UNITÉ DE RECHERCHE
INRIA-RENNES

Institut National
de Recherche
en Informatique
et en Automatique

Domaine de Voluceau
Rocquencourt
B.P. 105
78153 Le Chesnay Cedex
France
Tél.:(1) 39 63 55 11

Rapports de Recherche

N° 1501

Programme 1
Architectures parallèles, Bases de données,
Réseaux et Systèmes distribués

CALCULATING THE BUSY PERIOD DISTRIBUTION OF THE M/M/1 QUEUE

Louis-Marie LE NY
Gerardo RUBINO
Bruno SERICOLA

Septembre 1991

IRISA

INSTITUT DE RECHERCHE EN INFORMATIQUE
ET SYSTÈMES ALÉATOIRES

Campus Universitaire de Beaulieu
35042 - RENNES CÉDEX
FRANCE
Téléphone : 99 36 20 00
Télex : UNIRISA 950 473 F
Télécopie : 99 38 38 32

Calculating the busy period distribution of the M/M/1 queue

Louis-Marie Le Ny, Gerardo Rubino, Bruno Sericola

Programme I

Publication Interne n° 595 - Juillet 1991 - 10 pages

Abstract

This paper deals with the computation of the busy period distribution of the M/M/1 queue. We present a simple expression of this distribution, without any use of transforms or Bessel functions. This expression leads to an algorithm which performs the computation with an error tolerance specified by the user.

Index Terms – M/M/1 queue, busy period distribution, Catalan's numbers.

Calcul de la distribution de la période d'activité de la file M/M/1

Résumé

Nous étudions dans ce rapport la distribution de la période d'activité du serveur d'une file M/M/1. Nous donnons une expression simple de cette distribution n'utilisant ni transformées, ni fonctions de Bessel. Cette expression conduit à un algorithme faisant le calcul avec une précision définie par l'utilisateur.

Mots clés – File M/M/1, distribution de la période d'activité, nombres de Catalan.

CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE
(L. A. 227)
UNIVERSITÉ DE RENNES 1 I. N. S. A. DE RENNES

INSTITUT NATIONAL DE RECHERCHE
EN INFORMATIQUE ET EN AUTOMATIQUE
(LABORATOIRE DE RENNES)

1 Introduction

Several papers have been proposed for the numerical computation of the M/M/1 transient behaviour. We reference only one of the most recent, written by J. Abate and W. Whitt [1], in which numerous methods and references could be found for the analysis of this queue. It is interesting to note that almost all these papers are based on the use of Bessel functions.

In this paper, we focus on the busy period distribution of the M/M/1 queue. Using results on sojourn times in Markov processes [2], we derive a simple expression of this distribution avoiding Bessel functions. This expression is given in Section 2 and it holds for any value of the arrival and service rates. It consists of a series involving the Poisson distribution and Catalan's numbers. In Section 3, we show how to compute this series by truncating it such that its rest is as small as desired, and we give an algorithmic scheme to organize the computations. The last section is devoted to some conclusions.

2 The busy period distribution

Consider the classical M/M/1 queue with arrival rate λ and service rate μ. We denote by A the infinitesimal generator of the continuous time Markov process associated to the state of the queue. The non zero entries of this matrix are

$$A(0,0) = -\lambda \text{ and for any } i \geq 1, \;\; A(i,i-1) = \mu, \;\; A(i,i) = -(\lambda+\mu), \;\; A(i,i+1) = \lambda.$$

Let BP_i, $i \geq 1$, be the duration of the ith busy period. If the initial state is state 0 or state 1, these random variables are i.i.d. The random variable BP_i can be seen as the ith sojourn time of the process in the subset of states $\{1,2,\ldots\}$. Using the results obtained in [2] for finite Markov processes and observing that the transition rates of the M/M/1 queue are uniformly bounded (for every $i,j \geq 0$, $|A(i,j)| \leq \lambda+\mu$), it can be easily verified that the common distribution of these sojourn times denoted now by BP is given by

$$\mathbb{P}(BP \leq t) = 1 - \alpha e^{A_1 t} 1^T \tag{1}$$

where $\alpha = (1,0,0,\ldots)$ and A_1 is the submatrix deduced from A by deleting the row and the column corresponding to state 0. The row vector $1 = (1,1,1,\ldots)$ has all its entries equal to 1 and T denotes the transpose operator.

We consider now the embedded discrete time Markov chain at the instants of state change and we denote by P its transition probability matrix. In the same way as for matrix A, we define P_1 as the submatrix deduced from P by deleting the row and the column corresponding to state 0. For every $i \geq 1$, the non zero entries of P_1 are

$$P_1(i,i-1) = \frac{\mu}{\lambda+\mu}, \;\; P_1(i,i+1) = \frac{\lambda}{\lambda+\mu}.$$

In matrix notation, we have

$$P_1 = I + \frac{1}{\lambda+\mu} A_1$$

or equivalently,

$$A_1 = -(\lambda+\mu)(I - P_1),$$

where I denotes the identity matrix. Relation (1) can be now written

$$\begin{aligned}
\mathbb{P}(BP \leq t) &= 1 - \alpha e^{-(\lambda+\mu)(I-P_1)t} 1^T \\
&= 1 - e^{-(\lambda+\mu)t} \sum_{k=0}^{+\infty} \frac{(\lambda+\mu)^k t^k}{k!} \alpha P_1^k 1^T \\
&= \sum_{k=0}^{+\infty} e^{-(\lambda+\mu)t} \frac{(\lambda+\mu)^k t^k}{k!} \left(1 - \alpha P_1^k 1^T\right).
\end{aligned}$$

Let us denote by N the number of states visited during a busy period. For every $i \geq 1$ and $k \geq 0$, we denote by $\mathbb{P}_i(N = k)$ the probability that the number of states visited during a busy period is equal to k given that the initial state is i. Defining

$$p \equiv \frac{\lambda}{\lambda+\mu} \text{ and } q \equiv \frac{\mu}{\lambda+\mu},$$

we obtain the following renewal equations:

$$\begin{aligned}
\mathbb{P}_i(N = 0) &= 0 && \text{for every } i \geq 1, \\
\mathbb{P}_1(N = 1) &= q, && \\
\mathbb{P}_i(N = 1) &= 0 && \text{for every } i \geq 2, \\
\mathbb{P}_1(N = k) &= p\mathbb{P}_2(N = k-1) && \text{for every } k \geq 2, \\
\mathbb{P}_i(N = k) &= p\mathbb{P}_{i+1}(N = k-1) + q\mathbb{P}_{i-1}(N = k-1) && \text{for every } i \geq 2 \text{ and } k \geq 2.
\end{aligned}$$

Denoting by $V(k)$, $k \geq 0$, the column vector with ith entry, $i \geq 1$, equal to $\mathbb{P}_i(N = k)$, the previous relations can be written in matrix notation as

$$V(0) = 0, \quad V(1) = q\alpha^T, \quad V(k) = P_1 V(k-1) \text{ for every } k \geq 2.$$

The column vector $V(1)$ can also be written as $V(1) = (I - P_1)1^T$. We then deduce that, for every $k \geq 1$,

$$V(k) = P_1^{k-1}(I - P_1)1^T$$

and

$$\mathbb{P}_1(N = k) = \alpha V(k) = \alpha P_1^{k-1}(I - P_1)1^T.$$

It follows that

$$\mathbb{P}_1(N \leq k) = 1 - \alpha P_1^k 1^T \text{ for every } k \geq 0.$$

Moreover, since the initial state has been choosen to be 0, we set

$$\mathbb{P}(N \leq k) \equiv \mathbb{P}_0(N \leq k) = \mathbb{P}_1(N \leq k).$$

These considerations lead to the following formula for the busy period distribution:

$$\begin{aligned}\mathbb{P}(BP \leq t) &= \sum_{k=0}^{+\infty} e^{-(\lambda+\mu)t}\frac{(\lambda+\mu)^k t^k}{k!}\mathbb{P}(N \leq k) \\ &= \sum_{k=1}^{+\infty} e^{-(\lambda+\mu)t}\frac{(\lambda+\mu)^k t^k}{k!}\mathbb{P}(N \leq k).\end{aligned}$$

It is well known [3] that the probability of serving n customers in a busy period starting in state 1 is

$$\binom{2n-2}{n-1}\frac{p^{n-1}q^n}{n}.$$

The number of customers served in a busy period is $r+1$ iff the number of states visited in a busy period is $2r+1$. So, we obtain that for every $r \geq 0$,

$$\mathbb{P}(N = 2r+1) = \binom{2r}{r}\frac{p^r q^{r+1}}{r+1}$$

and obviously, we have $\mathbb{P}(N = 2r) = 0$ for every $r \geq 0$. For $r \geq 0$, the integers

$$\binom{2r}{r}\frac{1}{r+1}$$

are known as the Catalan's numbers. The distribution of the busy period becomes

$$\mathbb{P}(BP \leq t) = \sum_{k=1}^{+\infty} e^{-(\lambda+\mu)t}\frac{(\lambda+\mu)^k t^k}{k!}\sum_{r=1}^{k}\mathbb{P}(N = r),$$

that is,

$$\mathbb{P}(BP \leq t) = \sum_{k=1}^{+\infty} e^{-(\lambda+\mu)t}\frac{(\lambda+\mu)^k t^k}{k!}\sum_{r=0}^{\lfloor\frac{k-1}{2}\rfloor}\binom{2r}{r}\frac{p^r q^{r+1}}{r+1}. \tag{2}$$

Observe that

$$\lim_{t\longrightarrow+\infty}\mathbb{P}(BP < t) = \sum_{r=0}^{+\infty}\binom{2r}{r}\frac{p^r q^{r+1}}{r+1}.$$

It can be shown (see for instance [4]) that for $0 < x \leq 1/4$,

$$\sum_{r=0}^{+\infty}\binom{2r}{r}\frac{x^r}{r+1} = \frac{1-\sqrt{1-4x}}{2x}.$$

Here, $pq \leq 1/4$ since $p+q=1$, so

$$\sum_{r=0}^{+\infty}\binom{2r}{r}\frac{p^r q^{r+1}}{r+1} = \frac{1-\sqrt{1-4pq}}{2p} = \begin{cases} 1 & \text{if } p \leq \frac{1}{2} \\ \frac{1-p}{p} & \text{if } p > \frac{1}{2}\end{cases}$$

This remark leads to the well-known result:

$$\lim_{t \longrightarrow +\infty} \mathbb{P}(BP < t) = \begin{cases} 1 & \text{if} \quad \lambda \leq \mu, \\ \frac{\mu}{\lambda} & \text{if} \quad \lambda > \mu. \end{cases}$$

We will denote by $h(\lambda, \mu)$ this limit, that is,

$$h(\lambda, \mu) \equiv \min \left\{ 1, \frac{\mu}{\lambda} \right\}.$$

3 Computing the busy period distribution

In this section, we consider formula (2) in two different ways. In Subsection 3.1, we perform a truncation over index k such that the rest of the series becomes less than or equal to a given error tolerance ε. In Subsection 3.2, we permute the two sums in formula (2) and we perform a truncation over index r with the same error tolerance ε. The third subsection gives an outline on the computation of the distribution of the busy period. The reasons of these manipulations will appear along the discussion.

3.1 Truncating over index k

To compute the distribution of the busy period, we have first to evaluate the truncation step K in order to have the rest of the series (2) less than or equal to a given error tolerance ε. Denoting by $e(K)$ the rest of this series, we have

$$\begin{aligned} e(K) &= \sum_{k=K+1}^{+\infty} e^{-(\lambda+\mu)t} \frac{(\lambda+\mu)^k t^k}{k!} \sum_{r=0}^{\lfloor \frac{k-1}{2} \rfloor} \begin{pmatrix} 2r \\ r \end{pmatrix} \frac{p^r q^{r+1}}{r+1} \\ &\leq h(\lambda, \mu) \sum_{k=K+1}^{+\infty} e^{-(\lambda+\mu)t} \frac{(\lambda+\mu)^k t^k}{k!} \\ &= h(\lambda, \mu) \left(1 - \sum_{k=0}^{K} e^{-(\lambda+\mu)t} \frac{(\lambda+\mu)^k t^k}{k!} \right). \end{aligned}$$

The integer K is choosen as the smallest one verifying

$$h(\lambda, \mu) \left(1 - \sum_{k=0}^{K} e^{-(\lambda+\mu)t} \frac{(\lambda+\mu)^k t^k}{k!} \right) \leq \varepsilon. \tag{3}$$

We then have the following result:

$$0 \leq \mathbb{P}(BP \leq t) - \sum_{k=1}^{K} e^{-(\lambda+\mu)t} \frac{(\lambda+\mu)^k t^k}{k!} \sum_{r=0}^{\lfloor \frac{k-1}{2} \rfloor} \begin{pmatrix} 2r \\ r \end{pmatrix} \frac{p^r q^{r+1}}{r+1} \leq \varepsilon.$$

So, $\mathbb{P}(BP \leq t)$ can be computed using the following expression, with a precision equal to ε:

$$\sum_{k=1}^{K} e^{-(\lambda+\mu)t} \frac{(\lambda+\mu)^k t^k}{k!} \sum_{r=0}^{\lfloor \frac{k-1}{2} \rfloor} \begin{pmatrix} 2r \\ r \end{pmatrix} \frac{p^r q^{r+1}}{r+1}. \tag{4}$$

The number of terms that must be computed using this formula is equal to $(K+1)^2/4$ if K is odd, and to $K(K+2)/4$ if K is even.

3.2 Truncating over index r

Let us first permute the two sums in formula (2). We obtain

$$\mathbb{P}(BP \leq t) = h(\lambda,\mu) - \sum_{r=0}^{+\infty} \binom{2r}{r} \frac{p^r q^{r+1}}{r+1} \sum_{k=0}^{2r} e^{-(\lambda+\mu)t} \frac{(\lambda+\mu)^k t^k}{k!}.$$

Using the same argument as in the previous subsection, we get

$$\mathbb{P}(BP \leq t) = h(\lambda,\mu) - \sum_{r=0}^{R} \binom{2r}{r} \frac{p^r q^{r+1}}{r+1} \sum_{k=0}^{2r} e^{-(\lambda+\mu)t} \frac{(\lambda+\mu)^k t^k}{k!} - e'(R)$$

where

$$\begin{aligned} e'(R) &= \sum_{r=R+1}^{+\infty} \binom{2r}{r} \frac{p^r q^{r+1}}{r+1} \sum_{k=0}^{2r} e^{-(\lambda+\mu)t} \frac{(\lambda+\mu)^k t^k}{k!} \\ &\leq \sum_{r=R+1}^{+\infty} \binom{2r}{r} \frac{p^r q^{r+1}}{r+1} \\ &= h(\lambda,\mu) - \sum_{r=0}^{R} \binom{2r}{r} \frac{p^r q^{r+1}}{r+1}. \end{aligned}$$

So, integer R is choosen as the smallest one verifying

$$h(\lambda,\mu) - \sum_{r=0}^{R} \binom{2r}{r} \frac{p^r q^{r+1}}{r+1} \leq \varepsilon. \tag{5}$$

We then have

$$-\varepsilon \leq \mathbb{P}(BP \leq t) - \left[h(\lambda,\mu) - \sum_{r=0}^{R} \binom{2r}{r} \frac{p^r q^{r+1}}{r+1} \sum_{k=0}^{2r} e^{-(\lambda+\mu)t} \frac{(\lambda+\mu)^k t^k}{k!} \right] \leq 0.$$

This result allows to compute $\mathbb{P}(BP \leq t)$, with a precision equal to ε, by the following expression:

$$h(\lambda,\mu) - \sum_{r=0}^{R} \binom{2r}{r} \frac{p^r q^{r+1}}{r+1} \sum_{k=0}^{2r} e^{-(\lambda+\mu)t} \frac{(\lambda+\mu)^k t^k}{k!}. \tag{6}$$

The number of terms that must be computed using this formula is equal to $(R+1)^2$.

3.3 Algorithm

The two previous subsections lead naturally to a simple algorithm which uses expressions (4) or (6) depending on the values of the truncation steps K and R. The choice between these two expressions can be done by considering the number of terms appearing in their sums. These numbers of terms have been evaluated in each of the two previous subsections. It follows that if $K \leq 2R$ we shall use expression (4) and otherwise we shall use expression (6). This decision can be taken in the following way. Relations (3) and (5) are tested in the same loop and the first one satisfied will determine the corresponding truncation step and so the best expression that must be used. In all cases, the result is obtained with a precision ε given by the user.

One can note that the integer K is a function of λ, μ, ε and t, where R is only a function of λ, μ and ε. Moreover, it is easy to verify that if $t \leq t'$ then $K(\lambda,\mu,\varepsilon,t) \leq K(\lambda,\mu,\varepsilon,t')$. It follows that if relation (5) is the first satisfied for the value t, it will be also the first satisfied for all values greater than t. A general algorithm to compute the distribution of the busy period in M points $t_1 < \cdots < t_M$ can be written in the following way. For every integers k and r, we will use the notation:

$$\text{poi(k,t)} \equiv e^{-(\lambda+\mu)t}\frac{(\lambda+\mu)^k t^k}{k!} \quad \text{and} \quad \text{cat(r)} \equiv \begin{pmatrix} 2r \\ r \end{pmatrix} \frac{p^r q^{r+1}}{r+1}.$$

```
input : $\lambda$, $\mu$, $\varepsilon$, $t_1 < \cdots < t_M$
output : $\mathbb{P}(BP \leq t_1), \ldots, \mathbb{P}(BP \leq t_M)$
initialisation : chooseR := false; R := 0; K := 0; i := 0; sumcat := cat(0)
   while [ not chooseR and $i \leq M$ ] do
         i := i + 1
         sumpoi := poi(0,$t_i$)
         for l := 1 to K do sumpoi := sumpoi + poi(l,$t_i$) endfor
         while [ $h(\lambda,\mu)(1 -$ sumpoi$) > \varepsilon$ and $h(\lambda,\mu) -$ sumcat $> \varepsilon$ ] do
             K := K + 2
             sumpoi := sumpoi + poi(K-1,$t_i$) + poi(K,$t_i$)
             R := R + 1
             sumcat := sumcat + cat(R)
         endwhile
         if $h(\lambda,\mu)(1 -$ sumpoi$) \leq \varepsilon$ then $\mathbb{P}(BP \leq t_i)$ := Expression (4)
         else $\mathbb{P}(BP \leq t_i)$ := Expression (6); chooseR := true
         endif
   endwhile
   for j := i+1 to M do $\mathbb{P}(BP \leq t_j)$ := Expression (6) endfor
```

The first execution of the outer **while** loop is done for the value K=0, so that the **for** loop on index l is skipped. Assume that the inner **while** loop is left because $h(\lambda,\mu)(1 - \text{sumpoi}) \leq \varepsilon$. The busy period distribution is then computed using Expression (4) and the second value t_2 is considered. Note that the current values of the variables K and R are respectively $K(\lambda,\mu,\varepsilon,t_1)$ and $K(\lambda,\mu,\varepsilon,t_1)/2$ (this is due to the fact that the variable sumpoi cumulates two values at each step and the variable sumcat only one). At this point, the value of the variable sumcat is cat(0) + $\cdots$ + cat(R). Since $t_2 > t_1$, we know that $K(\lambda, \mu, \varepsilon, t_2) \geq K(\lambda, \mu, \varepsilon, t_1)$; moreover, recall that the truncation step R is independent of the values of t_i. That is why the **for** loop on index l is executed to compute the sum sumpoi of poi(l,t_2) for l $\leq$ K $= K(\lambda, \mu, \varepsilon, t_1)$. From that instant, the control passes to the inner **while** loop as in the first iteration step. It can be observed that the effective implementation of this algorithm needs some tradeoff between computing time and storage. Many intermediate calculations appearing in expressions (4) and (6) are performed when taking the decision on which of the two relations will be used. On the computations themselves, in addition of trivial points as the recurrent definition of poi(k,t) as a function of poi(k-1,t) or cat(r) as a function of cat(r-1), the reader can consult for instance [5] for some concerns on the underflow problem due to the exponential.

Remark that the choice between expressions (4) and (6) can lead to a considerable gain in computing time. Indeed, let for instance $\mu = 1.0$, $t = 100$ and $\varepsilon = 10^{-6}$. For $\lambda = 1.05$, we obtain $K = 276$ and $R = 9908$. For $\lambda = 0.99$, the corresponding values are $K = 270$ and $R = 166139$. On the other hand, for $\lambda = 10$, we obtain $K = 1244$ and $R = 7$.

4 Conclusions

In this paper, we have developped a simple algorithm to compute the busy period distribution of the M/M/1 queue. To obtain it, we have used results on sojourn times in Markov processes and Catalan's numbers. Further work could be the extension of the proposed method to the computation of other time-dependent measures for this queue and for similar models.

References

[1] J. Abate and W. Whitt, "Calculating time-dependent performance measures for the M/M/1 queue," *IEEE Transactions on Communications*, vol. 37, no. 10, pp. 1102–1104, 1989.

[2] G. Rubino and B. Sericola, "Sojourn times in finite Markov processes," *Journal of Applied Probability*, vol. 27, pp. 744–756, 1989.

[3] L. Kleinrock, *Queueing Systems, Volume 1.* John Wiley, 1975.

[4] R. L. Graham, D. E. Knuth, and O. Patashnik, *Concrete Mathematics.* Addison–Wesley, 1989.

[5] P. N. Bowerman, R. G. Nolty, and E. M. Scheuer, "Calculation of the poisson cumulative distribution function," *IEEE Transactions on Reliability*, vol. 39, no. 2, pp. 158–161, 1990.

LISTE DES DERNIERES PUBLICATIONS PARUES EN 1991

PI 582 PROGRAMMING REAL TIME APPLICATIONS WITH SIGNAL
Paul LE GUERNIC, Michel LE BORGNE, Thierry GAUTIER, Claude LE MAIRE
Avril 1991, 36 Pages.

PI 583 ELIMINATION OF REDUNDANCY FROM FUNCTIONS DEFINED BY SCHEMES
Didier CAUCAL
Avril 1991, 22 Pages.

PI 584 TECHNIQUES POUR LA MISE AU POINT DE PROGRAMMES REPARTIS
Michel ADAM, Michel HURFIN, Michel RAYNAL, Noël PLOUZEAU
Mai 1991, 10 Pages.

PI 585 TOWARDS THE CONSTRUCTION OF DISTRIBUTED DETECTION PROGRAMS, WITH AN APPLICATION TO DISTRIBUTED TERMINATION
Jean-Michel HELARY
Michel RAYNAL
Mai 1991, 24 Pages.

PI 586 OPAC : A COST-EFFECTIVE FOATING-POINT COPROCESSOR
André SEZNEC, Karl COURTEL
Mai 1991, 26 Pages.

PI 587 ON FAILURE DETECTION AND IDENTIFICATION : AN OPTIMUM ROBUST MIN-MAX APPROACH
Elias WAHNON, ALbert BENVENISTE
Mai 1991, 24 Pages.

PI 588 BOUNDED-MEMORY ALGORITHMS FOR VERIFICATION ON THE FLY
Claude JARD, Thierry JERON
Mai 1991, 14 Pages.

PI 589 UNE APPROCHE MULTIECHELLE A L'ANALYSE D'IMAGES PAR CHAMPS MARKOVIENS
Patrick PEREZ, Fabrice HEITZ
Juin 1991, 32 pages.

PI 590 THE IDEMPOTENT SOLUTIONS OF THE SEMI-UNIFICATION PROBLEM
Pascal BRISSET, Olivier RIDOUX
Juin 1991, 16 pages.

PI 591 AVARE UN PROGRAMME DE CALCUL DES ASSOCIATIONS ENTRE VARIABLES RELATIONNELLES
Mohamed OUALI ALLAH
Juin 1991, 32 pages.

PI 592 SCHEDULING IN DISTRIBUTED SYSTEMS : SURVEY AND QUESTIONS
Yasmina BELHAMISSI, Maurice JEGADO
Juin 1991, 36 pages.

PI 593	APPLICATION OF BELLEN'S PARALLEL METHOD TO ODE's WITH DISSIPATIVE RIGHT-HAND SIDE Philippe CHARTIER Juin 1991, 24 pages.
PI 594	PROGRAMMATION D'UN NOYAU UNIX EN GAMMA Pascale LE CERTEN, Hector RUIZ BARRADAS Juillet 1991, 48 pages.
PI 595	CALCULATING THE BUSY PERIOD DISTRIBUTION OF THE M/M/1 QUEUE Louis-Marie LE NY, Gerardo RUBINO, Bruno SERICOLA Juillet 1991, 11 pages.

ISSN 0249-6399