Approximate solution for two stage open networks with Markov-modulated queues minimizing the state space explosion problem

Orhan Gemikonakli\textsuperscript{a}, Enver Ever\textsuperscript{a,*}, Altan Kocyigit\textsuperscript{b}

\textsuperscript{a} Middlesex University, UK
\textsuperscript{b} Middle East Technical University, Turkey

Received 4 September 2007; received in revised form 22 December 2007

Abstract

Analytical solutions for two-dimensional Markov processes suffer from the state explosion problem. Two stage tandem networks are effectively used for analytical modelling of various communication and computer systems which have tandem system behaviour. Performance evaluation of tandem systems with feedbacks can be handled with these models. However, because of the numerical difficulties caused by large state spaces, considering server failures and repairs at the second stage employing multiple servers has not been possible. The solution proposed in this paper is approximate with a high degree of accuracy. Using this approach, two stage open networks with multiple servers, break downs, and repairs at the second stage as well as feedback can be modelled as three-dimensional Markov processes and solved for performability measures. Results show that, unlike other approaches such as spectral expansion, the steady state solution is possible regardless of the number of servers employed.

Keywords: Performability; Two stage tandem networks; State explosion

1. Introduction

Markov-modulated queues have a wide range of applications in Queuing systems. Such queues can be described by two-dimensional processes where transitions are only possible between adjacent states of a given model. The popularity of two-dimensional processes resulted in the development of various numerical procedures for their steady-state analysis [1,4,7,9,15,18,24,25,27,10]. These approaches have received some attention and the major ones have been reviewed in [22,14]. In these studies, the Matrix-geometric and Spectral-expansion approaches are compared. Review works highlight strengths as well as weaknesses of numerical techniques used.

Two-dimensional models with multiple components are effectively used for the modelling of queuing systems with multiple queues and/or multiple servers. Such models can be represented by certain two-dimensional Markov-processes on finite or semi-infinite lattice strips. A subset of these Markov processes are the Quasi-Birth-and-Death
processes. The functional equations arising in the analysis of such processes usually present significant analytic
difficulties [5]. These numerical difficulties are frequently caused by large number of states. In other words, difficulty
caused by the rapid increase in the size of the state space of the underlying Markov process is known as the state space
explosion problem.

The limiting impact of state explosion problem is more significant when both of the random variables \( I(t) \) and
\( J(t) \) (the random variables on horizontal and vertical dimensions of a lattice, respectively) [6,7] are used to represent
the number of jobs in two different queues. Two stage tandem networks are typical examples for such models as
considered in [17,7,12].

For multi-dimensional models where one component is finite, there are good analytic-algorithmic methods such as
Matrix-geometric solution [25], and Spectral-expansion [7] methods. These methods can be used to solve the state
explosion problem caused by large or infinite number of \( J(t) \) states. In [11] to overcome the state explosion problem,
a Stochastic Process Algebra based on Performance Evaluation Process Algebra (PEPA), called PEPA1ph, is formulated. PEPA1ph is used to specify systems with potentially infinitely many customers in an infinite queuing
system with durations given as phase-type distributions. Matrix-geometric solution method is used in order to calculate
the steady state probability of the models that arise from a fragment of PEPA1ph. Approximate approaches are used
for open queuing networks as well where the maximum number of jobs in the queue is infinite [8].

Although systems with unbounded queuing capacities can be handled by Spectral-expansion and Matrix-geometric
solution method, as the number of \( I(t) \) states increases they become computationally expensive. The numerical
complexity of the solutions depends on the number of states \( I(t) \) represents [23]. That number determines the size
of the matrix \( R \) used in Matrix-geometric method, and the number of eigenvalues and eigenvectors involved in
Spectral-expansion method. In both solution techniques the size of the matrices used depends on the number of states
represented with \( I(t) \). As the size of the matrices increases the computational requirements increase significantly.
Owing to the large size, and ill-conditioned matrices numerical problems occur [23]. Also the numerical stability of
the solution gets affected especially when heavily loaded systems are considered.

A simple geometric approximation is proposed [23] to be used together with spectral-expansion method to
overcome computational difficulties. The approach uses a restricted spectral expansion, based on a single dominant
eigenvalue and its associated eigenvector. The eigenvalue provides a geometric approximation for the queue size
distribution, while the eigenvector approximates the distribution of \( I(t) \). The dominant eigenvalue is the one nearest
to but strictly less than one. There are techniques for determining the eigenvalues that are near a given number [23].
The approach avoids completely the need to solve a set of linear equations which also means avoiding all problems
associated with ill-conditioned matrices. In order to show the accuracy of the new approach, the performability of
multi-server systems with breakdowns and repairs as well as the performance of tandem systems are considered.

Poisson arrivals and exponentially distributed service times are used for both systems. Tandem systems with multiple
servers, breakdowns and repairs at the second stage are not considered in this study. The limiting effect of state
explosion problem is greater for Tandem systems since \( I(t) \) is used to represent the number of jobs in the queue. In [23] it is stated that the exact method begins to experience numerical difficulties when the queue capacity exceeds 35, and the software (Matlab) starts issuing warnings to the effect that the matrix is ill-conditioned, and the results
may not be reliable. The results given in [23] show that dominant eigenvalue approach works well for the systems
under heavy traffic. The method works for relatively lower arrival rates too but with reduced accuracy.

In [7,12] two-dimensional Markov chains are used to model two stage tandem networks with feedbacks. Each
dimension is used for random variables which represent the number of jobs in each stage. In [12] such a model is used
in order to represent a local area network and a single server system which is connected to this network. However,
server failures and multi-server systems at the second stage could not be considered because of the large number of
states created by these. The analysis performed in these studies are actually pure performance analysis since server
breakdowns are not taken into account. However, since they assume that the system under study does not break
down, pure performance concepts for analysis of communication networks tend to be very optimistic [26]. A more
realistic analysis method, performability, has been introduced in [3] and a conceptual framework of performability
has been considered in Meyer [21]. It is possible to divide one of the random variables (i.e. \( I(t) \)) in order to present
the number of jobs in the system for operative and non-operative states of the server, but this will rapidly increase
the number of states required for two-dimensional representations. As the number of states used to represent such a
system increases, the existing solution methods start to suffer because of numerical cumbersomeness. It is desirable
to develop an approach for two stage tandem systems, in order to ease the numerical difficulties caused by the large
number of states. Such an approach can be used to handle possible failures at the second stage. Furthermore, it may be possible to handle multiple servers with breakdowns and repairs.

It is important to solve the numerical difficulties caused by large number of $I(t)$ states especially for Tandem systems. Owing to these limitations, it is difficult to consider more than one server at stage two which can be used for representation of many practical systems (e.g. communication channels and multi-server systems in tandem). The possible inclusion of multiple servers, breakdowns, and repairs increases the complexity of $I(t)$ and severely limits the maximum queue capacity that can be used for second stage.

This paper presents a new approach for analytical modelling of two stage tandem networks offering improvements in alleviating these problems. The proposed solution is an approximate solution with a high degree of accuracy. Using this approach, two stage tandem networks with multiple servers, breakdowns, and repairs at the second stage and feedback can be modelled as three-dimensional Markov processes and solved for performability measures. Results show that, unlike other approaches such as spectral expansion, the steady state solution is possible regardless of the number of servers employed.

The paper is organised as follows: The next section presents the two stage queuing systems under study. The section on three-dimensional solution explains the approach developed for performability evaluation of these systems. In the next section numerical results for the performability measures are presented in order to address the accuracy of the new approach. CPU times are also presented for the new approach, spectral expansion and simulations comparatively. Relevant conclusions are drawn in the conclusion section.

2. Two stage tandem networks under study

Markov-modulated queues have a wide range of applications in queuing systems, as explained in the previous section. In this study, two stage tandem networks with feedback, breakdowns, repairs and multiple servers at the second stage are considered. Fig. 1 illustrates such a system. An important limitation associated with the methods which can be used to model these systems is the size of $I(t)$ which is used to represent the number of jobs in stage two of a tandem network to be considered for performability evaluation. Existing solutions are capable of handling large numbers of jobs only at the first stage.

In this study, the spectral-expansion technique is adapted to three-dimensional Markov Processes to ease the state space explosion problem. The solution is approximate and iterative. The method produces accurate results for tandem systems with bounded queues. This approach has been used in [13] for performability evaluation of a Network Memory Server [20] with two servers at the second stage. However, the method has not been explained in that work.

In this system, jobs arrive at stage one as a Poisson stream with a mean rate of $\sigma_1$. The jobs are processed by a single server with exponentially distributed service times with a mean rate of $\mu_1$. Jobs leaving stage one are either forwarded to stage two with a probability of $(1 - \theta_1)$, where $0 \leq \theta_1 \leq 1$ or leave the system. Stage one has a bounded queuing capacity ($L_j$) and jobs arriving at a full queue are blocked. Stage two can receive jobs following a Poisson stream of arrivals with a mean rate of $\sigma_2$ from outside the system as well as from stage one. The jobs are served by $K$ identical, parallel processors at this stage. Service times are exponentially distributed with a mean service rate of $\mu_2$ for each processor. Completed jobs either leave the system with a probability of $(1 - \theta_2)$, where $0 \leq \theta_2 \leq 1$ or fed back to stage one for further processing. The queuing capacity of stage two is also limited ($L_i$) and hence, any jobs...
received when the queue is full are blocked. However, in our study, when the queue at the second stage is full, stage one ceases sending jobs.

Such systems are introduced in [17] and a practical example is presented in [13] where a network memory server is attached to a LAN. The LAN and the server are modelled as the first and second stages of a tandem network, respectively. The first stage is assumed to be highly reliable [20]. The second stage, however, suffers from server failures where operative periods are exponentially distributed with a mean failure rate of $\xi$. Following a failure, the failed processor is repaired at a mean repair rate of $\eta$. Repair times are also exponentially distributed. Although there are published work presenting solution to tandem networks with feedback [17,7], those solutions are limited by pure performance measures since server failures are not considered. Any attempt to consider server failures suffers from state space explosion problem. To mitigate the state space explosion problem, in this paper, we propose a three-dimensional Markov model and an approximate analytical solution method.

The system given in Fig. 1 is modelled as a three-dimensional Markov process. In this model $K$ is the total number of servers at stage two, $k$ is the number of the operative servers, and $I(t)$ represents the number of jobs, at stage two at time $t$. When all servers at stage two are broken, or the queue is full, stage one stops sending jobs to stage two. The resultant model is solved using the spectral-expansion method together with an iterative process. Then, the steady state probabilities of the three-dimensional Markov chain are calculated. Consider a discrete time, two-dimensional Markov process on a finite lattice strip. The Markov Process can be defined as $X = \{I(t), J(t); t \geq 0\}$ with a state space of $(0, 1, 2, \ldots, L_1) \times (0, 1, 2, \ldots, L_j)$. Then, $i = 0, 1, 2, \ldots, L_1$, and $j = 0, 1, 2, \ldots, L_j$ can be used to represent all possible states, $(i, j)$ on the lattice strip. This system can be solved using existing approaches where most popular ones are explained in [7,9]. The limiting factor here is the size of $I(t)$. When multiple homogeneous servers with breakdowns are considered for stage two, the size of $I(t)$ becomes $(K + 1)(L_i + 1)$. Clearly, when $K$ is large, only small queuing capacities for stage two can be assumed to avoid the ill-conditioned matrices caused by large size of transition matrices. In practice however, larger queuing capacities exist.

3. The three-dimensional solution approach

The system presented in the previous section can also be modelled by a three-dimensional process. The Markov process can be defined as $X_{3D} = \{I(t), J(t), P(t); t \geq 0\}$ with a state space of $(0, 1, 2, \ldots, L_1) \times (0, 1, 2, \ldots, L_j) \times (0, 1, 2, \ldots, K)$. This process can be seen as a composition of $K + 1$ two-dimensional processes, $X_k$, where $k = 0, 1, \ldots, K$. Therefore, $I(t)$ becomes independent of $K$.

Each $X_k$ can be considered as an independent two-dimensional Markov process. Defining the sum of the steady state probabilities of these two-dimensional processes gives:

$$S_k = \sum_{i=0}^{L_1} \sum_{j=0}^{L_j} P_{i,j,k} \text{ and } \sum_{k=0}^{K} S_k = 1, \quad k = 0, 1, 2, \ldots, K. \quad (1)$$

It is important to calculate $S_k$ for all $k$. For the system considered, on each lattice representing $X_k$, one-step downward transition rates are a function of $\mu_1$, and one-step upward transition rates are a function of $\mu_2 \theta_2$ and $\sigma_1$. Lateral transitions are determined by $\mu_1 (1 - \theta_1), \mu_2$, and $\sigma_2$. Transitions between $X_k, (k \geq 0)$ occur with rates $\xi$, and $\eta$. The sums ($S_k$) can be calculated as:

$$S_k = \left[ k! \sum_{l=0}^{K} \frac{(\eta/\xi)^{l-k}}{l!} \right]^{-1}, \quad k = 0, 1, 2, \ldots, K. \quad (2)$$

Let us introduce matrices $u_k$ for all $i, j$ and $k$, where $u_k = P_{i,j,k}$. For each $u_k, 0 < k < K$, the transition matrices are defined in terms of $\sigma_1, \sigma_2, \mu_1, \mu_2, \theta_1$, and $\theta_2$. For this case, the possible transitions in each two-dimensional process $X_k$ are shown in Fig. 2.

When all servers at stage two are broken (i.e. $k = 0$) or the queue is full (i.e. $i = L_1$) the terms in transition matrices having $\mu_1 (1 - \theta_1)$ are set to zero so that no jobs are sent to stage two from stage one. However, alternative transmission policies can also be incorporated.

The transition matrices $A, A_j, B, B_j$, and $C, C_j$, are used for spectral-expansion solution to the processes $X_k$ and can be summarized as follows:
\[ A_j(i, k) : \text{Purely lateral transition rates, from state } (i, j) \text{ to state } (l, j), \quad (i = 0, 1, \ldots, L_i; l = 0, 1, \ldots, L_i; \ i \neq l; \ j = 0, 1, \ldots, L_j), \text{ caused by a change in the state (i.e., a job has arrived at or departed from the second stage).} \]

\[
A_j = A = \begin{pmatrix}
0 & \sigma_2 & 0 & \cdots & 0 & 0 \\
\min(i, k)\mu_2(1 - \theta_2) & 0 & \sigma_2 & \cdots & 0 & 0 \\
0 & \min(i, k)\mu_2(1 - \theta_2) & 0 & \sigma_2 & \cdots & 0 \\
\vdots & 0 & \min(i, k)\mu_2(1 - \theta_2) & \cdots & \ddots & 0 \\
0 & \vdots & 0 & \ddots & \ddots & 0 \\
0 & 0 & \cdots & 0 & \min(i, k)\mu_2(1 - \theta_2) & 0 \\
\end{pmatrix}
\]

\[ B_j(i, k) : \text{One-step upward transition rate, from state } (i, j) \text{ to state } (l, j + 1), \quad (i = 0, 1, \ldots, L_i; l = 0, 1, \ldots, L_i; \text{ and } j = 0, 1, \ldots, L_j), \text{ caused by a job arrival at the first stage.} \]

\[
B_j = B = \begin{pmatrix}
\sigma_1 & 0 & 0 & \cdots & 0 & 0 \\
\min(i, k)\mu_2\theta_2 & \sigma_1 & 0 & \cdots & 0 & 0 \\
0 & \min(i, k)\mu_2\theta_2 & \sigma_1 & 0 & \cdots & 0 \\
\vdots & 0 & \min(i, k)\mu_2\theta_2 & \sigma_1 & 0 & 0 \\
0 & \vdots & 0 & \ddots & \ddots & 0 \\
0 & 0 & \cdots & 0 & \min(i, k)\mu_2\theta_2 & \sigma_1 \\
\end{pmatrix}
\]

\[ C_j(i, k) : \text{One-step downward transition rate, from state } (i, j) \text{ to state } (l, j - 1), \quad (i = 0, 1, \ldots, L_i; l = 0, 1, \ldots, L_i; \text{ and } j = 0, 1, \ldots, L_j), \text{ caused by the departure of a serviced job from the first stage.} \]
Fig. 3. The possible transitions between lattices $X_k$.

$$C_j = C = \begin{pmatrix} 
\mu_1 \theta_1 & \mu_1 (1 - \theta_1) & 0 & \cdots & 0 & 0 \\
0 & \mu_1 \theta_1 & \mu_1 (1 - \theta_1) & \cdots & 0 & 0 \\
0 & 0 & \mu_1 \theta_1 & \mu_1 (1 - \theta_1) & \cdots & 0 \\
\vdots & 0 & 0 & \mu_1 \theta_1 & \cdots & 0 \\
0 & \cdots & 0 & \cdots & 0 & \mu_1 (1 - \theta_1) \\
0 & \cdots & 0 & \cdots & 0 & \mu_1 \theta_1 
\end{pmatrix},$$

for $0 < j \leq L_j$ and $C_0 = [0]$.

From these, the approximate $P_{i,j,k}$ can be obtained using the spectral-expansion method for each $k$. Since the steady state probabilities for lattice $k$ are initially calculated independent of lattices $k-1$ and $k+1$, it is important to use a technique to compensate for the unaccounted effects of the latter two lattices on lattice $k$. This can be achieved through the use of the balance equations given below. If $s_{x,y}$ represents transitions from lattice $x$ to lattice $y$ where $x \neq y$ and $x, y = 0, 1, 2, \ldots, K$, then, the transitions can be summarized as shown in Fig. 3:

These transitions lead to a set of 27 balance equations in total. All approximate steady state probabilities, $u_k$, can be calculated for $k$ operative servers, where $k = 0, 1, 2, \ldots, K$ as long as there is a spectral-expansion solution. The balance equations can be given as follows:

$$P_{0,0,0} = \frac{\xi P_{0,0,1} + \mu_1 \theta_1 P_{0,1,0}}{\sigma_1 + \eta} \hspace{1cm} (3)$$

$$P_{0,j,0} = \frac{\xi P_{0,j-1,0} + \mu_1 \theta_1 P_{0,j+1,0}}{\sigma_1 + \mu_1 \theta_1 + \eta}, \hspace{1cm} 0 < j < L_j \hspace{1cm} (4)$$

$$P_{0,L_j,0} = \frac{\xi P_{0,L_j-1,0}}{\mu_1 \theta_1 + \eta} \hspace{1cm} (5)$$

$$P_{i,0,0} = \frac{\xi P_{i,0,1} + \mu_1 \theta_1 P_{i,1,0}}{\sigma_1 + \eta}, \hspace{1cm} 0 < i < L_i \hspace{1cm} (6)$$

$$P_{i,j,0} = \frac{\xi P_{i,j-1,0} + \mu_1 \theta_1 P_{i,j+1,0}}{\sigma_1 + \mu_1 \theta_1 + \eta}, \hspace{1cm} 0 < i < L_i, \hspace{1cm} 0 < j < L_j \hspace{1cm} (7)$$

$$P_{i,L_j,0} = \frac{\xi P_{i,L_j-1,0}}{\mu_1 \theta_1 + \eta} \hspace{1cm} (8)$$

$$P_{L_i,0,0} = \frac{\xi P_{L_i,0,1} + \mu_1 \theta_1 P_{L_i,1,0}}{\sigma_1 + \eta} \hspace{1cm} (9)$$

$$P_{L_i,j,0} = \frac{\xi P_{L_i,j-1,0} + \mu_1 \theta_1 P_{L_i,j+1,0}}{\sigma_1 + \mu_1 \theta_1 + \eta} \hspace{1cm} (10)$$

$$P_{0,0,k} = \frac{(k+1) \xi P_{0,0,k+1} + \eta P_{0,0,k-1} + \mu_2 (1 - \theta_2) P_{0,0,k} + \mu_1 \theta_1 P_{0,1,k}}{\sigma_1 + \sigma_2 + k \xi + \eta}, \hspace{1cm} 0 < k < K \hspace{1cm} (11)$$

$$P_{0,L_j,k} = \frac{(k+1) \xi P_{0,L_j,k+1} + \eta P_{0,L_j,k-1} + \mu_2 \theta_2 P_{0,L_j,k-1} + \mu_2 (1 - \theta_2) P_{0,L_j,k}}{\sigma_2 + \mu_1 + k \xi + \eta}, \hspace{1cm} 0 < k < K \hspace{1cm} (12)$$

$$P_{0,j,k} = \frac{(k+1) \xi P_{0,j,k+1} + \eta P_{0,j,k-1} + \mu_1 \theta_1 P_{0,j,k} + \mu_2 \theta_2 P_{0,j-1,k} + \mu_2 (1 - \theta_2) P_{0,j,k}}{\sigma_1 + \sigma_2 + k \xi + \eta}, \hspace{1cm} 0 < k < K, \hspace{1cm} 0 < j < L_j \hspace{1cm} (13)$$
by deleting some components appropriately (e.g. when (29)

\[ P_{i,0,k} = \frac{(k+1)\xi P_{i,0,k+1} + \eta P_{i,0,k-1} + \sigma_2 P_{i-1,0,k} + \min(k, i+1)\mu_2 (1 - \theta_2) P_{i+1,0,k}}{\sigma_1 + \sigma_2 + \mu_1 + \min(k, i) \mu_2 + k\xi + \eta} \]

\[ + \frac{\mu_1 \theta_1 P_{i,1,k} + \mu_1 (1 - \theta_1) P_{i-1,1,k}}{\sigma_1 + \sigma_2 + \mu_1 + \min(k, i) \mu_2 + k\xi + \eta} \] \quad \text{for } 0 < k < K, \ 0 < i < L_i \]  

\[ P_{L_i,0,k} = \frac{(k+1)\xi P_{L_i,0,k+1} + \eta P_{L_i,0,k-1} + \sigma_2 P_{L_i-1,0,k} + \mu_1 (1 - \theta_1) P_{L_i-1,0,k+1}}{\sigma_1 + \min(k, i) \mu_2 + k\xi + \eta} \]  

\[ + \frac{\mu_1 \theta_1 P_{L_i,1,k} + \mu_1 (1 - \theta_1) P_{L_i-1,1,k}}{\sigma_1 + \min(k, i) \mu_2 + k\xi + \eta} \] \quad \text{for } 0 < k < K \]  

\[ P_{i,L_j,k} = \frac{(k+1)\xi P_{i,L_j,k+1} + \eta P_{i,L_j,k-1} + \sigma_2 P_{i,L_j-1,k} + \min(k, i+1)\mu_2 \theta_2 P_{i+1,L_j,k}}{\sigma_2 + \mu_1 + \min(k, i) \mu_2 (1 - \theta_2) + k\xi + \eta} \]

\[ + \frac{\min(k, i+1)\mu_2 (1 - \theta_2) P_{i+1,L_j,k}}{\sigma_2 + \mu_1 + \min(k, i) \mu_2 (1 - \theta_2) + k\xi + \eta} \] \quad \text{for } 0 < k < K, \ 0 < i < L_i \]  

\[ P_{L_i,L_j,k} = \frac{(k+1)\xi P_{L_i,L_j,k+1} + \eta P_{L_i,L_j,k-1} + \sigma_2 P_{L_i,L_j-1,k} + \mu_1 \theta_1 P_{L_i-1,L_j,k}}{\sigma_1 + \min(k, L_i) \mu_2 + k\xi + \eta} \]  

\[ + \frac{\mu_1 \theta_1 + \min(k, L_i) \mu_2 + k\xi + \eta}{\sigma_1 + \min(k, L_i) \mu_2 + k\xi + \eta} \] \quad \text{for } 0 < k < K \]  

\[ P_{L_i,j,k} = \frac{(k+1)\xi P_{L_i,j,k+1} + \eta P_{L_i,j,k-1} + \sigma_2 P_{L_i,j-1,k} + \mu_1 \theta_1 P_{L_i-1,j,k}}{\sigma_1 + \min(k, L_i) \mu_2 + k\xi + \eta} \]

\[ + \frac{\mu_1 \theta_1 + \min(k, L_i) \mu_2 + k\xi + \eta}{\sigma_1 + \min(k, L_i) \mu_2 + k\xi + \eta} \] \quad \text{for } 0 < k < K \]  

These balance equations are derived from Eq. (20) by deleting some components appropriately (e.g. when \( k = K \), all \( k + 1 \) terms are set to zero, for \( k = 0 \), all terms involving \( \mu_2 \) and \( \xi \) are set to zero etc.). The probabilities obtained within \( u_k \) are approximate because transitions between neighbouring lattices are not taken into account. Once the approximate steady state probabilities are calculated, the balance equations (3)–(29) can be used to calculate the steady state probabilities more accurately. An iterative procedure is followed to accurately calculate \( P_{i,j,k} \). The procedure can be given as follows:

(i) \( u_k \) are calculated for \( k = 0, 1, 2, \ldots, K \) using the spectral-expansion method together with (2),

(ii) The balance equations given in (3)–(29) are used to calculate the correct steady state probabilities.

(iii) Mean queue length is calculated for the queuing system considered.

(iv) Second and third steps are repeated until the mean queue length converges sufficiently.
Table 1
MQL for stage one and associated CPU times as a function of the mean arrival rate at stage one, for spectral expansion and three-dimensional approach and \(K = 2\)

<table>
<thead>
<tr>
<th>(\sigma_1)</th>
<th>MQL(_j) spectral expansion</th>
<th>MQL(_j) 3D approach</th>
<th>CPU time spectral expansion (s)</th>
<th>CPU time 3D approach (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5</td>
<td>0.121952</td>
<td>0.121952</td>
<td>67.66</td>
<td>23.225</td>
</tr>
<tr>
<td>1.0</td>
<td>0.277779</td>
<td>0.277779</td>
<td>40.493</td>
<td>39.82</td>
</tr>
<tr>
<td>1.5</td>
<td>0.483874</td>
<td>0.483874</td>
<td>28.134</td>
<td>25.96</td>
</tr>
<tr>
<td>2.0</td>
<td>0.769237</td>
<td>0.769237</td>
<td>17.251</td>
<td>18.207</td>
</tr>
<tr>
<td>2.5</td>
<td>1.190488</td>
<td>1.190487</td>
<td>19.217</td>
<td>20.386</td>
</tr>
<tr>
<td>3.0</td>
<td>1.875025</td>
<td>1.875014</td>
<td>17.408</td>
<td>20.477</td>
</tr>
<tr>
<td>3.5</td>
<td>3.181875</td>
<td>3.181856</td>
<td>20.253</td>
<td>20.296</td>
</tr>
<tr>
<td>4.0</td>
<td>6.666817</td>
<td>6.666859</td>
<td>21.184</td>
<td>23.545</td>
</tr>
<tr>
<td>4.5</td>
<td>45.006128</td>
<td>45.005791</td>
<td>18.232</td>
<td>167.019</td>
</tr>
<tr>
<td>5.0</td>
<td>988.459315</td>
<td>976.44794</td>
<td>18.169</td>
<td>185.829</td>
</tr>
</tbody>
</table>

Table 2
MQL for stage two as a function of the mean arrival rate of stage one, for spectral expansion and three-dimensional approach and \(K = 2\)

<table>
<thead>
<tr>
<th>(\sigma_1)</th>
<th>MQL(_j) spectral expansion</th>
<th>MQL(_j) 3D approach</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5</td>
<td>0.036249</td>
<td>0.036249</td>
</tr>
<tr>
<td>1.0</td>
<td>0.07258</td>
<td>0.07258</td>
</tr>
<tr>
<td>1.5</td>
<td>0.109066</td>
<td>0.109066</td>
</tr>
<tr>
<td>2.0</td>
<td>0.145781</td>
<td>0.145781</td>
</tr>
<tr>
<td>2.5</td>
<td>0.1828</td>
<td>0.1828</td>
</tr>
<tr>
<td>3.0</td>
<td>0.220202</td>
<td>0.2202</td>
</tr>
<tr>
<td>3.5</td>
<td>0.258065</td>
<td>0.258063</td>
</tr>
<tr>
<td>4.0</td>
<td>0.296474</td>
<td>0.296472</td>
</tr>
<tr>
<td>4.5</td>
<td>0.335517</td>
<td>0.335515</td>
</tr>
<tr>
<td>5.0</td>
<td>0.336695</td>
<td>0.336495</td>
</tr>
</tbody>
</table>

The most recent state probabilities are used to compute various performability measures. The main advantage of the new method is the independent two-dimensional solutions for each \(k\) value. This makes it possible to handle larger number of servers at the second stage. Also this approach introduces another degree of flexibility to the model. Since each \(u_k\) is considered independently, it is possible to handle the transition matrices independently for each \(u_k\). For example, when there are no operative servers at the second stage (hence, lattice \(u_0\) is used), then, the first stage will not send any jobs to the second stage. In this case, it is possible to arrange the transition matrix \(C\) accordingly (i.e. since \(k = 0, \mu_1(1 - \theta_1)\) is replaced by 0). A different transmission policy will result in different transition matrices.

4. The accuracy of the three-dimensional solution approach

In this section numerical results are presented comparatively with the results obtained from the spectral-expansion method. Also, simulation results are obtained to check the accuracy of the proposed approach for the case of a large number of servers at the second stage. Such systems cannot be solved using the spectral-expansion method. First, numerical results obtained using the new approach are compared with the numerical results obtained using the spectral-expansion method. When spectral expansion is used, the random variable \(I(t)\) is divided into sections to represent the number of jobs at stage two for various operative states of the servers. This limits the maximum number of servers at the second stage to \(K = 2\). The CPU times are also given for the comparison of computational efficiency. The other parameters are, \(L_j = 1000, L_i = 15\) (it has not been possible to obtain results using spectral expansion and \(L_i > 15\) for \(K = 2\), \(\mu_1 = 5, \mu_2 = 6, \xi = 0.001, \eta = 0.5, \theta_1 = 0.6, \theta_2 = 0.2, \) and \(\sigma_2 = 0\). The results are illustrated in Tables 1 and 2 for the mean queue lengths of the first stage MQL\(_j\), and the second stage MQL\(_j\), respectively. A laptop with 2.00 GHz Intel(R)Core(TM)2 Duo processor and 2 GB RAM, MS VC++ 6.0, and the NAG library (for spectral expansion only) are used to achieve all the results presented in this study.

Tables 1 and 2 show that the new method is accurate and the maximum discrepancy is less than 3.38%. Clearly, results are obtained at the expense of computation time using the proposed approach. When the computational
efficiency of the new approach is compared to that of the spectral-expansion method, in most cases, the latter performs better when such calculations are possible (i.e. as long as spectral expansion can still handle the number of states). CPU times of the three-dimensional approach increase as the load of the system increases jumping sharply for loaded networks. Spectral-expansion method is superior for small $K$. However, as $K$ becomes larger, the state space explosion problem shows itself as a limiting factor for the spectral-expansion method. The main advantage of the proposed technique is the ability to work with large $K$.

A special simulation software is written in C++ language to simulate the actual system. The results obtained from the simulations are compared with the analytical results which are obtained by applying the three-dimensional solution approach to the Markov model of the system. The simulator measures the mean queue lengths at both first and second stages for a given set of system parameters. Each result obtained from the simulations is within the confidence interval of 5% with a probability of 0.95.

In Fig. 4, $\text{MQL}_j$ values are presented as a function of mean arrival rate for 4, 8, and 12 servers at the second stage. The other parameters are $L_j = 100$, $L_i = 40$, $\mu_1 = 5$, $\mu_2 = 0.5$, $\xi = 0.001$, $\eta = 0.5$, $\theta_1 = 0.6$, $\theta_2 = 0.2$, and $\sigma_2 = 0$. $\text{MQL}_j$ results obtained with the new approach are compared with the simulation results. The maximum discrepancy for performability measure $\text{MQL}_j$ is less then 3.88%.

Fig. 5 shows the $\text{MQL}_i$ values computed with the same parameters used in obtaining Fig. 4. For $\text{MQL}_i$ values the maximum discrepancy is less than 4.31% when the analytical results are compared to the simulation results.
parameters are used for 4 and 8 server systems with external arrivals at the second stage where $\sigma_2 = 3$ and service rate of the second stage $\mu_2 = 1$. The results are shown in Fig. 6. For $MQL_j$ and $MQL_i$ values the discrepancies are less then 4.42% and 4.66% respectively.

In Figs. 7 and 8 $MQL_j$ and $MQL_i$ measures are presented respectively as a function of mean arrival rate. This time parameters are taken as $L_i = 1000$, $L_j = 15$, $\mu_1 = 5$, $\mu_2 = 0.5$, $\xi = 0.001$, $\eta = 0.5$, $\theta_1 = 0.5$, $\theta_2 = 0.5$, and $\sigma_2 = 0$. Again tandem systems with 4, 8, and 12 servers at the second stage are considered.

When analytical and simulation results are compared for Figs. 7 and 8, the maximum discrepancies are less than 4.59% and 3.96% for performability measures $MQL_j$ and $MQL_i$ respectively.

Overall results presented in Figs. 4–8 show that the new approach is accurate even for larger number of servers at the second stage of the tandem network.

5. The efficacy of the proposed method

Approaches such as analytical modelling, and simulation enable performing analysis cost effectively, without the need for prototyping. Simulation is mimicking the operation of a real world process over time. This approach is very flexible, and it gives fairly accurate and acceptable results, but for sufficient accuracy, simulations require relatively high computation times. Analytical modelling gives rise to formulae and/or numerical procedures that
Fig. 6. MQL$_j$ and MQL$_i$ as a function of $\sigma_1$ for tandem networks with multiple servers at the second stage where $\sigma_2 = 3$, $L_i = 40$ and $L_j = 100$ ((a) MQL$_j$, $K = 4$, (b) MQL$_i$, $K = 4$, (c) MQL$_j$, $K = 8$, (d) MQL$_i$, $K = 8$).

are computationally more efficient as compared to simulation [2,19]. Analytical modelling of a system requires less computation. When simulations are employed, it may be possible to search the space of parameters and their interactions for the optimum combination, but the tradeoffs among the parameters and among performance measures may not be clear in a few runs, and this would normally take a long time.

Analytical models generally provide the best information for the effects of various parameters and their interactions on system performance, availability, performability etc. [16]. This approach is ideal for quick and, once validated, for relatively accurate results [26,2].

The results presented in the previous section shows the accuracy of the proposed approach. The difference between simulation results and results from new approach is less than 5% when the confidence interval of 5% is considered for simulation. As explained in the previous sections, using spectral expansion when multiple servers at the second stage are considered gives rise to state space explosion. The new approach makes it possible to use analytical modelling for systems with multiple servers at the second stage. However, since the new approach is iterative, it is necessary to show the effectiveness of the new approach. In this section numerical results are presented to compare the CPU times of simulation results and that of the new approach.

*Table 3* shows the percentage discrepancies for results obtained for mean queue lengths of first and second stages respectively. The CPU times of three-dimensional approach and simulation is also shown for, $L_j = 1000$, $L_i = 15$, $\mu_1 = 500$, $\mu_2 = 600$, $\xi = 0.001$, $\eta = 0.5$, $\theta_1 = 0.6$, $\theta_2 = 0.2$, and $\sigma_2 = 0$. 

\begin{table}[h]
\centering
\begin{tabular}{|c|c|}
\hline
Parameter & Value \\
\hline
$L_j$ & 1000 \\
\hline
$L_i$ & 15 \\
\hline
$\mu_1$ & 500 \\
\hline
$\mu_2$ & 600 \\
\hline
$\xi$ & 0.001 \\
\hline
$\eta$ & 0.5 \\
\hline
$\theta_1$ & 0.6 \\
\hline
$\theta_2$ & 0.2 \\
\hline
$\sigma_2$ & 0 \\
\hline
\end{tabular}
\caption{Parameter values for simulation.}
\end{table}
Fig. 7. MQL\(_j\) as a function of \(\sigma_1\) for tandem networks with multiple servers at the second stage where \(L_i = 15\) and \(L_j = 1000\) ((a) \(K = 4\), (b) \(K = 8\), (c) \(K = 12\)).

Table 3
Discrepancy for stage one and two and associated CPU times for simulation and three-dimensional approach where \(K = 2\)

<table>
<thead>
<tr>
<th>(\sigma_1)</th>
<th>Discrepancy MQL(_j) (%)</th>
<th>Discrepancy MQL(_i) (%)</th>
<th>CPU time simulation (s)</th>
<th>CPU time 3D approach (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5</td>
<td>0.2034</td>
<td>0.1407</td>
<td>134.908</td>
<td>21.528</td>
</tr>
<tr>
<td>1.0</td>
<td>0.0072</td>
<td>0.0262</td>
<td>271.3</td>
<td>18.376</td>
</tr>
<tr>
<td>1.5</td>
<td>0.1085</td>
<td>0.0293</td>
<td>406.426</td>
<td>14.758</td>
</tr>
<tr>
<td>2.0</td>
<td>0.0467</td>
<td>0.0096</td>
<td>542.74</td>
<td>12.838</td>
</tr>
<tr>
<td>2.5</td>
<td>0.0163</td>
<td>0.0055</td>
<td>669.255</td>
<td>9.251</td>
</tr>
<tr>
<td>3.0</td>
<td>0.2680</td>
<td>0.0363</td>
<td>796.879</td>
<td>23.354</td>
</tr>
<tr>
<td>3.5</td>
<td>0.0072</td>
<td>0.0364</td>
<td>935.876</td>
<td>51.745</td>
</tr>
<tr>
<td>4.0</td>
<td>0.0943</td>
<td>0.0061</td>
<td>1092.175</td>
<td>20.437</td>
</tr>
<tr>
<td>4.5</td>
<td>0.6390</td>
<td>0.0048</td>
<td>1181.029</td>
<td>326.493</td>
</tr>
<tr>
<td>5.0</td>
<td>3.348</td>
<td>1.76</td>
<td>1237.423</td>
<td>183.753</td>
</tr>
</tbody>
</table>

Table 3 clearly shows that the new analytical approach performs significantly better than simulation in terms of computational cost while the percentage difference is less than 5% for both first and second stages.

In order to further emphasise the efficiency of the new analytical approach, Fig. 9 is presented. In this figure MQL values are presented for first and second stages together with CPU times, to show that computation time of analytical approach is much shorter than simulation time while the discrepancy is less than 5%. Computations are performed for
systems with various number of servers at the second stage. The other parameters are taken as, \( L_i = 1000 \), \( L_j = 40 \), \( \mu_1 = 500 \), \( \mu_2 = 50 \), \( \xi = 0.001 \), \( \eta = 0.5 \), \( \theta_1 = 0.6 \), \( \theta_2 = 0.2 \), \( \sigma_1 = 300 \) and \( \sigma_2 = 0 \).

The maximum discrepancies obtained in Fig. 9 are less than 0.55% and 3.86% for MQL\(_j\), and MQL\(_i\) values respectively. The figure clearly shows that the computation time of analytical approach is much shorter than the simulation time. When \( K = 2 \) is considered, the CPU time of simulation is 23 275.622 s (longer than 6 h and 26 min) where the analytical approach gives the results in 87.1 s. Excess computation time for simulation is mainly due to the heavy load expected in the second stage.

The results shown in this section indicate that, the new approach has significantly shorter computation times than simulation. Since the approach is iterative and approximate, the approach is not as fast as spectral-expansion solution when heavy loads are expected. However, although it is approximate, it still uses the advantages of analytical modelling and gives accurate results.

6. Conclusion

In this paper an approximate solution is presented for three-dimensional Markov processes. The method proposed is capable of working with larger state spaces and has a high degree of accuracy. Using this approach, two-stage open networks with multiple servers, breakdowns, and repairs at the second stage and feedback are modelled as three-
dimensional Markov processes and solved for performability measures. Comparative results are presented using the existing approaches and simulation results.

The proposed three-dimensional solution is slower than most numerical techniques especially for heavily loaded systems, however, it is significantly faster than simulation and more importantly, unlike most numerical techniques, it can handle larger state space. An important contribution of the proposed technique is that, multiple servers in second stage can be handled without causing state space explosion. Furthermore, the use of three dimensions increases the flexibility of the solution making it possible to use known methods (e.g. spectral expansion) for each lattice, $X_k$, with appropriate transitions complementing the solution. The method is applicable to finite queues only, but very large queuing capacities can be used at the first stage. The CPU time will increase with $L_i$, $L_j$, $K$ and $\sigma_1$. A jump in CPU times will occur when the network approaches instability. The technique can further be extended to apply to computing clusters reached through a LAN, where state explosion problem is a serious issue for numerical techniques such as the spectral expansion.

One possible application area of the new approach is to model a multi-server system attached to a highly reliable network. Such a model is successfully used for modelling Network Memory Servers.

References