Reducing the complexity of the performance analysis of a multi-server facilities

Tulin ATMACA, Thomas BEGIN, Alexandre BRANDWAJN, Hind CASTEL

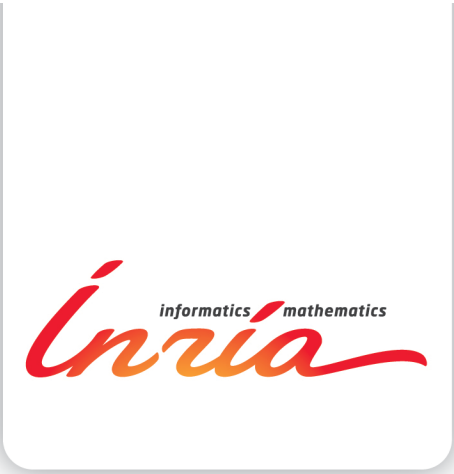
RESEARCH REPORT

N° 8617

21/10/2014

Project-Team DANTE

ISSN 0249-6399



Reducing the complexity of the performance analysis of a multi-server facilities

Tulin Atmaca[1], Thomas Begin[2], Alexandre Brandwajn[3], and Hind Castel[1]

Project-Team DANTE

Research Report N° 8617 — 21/10/2014 —12 pages.

Abstract: Systems with multiple servers are common in many areas and their correct dimensioning is in general a difficult problem under realistic assumptions on the pattern of user arrivals and service time distribution. We present an approximate solution for the underlying *Ph/Ph/c/N* queueing model. Our approximation decomposes the solution of the *Ph/Ph/c/N* queue into solutions of simpler *M/Ph/c/N* and *Ph/M/c/N* queues. It is conceptually simple, easy to implement and produces generally accurate results for the mean number in the system, as well as the loss probability. A significant speed advantage compared to the numerical solution of the full *Ph/Ph/c/N* queue can be gained as the number of phases representing the arrival process and/or the number of servers increases.

Key-words: Multi-agent systems, *Ph/Ph/c/N* queue, Approximate solution, Fixed-point iteration, Loss probability.

[1] Institut Telecom, Telecom SudParis, Evry, France

[2] Université Lyon 1 / LIP (UMR INRIA, ENS Lyon CNRS, UCBL), Lyon , France

[3] University of California Santa Cruz, Baskin School of Engineering, USA

RESEARCH CENTRE
GRENOBLE - RHÔNE-ALPES

Inovallée
655 avenue de l'Europe - Montbonnot
38334 Saint Ismier Cedex France

Réduire la complexité de l'analyse des performances pour les systèmes multi-serveurs

Résumé : Les systèmes avec serveurs multiples sont fréquents dans de nombreux domaines et leur dimensionnement est en général un problème difficile lorsqu'on prend en compte des hypothèses réalistes sur la forme des arrivées des clients et de la distribution du temps de service. Nous présentons une solution approchée pour la file associée Ph/Ph/c/N. Notre approximation recherche la solution de la file Ph/Ph/c/N en considérant les solutions de deux files plus simples que sont les files M/Ph/c/N et Ph/M/c/N. L'approximation est conceptuellement simple, facile à programmer et produit en général des résultats précis pour le nombre moyen dans le système, ainsi que pour la probabilité de perte. Un avantage significatif en vitesse en comparaison de la solution numérique de la file Ph/Ph/c/N peut être obtenu lorsque le nombre de phases représentant le processus d'arrivée et/ou le nombre de serveurs s'accroît.

Mots clés : Systèmes multi-agents, file *Ph/Ph/c/N*, Solution approchée, Itération point fixe, Probabilité de perte.

1 Introduction

Systems with many (dozens or hundreds) agents (or servers) such as call centers [GAN03] are a reality in many areas of our everyday life. Their correct dimensioning so as to achieve an acceptable performance while minimizing their cost is not a trivial problem. In particular, questions such as the amount of improvement (or, conversely, degradation) in the expected waiting time of a user, or even the ability of the user to join the queue, as one adds (or removes) servers may not be easily answerable. Indeed, under realistic assumptions on the pattern of user arrivals and the distribution of the service time of a user (e.g. high variability and long-term dependencies), it is not possible to obtain acceptable results using simple *M*/*M*/*c*/*N* or Erlang queueing models (c is the number of servers and N is the buffer size, i.e. the maximum number of users that can be present in the system). Unfortunately, the more appropriate *G*/*G*/*c*/*N* model does not possess in general a known analytical solution. Therefore, a common approach (besides simulation) is to represent the "general" distributions by their phase-type equivalents and solve the resulting *Ph*/*Ph*/*c*/*N* queueing system numerically.

As long as the number of servers c and the number of phases in the model remain moderate, the balance equations of the *Ph*/*Ph*/*c*/*N* queue can be solved via direct iteration [TAK76, SEE86, BRA09] or, more efficiently and elegantly, via matrix-geometric methods [RAM86, LAT93, LAT99, BIN05]. However, as the number of servers and phases grows, the number of equations to solve grows combinatorially ("dimensionality curse"), effectively precluding the solution of systems with larger numbers of servers and phases.

While a few approximations have been proposed in the literature [BOL05], it has been shown that even in the simpler case of an *M*/*Ph*/*c* queue, these approximations fail to capture the fundamental dependency of performance measures such as the expected number in the system on higher-order properties of the service time distribution [GUP10,BEG13].

We propose a simple approximate solution for the *Ph*/*Ph*/*c*/*N* queueing system whose goal is to reduce the complexity of the problem when the distribution of the time between arrivals comprises a larger number of phases (say, 4 or more). In essence, our approach replaces the solution of a *Ph*/*Ph*/*c*/*N* queue by an iteration between the solutions of simpler *M*/*Ph*/*c*/*N* and *Ph*/*M*/*c*/*N* queues, resulting in potentially significant reduction in overall complexity. It is a generalization of the approximate solution of the *Ph*/*Ph*/*1* queue presented in [BRA12a].

Our paper is organized as follows. In Section 2, we describe in more detail the queueing model under consideration and we present our proposed approximate solution. Section 3 is devoted to numerical results illustrating the good accuracy of the proposed approximation, as well as the considerable expected gain in the speed of the solution. Section 4 concludes this paper.

2 System analyzed and its approximate solution

The *Ph*/*Ph*/*c*/*N* queueing model under consideration is represented in Figure 1. We denote by a and b the number of phases used to represent the distributions of the time between arrivals and of the service time, respectively. We also denote by $p(n), n = 0,...,N$ the steady-state probability that there are n users in the system (queued and in service).

In our method, we iterate between the solution of an *M*/*Ph*/*c*/*N* and a *Ph*/*M*/*c*/*N* queue. For the *M*/*Ph*/*c*/*N* queue, the arrivals are represented by a state-dependent rate of arrivals $w(n), n \geq 0$, and the service time distribution is the complete phase-type distribution with b phases. The solution of this queue produces $p(n)$ and the conditional rate of service $u(n), n \geq 1$ given that there are n users in the queue. This rate of service is used to solve the *Ph*/*M*/*c*/*N* queue with the complete phase-type distribution of the time between arrivals with a phases. The solution of this queue produces $p(n)$, as well as the conditional rate of arrivals given that there are n users in the queue, $w(n), n \geq 0$.

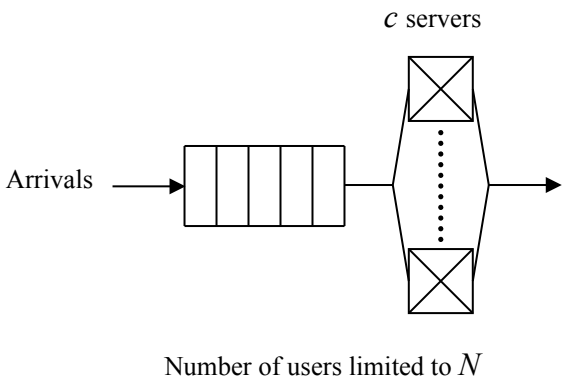


Fig. 1. The *Ph*/*Ph*/*c*/*N* queue

Thus, we need the $w(n)$ to solve the *M*/*Ph*/*c*/*N* queue to obtain the $u(n)$ needed to solve the *Ph*/*M*/*c*/*N* queue to produce the values of $w(n)$, naturally leading to a fixed-point iteration. We stop the iteration when the steady-state distributions $p(n)$ produced by the two models become sufficiently close, as measured by the mean number of users in the system $\bar{n} = \sum_{n=0}^{N} np(n)$.

The resulting fixed-point iteration can be summarized as follows:

Algorithm 1

1. Initialize the values of $w(n), n \geq 0$ to the inverse of the mean time between arrivals.
2. Solve the *M*/*Ph*/*c*/*N* queue using the current values of $w(n), n \geq 0$.
   a. Obtain current values for $p(n)$ and for $u(n)$.
   b. Compute the current value of $\overline{n}$ from this model.
3. Solve the *Ph*/*M*/*c*/*N* queue using the current values of $u(n), n \geq 1$ from Step 2
   a. Obtain current values for $p(n)$ and for $w(n)$.
   b. Compute the current value of $\overline{n}$ from this model.
4. If the values of $\overline{n}$ from Step 2 and Step 3 deviate by less than $\varepsilon > 0$ then exit, otherwise go to Step 2.

Note that the steady-state probability $p(n)$ can be expressed as

$$p(n) = \frac{1}{G} \prod_{i=1}^{n} \frac{w(i-1)}{u(i)}, \quad n = 0,1,...,N \tag{1}$$

where G is a normalizing constant. Hence, we have $u(n) = p(n-1)w(n-1)/p(n)$. This formula allows one to determine the values of the conditional service rates $u(n), n \geq 1$ if a method such as the matrix-geometric is used to solve the *M*/*Ph*/*c*/*N* queue. We solve the *Ph*/*M*/*c*/*N* queue using the fast and stable recurrence described in [BRA12b], which produces directly the values of $w(n), n \geq 0$.

We don't have a theoretical proof of the convergence of the proposed fixed-point iteration to a unique solution. In practice, we used the value of $\varepsilon = 10^{-7}$ for the exit test. In the several thousand numerical examples we have explored, the method never failed to converge within typically just a few tens of iterations.

a	Number of phases for the inter-arrival time distribution
b	Number of phases for the service time distribution
c	Number of servers
N	Buffer space, i.e. maximum of users in the system (queued and in service)
$p(n)$	Marginal probability that there are n users in the system
$u(n)$	Overall departure rate from the set of c servers given that the current number of users in the system is n

$w(n)$	Arrival rate at the queue given that the current number of users in the system is n
$\bar{n}$	Mean number of users in the system
p_{loss}	Loss probability (i.e. probability that a user finds the buffer full upon arrival)

Table 1. Notation used

The proposed approach decomposes the solution of a *Ph*/*Ph*/*c*/*N* queue into the solution of an *M*/*Ph*/*c*/*N* queue with state-dependent rate of arrivals, and that of a *Ph*/*M*/*c*/*N* queue with state-dependent service rates. Such decomposition would be exact if we knew the rates of arrivals as a function of both the number of users n and the current phase of the service process, as well as the rates of service as a function of n and of the current phase of the arrival process. Since we determine them only as a function of n, the method is approximate. As pointed out in [BRA12a], in the case of a single server ($c = 1$) the missing phase information may be important when there is a small number of users in service (mostly 1). Therefore, we would expect that the accuracy of the approximation would tend to improve as the number of servers increases. This is confirmed in our numerical results.

3 Accuracy and speed

To assess the accuracy of the proposed approximation, we examine the relative percentage errors for the mean number in system ${}_N\bar{n}$ and for the loss probability p_{loss} which can be expressed as $p_{loss} = w(N)p(N) / \sum_{n=0} w(n)p(n)$.

In all examples presented in this paper, we use a 3-phase hyper-exponential distribution with mean equal to 1 and coefficient of variation close to 1.5. Such a distribution corresponds to a mixture of short, medium and much longer service times. To study the accuracy of the proposed method, we use two types of distributions for the time between arrivals. The first one is a hyper-exponential distribution with 4 phases (H-4) and a coefficient of variation of 3. The second distribution is a 16-phase representation of a Pareto-like heavy-tailed distribution with shape parameter 1.5 and scale parameter 4 obtained using the PhFit package [HOR02]. Details of the probability distributions used in our examples are given in the Appendix.

For simplicity, when considering different workload level, we use the notion of offered load per server, defined as the ratio of the mean rate of user arrivals (including arrivals lost due to buffer overflow) to the number of servers, recalling that the mean service time is set to 1.

Figure 2 shows the relative percentage error for the average number in system $\overline{n}$ as a function of the number of servers c and of the offered load per server. The buffer size is set to $N = 4c$.

For the H-4 distribution of the times between arrivals, we observe that the relative error remains below 6%. It tends to be largest when the offered load per server is around 1, and it tends to decrease as the number of servers increases. We notice a similar behavior with the Pareto-like distribution except that the accuracy is slightly better and the improvement as the number of servers increases is slower.

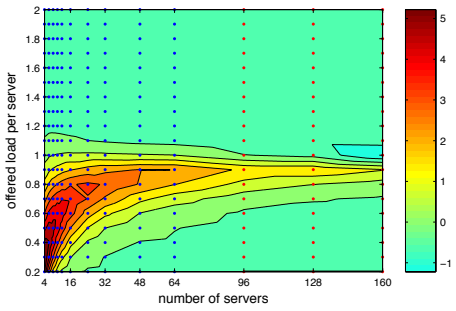


Fig. 2a. H-4 distribution for inter-arrival times

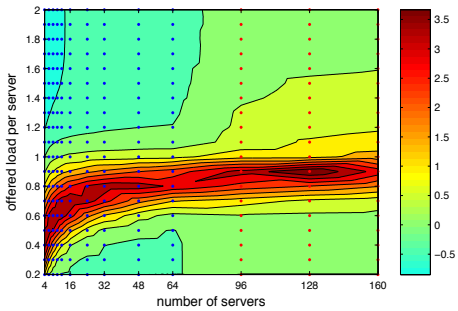


Fig. 2b. Pareto-like distribution for inter-arrival times

Fig. 2. Percentage relative errors of the approximate solution for the mean number of users in *Ph*/*Ph*/*c*/*N* queue with $N = 4c$

Table 2 shows the overall frequency distribution of relative errors for $\overline{n}$ for both types of distributions of times between arrivals. We observe that in over 99% of cases the relative error remains below 5% and the mean relative error is less than 1%. These results pertain to a total of over 450 study cases.

Mean	<5%	5-10%	10-15%	>15%
0.91%	99.14%	0.86%	0 %	0%

Table 2. Overall accuracy of the approximate solution for the mean number of users in *Ph*/*Ph*/*c*/*N* queue

Figure 3 shows the relative error for the loss probability p_{loss} with offered per server load kept at 1 as a function of the number of servers c and of the buffer size N. This type of figure is of interest when dimensioning the system to assure an acceptable loss ratio. We selected the value 1 for the offered load because with this offered load the loss probability is highly sensitive to the buffer size. Moreover, it is for this value of the offered load that our approximation tends to be the least accurate so that an even better accuracy can be expected for other values of the offered per server

load. We observe that the relative error in p_{loss} remains below 7% for both the H-4 and Pareto-like distributions. Table 3 gives the overall frequency distribution for the relative errors in the loss probability. In some 83% of cases the relative error for the loss probability is below 5%, and the mean relative error is less than 3%. These results were obtained from over 480 study cases.

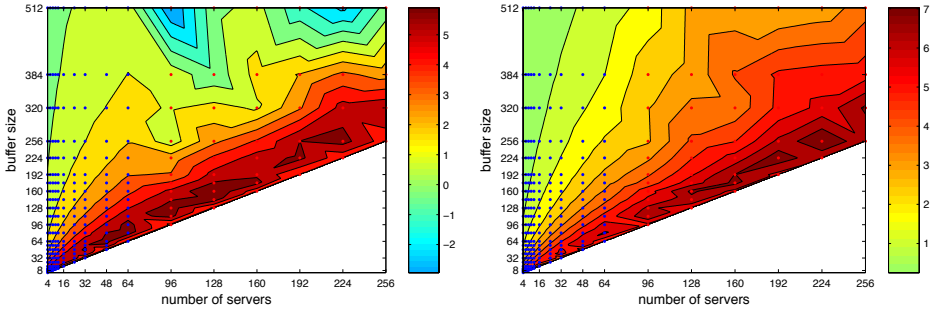


Fig. 3a. H-4 distribution for inter-arrival times

Fig. 3b. Pareto-like distribution for inter-arrival times

Fig. 3. Percentage relative errors of the approximate solution for the loss probability in *Ph*/*Ph*/*c*/*N* queue with an offered load equal to 1

Mean	<5%	5-10%	10-15%	>15%
2.82%	83.37%	16.63%	0%	0%

Table 3. Overall accuracy of the approximate solution for the loss probability in *Ph*/*Ph*/*c*/*N* queue with an offered load equal to 1

Overall, the results presented illustrate the good accuracy of the proposed approximation.

We now examine its practical numerical behavior starting with its speed of convergence. We denote by Δ the absolute value of the difference between the mean numbers of users in the systems obtained from the *M*/*Ph*/*c*/*N* and *Ph*/*M*/*c*/*N* models in the course of our fixed-point iteration between these two models.

In Figure 4, we illustrate the decrease in Δ as the iteration progresses. As an example, we display results obtained for a system with $c = 16$, $N = 64$ and offered per server load of 1. Here, for the distribution of the times between arrivals, in addition to the Pareto-like and H-4 distributions, we use also a hyper-exponential with 8 phases (H-8) and the same coefficient of variation of 3 as the H-4 (see Appendix). Note that Figure 4 uses a logarithmic scale for the y-axis clearly showing the rapid convergence of our iteration. We observe an essentially monotonous fast decrease in Δ. This type of behavior seems typical for our method. Although we do not have a theo-

retical proof of convergence, in the thousands of examples we have explored, our fixed point never failed to converge within a small to moderate number of iterations.

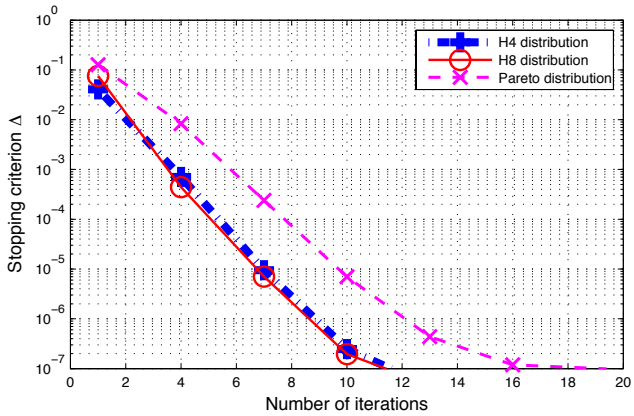


Fig. 4. Convergence speed of the approximate solution for a *Ph*/*Ph*/*c*/*N* queue with c=16, N=64 and an offered load equal to 1

Clearly, even with its good accuracy, our method is only of interest if its overall execution time is significantly faster than that of a full exact solution of the original *Ph*/*Ph*/*c*/*N* queue. It is intuitively clear that, for a given service time distribution, the attractiveness of our method is likely to increase as the number of phases in the arrival process increases. We now examine the ratio of the execution time of the full numerical solution of the *Ph*/*Ph*/*c*/*N* queue to the execution time of the proposed approximation. To keep this ratio meaningful, we use the same type of method (e.g. matrix-geometric, direct iteration, etc) to solve the *M*/*Ph*/*c*/*N* queue within our fixed-point iteration and the full *Ph*/*Ph*/*c*/*N* queue. Thus our execution time ratio compares multiple invocations of the solution of the *M*/*Ph*/*c*/*N* queue versus a single invocation of the *Ph*/*Ph*/*c*/*N* queue.

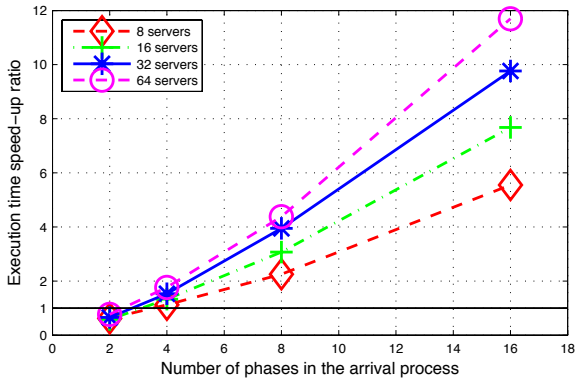


Fig. 5. Gain obtained in execution times when using the approximate solution for *Ph*/*Ph*/*c*/*N* queues with $N = 4c$ and an offered load equal to 1

In Figure 5, we display the ratio of the execution times between the solution of a full *Ph*/*Ph*/*c*/*N* queue and our approximation as a function of the number of phases in the arrival process for 4 different values of the number of servers c. We note that starting with 4 phases in the arrival process our approximation outperforms the single invocation of the solution of the *Ph*/*Ph*/*c*/*N* queue. The speed advantage of our approximation tends to increase with the number of servers. The approximation is roughly twice as fast as the full solution for some 5 or 6 phases and its speed advantage increases rapidly with the number of phases in the arrival process. This makes our approximation attractive from the standpoint of execution speed when the number of phases is 4 or more.

4 Conclusions

In this paper, we consider the solution of a model of a multi-server (or agent) facility with general inter-arrival and service time distributions and a finite buffer size. It may be noted that a simple *M*/*M*/*c*/*N* queue with the same mean service and inter-arrival time in general does not produce results accurate enough to be of practical use. Few reliable approximations seem to exist for the *Ph*/*Ph*/*c*/*N* queue, and its full numerical solution suffers from the well known "dimensionality curse". We present an approximate solution for such a queueing system. Our solution involves a fixed-point iteration between the solutions of two simpler queues: the *M*/*Ph*/*c*/*N* and the *Ph*/*M*/*c*/*N* queue. The proposed approximation is conceptually simple; it is also easy to implement. Although we do not have a theoretical proof of convergence of the fixed point to a unique solution, in many thousands of numerical examples it never failed to converge within typically a small to moderate number of iterations.

A comparison of its execution speed versus that of the direct solution of the full *Ph*/*Ph*/*c*/*N* queue indicates that our approximation becomes interesting from the standpoint of its execution speed starting with some 4 phases in the arrival process. It is important to note that our approximation partitions the state space by de-coupling the complexity of the service and arrival processes thus requiring a smaller memory space that the full numerical solution of the *Ph*/*P*/*h*/*c*/*N* queue.

This paper presents only a small fraction of our numerical results. We studied the accuracy of the approximation for a number of combinations of distributions for the inter-arrival and service times, including low and high variability distributions (hypo-exponential, Erlang, Cox, hyper-exponential, Pareto-like). We also explored a large number of configurations with different numbers of servers and buffer sizes for a large range of offered loads. It is our conclusion that the accuracy of the approximation is generally good with respect to the mean number in the system and the loss probability.

References

[BEG13] Begin, T., and Brandwajn, A. A note on the accuracy of several existing approximations for *M/Ph/m* queues. In Proceedings of *HSNCE*, Kyoto, Japan (2013).

[BIN05] Bini, D. A., Latouche, G., and Meini, B. *Numerical Methods for Structured Markov Chains*. Oxford University Press, Inc. (2005)

[BOL05] Bolch, G., Greiner, S., Meer, H., and Trivedi, K. *Queueing Networks and Markov Chains*. Second Edition, Wiley-Interscience (2005).

[BRA09] Brandwajn, A., and Begin, T. Preliminary Results on a Simple Approach to *G/G/c*-like Queues. In Proceedings of *Analytical and Stochastic Modeling Techniques and Applications (ASMTA)*. pp. 159-173 (2009).

[BRA12a] Brandwajn, A., and Begin, T. An approximate solution for *Ph/Ph/1* and *Ph/Ph/1/N* queues. In *Proceedings of the ACM/SPEC International Conference on Performance Engineering (ICPE)* (2012).

[BRA12b] Brandwajn, A., and Begin, T. A Recurrent Solution of *Ph/M/c/N*-like and *Ph/M/c*-like Queues. *Journal of Applied Probability* 49.1, pp. 84-99 (2012).

[GAN03] Gans, N., Koole, G., and Mandelbaum, A. Telephone call centers: tutorial, review, and research prospects. *Manufacturing and Service Operations Management*, 5, pp. 79-141 (2003).

[GUP10] Gupta, V., Harchol-Balter, M., Dai, J., and Zwart, B. On the inapproximability of *M/G/K*: why two moments of job size distribution are not enough. *Queueing Systems*, Vol. 64 (1), pp. 5-48 (2010).

[HOR02] Horváth, A., and Telek, M. Phfit: A general phase-type fitting tool. In Proceedings of *Computer Performance Evaluation: Modelling Techniques and Tools* (TOOLS) pp. 82-91 (2002).

[LAT93] Latouche, G., and Ramaswami, V. A logarithmic reduction algorithm for quasi-birth-and-death processes. *Journal of Applied Probability*. vol. 30, pp. 650-674 (1993).

[LAT99] Latouche , G.,and Ramaswami, V. *Introduction to Matrix Analytic Methods in Stochastic Modeling*, ASA (1999).

[RAM86] Ramaswami, V., and Lucantoni, D.M. Algorithms for the multi-server queue with phase type service, *Stochastic Models,* Vol. 1, pp. 393-417 (1985).

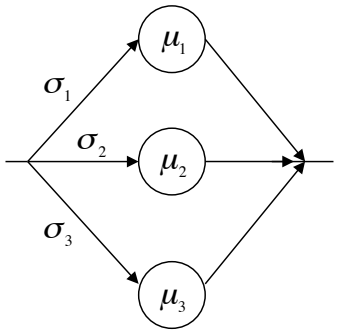
[SEE86] Seelen, L. P. An Algorithm for *Ph/Ph/c* Queues, *European Journal of the Operations Research Society*, Vol. 23, pp. 118-127 (1986).

[TAK76] Takahashi, Y., and Takami, Y. A Numerical Method for the Steady-State Probabilities of a *GI/G/s* Queueing system in a General Class, *Journal of the Operations Research Society of Japan*, Vol. 19, pp. 147-157 (1976).

Appendix

1 - H-3 service time distribution

The parameters of this distribution have been selected to represent a mixture of short, medium and long service times. The resulting distribution has a mean of 1 and a coefficient of variation of 1.46. Figure 6 shows the distribution and gives the values of its parameters.

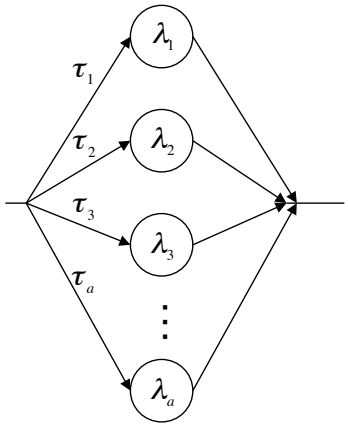


Probabilities		Phase rate	
σ_1	0.2	μ_1	1
σ_2	0.6	μ_2	1/5
σ_3	0.2	μ_2	1/15

Fig. 6. Distribution for the service times

2 - H-4 and H-8 inter-arrival time distributions

Figure 7 shows the structure of these distributions. The parameter values shown correspond to a mean of 1 and a coefficient of variation of 3. For inter-arrival times different from 1, the rates of all phases are scaled so as to produce the proper mean.



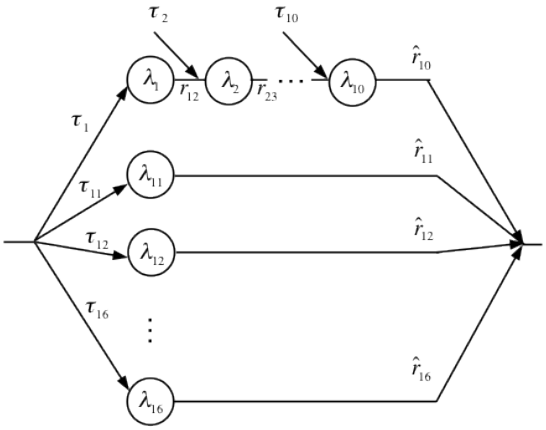
Probabilities		Phase rate	
τ_1	0.5	λ_1	1
τ_2	0.3	λ_2	1/10
τ_3	0.199	λ_3	1/100
τ_4	$\tau_4 = 1 - \tau_1 - \tau_2 - \tau_3$	λ_4	1/1000

Probabilities		Phase rate	
τ_1	0.5	λ_1	1
τ_2	0.1	λ_2	1/5
τ_3	0.1	λ_3	1/25
τ_4	0.1	λ_4	1/50
τ_5	0.0101	λ_5	1/100
τ_6	0.1	λ_6	1/300
τ_7	0.07788	λ_7	1/500
τ_8	$\tau_8 = 1 - \sum_{i=1}^{7} \tau_i$	λ_8	1/1000

Fig. 7. Hyper-exponential distributions for the inter-arrival times

3 - Pareto-like inter-arrival time distribution

The structure of the Pareto-like distribution used in our examples is shown in Figure 8. The parameter values represented correspond to a mean time between arrivals of 7.797 and a coefficient of variation of 13.77. To obtain any desired time between arrivals, we scale the rates of all phases by the appropriate factor.



Probabilities		**Phase rate**	
τ_1	1.39201642e-004	λ_1	1.49162788e+001
τ_2	1.77752368e-004	λ_2	1.34254874e+001
τ_3	1.91426701e-005	λ_3	1.06568964e+001
τ_4	3.45978796e-005	λ_4	9.00035659e+000
τ_5	4.75808142e-005	λ_5	7.61426145e+000
τ_6	1.99877132e-005	λ_6	4.77130864e+000
τ_7	7.26038667e-005	λ_7	3.75562889e+000
τ_8	1.39327654e-004	λ_8	2.37959901e+000
τ_9	6.32355379e-002	λ_9	1.97633786e+000
τ_{10}	2.83641569e-001	λ_{10}	1.76580558e+000
τ_{11}	2.00491113e-006	λ_{11}	2.23396046e-005
τ_{12}	3.08716391e-005	λ_{12}	1.69138281e-004
τ_{13}	4.11347558e-004	λ_{13}	9.82050775e-004
τ_{14}	5.33726475e-003	λ_{14}	5.48273618e-003
τ_{15}	6.52276871e-002	λ_{15}	3.01135332e-002
τ_{16}	5.81463522e-001	λ_{16}	1.55607459e-001
$r_{12}, r_{23}, ... = \hat{r}_{10}, \hat{r}_{11}, ... = 1$			

Fig. 8. Pareto-like distribution for the inter-arrival times

RESEARCH CENTRE
GRENOBLE - RHÔNE-ALPES

Inovallée
655 avenue de l'Europe - Montbonnot
38334 Saint Ismier Cedex France

Publisher
Inria
Domaine de Voluceau - Rocquencourt
BP 105 - 78153 Le Chesnay Cedex
inria.fr
ISSN 0249-6399