Analysis of Queues with Batch Arrivals *

G. Wolfner[1] and M. Telek[2]

[1]Hungarian Telecommunication Company, Telecommunications Development Institute
H-1456 Budapest, PoB 2 – Hungary

[2]Department of Telecommunications, Technical University of Budapest
Sztoczek 2, H-1111 Budapest – Hungary

Email: {wolfner,telek}@hit.bme.hu

Abstract

The steady state distribution of quasi-birth-death processes can be efficiently obtained by matrix-geometric (MG) methods. Since a number of telecommunication problems are modelled by processes with batch arrivals, the extension of MG methods for these processes has practical importance. This paper presents an extension of MG methods which is effective for the analysis of quasi-birth-death processes with batch arrivals. The proposed method is compared with one of the well-known methods.

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 cell payload+header) are the base data units. At the border of these communication protocols the arrival of a 1500 byte packet on the one side requires the transmission of 32 cells on the other side. 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 [8]. 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 (MG) method [8] and is based on an iterative procedure called Simple Substitution (SS) method. Mitrani and Chakka proposed a one step method based on the spectral expansion of submatrices [2, 6]. While Latouche and Ramaswami proposed an other iterative procedure with better numerical properties [5]. Naoumov et al. enhanced this method by reducing the complexity of the iteration steps [7] with a higher memory requirement.

To analyze real communication networks an effective extension of these methods for batch arrivals is necessary. An obvious solution of processes with batch arrivals is to enlarge of the

*This work was supported by the European Community through COPERNICUS'94 contract n. 1463, by the Hungarian OTKA contract n. F-23971 and by the Hungarian Ministry of Culture and Education FKFP-0420/1997 project.

matrix block size in order to obtain a real QBD structure. The price of this block size enlargement is the significant increase of computation complexity and computer storage requirement. Spectral expansion is one of the methods has been developed for batch arrivals and has shown a good performance compared to the others [4]. There is an extension of the SS algorithm for fix size batch arrivals that also shows some advantages in performance and storage requirement [10]. In this paper we propose an alternative method that generally has a better performance both in computation complexity and computer storage requirement than the above mentioned ones. Recently a paper that uses a similair approach for finit state space came to the authers' knowledge. [11].

The rest of the paper is organized as follows. The next Section describes the analyses of the QBD processes. In Section 3 the problem of batch arrivals and the proposed algorithm is introduced. that is an extension of MG solution. Section 4 is devoted to the comparison of the proposed algorithm and the algorithm proposed by Naoumov et al. which seems to be the best methods for a general QBD process. The paper is concluded in Section 5.

2 Analyses of quasi-birth-death processes

Consider a Discrete Time Markov Chain (DTMC) where the state of system is described by two random variables: $Z_n = \{I_n, J_n\}$; I_n is taking its value from $\{1, 2, \ldots, N\}$ and J_n is taking its value from $\{0, 1, \ldots\}$. This DTMC is called a *quasi-birth-death* process if only the state transitions where $J_{n+1} - J_n \in \{-1, 0, 1\}$ have a positive probability. The states with the same value of J_n ($J_n = j$) defines a level (j). By the above definition state transitions are possible inside the levels and between the neighbouring levels.

The nonzero transition probabilities are given by the submatrices

- $A^{(j)}$ (lateral transitions): $A^{(j)}(i,k) = \Pr(I_{n+1} = k, J_{n+1} = j | I_n = i, J_n = j)$[1] ($i, k \in \{1, 2 \ldots, N\}$, $j \in \{0, 1, \ldots\}$);
- $B^{(j)}$ (upward transitions): $B^{(j)}(i,k) = \Pr(I_{n+1} = k, J_{n+1} = j + 1 | I_n = i, J_n = j)$ ($i, k \in \{1, 2, \ldots, N\}$, $j \in \{0, 1, \ldots\}$);
- $C^{(j)}$ (downward transitions): $C^{(j)}(i,k) = \Pr(I_{n+1} = k, J_{n+1} = j - 1 | I_n = i, J_n = j)$ ($i, k \in \{1, 2, \ldots, N\}$, $j \in \{1, 2, \ldots\}$);

and all the other transition probabilities equal to 0.

As it is seen from these definitions $A^{(j)}$, $B^{(j)}$ and $C^{(j)}$ are matrices of size $N \times N$. 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 is shown in Figure 1. The upper-left submatrix of size $(Nm) \times (Nm)$ is referred to as the irregular part of the transition probability matrix and the rest as its regular part.

The methods developed for the analysis of level independent QBD processes, such as the spectral expansion or the MG methods, profit from the structure of the regular part and allow any kind of behaviour in the irregular part including transitions between non-neighbouring levels. Denote the steady state distribution by

$$p_{i,j} = \lim_{n \to \infty} \Pr(I_n = i, J_n = j)$$

[1]$\cdot (i, k)$ denotes the kth element of the ith row of a matrix

$$
\Pi = \begin{bmatrix}
A^{(0)} & B^{(0)} & & & & & \\
C^{(1)} & A^{(1)} & B^{(1)} & & & & \\
 & C^{(2)} & A^{(2)} & B & & & \\
 & & C^{(3)} & A & B & & \\
 & & & C & A & B & \\
 & & & & \ddots & \ddots & \ddots
\end{bmatrix}
$$

Figure 1: The nonzero blocks of the transition probability matrix of a QBD process ($m = 3$)

and introduce vectors v_j $(j \geq 0)$ as

$$v_j = [p_{0,j}, \dots, p_{N,j}].$$

As a consequence of the block structure of the transition probability matrix the steady state distribution of a QBD process can be obtained by solving the following system of vector equations:

$$
\begin{array}{llcll}
(a) & v_0 & = & v_0 A^{(0)} + v_1 C^{(1)} & \\
(b) & v_j & = & v_{j-1} B^{(j-1)} + v_j A^{(j)} + v_{j+1} C^{(j+1)} & m > j > 0 \\
(c) & v_j & = & v_{j-1} B + v_j A + v_{j+1} C & j \geq m \\
(d) & 1 & = & \sum_{j=0}^{\infty} v_j e_N^T &
\end{array}
\tag{1}
$$

where e_N^T is the column vector of 1s of size N.

The methods developed for the steady-state analysis are based on the fact [8] that $\forall j \geq m-1$

$$v_{j+1} = v_j R, \tag{2}$$

where R is a matrix of size $N \times N$, and it is the minimal non-negative solution of the matrix equation

$$B + RA + R^2 C = R. \tag{3}$$

A simple way to evaluate R is to apply the SS algorithm whose iteration step is shown in Figure 2 [8]. The sequence of R_n $(n = 0, 1, \dots)$ is entry-wise nondecreasing and it converges to matrix R [8].

$$
\begin{array}{l}
R_0 = 0 \\
n = 0 \\
DO \\
\quad R_{n+1} = B + R_n A + R_n^2 C \\
\quad n = n + 1 \\
WHILE\ (||R_n - R_{n-1}|| \geq \epsilon)
\end{array}
$$

Figure 2: The simple substitution (SS) algorithm

The disadvantage of the method is the slow convergence, especially when the utilization of the modelled system approaches 1. Latouche et Ramaswami proposed another iterative method, a Logarithmic Reduction (LR) algorithm, that converges much faster [5].

Recently Naoumov et al. proposed an enhancement of the LR algorithm (Figure 3), that requires less operations per iteration [7]. An iteration step of the LR method is more complicated than a step of the SS algorithm, but the fewer iterations makes the LR algorithm faster. The experiences so far have shown that usually the number of needed iteration steps are less than 20 [4, 5, 10].

Mitrani and Chakka proposed a direct method, called spectral expansion [2, 6]. In this method the least N eigenvalues and the associated eigenvectors must be obtained from

$$\lambda \phi = \phi \left(B + \lambda A + \lambda^2 C \right),$$

$N = A - I$
$L = B$
$M = C$
$W = A - I$
DO
  $X = -N^{-1}L$
  $Y = -N^{-1}M$
  $Z = LY$
  $W = W + Z$
  $N = N + Z + MX$
  $L = LX$
  $M = MY$
$WHILE\ (||Z|| \geq \epsilon)$
$R = -BW^{-1}$

Figure 3: The logarithmic reduction (LR) algorithm of Naoumov et al.

where λ is a complex number and ϕ is a vector of N complex elements.

The advantage of this method is the direct solution and the easy calculation of the state probabilities based on the eigenvalues and the eigenvectors, but numerical problems can arise by the close eigenvalues and eigenvectors.

The detailed comparison of this method and the algorithm of Naoumov et al has not published yet. Since the theoretical complexity of the algorithms are similar ($O(N^3)$) the difference between the algorithms can be shown by experiences. The previously published results show a better performance of spectral expansion, especially when the utilisation is near 1, but spectral expansion may lead inaccurate results because of its numerical problems.

3 The extension of MG approach for batch arrivals

3.1 Processes with batch arrivals

Consider the DTMC $Z_n = \{I_n, J_n\}$, where I_n is taking its value from $\{1, 2, \ldots, N\}$ and J_n is taking its value from $\{0, 1, \ldots\}$, as in the previous section. Models with batch arrivals and single server differs from QBD processes only by the allowed upward transitions. Now, upward transitions are allowed from level j to level $j + l$ ($l = 1, 2, \ldots, y$), where y is the maximum batch size.

In this case the nonzero transition probabilities are given by the submatrices

- $A^{(j)}$ (lateral transitions): $A^{(j)}(i, k) = \Pr(I_{n+1} = k, J_{n+1} = j | I_n = i, J_n = j)$ ($i, k \in \{1, 2, \ldots, N\}$, $j \in \{0, 1, \ldots\}$);
- $B_l^{(j)}$ (upward transitions): $B^{(j)}(i, k) = \Pr(I_{n+1} = k, J_{n+1} = j + 1 | I_n = i, J_n = j)$ ($i, k \in \{1, 2, \ldots, N\}$, $j \in \{0, 1, \ldots\}$ and $l \in \{0, 1, \ldots, y\}$);
- $C^{(j)}$ (downward transitions): $C^{(j)}(i, k) = \Pr(I_{n+1} = k, J_{n+1} = j - 1 | I_n = i, J_n = j)$ ($i, k \in \{1, 2, \ldots, N\}$, $j \in \{1, 2, \ldots\}$);

and all the other transition probabilities equal to 0.

Assume that an m ($m \geq y$) threshold exists such that

- $A^{(j)} = A, \ \forall j \geq m$;
- $B^{(j)} = B, \ \forall j \geq m - y$;
- $C^{(j)} = C, \ \forall j \geq m + 1$;

which means that the transition probabilities are level independent if $j \geq m$. The block structure of the transition probability matrix of a process with batch arrivals is shown in Figure 4 (where $y = 3$ and $m = 6$).

$$
\Pi = \left[\begin{array}{ccc|ccc|ccc|ccc}
A^{(0)} & B_1^{(0)} & B_2^{(0)} & B_3^{(0)} & & & & & & & & \\
C^{(1)} & A^{(1)} & B_1^{(1)} & B_2^{(1)} & B_3^{(1)} & & & & & & & \\
 & C^{(2)} & A^{(2)} & B_1^{(2)} & B_2^{(2)} & B_3^{(2)} & & & & & & \\
\hline
 & & C^{(3)} & A^{(3)} & B_1^{(3)} & B_2^{(3)} & B_3 & & & & & \\
 & & & C^{(4)} & A^{(4)} & B_1^{(4)} & B_2 & B_3 & & & & \\
 & & & & C^{(5)} & A^{(5)} & B_1 & B_2 & B_3 & & & \\
\hline
 & & & & & C^{(6)} & A & B_1 & B_2 & B_3 & & \\
 & & & & & & C & A & B_1 & B_2 & B_3 & \\
 & & & & & & & \ddots & \ddots & \ddots & \ddots & \ddots
\end{array}\right]
$$

Figure 4: The block structure of the transition probability matrix of a process with batch arrivals ($y = 3$, $m = 6$)

To obtain the steady state distribution the following system of equations must be solved:

$$
\begin{array}{lccll}
(a) & v_j & = & \sum_{i=0}^{j-1} v_i B_{j-i}^{(i)} + v_j A^{(j)} + v_{j+1} C^{(j+1)} & j < y \\
(b) & v_j & = & \sum_{i=1}^{y} v_{j-i} B_i^{(j-i)} + v_j A^{(j)} + v_{j+1} C^{(j+1)} & m > j \geq y \\
(c) & v_j & = & \sum_{i=1}^{y} v_{j-i} B_i + v_j A + v_{j+1} C & j \geq m \\
(d) & 1 & = & \sum_{j=0}^{\infty} v_j e_N^T &
\end{array} \tag{4}
$$

3.2 Analysis by block size enlargement

A possible solution to analyse systems of this kind is the transformation of the problem into a QBD process by introducing blocks of size $yN \times yN$ (Figure 4) [4]. The advantage of this block size enlargement is that the standard QBD methods can be used, but its disadvantage is the increase in computation complexity ($O(y^3 N^3)$) and storage requirement ($O(y^2 N^2)$).

The spectral expansion method has been extended to analyse systems with multi-level jumps without block size enlargement [2, 6]. Although the computation complexity of solving the spectral decomposition is similar to the block enlargement ($O(y^3 N^3)$) numerical experiences shows a better performance of this method [4]. But the aforementioned numerical instability remains its main problem.

3.3 A level-block-size method

Here we propose a method, which is more effective than the block size enlargement both in computation complexity and in storage requirement and does not have the numerical problems of spectral expansion. The proposed method is an extension of the SS method for processes with batch arrivals.

If the MG methods (either the simple substitution or the logarithmic reduction algorithm) with block size enlargement are used then matrices of size $yN \times yN$ are treated. As a result of the MG methods an $\mathbf{R}$ matrix of size $yN \times yN$ is obtained. Denote the $N \times N$ submatrices of this $\mathbf{R}$ as

$$
\mathbf{R} = \begin{bmatrix} R_{1,1} & \cdots & R_{1,y} \\ \vdots & & \vdots \\ R_{y,1} & \cdots & R_{y,y} \end{bmatrix}.
$$

The extension of the SS method is based on the following important theorem:

Theorem 3.1 *If the steady state distribution of the process with batch arrival exists then*

$$v_j = \sum_{i=0}^{y-1} v_{j-y+i} T_i, \quad \forall j \geq m,$$

where $T_i = R_{i+1,1}$.

Proof: Consider the process as a QBD process with block size enlargement. Equation (2) holds for the first yN size vector of the regular part:

$$[v_m, \ldots, v_{m+y-1}] = [v_{m-y}, \ldots, v_{m-1}]\mathbf{R},$$

i.e., the theorem holds for $j = m$.

Now, assuming that the size of the irregular part is $m' = m + l, \ l \geq 0$ the regular part of the process remains the same, and so does $\mathbf{R}$. In this case Equation (2) gives:

$$[v_{m+l}, \ldots, v_{m+l+y-1}] = [v_{m+l-y}, \ldots, v_{m+l-1}]\mathbf{R}.$$

Since this equation is satisfied $\forall \ j = l \geq 0$ the theorem is proved. □

The main consequence of the theorem is that the first N columns of $\mathbf{R}$ ($T_i, \ i = 0, \ldots y-1$) contains sufficient information to determine the steady state distribution of the process. The following two theorems allow to obtain the T_i ($i = 0, \ldots y-1$) matrices.

Theorem 3.2 *The* $T_i, i = 0, \ldots, y-1$ *matrices are the minimal nonnegative solutions of the following system of matrix equations:*

$$\begin{aligned} T_0 &= B_y + T_0(A + T_{y-1}C) \\ T_i &= B_{y-i} + T_i(A + T_{y-1}C) + T_{i-1}C \qquad i = 1, \ldots, y-1 \end{aligned} \tag{5}$$

Proof: Applying Theorem 3.1 for the left hand side of Equations (4c) we have:

$$v_j \ = \ \textstyle\sum_{i=0}^{y-1} v_{j-y+i} T_i \tag{6}$$

and applying it for the right hand side of Equations (4c) we have:

$$\begin{aligned} &\textstyle\sum_{i=1}^{y} v_{j-i}B_i + v_j A + v_{j+1}C = \sum_{i=1}^{y} v_{j-i}B_i + v_j A + \left(\sum_{i=0}^{y-1} v_{j+1-y+i}T_i\right) C = \\ &= \textstyle\sum_{i=1}^{y} v_{j-i}B_i + v_j A + \left(\sum_{i=0}^{y-2} v_{j+1-y+i}T_i\right) C + v_j T_{y-1} C = \\ &= \textstyle\sum_{i=1}^{y} v_{j-i}B_i + v_j(A + T_{y-1}C) + \left(\sum_{i=0}^{y-2} v_{j+1-y+i}T_i\right) C = \\ &= \textstyle\sum_{i=0}^{y-1} v_{j-y+i}B_{y-i} + \left(\sum_{i=0}^{y-1} v_{j-y+i}T_i\right)(A + T_{y-1}C) + \left(\sum_{i=1}^{y-1} v_{j-y+i}T_{i-1}\right) C = \\ &= v_{j-y}\left(B_y + T_0(A + T_{y-1}C)\right) + \textstyle\sum_{i=1}^{y-1} v_{j-y+i}\left(B_{y-i} + T_i(A + T_{y-1}C) + T_{i-1}C\right) \end{aligned} \tag{7}$$

The theorem comes from the equality of the coefficients of $v_{j-y+i}, \ i = 0, 1, \ldots, y-1$ in Equation (6) and (7). □

Theorem 3.3 *If* $X_i^{(0)} = 0, \ i = 0, 1, \ldots, y-1$ *then the iteration*

$$\begin{aligned} X_0^{(n+1)} &= B_y + X_0^{(n)}(A + X_{y-1}^{(n)}C) \\ X_i^{(n+1)} &= B_{y-i} + X_i^{(n)}(A + X_{y-1}^{(n)}C) + X_{i-1}^{(n)}C \qquad i = 1, \ldots, y-1 \end{aligned}$$

converges on the minimal non-negative solutions of Equation (5).

Proof: This algorithm is the adaptation of the SS algorithm for our case. The convergence of the algorithm can be established in the same way as it is in [8]: first it is proved that the sequences of X_i^n are entry-wise nondecreasing then the convergence is verified.

Since the B_y, A and C matrices consist of nonnegative elements, $X_i^{(1)} \geq X_i^{(0)} = 0,\ i = 0,\dots,y-1$ entry-wise. The increase of $X_i^{(n+1)},\ i = 0,\dots,y-1,\ n \geq 1$ can be proved by induction:

$$X_0^{(n+1)} = B_y + X_0^{(n)}(A + X_{y-1}^{(n)}C) \ \geq\ B_y + X_0^{(n-1)}(A + X_{y-1}^{(n-1)}C) = X_0^{(n)}$$

and

$$\begin{aligned} X_i^{(n+1)} &= B_{y-i} + X_i^{(n)}(A + X_{y-1}^{(n)}C) + X_{i-1}^{(n)}C \ \geq \\ &\geq\ B_{y-i} + X_i^{(n-1)}(A + X_{y-1}^{(n-1)}C) + X_{i-1}^{(n-1)}C = X_i^{(n)} \qquad i = 1,\dots,y-1 \end{aligned}$$

$X_i^{(0)} \leq T_i,\ i = 0,\dots,y-1$ entry-wise and the $X_i^{(n)} \leq T_i,\ i = 0,\dots,y-1,\ n \geq 0$ also can be verified by induction:

$$X_0^{(n+1)} = B_y + X_0^{(n)}(A + X_{y-1}^{(n)}C) \ \leq\ B_y + T_0(A + T_{y-1}C) = T_0$$

and

$$\begin{aligned} X_i^{(n+1)} &= B_{y-i} + X_i^{(n)}(A + X_{y-1}^{(n)}C) + X_{i-1}^{(n)}C \ \leq \\ &\leq\ B_{y-i} + T_i(A + T_{y-1}C) + T_{i-1}C = T_i \qquad i = 1,\dots,y-1 \end{aligned}$$

Since an upper-bounded monotone increasing sequence must converge, the sequences $X_i^{(n)},\ i = 0,\dots,y-1$ converge entry-wise. The limit matrices satisfy Equation (5) and they are not greater than the minimal nonnegative solutions, thus sequences of $X_i^{(n)},\ i = 0,\dots,y-1$ converge on the minimal non-negative solutions of Equation (5). □

As the result of Theorem 3.3 the algorithm in Figure 5 can be used to obtain the $T_i,\ i = 0,\dots,y-1$ matrices. The complexity of one iteration step of the algorithm is $O(y\ N^3)$ that is significantly better than the complexity of the other mentioned methods.

```
FOR i = 0 TO y − 1
    T_i^(0) = 0
ENDFOR
n = 0
DO
    T_0^(n+1) = B_y + T_0^(n) (A + T_{y−1}^(n) C)
    FOR i = 1 TO y − 1
        T_i^(n+1) = B_{y−i} + T_i^(n) (A + T_{y−1}^(n) C) + T_{i−1}^(n+1) C
    ENDFOR
    n = n + 1
WHILE (max_i(||T_i^(n) − T_i^(n−1)|| ≥ ε)
```

Figure 5: The proposed numerical method to obtain the T_i matrices

3.4 The steady state distribution

When the $T_i,\ i = 0,\dots,y-1$ matrices are known only the vectors $v_0, v_1, \dots, v_{m-1}$ miss to determine the steady state distribution of the process. Equations (4a), (4b) and (4d) can be used to obtain these unknowns. The number of unknowns is mN and the number of linearly independent equations is the same. Since $v_m = \sum_{i=0}^{y-1} v_{m-y+i} T_i$ (see Theorem 3.1) the unknowns in Equation (4a) and (4b) are $v_0, v_1, \dots, v_{m-1}$. The infinite sum in Equation (4d) can be resolved by the following theorem:

Theorem 3.4

$$\sum_{j=m}^{\infty} v_j = \sum_{i=0}^{y-1} v_{m-y+i} \left(\sum_{n=0}^{i} T_n \left(I - \sum_{l=0}^{y-1} T_l \right)^{-1} \right)$$

Proof: Let $s = \sum_{j=m}^{\infty} v_j$ and make the following transformations:

$$s = \sum_{j=m}^{\infty} v_j = \sum_{i=0}^{y-1} \sum_{j=0}^{\infty} v_{m+jy+i} = \sum_{i=0}^{y-1} \sum_{j=0}^{\infty} \sum_{n=0}^{y-1} v_{m+(j-1)y+i+n} T_n = \sum_{n=0}^{y-1} \left(\sum_{i=0}^{y-1} \sum_{j=0}^{\infty} v_{m+(j-1)y+i+n} \right) T_n = \sum_{n=0}^{y-1} \left(\sum_{i=n}^{y-1} v_{m-y+i} + s \right) T_n$$

As of consequence of these the Theorem comes as:

$$\begin{aligned} s &= \sum_{n=0}^{y-1} \left(\sum_{i=n}^{y-1} v_{m-y+i} + s \right) T_n \\ s \left(I - \sum_{l=0}^{y-1} T_l \right) &= \sum_{i=0}^{y-1} v_{m-y+i} \left(\sum_{n=0}^{i} T_n \right) \\ s &= \sum_{i=0}^{y-1} v_{m-y+i} \left(\sum_{n=0}^{i} T_n \left(I - \sum_{l=0}^{y-1} T_l \right)^{-1} \right) \quad \square \end{aligned}$$

By this theorem we have a system of equations with the same number unknowns and independent equations. The result of the system of equations is the $v_0, v_1, \ldots, v_{m-1}$ vectors, thus the steady state distribution is obtained.

3.5 Continuous time processes

So far the discrete time Markov chains has been discussed, but the results can be applied to continuous time Markov chains (CTMC) as well. A simple way to do so is the application of the method of randomization, that produces a DTMC from a CTMC with the same steady state distribution. Let Q be the generator matrix of the CTMC, and $q = max_{i,j}|Q(i,j)|$. The DTMC with transition probability matrix

$$\Pi = Q/q + I,$$

where the division means division of all entry of the matrix and I is the identity matrix with the appropriate dimension has the same steady state distribution [3].

4 Performance comparison

4.1 The system model

A simple queueing system has been evaluated to investigate the performance of the proposed method. A system with a Markov modulated source is considered. The source transmits packets to an output link. The output link works in a slotted manner: there are fix size time slots and in every time slot at most one data unit can be transmitted. The transmission of a data begins at the beginning of a time slot. We refer to data units as cells below. An infinite buffer is assumed at the output link.

The source submits at most one packet at the end of the time slots and all of these packets have the same size. The probability of a packet arrival in a time slot depends on phase of the Markov modulated source. The source may change its phase at the end of the time slots independent of packet arrivals.

These assumptions are realistic considering a file server where TCP/IP over ATM is used. The slotted output link has the properties of ATM and packets consisting of a fix number of cells is a possible model for large file transfers since most of the IP packets has the size of maximum transfer unit (MTU) during bulk transfer [9]. For example in Ethernet-based networks the MTU of an IP datagram is 1500 bytes, therefore the maximum 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 Markov modulated source represents a phase dependent arrival, e.g., the phase refers to the number of simultaneously active connections or arrivals are according to a renewal process with phase-type distributed interarrival times.

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 phase of the source (I_n).

The system has the following parameters:

- C: the number of phases of the source;
- r: the number of cells in a packets;
- $D_0(i,k) = \Pr(I_{n+1} = k,$ no message arrives $|I_n = i)$, D_0 is a matrix of size $C \times C$
- $D_1(i,k) = \Pr(I_{n+1} = k,$ message arrives $|I_n = i)$, D_1 is a matrix of size $C \times C$

The stochastic process $\{I_n, J_n\}$ is a DTMC. From the steady state distribution of this DTMC the the queue length distribution and the packet delay distribution can be obtained. The state transition of the system are as follows:

- If no packet arrives then a cell leaves the buffer, if it was not empty, at the beginning of the time slot, and the source has a phase transition from phase i to k:

  $$(i,j) \to (k, \max(j-1, 0)) \tag{8}$$

  The probability of this state transition is $D_0(i,k)$.

- If packet arrives then it is stored in the buffer and a cell leaves the buffer, if it was not empty, at the beginning of the time slot, and the source has a phase transition from phase i to k:

  $$(i,j) \to (k, \max(j-1, 0) + r) \tag{9}$$

  The probability of this state transition is $D_1(i,k)$.

As a consequence of (8) and (9) the block structure of the transition probability matrix is as in Figure 6. This structure corresponds to the problem presented in Section 3.

$$\begin{bmatrix} D_0 & 0 & 0 & 0 & D_1 & & \\ D_0 & 0 & 0 & 0 & D_1 & & \\ & D_0 & 0 & 0 & 0 & D_1 & \\ & & \ddots & & & & \ddots \end{bmatrix}$$

Figure 6: The block structure of transition probability matrix ($r = 4$)

4.2 Numerical results

We have compared the proposed method (referred to as Level-Block-Size (LBS) method) to the method proposed by Naoumov et al. (see Figure 3, referred to as LR method), since the LR method is one of the best among the published general methods for the steady state analysis of level independent QBD processes.

Both methods have been implemented in C using the Meschach library[2] for matrix operation. The CPU time measurements have been performed on a PC with Intel Pentium processor, using

[2]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).

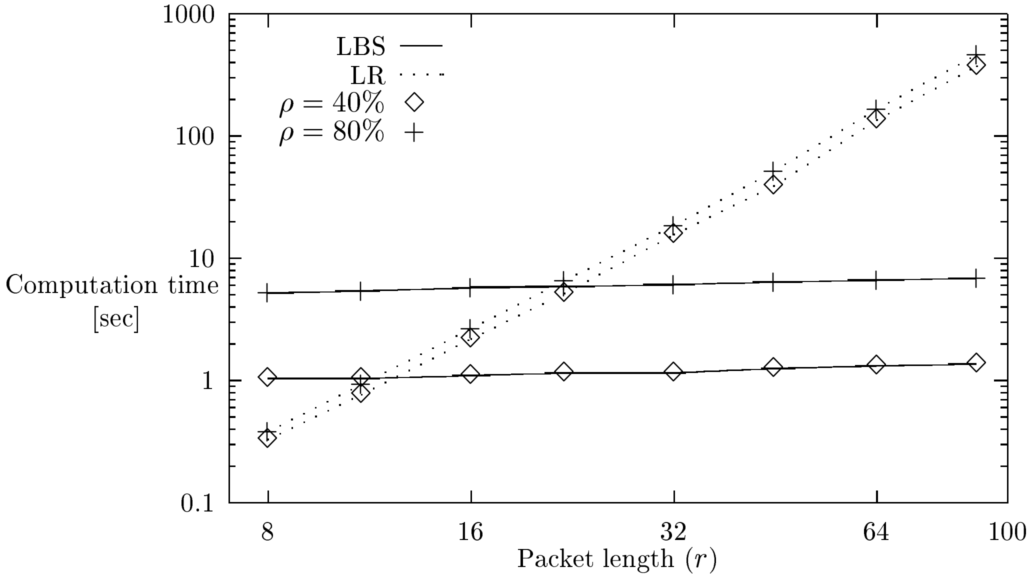

Figure 7: Computation time versus the packet length ($C = 5$ and $\rho = 40,\ 80\%$)

Utilization	Number of iteration step ($C = 5,\ r = 16$)		Number of iteration step ($C = 5,\ r = 32$)	
	LR	MG	LR	MG
20%	10	440	9	241
40%	11	1203	10	642
60%	13	2601	12	1374
80%	14	6341	13	3345
90%	15	13087	14	6917
95%	16	25370	15	13445
97%	17	40471	16	21499

Table 1: The number of iteration step

MSDOS and GNU C compiler. The reported CPU time includes only the time needed to obtain the $\mathbf{R}$ matrix in the LR algorithm and the T_i matrices in the proposed LBS method. In the experiments the required relative accuracy (ϵ) was set to 10^{-10} in both algorithms. The results show that the differences in the obtained steady state distribution are marginal at this relative accuracy.

First the influence of the batch size, that is related to the packet length ($y = r - 1$), to the computation time has been investigated (Figure 7). In these experiments the number of phases of the source (C) was 5. We have compared the performance of these methods with two different system utilization parameters (40% and 80%). The packet length has only a little influence on the computation time with the proposed method, but it has a significant impact with the LR method which uses block size enlargement. This difference can be explained by the complexity of the algorithms, the complexity of LR algorithm is $O(r^3N^3)$, while the complexity of the proposed method is $O(rN^3)$.

In Figure 7 it can be observed that the system utilization influenced the computation time as well, thus we investigated the effect of system utilization. We have found that the proposed algorithm is very sensitive to the utilization and becomes inefficient comparing to the LR algorithm when utilization converges on 1 (Figure 8). This can be explained by the high number of iteration steps of the LBS algorithm (Table 1) and this fact corresponds to the previous results with the SS algorithm [2, 6]. The utilization level when the LR algorithm becomes more efficient strongly depends on the maximum batch size.

We then investigated The impact of the number of phases of the source (C) is depicted

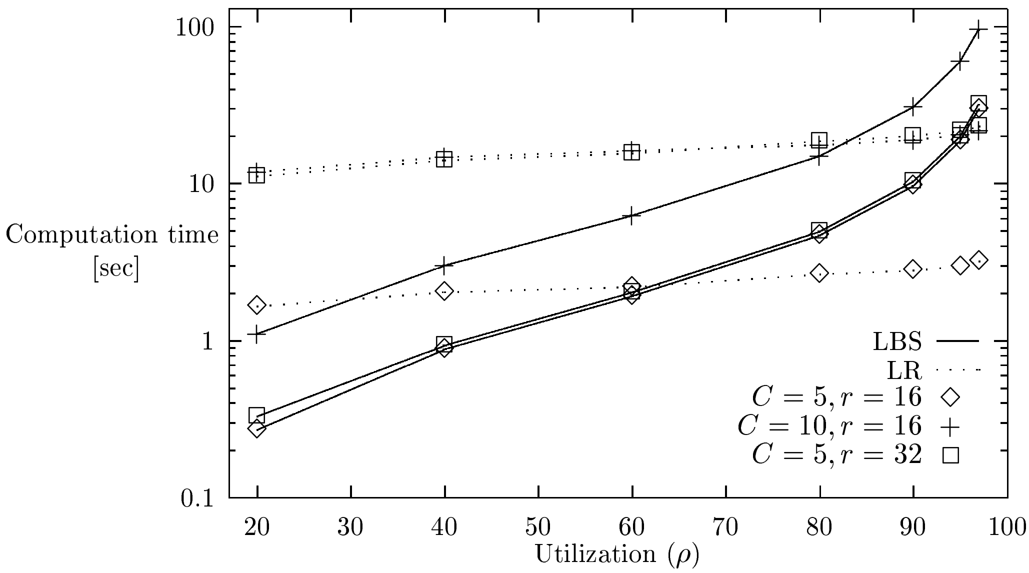

Figure 8: Computation time versus the utilization ($C = 5$ and $r = 16,\ 32$)

in Figure 9. For this experiment the message length (r) was 16, i.e., the batch size was 15. The behaviour of the algorithms are quite similar, although the proposed algorithm seems more efficient when the number of phases of the source is higher. The larger memory requirement of the LR algorithm may cause this phenomenon.

Our last investigation was the sensitivity of the algorithms on the required accuracy (stoping criteria, ϵ). Figure 10 shows results with a network utilisation of 60%. The curves in the figure show that the required accuracy has only a little influence of the computation time of the LR algorithm, while computation time of the LBS method is strongly depends on it. In the previously presented results we used a strict required accuracy, so the results would show a better performance of the LBS algorithm in the case of less strict required accuracy.

5 Conclusions

In this paper an extension of the MG approach for processes with batch arrivals has been presented and a numerical method has been proposed to obtain the steady state behaviour. A process of this kind can be analysed as a QBD process with a larger block size, thus the algorithms available for the analysis of QBD processes are inefficient. We proposed a method that performs the computation without block size enlargement. The proposed approach reduces the computation complexity and the memory requirement of the numerical analysis.

A performance comparison of the proposed method with an efficient general method (LR) has also been presented. The results show that the proposed method is efficient in case of large batch size, but it becomes inefficient if the system utilization converges on 1.

References

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

[2] Ram Chakka. *Performance and Reliability Modelling of Computing Systems Using Spectral Expansion*. PhD thesis, University of Newcastle upon Tyne, 1995.

[3] W. Feller. *An Introduction to Probability Theory and Its Application*, volume 2. John Wiley & Sons, 1971. 2nd edition.

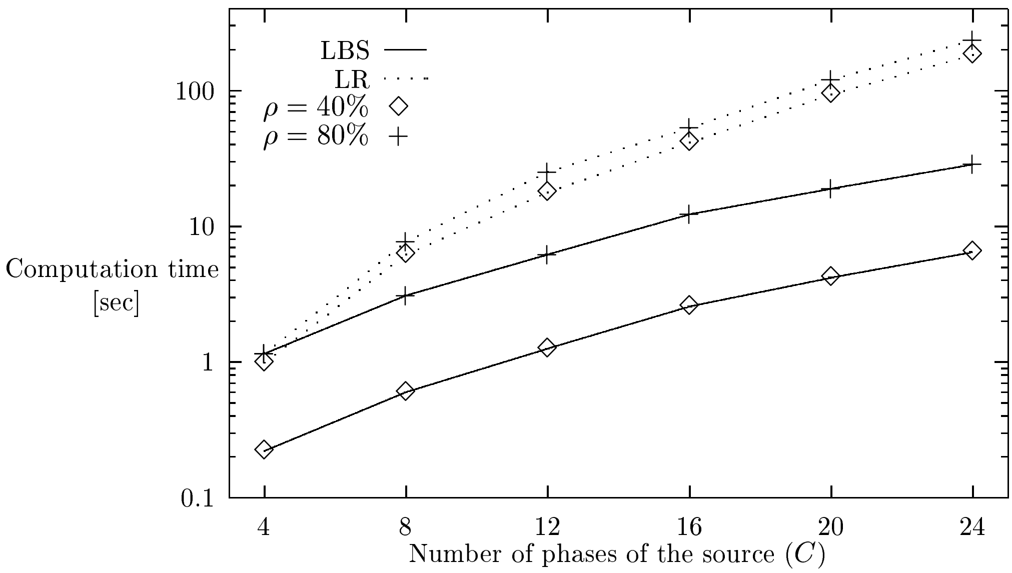

Figure 9: Computation time versus the number of phases of the source

[4] B. Haverkort and A. Ost. Steady state analyses of infinite stochastic petri nets: A comparing between the Spectral Expansion and the Matrix Geometric method. In *Proceedings of 7th International Workshop on Petri Nets and Performance Models*, pages 335–346, Saint Malo, France, 1997.

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

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

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

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

[9] G. Wolfner, T.V. 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.

[10] G. Wolfner and M. Telek. A numerical analysis method of queues with batch arrivals. In *Proceedings of Fifth International Conference on Advanced Computing*, Chennai (Madras), India, 1997.

[11] K. Wuyts and R.K. Boel. A matrix geometric algorithm for finite buffer systems with B-ISDN applications. In *Proceedings of the ITC Specialists Seminar on Control in Communications*, pages 265–276, Lund, Sweden, September 1996.

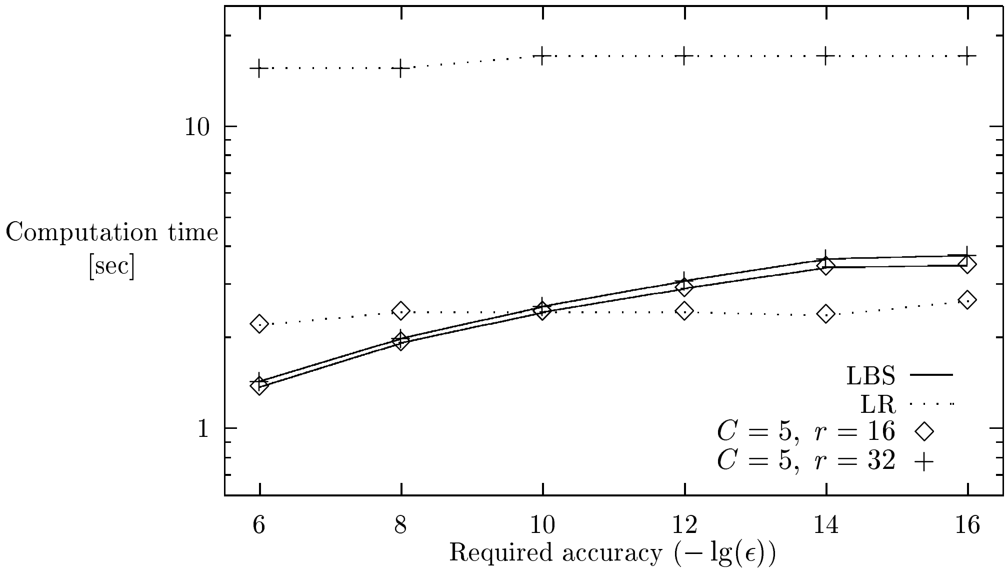

Figure 10: Computation time versus the accuracy