

SEMI-MARKOV-BASED APPROACH FOR THE ANALYSIS OF OPEN TANDEM NETWORKS WITH BLOCKING AND TRUNCATION

WALENTY ONISZCZUK

Faculty of Computer Science
Białystok Technical University, ul. Wiejska 45A, 15–351 Białystok, Poland
e-mail: walenty@wi.pb.edu.pl

This paper describes an analytical study of open two-node (tandem) network models with blocking and truncation. The study is based on semi-Markov process theory, and network models assume that multiple servers serve each queue. Tasks arrive at the tandem in a Poisson fashion at the rate λ , and the service times at the first and the second node are non-exponentially distributed with means s^A and s^B , respectively. Both nodes have buffers with finite capacities. In this type of network, if the second buffer is full, the accumulation of new tasks by the second node is temporarily suspended (a blocking factor) and tasks must wait on the first node until the transmission process is resumed. All new tasks that find the first buffer full are turned away and are lost (a truncation factor). First, a Markov model of the tandem is investigated. Here, a two-dimensional state graph is constructed and a set of steady-state equations is created. These equations allow calculating state probabilities for each graph state. A special algorithm for transforming the Markov model into a semi-Markov process is presented. This approach allows calculating steady-state probabilities in the semi-Markov model. Next, the algorithms for calculating the main measures of effectiveness in the semi-Markov model are presented. In the numerical part of this paper, the author investigates examples of several semi-Markov models. Finally, the results of calculating both the main measures of effectiveness and quality of service (QoS) parameters are presented.

Keywords: two-node network with blocking and truncation, semi-Markov model, Markov exact algorithm.

1. Introduction

Queueing network models (QNMs) with finite capacity queues and blocking are often used in mathematical models of discrete flow systems. These models are realistic and effective tools for performance analysis of wide classes of systems such as computer systems and networks, telecommunications networks, transportation networks, production lines, or flexible manufacturing systems (Balsamo *et al.*, 2003; Badrah *et al.*, 2002; Economou and Fakinos, 1998; Kaufman and Rege, 1996; Martin, 2002; Oniszczuk, 2006; Sereno, 1999; Zhuang, 1996).

Over the years, many publications have been written related to the analysis and application of QNMs with finite capacity queues and blocking in the field of computer science, operations research, traffic engineering or industrial engineering (Akyildiz, 1988; Balsamo and de Nitto Persone, 1994; Boucherie and van Dijk, 1997; Clo, 1998; Kouvatso and Almond, 1988; Morrison, 1996; Onvural, 1990; Sharma and Virtamo, 2002).

Most of the investigation results in these areas were

collected and ordered in the well-known books such as *Queueing Networks with Blocking. Exact and Approximate Solution* (Perros, 1994) and *Analysis of Queueing Networks with Blocking* (Balsamo *et al.*, 2001). Similarly, entire issues of the *Annals of Operations Research* (on *Queueing Networks with Finite Capacity*, Vol. 79, 1998, and *Performance Evaluation*, Vol. 51, No. 2–3, 2003) were dedicated to queueing networks with blocking, where some sections covered exact analysis, approximate methods and applications. However, there is still a great interest in systems with buffer capacity limitations under different blocking mechanisms (Bouhchouch *et al.*, 1996; Gomez-Corral, 2002; Strelen *et al.*, 1998). A blocking mechanism restricts the total intensity of input streams by forcing certain limitations on the blocking and synchronization procedures (Kouvatso *et al.*, 2000; Ramesh and Perros, 2000; Tolio and Gershwin, 1998; Zhuang *et al.*, 1994).

Most research in the area of two-node (tandem) open networks with blocking (see, e.g., (Perros, 1994)) assumes that each queue is served by a single server, where the

first node has an infinite or a finite capacity and the second node has a finite capacity. The state of this queuing network can be described by the pair of variables indicating the number of tasks in the first node and the number of tasks in the second node. Various closed-form results related to the single server queuing network with exponentially distributed service times include the following two limiting cases: a task at the first node receives an infinitesimal amount of service and the first node is saturated (Perros, 1994; Balsamo *et al.*, 2001). In the first case, if the task arrives at the tandem when the second node buffer is not full, it goes through the first node and immediately joins the second node. In the case of the saturated first node (this node is never empty), the server is either busy serving or blocked. In view of this, the second node becomes an $M/M/1$ finite waiting capacity queue with an overall arrival rate equal to the first node service rate (the single-node approximation). Another special tandem model with blocking assumes that multiple servers serve each queue (Perros, 1994). In this case, a task will get blocked upon completion of service at the first node if at that moment the second node is full. A closed-form solution for the queue-length distribution of this model was obtained with the assumption that the first node is saturated (the single-node approximation). This model is equivalent to a queue with state-dependent arrivals. We say that a node is saturated when there is always at least one task waiting for service, i.e., the node is never empty. Another way to study a tandem configuration is motivated by a Kanban scheme (Boucherie and van Dijk, 1997), where the first node was assumed to be saturated and it continues to serve tasks during the time that the second node is full (the served tasks remain in first node). This approach belongs to single-node decomposition. Similarly, other authors studied the tandem configuration with exponential service times and no intermediate buffers, and no queue in front of the first node or where the first node was assumed to have an infinite (or a finite) capacity (see, for example, (Balsamo and de Nitto Persone, 1994; Gomez-Corral, 2002)).

This paper extends the author's previous research on the open tandem model with blocking (Oniszczyk, 2006). The former paper only considered Markov multiple servers, two-node queuing networks with blocking separated serving lines assuming that the first node is under heavy load. The current publication examines an open non-Markov tandem (the two-node approximation) with blocking separated lines at the first node assuming that the first node has a finite capacity buffer. In both cases, when a departure occurs from the second node, one of the blocked tasks will enter the second node and its associated serving line will become unblocked.

This paper provides a mathematical study of a special type of network configuration (tandem), as shown in Fig. 1. This kind of network has N parallel servers

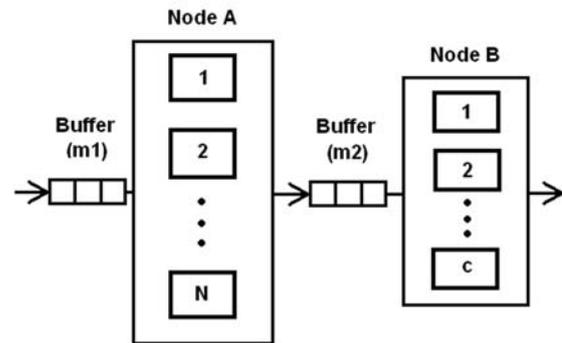


Fig. 1. Tandem model with blocking and truncation.

and a buffer $m1$ at the first node, and c parallel servicing lines (servers) at the other node. Between these nodes is a common waiting buffer with a finite capacity, for example, equal to $m2$. When this buffer is full, the accumulation of new tasks from the first node is temporarily suspended and a phenomenon called blocking occurs, until the queue empties and allows new inserts. This is the classical mechanism for controlling the intensity of the arriving task stream, which comes to the two-node network.

In this kind of tandem configuration, no more than $N + m1 + m2 + c$ tasks can be processed simultaneously and the tandem becomes idle if there are no tasks in both nodes. Assuming that the input stream to the tandem network represents a Poisson process and service time in both nodes corresponds to a random variable with non-exponential distribution, it is a non-Markov model of tandem with blocking and truncation. At the beginning of this paper, the author calculates all possible states of the tandem network, then steady state probabilities and the main tandem measures of effectiveness. Additionally, he shows algorithms for the calculation of blocking and truncation probabilities, delay time in the buffers, blocking time in the node A , the percentage of buffers filling, etc.

The structure of the paper is as follows: Section 2 specifies the tandem model and shows a procedure for finding the state probabilities in a semi-Markov tandem model. Section 3 gives the procedures for calculating the main measures of effectiveness. Section 4 describes the implementation model and numerical examples. Section 5 includes conclusions.

2. Analysis of a semi-Markov tandem with blocking and truncation

Let us consider the two-node network with blocking as shown in Fig. 1. The input task stream comes to the node A . This node has a finite capacity buffer and it can accept only $N + m1$ tasks. New tasks, arriving at the first node, which is full, are not accepted and are rejected. Each task at the first node is processed on the parallel servers and upon service completion is sent to the node B . If

there are free lines on this node, the service process starts immediately, if not, the tasks must wait in the buffer. If the buffer is full, any task upon service completion at the node A is forced to wait and blocks this service line.

The general assumptions for this tandem model are as follows:

1. The external task stream arriving at the node A is assumed to be a Poisson stream, with the rate $\lambda = 1/a$, where a is the mean inter-arrival time.
2. The node A has N parallel servers.
3. The node B has c parallel servers.
4. For both nodes, the service time for each task represents a non-exponentially distributed random variable, with the means $s^A = 1/\mu^A$ and $s^B = 1/\mu^B$, where μ is the mean service rate.
5. The buffer capacity is finite, e.g., equal to $m1$ and $m2$.

If the second buffer is full, the completed task at the node A is forced to wait in this service line, because the transfer process from the node A depends only on the service process in the node B . Physically, blocked tasks stay on the node A , but the nature of the service process in the node B allows us to treat them as located in additional places in the second buffer and they belong to the node B . In this case, there can be maximally $c + m2 + N$ tasks assigned to the second node including all tasks in the first node that might be blocked.

In turn, the maximal number of non-blocked tasks in the first node (the possible number of unblocked, active servers) is equal to N . This means that the current number of tasks that belong to the second node depends on the number of non-blocked tasks in the node A (let it be fixed as i). Therefore, the current number of states in the node B (let us denote it as j) is equal to $j = c + m2 + N - i$.

This paper defines the semi-Markov model of an open two-node network (tandem). The numbers of tasks located simultaneously at the tandem in the first and second nodes are denoted by i and j . This means that a semi-Markov model with two-dimensional state space has a unique path from the state $(0, 0)$ to any state (i, j) and back to the state $(0, 0)$ (see Figs. 2 and 3).

According to the general approach to the analysis of semi-Markov models, the first step is to find solutions to the classical Markov model.

Generally, queuing networks with blocking and truncation are difficult to solve, because their steady state probabilities could not be shown to have a product form solution. Hence, most of the techniques that are employed to analyse these networks are in the form of approximation or numerical techniques. Numerical methods are particularly useful in cases where it is not possible to obtain an analytic solution for the queuing system under

study. The equivalent Markov queuing system under study (with the same rate transition matrix as that of the semi-Markov model) is first formulated as a continuous-time Markov process with discrete states, and subsequently its steady-state probability vector is calculated using an equation solving technique (Balsamo *et al.*, 2001; Korolyuk and Korolyuk, 1999; Perros, 1994). Under appropriate assumptions, a queuing network with blocking and truncation can be formulated as a Markov process and the stationary probability vector can be obtained using numerical methods for systems of linear equations.

Before describing the calculation algorithm for steady-state probabilities, we have to define the service rates for the node A :

$$\begin{aligned} \mu_1^A &= \mu^A, \\ \mu_2^A &= 2\mu^A, \\ &\vdots \\ \mu_i^A &= i\mu^A, \\ &\vdots \\ \mu_N^A &= N\mu^A, \\ &\vdots \\ \mu_{N+m1}^A &= N\mu^A, \end{aligned} \quad (1)$$

and the node B :

$$\begin{aligned} \mu_1^B &= \mu^B, \\ \mu_2^B &= 2\mu^B, \\ &\vdots \\ \mu_c^B &= c\mu^B, \\ &\vdots \\ \mu_{c+m2+N}^B &= c\mu^B. \end{aligned} \quad (2)$$

Based on the analysis of the state space diagrams, the process of constructing the steady-state equations in the Markov model can be divided into several independent steps, which describe some similar, repeatable schemes (see Figs. 2 and 3). These steady-state equations are

$$\lambda p_{0,0} = \mu_1^B p_{0,1} \quad \text{for } i = 0, \quad j = 0,$$

$$\begin{aligned} (\lambda + \mu_j^B) p_{0,j} &= \mu_1^A p_{1,j-1} + \mu_{j+1}^B p_{0,j+1} \\ \text{for } i &= 0, \quad j = 1, \dots, c + m2, \end{aligned}$$

$$\begin{aligned} (\lambda + \mu_i^A) p_{i,0} &= \lambda p_{i-1,0} + \mu_1^B p_{i,1} \\ \text{for } i &= 1, \dots, N + m1 - 1, \quad j = 0, \end{aligned}$$

$$\begin{aligned} (\lambda + \mu_j^B + \mu_i^A) p_{i,j} &= \lambda p_{i-1,j} + \mu_{i+1}^A p_{i+1,j-1} + \mu_{j+1}^B p_{i,j+1} \\ \text{for } i &= 1, \dots, N + m1 - 1, \quad j = 1, \dots, c + m2, \end{aligned}$$

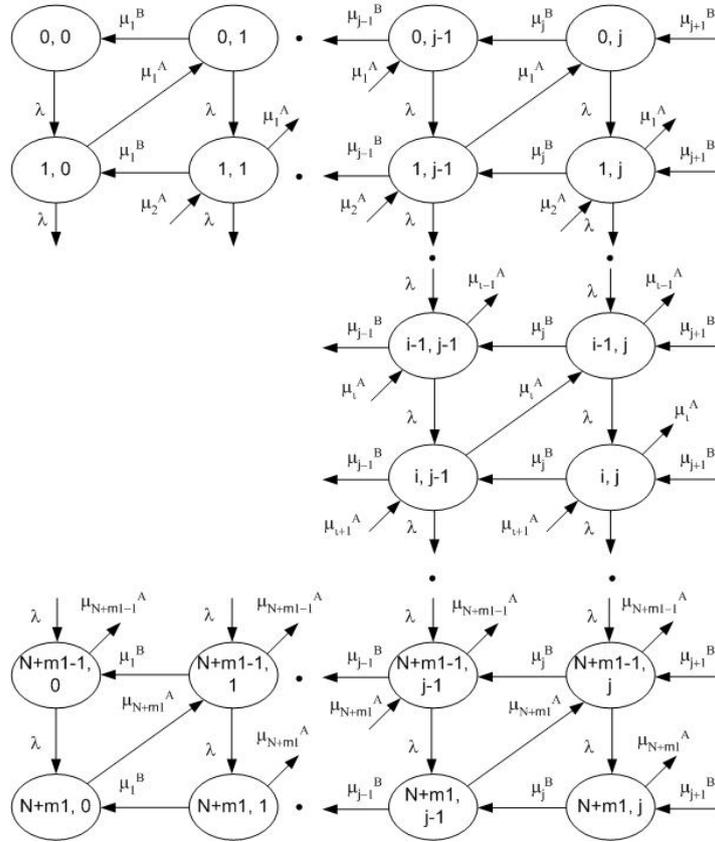


Fig. 2. Two-dimensional tandem state diagram (part I).

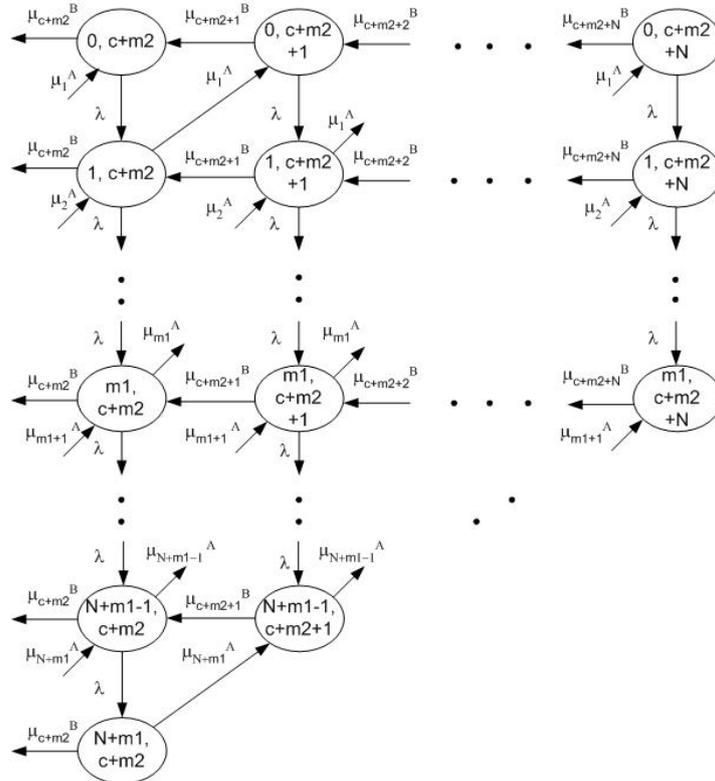


Fig. 3. Two-dimensional tandem state diagram (part II).

$$\begin{aligned} \mu_{N+m1}^A p_{N+m1,0} &= \lambda p_{N+m1-1,0} + \mu_1^B p_{N+m1,1} \\ &\quad \text{for } i = N + m1, \quad j = 0, \\ (\mu_j^B + \mu_{N+m1}^A) p_{N+m1,j} &= \lambda p_{N+m1-1,j} + \mu_{j+1}^B p_{N+m1,j+1} \\ &\quad \text{for } i = N + m1, \quad j = 1, \dots, c + m2 - 1, \\ (\mu_{c+m2}^B + \mu_{N+m1}^A) p_{N+m1,c+m2} &= \lambda p_{N+m1-1,c+m2} \\ &\quad \text{for } i = N + m1, \quad j = c + m2. \quad (3) \end{aligned}$$

For states with blocking, the equations are

$$\begin{aligned} (\lambda + \mu_j^B) p_{0,j} &= \mu_1^A p_{1,j-1} + \mu_{j+1}^B p_{0,j+1} \\ &\quad \text{for } i = 0, \\ &\quad j = c + m2 + 1, \dots, c + m2 + N - 1, \\ (\lambda + \mu_{c+m2+N}^B) p_{0,c+m2+N} &= \mu_1^A p_{1,c+m2+N-1} \\ &\quad \text{for } i = 0, \quad j = c + m2 + N, \\ (\lambda + \mu_{c+m2+N}^B) p_{i,j} &= \lambda p_{i-1,j} + \mu_{i+1}^A p_{i+1,j-1} \\ &\quad \text{for } i = 1, \dots, m1 - 1, \quad j = c + m2 + N, \\ \mu_{c+m2+N}^B p_{m1,c+m2+N} &= \lambda p_{m1-1,c+m2+N} \\ &\quad + \mu_{m1+1}^A p_{m1+1,c+m2+N-1} \\ &\quad \text{for } i = m1, \quad j = c + m2 + N, \\ (\lambda + \mu_j^B + \mu_i^A) p_{i,j} &= \lambda p_{i-1,j} + \mu_{i+1}^A p_{i+1,j-1} + \mu_{j+1}^B p_{i,j+1} \\ &\quad \text{for } i = 1, \dots, m1, \\ &\quad j = c + m2 + 1, \dots, c + m2 + N - 1, \\ (\lambda + \mu_j^B + \mu_i^A) p_{i,j} &= \lambda p_{i-1,j} + \mu_{i+1}^A p_{i+1,j-1} \\ &\quad + \mu_{j+1}^B p_{i,j+1} \\ \text{for } i &= m1 + 1, \dots, N + m1 - 2, \\ &\quad j = c + m2 + 1, \dots, c + m2 + N - 1 - i + m1, \\ (\mu_{c+m2+N-i+m1}^B + \mu_i^A) p_{i,j} &= \lambda p_{i-1,j} \\ &\quad + \mu_{i+1}^A p_{i+1,c+m2+N-(i+1)+m1} \\ &\quad \text{for } i = m1 + 1, \dots, N + m1 - 1, \\ &\quad j = c + m2 + N - i + m1. \quad (4) \end{aligned}$$

The process of solving the equations sets given by (3) and (4) within common algorithms, independently of the initial tandem configuration, is not trivial because part of the graph has an irregular and triangular shape. There are many methods of solving a system of linear algebraic equations but some of these are restricted to certain regular structures of the parameter matrix.

It sometimes happens that the rate transition matrix of a given Markov chain is so highly structured (as in the case of the special type network with blocking and truncation) that it is more efficient to write a specific solution procedure for that problem than to use existing software packages, due to the repetitive nature of the rate transition matrix.

If there is a tandem model with a finite number of states, its steady-state probabilities can be found directly from (3) and (4) by using some iteration method and the normalizing condition for the sum of state probabilities. From the analysis of the tandem state graphs (see Figs. 2 and 3), we can easily calculate the total number K of the tandem states. For the first iteration, as an approximation of probability values, we can assume that all the values are equal to

$$p_{i,j} = \frac{1}{K}. \quad (5)$$

To speed up the convergence of the iteration process before a next, e.g., the $(k+1)$ -th, iteration step, the value of each probability may be calculated as the mean from steps (k) and $(k-1)$. These values are used in the $(k+1)$ -th iteration by substituting them into the right-hand side of (3) and (4), etc.

Sometimes, for a more precise investigation of tandems with blocking and truncation, the methods appropriate for continuous-time Markov processes with discrete states may not be precise enough because in many cases the real service time distributions are different than the exponential ones. These kinds of models may be investigated by using, e.g., semi-Markov process theory and algorithms or methods as shown in (Bradley and Wilson, 2005; Heindl, 2003; Korolyuk and Korolyuk, 1999; Kouvatso and Almond, 1988; Martin, 2002).

Let us examine the semi-Markov model with a finite number of its states, which is equal to K . In the semi-Markov model, during the investigation procedure we try to find the steady-state probabilities $q_{i,j}$ that the model is in state (i, j) . Assume that we have a Markov tandem model with the identical state transition rates as the semi-Markov model. Traditionally, steady-state distributions of semi-Markov processes are found from the embedded Markov chain with a given rate transition matrix (Korolyuk and Korolyuk, 1999). Clearly this still requires the calculation of the entire steady-state distribution of the embedded Markov process. What we achieve in this paper is the direct calculation method of individual steady-state probabilities of the semi-Markov process, which are functions of the sojourn time in a state. According to (Bradley and Davies, 2000), this gives us the possibility of calculating individual steady-state probabilities in less time than it would take to calculate the whole distribution using traditional methods. Let us assume that for each model state the mean sojourn time $m_{i,j}$ (the expected time the process remains in the (i, j) -th state dur-

ing each visit) is known in a semi-Markov model. If we choose the number of state changes L in the equivalent Markov model large enough, we can say that this model visited the (i, j) -th state $L_{i,j} = p_{i,j}L$ times. By the way, the total sojourn time in the semi-Markov model in any given state is

$$T_{i,j} = p_{i,j}Lm_{i,j}. \tag{6}$$

By using some convenient state ordering for the tandem model, we may transform the state description (i, j) to the state number k , where $k = 1, 2, \dots, K$. In the semi-Markov model, the average time for L state changes may be calculated from the following formula:

$$T = \sum_{k=1}^K T_k = L \sum_{k=1}^K (p_k m_k). \tag{7}$$

Assuming that q_k is the (i, j) state probability in the semi-Markov model, its entire mean sojourn time (for this state) during interval T is

$$T_k = q_k T = q_k L \sum_{k=1}^K (p_k m_k), \tag{8}$$

and directly from the relation (6) and (8) we have

$$q_k = \frac{p_k m_k}{\sum_{k=1}^K (p_k m_k)}, \tag{9}$$

where $q_k = q_{i,j}$, $p_k = p_{i,j}$, $m_k = m_{i,j}$, and

$$\sum_{k=1}^K q_k = 1. \tag{10}$$

Recapitulating, in semi-Markov processes any steady-state probability can be calculated from the equivalent Markov models (Korolyuk and Korolyuk, 1999). The final problem, which has to be solved, is how to calculate the parameter $m_{i,j}$ in the semi-Markov model, which means calculating the mean sojourn time for state (i, j) . At the beginning, we try to calculate another parameter, i.e., $m_{i,j}^*$, the mean time simultaneously spent by the tasks during processing by the node A and the node B , if new admissions do not appear (the expected time until the next service completion in either node), where the index i belongs to the node A and the index j belongs to the node B . Let τ_A be the time until the next service completion in the node A , and let τ_B be the time until the next service completion in the node B (independent random variables). Let τ_{AB} be a random variable for the duration of time until the next service completion in the node A or B , whichever comes first (the simultaneous service time). This random variable distribution for any tandem state can be given by the following expressions:

$$\begin{aligned} P(\tau_{AB} > x) &= P(\tau_A > x, \tau_B > x) = \Phi_A(x)\Phi_B(x), \\ \Phi_A(x) &= 1 - F_A(x), \\ \Phi_B(x) &= 1 - F_B(x), \end{aligned} \tag{11}$$

where $F_A(x)$ and $F_B(x)$ are cumulative distribution functions of the random service times in the nodes A and B . From the relations (11), we directly have

$$m_{AB}^* = \int_0^\infty \Phi_A(x)\Phi_B(x) dx. \tag{12}$$

This parameter is the mean sojourn time for the tandem states $(N+m1, 1), \dots, (N+m1, c+m2), (N+m1-1, c+m2+1), \dots, (m1, c+m2+N)$ (see Figs. 2 and 3, the bottom graph states, except the state $(N+m1, 0)$). The calculation process of the mean sojourn time in the remaining tandem states (except the state $(0, 0)$) has to include the task arrival factor, because any visit to state (i, j) would finish upon the next service completion or task arrival. According to the general assumptions for the tandem model, the external task stream is a Poisson process, where the probability of k arrivals in an interval $(0, t)$ is given by

$$p_k(t) = \frac{(\lambda t)^k}{k!} e^{-\lambda t} \text{ for } k \geq 0, t \geq 0. \tag{13}$$

For the states (i, j) mentioned above, and during its simultaneity service time τ_{AB} , only one new task arrival may appear or not (if it appears, the state is changed to $(i+1, j)$). This means that here we have only two events (task occurrence or not):

$$p_0(t) + p_1(t) = 1 \text{ for } t = \tau_{AB}, \tag{14}$$

where

$$p_0(t) = e^{-\lambda t}, \quad p_1(t) = \lambda t e^{-\lambda t} = \lambda t p_0(t)$$

and

$$p_0(t) = \frac{1}{1 + \lambda t} \text{ for } t = \tau_{AB}.$$

This is a task arrival factor. Remembering that a simultaneity service time τ_{AB} has the mean value equal to m_{AB}^* , we may calculate a mean sojourn time m_{AB} directly from the following relation:

$$\begin{aligned} m_{AB} &= m_{AB}^* p_0(t) \\ &= \frac{m_{AB}^*}{1 + \lambda m_{AB}^*} \text{ for } t = m_{AB}^*. \end{aligned} \tag{15}$$

3. Main measures of effectiveness for a tandem with blocking and truncation

The procedures for calculating the basic measures of effectiveness in the tandem model use steady-state probabilities in the following manner:

1. Probability of tandem blocking q_{bl} :

$$q_{bl} = \sum_{i=0}^{m1} \sum_{j=c+m2+1}^{c+m2+N} q_{i,j} + \sum_{i=m1+1}^{m1+N-1} \sum_{j=c+m2+1}^{c+m2+N+m1-i} q_{i,j}. \quad (16)$$

2. Task truncation probability q_{tr} :

$$q_{tr} = \sum_{j=0}^{c+m2} q_{N+m1,j} + \sum_{j=c+m2+1}^{c+m2+N} q_{c+m2+N+m1-j,j}. \quad (17)$$

3. Idle tandem probability q_{idle} :

$$q_{idle} = q_{0,0}. \quad (18)$$

4. The average number of blocked lines in the node A :

$$n_{bl} = \sum_{i=0}^{m1} \sum_{j=c+m2+1}^{c+m2+N} (j - c - m2) q_{i,j} + \sum_{i=m1+1}^{N+m1-1} \sum_{j=c+m2+1}^{c+m2+N+m1-i} (j - c - m2) q_{i,j}. \quad (19)$$

5. The average number of active (non-blocked) tasks in the node A :

$$l_A = \sum_{i=1}^N \sum_{j=0}^{c+m2} i q_{i,j} + \sum_{i=N+1}^{N+m1} \sum_{j=0}^{c+m2} N q_{i,j} + \sum_{j=c+m2+1}^{c+m2+N-1} \sum_{i=1}^{N+c+m2-j} i q_{i,j} + \sum_{j=c+m2+1}^{c+m2+N-1} \sum_{i=N+1+c+m2-j}^{N+m1+c+m2-j} (N + c + m2 - j) q_{i,j}. \quad (20)$$

6. The average number of tasks in the first buffer v_A :

$$v_A = \sum_{i=N+1}^{N+m1} \sum_{j=0}^{c+m2} (i - N) q_{i,j} + \sum_{j=c+m2+1}^{c+m2+N} \sum_{i=N+1+c+m2-j}^{N+m1+c+m2-j} (i + j - c - m2 - N) q_{i,j}. \quad (21)$$

7. The average number of tasks in the node A (buffer + node) n_A :

$$n_A = \sum_{i=1}^{N+m1} \sum_{j=0}^{c+m2} i q_{i,j} + \sum_{j=c+m2+1}^{c+m2+N} \sum_{i=1}^{N+m1+m2+c-j} (i + j - c - m2) q_{i,j}. \quad (22)$$

8. The average number of tasks on the servers in the node B :

$$l_B = \sum_{i=0}^{N+m1} \sum_{j=1}^c j q_{i,j} + \sum_{i=0}^{N+m1} \sum_{j=c+1}^{c+m2} c q_{i,j} + \sum_{i=0}^{m1} \sum_{j=c+m2+1}^{N+c+m2} c q_{i,j} + \sum_{i=m1+1}^{N+m1-1} \sum_{j=c+m2+1}^{N+c+m2+m1-i} c q_{i,j}. \quad (23)$$

9. The average number of tasks in the second buffer v_B :

$$v_B = \sum_{i=0}^{N+m1} \sum_{j=c+1}^{c+m2} (j - c) q_{i,j} + \sum_{i=0}^{m1} \sum_{j=c+m2+1}^{N+c+m2} m2 q_{i,j} + \sum_{i=m1+1}^{N+m1-1} \sum_{j=c+m2+1}^{N+c+m2+m1-i} m2 q_{i,j}. \quad (24)$$

10. The average number of tasks in the node B (buffer + node) n_B :

$$n_B = \sum_{i=0}^{N+m1} \sum_{j=1}^{c+m2} j q_{i,j} + \sum_{i=0}^{m1} \sum_{j=c+m2+1}^{N+c+m2} (m2 + c) q_{i,j} + \sum_{i=m1+1}^{N+m1-1} \sum_{j=c+m2+1}^{N+c+m2+m1-i} (m2 + c) q_{i,j}. \quad (25)$$

11. The mean blocking time in the node A (from Little's formula):

$$t_{bl} = \frac{n_{bl}}{c\mu_B}. \quad (26)$$

12. The mean response time at the node A :

$$q_A = \frac{l_A + v_A}{\lambda} + t_{bl}. \quad (27)$$

13. The mean response time at the node B :

$$q_B = \frac{v_B}{c\mu^B} + \frac{1}{\mu^B}. \quad (28)$$

14. The mean waiting time in the first buffer (from Little's formula):

$$w_A = \frac{v_A}{\lambda}. \quad (29)$$

15. The mean waiting time in the second buffer:

$$w_B = \frac{v_B}{c\mu^B}. \quad (30)$$

16. The tandem response time:

$$t_{thr} = \frac{l_A + v_A}{\lambda} + t_{bl} + q_B. \quad (31)$$

4. Model implementation and numerical examples

The author chose the non-exponential distribution of the service times for experimental investigation of the semi-Markov tandem models.

4.1. Service time: A constant value for the node A and the Erlang- k distribution for the node B . According to the initial assumptions, the service time for the node A has a constant value, e.g., equal to b , meaning $m_A = b$. The service time in the node B has the Erlang- k distribution with the mean value equal to $m_B = k/\mu^B$ and the distribution function given by

$$F_B(x) = 1 - \sum_{r=0}^{k-1} \frac{e^{-\mu^B x} (\mu^B x)^r}{r!}. \quad (32)$$

Therefore,

$$\Phi_B(x) = 1 - F_B(x) = \sum_{r=0}^{k-1} \frac{e^{-\mu^B x} (\mu^B x)^r}{r!}. \quad (33)$$

The distribution function for the node A is given by

$$F_A(x) = \begin{cases} 0, & x \leq b, \\ 1, & x > b. \end{cases} \quad (34)$$

Then, according to (12), we have

$$m_{AB}^* = \int_0^b \left(\sum_{r=0}^{k-1} \frac{e^{-\mu^B x} (\mu^B x)^r}{r!} \right) dx \\ = \sum_{r=0}^{k-1} \int_0^b \frac{e^{-\mu^B x} (\mu^B x)^r}{r!} dx$$

$$= \sum_{r=0}^{k-1} \frac{(\mu^B)^r}{r!} e^{-\mu^B b} \left(-\frac{b^r}{\mu^B} - \frac{rb^{r-1}}{(\mu^B)^2} \right. \\ \left. - \frac{r(r-1)b^{r-2}}{(\mu^B)^3} - \dots \right. \\ \left. - \frac{r!b}{(\mu^B)^r} - \frac{r!}{(\mu^B)^{r+1}} \right) + \frac{1}{\mu^B} \\ = \frac{1}{\mu^B} \sum_{r=0}^{k-1} \left[1 - e^{-\mu^B b} \left(\frac{(\mu^B b)^r}{r!} \right. \right. \\ \left. \left. + \frac{(\mu^B b)^{r-1}}{(r-1)!} + \dots + 1 \right) \right] \\ = \frac{k}{\mu^B} - \frac{e^{-\mu^B b}}{\mu^B} \sum_{r=0}^{k-1} \sum_{l=0}^r \frac{(\mu^B b)^l}{l!}. \quad (35)$$

According to the two-dimensional state diagram for the semi-Markov model, now we must calculate the adequate service rates in the nodes A and B , indicated here as μ_{exp}^A and μ_{exp}^B , in the equivalent model with exponential distributed service times (see Figs. 2 and 3). Thus

$$\mu_{exp}^A = \frac{1}{b}, \quad \mu_{exp}^B = \frac{\mu^B}{k}. \quad (36)$$

Next, from these formulae, we may calculate the service rates μ_j^B and μ_i^A according to the algorithms given in (1) and (2):

$$\mu_j^B = \begin{cases} \mu_{exp}^B j & \text{for } j = 0, 1, 2, \dots, c \\ \mu_j^B = \mu_{exp}^B c & \text{for } j = c + 1, c + 2, \dots, \\ & c + m_2 + N, \end{cases}$$

and the service rate μ_i^A for all states without blocking is

$$\mu_i^A = \begin{cases} \mu_{exp}^A i & \text{for } i = 0, 1, 2, \dots, N \\ \mu_i^A = \mu_{exp}^A N & \text{for } i = N + 1, N + 2, \dots, \\ & N + m_1, \end{cases} \quad (37)$$

while for all states with blocking it is

$$\mu_i^A = \begin{cases} \mu_{exp}^A i + (j - c - m_2) \mu_{exp}^B & \text{for } j = c + m_2 + 1, \dots, c + m_2 + N, \\ & i = 0, 1, \dots, N + m_1 + c + m_2 - j \\ \text{if } i + j - c - m_2 \leq N, \\ (N + c + m_2 - j) \mu_{exp}^A & \\ + (j - c - m_2) \mu_{exp}^B & \\ \text{for } j = c + m_2 + 1, \dots, c + m_2 + N, \\ & i = 0, 1, \dots, N + m_1 + c + m_2 - j \\ \text{if } i + j - c - m_2 > N. \end{cases}$$

Based on the state diagrams from Figs. 2 and 3, the mean sojourn time m_{AB} and the mean simultaneity ser-

vice time m_{AB}^* for states without blocking can be calculated according to the following formulae:

$$\begin{aligned}
m_{0,0} &= \frac{1}{\lambda}, \\
m_{N+m1,0} &= \frac{1}{\mu_{N+m1}^A}, \\
m_{i,j} &= \frac{1}{\mu_j^B} - \frac{e^{-\frac{\mu_j^B k}{\mu_i^A}}}{\mu_j^B k} \sum_{r=0}^{k-1} \sum_{l=0}^r \frac{\left(\frac{\mu_j^B k}{\mu_i^A}\right)^l}{l!} \\
&\quad \text{for } i = N + m1, \quad j = 1, 2, \dots, c + m2, \\
m_{0,j}^* &= \frac{1}{\mu_j^B} \\
&\quad \text{for } j = 1, 2, \dots, c + m2, \\
m_{i,0}^* &= \frac{1}{\mu_j^A} \\
&\quad \text{for } i = 1, 2, \dots, N + m1 - 1, \\
m_{i,j}^* &= \frac{1}{\mu_j^B} - \frac{e^{-\frac{\mu_j^B k}{\mu_i^A}}}{\mu_j^B k} \sum_{r=0}^{k-1} \sum_{l=0}^r \frac{\left(\frac{\mu_j^B k}{\mu_i^A}\right)^l}{l!} \\
&\quad \text{for } i = 1, 2, \dots, N + m1 - 1, \\
&\quad j = 1, 2, \dots, c + m2, \tag{38}
\end{aligned}$$

and for the states with blocking (here a differently calculated service rate μ_i^A):

$$\begin{aligned}
m_{m1,c+m2+N} &= \frac{1}{\mu_{c+m2+N}^B}, \\
m_{i,j} &= \frac{1}{\mu_j^B} - \frac{e^{-\frac{\mu_j^B k}{\mu_i^A}}}{\mu_j^B k} \sum_{r=0}^{k-1} \sum_{l=0}^r \frac{\left(\frac{\mu_j^B k}{\mu_i^A}\right)^l}{l!} \\
&\quad \text{for } j = c + m2 + 1, \dots, \\
&\quad c + m2 + N - 1, \\
&\quad i = N + m1 + c + m2 - j, \\
m_{i,c+m2+N}^* &= \frac{1}{\mu_{c+m2+N}^B} \\
&\quad \text{for } i = 0, 1, \dots, m1 - 1, \\
m_{0,j}^* &= \frac{1}{\mu_j^B} \\
&\quad \text{for } j = c + m2 + 1, \dots, \\
&\quad c + m2 + N - 1, \\
m_{i,j}^* &= \frac{1}{\mu_j^B} - \frac{e^{-\frac{\mu_j^B k}{\mu_i^A}}}{\mu_j^B k} \sum_{r=0}^{k-1} \sum_{l=0}^r \frac{\left(\frac{\mu_j^B k}{\mu_i^A}\right)^l}{l!} \\
&\quad \text{for } j = c + m2 + 1, \dots, \\
&\quad c + m2 + N - 1, \\
&\quad i = 1, \dots, \\
&\quad N + m1 + c + m2 - j - 1.
\end{aligned}$$

In this set of expressions, all $m_{i,j}^*$ can be transformed

to $m_{i,j}$ (mean sojourn time) directly, by using the relation (15).

In this section, the author demonstrates the analysis of tandem with blocking and truncation. The following configuration is chosen: $N = 10$, $c = 7$, $m_1 = 5$, $m_2 = 8$, with the inter-arrival and the service rate in the node A equal to $\lambda = 4.0$, $\mu^A = 0.7$ (the constant service rate, means $b = 1.42857$) and with the service rate μ^B (the Erlang-4 distributed service time) in the node B that changes within a range from 6.8 to 1.6 (for the studying of a model with different utilizations). This model has 361 states; 256 states are without blocking and 105 states are with blocking.

For the model above, the following major results were obtained and presented in Fig. 4 and Tab. 1.

The results of the experiment show that the effect of blocking and truncation appears in a small interval when the utilization of the node B is greater than 0.75. Here, the tandem throughput time and the blocking time quickly increase (bad quality of service parameters). In the moderate utilization interval, the tandem works properly and the quality of service (QoS) parameter is easy to keep on the appropriate level.

4.2. Service time in the nodes A and B : Thinned discrete distributions. In computer networks, the tasks flow and service processes are frequently characterized by using the concept of thinned streams. Here, during the investigative process, a renewal process is used as the base of the thinning operation (Bernoulli thinning). In this kind of renewal process, on a renewal interval a task may or may not occur with probabilities $1 - \pi$ and π . This occurrence probability is not dependent on whether another task occurs or not in the previous interval. The probability π is called the thinning parameter.

In practical applications, another kind of renewal process called a discrete (deterministic) thinned stream may appear. For this process, the renewal intervals have a discrete density function given by the following expression:

$$f_\pi(x) = (1 - \pi) \sum_{i=1}^{\infty} \pi^{i-1} \delta(iT - x), \tag{39}$$

where T is the renewal interval on the discrete output stream and δ is the Dirac delta function.

Here, the distribution functions have a piecewise constant form:

$$\Phi_\pi(x) = \pi^i \quad \text{for } iT \leq x \leq (i+1)T. \tag{40}$$

This kind of discrete thinned stream is often called a discrete Poisson stream, because it has the same characteristics as a Poissonian stream at some points (points that are the multiple renewal intervals T). For these points, the following relations hold true:

$$\pi^i = e^{-\lambda iT}, \quad \lambda = -\frac{\ln \pi}{T}. \tag{41}$$

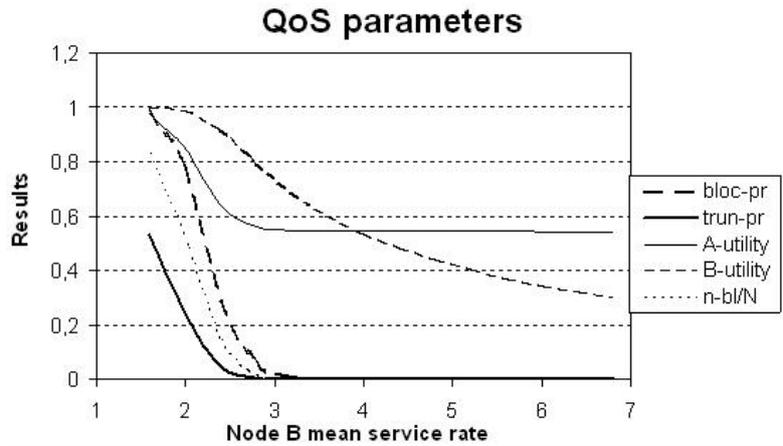


Fig. 4. Graphs of QoS parameters, where *bloc-pr* is the blocking probability, *trun-pr* is the truncation probability, *A-utility* is the node *A* utilization factor (active and blocked tasks), *B-utility* is the node *B* utilization factor, *n - bl/N* is the ratio of the mean number of blocked lines to the parameter *N*.

The main characteristics of thinned deterministic streams are the mean value and the standard deviation, given by the following expressions:

$$m = \int_0^\infty \Phi_\pi(x) dx = \frac{T}{1 - \pi}, \quad (42)$$

$$\sigma = m\sqrt{\pi} = \frac{T\sqrt{\pi}}{1 - \pi}.$$

If we have two thinned discrete streams with given parameters π_1 and π_2 and an identical renewal interval T and with at least one common renewal point, then the superposition of two streams is the same discrete thinned regular stream with the thinning parameter equal to $\pi = \pi_1\pi_2$ and with the distribution function

$$\Phi_\pi(x) = (\pi_1\pi_2)^i \quad \text{for } iT \leq x \leq (i + 1)T. \quad (43)$$

In fact, according to (11), we have

$$P(\tau_\pi > iT) = P(\tau_{\pi_1} > iT)P(\tau_{\pi_2} > iT) = (\pi_1\pi_2)^i \quad \text{for } i = 0, 1, \dots \quad (44)$$

Often, the service time distributions are close to thinned deterministic distributions in computer networks. Let us assume the tandem with service times in both nodes having thinned deterministic distributions with parameters π_A and π_B and with the same thinning interval equal to T . If we assume the semi-Markov model with both nodes processing tasks simultaneously, then according to the expressions (43) and (44) the simultaneity service time in this case has the following distribution:

$$\Phi_{AB} = (\pi_A\pi_B)^i \quad \text{for } iT \leq x \leq (i + 1)T. \quad (45)$$

Based on (42), its mean value is

$$m_{AB}^* = \frac{T}{1 - \pi_A\pi_B}. \quad (46)$$

In the next analysis step, a Markov model equivalent to semi-Markov model is chosen, with thinned deterministic distribution of the service time in both nodes. This relation can be found from (41). Therefore,

$$\mu = -\frac{\ln \pi}{T}, \quad T = -\frac{\ln \pi}{\mu}. \quad (47)$$

Knowing that for a Markov tandem model, its mean simultaneity service time for the states which simultaneous tasks processed can be calculated from

$$m_{AB}^{\text{exp}*} = \int_0^\infty e^{-\mu_A x} e^{-\mu_B x} dx = \int_0^\infty e^{-(\mu_A + \mu_B)x} dx = \frac{1}{\mu_A + \mu_B}. \quad (48)$$

Substituting in (47) $\pi_A\pi_B$ and $\mu_A + \mu_B$ for π and μ , respectively, yields the marking service times (cf. (46)):

$$m_{AB}^{\pi*} = T \frac{1}{1 - \pi_A\pi_B} = \frac{-\ln(\pi_A\pi_B)}{(\mu_A + \mu_B)(1 - \pi_A\pi_B)} = -m_{AB}^{\text{exp}*} \frac{\ln(\pi_A\pi_B)}{(1 - \pi_A\pi_B)}. \quad (49)$$

These expressions allow us to calculate the mean simultaneity service time for each state in the semi-Markov tandem model with states presented in Figs 2 and 3. Now, for the equivalent Markov model, we must calculate the service rates in nodes *A* and *B*. Let us denote these service rates by μ_{exp}^A and μ_{exp}^B . Therefore,

$$\mu_{\text{exp}}^A = -\frac{\ln \pi_A}{T}, \quad \mu_{\text{exp}}^B = -\frac{\ln \pi_B}{T}. \quad (50)$$

These formulae allow us to calculate the service rates μ_i^A and μ_j^B from (37).

Table 1. Comparison of mean time parameters.

μ_B	Mean time						Node A	Node B
	q_A	tq_B	w_A	w_B	t_{bl}	t_{thr}	utilizat.	utilizat.
1.6	4.250	5.342	0.925	2.843	2.953	9.592	0.975	1.000
2.0	2.832	4.028	0.518	2.046	1.499	6.861	0.851	0.991
2.4	1.713	2.493	0.129	0.965	0.342	4.026	0.641	0.917
2.8	1.425	1.459	0.031	0.326	0.044	2.884	0.562	0.793
3.2	1.387	0.973	0.019	0.114	0.006	2.360	0.548	0.687
3.6	1.381	0.716	0.018	0.045	0.001	2.097	0.545	0.604
4.0	1.379	0.558	0.018	0.019	0.000	1.937	0.545	0.538
4.4	1.378	0.540	0.018	0.009	0.000	1.829	0.544	0.485
4.8	1.378	0.372	0.018	0.005	0.000	1.750	0.544	0.441
5.2	1.377	0.314	0.018	0.002	0.000	1.691	0.544	0.405
5.6	1.377	0.268	0.018	0.001	0.000	1.644	0.544	0.373
6.0	1.376	0.232	0.018	0.001	0.000	1.608	0.543	0.346
6.4	1.376	0.202	0.018	0.000	0.000	1.578	0.543	0.323
6.8	1.375	0.178	0.018	0.000	0.000	1.553	0.543	0.302

Based on the state diagrams, the mean sojourn time m_{AB} and the mean simultaneity service time m_{AB}^* for states without blocking can be calculated according to

$$\begin{aligned}
m_{0,0}^\pi &= \frac{1}{\lambda}, \\
m_{N+m1,0}^\pi &= m_{N+m1,0}^{\exp} \frac{-\ln \pi_A}{1-\pi_A} = \frac{1}{\mu_{N+m1}^A} \frac{-\ln \pi_A}{(1-\pi_A)}, \\
m_{i,j}^\pi &= m_{i,j}^{\exp} \frac{-\ln(\pi_A \pi_B)}{(1-\pi_A \pi_B)} \\
&= \frac{1}{(\mu_i^A + \mu_j^B)} \frac{-\ln(\pi_A \pi_B)}{(1-\pi_A \pi_B)} \\
&\quad \text{for } i = N+m1, j = 1, 2, \dots, c+m2, \\
m_{0,j}^{\pi*} &= m_{0,j}^{\exp*} \frac{-\ln \pi_B}{1-\pi_B} = \frac{1}{\mu_j^B} \frac{-\ln \pi_B}{(1-\pi_B)} \\
&\quad \text{for } j = 1, 2, \dots, c+m2, \\
m_{i,0}^{\pi*} &= m_{i,0}^{\exp*} \frac{-\ln \pi_A}{1-\pi_A} = \frac{1}{\mu_i^A} \frac{-\ln \pi_A}{(1-\pi_A)} \\
&\quad \text{for } i = 1, 2, \dots, N+m1-1, \\
m_{i,j}^{\pi*} &= m_{i,j}^{\exp*} \frac{-\ln(\pi_A \pi_B)}{(1-\pi_A \pi_B)} \\
&= \frac{1}{(\mu_i^A + \mu_j^B)} \frac{-\ln(\pi_A \pi_B)}{(1-\pi_A \pi_B)} \\
&\quad \text{for } i = 1, 2, \dots, N+m1-1, \\
&\quad j = 1, 2, \dots, c+m2, \tag{51}
\end{aligned}$$

and for the states with blocking (here for a differently calculated service rate μ_i^A):

$$m_{m1,c+m2+N}^\pi = m_{m1,c+m2+N}^{\exp} \frac{-\ln \pi_B}{1-\pi_B}$$

$$\begin{aligned}
&= \frac{1}{\mu_{c+m2+N}^B} \frac{-\ln \pi_B}{(1-\pi_B)} \\
m_{i,j}^\pi &= m_{i,j}^{\exp} \frac{-\ln(\pi_A \pi_B)}{(1-\pi_A \pi_B)} \\
&= \frac{1}{(\mu_i^A + \mu_j^B)} \frac{-\ln(\pi_A \pi_B)}{(1-\pi_A \pi_B)} \\
&\quad \text{for } j = c+m2+1, \dots, \\
&\quad c+m2+N-1, \\
&\quad i = N+m1+c+m2-j, \\
m_{i,c+m2+N}^{\pi*} &= m_{i,c+m2+N}^{\exp*} \frac{-\ln \pi_B}{1-\pi_B} \\
&= \frac{1}{\mu_{c+m2+N}^B} \frac{-\ln \pi_B}{(1-\pi_B)} \\
&\quad \text{for } i = 0, 1, \dots, m1-1, \\
m_{0,j}^{\pi*} &= m_{0,j}^{\exp*} \frac{-\ln \pi_B}{1-\pi_B} = \frac{1}{\mu_j^B} \frac{-\ln \pi_B}{(1-\pi_B)} \\
&\quad \text{for } j = c+m2+1, \dots, \\
&\quad c+m2+N-1, \\
m_{i,j}^{\pi*} &= m_{i,j}^{\exp*} \frac{-\ln(\pi_A \pi_B)}{(1-\pi_A \pi_B)} \\
&= \frac{1}{(\mu_i^A + \mu_j^B)} \frac{-\ln(\pi_A \pi_B)}{(1-\pi_A \pi_B)} \\
&\quad \text{for } j = c+m2+1, \dots, \\
&\quad c+m2+N-1, \\
&\quad i = 1, \dots, N+m1+c \\
&\quad +m2-j-1.
\end{aligned}$$

In this set of expressions, all $m_{i,j}^{\pi*}$ can be transformed to $m_{i,j}$ (mean sojourn time) directly, by using (15).

Table 2. Comparison of the average number of tasks in both nodes.

λ	Tasks average number					q_{bl}	q_{tr}	Node A utiliz.	Node B utiliz.
	n_{bl}	l_A	v_A	l_B	v_B				
0.5	0.000	0.603	0.000	0.896	0.006	0.000	0.000	0.067	0.224
1.0	0.008	1.436	0.000	2.012	0.206	0.004	0.000	0.160	0.503
1.5	0.564	2.225	0.093	3.148	1.553	0.142	0.009	0.310	0.787
2.0	5.052	1.866	1.919	3.908	4.319	0.760	0.254	0.769	0.977
2.5	7.674	1.010	3.702	3.996	4.951	0.974	0.565	0.965	0.999
3.0	8.246	0.691	4.325	4.000	4.995	0.996	0.714	0.993	1.000
3.5	8.423	0.559	4.585	4.000	4.999	0.999	0.794	0.998	1.000
4.0	8.499	0.485	4.717	4.000	5.000	1.000	0.843	0.999	1.000
4.5	8.539	0.459	4.794	4.000	5.000	1.000	0.875	1.000	1.000
5.0	8.561	0.438	4.842	4.000	5.000	1.000	0.899	1.000	1.000

For the series of tandem model experiments, the following configuration is chosen: $N = 9$, $c = 4$, $m_1 = 5$, $m_2 = 8$, with the inter-arrival rate λ that changes within a range from 0.5 to 5.0 (for studying a model with different utilizations), and with the thinning parameter $\pi^A = 0.3$ for the node A and $\pi^B = 0.4$ for the node B. The renewal interval in both nodes is equal to $T = 2.0$. This model has 240 states; 150 states are without blocking and 90 states are with blocking.

For the above model with blocking and truncation some results were obtained, the majority of them being presented in Table 2. They show that the blocking probability, the truncation probability or filling buffers parameters rapidly grow when the inter-arrival rate to the tandem grows. In this case, the number of blocked lines in the first node quickly grows. Simultaneously, both buffers are quickly filled and the number of truncated tasks quickly increases. All these negative factors depend on the utilization parameter of the second node. The tandem works properly when the second node utilization parameter is less than 0.75.

5. Conclusion

A mathematical model of a two-node stochastic transition system with blocking and truncation, treated as a semi-Markov process, has been presented. The phenomenon of blocking and truncation appears simultaneously in this tandem model. Mathematical procedures allow calculating the main measures of effectiveness including blocking and truncation probabilities. These measures may be calculated for any tandem configuration if we have service rates in both nodes and when inter-arrival rates to the tandem are given.

The results of experiments presented in Section 4 show that, depending on the model configuration and its

characteristics, mathematical modelling makes it possible to find a proper rate range of the input stream. In the opposite case, given an input stream rate, the analysis allows finding another model characteristic, e.g., buffer capacities, which guarantees that blocking and truncation probabilities should be in the proper range.

The presented model can be exploited for capacity planning and performance evaluation of real-time computer networks or telecommunications networks in the case of blocking and truncation.

Acknowledgment

This work supported by the Białystok University of Technology under Grant No. W/WI/7/03.

References

- Akyildiz, I.F. (1988). Mean value analysis for blocking queueing networks, *IEEE Transactions on Software Engineering* **14**(4): 418–428.
- Balsamo, S. and de Nito Persone, V. (1994). A survey of product form queueing networks with blocking and their equivalences, *Annals of Operations Research* **48**(1/4): 31–61.
- Balsamo, S., de Nito Persone, V. and Onvural, R. (2001). *Analysis of Queueing Networks with Blocking*, Kluwer Academic Publishers, Boston, MA.
- Balsamo, S., de Nito Persone, V. and Inverardi, P. (2003). A review on queueing network models with finite capacity queues for software architectures performance predication, *Performance Evaluation* **51**(2-4): 269–288.
- Badrah, A., Czachórski, T., Domańska, J., Fourneau, J.-M. and Quessette, F. (2002). Performance evaluation of multistage interconnection networks with blocking—Discrete and continuous time Markov models, *Archiwum Informatyki Teoretycznej i Stosowanej* **14**(2): 145–162.

- Boucherie, R.J. and van Dijk, N.M. (1997). On the arrival theorem for product form queueing networks with blocking, *Performance Evaluation* **29**(3): 155–176.
- Bouhchouch, A., Frein, Y. and Dallery, Y. (1996). Performance evaluation of closed tandem queueing networks, *Performance Evaluation* **26**: 115–132.
- Bradley, J.T. and Davies, N.J. (2000). A matrix-based method for analysing stochastic process algebras, *Proceedings of the 8-th International Workshop on Process Algebra and Performance Modelling, ICALP Workshops, PAPM'00*, Geneva, Switzerland, pp. 579–590.
- Bradley, J.T. and Wilson, H.J. (2005). Iterative convergence of passage-time densities in semi-Markov performance models, *Performance Evaluation* **60**(1–4): 237–254.
- Clo, M.C. (1998). MVA for product-form cyclic queueing networks with blocking, *Annals of Operations Research* **79**: 83–96.
- Economou, A. and Fakinos, D. (1998). Product form stationary distributions for queueing networks with blocking and rerouting, *Queueing Systems* **30**(3/4): 251–260.
- Gomez-Corral, A. (2002). A tandem queue with blocking and Markovian arrival process, *Queueing Systems* **41**(4): 343–370.
- Heindl, A. (2003). Decomposition of general queueing networks with MMPP inputs and customer losses, *Performance Evaluation* **51**(2–4): 117–136.
- Kaufman, J.S. and Rege, K.M. (1996). Blocking in a shared resource environment with batched arrival processes, *Performance Evaluation* **24**(4): 249–263.
- Korolyuk, V.S. and Korolyuk, V.V. (1999). *Stochastic Models of Systems*, Kluwer Academic Publishers, Dordrecht.
- Kouvatsos, D. and Almond, J. (1988). Maximum entropy two-station cyclic queues with multiple general servers, *Acta Informatica* **26**(3): 241–267.
- Kouvatsos, D.D., Awan, I.U., Fretwell, R.J. and Dimakopoulos, G. (2000). A cost-effective approximation for SRD traffic in arbitrary multi-buffered networks, *Computer Networks* **34**(1):97–113.
- Martin, J.B. (2002). Large tandem queueing networks with blocking, *Queueing Systems* **41**(1/2): 45–72.
- Morrison, J.A. (1996). Blocking probabilities for multiple class batched arrivals to a shared resource, *Performance Evaluation* **25**(2): 131–150.
- Oniszczyk, W. (2006). Tandem models with blocking in the computer subnetworks performance analysis, in K. Saeed, J. Pejaš and R. Mosdorf (Eds.), *Biometrics, Computer Security Systems and Artificial Intelligence Applications*, Springer, New York, NY, pp. 259–267.
- Onvural, R. (1990). Survey of closed queueing networks with blocking, *Computer Survey* **22**(2): 83–121.
- Perros, H.G. (1994). *Queueing Networks with Blocking. Exact and Approximate Solution*, Oxford University Press, New York, NY.
- Ramesh, S. and Perros, H.G. (2000). A two-level queueing network model with blocking and non-blocking messages, *Annals of Operations Research* **93**(1/4): 357–372.
- Sereno, M. (1999). Mean value analysis of product form solution queueing networks with repetitive service blocking, *Performance Evaluation* **36–37**(1): 19–33.
- Sharma, V. and Virtamo, J.T. (2002). A finite buffer queue with priorities, *Performance Evaluation* **47**(1): 1–22.
- Strelen, J.Ch., Bärk, B., Becker, J. and Jonas, V. (1998). Analysis of queueing networks with blocking using a new aggregation technique, *Annals of Operations Research* **79**(0): 121–142.
- Tolio, T. and Gershwin, S.B. (1998). Throughput estimation in cyclic queueing networks with blocking, *Annals of Operations Research* **79**(0): 207–229.
- Zhuang, L., Buzacott, J.A. and Liu, X.-G. (1994). Approximate mean value performance analysis of cyclic queueing networks with production blocking, *Queueing Systems* **16**(1–2): 139–165.
- Zhuang, L. (1996). Acceptance instant distributions in product-form closed queueing networks with blocking, *Performance Evaluation* **26**(2): 133–144.

Walenty Oniszczyk received his M.Sc. and Ph.D. degrees in computer science in 1976 and 1980, respectively. Since 1980 he has been employed at the Faculty of Computer Science of Białystok Technical University (Poland). His main research interests include discrete-time queueing models with priority scheduling, queueing models with blocking, analytical investigation of large computer systems and computer networks, the concepts and theory of simulation plus simulation languages, traffic control, the application of hidden Markov models, etc.

Received: 14 June 2007

