2}{2(1 - \rho_\theta)}$. The remaining term $\alpha(x - \theta)/(1 - \rho_\theta)$ is the time spent in the low priority queue.

To find $\alpha(x)$ we use the interpretation of the lower priority queue as a PS system with batch arrivals.

Let us denote

$$\overline{F}_\theta^i = \tilde{p}_i e^{-\mu_i \theta}, \quad i = 1, \dots, N,$$

and prove the following Theorem:

Theorem 6 . *The derivative of the expected conditional response time in TLPS system, when the job size distribution is a hyper-exponential distribution function given by (11) is given by:*

$$\begin{aligned} \alpha'(x) &= c_0(\theta) + \sum_k c_k(\theta) e^{-b_k(\theta)x}, \\ c_0(\theta) &= \frac{1 - \rho_\theta}{1 - \rho}, \\ c_k(\theta) &= \frac{b}{2\lambda\bar{n}} \left(\frac{\prod_{q=1, \dots, N} (\mu_q^2 - b_k^2(\theta))}{b_k(\theta) \prod_{q \neq k} (b_q^2(\theta) - b_k^2(\theta))} \right), \quad k = 1, \dots, N, \end{aligned}$$

and $b_i(\theta)$ are roots of

$$1 - \frac{\lambda}{1 - \rho_\theta} \sum_i \frac{\overline{F}_\theta^i}{s + \mu_i} = 0,$$

and satisfy the following inequalities:

$$0 < b_N(\theta) < \mu_N, \quad \mu_{i+1} < b_i(\theta) < \mu_i, \quad \forall i = 1, \dots, N - 1.$$

Proof. As was shown in [6], $\alpha'(x) = d\alpha/dx$ is the solution of the integral equation we looked at in Section 2

$$\alpha'(x) = \lambda\bar{n} \int_0^\infty \alpha'(y)B(x+y)dy + \lambda\bar{n} \int_0^x \alpha'(y)B(x-y)dy + bB(x) + 1.$$

Here the average batch size is given by $\bar{n} = \frac{1 - F(\theta)}{1 - \rho_\theta}$ and the average number of jobs that arrive to the low priority queue in addition to the tagged job is given by $b = \frac{2\lambda(1 - F(\theta))(\overline{W}(\theta) + \theta)}{(1 - \rho_\theta)}$.

The complementary truncated distribution is $\overline{B}(x) = \frac{1-F(x+\theta)}{1-F(\theta)}$, then:

$$\begin{aligned}\overline{B}(x) &= \frac{\overline{F}(x+\theta)}{\overline{F}(\theta)} = \frac{1}{\overline{F}(\theta)} \sum_i \tilde{p}_i e^{-\mu_i(x+\theta)} = \sum_i \frac{\tilde{p}_i e^{-\mu_i\theta}}{\overline{F}(\theta)} e^{-\mu_i x} = \sum_i p_i e^{-\mu_i x}, \\ p_i &= \frac{1}{\overline{F}(\theta)} \tilde{p}_i e^{-\mu_i\theta} = \frac{1}{\overline{F}(\theta)} \overline{F}_\theta^i, \quad \forall i.\end{aligned}$$

We apply the results of Theorem 2 for the TLPS model. To calculate c_0 we use (8) and (4) and get the following

$$\frac{1}{c_0(\theta)} = \frac{\prod_i b_i(\theta)}{\prod_i \mu_i} = \Psi(s)|_{s=0} = 1 - \lambda \bar{n} \sum_i \frac{p_i}{\mu_i} = 1 - \frac{\lambda}{1-\rho\theta} \sum_i \frac{\overline{F}_\theta^i}{\mu_i} = 1 - \frac{\lambda(m - \overline{X}_\theta^1)}{1-\rho\theta} = \frac{1-\rho}{1-\rho\theta}.$$

So, then

$$c_0(\theta) = \frac{1-\rho\theta}{1-\rho}.$$

For $c_k(\theta), b_k(\theta)$ we use the results of Theorem 2 and get the the statement of Theorem 6. □

Corollary 7 . *The coefficients $c_k(\theta) \geq 0$, $k = 1, \dots, N$.*

Proof. As

$$\begin{aligned}0 &< b_N(\theta) < \mu_N, \quad \mu_{i+1} < b_i(\theta) < \mu_i, \quad \forall i = 1, \dots, N-1, \\ 0 &< \mu_N < \mu_{N-1} < \dots < \mu_2 < \mu_1 < \infty,\end{aligned}$$

then the statement of Corollary holds. □

Corollary 8 . *The expected conditional sojourn time function in the TLPS system with hyper-exponential job size distribution given by (11) is a concave function with respect to job sizes.*

Proof. The function

$$\alpha(x) = c_0(\theta)x - \sum_k \frac{c_k(\theta)}{b_k(\theta)} e^{-b_k(\theta)x} + \sum_k \frac{c_k(\theta)}{b_k(\theta)}, \quad \alpha(0) = 0$$

is a concave function if $\alpha''(x) \leq 0$.

$$\alpha''(x) = - \sum_k c_k(\theta) b_k(\theta) e^{-b_k(\theta)x} \leq 0$$

as $c_k(\theta) \geq 0, b_k(\theta) > 0$, $k = 1, \dots, N$, as it follows from Corollary 7 and Theorem 6. □

3.3 Expected sojourn time for TLPS model

Theorem 9 . *The expected sojourn time in the TLPS system with hyper-exponential distribution function is given by the following equation:*

$$\bar{T}(\theta) = \frac{\bar{X}_1^\theta + \bar{W}(\theta)\bar{F}(\theta)}{1 - \rho_\theta} + \frac{(m - \bar{X}_1^\theta)}{1 - \rho} + \frac{(\bar{W}(\theta) + \theta)}{1 - \rho_\theta} \sum_{i,j} \frac{\bar{F}_\theta^i}{b_j(\theta)(\mu_i + b_j(\theta))} \frac{\prod_q (\mu_q^2 - b_j^2(\theta))}{\prod_{q \neq k} (b_q^2(\theta) - b_j^2(\theta))},$$

where $b_i(\theta)$ are defined as in Theorem 6.

Proof. According to [6] and Theorem 6 we have the following:

$$\begin{aligned} \bar{T}(\theta) &= \frac{\bar{X}_\theta^1 + \bar{W}(\theta)\bar{F}(\theta)}{1 - \rho_\theta} + \frac{1}{1 - \rho_\theta} \bar{T}^{BPS}(\theta), \\ \bar{T}^{BPS}(\theta) &= \int_\theta^\infty \alpha(x - \theta) dF(x) = \int_0^\infty \alpha'(x) \bar{F}(x + \theta) dx, \\ (\text{Theorem 6}) \implies \bar{T}^{BPS}(\theta) &= c_0(\theta)(m - \bar{X}_\theta^1) + \sum_{i,j} \frac{\bar{F}_\theta^i c_j(\theta)}{\mu_i + b_j(\theta)}, \end{aligned}$$

then

$$\bar{T}(\theta) = \frac{\bar{X}_\theta^1 + \bar{W}(\theta)\bar{F}(\theta)}{1 - \rho_\theta} + \frac{(m - \bar{X}_\theta^1)}{1 - \rho} + \frac{(\bar{W}(\theta) + \theta)}{1 - \rho_\theta} \sum_{i,j} \frac{\bar{F}_\theta^i}{b_j(\theta)(\mu_i + b_j(\theta))} \frac{\prod_q (\mu_q^2 - b_j^2(\theta))}{\prod_{q \neq j} (b_q^2(\theta) - b_j^2(\theta))}.$$

□

4 Appendix

Lemma 10 . *The solution of the following system of linear equations*

$$\sum_j \frac{x_j}{\mu_q^2 - b_j^2} = 1, \quad q = 1, \dots, N, \quad (12)$$

is given by:

$$x_k = \frac{\prod_{q=1, \dots, N} (\mu_q^2 - b_k^2)}{\prod_{q \neq k} (b_q^2 - b_k^2)}.$$

Proof. Let

$$\begin{aligned} x &= [x_j]_{j=1:N}, \\ \underline{1} &= [1, \dots, 1]_N \end{aligned}$$

be two vectors of size N ,

$$D = \begin{pmatrix} \frac{1}{\mu_1^2 - b_1^2} & \frac{1}{\mu_1^2 - b_2^2} & \cdots & \frac{1}{\mu_1^2 - b_N^2} \\ \frac{1}{\mu_2^2 - b_1^2} & \frac{1}{\mu_2^2 - b_2^2} & \cdots & \frac{1}{\mu_2^2 - b_N^2} \\ \dots & \dots & \dots & \dots \\ \frac{1}{\mu_N^2 - b_1^2} & \frac{1}{\mu_N^2 - b_2^2} & \cdots & \frac{1}{\mu_N^2 - b_N^2} \end{pmatrix},$$

then equation (12) could be rewritten

$$Dx = \underline{1},$$

and with the Cramer formulas [11] the following is correct:

$$\begin{aligned} x_k &= \frac{\det D_k}{\det D} = \frac{\det [D_{[1]}, \dots, D_{[k-1]}, \underline{1}, D_{[k+1]}, \dots, D_{[N]}]}{\det D}, \quad k = 1, \dots, N, \\ D_k &= [D_{[1]}, \dots, D_{[k-1]}, \underline{1}, D_{[k+1]}, \dots, D_{[N]}]. \end{aligned}$$

Since D is a Cauchy matrix, its determinant is known [11]:

$$\det D = \frac{\prod_{1 \leq j < k \leq N} (\mu_j^2 - \mu_k^2)(b_k^2 - b_j^2)}{\prod_{j, k=1, \dots, N} (\mu_j^2 - b_k^2)}. \quad (13)$$

As $0 < b_N < \mu_N < \dots < \mu_{i+1} < b_i < \mu_i < \dots < b_1 < \mu_1$, then $\det D > 0$ and we can use Cramer formulas to calculate x_k . Let us find

$$\det D_k = \det [D_{[1]}, \dots, D_{[k-1]}, \mathbf{1}, D_{[k+1]}, \dots, D_{[N]}].$$

$$\begin{aligned} D_k &= \begin{pmatrix} \frac{1}{\mu_1^2 - b_1^2} & \dots & \frac{1}{\mu_1^2 - b_{k-1}^2} & 1 & \frac{1}{\mu_1^2 - b_{k+1}^2} & \dots & \frac{1}{\mu_1^2 - b_N^2} \\ \frac{1}{\mu_2^2 - b_1^2} & \dots & \frac{1}{\mu_2^2 - b_{k-1}^2} & 1 & \frac{1}{\mu_2^2 - b_{k+1}^2} & \dots & \frac{1}{\mu_2^2 - b_N^2} \\ \dots & & & & & & \\ \frac{1}{\mu_N^2 - b_1^2} & \dots & \frac{1}{\mu_N^2 - b_{k-1}^2} & 1 & \frac{1}{\mu_N^2 - b_{k+1}^2} & \dots & \frac{1}{\mu_N^2 - b_N^2} \end{pmatrix} \\ \det D_k &= \begin{vmatrix} \frac{1}{\mu_1^2 - b_1^2} & \dots & \frac{1}{\mu_1^2 - b_{k-1}^2} & 1 & \frac{1}{\mu_1^2 - b_{k+1}^2} & \dots & \frac{1}{\mu_1^2 - b_N^2} \\ \frac{1}{\mu_2^2 - b_1^2} & \dots & \frac{1}{\mu_2^2 - b_{k-1}^2} & 1 & \frac{1}{\mu_2^2 - b_{k+1}^2} & \dots & \frac{1}{\mu_2^2 - b_N^2} \\ \dots & & & & & & \\ \frac{1}{\mu_N^2 - b_1^2} & \dots & \frac{1}{\mu_N^2 - b_{k-1}^2} & 1 & \frac{1}{\mu_N^2 - b_{k+1}^2} & \dots & \frac{1}{\mu_N^2 - b_N^2} \end{vmatrix}_N \\ &= (-1)^{k-1} \begin{vmatrix} 1 & \frac{1}{\mu_1^2 - b_1^2} & \dots & \frac{1}{\mu_1^2 - b_{k-1}^2} & \frac{1}{\mu_1^2 - b_{k+1}^2} & \dots & \frac{1}{\mu_1^2 - b_N^2} \\ 1 & \frac{1}{\mu_2^2 - b_1^2} & \dots & \frac{1}{\mu_2^2 - b_{k-1}^2} & \frac{1}{\mu_2^2 - b_{k+1}^2} & \dots & \frac{1}{\mu_2^2 - b_N^2} \\ \dots & & & & & & \\ 1 & \frac{1}{\mu_N^2 - b_1^2} & \dots & \frac{1}{\mu_N^2 - b_{k-1}^2} & \frac{1}{\mu_N^2 - b_{k+1}^2} & \dots & \frac{1}{\mu_N^2 - b_N^2} \end{vmatrix}_N \end{aligned}$$

For simplify the following computations let us denote the following notations:

$$\begin{aligned} \beta_i &= -b_{i-1}^2, i = 2, \dots, k, \\ \beta_i &= -b_i^2, i = k+1, \dots, N. \end{aligned}$$

Then

$$\begin{aligned} \det D_k &= (-1)^{k-1} \begin{vmatrix} 1 & \frac{1}{\mu_1^2 + \beta_2} & \dots & \frac{1}{\mu_1^2 + \beta_k} & \frac{1}{\mu_1^2 + \beta_{k+1}} & \dots & \frac{1}{\mu_1^2 + \beta_N} \\ 1 & \frac{1}{\mu_2^2 + \beta_2} & \dots & \frac{1}{\mu_2^2 + \beta_k} & \frac{1}{\mu_2^2 + \beta_{k+1}} & \dots & \frac{1}{\mu_2^2 + \beta_N} \\ \dots & & & & & & \\ 1 & \frac{1}{\mu_N^2 + \beta_2} & \dots & \frac{1}{\mu_N^2 + \beta_k} & \frac{1}{\mu_N^2 + \beta_{k+1}} & \dots & \frac{1}{\mu_N^2 + \beta_N} \end{vmatrix}_N \\ \det D_k &= (-1)^{k-1} \begin{vmatrix} 1 & \frac{1}{\mu_1^2 + \beta_2} & \dots & \frac{1}{\mu_1^2 + \beta_k} & \dots & \frac{1}{\mu_1^2 + \beta_N} \\ 0 & \frac{\mu_1^2 - \mu_2^2}{(\mu_2^2 + \beta_2)(\mu_1^2 + \beta_2)} & \dots & \frac{\mu_1^2 - \mu_2^2}{(\mu_2^2 + \beta_k)(\mu_1^2 + \beta_k)} & \dots & \frac{\mu_1^2 - \mu_2^2}{(\mu_2^2 + \beta_N)(\mu_1^2 + \beta_N)} \\ \dots & & & & & \\ 0 & \frac{\mu_1^2 - \mu_N^2}{(\mu_N^2 + \beta_2)(\mu_1^2 + \beta_2)} & \dots & \frac{\mu_1^2 - \mu_N^2}{(\mu_N^2 + \beta_k)(\mu_1^2 + \beta_k)} & \dots & \frac{\mu_1^2 - \mu_N^2}{(\mu_N^2 + \beta_N)(\mu_1^2 + \beta_N)} \end{vmatrix}_N \\ &= (-1)^{k-1} \frac{(\mu_1^2 - \mu_2^2) \dots (\mu_1^2 - \mu_N^2)}{(\mu_1^2 + \beta_2) \dots (\mu_1^2 + \beta_N)} \begin{vmatrix} \frac{1}{\mu_2^2 + \beta_2} & \dots & \frac{1}{\mu_2^2 + \beta_k} & \frac{1}{\mu_2^2 + \beta_{k+1}} & \dots & \frac{1}{\mu_2^2 + \beta_N} \\ \dots & & & & & \\ \frac{1}{\mu_N^2 + \beta_2} & \dots & \frac{1}{\mu_N^2 + \beta_k} & \frac{1}{\mu_N^2 + \beta_{k+1}} & \dots & \frac{1}{\mu_N^2 + \beta_N} \end{vmatrix}_{N-1} \end{aligned}$$

So, as the received matrix is a Cauchy matrix of size $N - 1$, the following equation holds:

$$\det D_k = (-1)^{k-1} \frac{(\mu_1^2 - \mu_2^2) \dots (\mu_1^2 - \mu_N^2)}{(\mu_1^2 + \beta_2) \dots (\mu_1^2 + \beta_N)} \frac{\prod_{2 \leq j < p \leq N} (\mu_j^2 - \mu_p^2)(\beta_j - \beta_p)}{\prod_{j,q=2,\dots,N} (\mu_j^2 + \beta_q)}.$$

Let us remember, that $\beta_i = -b_{i-1}^2$, $i = 2, k$, $\beta_i = -b_i^2$, $i = k + 1, \dots, N$, then

$$\begin{aligned} \det D_k &= (-1)^{k-1} \frac{\prod_{q=2,\dots,N} (\mu_1^2 - \mu_q^2)}{\prod_{q=2,\dots,N} (\mu_1^2 + \beta_q)} \frac{\prod_{2 \leq j < p \leq N} (\mu_j^2 - \mu_p^2)(\beta_j - \beta_p)}{\prod_{j,q=2,\dots,N} (\mu_j^2 + \beta_q)} \\ &= (-1)^{k-1} \frac{\prod_{1 \leq j < q \leq N} (\mu_j^2 - \mu_q^2)}{\prod_{j=1,\dots,N, q=2,\dots,N} (\mu_j^2 + \beta_q)} \prod_{2 \leq j < q \leq N} (\beta_j - \beta_q) \\ &= (-1)^{k-1} \frac{\prod_{1 \leq j < q \leq N} (\mu_j^2 - \mu_q^2)}{\prod_{j,q=1,\dots,N, q \neq k} (\mu_j^2 - b_q^2)} \prod_{1 \leq j < q \leq N, j, q \neq k} (b_q^2 - b_j^2) \\ &= (-1)^{k-1} \frac{\prod_{1 \leq j < q \leq N} (\mu_j^2 - \mu_q^2)}{\prod_{j,q=1,\dots,N, q \neq k} (\mu_j^2 - b_q^2)} \prod_{1 \leq j < q \leq N} (b_q^2 - b_j^2) \\ &= (-1)^{k-1} \frac{\prod_{1 \leq j < q \leq N} (\mu_j^2 - \mu_q^2)(b_q^2 - b_j^2)}{\prod_{j,q=1,\dots,N, q \neq k} (\mu_j^2 - b_q^2)(-1)^{k-1} \prod_{q=1,\dots,N, q \neq k} (b_q^2 - b_k^2)} \\ &= \frac{\prod_{1 \leq j < q \leq N} (\mu_j^2 - \mu_q^2)(b_q^2 - b_j^2)}{\prod_{j,q=1,\dots,N} (\mu_j^2 - b_q^2)} \frac{\prod_{j=1,\dots,N} (\mu_j^2 - b_k^2)}{\prod_{q=1,\dots,N, q \neq k} (b_q^2 - b_k^2)}. \end{aligned}$$

Then, as (13), we have the following expressions for x_k :

$$x_k = \frac{\prod_{q=1,\dots,N} (\mu_q^2 - b_k^2)}{\prod_{q=1,\dots,N, q \neq k} (b_q^2 - b_k^2)}.$$

□

Corollary 11 . *The solution of equation (12) is positive. Namely, $x_k > 0$, for $k = 1, \dots, N$.*

Proof. As

$$0 < b_N < \mu_N, \quad \mu_{i+1} < b_i < \mu_i, \quad \forall i = 1, \dots, N-1,$$

and

$$0 < \mu_N < \mu_{N-1} < \dots < \mu_2 < \mu_1 < \infty,$$

then the statement of Corollary holds.

□

References

- [1] S. Aalto and U. Aesta, “Mean delay analysis of multilevel processor sharing disciplines”, In Proceedings of IEEE INFOCOM 2006.
- [2] K. Avrachenkov, U. Aesta, P. Brown, “Batch Arrival Processor-Sharing with Application to Multi-Level Processor-Sharing Scheduling”, *Queueing systems* 50, pp.459-480, 2005.
- [3] K. Avrachenkov, P. Brown, N. Osipova, “Optimal choice of Threshold in Two Level Processor Sharing”, preprint.
- [4] N. Bansal. “Analysis of the M/G/1 Processor-Sharing Queue with Bulk Arrivals”, *Operation Research Letters*, 31(5), September 2003, pp.401-405.
- [5] Kleinrock, L., R. R. Muntz, and E. Rodemich, “The Processor-Sharing Queueing Model for Time-Shared Systems with Bulk Arrivals,” *Networks Journal*,1,1-13 (1971).
- [6] L. Kleinrock, *Queueing systems, vol. 2*, John Wiley and Sons, 1976.
- [7] G. Fayolle, I. Mitrani, and R. Iasnogorodski, “Sharing a processor among many job classes”, *Journal of the ACM*, 27(3):519-532, 1980.
- [8] H. Feng and V. Misra, “Asymptotic bounds for $M^X/G/1$ processor sharing queues”, Technical report CUCS-00-04, Columbia University, jul.2003
- [9] A. Feldmann, W. Whitt, “Fitting mixtures of exponentials to long-tail distributions to analyze network performance models”, *Performance Evaluation*, v.31, 1998, pp.245-258.
- [10] Kim and Kim, “Concavity of the conditional mean sojourn time in the M/G/1 processor-sharing queue with batch arrivals”, submitted.
- [11] A.G. Kurosh, “Higher algebra”, MIR, 1972.
- [12] K.M. Rege and B. Sengupta, “The M/G/1 Processor-Sharing Queue with Bulk Arrivals”. In Proceedings of Modelling and Evaluation of ATM Networks 1993, pp.417-432.

Contents

1	Introduction	3
2	Batch Arrival PS queue analysis	5
2.1	Main definitions	5
2.2	The expected conditional sojourn time for BPS	5
3	Application to TLPS model	11
3.1	Main definitions	11
3.2	Expected conditional sojourn time in TLPS system	11
3.3	Expected sojourn time for TLPS model	14
4	Appendix	15



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