



Research article

Dual core processors: Coupled queues: Transient performance evaluation

Shaik Salma^{*}, Garimella Ramamurthy*Department of CSE, Mahindra University, Bahadurpalli, 500043, India*

ARTICLE INFO

Keywords:

Coupled Queues
Markov Chain
Matrix Exponential
Transient Analysis
Multi-core Processor

ABSTRACT

High-Performance Computing (HiPC) systems routinely employ multi/many – core processors. Specifically, dual – core processors find many applications in pervasive computing devices. Dual-core processors employ buffers for queueing the incoming jobs. Traditionally, the queues at the processors are assumed to be independent and the queueing system is analyzed in equilibrium for tractability purposes. Queues are modeled using Continuous Time Markov Chains (CTMC's) and the equilibrium performance measures are determined to analyze as well as design the queueing systems. In most interesting cases, the incoming jobs are routed to the queues using the Join the Shortest Queue (JSQ) policy. Thus, with such an adaptive routing algorithm, the two queues are evidently coupled and are not statistically independent. Hence traditional equilibrium performance evaluation doesn't provide realistic performance measures. In this research paper, the two queues associated with buffers in dual-core processors are considered to be coupled. The Coupled Queues are modeled using a Quasi – Birth – and – Death (QBD) process. Using traditional results related to QBD processes, equilibrium performance measures are determined. More interestingly, we demonstrate the tractability of the computation of transient probability distribution of a QBD process. In the research literature, transient analysis of the QBD process was shown to be tractable in the Laplace transform domain. But in this research paper, we prove that the matrix exponential e^{Qt} arising in transient analysis (where Q is the generator matrix of the QBD process) can be computed directly in the time domain rendering efficient transient analysis of QBD process. Using the transient probability mass function of queue length, estimation of transient performance measures such as expected queue length, average delay, and tail distribution can be determined. Further, optimal adaptive routing algorithms for coupled queues can be designed.

1. Introduction

Computing Systems based on a single processor were utilized in many applications. Multi-Processor (especially multiple cores on a single die) based computing systems provided parallel processing capability [11], thereby enabling high-performance computing. Application-specific processors such as graphics processing units (GPUs) enabled the deployment of versatile computer graphics-

^{*} Corresponding author.

E-mail addresses: salma20pcse004@mahindrauniversity.edu.in (S. Salma), rama.murthy@mahindrauniversity.edu.in (G. Ramamurthy).

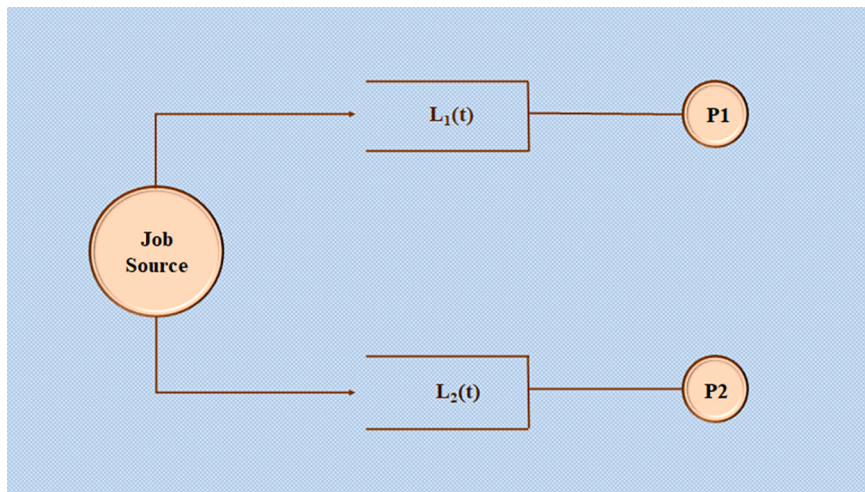


Fig. 1. Coupled queues that are statistically dependent.

based applications. Such hardware accelerators (with hundreds of GPUs) were utilized for general-purpose computing tasks (namely GP GPUs) also.

In most computing platforms (such as multicore processors), buffers are provided to queue the incoming jobs [16]. These buffers could be implemented in software/hardware and the finite buffer size assumption is more reasonable. Specifically, consider a dual-core processor with two buffers and let the incoming computing jobs be routed to the buffers, as shown in Fig. 1.

Most stochastic models of the queueing system assume that the queues are independent. But, most interesting routing algorithms should utilize the queue state information (queue size) for routing the packets to the buffers. Thus, the two queues are not statistically independent and are coupled.

In this research paper, motivated by the application to multicore processors [14], we derive a tractable stochastic model, when there are two queues (The generalization to more than two queues is straightforward) that are coupled.

Strengths:

Traditionally, transient analysis of the QBD process was shown to be tractable [5], [10] in the Laplace transform domain. It requires the additional step of inverse Laplace transform compared to e^{Qt} computation (8) (where Q is the generator matrix).

In this paper, an analytical approach to truncate the matrix power series for e^{Qt} is provided. This approach works for an arbitrary matrix Q (not just a generator matrix). The approach for transient analysis potentially enables real-time computation of transient performance measures of adaptive routing schemes.

2. Review of related research literature

In communication networks, to increase throughput and reduce time delay, various adaptive routing, flow control, and congestion control schemes have been proposed [1]. Specifically, in the context of adaptive routing schemes, direct or indirect knowledge of queue lengths is utilized for the flow control of packets. Traditionally Join-the-Shortest Queue is a popular scheme in which packets are routed to the shortest of two queues (picking one of the two alternate paths). Equilibrium performance evaluation of JSQ policy based on stochastic models was proposed in [2–4]. Optimality of shortest queue routing was established if the length of the queues was continuously monitored.

In optimal, and advanced routing algorithms, the queue length information has to be passed from node to node introducing considerable communication and processing overhead. By transmitting the queue length information infrequently, communication overhead can be reduced. But the information available is out-of-date. Hence, there is a need to make intelligent use of delayed information in adaptive routing algorithms. In order to implement adaptive routing, it is useful to develop a model which enables the prediction of queue length, given an estimate at a prior time. For the purposes of tractability average queue length information conditioned on prior information should be estimated.

In [9], considering the queues to be decoupled and with finite buffer size, an exact closed-form expression for the evolution of queue length distribution is derived.

In [6], we proposed a Quasi-Birth-and-Death (QBD) process model of coupled queues arising in communication networks.

In the following discussion, we provide chronological development of equilibrium and transient analysis of QBD Processes. We explain the motivation for the current approach.

BACKGROUND: Skip-free Markov Chains (e.g., G/M/1 – type Markov Chains, M/G/1 – type Markov Chains) are endowed with geometric recursion for the equilibrium probabilities. i.e.

$$\bar{\pi}(n + 1) = \bar{\pi}(n)R, \tag{1}$$

where ‘R’ (rate matrix) is the matrix solution of the matrix power series equation.

$$\sum_{i=0}^{\infty} R^i A_i \equiv \bar{0} \tag{2}$$

(For continuous time skip-free Markov Chains of G/M/1 – type) and in (1), $\bar{\pi}(n)$ is the equilibrium probability vector of states on level ‘n’. In the case of Quasi – Birth and Death (QBD) continuous time Markov Chains, the rate matrix R is a matrix solution of a matrix quadratic equation (2), (3).

$$R^2 A_2 + RA_1 + A_0 \equiv \bar{0} \tag{3}$$

In the context of level-dependent QBD processes, Latouche et al. showed the existence of matrix geometric recursions of the form.

$$\bar{\pi}(n + 1) = \bar{\pi}(n)R(n), \tag{4}$$

where rate matrices ($R(n)$'s) (4) are solutions of matrix quadratic equations.

Beuermann and Coyle [21] proposed an approach for equilibrium analysis of QBD processes using a state space expansion method (i.e., the number of states at each level are increased) based on the concept of LCI completeness (Level Crossing Information – LCI). In this approach, we have matrix geometric recursion.

$$\bar{\pi}(n + 1) = \bar{\pi}(n)\bar{W}, \tag{5}$$

where \bar{W} is the solution of a linear matrix equation and can be computed easily.

Zhang and Coyle [10] were led to the question of whether LCI completeness can be used in the transient analysis of QBD processes. They succeeded in their effort and arrived at a matrix geometric recursion (5) in the Laplace transform (18) domain i.e.

$$\bar{\pi}_{(n+1)}(s) = \bar{\pi}_{(n)}(s)\bar{W}(s), \tag{6}$$

where $\bar{W}(s)$ is the solution of the linear matrix equation. (For a level independent QBD Process)

$$\bar{\pi}_{(n+1)}(s) \dots \text{Laplace transform of components of vector } \bar{\pi}_{(n+1)}(t)$$

The transient probability distribution is determined by inverse Laplace transform numerically. Also, with this approach, the number of states at each level can be doubled in the worst case. This approach demonstrated the “feasibility” of transient analysis of QBD processes, arising in applications.

The research reported in this research paper is motivated by the question of whether the transient probability distribution can be “efficiently” computed in the time domain without state space expansion.

Goal: To compute $\bar{\pi}_n(t_0)$ for any specific time value “ t_0 ” (before the equilibrium is reached) with the time complexity being smaller than by Laplace transform based approach.

3. Coupled buffers: Quasi-Birth-and-Death process model

Take a glance at Fig. 1, for an instance of a basic job scheduler [16]. Jobs are routed from the source to the two processors.

3.1. Coupled queues: a more precise model

Consider queue length $L_i(t)$ (a random variable) available at node i . Let the service time be assumed to be exponential with a mean of $1/\mu_i$ (for simplicity). At a Poisson rate λ , packets arrive at the job source (scheduler) and are routed to either node 1 or node 2.

The source receives the values of $L_1(t)$ and $L_2(t)$. Based on this information [2], the source directs packets through the intermediary nodes with the shortest queues (it has been demonstrated that if “fresh” information is available, this join-the-shortest-queue approach outperforms the best routing method [13] that ignores queue length information). The paradigm of two independent M/M/1 queues [19] presented in [9,15,18] is no longer applicable if the above control is implemented. It is, nonetheless, a useful approximation.

The jobs received at the source are forwarded to the node with the shortest queue length, as previously indicated. If the queue lengths are identical, an independent (statistically) coin toss is done, and the job is routed to either node 1 or node 2. In this scenario, let ‘p’ be the probability of routing to queue 1. It is expected that concurrent [20] arrival and departure of jobs [16] from node 1 or node 2 buffers are not permitted (the probability of simultaneous arrival and departure is very small).

Based on the foregoing modeling [12] assumptions, the queues at buffer 1 and buffer 2 are not independent as they have been supposed and utilized in the past.

Table 1
Parameters in the stochastic model.

| S.No | Parameter | Description |
|------|-----------|----------------------------------|
| 1 | μ | Service rate at both buffers |
| 2 | λ | Packet arrival rate at source |
| 3 | p | Probability of selecting queue 1 |
| 4 | Q | Generator Matrix |
| 5 | C_N | Companion Matrix |

With the aforementioned modeling assumptions, let us consider $L_1(t)$ to be the length of the queue at buffer 1 at time t and $L_2(t)$ to be the length of the queue at buffer 2 at time t . It is easy to see that $[L_1(t), L_2(t)]$ constitutes a finite state space Quasi-Birth-and-Death process. The stochastic model parameters are incorporated in Table 1.

3.2. Quasi birth and death process generator matrix

Let the size of the buffer at the first node be N and the other node also be N . As a result, the state space of the level-dependent quasi-birth and death process is defined as follows

$$E = \{(i, j) : 0 \leq i \leq N : 0 \leq j \leq N\}$$

The ensuing quasi-birth-death process should be level dependent, based on the modeling assumptions. It will be in the following form.

$$Q = \begin{pmatrix} A_{00} & B_{01} & 0 & 0 & \dots & \dots & \dots & \dots \\ C_{10} & A_{11} & B_{12} & 0 & \dots & \dots & \dots & \dots \\ 0 & C_{21} & A_{22} & B_{23} & \dots & \dots & \dots & \dots \\ \vdots & \vdots & \vdots & \vdots & \dots & \dots & C_{NN-1} & A_{NN} \end{pmatrix}$$

In the Generator Matrix, each sub-matrix is of the dimension $N \times N$. The sub-matrices in the generator matrix are given in the following description.

$$A_{00} = \begin{pmatrix} -\lambda & (1-p)\lambda & 0 & 0 & 0 & 0 & 0 & 0 \\ \mu & -(\mu + \lambda) & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \mu & -(\mu + \lambda) & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \mu & -(\mu + \lambda) & 0 & 0 & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & \dots & \dots & \mu & -(\mu + \lambda) \end{pmatrix}$$

$$A_{NN} = \begin{pmatrix} -(\mu + \lambda) & \lambda & 0 & 0 & 0 & 0 & 0 & 0 \\ \mu & -2(\mu + \lambda) & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \mu & -2(\mu + \lambda) & \lambda & 0 & 0 & 0 & 0 \\ 0 & 0 & \mu & -2(\mu + \lambda) & \lambda & 0 & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & \dots & \dots & \mu & -2\mu \end{pmatrix}$$

$$C_{kk-1} = \begin{pmatrix} \mu & 0 & 0 & \dots \\ 0 & \mu & 0 & \dots \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \mu \end{pmatrix}$$

i.e. a diagonal matrix for $k > 0$. The remaining sub-matrices (11) are as below.

$$B_{kk+1} = \begin{pmatrix} 0 & \dots & 0 & \dots & \dots & \dots & \dots \\ \dots & \dots & 0 & \dots & \dots & \dots & \dots \\ 0 & \dots & 0 & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & p\lambda & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \lambda & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \lambda & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \lambda \end{pmatrix}$$

i.e. the K^{th} row and K^{th} column entry is $p\lambda$. The other higher entries on the diagonal are all λ 's. All the other lower entries in the matrix are zeroes.

$$A_{kk} = \begin{pmatrix} -(\mu + \lambda) & \lambda & 0 & 0 & \dots & \dots & \dots & 0 \\ \mu & -(2\mu + \lambda) & \lambda & 0 & \dots & \dots & \dots & 0 \\ 0 & \mu & -(2\mu + \lambda) & \lambda & \dots & \dots & \dots & 0 \\ 0 & 0 & \mu & -(2\mu + \lambda) & \lambda & \dots & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \mu & 0 & -(\lambda + 2\mu) & (1 - p)\lambda & \dots & -2\mu \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \mu & -(\lambda + 2\mu) \end{pmatrix}$$

for $k \neq N, 0$. Except for the first and the last block matrix on the diagonal i.e. except for A_{00}, A_{NN} all the other block diagonal matrices are of the above form i.e. only the $(k + 1)^{th}$ row of the block diagonal matrix has the adjacent entries $(\lambda + 2\mu)$ and $(1 - p)\lambda$. The last row has all the entries zero except for the last two entries $\mu, -(\lambda + 2\mu)$.

3.3. Transient performance evaluation

We now discuss the computation of the Transient probability distribution of continuous Time Markov Chains. We first derive the transient probability distribution computation in the Laplace Transform domain.

3.3.1. Algorithm for the computation of transient probability distribution (Laplace transform domain)

It is generally known that the vector-matrix differential equation satisfied by the transient probability distribution of any continuous-time Markov chain [17] (and hence the QBD process), is

$$\frac{d}{dt} \bar{\pi}(t) = \bar{\pi}(t)Q, \tag{7}$$

where Q is Generator Matrix. The solution to such a differential equation (7) is

$$\bar{\pi}(t) = \bar{\pi}(0)e^{Qt} \text{ for } t \geq 0 \tag{8}$$

In the case of infinite state space CTMC, e^{Qt} can be computed by the well-known method in control theory. We now provide an approach in the Laplace Transform domain. Applying both sides Laplace transformation (8) [10], we have

$$s\bar{\pi}(s) - \bar{\pi}(0) = \bar{\pi}(s)Q \tag{9}$$

we easily extract the following equation from the above (9) equation.

$$\bar{\pi}(s) = \bar{\pi}(0)(sI - Q)^{-1} \tag{10}$$

As a result, efficient transient probability distribution computation simply comes down to efficient inversion of the matrix $(sI - Q)$ (10). The Laplace transformation of the transient probability distribution can likewise be gotten recursively (as exhibited in [10]) as shown in the following Lemma.

Notation:

$\bar{\pi}(t)$: Transient PMF vector at time t

$\bar{\pi}(s)$: Laplace transform vector of $\bar{\pi}(t)$

Lemma. *The Laplace transform of the level dependent quasi birth and death process transient probability vector is obtained recursively using the following recursion.*

$$\bar{\pi}_n(s) = \bar{\pi}_{n+1}(s)W_n(s) \tag{11}$$

$W_n(s)$ is computed using the sub-matrices of the generator matrix's.

Proof. Refer [10].

3.3.2. Numerical implementation

Step 1: Using the results in [3], the recursion matrices $W_n(s)$ are computed in closed form based on the sub-matrices in the generator matrix.

Step 2: The vectors $\{\bar{\pi}_n(s) \text{ for } n \geq 0\}$ are determined for several values of 's' (say 'k') (in the region of convergence of the Laplace transform).

Step 3: Using the inverse Laplace transform routine, the vectors $\{\bar{\pi}_n(t), n \geq 0\}$ are determined for several desired values of 't'.

3.3.3. Transient analysis in the time domain

We now propose a new method for transient probability distribution computation of arbitrary Continuous Time Markov Chains directly in the time domain.

This method is (applies to the Transient analysis of Finite Markov chains (DTMC, CTMC)) based on a novel method of computation of e^{Qt} (7, 8) based on results in [7], [8]. We now briefly summarize the results in [7], [8].

To illustrate the main theorem in [8], we first provide the details in the case of ‘ Q ’ with 2 states. The approach works for an arbitrary 2×2 matrix X .

When X is a 2×2 matrix, using the Cayley Hamilton theorem, we infer that $X^{j's}$ for $j \geq 2$ (with fixed pair of eigenvalues of X) can be expressed in terms of matrices X, I using scalar coefficients determined by the coefficients of the characteristic polynomial of X . For instance, we readily have that

$$X^3 = (b_1^2 - b_0)X + (b_1 b_0)I \tag{12}$$

Where $Det(\lambda * I - X) = \lambda^2 + b_1 \lambda + b_0$, with $b_1 = -Trace(X)$ and $b_0 = Determinant(X)$ ($X^2 I + b_1 X + b_0 I \equiv \bar{O}$)

Now letting,

$$b_1^{(2)} = b_1, b_0^{(2)} = b_0 \text{ and}$$

$$b_1^{(3)} = (b_1^2 - b_0), b_0^{(3)} = b_1 b_0.$$

We have that, the following recursive equation holds:

$$\begin{pmatrix} -b_1^{(3)} \\ -b_0^{(3)} \end{pmatrix} = \begin{pmatrix} -b_1^{(2)} & 1 \\ -b_0^{(2)} & 0 \end{pmatrix} * \begin{pmatrix} -b_1^{(2)} \\ -b_0^{(2)} \end{pmatrix} = C_{(2)} \begin{pmatrix} -b_1^{(2)} \\ -b_0^{(2)} \end{pmatrix}$$

where C_2 is a Companion Matrix. This recursion was first observed in [7].

Letting, $X^4 = -b_1^{(4)} X - b_0^{(4)} I$

It can be readily shown that

$$\begin{pmatrix} -b_1^{(4)} \\ -b_0^{(4)} \end{pmatrix} = (C_{(2)})^2 \begin{pmatrix} -b_1^{(2)} \\ -b_0^{(2)} \end{pmatrix}$$

In general, X^M can be expressed in the terms of (X, I) with two suitable coefficients. The two coefficients are obtained using the following recursion (generalization of the above result):

$$\begin{pmatrix} -b_1^{(M)} \\ -b_0^{(M)} \end{pmatrix} = (C_{(2)})^{M-2} \begin{pmatrix} -b_1^{(2)} \\ -b_0^{(2)} \end{pmatrix}$$

Thus, the coefficients of higher power of X can be expressed in terms of a 2×2 companion matrix and the coefficients of characteristics polynomial of X (12).

The following theorem is a generalization of the above result and enables efficient computation of e^{Qt} by truncation analytically.

In the following, we consider Q instead of X . In general, Q^m , for $m \geq N$ can be expressed in terms of $I, Q, Q^2, Q^3, \dots, Q^{N-1}$ with suitable coefficients. The coefficients can be obtained using the following recursion:

$$\begin{pmatrix} -b_{N-1}^{(m)} \\ -b_{N-2}^{(m)} \\ -b_{N-3}^{(m)} \\ \vdots \\ -b_2^{(m)} \\ -b_1^{(m)} \\ -b_0^{(m)} \end{pmatrix} = (C_N)^{m-N} \begin{pmatrix} -b_{N-1}^{(N)} \\ -b_{N-2}^{(N)} \\ -b_{N-3}^{(N)} \\ \vdots \\ -b_2^{(N)} \\ -b_1^{(N)} \\ -b_0^{(N)} \end{pmatrix}$$

Where C_N is

$$C_N = \begin{pmatrix} -b_{N-1} & 1 & 0 & 0 & \dots & \dots & 0 & 0 \\ -b_{N-2} & 0 & 1 & 0 & \dots & \dots & 0 & 0 \\ -b_{N-3} & 0 & 0 & 1 & \dots & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ -b_2 & 0 & 0 & 0 & \dots & \dots & 0 & 1 \\ -b_1 & 0 & 0 & 0 & \dots & \dots & 0 & 1 \\ -b_0 & 0 & 0 & 0 & \dots & \dots & 0 & 0 \end{pmatrix}$$

i.e. C_N is the Companion matrix associated with Characteristic Polynomial of Q .

Theorem. $(Q)^m = -b_{N-1}^{(m)} * Q^{N-1} - b_{N-2}^{(m)} * Q^{N-2} - b_{N-3}^{(m)} * Q^{N-3} \dots - b_1^{(m)} * Q - b_0^{(m)} * I$. Where Q is $N \times N$ matrix and $m \geq N \geq 3$.

Proof. Refer [8].

Now we use the above theorem to compute e^{Qt}

$$e^{Qt} = \sum_{j=0}^{\infty} \frac{Q^j}{j!} t^j \tag{13}$$

By Cayley Hamilton theorem and the above (13) theorem

$$e^{Qt} = \sum_{j=0}^{N-1} \frac{Q^j t^j}{j!} + \sum_{j=N}^{\infty} \left(- \sum_{i=0}^{N-1} b_i^{(j)} Q^i \right) \frac{t^j}{j!} \tag{14}$$

Changing indices in the second summation (14)

$$e^{Qt} = \sum_{j=0}^{N-1} \left[\frac{Q^j t^j}{j!} \right] - \sum_{j=0}^{N-1} \left[\sum_{i=N}^{\infty} b_j^{(i)} Q^j \frac{t^i}{i!} \right] \tag{15}$$

$$e^{Qt} = \sum_{j=0}^{N-1} \left[\frac{Q^j t^j}{j!} \right] - \sum_{j=0}^{N-1} \left[\sum_{i=N}^{\infty} \frac{b_j^{(i)} t^i}{i!} \right] Q^j$$

We readily have that $\bar{\pi}(t) = \bar{\pi}(0)e^{Qt}$, where $\bar{\pi}(t)$ is the vector of transient probabilities of a CTMC and $\bar{\pi}(0)$ is the initial probability vector.

It should be noted that by theorem, the coefficients $b_j^{(i)}$ s can be (15) computed recursively.

Using the transient probability mass function, various time-varying performance measures can be readily computed. Some of them are directly related to the moments of the transient PMF.

3.3.4. Efficient computation of transient probability distribution of CTMC

$$e^{Qt} = \sum_{j=0}^{N-1} \left[\frac{Q^j t^j}{j!} \right] - \sum_{j=0}^{N-1} \left[\sum_{i=N}^{\infty} \frac{b_j^{(i)} t^i}{i!} \right] Q^j \tag{16}$$

Using the recursion for the coefficients $b_j^{(i)}$ s.

I.e. $\bar{b}^{(i)} = \bar{C}\bar{b}^{(i-1)}$ (for $i \geq N$), the inner summation in the above (16) expression is truncated (e.g. when L^2 - norm ($\bar{b}^{(i)}$) is sufficiently small). Thus, we have

$$e^{Qt} = \sum_{j=0}^{N-1} Q^j \left(\frac{t^j}{j!} - \sum_{i=N}^L b_j^{(i)} \frac{t^i}{i!} \right) \tag{17}$$

In the above expression (leading to transient PMF), $\{Q^j$ for $j \geq 2\}$ can be recursively computed. We may be interested in computing e^{Qt} at integer multiples of say “ t_0 ”.

Thus $\{e^{Qt} : t \in \{t_0, 2t_0, \dots\}\}$ is precomputed using the above (17) approximation. Using such approximation transient PMF and hence the transient performance measures are determined at integer multiples of times “ t_0 ”.

4. Numerical results

A specific example is used in the following discussion to demonstrate the efficient computation of equilibrium distribution.

At Node 1 and Node 2, the Max buffer length is 2

Service rate at both nodes is $\mu = 0.5$

Packet arrival rate at source is $\lambda = 0.3$

Probability of selecting queue 1 when both have the same size is $p = 0.6$

The following description of the generator’s sub-matrices is given with these parameters. They are inferred from the sub matrix general expressions in Section 3. However certain sub-matrices are included here for clarification.

$$Q = \begin{pmatrix} A_{00} & B_{01} & 0 \\ C_{10} & A_{11} & B_{12} \\ 0 & C_{21} & A_{22} \end{pmatrix}$$

For the above values of λ and μ , the sub-matrices are given by

$$A_{00} = \begin{pmatrix} -0.3 & 0.12 & 0 \\ 0.5 & -0.8 & 0 \\ 0 & 0.5 & -0.8 \end{pmatrix}$$

$$A_{11} = \begin{pmatrix} -0.8 & 0.3 & 0 \\ 0.5 & -1.3 & 0.12 \\ 0 & 0.5 & -1.3 \end{pmatrix}$$

$$A_{22} = \begin{pmatrix} -0.8 & 0.3 & 0 \\ 0.5 & -1.3 & 0.3 \\ 0 & 0.5 & -1 \end{pmatrix}$$

$$B_{01} = \begin{pmatrix} 0.18 & 0 & 0 \\ 0 & 0.3 & 0 \\ 0 & 0 & 0.3 \end{pmatrix}$$

$$B_{12} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0.18 & 0 \\ 0 & 0 & 0.3 \end{pmatrix}$$

$$C_{10} = \begin{pmatrix} 0.5 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 0.5 \end{pmatrix}$$

$$C_{21} = \begin{pmatrix} 0.5 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 0.5 \end{pmatrix}$$

Now we will compute the values of I_0, I_1 and I_2 recursively.

$$I_0 = A_{00} = \begin{pmatrix} -0.3 & 0.12 & 0 \\ 0.5 & -0.8 & 0 \\ 0 & 0.5 & -0.8 \end{pmatrix}$$

$$I_1 = A_{11} - C_{10}I_0^{-1}B_{01} = \begin{pmatrix} -0.4 & 0.4 & 0 \\ 0.75 & -1.05 & 0.12 \\ 0.1563 & 0.6563 & -1.1125 \end{pmatrix}$$

$$I_2 = A_{22} - C_{21}I_1^{-1}B_{12} = \begin{pmatrix} -0.8 & 0.7238 & 0.0762 \\ 0.5 & -0.8762 & 0.3762 \\ 0 & 0.8095 & -0.8095 \end{pmatrix}$$

π_2 , denotes the equilibrium probability of states on level ‘ i ’.

Starting with $\bar{\pi}_2$, the equilibrium probabilities are recursively computed and normalized.

$$\pi_0 = -\pi_1 C_{10} I_0^{-1} = (0.5330 \quad 0.1390 \quad 0.0085)$$

$$\pi_1 = -\pi_2 C_{21} I_1^{-1} = (0.1808 \quad 0.0860 \quad 0.0136)$$

$$\pi_2 = null(I_2) = (0.0114 \quad 0.0182 \quad 0.0095)$$

$null(I_2)$ is the left null space of matrix I_2 .

Using the transient analysis methods, the transient PMF and transient performance measures are computed.

Zhang and Coyle approach requires state space expansion which requires doubling the number of states at each level (in the worst case). For 100 values of ‘ s ’, the accuracy of the Laplace transform approach and our approach is the same. As ‘ s ’ increases, the accuracy has increased for any value of ‘ t ’ under transient conditions.

For each value of ‘ s ’, a matrix inversion is needed whose time complexity is $O(M^3)$ Where ‘ M ’ is the number of states at each level after state space expansion (to ensure LCI completeness). Also, one matrix multiplication is required to compute $W(S)$.

To be able to determine the transient probability mass function over a finite time horizon (till equilibrium is attained), $\bar{W}_n(s)$ needs to be computed for a large, finite number of values of ‘ s ’. Also, the inverse Laplace transform needs to be computed to determine.

$\bar{\pi}(t)$ for $t \geq 0$ (till the equilibrium reached)

Furthermore, since the QBD model (of coupled queues) is level dependent, a recursion of the form.

$$\bar{\pi}_{(n+1)}(s) = \bar{\pi}_{(n)}(s)\bar{W}_n(s) \tag{18}$$

Thus, the recursion matrix, $\bar{W}_n(s)$ is also level dependent, and it needs to be computed for each level ‘ n ’. This increases the time complexity of the algorithm (for transient analysis based on the Laplace transform-based approach (6) of Zhang and Coyle).

Thus, the time complexity of such an approach depends on

Table 2
Transient evolution of probabilities and expected value.

| S.No | Time (Sec) | $\pi_{(0,1)}(t)$ | $\pi_{(1,1)}(t)$ | $E[X(t)]$ |
|------|------------|------------------|------------------|-----------|
| 1 | 0 | 0.31 | 0 | 0 |
| 2 | 0.17 | 0.26 | 0.075 | 0.9 |
| 3 | 0.27 | 0.21 | 0.101 | 0.11 |
| 4 | 0.37 | 0.18 | 0.103 | 0.115 |
| 5 | 0.47 | 0.16 | 0.1 | 0.117 |
| 6 | 0.74 | 0.15 | 0.09 | 0.15 |

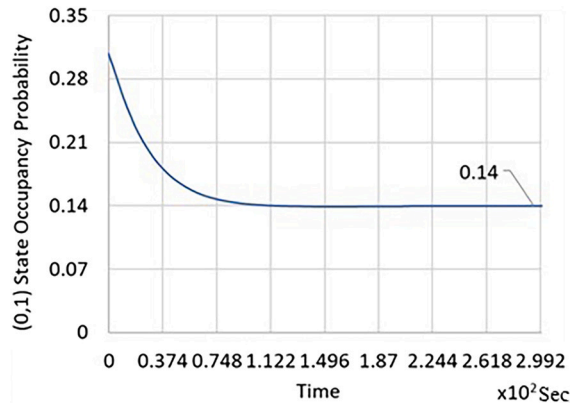


Fig. 2. Time dependent probability evolution for QBD that starts in (0,1) state with probability 0.31 (i.e. Queue 2 has one job in the queue at time 0 and no jobs in queue 1).

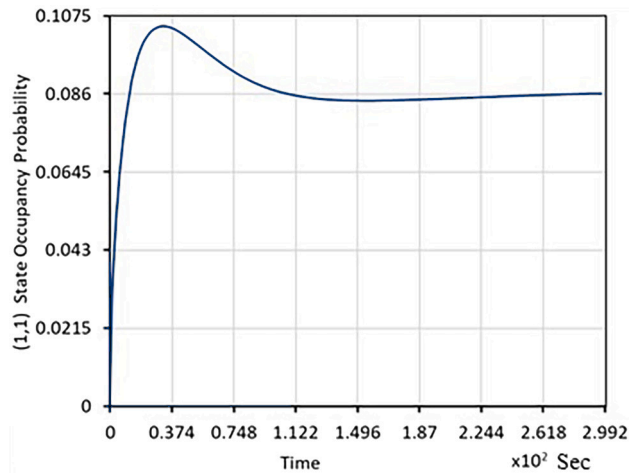


Fig. 3. Probability as a function of time that the process is in the state (1,1). (At time 0, the probability that queues 1 and 2 have one job is zero.).

Number values of ‘s’ for which $\bar{W}_n(s)$ is evaluated.

Also, computations involved in determining the inverse Laplace transform will further increase the time complexity. The exact determination of time complexity is avoided for the sake of brevity.

Summary: The time Complexity of the Laplace transform approach is much larger than that of our approach.

In the adaptive routing, the parameters λ , μ potentially change. In our approach, we only compute transient PMF for a single time instant, say ‘ t_0 ’.

The following plots illustrate the transient analysis of coupled queues arising in dual-core processor performance evaluation. Table 2 explicitly describes the transient evolution of $\pi_{(0,1)}(t)$, $\pi_{(1,1)}(t)$, $E[X(t)]$.

We now briefly summarize the information in the following graphs related to transient performance evaluation.

- In Fig. 2, with one job in Queue 2 and no jobs in Queue 1 initially, it is shown how the transient probability reaches the equilibrium probability.
- In Fig. 3, with zero probability that queues 1 and 2 have even one job; the transient evolution of $\pi_{(1,1)}(t)$ is depicted.

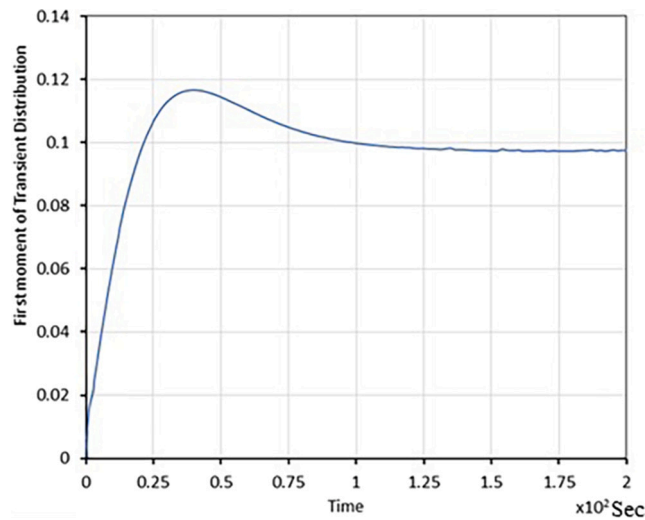


Fig. 4. Time varying Average Queue length of Queue, 1 (for different initial conditions).

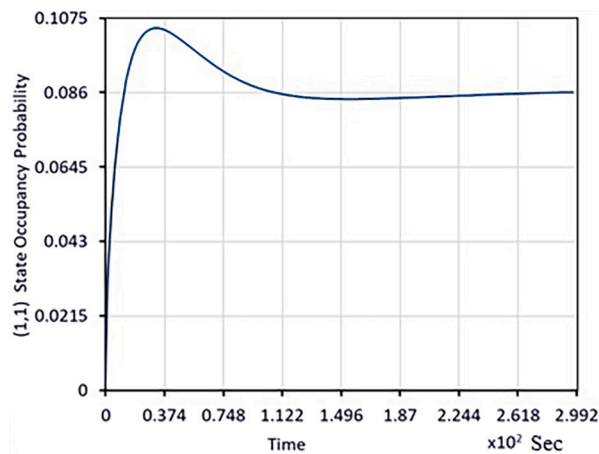


Fig. 5. The tail distribution of QBD Modeling coupled queues (probability that queue length in queue 1 is more than 1).

- In Fig. 4, with zero average queue length at Queue 1, the time evolution of mean queue length is shown.
- In Fig. 5, the time-dependent probability that queue length in Queue 1 is more than 1 is illustrated.

From Fig. 2, 3, 4; the relevant important information is summarized in Table 2.

4.1. Utility of methods for determining the transient probability distribution

The Laplace transformation-based approach is utilized when the transient probability distribution is determined over a wide time horizon (including the convergence to the equilibrium probability distribution).

The time domain approach is preferred when transient probability distribution needs to be determined probability for few (mostly one) time instances.

5. Conclusion

In this research paper, a time domain approach to the transient analysis of arbitrary, finite state space CTMC is proposed. It is computationally more efficient than Laplace transform (S - domain) domain approach. Also, by modeling queues arising in dual-core processors as being coupled, efficient transient performance evaluation is demonstrated.

The proposed approach doesn't directly lead to real-time computation of transient performance measures. We hope to investigate real-time transient performance evaluation in future work. Generalizations of the stochastic model for applications in high-performance computing systems and communication networks are proposed for future work.

CRediT authorship contribution statement

Garimella Ramamurthy: Conceived and designed the experiments; Analyzed and interpreted the data.

Shaik Salma: Performed the experiments; Contributed reagents, materials, analysis tools, or data; Wrote the paper.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgement

We would like to thank the anonymous reviewers for their critical comments which helped the content and presentation of our results in the paper.

References

- [1] I.J.B.F. Adan, J. Wessels, W.H.M. Zijm, Analysis of the asymmetric shortest queue problem, *Queueing Syst.* 8 (1991) 1–58, <https://doi.org/10.1007/BF02412240>.
- [2] A. Ephremides, P. Varaiya, J. Walrand, A simple dynamic routing problem, *IEEE Trans. Autom. Control* AC-25 (4) (1980) 690–693, <https://doi.org/10.1109/TAC.1980.1102445>.
- [3] D.P. Gaver, P.A. Jacobs, G. Latouche, Finite birth-and-death models in randomly changing environments, *Adv. Appl. Probab.* 16 (1984) 715–731, <https://doi.org/10.2307/1427338>.
- [4] H.C. Lin, C.S. Raghavendra, An analysis of the join the shortest queue (JSQ) policy, in: *Proceedings of the 12th Int'l Conference on Distributed Computing Systems*, June 1992, pp. 362–366.
- [5] G. Rama Murthy, Transient and equilibrium analysis of computer networks: finite memory and matrix geometric recursions, Ph.D. Thesis, Purdue University, 1989, Easy Chair Preprint No. 490.
- [6] G. Rama Murthy, Hemant Gogineni, Bharat Bhargava, Modeling adaptive routing in wireless and other networks using coupled queues, in: *Proceedings of IEEE Workshop on Next Generation Wireless Networks*, 2005, pp. 521–529.
- [7] G. Rama Murthy, A. Rummyantsev, On an exact solution of the rate matrix of G/M/1 – type Markov process with a small number of phases, *J. Parallel Distrib. Comput.* 119 (2018) 172–178, <https://doi.org/10.1016/j.jpdc.2018.04.013>, Elsevier Science.
- [8] G. Rama Murthy, B. Chandra Neil, A. Rummyantsev, Coefficients of higher powers of a matrix: companion matrix, 2021, Easy Chair Preprint No. 6678.
- [9] T. Stern, Approximations of queue dynamics and their application to adaptive routing in computer communication networks, *IEEE Trans. Commun. Com-27* (9) (1979), <https://doi.org/10.1109/TCOM.1979.1094546>.
- [10] J. Zhang, E.J. Coyle, Transient analysis of quasi – birth – and death processes, *Commun. Stat., Stoch. Models* 5 (3) (1989) 459–496, <https://doi.org/10.1080/15326348908807119>.
- [11] B. Berg, R. Vesilo, M. Harchol-Balter, heSRPT: parallel scheduling to minimize mean slowdown, in: *38th International Symposium on Computer Performance, Modeling, Measurement, and Evaluation (IFIP PERFORMANCE 2020)*, Milan, Italy, 2020.
- [12] M. Harchol-Balter, Performance Modeling and Design of Computer Systems: Queueing Theory in Action, Cambridge University Press, Cambridge, 2013.
- [13] Y. Raaijmakers, S. Borst, O. Boxma, Stability of redundancy systems with processor sharing, in: *Proceedings of the 13th International Conference on Performance Evaluation Methodologies and Tools (VALUETOOLS'20)*, 2020, pp. 120–127.
- [14] W. Wang, Q. Xie, M. Harchol-Balter, Zero queueing for multi-server jobs, arXiv:2011.10521, 2020, <https://doi.org/10.1145/3447385>.
- [15] P.D. Welch, On a generalized M/G/1 queueing process in which the first customer of each busy period receives exceptional service, *Oper. Res.* 12 (1964) 736–752, <https://doi.org/10.1287/OPRE.12.5.736>.
- [16] W. Weng, W. Wang, Dispatching parallel jobs to achieve zero queueing delay, arXiv:2004.02081v2, 2020.
- [17] J. Abate, G.L. Choudhury, W. Whitt, Asymptotics for steady-state tail probabilities in structured Markov queueing models, *Stoch. Models* 10 (1) (1994) 99–143, <https://doi.org/10.1080/15326349408807290>.
- [18] A. Gandhi, S. Doroudi, M. Harchol-Balter, A. Scheller-Wolf, Exact analysis of the M/M/k/setup class of Markov chains via recursive renewal reward, in: *Queueing Syst. Theory Appl.*, vol. 77(2), 2014, pp. 177–209.
- [19] S.S.L Kim, M/M/s queueing system where customers demand multiple server use, Ph.D. thesis, Southern Methodist University, 1979.
- [20] K. Dragicevic, D. Bauer, A survey of concurrent priority queue algorithms, in: *International Symposium on Parallel and Distributed Processing*, IEEE, April 2008, pp. 1–6.
- [21] S.L. Beuerman, E.J. Coyle, State space expansions and the limiting behavior of quasi-birth-and-death processes, *Adv. Appl. Probab.* 21 (2) (1989) 284–314, <https://doi.org/10.2307/1427161>.

Shaik Salma is a Ph.D. candidate in the Computer Science department at Mahindra Ecole Centrale. She received her Masters degree in the Department of Computer Science from JNTU Anantapur. Her research interests lie in the fields of Machine Learning, Deep Learning, and HiPC. Salma also worked as an Asst. Prof and her e-mail address is: salma20pcse004@mahindrauniversity.edu.in

Dr. Garimella Ramamurthy is a Professor of Computer Science at Mahindra Ecole Centrale, Hyderabad, India. He received Ph.D. in computer engineering from Purdue University, West Lafayette, USA. He has around 30 years of teaching and industrial experience. He has around 300 research papers to his credit. He is a senior member of IEEE, ACM, and a fellow of IETE, India. He received many awards including Rashtriya Gaurav Award. His e-mail address is rama.murthy@mahindrauniversity.edu.in.