

Sojourn Times in Multiple-Server Processor Sharing Systems with Priorities

¹R.D. van der Mei^{a,b}, J.L. van den Berg^{a,c}, R. Vranken^a and B.M.M. Gijsen^a

^aKPN Research, Expertise Group Quality of Service Control
P.O. Box 421, 2260 AK Leidschendam, The Netherlands

^bFree University, Mathematics and Computer Science
De Boelelaan 1081, 1081 HV Amsterdam, The Netherlands

^cUniversity of Twente, Faculty of Mathematical Sciences
P.O. Box 217, 7500 AE Enschede, The Netherlands

Abstract

We study the sojourn times in multiple-server Processor Sharing systems with two priority classes and with general service time distributions at both classes. The goal of this paper is to obtain simple, explicit and accurate approximations for the mean sojourn times for customers in each of the priority classes. We give a closed-form expression for the mean sojourn time of high priority customers, based on a classical result on Generalized Processor Sharing models. For the low priority customers we derive a simple closed-form expression for the mean sojourn time in several special cases; for the general case, we develop simple and explicit approximations. Extensive numerical experiments demonstrate that the approximations are accurate for a wide range of parameter settings. As a by-product, we observe the counter-intuitive result that the mean sojourn time of the low-priority customers *decreases* when the variability of the service-time distribution of the low-priority customers increases.

1 Introduction

Processor Sharing (PS) models are widely applicable to situations in which different users receive a share of a scarce common system resource. In particular, over the past few decades, PS models have found many applications in the field of the performance evaluation of computer-communication systems. The standard PS model consists of a single server assigning each customer a fraction $1/n$ of the service rate when there are n customers in the system. Cohen [9] generalizes the PS-model to the so-called Generalized Processor Sharing (GPS) model, where each customer receives a fraction $f(n)$ of the service speed when there are n customers at a node, where $f(\cdot)$ is an arbitrary function (under weak assumptions). The GPS model significantly enhances the modeling capabilities of the PS model. Interestingly, over the past few years the GPS model studied by Cohen in [9] in 1979 has received a renewed interest in the literature on performance of computer-communication networks (cf., e.g., [17], [6], [5], [2], [20]). A particularly attractive feature of GPS models is that in many applications they cover the main factors determining performance, and on the other hand, are still simple enough to be analytically tractable (cf., e.g., [2], [20]). This makes GPS models in many cases favorable to more detailed and complicated models that are only tractable by simulation or cumbersome numerical analysis.

¹Corresponding author: email r.d.vandermei@kpn.com

We consider the sojourn times in Processor Sharing models with multiple servers and with two priority classes (without loss of generality). When the number of high-priority customers does not exceed C , the number of servers, each high priority customer occupies a single server and is served at unit rate. When the number of high-priority customers is larger than C , the system switches to a processor sharing mode and the total service capacity C is equally shared among the high priority customers. The service process of low priority customers proceeds in a similar way, but with two specific restrictions: (1) high priority customers have strict priority (in a preemptive-resume fashion) over low priority customers, and (2) at any moment in time the only servers available to low priority customers are those servers that are not used by high priority customers at that moment.

The motivation for the study of this model is based on two applications in the field of computer-communication networks that, interestingly, lead to the same model. The first application is a file server running on a multi-processor machine in a multi-threaded environment. The file server processes incoming transaction requests, with different priority classes. Each transaction request is handled by a single thread of execution and in many cases includes a significant amount of CPU-intensive server-side processing. When the number of running transactions does not exceed the number of processors, each thread runs on a single processor. However, the effective processing power that each of the active threads receives decreases when the number of threads exceeds the number of processors, because the different threads effectively share the same underlying set of processors. In this example, the transaction times of the file server are represented by the sojourn times in the model discussed above, where the customers represent the transaction requests. We refer to [15, 22, 16] (and references therein) for more details on transaction-server models. The second application can be found in packet-switched networks supporting service differentiation by serving the packets of high quality traffic flows with strict priority over packets of low quality ("best effort") traffic flows in the network nodes (e.g. routers in IP network). Data traffic flows are usually subjected to a network flow control mechanism such as TCP (Transport Control Protocol). When bandwidth on an end-to-end path is temporarily limited the flow control mechanism decreases the transmission rate of each of the elastic flows assigning each flow a share of the available bandwidth. On the other hand, when bandwidth is not a limiting factor, the bandwidth effectively consumed by each elastic flow may be limited by other factors, such as the access line speed (e.g., modem speed). The flow-level performance of elastic traffic can be effectively modeled by the model under consideration (cf., e.g., [2], [20]), where customers represent traffic flows and where the sojourn times represent the download times of flows (i.e. the download time of e.g. file or web page).

In the literature, a variety of papers focus on processor sharing models. For the M/G/1-PS system Yashkov [25], Ott [19] and Van den Berg and Boxma [3] derive (implicit) expressions for the Laplace Stieltjes Transform (LST) of the sojourn time distribution. Van den Berg [4] obtains a simple and fast approximation for the second moment of the sojourn time in the M/G/1-PS queue. Cohen [9] considers so-called Generalized Processor Sharing (GPS) systems, in which the service rate of the customers in the system is an arbitrary function of the number of customers present.² He derives explicit expressions for the distribution of the number of customers in the system, see Section 3 for more details. The reader is

²Note, that Cohen's GPS system is not the same as the in the context of ATM and IP networks well known Generalized Processor Sharing cell/packet scheduling mechanism (also called Weighted Fair Queueing).

referred to Yashkov [26, 27] for overviews of the available results on processor sharing systems. A specific feature of the model studied in the present paper is that the service rate available to low priority customers varies in time due to the fluctuations in the number of high priority customers. In the literature several papers are devoted to processor sharing models with fluctuating service rates. Nunez-Queija [18] considers an M/M/1-PS model with an ON/OFF server, and derives closed-form expressions for several sojourn time statistics. In particular, he also derives closed-form expressions for the limiting sojourn-time distribution under heavy traffic assumptions. Nunez-Queija et al. [17] consider a multiple-server model with two priority classes, where the high priority customers may be blocked when all servers are busy, whereas the low priority customers utilize the remaining service capacity in a processor-sharing fashion. For this model, expressions for the blocking probabilities for the high priority customers, and the sojourn times for the low priority customers are given. Litjens and Boucherie [14] consider an extension of the model in Nunez-Queija et al. [17] assuming that the high priority customers can be buffered, and propose a numerical approach to calculate the performance parameters of interest.

In this paper we study the mean sojourn times in the multiple-server queueing model with processor sharing service discipline and two priority classes described above. The aim of this paper is to obtain *simple, explicit* and *accurate* approximations for the expected sojourn times for customers in each of the priority classes. Thus, simplicity and transparency of the approximations, providing insight in the impact of the system parameters on the mean sojourn times, is of main importance. For the high priority class, we present closed-form expressions for the mean sojourn times in a general parameter setting, which is a special case of known results for the GPS model (cf. [9]). For low priority customers, closed-form expressions are derived for several special cases: the single-server case where the service times of the low-priority customers are exponentially distributed, and the multiple-server case with exponential service times with the same means. In all other cases, exact explicit expressions for the mean sojourn times of the low priority customers cannot be obtained. Therefore, we propose and test a simple and explicit approximation. Numerical results demonstrate that the approximation is accurate for a broad range of parameter settings. As a by-product, we observe the interesting and counter-intuitive result that the mean sojourn times of the low-priority customers tend to decrease when the variability of the service times of the low-priority customers increases.

The remainder of this paper is organized as follows. In section 2 the model under consideration is described in detail. In section 3 we discuss some preliminary results that will be used for the analysis in section 4 and section 5. In section 4 we give exact results for the mean sojourn times for the high-priority customers. In section 5 we derive exact expressions for the mean sojourn times of the low-priority customers in a few special cases, and develop an approximation for the mean sojourn time in the general case. In section 6 the accuracy of the approximation is validated by comparing the approximations with simulation results. Finally, in section 7 we address a number of topics for further research.

2 Model

Consider an M/G/C processor sharing model with two priority classes. The C servers are identical and process requests at unit rate. High priority customers have strict priority (preemptive resume) over low priority customers. High and low priority customers arrive according to independent Poisson processes

with rates λ_H and λ_L , respectively. The service-time requirements of the high and low priority customers are generally distributed with finite first two moments $\beta_H, \beta_H^{(2)}, \beta_L$ and $\beta_L^{(2)}$, respectively. The squared coefficients of variation for high and low priority service times are denoted by c_H^2 and c_L^2 , respectively. The average load of the high and low priority classes is denoted by $\rho_H := \lambda_H \beta_H$ and $\rho_L := \lambda_L \beta_L$, and the total load of the system is denoted by $\rho_{H+L} := \rho_H + \rho_L$. The service process of high priority customers alternates between two modes: a *normal mode* and a *processor sharing mode*. The process is in normal mode when the number of high priority customers does not exceed C . In that case, each high priority customer occupies a single server and is served at unit rate. When the number of high-priority customers is larger than C , the system switches to a processor sharing mode. In that case, the total service capacity C is equally shared among the high priority customers: when there are $n_H \geq C$ high-priority customers in the system, each of these customers is served in a processor sharing fashion with rate C/n_H . Notice that customers are not buffered, and there is no customer blocking. The servers not used by the high priority customers are available for service of the low priority customers. The service process of low priority customers also switches between a normal model and a processor sharing mode. The low priority service process is in normal mode if the total number of customers in the system does not exceed C ; in that case, each customer is served by a single server at unit rate. The low priority service process switches to processor sharing mode when the total number of customers exceeds C : when there are C_L servers available for serving low priority customers and there are $n_L \geq C_L$ low priority customers in the system, then each of these customers is served at rate C_L/n_L . When all servers are occupied by the high priority customers (i.e., $n_H \geq C$), the service of the low priority customers is stopped; their service is continued as soon as the number of high priority customers becomes less than C . Recall that the priority rule is pre-emptive resume. The stability condition of the system is $\rho_{H+L} < C$. Throughout, it is assumed that the system is stable and in steady state. Denote by S_H and S_L the steady state sojourn time of an arbitrary high priority and low priority customer respectively. In this paper, our main focus is on $E[S_H]$ and $E[S_L]$, i.e. the mean sojourn times of both high and low priority customers.

3 Preliminaries

In this subsection we review a key result obtained by Cohen [9] for the Generalized Processor Sharing (GPS) model. In the GPS model, whenever there are i customers present in the system, each customer receives service at a rate $f(i)$, where $f(\cdot)$ is an arbitrary function (under some weak assumptions). Cohen derives the following general result for the joint stationary distribution of the number of customers N in the GPS system and their residual service-time requirements $\underline{T} := (T(1), \dots, T(N))$, cf. formula (7.19) in [9]:

$$Pr[N = n, \underline{T} = \underline{\tau}] = \frac{\frac{\rho^n}{n!} \varphi(n)}{\sum_{k=0}^{\infty} \frac{\rho^k}{k!} \varphi(k)} \prod_{i=1}^n \frac{1 - B(\tau(i))}{\beta}, \quad n = 0, 1, \dots, \quad \tau(i) \geq 0, \quad (1)$$

where $\varphi(0) := 1$ and $\varphi(n) := (\prod_{i=1}^n f(i))^{-1}$, for $n = 1, 2, \dots$, and where $B(\cdot)$ denotes the customers' service requirement distribution, β is the mean service requirement, λ is the customer arrival rate and $\rho := \lambda\beta$. This result will be exploited in the remainder of this paper.

Remark 3.1

We emphasize that formula (1) is a remarkable result with a striking simplicity. First, formula (1) implies

that the stationary distribution of the number of customers in the system is insensitive to the service-time distribution $B(\cdot)$ in the sense that it depends on $B(\cdot)$ only through the first moment. Second, we observe that the residual service requirements of the individual customers in a GPS system are all *identically distributed*, mutually *independent* and independent of the total number of customers present in the system. In particular, this distribution is the so-called excess distribution (often also referred to as the forward recurrence time distribution) of the initial service requirements well known from renewal theory (see, e.g., Chapter 1 of Tijms [23]).

4 High Priority Traffic

From the model description (see section 2) it is clear that the behavior of the high priority customers is not influenced by the presence of the low priority customers. In fact, the stochastic behavior of number of high-priority customers in the system can be effectively modeled as a special case of the GPS model, by taking $f(i) := 1$ if $0 \leq i \leq C$, and $f(i) = C/i$ if $i > C$. Consequently, for the present model we have $\varphi(n) := 1$ if $0 \leq n \leq C$, and $\varphi(n) := n!C^{C-n}/C!$ if $n > C$. Inserting this into (1) and integrating over all values of the residual service requirements $\tau(i)$ one can derive the following explicit expression for the mean number of customers $E[N_{GPS}(\rho)]$ in the GPS system with load ρ : For $0 \leq \rho < C$,

$$E[N_{GPS}(\rho)] = \frac{C^C}{C!A(C, \rho)} \left(\frac{C(\rho/C)^{C+1}}{1 - \rho/C} + \frac{(\rho/C)^{C+1}}{(1 - \rho/C)^2} \right) + \frac{1}{A(C, \rho)} \sum_{n=1}^C \frac{\rho^n}{(n-1)!}, \quad (2)$$

with

$$A(C, \rho) = \frac{C^C}{C!} \frac{(\rho/C)^{C+1}}{1 - \rho/C} + \sum_{n=0}^C \frac{\rho^n}{n!}. \quad (3)$$

Then, application of Little's formula leads to the following expression for the mean sojourn time $E[S_H]$:

$$E[S_H] = \frac{E[N_H]}{\lambda_H} = \frac{E[N_{GPS}(\rho_H)]}{\lambda_H}, \quad (4)$$

where $E[N_{GPS}(\cdot)]$ is given by (2) and (3).

This result for high priority customers is *insensitive* to the service requirement distribution, apart from its first moment, cf. Remark 3.1. It is important to notice, however, that the mean sojourn times of the *low* priority customers are *not* insensitive to the service-time distribution of the high-priority customers (and of their own service time distribution, see section 6).

5 Low Priority Traffic

The analysis of sojourn times for low priority customers is more complicated. The complication is due to the fact that the service rate at which low priority customers are served fluctuates, depending on the variation in the number of high priority customers in the system. An exact mathematical analysis of the mean sojourn times appears to be possible only in a few special cases. In section 5.1 we derive exact expressions for $E[S_H]$ for the multiple-server case with exponentially distributed service times at both priority classes with the same means. In section 5.2 we give exact expressions for $E[S_H]$ for the special case $C = 1$ and where the service times of low-priority customers are exponentially distributed. In section

5.3 we propose simple and fast approximations for $E[S_H]$ for model instances that are not covered by the results in section 5.1 and section 5.2.

5.1 The case with exponential service times with $\beta_H = \beta_L$

It is readily seen that the number of low-priority customers in the system is given by

$$E[N_L] = E[N_{H+L}] - E[N_H], \quad (5)$$

where N_{H+L} stands for the *total* number of customers in the system, and N_L and N_H are the number of low and high priority customers in the system, respectively. An exact expression for $E[N_{H+L}]$ can be obtained by observing that the dynamic behavior of our priority system is stochastically identical to that of the 'corresponding' aggregated GPS system described in Cohen [9], i.e. the GPS system where no distinction is made between the priority classes and where customers with exponentially distributed service requirements (with mean $\beta_H = \beta_L$) arrive according to a Poisson process with rate $\lambda_H + \lambda_L$. To this end, note that the state diagrams of the Markov chains describing the dynamics of $N_{H+L}(t)$ (defined as total the number of customers in the system at time t) are identical. Consequently, $E[N_{H+L}] = E[N_{GPS}(\rho_{H+L})]$. From the analysis for the high priority customers in the previous section we have $E[N_H] = E[N_{GPS}(\rho_H)]$. Finally, from (5) and Little's formula, we obtain the mean sojourn time $E[S_L]$ of the low priority customers:

$$E[S_L] = \frac{E[N_{GPS}(\rho_{H+L})] - E[N_{GPS}(\rho_H)]}{\lambda_L}, \quad (6)$$

where $E[N_{GPS}(\cdot)]$ is given by (2) and (3).

5.2 The case with $C = 1$ and exponential service times for low-priority traffic

The exact analysis of the previous subsection does not apply when the service times of low and high priority customers are not identically exponentially distributed. However, exact results are still available in the single-server case (i.e., $C = 1$) with exponentially distributed service times for low-priority customers by focussing on the amount of unfinished work for both priority classes, rather than the number of customers. Denote by W_H , W_L and W_{H+L} the amount of unfinished work for high-priority customers, low-priority customers, and the total amount of unfinished work in the system, respectively. Then, obviously we have

$$E[W_L] = E[W_{H+L}] - E[W_H]. \quad (7)$$

Since for $C = 1$ the system is work conserving (i.e., runs on full speed whenever there is work in the system), we have

$$E[W_{H+L}] = E[W_{M/G/1}], \quad (8)$$

where $W_{M/G/1}$ stands for the total amount of unfinished work in the $M/G/1$ (PS or FCFS) system without priorities, with arrival rate $\lambda_H + \lambda_L$ and where the first two moments of the service-time distribution are given by

$$\beta_{M/G/1} := \frac{\lambda_H}{\lambda_H + \lambda_L} \beta_H + \frac{\lambda_L}{\lambda_H + \lambda_L} \beta_L, \quad \text{and} \quad \beta_{M/G/1}^{(2)} := \frac{\lambda_H}{\lambda_H + \lambda_L} \beta_H^{(2)} + \frac{\lambda_L}{\lambda_H + \lambda_L} 2\beta_L^2, \quad (9)$$

respectively. The classical Pollaczek-Khintchine formula for the M/G/1-queue implies that

$$E[W_{M/G/1}] = \frac{\rho_{H+L}}{1 - \rho_{H+L}} \frac{\beta_{M/G/1}^{(2)}}{2\beta_{M/G/1}}. \quad (10)$$

Then after several straightforward manipulations we obtain the following expression:

$$E[W_L] = E[W_{H+L}] - E[W_H] = \frac{\rho_{H+L}}{2(1 - \rho_{H+L})} \left(\lambda_H \beta_H^{(2)} + 2\lambda_L \beta_L^2 \right) - \frac{\rho_H \lambda_H \beta_H^{(2)}}{2(1 - \rho_H)}. \quad (11)$$

To obtain an expression for $E[S_L]$, we need two additional arguments. First, using the memoryless property of the exponential distribution, the mean amount of unfinished low priority work $E[W_L]$ can easily be related to the mean number of low priority customers $E[N_L]$ in the system:

$$E[N_L] = \frac{E[W_L]}{\beta_L}, \quad (12)$$

and second, from Little's formula we obtain

$$E[S_L] = \frac{E[N_L]}{\lambda_L}. \quad (13)$$

Finally, combining (11)-(13) we obtain the following expression after several straightforward algebraic manipulations:

$$E[S_L] = \frac{\beta_L}{(1 - \rho_{H+L})} + \frac{\lambda_H \beta_H^{(2)}}{2(1 - \rho_{H+L})(1 - \rho_H)}. \quad (14)$$

Remark 5.1

It is easy to verify that this result coincides with the mean sojourn time result obtained by Nunez-Queija [18] for the M/M/1-PS model with an ON/OFF server. To this end, take the ON and OFF periods in Nunez-Queija's model equal to the idle and busy periods of the high priority customers in our model. More precisely, the idle periods are exponentially distributed with mean $1/\lambda_H$, and the first two moments of the M/G/1 busy periods are given by $m_1 := \beta_H/(1 - \rho_H)$ and $m_2 := 2\beta_H^{(2)}/(1 - \rho_H)^3$, respectively.

5.3 Approximation for the general case

In the general case exact explicit expressions for the mean sojourn times of the low-priority customer can not be obtained. Therefore, in this section we develop an approximation for $E[S_H]$. We reemphasize that an important requirement of the approximation is that it should be simple and explicit, providing insight in the relation between the response time performance and the system parameters. This is preferred to numerical approximation techniques that require the solution of sets of linear equations.

To develop an approximation for $E[S_L]$, we adopt the derivation for the case of $C = 1$ in the previous section. So our starting point is: For $C > 1$,

$$E[W_H] + E[W_L] = E[W_{L+H}] \approx E[W_{GPS}], \quad (15)$$

where W_{GPS} stands for the total amount of unfinished work in the system *without priorities*; that is, the system (throughout referred to as the GPS system) with arrival rate $\lambda_H + \lambda_L$, and where the service-time distribution is a mixture of the service-time distribution of high- and low-priority customers, respectively.

It is readily seen that the first two moments of the services-time distribution in the GPS system are given by the following expressions (cf. (9)):

$$\beta_{GPS} := \frac{\lambda_H}{\lambda_H + \lambda_L} \beta_H + \frac{\lambda_L}{\lambda_H + \lambda_L} \beta_L, \quad \text{and} \quad \beta_{GPS}^{(2)} := \frac{\lambda_H}{\lambda_H + \lambda_L} \beta_H^{(2)} + \frac{\lambda_L}{\lambda_H + \lambda_L} \beta_L^{(2)}. \quad (16)$$

Recall from section 3 that the total amount of unfinished work in the GPS system can be obtained from Cohen's general result for the joint distribution of number of customers and their residual service requirements (see also Remark 3.1):

$$E[W_{GPS}] = \frac{\beta_{GPS}^{(2)}}{2\beta_{GPS}} E[N_{GPS}(\rho_{H+L})], \quad (17)$$

where β_{GPS} and $\beta_{GPS}^{(2)}$ are defined in (16). Similarly, for the high-priority customers we have

$$E[W_H] = \frac{\beta_H^{(2)}}{2\beta_H} E[N_{GPS}(\rho_H)]. \quad (18)$$

Now, applying the approximation step (15), we obtain the following estimation for the mean amount of unfinished work $E[W_L]$ belonging to the low priority customers in the system:

$$E[W_L] \approx E[W_{GPS}] - E[W_H] = \frac{\beta_{GPS}^{(2)}}{2\beta_{GPS}} E[N_{GPS}(\rho_{H+L})] - \frac{\beta_H^{(2)}}{2\beta_H} E[N_{GPS}(\rho_H)]. \quad (19)$$

Recall from Remark 3.1 and the analysis in section 4 that equation (1) implies that for high-priority customers the amount of unfinished work is the independent sum of the remaining service times of all high-priority customers in the system. We adopt this idea to low-priority customers to obtain the following approximate relation between mean the number of low-priority customers $E[N_L]$ and corresponding mean amount of unfinished work $E[W_L]$:

$$E[N_L] \approx \frac{E[W_L]}{\beta_L^{(2)}/2\beta_L}. \quad (20)$$

Note that this relation is generally not exact, unless the service times for low priority customers are exponentially distributed (in this case equation (12) holds). Then, from Little's formula we obtain

$$E[S_L] = \frac{E[N_L]}{\lambda_L}. \quad (21)$$

Finally, combining (19)-(21) we obtain the following approximation for $E[S_L]$:

$$E_{app}[S_L] := \frac{2\beta_L}{\lambda_L \beta_L^{(2)}} \left(\frac{\beta_{GPS}^{(2)}}{2\beta_{GPS}} E[N_{GPS}(\rho_{H+L})] - \frac{\beta_H^{(2)}}{2\beta_H} E[N_{GPS}(\rho_H)] \right). \quad (22)$$

Remark 5.2

By combining equations (16) and (2), it is readily verified that $E_{app}[S_L]$ in (22) is a *decreasing* function of $\beta_L^{(2)}$. In other words, the mean sojourn time of the low-priority customers decreases when the variability of the service times of the high-priority customers increases. In the next section we demonstrate that this *counter-intuitive* monotonicity property for the approximated sojourn times is supported by simulation results. We emphasize that this observation provides new fundamental insight in the performance of the model under consideration, which also addresses the importance of the development of simple and explicit approximations, in favor of heavy-weight numerical "exact" solution techniques.

6 Numerical Results

To assess the accuracy of the approximation discussed in the previous section, we have performed numerous numerical experiments by comparing the approximation with simulation results. The results of this comparison are outlined below.

Exponential service times

Let us first consider the situation where the service times of the high-priority and low-priority customers are exponentially distributed. We have calculated the exact and approximated values of the mean sojourn times for the low priority customers for different values of the load per server (i.e., ρ_{H+L}/C). The "exact" values, denoted by $E_{exact}[S_L]$, have been obtained via simulations. The approximations, denoted by $E_{app}[S_L]$, have been calculated from (22). The relative error of the approximations is defined as follows:

$$\Delta\% := 100\% * \left(\frac{E_{approx}[S_L] - E_{exact}[S_L]}{E_{exact}[S_L]} \right). \quad (23)$$

To assess the accuracy of the approximations, let us call the approximation "extremely good" if the absolute error is less than 2%, "very good" for error range 2-5%, "good" for 5-10% and "fair" for 10-20%. Table 1 shows the expected sojourn times ("exact") and their approximations ("app") for the low priority customers as a function of the load per server (i.e., ρ_{H+L}/C), indicated as "load", for varying numbers of servers. The mean service times of the high and low priority customers are $\beta_H = 1/4$ and $\beta_L = 1$, respectively; the arrival rate λ_L has been chosen such that $\rho_L/C = 0.50$ in all cases.

$\beta_H = 0.25, c_H^2 = 1, \beta_L = 1, c_L^2 = 1, \rho_L/C = 0.50$									
	$C = 2$			$C = 4$			$C = 10$		
load	exact	app	$\Delta\%$	exact	app	$\Delta\%$	exact	app	$\Delta\%$
0.60	1.60	1.59	-0.3	1.19	1.19	-0.3	1.03	1.03	-0.0
0.70	2.07	2.05	-0.7	1.41	1.39	-0.9	1.09	1.08	-0.4
0.80	3.06	3.03	-0.9	1.88	1.86	-1.4	1.25	1.24	-1.1
0.90	6.12	6.08	-0.7	3.40	3.36	-1.4	1.83	1.80	-1.4
0.95	12.38	12.28	-0.8	6.50	6.45	-0.8	3.06	3.02	-1.5

Table 1: Exact and approximated mean sojourn times for low priority traffic.

Table 2 below shows the results for the model in Table 1, but with $\beta_H = 4$ (instead of 0.25).

$\beta_H = 4, c_H^2 = 1, \beta_L = 1, c_L^2 = 1, \rho_L/C = 0.5$									
	$C = 2$			$C = 4$			$C = 10$		
load	exact	app ₁	$\Delta\%$	exact	app ₁	$\Delta\%$	exact	app ₁	$\Delta\%$
0.60	1.94	2.00	+3.3	1.28	1.32	+3.5	1.04	1.05	+1.0
0.70	3.31	3.43	+3.8	1.81	1.92	+6.1	1.15	1.19	+3.5
0.80	6.61	6.81	+3.6	3.31	3.50	+6.0	1.58	1.70	+7.0
0.90	18.06	18.30	+1.3	8.88	9.15	+3.1	3.60	3.80	+5.8
0.95	42.53	42.67	+0.3	20.98	21.29	+1.5	8.42	8.58	+1.9

Table 2: Exact and approximated mean sojourn times for low priority traffic.

The results presented in Tables 1 and 2 lead to a number of observations. First, we observe that the accuracy of the approximations ranges from "good" to "extremely good" (as defined above). The relative

error is no more than 6% for the whole range of values for the load per server, and for a broad range of values of C , the number of servers. We also observe that the approximation results for the case $\beta_H > \beta_L$ are consistently less accurate than in the case $\beta_H < \beta_L$, but still good to very good in most cases. Second, the approximation results for the case $\beta_H < \beta_L$ consistently tend to *underestimate* the exact (simulated) value of $E[S_L]$, whereas in the case $\beta_H > \beta_L$ the approximations lead to overestimations of $E[S_L]$. To give an intuitive explanation for this observation, notice that since in this example the service times of the low-priority customers are exponentially distributed, the only source of the inaccuracy in the approximation (22) is the approximation of the total mean amount of work in the system (15). Let us call the "aggregated system" the system without priorities and with arrival rate $\lambda_H + \lambda_L$ and hyper-exponentially distributed service times with the first two moments given by (16). The "priority system" is our model as described in section 2. Then in the case $\beta_H \gg \beta_L$ in the aggregated system the customers with relatively large service-time requirements (these are the high-priority customers in the priority system) tend to receive a lower service rate than they would receive in the priority system. Consequently, the total amount of unfinished work in the aggregated system tends to be due to a relatively small number of large ('high priority') customers, while in the priority system the unfinished work tend to be due to a large number of small ('low priority') customers. This is 'disadvantageous' for the aggregate system as it will be more often in the situation that not all servers are used. Consequently, the total mean amount of unfinished work in the aggregated system is expected to be larger than in the priority system, which supports the results shown in Table 2. A similar argument can be used to support the observed underestimation of $E[S_L]$ in Table 1 for the case $\beta_H \ll \beta_L$. A third observation is that the absolute error considered as a function of the load per server tends to increase up to some maximum value (around load-per-server values of 80-90%) and tends to decrease when the system is close to saturation. In this context, note that in light traffic (when the arrival rates tends to 0) the approximation is asymptotically exact, since in the limiting case the sojourn-time distribution (and hence the mean) converges to the service-time distribution. Also, one may suspect that the approximation for the case on exponential service times is asymptotically exact when the load per server tends to unity: As long as all servers are occupied, the work load processes in the priority system and in the (approximate) aggregated system behave in exactly the same way. A rigorous proof of such an asymptotic results is beyond the scope of this paper.

Non-exponential service times

To assess the accuracy of the approximation for the case of non-exponential service times, we consider the case that the service times of both the high- and low-priority customers are hyper-exponentially distributed (with balanced means [23]), for various combinations of the squared coefficients of variation c_H^2 and c_L^2 . The mean service times are fixed: $\beta_H = 1/4$ and $\beta_L = 1$; the load due to low priority customers is also fixed: $\rho_L/C = 0.5$. The total load (per server) varies from 0.70 to 0.95. Table 3 shows the exact and approximated results and the relative errors for the case of 4 servers; Table 4 shows the results for the 20-server case.

$C = 4, \beta_H = 1/4, \beta_L = 1, \rho_L/C = 0.50$													
		load = 0.70			load = 0.80			load = 0.90			load = 0.95		
c_H^2	c_L^2	exact	app	$\Delta\%$	exact	app	$\Delta\%$	exact	app	$\Delta\%$	exact	app	$\Delta\%$
0	0	1.40	1.39	-0.3	1.86	1.86	-0.4	3.37	3.36	-0.5	6.48	6.45	-0.5
0	1	1.39	1.37	-1.0	1.83	1.80	-1.6	3.22	3.16	-1.6	6.02	5.95	-1.1
0	4	1.39	1.36	-1.6	1.81	1.77	-2.5	3.13	3.05	-2.8	5.87	5.66	-3.6
0	16	1.39	1.36	-2.0	1.81	1.75	-2.9	3.06	2.99	-2.1	5.90	5.52	-6.6
1	0	1.43	1.43	+0.2	1.95	1.97	+0.7	3.71	3.74	+0.8	7.40	7.44	+0.5
1	1	1.41	1.39	-0.9	1.88	1.86	-1.4	3.40	3.36	-1.3	6.50	6.45	-0.8
1	4	1.40	1.37	-2.0	1.85	1.79	-3.1	3.23	3.12	-3.4	6.10	5.86	-4.1
1	16	1.40	1.36	-2.5	1.83	1.76	-4.0	3.13	3.01	-3.6	6.01	5.57	-7.3
4	0	1.50	1.53	+2.6	2.20	2.29	+4.3	4.70	4.90	+4.2	10.07	10.41	+3.3
4	1	1.45	1.45	-0.2	2.02	2.02	-0.1	3.94	3.94	-0.2	7.87	7.93	-0.8
4	4	1.43	1.39	-2.5	1.94	1.86	-4.2	3.54	3.36	-5.3	6.73	6.45	-4.3
4	16	1.42	1.37	-3.5	1.89	1.78	-6.0	3.34	3.08	-7.6	6.30	5.75	-8.7
16	0	1.72	1.96	+14.1	3.09	3.61	+17.0	8.44	9.54	+13.0	20.35	22.28	+9.5
16	1	1.58	1.66	+4.7	2.52	2.68	+6.4	5.93	6.25	+5.5	13.12	13.87	+5.7
16	4	1.51	1.48	-2.0	2.22	2.12	-4.6	4.57	4.28	-6.2	9.26	8.82	-4.7
16	16	1.46	1.39	-4.9	2.06	1.86	-9.9	3.91	3.36	-14.2	7.44	6.45	-13.4

Table 3: Exact and approximated mean sojourn times for low priority traffic for different values of c_H^2, c_L^2 and the load per server ($C = 4$).

$C = 20, \beta_H = 1/4, \beta_L = 1, \rho_L/C = 0.50$													
		load = 0.70			load = 0.80			load = 0.90			load = 0.95		
c_H^2	c_L^2	exact	app	$\Delta\%$	exact	app	$\Delta\%$	exact	app	$\Delta\%$	exact	app	$\Delta\%$
0	0	1.02	1.02	-0.1	1.08	1.07	-0.4	1.34	1.31	-1.0	1.95	1.93	-1.2
0	1	1.02	1.02	-0.1	1.08	1.07	-0.6	1.32	1.30	-1.4	1.87	1.84	-1.7
0	4	1.02	1.02	-0.1	1.07	1.07	-0.8	1.31	1.29	-2.0	1.84	1.79	-2.8
0	16	1.02	1.02	-0.1	1.07	1.06	-0.9	1.31	1.28	-2.1	1.86	1.77	-5.2
1	0	1.02	1.02	-0.1	1.08	1.08	-0.1	1.39	1.39	-0.1	2.10	2.10	-0.3
1	1	1.02	1.02	-0.1	1.08	1.07	-0.6	1.35	1.33	-1.3	1.95	1.93	-1.4
1	4	1.02	1.02	-0.1	1.08	1.07	-0.9	1.33	1.30	-2.3	1.89	1.82	-3.3
1	16	1.02	1.02	-0.1	1.08	1.07	-1.0	1.32	1.28	-2.8	1.89	1.78	-5.9
4	0	1.02	1.02	+0.3	1.10	1.11	+1.2	1.51	1.55	+2.8	2.54	2.61	+2.7
4	1	1.02	1.02	-0.0	1.09	1.09	-0.1	1.42	1.41	-0.1	2.18	2.18	-0.1
4	4	1.02	1.02	-0.1	1.08	1.07	-0.9	1.37	1.33	-2.6	2.00	1.93	-3.7
4	16	1.02	1.02	-0.0	1.08	1.07	-1.3	1.34	1.29	-3.6	1.94	1.81	-7.0
16	0	1.02	1.04	+1.8	1.13	1.23	+8.2	1.91	2.21	+15.8	4.11	4.64	+12.9
16	1	1.02	1.03	+0.7	1.11	1.15	+3.0	1.64	1.74	+6.6	3.01	3.20	+6.2
16	4	1.02	1.02	+0.1	1.10	1.10	-0.1	1.47	1.46	-0.8	2.37	2.33	-1.6
16	16	1.02	1.02	-0.1	1.09	1.07	-1.3	1.39	1.33	-4.4	2.09	1.93	-7.8

Table 4: Exact and approximated mean sojourn times for low priority traffic

for different values of c_H^2 , c_L^2 and the load per server ($C = 20$).

First, we observe that accuracy of the approximations is good to extremely good in the majority of the cases considered in Tables 3 and 4, for both a small ($C = 4$) and a large ($C = 20$) number of servers. Second, the results in Tables 3 and 4 consistently support the counter-intuitive observation that the mean sojourn times for low-priority customers *decreases* when the variability of the service times of the low-priority customers increases. In this context, we reemphasize that the approximation (22) also possesses this monotonicity property (see Remark 5.2). This observation is quite intriguing, because in PS systems without priorities the mean sojourn times are independent of the service requirement variability (cf. Section 3) and in an M/G/1 queue with FIFO service discipline the mean service time is an increasing function of the service time variance! Apparently, in the present PS system the fluctuation of the service speed (due to the presence of high priority customers) is less disadvantageous for very small and very large low priority customers (small customers tend to "slip through"; large customers experience the mean service speed) than for average size customers.

Weaknesses of the approximation

Each approximation almost by definition has "weak spots", i.e., combinations of parameter values for which the accuracy of the approximations tends to degrade. First, we observe from Tables 3 and 4 that the worst-case approximations are consistently found when c_H^2 is large (i.e., when the variability of the service-time distributions of the high-priority customers is high). Second, the accuracy of the approximations depends strongly on the asymmetry in the mean service time. To illustrate this, consider the model in Table 1 with $C = 4$. We have calculated the exact and approximated values of $E[S_L]$ as a function of the total load per server, for $\beta_H/\beta_L = 0.25, 1.00$ and 4.00 . The squared coefficients of variation are the same for the high and low priority customers: $c_H^2 = c_L^2 = \gamma$, where γ is varied as 0, 4 and 16. The cases $\gamma = 0, 4$ and 16 have been taken to represent service-time distributions that are constant, "fairly variable" and "highly variable", respectively. In the case $\gamma = 0$ the service times are deterministic, and in the cases $\gamma = 4$ and $\gamma = 16$, the service times for both priority classes are hyper-exponentially distributed, with balanced means (cf. [23]). Notice that in all cases considered here we have $c_H^2 = c_L^2$, so that the approximations are the same for all values of γ . Figures 1, 2 and 3 below show the results.

The results in Figures 1, 2 and 3 demonstrate that in all cases the accuracy of the approximation is "extremely good" to "good" when $\beta_H \leq \beta_L$. In Figures 1 and 2, the worst-case relative error was found to be no more than 11%, which is still considered fairly good. However, Figure 3 demonstrates the fact that the accuracy tends to degrade significantly when $\beta_H \gg \beta_L$. This phenomenon is due to the fact that when $\beta_H \gg \beta_L$ the approximation step in (15) becomes less accurate. An intuitive explanation for this observation is given above in the discussion of the results presented in Tables 1 and 2. In this context, it is important to notice that this situation is of minor practical relevance, because in practice usually $\beta_H \ll \beta_L$, i.e., the service requirements of high priority transactions are generally small compared to the service requirements of low priority transactions. For example, in computer-communication systems, the heavy-weight transactions (e.g., document transfers) are typically lower priority than light-weight transactions in real-time systems.

To summarize, the numerical results demonstrate that the approximations satisfies our goals of being simple, explicit and accurate. In fact, the accuracy of the approximations can be classified as good to extremely good in most cases. The "weak spots" in the approximations are consistently found to be the

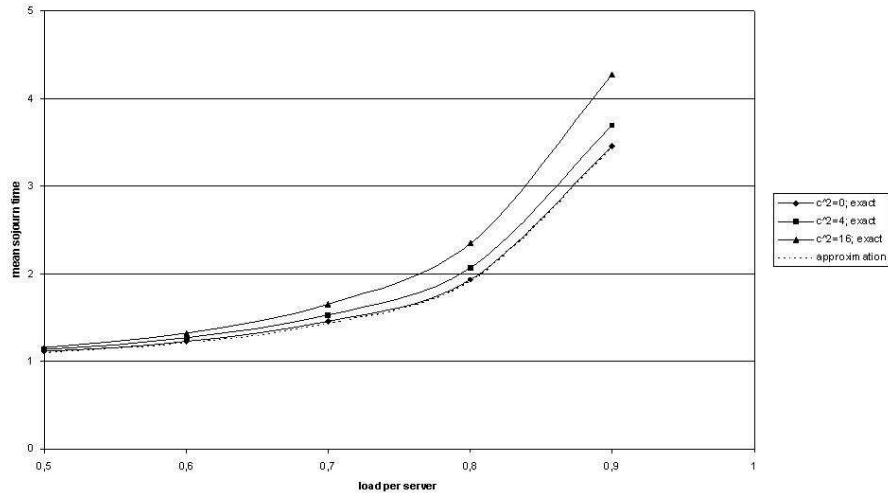


Figure 1: Exact and approximated mean sojourn times for low priority traffic for different values of the load per server ($\beta_H/\beta_L = 0.25$).

cases in which the following conditions are met: (1) the mean service time for the high priority customers is large compared to the mean service times of the low priority customers (i.e., $\beta_H \gg \beta_L$) and (2) the service-time distributions of high-priority customers are highly variable.

7 Topics for Further Research

The results presented in this paper lead to a number of interesting topics for further research. First, each approximation method (almost by definition) has a parameter-value range for which the approximation becomes less accurate. The numerical results in section 6 have demonstrated that the worst-case approximations are consistently found when $\beta_H \gg \beta_L$ and c_H^2 is high. Refinement of each of the two approximation steps (15) and (20) is a challenging topic for further research. Second, in the present paper it is assumed that the arrival processes are independent homogeneous Poisson arrival processes. However, in many applications the arrival processes are correlated, so that the Poisson assumption is not realistic. For instance, the arrival process of transaction requests at Web servers is characterized by successive burst of arrivals. To capture the impact of correlated arrivals on the processing times of servers with correlated arrivals, the Poisson assumption needs to be relaxed. Analysis of the model with non-Poisson type of arrival processes is topic for further research Third, in the present paper it is assumed that the customers are served in a processor sharing fashion. This assumption is motivated by the modeling of transaction servers that are CPU-bound. Examples are multi-threaded HTTP Web servers with significant server-side scripting (cf. [2]). In many applications, however, the CPU is not necessarily

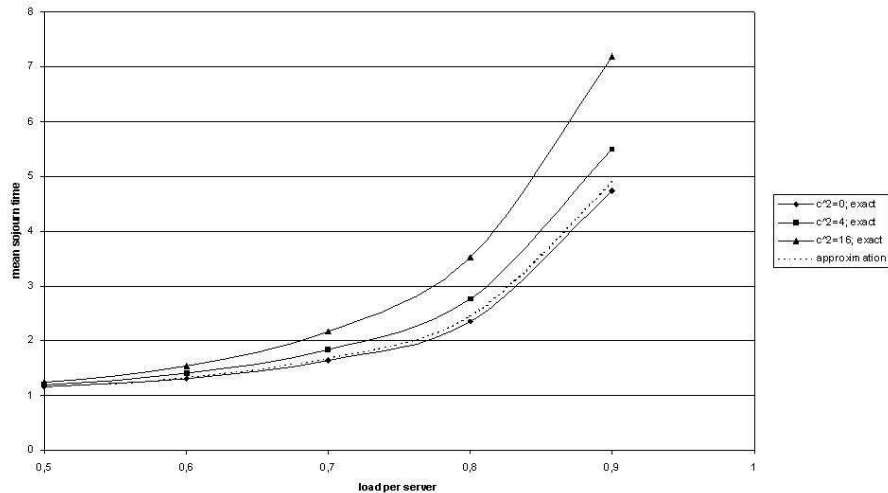


Figure 2: Exact and approximated mean sojourn times for low priority traffic for different values of the load per server ($\beta_H/\beta_L = 1.00$).

the limiting factor. For example, for file servers that primarily handle plain text files the disk I/O speed is more likely to be the performance bottleneck. This type of applications leads to FIFO-type, rather than PS-type queueing models. Analysis of multi-server FIFO queueing models with multiple priority classes is a challenging topic for further research. Fourth, with the advent of on-line services Web servers are being used as front-end servers in many distributed multi-tiered system architecture. To speed up performance these Web servers are typically implemented to support multi-threading, which in many cases leads to synchronization and locking issues in situations where different threads of execution need to access shared data, which may have a significant impact on the performance of the Web servers and the end-to-end performance of the system. Inclusion of the impact of synchronization among the different threads is an interesting topic for further research. Finally, the present model can be used to describe the flow-level characteristics of the Transport Control Protocol (TCP), supporting the majority of Internet traffic. Currently, the Internet is migrating from a best-effort networking technology to a QoS-enabled network supporting QoS differentiation. Flow-level models have been shown to be highly effective for dimensioning Internet access lines connecting end users to Internet Service Providers (ISPs), see [2, 20]. In this context, the present model and the proposed approximations can be used to quantify the potential for QoS differentiation at the TCP layer and for the dimensioning of ISP access trunks. To this end, it is a challenging topic for further research to validate the flow-level performance model (with priorities) by comparing the performance predictions based on the model with real TCP performance data.

Acknowledgments: The authors are indebted to S.C. Borst, R. Boucherie, O.J. Boxma, R. Nunez-

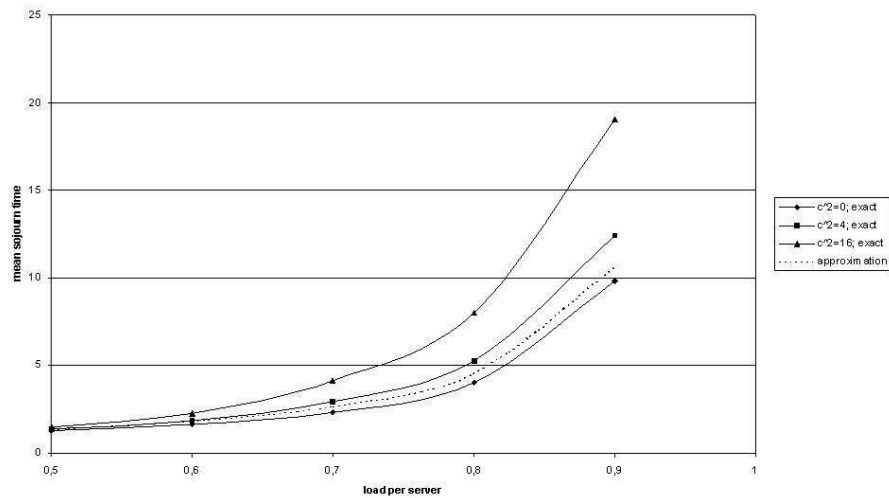


Figure 3: Exact and approximated mean sojourn times for low priority traffic for different values of the load per server ($\beta_H/\beta_L = 4.00$).

Queija and J.A.C. Resing for their contributions to this project.

References

- [1] P. Barford and M. Crovella (1999). A performance evaluation of hyper text transfer protocols. Proceedings of the *ACM Sigmetrics International Conference on Measurement and Modeling of Computer Systems*, 188-197.
- [2] J.V.L. Beckers, I. Hendrawan, R.E. Kooij and R.D. van der Mei (2001). Generalized Processor Sharing models for Internet access lines. In: *Proc. 9th IFIP conference on Performance Modeling and Evaluation of ATM & IP Networks* (Budapest, June), 101-112.
- [3] J.L. van den Berg and O.J. Boxma (1991). The M/G/1 queue with processor sharing and its relation to a feedback queue. *Queueing Systems* 9, 365-401.
- [4] J.L. van den Berg (1990). *Sojourn times in Feedback and Processor Sharing Queues*. Ph.D. Thesis, University of Utrecht.
- [5] J.L. van den Berg, R.D. van der Mei, B.M.M. Gijsen, M.J. Pikaart and R. Vranken (2001). Processing times for transaction servers with Quality of Service differentiation. In: *Proc. international conference on Performance Modeling and Evaluation of Communication Systems* (Aachen, September), 241-252.

- [6] T. Bonald en J.W. Roberts (2000). Performance of bandwidth sharing mechanisms for service differentiation in the Internet. In: *Proc. ITC Specialists Seminar on IP Measurement, Modeling and Management* (Monterey, September), 22-1 - 22-10.
- [7] G.L. Choudhury and D. Houck (1994). Combined queueing an activity network based modeling of sojourn time distributions in distributed communication systems. In: *Proceedings of ITC-14* (eds. J. Labetoulle and J.W. Roberts), 525-534.
- [8] M.E. Crovella, R. Frangioso and M. Harchol-Bacher (1999). Connection scheduling in Web servers. *Proceedings of the USENIX Symposium in Internet Technologies and Systems*.
- [9] J.W. Cohen (1979). The multiple phase service network with generalized processor sharing. *Acta Informatica* 12, 245-284.
- [10] J. Dille, R. Friedrich, T. Jin and J. Rolia (1998). Web server performance measurements and modeling techniques. *Performance Evaluation* 33, 5-26.
- [11] W.K. Ehrlich, R. Hariharan, P.K. Reeser and R.D. van der Mei (2001). Performance of Web servers in a distributed computing environment. To appear in the proceedings of the *17th International Teletraffic Congress* (Salvador, Brazil).
- [12] J. Heidemann, K. Obraczka and J. Touch (1997). Modeling the performance of HTTP over several transport protocols. *IEEE Transaction on Networking* 5, 616-630.
- [13] L. Kleinrock (1976). *Queueing Systems, Vol. II*. Wiley, New York.
- [14] R.M. Litjens and R. Boucherie (2000). Radio resource sharing in an GSM/GPRS network. In: *Proceedings ITC Specialists Seminar on Mobile Systems and Mobility* (ed. P.J. Emstad), Lillehammer, Norway, 261-274.
- [15] R.D. van der Mei, R. Hariharan and P.K. Reeser (2001). Web server performance modeling. *Telecommunications Systems* 16, 361-378.
- [16] R.D. van der Mei, W.K. Ehrlich, P.K. Reeser and J.P. Francisco (2000). A decision support system for tuning Web servers in distributed object-oriented network architectures. *ACM Performance Evaluation Review* 27, 57-62.
- [17] R. Nunez-Queija, J.L. van den Berg and M. Mandjes (1999). Performance evaluation of strategies of elastic and stream traffic. In: *Teletraffic Engineering in a Competitive World* (eds. P. Key and D. Smith), 1039-1050.
- [18] R. Nunez-Queija (2000). Sojourn times in a processor sharing queue with service interruptions. *Queueing Systems* 34, 351-386.
- [19] T.J. Ott (1984). The sojourn-time distribution in the M/G/1 queue with processor sharing. *J. Appl. Prob.* 21, 360-378.
- [20] A. Riedl, T. Bauschert, M. Perske and A. Probst (2000). Investigation of the M/G/R processor sharing system for dimensioning IP access networks with elastic traffic. In: *Proc. 1st Polish-German Symposium of Telecommunication Systems*.

- [21] L.P. Slothouber. A Model of Web Server Performance.
<http://louvx.biap.com/whitepapers/performance/overview/>.
- [22] P.K. Reeser, R.D. van der Mei and R. Hariharan (1999). An Analytic Model of a Web Server. In: *Teletraffic Engineering in a Competitive World*, eds. P. Key and D. Smith (Elsevier, Amsterdam), 1199-1208.
- [23] H.C. Tijms (1986). *Stochastic Modelling and Analysis*. Wiley, New York.
- [24] W. Whitt (1983). The queueing network analyzer. *The Bell Systems Technical Journal* 62, 2779-2812.
- [25] Yashkov, S.F. (1983). A derivation of response time distribution for a M/G/1 processor sharing queue, *Problems Contr. & Info. Theory* 12, 133-148.
- [26] Yashkov, S.F. (1992). Mathematical problems in the theory of processor-sharing queueing systems. *Journal of Soviet Mathematics* 58, 101-147.
- [27] Yashkov, S.F. (1987). Processor-sharing queues: some progress in analysis. *Queueing Systems* 2, 1-17.