A numerical analysis method of queues with batch arrivals*

György Wolfner, Miklós Telek
Department of Telecommunication
Technical University of Budapest
Stoczek u. 2., H-1111 Budapest, Hungary
{wolfner,telek}@hit.bme.hu

Abstract

The phenomena of batch arrivals arises often in modern communication networks which are characterized by the coexistence of different transmission speeds and protocols. This paper proposes an effective numerical algorithm to evaluate queueing models with fix size batch arrivals.

The class of problems considered results in a quasi-birth-death (QBD) Markov chain whose block size linearly increases with the number of customers arrive to the system at an arrival instance. In the case of a fix batch size the submatrices of the QBD process has a special structure which is utilized to reduce the computational complexity of the analysis. The steps of the proposed numerical method is discussed together with considerations on the memory and computation requirements.

Numerical examples of "real" problem, which occurs when local area network data packages are transmitted through an ATM network, are evaluated.

Keywords: Quasi Birth Death process, batch arrival, queue analysis.

1. Introduction

With the extremely rapid evolution of communication and computer systems and with the intention of their integration, whose most well-known example is the introduction of the asynchronous transfer mode (ATM), the present and future communication networks are characterized by the coexistence of different transmission/service requirements, communication protocols and transmission speeds. With very simple assumptions on the stochastic behaviour of the network traffic (memoryless or Markov modulated arrival and service) the transfer of data from one part of a network to an other results in complex queue behaviour at the transfer point. For example, in one part of the network packets of size 1500 byte are transmitted (IP packet size used in Ethernet LANs) while in an other part cells of size 48+5 byte (the size of ATM cells, payload+header) are the base data units. This way when a packet of size 1500 byte arrive to a node from which cells are forwarded a single arrival instance means the arrival of 32 data cells. This phenomena is commonly referred to as batch arrival.

A queueing system with memoryless arrival and service can be analysed by the underlying Markov chain. When in addition the arrival and service is queue length independent and the batch size is bounded the underlying Markov chain has a nice block structure and is referred to as a quasi-birth-death (QBD) Markov chain [6]. For a good overview of the field we refer to the book of Neuts [6].

There are several numerical methods to evaluate the steady state behaviour of QBD processes. The most well-known is the one proposed by Neuts which is often referred to as Matrix Geometric Method (MGM) [6] and is based on an iterative procedure. In this paper we refer to this method as SS (simple substitution) method to distinguish it from other QBD solution methods. Mitrani and Chakka proposed a one step method based on the spectral expansion of submatrices [4]. While Latouche and Ramaswami proposed another iterative procedure with better numerical properties [3]. Naoumov et al. enhanced this method by reducing the complexity of the iteration steps [5] with a higher memory requirement. These numerical methods are quite recent hence the proper evaluation of their performance is still an open research problem. Some comparison can be found in [4, 3, 2], but we believe that further investigation is needed to provide stable implementations and a sufficient large set of evaluated

*This work was partially supported by OTKA F-23971 and Copernicus 1463.

examples.

The special problem considered in this paper is the effective analysis of QBD Markov chains, when the regular blocks of the QBD matrix has a special sub-block structure due to the fact of fix size batch arrivals. Special block structured Markov chains were already considered in previous works [7, 4], but the class of QBD processes considered in this paper differs from the previous works, because it has a very special sub-block structure, and we give an effective way of solution only for this particular case. Although as the examples show this special case arises often in real networks.

The rest of the paper is organized as follows. In the next Section the detailed description of the problem can be found. In Section 3 we present the proposed algorithm for the introduced problem. In Section 4 a model that suitable for the application of the proposed method is presented, while Section 5 is devoted to the comparison of the proposed algorithm and the algorithm of Naoumov et al., which is one of the best methods for a general QBD process.

2. Quasi-Birth-Death processes

Consider an irreducible, homogeneous Discrete Time Markov Chain where the state of system is described by two random variables: I_n and J_n; I_n is taking its value from $\{0, 1, \ldots, N\}$ and J_n is taking its value from $\{0, 1, \ldots\}$ and the possible one-step transition probabilities are given by the submatrices

- A_j –lateral transition– from state (i, j) to state (k, j), $(i, k \in \{0, 1, \ldots, N\}$, $j \in \{0, 1, \ldots\})$;
- B_j –upward transition– from state (i, j) to state $(k, j+1)$, $(i, k \in \{0, 1, \ldots, N\}$, $j \in \{0, 1, \ldots\})$;
- C_j –downward transition– from state (i, j) to state $(k, j-1)$, $(i, k \in \{0, 1, \ldots, N\}$, $j \in \{1, 2, \ldots\})$;

and all the other transition probabilities equal to 0.

As it is seen from these definition A_j, B_j and C_j are matrices of size $(N+1) \times (N+1)$. This kind of Markov processes is called quasi-birth-death processes since there are transitions only between neighbouring levels (the level is defined by the value of J_n). Assume that an m $(m \geq 1)$ threshold exists such that

- $A_j = A, \ \forall j \geq m$;
- $B_j = B, \ \forall j \geq m-1$;
- $C_j = C, \ \forall j \geq m+1$;

which means that the transition probabilities are level independent if $j \geq m$. The block structure of the transition probability matrix of a QBD process with $m = 3$ is:

$$\Pi = \begin{bmatrix} A_0 & B_0 & & & & \\ C_1 & A_1 & B_1 & & & \\ & C_2 & A_2 & B & & \\ & & C_3 & A & B & \\ & & & C & A & B \\ & & & & \ddots & \ddots & \ddots \end{bmatrix} \tag{1}$$

The left-upper corner submatrix of size $(N+1)m \times (N+1)m$ of the transition probability matrix is referred to as the irregular part and rest as the regular part of the transition probability matrix.

The methods developed for the analysis of the level independent QBD processes, such as the spectral expansion or the SS methods, profit from the regular part and allow any structure in the irregular part including transitions between non-neighbouring levels. The complexity of these methods is $O(L^3)$, where L is the size of submatrices which is $(N+1)$ in this case.

2.1. Processes with batch arrivals

A number of problems in telecommunication networks can be modeled with batch arrivals. A model with fix size batch arrivals and single service differs from the QBD processes only on the upward transitions: the upward transitions are from level j to level $j+K$ instead of $j+1$. In this case the formal description modifies to:

- B_j –upward transition– from state (i, j) to state $(k, j+K)$
- $B_j = B, \ \ \forall j \geq m-K$;

The block structure of the transition probability matrix of a process with batch arrivals ($K = 2$, $m = 3$) is:

$$\Pi = \left[\begin{array}{cc|cc|cc|c} A_0 & 0 & B_0 & & & & \\ C_1 & A_1 & 0 & B & & & \\ \hline & C_2 & A_2 & 0 & B & & \\ & & C_3 & A & 0 & B & \\ \hline & & & C & A & 0 & B \\ & & & & \ddots & \ddots & \ddots \end{array}\right] \tag{2}$$

One of the possible solutions to analyse this system is through the transformation of the process to a QBD process by introducing block size of $K(N+1) \times K(N+1)$. The advantage of block size enlargement is that the general methods of QBD processes can be used, but the disadvantage is the increase of the computation time and storage requirement.

3. A method for fix size batch arrivals

In the description of the algorithm we introduce bold capital letters ($\mathbf{A}$, $\mathbf{B}$ etc.) to denote the matrices of size $(N+1)K \times (N+1)K$ while the matrices of size $(N+1)\times(N+1)$ are denoted by italic letters (A, B etc). The size of the irregular part of the QBD process is denoted by m' and the transition probability submatrices of the QBD process with size of $(N+1)K \times (N+1)K$ are denoted by

- $\mathbf{A}_j$ ($\mathbf{A_j} = \mathbf{A}$, $j \geq m'$) for the lateral transitions,
- $\mathbf{B}_j$ ($\mathbf{B_j} = \mathbf{B}$, $j \geq m'-1$) for the upward transitions and
- $\mathbf{C}_j$ ($\mathbf{C_j} = \mathbf{C}$, $j \geq m'+1$) for the downward transitions.

Denote the steady state distribution of the DTMC by $p_{i,j}$ and introduce vector q_j, $j \geq 0$ as

$$q_j = [p_{0,j}, \ldots, p_{N,j}]$$

and vector v_j, $j \geq 0$ as

$$v_j = [q_{jK}, \ldots, q_{(j+1)K-1}] \ .$$

The steady state distribution can be obtained by solving the following system of equations:

$$\begin{array}{ll} 0 = v_0(\mathbf{A}_0 - \mathbf{I}) + v_1\mathbf{C}_1 & \\ 0 = v_{j-1}\mathbf{B}_{j-1} + v_j(\mathbf{A}_j - \mathbf{I}) + v_{j+1}\mathbf{C}_{j+1} & m' > j > 0 \\ 0 = v_{j-1}\mathbf{B} + v_j(\mathbf{A} - \mathbf{I}) + v_{j+1}\mathbf{C} & j \geq m' \end{array} \tag{3}$$

The solution of (3) can be found as $v_j = v_{m'-1}\mathbf{R}^{j-m'+1}$, $j \geq m'-1$, where $\mathbf{R}$ is the minimal non-zero solution of the following matrix equation [6]:

$$\mathbf{B} + \mathbf{R}(\mathbf{A} - \mathbf{I}) + \mathbf{R}^2\mathbf{C} = 0 \tag{4}$$

and it can be get by the following iteratation (SS method) [6]:

0) $\mathbf{R}_0 = 0$

1) $\mathbf{R}_{n+1} = -\mathbf{B}(\mathbf{A}-\mathbf{I})^{-1} - \mathbf{R}_n^2\mathbf{C}(\mathbf{A}-\mathbf{I})^{-1}$

2) IF ($\|\mathbf{R}_{n+1} - \mathbf{R}_n\| > \epsilon$) GOTO 1

3.1. Modified SS method for fix size batch arrivals

In case of fix size batch arrivals the matrices of the QBD process has a special block structure:

- Lateral transitions: $\mathbf{A} = \begin{bmatrix} A & 0 & 0 & \ldots \\ C & A & 0 & \ldots \\ & \ddots & \ddots & \end{bmatrix}$
- Upward steps: $\mathbf{B} = \begin{bmatrix} B & 0 & \ldots \\ 0 & B & \\ \vdots & & \ddots \end{bmatrix}$
- Downward steps: $\mathbf{C} = \begin{bmatrix} \cdots & 0 & C \\ \cdots & 0 & 0 \\ & & \vdots \end{bmatrix}$

For using the SS method we need to obtain $(\mathbf{A}-\mathbf{I})^{-1}$ that we get as:

$$\begin{bmatrix} A-I & 0 & 0 & \ldots \\ C & A-I & 0 & \ldots \\ 0 & C & A-I & \\ \vdots & & \ddots & \ddots \end{bmatrix}^{-1} = \begin{bmatrix} S & 0 & 0 & \ldots \\ -S(CS) & S & 0 & \ldots \\ S(CS)^2 & -S(CS) & S & \\ \vdots & \ddots & \ddots & \ddots \end{bmatrix} \tag{5}$$

where $S = (A-I)^{-1}$.

As a consequence of the special structure of the matrices and Equation (5) we can get $\mathbf{B}(\mathbf{A}-\mathbf{I})^{-1}$ and $\mathbf{C}(\mathbf{A}-\mathbf{I})^{-1}$ with the following equations:

$$\mathbf{B}(\mathbf{A}-\mathbf{I})^{-1} = \begin{bmatrix} BS & 0 & 0 & \ldots \\ -BS(CS) & BS & 0 & \ldots \\ BS(CS)^2 & -BS(CS) & BS & \\ \vdots & \ddots & \ddots & \ddots \end{bmatrix} \tag{6}$$

$$\mathbf{C}(\mathbf{A}-\mathbf{I})^{-1} = \begin{bmatrix} (CS)^K & -(CS)^{K-1} & \ldots \\ 0 & 0 & \\ \vdots & & \end{bmatrix} \tag{7}$$

Since only the first row of blocks ($(N+1)$ rows) contains non-zero elements of $\mathbf{C}(\mathbf{A}-\mathbf{I})^{-1}$ we only need to calculate the first column of blocks ($(N+1)$ columns) of the $\mathbf{R}_n^2$ matrix:

$$\begin{bmatrix} \sum_k R_n^{(1k)} \cdot R_n^{(k1)} \\ \sum_k R_n^{(2k)} \cdot R_n^{(k1)} \\ \vdots \end{bmatrix} \tag{8}$$

where $R_n^{(ij)}$ are the $(N+1)\times(N+1)$ submatrices of $\mathbf{R}_n$.

Summarizing the results of (6), (7) and (8) we get a modification of SS algorithm for our problem:

0) $\mathbf{R}_0 = \mathbf{0}$

1) $\mathbf{R}_{n+1} = -\mathbf{B}(\mathbf{A}-\mathbf{I})^{-1} - \mathbf{R}_n^2\mathbf{C}(\mathbf{A}-\mathbf{I})^{-1} =$

$$- \begin{bmatrix} BS & 0 & 0 & \ldots \\ -BS(CS) & BS & 0 & \ldots \\ BS(CS)^2 & -BS(CS) & BS & \\ \vdots & \ddots & \ddots & \ddots \end{bmatrix} - \begin{bmatrix} \sum_k R_n^{(1k)} \cdot R_n^{(k1)} \\ \sum_k R_n^{(2k)} \cdot R_n^{(k1)} \\ \vdots \end{bmatrix} \left[\ (CS)^K \ -(CS)^{K-1} \ldots \right]$$

2) IF ($\|\mathbf{R}_{n+1} - \mathbf{R}_n\| > \epsilon$) GOTO 1

The complexity of an iteration step of the original SS method is approximately $O(2K^3(N+1)^3)$. In the modified version the complexity of one iteration step is $O(K^2(N+1)^3)$, since we only have to calculate the first block of column ($(N+1)$ columns) of $\mathbf{R}^2$, and the multiplication of $\mathbf{R}_n^2$ with $\mathbf{C}(\mathbf{A}-\mathbf{I})^{-1}$ becomes a vector-vector multiplication on the $(N+1)\times(N+1)$ block level. Furthermore as a result of (6) and (7) the complexity of the initialization of the iteration is reduced from $7/3K^3(N+1)^3$ to $2K^2(N+1)^3$.

There is a gain in the storage requirements as well. In the original SS method we have to store 4 matrices ($\mathbf{R}_n$, $\mathbf{R}_n^2$, $\mathbf{B}(\mathbf{A}-\mathbf{I})^{-1}$ and $\mathbf{C}(\mathbf{A}-\mathbf{I})^{-1}$) of size $(N+1)K\times(N+1)K$. In the proposed method we only have to store the full $\mathbf{R}_n$ matrix, the first block of columns ($(N+1)$ columns) of $\mathbf{C}(\mathbf{A}-\mathbf{I})^{-1}$ and $\mathbf{R}_n^2$, and the first block of rows ($(N+1)$ rows) of the $\mathbf{C}(\mathbf{A}-\mathbf{I})^{-1}$.

The advantage is significant both in term of computational complexity and storage requirement when K is large and N is small.

4. The system model

Consider a system where identical on-off sources transmit packets on a single output link. The output link works in a slotted manner: there are fix size time slots and in every time slot one data unit can be transmitted. The transmission of a data unit may begin only at the beginning of a time slot. This data unit will be referred to as cell below. Moreover an infinite buffer is assumed at the output link.

The packets arrive independently at the end of the time slots and all of them has the same size. The probability of a packet arrival in a time slot depends on the number of active sources, but only one packet may arrive in a slot (even if there are more than one active sources). The sources may change their states at the end of the time slots independent of the arrivals and the states of the others.

The system has the following parameters:

- S: the number of on-off sources;
- M: the number of cells in a packet;
- α: the probability that an inactive source becomes active;
- β: the probability that an active source becomes inactive;
- $\gamma(i)$: the probability of a packet arrival when i sources are active at the beginning of the time slot.

These assumptions are reasonable if we consider a file server where TCP/IP over ATM is used. The slotted output link has the property of the ATM and packets consisting of fix number of cells is a possible model for large file transfer since most of the IP packets has the maximum transfer unit (MTU) size during bulk transfers [8]. In Ethernet-based networks the MTU of IP datagrams is set to be 1500 byte, therefore the allowed packet size is 32 cells. The default MTU value in IP over ATM environment is chosen to be 9180 byte, and thus the MTU size is 192 cells [1]. The number of active sources may refer to the number of simultaneously active connections.

The system behaviour at the end of the nth time slot is characterized by

- the number of cells in the buffer of the output link (J_n) and
- the number of active sources (I_n).

The stochastic process $\{I_n, J_n\}$ is Discrete Time Markov Chain. We analyse this DTMC to get the steady state distribution of the queue length. Moreover from the steady state distribution performance parameters, like the packet delay distribution, can be obtained.

The buffer content changes as follows:

- If no packet arrives then a cell leaves the buffer if it has not been empty at the beginning of the time slot: $j \to \max(j-1, 0)$
- If packet arrives then it is put in the buffer and a cell leaves the buffer if it has not been empty at the beginning of the time slot: $j \to \max(j-1,0)+M$

The probabilities of the state transitions can be obtained from the parameters of the system model:

$$D_0(i,j) := \Pr(I_{n+1}=j, \text{no packet arrives} | I_n = i) =$$
$$= (1-\gamma(i)) \cdot \sum_{k=\max(0,i-j)}^{\min(i,N-j)} \binom{i}{k}\beta^k(1-\beta)^{i-k} \binom{N-i}{k+j-i}\alpha^{k+j-i}(1-\alpha)^{N-k-j}$$

$$D_1(i,j) := \Pr(I_{n+1}=j, \text{ packet arrives } | I_n = i) =$$
$$= \gamma(i) \cdot \sum_{k=\max(0,i-j)}^{\min(i,N-j)} \binom{i}{k}\beta^k(1-\beta)^{i-k} \binom{N-i}{k+j-i}\alpha^{k+j-i}(1-\alpha)^{N-k-j}$$

where $\bullet(i,j)$ identify a matrix element.

With packets consist of 4 cells ($K=4$) the structure of the transition probability matrix is

$$\begin{bmatrix} D_0 & 0 & 0 & 0 & D_1 & & \\ D_0 & 0 & 0 & 0 & D_1 & & \\ & D_0 & 0 & 0 & 0 & D_1 & \\ & & \ddots & & & & \ddots \end{bmatrix} \qquad (9)$$

This structures corresponds to problem presented in Section 2, where $N=S$, $K=M-1$, $m'=2$, $A=0$, $C=D_0$ and $B=D_1$.

5. Performance comparison

Several papers indicated that the number of iteration steps in the SS method can be extremely high, especially when the utilization of the system (i.e. $\lim_{n\to\infty}\sum_{i=1}^{\infty} Pr(J_n = i)$) approaches to 1 [3, 4, 2].

Latouche and Ramaswami proposed a logarithmic reduction (LR) algorithm in which the number of iteration steps is logarithm (base 2) of the number of steps of the SS method [3]. Experiences with this algorithm show that the number of required iteration steps is not more than 40 for any reasonable example [3, 2].

Naoumov et al. enhanced this algorithm by decreasing the number of operation per iteration from $O(26/3L^3)$ to $O(19/3L^3)$, where L is the block size of the submatrices that is $(N+1)K$ in our case:

0) $\mathbf{N} = \mathbf{A}$; $\mathbf{W} = \mathbf{A}$; $\mathbf{L} = \mathbf{B}$; $\mathbf{M} = \mathbf{C}$

1) $\mathbf{X} = -\mathbf{N}^{-1}\mathbf{L}$; $\mathbf{Y} = -\mathbf{N}^{-1}\mathbf{M}$; $\mathbf{Z} = \mathbf{LY}$;
$\mathbf{W} = \mathbf{W} + \mathbf{Z}$; $\mathbf{N} = \mathbf{N} + \mathbf{Z} + \mathbf{MX}$;
$\mathbf{L} = \mathbf{LX}$; $\mathbf{M} = \mathbf{MY}$

2) IF $(||\mathbf{Z}|| > \epsilon)$ GOTO 1

3) $\mathbf{R} = -\mathbf{B}\mathbf{W}^{-1}$

Since this is one of the best general methods published so far to obtain the steady state distribution of QBD processes with level independent transitions we compare our method to this one.

Both solution methods (the LR method and the one presented in Section 3) have been implemented in C using the Meschach library[1] for matrix operation. The CPU time measurements have been performed on an IBM RISC 6000 570 workstation and the reported CPU time is only the time needed to find the $\mathbf{R}$ matrix. In the experiments we have set the required relative accuracy in both algorithm to $\epsilon = 10^{-10}$, since the results have shown that the changes in the $\mathbf{R}$ matrix are quite marginal if we use smaller required accuracy. The intensity of the sources is specified as $\gamma_i = \min(i \cdot d, 1)$ where d is the parameter of the sources. In our examples d has been chosen to be less than $1/N$.

The influence of the number of users on the computational complexity have been investigated in two scenarios when the packet size has been 32 cells (Figure 1). All parameters were the same in the two scenarios except d. The second scenario, when d is higher, results in a higher system utilization:

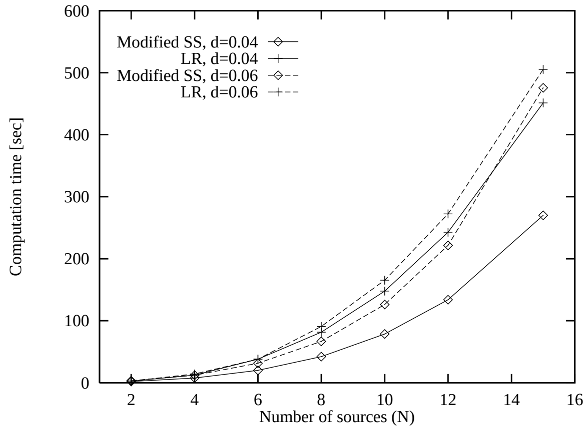

Figure 1. Computation time versus the number of sources

N	K	L	Utilization $d=0.4$	Utilization $d=0.6$
2	31	93	3.7 %	6.3 %
4	31	155	7.5 %	12.7 %
6	31	186	11.4 %	19.2 %
8	31	248	15.3 %	25.9 %
10	31	310	19.3 %	32.8 %
12	31	372	23.5 %	39.9 %
15	31	465	29.8 %	51.0 %

It can be observed that for small N the difference between the algorithms is not significant, but for larger N the proposed algorithm is more efficient.

The sensitivity on the utilization of the system also has been investigated (Figure 2). In the first and the second scenario we used the packet size of 32 with different number of sources, in the third scenario the packet size was increased to 192 cells. The block sizes in the scenarios were the following:

n. of sources (N)	upward step size (K)	block size (L)
4	31	124
10	31	310
2	191	382

The utilization has been increased by the increase the source intensity (d). We have found that the proposed algorithm is very sensitive to the utilization and becomes inefficient compared to the LR algorithm when the utilization converges to 1. This is due to the high number of iteration steps of the SS algorithm again. As it was expected from the theoretical computational complexity the batch size of 192 cells has improved the performance of the proposed algorithm.

We have also studied the impact of the stopping criterion (i.e. the relative accuracy, ϵ) in the case of a low (7.8%) and a high (72.2%) utilization (Figure 3). We used the parameters of the second scenario in this

[1] Meschach library for matrix computation is developed at School of Mathematical Sciences, Australian National University by David E. Stewart and Zbigniew Leyk and it is available via netlib (ftp.netlib.org/c/meschach).

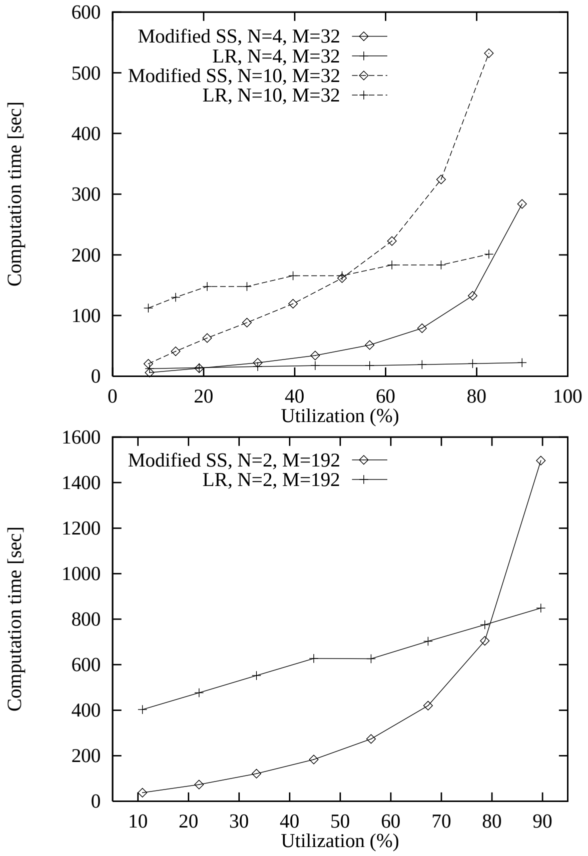

Figure 2. Computation time versus the utilization

experiment. As it was expected, the presented curves show that the proposed algorithm are much more sensitive to the required accuracy than the LR algorithm, since the number of needed iteration steps increases much more in the SS algorithm. As a consequence of this sensitivity we would have got a better image of the proposed algorithm if we had used a larger ϵ in the previous investigations.

6. Conclusion

This paper presented a numerical algorithm to evaluate the steady state distribution of models with fix size batch arrivals and single service. This class of problems results in quasi-birth-death processes with large block size, thus the general processes available for analysis of QBD processes are inefficient. The proposed method profits from the special block structure of the QBD submatrices and reduces the computational complexity and memory requirement of the numerical analysis.

Numerical example of "real" problems, which occurs when IP packages are transmitted through an ATM network, are evaluated. The efficiency of the proposed algorithm is demonstrated by comparing the computation time to an efficient general method (LR) for QBD analysis. The numerical results show that the proposed algorithm is efficient in the case of large size batch arrivals, but it becomes inefficient if the system utilization converges on 1.

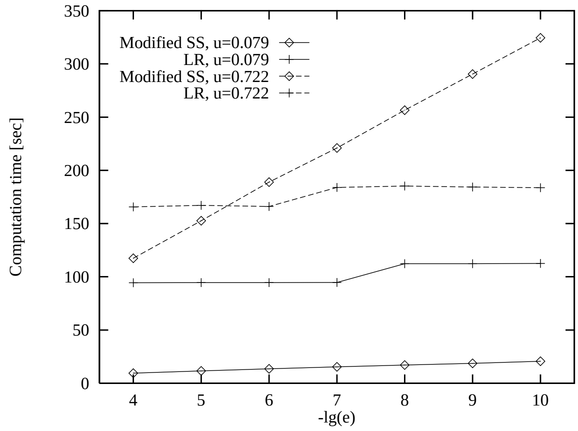

Figure 3. The computation time versus the required accuracy

References

[1] R. Atkinson. *Default IP MTU for use over ATM AAL5, Internet RFC 1626.* Naval Research Laboratory, 1994.

[2] B. Haverkort and A. Ost. Steady state analyses of infinite SPNs: Comparing the Spectral Expansion and the Matrix Geometric method. In *Proc. of PNPM'97*, pages 335–346, Saint Malo, France, 1997.

[3] G. Latouche and V. Ramaswami. A logarithmic reduction algorithm for quasi-birth-death processes. *Journal of Applied Probability*, 30:650–674, 1993.

[4] I. Mitrani and R. Chakka. Spectral expansion solution for a class of Markov models: application and comparison with the matrix-geometric method. *Performance Evaluation*, 23:241–260, 1995.

[5] Naoumov, Krieger, and Wagner. Analysis of a multiserver delay-loss system with a general Markovian arrival process. *Marix-analytic methods in stochastic models*, 1997.

[6] M. F. Neuts. *Matrix Geometric Solutions in Stochastic Model.* Johns Hopkins University Press, Baltimore, 1981.

[7] M. F. Neuts. *Structured Stochastic Matrices of M/G/1 Type and Theeir Matrices.* Marcel Dekker, Inc, 1989.

[8] G. Wolfner, T. Do, and M. Telek. A source model for file transfer applications in local ATM networks. In *Proceedings of 5th IFIP Workshop on Performance Modelling and Evaluation of ATM Networks*, Ilkley (UK), 1997.