Evaluating Dynamic Failure Probability for Streams with (*m*, *k*)-Firm Deadlines

Moncef Hamdaoui, *Member*, *IEEE*, and Parameswaran Ramanathan, *Member*, *IEEE*

Abstract—A real-time stream is said to have (m, k)-firm deadlines if at least m out of any k consecutive customers from the stream must meet their respective deadlines. Such a stream is said to have encountered a *dynamic failure* if fewer than m out of any k consecutive customers meet their deadlines. Hamdaoui and Ramanathan recently proposed a scheduling policy called Distance Based Priority (DBP) in which customers are serviced with a higher priority if their streams are closer to a dynamic failure. In terms of reducing the probability of dynamic failure, Hamdaoui and Ramanathan also showed, using simulation, that the DBP policy is better than a policy in which all customers are serviced at the same priority level.

In this paper, an analytic model is developed for computing the probability of dynamic failure of a real-time stream for the DBP and the single priority schemes. This model is useful for providing statistical quality of service guarantees to real-time streams. The probability of dynamic failure computed using this model is compared to the results from a discrete-event simulator. The comparison shows that the model is accurate for low and moderate loads.

Index Terms—Real-time systems, dynamic failure, priority queues, analytic modeling, quality of service guarantees.

1 Introduction

THE stringency of timing constraints distinguish real-time applications from non-real-time applications. In a real-time application, tasks have deadlines by which they are expected to complete their computation. Traditionally, most real-time applications were control applications in which the consequences of not meeting a deadline were very severe. However, in recent years, many new real-time applications have emerged in which it is not necessary to meet all the task and message deadlines as long as the misses are adequately spaced.

For example, in an anti-lock braking system, a real-time task typically determines the onset of locking by repeatedly sampling the rotational speed of each wheel. Since the speed of a wheel can be projected from a recent history of the speeds, it is usually not necessary for every instance of this task to complete its computation within the assigned deadline. However, if several consecutive instances of this task miss their deadlines, then the accuracy of the prediction becomes poor and the benefit of anti-lock braking is not realized. A similar situation occurs in a multimedia application where the video images are transmitted as a sequence of packets across nodes of a distributed system. To avoid distortion in the reconstructed image, these packets have deadline constraints by which they are expected to reach the destination. Occasionally, if some of these packets do not reach the destination on time, interpolation techniques can be used to satisfactorily reconstruct the image. However, if several consecutive packets miss their deadlines, then the quality of interpolation deteriorates and a vital portion of the image may be lost in the reconstruction process.

In both these applications, the computer system must not only limit the fraction of missed deadlines, but also ensure that the misses are adequately spaced. To represent such quality of service constraints, Hamdaoui and Ramanathan recently proposed a model called (m, k)-firm deadlines. Specifically, they define a *real-time stream* to be a sequence of related customers[1] with common deadline constraints. Furthermore, a real-time stream is said to have (m, k)-firm deadlines if at least m out of any k consecutive customers must meet their respective deadlines. If fewer than m out of k consecutive customers meet their deadlines, then the quality of service perceived by the stream is below acceptable limits and the stream is said to have experienced a *dynamic failure*. The problem, therefore, is to schedule competing customers from different streams in such a way that the probability of dynamic failure is as small as possible.

In [5], Hamdaoui and Ramanathan proposed a scheduling policy called *Distant-Based Priority Assignment* (DBP) for reducing the probability of dynamic failure. Using simulation, they show that their policy outperforms other policies in this respect. However, the main problem with their work is that they do not address the issue of providing quality of service guarantees to streams with (m, k)-firm deadlines. Since quality of service guarantees are essential for predictability in system performance, in this paper, we develop a method which can be used for providing such guarantees.

A commonly used approach for providing quality of service guarantees is to admit only streams whose requirements can be guaranteed without violating the guarantees promised earlier to other streams. This approach, however,

• *M. Hamdaoui is with Northern Telecom Inc., Tunisia, Immeuble Jugurtha, Berges du Lac, 2045 Tunis, Tunisie.*
• *P. Ramanathan is with the Department of Electrical and Computer Engineering, University of Wisconsin-Madison, Madison, WI 53706-1691. E-mail: parmesh@ece.wisc.edu.*

Manuscript received 28 June 1994.
For information on obtaining reprints of this article, please send e-mail to: tc@computer.org, and reference IEEECS Log Number 103692.

1. A customer can be either a task or a message depending on the application under consideration.

requires a performance characterization of the scheduling policy. Therefore, in this paper, we develop an analytic model for evaluating the expected probability of dynamic failure for an incoming stream given the other streams present in the system. We develop this model for two different scheduling policies, First-In-First-Out (FIFO) and Distance Based Priority Assignment (DBP). In developing this model, we make few simplifying assumptions about the policy. However, we show that the probability of dynamic failure predicted using this model is fairly close to the ones observed in a simulation without these assumptions.

Related work in literature are in the areas of real-time systems and high-speed networks. Hong et al. [6] develop an analytic to compute the steady-state probability of deadline miss for the Minimum Laxity and the Earliest Deadline First policies. Their analysis is over an infinite time horizon metric, as opposed a finite horizon metric in this paper. Analytic models for finite horizon metrics, called *interval QoS* and *block QoS*, are described in [12]. Although these metrics are very general, the models developed in [12] do not deal with customer deadlines; the models evaluate the fraction of customers lost over a finite horizon due to buffer overflows. More recently, Kant and Sanders [7] use Stochastic Activity Networks for characterizing the loss process within a switch of a high-speed network. The characterization is used to evaluate the distribution of consecutive cell losses. Here, again, deadline constraints are not considered.

The rest of the paper is organized as follows. A brief overview of the deadline model, the single priority scheme, and the distance-based priority scheme is presented in Section 2. The analytic model is developed and evaluated in Section 3. The paper concludes with Section 4.

2 System Model and Service Policies

In this paper, we consider a system with N customer streams, $R_1, R_2, \ldots, R_N$, and a single server. Customers from each stream are numbered 1, 2, 3, …, in the order of their arrival. These customers can be tasks, messages, or any other schedulable entities. Customers from the same stream are serviced in First-In First-Out order. This can be accomplished by maintaining a separate First-In First-Out queue for each stream. Only the heads of these queues are candidates for service. The selection of which stream to service when the server becomes idle depends on the policy being used [1], [2], [3], [4], [9], [10], [11], [13].

Each customer has a deadline before which it expects complete service. A customer *meets* the deadline if it is fully serviced before the deadline expires. Otherwise, the customer is said to have *missed* the deadline. We assume that stream R_j can tolerate, at most, $k_j - m_j$ deadline misses in any window of k_j consecutive customers, i.e., stream R_j has (m_j, k_j)-firm deadlines. The parameters m_j and k_j, therefore, specify a desired quality of service (QOS) for stream R_j. This QOS is violated when more than $k_j - m_j$ customers miss their deadlines in a window of k_j consecutive customers from stream R_j. When this occurs, we say that stream R_j experienced a dynamic failure. The probability of dynamic failure is therefore a measure of how often the QOS requirement is violated.

More formally, let $\mathtt{m}$ and $\mathtt{M}$ denote a deadline miss and a deadline meet, respectively, and let δ_i^j be a binary random variable denoting the status of the ith customer, $i \geq 1$, from stream R_j, i.e.,

$$\delta_i^j = \begin{cases} \mathtt{m} & \text{if the } i\text{th customer from } R_j \text{ misses its deadline,} \\ \mathtt{M} & \text{otherwise.} \end{cases} \tag{1}$$

Further, for convenience of presentation, we assume that $\delta_i^j = \mathtt{m}$ if $i \leq 0$. Then, stream R_j is said to have experienced a dynamic failure at the ith customer if

$$\sum_{l=i-k_j+1}^{i} I_{\{\delta_l^j = \mathtt{M}\}} < m_j, \tag{2}$$

where $I_{\{\delta_l^j = \mathtt{M}\}} = 1$ if $\delta_i^j = \mathtt{m}$ and $I_{\{\delta_l^j = \mathtt{M}\}} = 0$, otherwise. Let $P_{fail,i}^j$ denote the probability of stream R_j experiencing a dynamic failure at the ith customer, i.e.,

$$P_{fail,i}^j = \Pr\left[\sum_{l=i-k_j+1}^{i} I_{\{\delta_l^j = \mathtt{M}\}} < m_j\right]. \tag{3}$$

We assume that there exists a probability P_{fail}^j such that

$$\lim_{i \to \infty} P_{fail,i}^j = P_{fail}^j.$$

We refer to P_{fail}^j as the steady state probability of dynamic failure of stream R_j. The objective of this paper is to compute this probability under different service policies.

In this paper, we consider two policies for selecting the next customer to service: the single priority (SP) scheme and the distance-based priority (DBP) scheme [5]. In the SP scheme, all customers are serviced at the same priority level. When the server becomes idle, it selects the customer with the earliest arrival time, i.e., customers are serviced in First-In First-Out order. The DBP scheme is a dynamic priority assignment technique in which customers are assigned priorities based on the state of their corresponding streams. When the server becomes idle, it selects the customer with the highest priority. Within the same priority level, customers are serviced in First-In First-Out order.

The assignment of priorities in the DBP scheme works as follows (a more detailed description is given in [5]): The system maintains the state of each stream. The state of stream R_j captures the history of the k_j most recent customers from stream R_j. Namely, the state of stream R_j at a given time is the k_j-tuple $(\delta_{i-k_j+1}^j, \ldots, \delta_{i-1}^j, \delta_i^j)$, where i is the index of the most recent customer serviced from stream R_j. Stream R_j can therefore be in one of 2^{k_j} possible states. The states with fewer than m_j meets are called *failing states.* When a customer reaches the head of its stream queue (i.e., the customer is ready to be serviced), it is assigned a priority value. Let s be the current state of stream R_j. Then, the next customer from stream R_j is assigned a priority value equal to the *distance* from state s to a failing state of stream R_j. This distance is defined as the minimum number of consecutive misses required to take the stream from state s to a failing state. Customers with a lower priority value are serviced ahead of customers with a higher priority value.

The state of stream R_j is denoted by a k_j-letter string, where the right-most letter indicates the status of the last customer, the second letter from the right indicates the status of the second to last customer, and so on. For example, consider a stream with (1, 2)-firm deadlines. If the most recent customer from the stream has met its deadline and the customer before it has missed its deadline, then the stream is in state `mM`. If the next customer from the stream misses its deadline, the stream's state becomes `Mm`. The set of possible states for this stream is S = {`mm`, `mM`, `Mm`, `MM`}. Out of these states, `mm` is the only failing state. When the stream is in this state, the stream is already in a failing state and, so, its next customer is assigned a priority value of 0, i.e., the highest priority. When the stream is in state `Mm`, the stream is one miss away from a failing state, and, therefore, its next customer is assigned a priority value of 1. Finally, when the stream is in states `mM` or `MM`, the stream is two misses away from a failing state, and, therefore, its next customer is assigned a priority value of 2.

3 Analytic Model

In this section, we compute the probability of dynamic failure of stream R_j, $1 \le j \le N$, in both the SP and the DBP schemes. We define the following random variables:

- S_i^j — Service time of the ith customer from stream R_j.
- C_i^j — Interarrival time between customers i and $i + 1$ of stream R_j.
- D_i^j — Relative deadline of the ith customer from stream R_j. A customer with a relative deadline d is said to have *missed* its deadline (and is considered *lost*) if it is not completely serviced within d time units.
- X_i^j — System time (waiting time + service time) of the ith customer from stream R_j.
- Y_i^j — The total service time of the customers serviced between customers i and $i + 1$ of stream R_j (these customers, if any, are generated by streams other than R_j).

Let f_R and F_R denote the probability density function and the probability distribution function, respectively, of the random variable R. In the following analysis, we assume that, for each stream, customer interarrival times are independent and identically distributed (i.i.d.), customer service times are i.i.d., and customer deadlines are i.i.d.

The probability of dynamic failure depends on the probability that a customer misses its deadline. The difficulty in computing this probability stems from the fact that there is usually a strong correlation between the waiting times of consecutive customers. The probability that a customer misses its deadline, therefore, depends on whether previous customers met or missed their deadlines. For example, consider a recently generated customer. If the previous customer from the same stream encountered a long queue and missed its deadline, then it is likely that the current customer will also encounter a long queue and, thus, miss its deadline. In general, the probability of a customer missing its deadline depends on the status of every previous customer from the stream. In the following analysis, however, the probability of a customer missing (or meeting) its deadline is conditioned on the status of the previous customer (from the same stream), but not on earlier customers. It is shown later, through example, that the effect of earlier customers is negligible and it is, therefore, ignored. In summary, the analysis presented in this section is based on the following two assumptions:

A1. For each $j \in \{1, 2, \ldots, N\}$, $\{C_1^j, C_2^j, \ldots\}$, $\{S_1^j, S_2^j, \ldots\}$, and $\{D_1^j, D_2^j, \ldots\}$ are mutually independent. Furthermore, the random variables in each set are independent and identically distributed.

A2. For all $j = 1, 2, \ldots, N$ and for all $i = 1, 2, \ldots$,

$$\Pr\left[\delta_{i+1}^j = \mathtt{m} \,\middle|\, \delta_i^j, \delta_{i-1}^j, \delta_{i-2}^j, \ldots\right] = \Pr\left[\delta_{i+1}^j = \mathtt{m} \,\middle|\, \delta_i^j\right]. \quad (4)$$

We assume that a steady state exists and that the system reaches it.[2] We are interested in computing the probability of dynamic failure of stream R_j at steady state. We model stream R_j as a Markov chain with 2^{k_j} states, where a state indicates the status of the k_j most recent customers from R_j, as described earlier. Let $s = (\delta_{i-k_j+1}^j, \delta_{i-k_j+2}^j, \ldots, \delta_i^j)$ be the current state of stream R_j. When the next customer from stream R_j is serviced, the stream transits to one of two states, depending on whether the customer misses or meets the deadline. If the customer meets the deadline, the next state is

$$s \leftarrow \mathtt{M} \overset{\text{def}}{=} \left(\delta_{i-k_j+2}^j, \ldots, \delta_i^j, \mathtt{M}\right). \quad (5)$$

Otherwise, the next state is

$$s \leftarrow \mathtt{m} \overset{\text{def}}{=} \left(\delta_{i-k_j+2}^j, \ldots, \delta_i^j, \mathtt{m}\right). \quad (6)$$

The transition probabilities, therefore, correspond to the probability that the next customer meets its deadline and the probability that the next customer misses its deadline, respectively. Given the transition probabilities from each state, the Markov chain can be solved for the steady state distribution. The probability of dynamic failure can then be computed from the steady state distribution as

$$P_{fail}^j = \sum_{s \in S_0^j} \pi^j(s), \quad (7)$$

where $\pi^j(s)$ is the steady state probability that stream R_j is in state s and S_0^j is the set of failing states.

Without loss of generality, let R_1 be the stream of interest. In the remainder of this section, the superscript in the above defined variables is omitted and is assumed to be 1 unless otherwise specified.

2. Let $\pi_i^j(s)$ be the probability that stream R_j is in state s when the ith customer gets serviced. Then, we say that a steady state exists if, for each stream R_j and for each state s of R_j, there exists $\pi^j(s)$ such that $\pi_i^j(s) \to \pi^j(s)$ as $i \to \infty$.

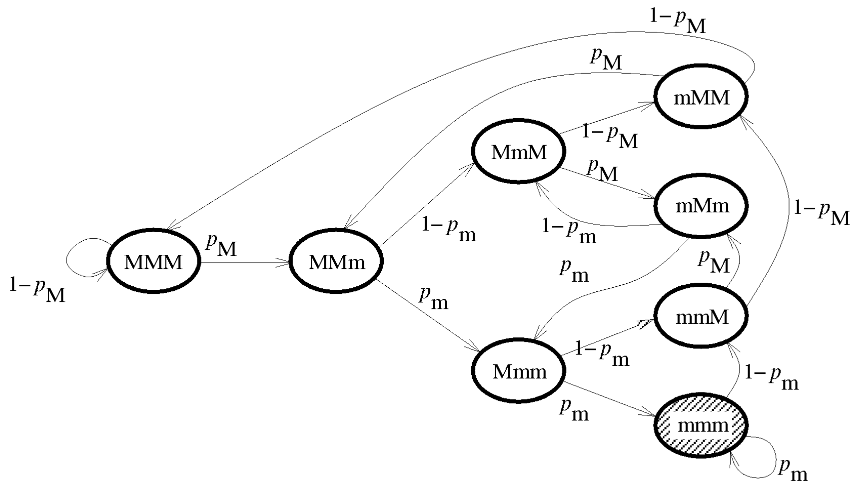

Fig. 1. Markov chain for a stream with (1, 3)-firm deadlines when the SP scheme is used.

3.1 Single Priority Scheme

In this policy, all customers are assigned the same priority, regardless of which state the corresponding stream is in. Fig. 1 shows the Markov chain for a stream with (1, 3)-firm deadlines. The probability $p_{\mathtt{m}}$ ($p_{\mathtt{M}}$) denotes the conditional probability of a customer missing its deadline given that the previous customer from the same stream missed (met) its deadline. The transition probabilities from a state are, therefore, either $p_{\mathtt{m}}$ and $1 - p_{\mathtt{m}}$, or $p_{\mathtt{M}}$ and $1 - p_{\mathtt{M}}$, depending on whether the most recent customer missed or met its deadline. For example, the transition probabilities from state $\mathtt{mMM}$ are $p_{\mathtt{M}}$ and $1 - p_{\mathtt{M}}$, since the last customer met its deadline.

A customer misses its deadline if its system time is greater than its deadline and meets its deadline otherwise. Consider a typical customer $i + 1$. The conditional probability that this customer misses its deadline given that the ith customer met its deadline can be written as

$$p_{\mathtt{M}} = \Pr\left[X_{i+1} > D_{i+1} \,\middle|\, X_i \le D_i\right] = 1 - \Pr\left[X_{i+1} \le D_{i+1} \,\middle|\, X_i \le D_i\right] = 1 - \frac{\Pr\left[X_{i+1} \le D_{i+1}, X_i \le D_i\right]}{\Pr\left[X_i \le D_i\right]}. \tag{8}$$

From the joint probability distribution, we can also compute the conditional probability that the $(i + 1)$th customer misses its deadline, given that the ith customer missed its deadline as follows:

$$p_{\mathtt{m}} = \Pr\left[X_{i+1} > D_{i+1} \,\middle|\, X_i > D_i\right] = 1 - \Pr\left[X_{i+1} \le D_{i+1} \,\middle|\, X_i > D_i\right] = 1 - \frac{\Pr\left[X_{i+1} \le D_{i+1}\right] - \Pr\left[X_{i+1} \le D_{i+1}, X_i \le D_i\right]}{\Pr\left[X_i > D_i\right]}. \tag{9}$$

We need to derive an expression for the joint probability $\Pr[X_{i+1} \le t_{i+1}, X_i \le t_i]$. Recall that customers are serviced in the order in which they arrive. When customer i is dequeued for service, all the earlier customers have already been serviced. After customer i is fully serviced, the server first services all the customers (from other streams) that arrived between customers i and $i + 1$, and, then, it services customer $i + 1$. We distinguish the following two cases:

1) Customer $i + 1$ arrives *before* customer i is fully serviced, i.e., $C_i \le X_i$, and
2) Customer $i + 1$ arrives *after* customer i is fully serviced, i.e., $C_i > X_i$, and consider each separately.

Let the time $t = 0$ denote the arrival time of customer i. (Customer $i + 1$, therefore, arrived at time C_i and customer i is fully serviced at time X_i.)

Case 1. $C_i \le X_i$
This case is depicted in Fig. 2a. Since customer $i + 1$ arrives before customer i is fully serviced, customer $i + 1$ has to wait $X_i - C_i$ until customer i is serviced and, then, wait Y_i while other customers (from other streams) are being serviced before it is dequeued for service. Its overall system time is, therefore, $X_{i+1} = X_i - C_i + Y_i + S_{i+1}$.

Case 2. $C_i > X_i$
In this case, customer $i + 1$ arrives $(C_i - X_i)$ time units after customer i is fully serviced, as depicted in Fig. 2b. During this time period ($[X_i, C_i]$), a portion Y_i' of Y_i ($0 \le Y_i' \le Y_i$) has also been serviced. Therefore, customer $i + 1$ has to wait for $Y_i - Y_i'$ before it is dequeued for service. Its system time is then $X_{i+1} = Y_i - Y_i' + S_{i+1}$. However, determining the exact value of Y_i' is difficult, since it depends on *when* each of the intermediate customers[3] arrives. For example, if all the intermediate customers arrive at C_i^-, i.e., right before customer $i + 1$ arrives, then $Y_i' = 0$. On the other hand, if they all arrive before X_i, i.e., before customer i is fully serviced, then $Y_i' = \min\{Y_i, C_i - X_i\}$. In our model, we approximate Y_i' as follows. If $Y_i > C_i - X_i$, then we approximate Y_i' as $C_i - X_i$, and, therefore, $X_{i+1} = Y_i - Y_i' + S_{i+1} \cong X_i - C_i + Y_i + S_{i+1}$. Otherwise, if $Y_i \le C_i - X_i$, we assume that all the intermediate customers are serviced before customer $i + 1$ arrives, i.e., $Y_i' \cong Y_i$. In this case, customer $i + 1$ arrives to an empty system (empty queue and idle server) and is, therefore, serviced right away; its system time equals its service time, S_{i+1}. These assumptions are reasonable, since all the inter-

3. The customers that arrive between customers i and $i + 1$ of the stream of interest.

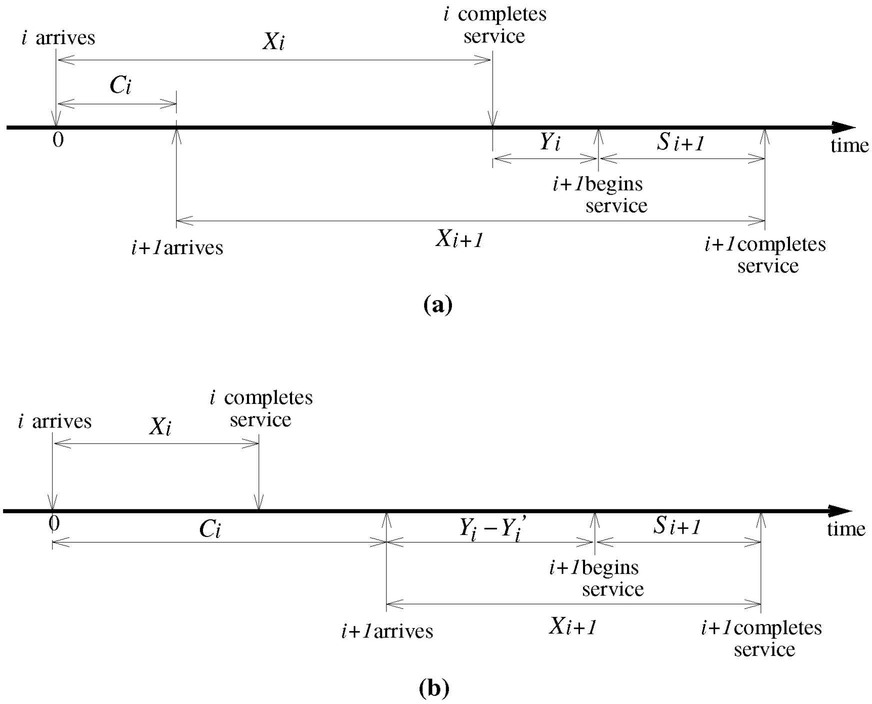

Fig. 2. System times of customers *i* and *i* + 1.

mediate customers that arrive during [0, X_i] are ready for service at X_i. Note that, in both cases, Y_i' is overestimated and that the approximation becomes more accurate as the load increases.

By combining the above two cases, the system time of customer i + 1 can be written as:

$$X_{i+1} \cong \begin{cases} S_{i+1} & \text{if } X_i + Y_i \le C_i \\ X_i + Y_i - C_i + S_{i+1} & \text{otherwise.} \end{cases} \quad (10)$$

It should be noted that, because Y_i' is overestimated, the above expression of X_{i+1} is a lower bound. Also, if there is only one stream in the system, we have $Y_i = 0$ and the above expression becomes exact. Using (10), we have

$$\begin{aligned} \Pr\left[X_i \le t_i, X_{i+1} \le t_{i+1}\right] &= \Pr\left[X_i \le t_i, X_{i+1} \le t_{i+1}, X_i + Y_i \le C_i\right] + \\ &\quad \Pr\left[X_i \le t_i, X_{i+1} \le t_{i+1}, X_i + Y_i > C_i\right] \\ &= \Pr\left[X_i \le t_i, S_{i+1} \le t_{i+1}, X_i + Y_i \le C_i\right] + \\ &\quad \Pr\left[X_i \le t_i, X_i + Y_i - C_i + S_{i+1} \le t_{i+1}, X_i + Y_i > C_i\right]. \quad (11) \end{aligned}$$

From assumption A1, C_i and S_{i+1} are independent of each other. Further, since the customers are serviced in First-In-First-Out order, X_i does not depend either on the service time S_{i+1} of the following customer or on the time of arrival C_i of the following customer. Thus, X_i, C_i, and S_{i+1} are mutually independent,[4] and, therefore,

$$\begin{aligned} &\Pr\left[X_i \le t_i, S_{i+1} \le t_{i+1}, X_i + Y_i \le C_i\right] \\ &= \Pr\left[S_{i+1} \le t_{i+1}\right] \cdot \Pr\left[X_i \le t_i, X_i + Y_i \le C_i\right] \\ &= \Pr\left[S_{i+1} \le t_{i+1}\right] \cdot \int_{x=0}^{t_i} \Pr\left[x + Y_i \le C_i\right] f_{X_i}(x)dx \\ &= \Pr\left[S_{i+1} \le t_{i+1}\right] \cdot \int_{x=0}^{t_i} \int_{c=x}^{\infty} \Pr\left[Y_i \le c - x\right] f_{C_i}(c) f_{X_i}(x) dc\, dx. \quad (12) \end{aligned}$$

Similarly, we have

$$\begin{aligned} &\Pr\left[X_i \le t_i, X_i + Y_i - C_i + S_{i+1} \le t_{i+1}, C_i < X_i + Y_i\right] \\ &= \int_{c=0}^{\infty} \Pr\left[X_i \le t_i, X_i + Y_i - c + S_{i+1} \le t_{i+1}, c < X_i + Y_i\right] \cdot \\ &\qquad f_{C_i}(c)dc \\ &= \int_{c=0}^{\infty} \int_{x=0}^{t_i} \Pr\left[x + Y_i - c + S_{i+1} \le t_{i+1}, c < x + Y_i\right] \cdot \\ &\qquad f_{X_i}(x) f_{C_i}(c) dx\, dc \\ &= \int_{c=0}^{\infty} \int_{x=0}^{t_i} \int_{s=0}^{\infty} \Pr\left[x + Y_i - c + s \le t_{i+1}, c < x + Y_i\right] \cdot \\ &\qquad f_{S_{i+1}}(s) f_{X_i}(x) f_{C_i}(c) ds\, dx\, dc \\ &= \int_{c=0}^{\infty} \int_{x=0}^{t_i} \int_{s=0}^{\infty} \Pr\left[c - x < Y_i \le t_{i+1} - s + c - x\right] \\ &\qquad f_{S_{i+1}}(s) f_{X_i}(x) f_{C_i}(c) ds\, dx\, dc. \quad (13) \end{aligned}$$

The joint probability distribution Pr[$X_{i+1} \le t_{i+1}$, $X_i \le t_i$] can now be evaluated by substituting (12) and (13) into (11). Given the joint probability distribution, the probabilities $p_{\mathtt{M}}$ and $p_{\mathtt{m}}$ are evaluated using (8) and (9) and the Markov chain is solved for the steady state distribution.

The analytic model assumes that the probability of a customer missing its deadline depends on the status of the previous customer, but not on the status of earlier customers from the corresponding stream. Consider, for example, the case when a stream is in state `MMM`, i.e., the last three customers from the stream met their deadlines. The probability of deadline miss, given that the last three customers met their deadlines, is smaller than the probability of deadline miss given that the last one customer met its deadline.

4. Note, however, that Y_i depends on C_i.

The probability of miss from this state is, therefore, overestimated in the analytic model. Since streams are often in this all-meet state, especially at lower loads, the model tends to over-estimate the probability of dynamic failure because of this assumption.

The second approximation is made when evaluating the joint probability $\Pr[X_{i+1} \le t_{i+1}, X_i \le t_i]$. The system time of customer $i + 1$, as given by (10), is an approximation of the actual system time. This approximation is a lower bound, which gets tighter as the load increases. Because the system time of customer $i + 1$ is underestimated, the joint probability $\Pr[X_{i+1} \le t_{i+1}, X_i \le t_i]$ is overestimated. The probability that a customer misses its deadline, given that the previous customer *met* its deadline (as given by (8)), is, therefore, underestimated. On the other hand, the probability that a customer misses its deadline, given that the previous customer *missed* its deadline, is overestimated (see (9)). It is not clear, however, whether the combined effect of this approximation over or under-estimates the probability of dynamic failure.

3.1.1 Example: An M/M/1 System

Consider, for example, the case where customer interarrival times and service times are exponentially distributed, with rates λ and μ, respectively, i.e., an M/M/1 system. Let λ_j be the customer arrival rate from stream j such that

$$\sum_{j=1}^{N} \lambda_j = \lambda .$$

Again, let R_1 be the stream of interest, and let $\lambda_r = \lambda - \lambda_1$. We have, for $t \ge 0$,

$$f_{C_i}(t) = \lambda_1 e^{-\lambda_1 t} \tag{14}$$

$$f_{S_i}(t) = \mu e^{-\mu t} . \tag{15}$$

At steady state (i.e., $i \gg 1$), the probability density function of the system time is ([8])

$$f_{X_i}(t) = \mu(1-\rho)e^{-\mu(1-\rho)t}, \quad t \ge 0 , \tag{16}$$

where $\rho = \lambda / \mu$. The probability distribution function of the system time is

$$F_{X_i}(t) = \Pr\left[X_i \le t\right] = 1 - e^{-\mu(1-\rho)t}, \quad t \ge 0. \tag{17}$$

Recall that Y_i represents the total service time of customers that arrive between customers i and $i + 1$ of stream stream R_1. It therefore depends on C_i, the interarrival time between customers i and $i + 1$. The probability distribution of Y_i, conditioned on C_i, is derived in Appendix A, and is given by

$$\Pr\left[Y_i \le t \,\middle|\, C_i = c\right] = 1 - e^{-\mu t} e^{-\lambda_r c} \sum_{n=0}^{\infty} \frac{(\mu t)^n}{n!} \sum_{k=n+1}^{\infty} \frac{(\lambda_r c)^k}{k!} \tag{18}$$

for all $t, c \ge 0$.

Equations (12) and (13) can now be evaluated using the above expressions for $f_{C_i}, f_{S_i}, f_{X_i}$, and $F_{Y_i|c}$. The details are shown in Appendix B. The transition probabilities can then be computed from p_{m} and p_{M}, as given by (8) and (9).

Numerical Example 1. Consider a system consisting of seven streams with (1, 3)-firm deadlines, such that $\lambda_1 = \cdots = \lambda_7 = 0.8/7$ ($\lambda = 0.8$). Let $\mu = 1.0$ and $D_i = 5$ for all i. Let $i \gg 1$. In this case, the probability of a customer missing its deadline is $\Pr[X_i > 5] = 1 - F_{X_i}(5) = 0.367$. However, the conditional probability that a customer misses its deadline given that the previous customer from the same stream missed its deadline is ((9))

$$p_{\mathrm{m}} = \Pr\left[X_{i+1} > D_{i+1} \,\middle|\, X_i > D_i\right] = 1 - \frac{\Pr\left[X_{i+1} \le 5\right] - \Pr\left[X_{i+1} \le 5, X_i \le 5\right]}{\Pr\left[X_i > 5\right]} = 0.793. \tag{19}$$

On the other hand, the conditional probability that a customer misses its deadline, given that the previous customer met its deadline, is ((8))

$$p_{\mathrm{M}} = \Pr\left[X_{i+1} > D_{i+1} \,\middle|\, X_i \le D_i\right] = 1 - \frac{\Pr\left[X_{i+1} \le 5, X_i \le 5\right]}{\Pr\left[X_i \le 5\right]} = 0.120. \tag{20}$$

Using the above values of p_{m} and p_{M}, the steady state distribution of the Markov chain, shown in Fig. 1, is (see Appendix C)

$$\begin{aligned} &\pi(\mathrm{mmm}) = 0.2318, \pi(\mathrm{mmM}) = 0.0602, \pi(\mathrm{mMm}) = 0.0091, \\ &\pi(\mathrm{mMM}) = 0.0668, \pi(\mathrm{Mmm}) = 0.0602, \pi(\mathrm{MmM}) = 0.0156 \\ &\pi(\mathrm{MMm}) = 0.0668, \pi(\mathrm{MMM}) = 0.4895. \end{aligned} \tag{21}$$

Since state mmm is the only failing state, we have

$$P_{fail} = \pi(\mathrm{mmm}) = 0.2318 . \tag{22}$$

The analytic model developed above is not exact. As indicated earlier, the errors are due to

1) the assumption that the probability of a customer missing its deadline depends on the status of the previous customer only, and
2) the approximation of the system time of customer $i + 1$ in terms of the system time of customer i, as given by (10).

The accuracy of the model is gauged by comparing the probability of dynamic failure as computed analytically to that obtained through simulation. The simulator takes as input the distributions of customer interarrival times, customer service times, and customer deadlines for each stream in the system. The system is then simulated and the steady state probability of dynamic failure is determined for each stream. Since the simulator computes the exact customer system times, the results obtained from the simulation are exact for the given distributions.

Fig. 3 shows plots of the probability of dynamic failure as computed analytically and as obtained through simulation for the system described in Example 1 when the streams have

1) (1, 3)-firm deadlines and
2) (2, 3)-firm deadlines.

The arrival rate, which is the same for each stream, is varied so that the overall load is varied from 0.2 to 0.9. When the deadlines are (1, 3)-firm, the probability of dynamic failure is $P_{fail} = \pi(\mathrm{mmm})$. However, when the deadlines are (2, 3)-firm, the set of failing states is S_0 = {mmm, mmM, mMm, Mmm} and, therefore,

$$P_{fail} = \pi(\text{mmm}) + \pi(\text{mmM}) + \pi(\text{mMm}) + \pi(\text{Mmm}). \quad (23)$$

As can be seen from the plots, the analytic model slightly overestimates the probability of dynamic failure for both the (1, 3)-firm and (2, 3)-firm cases. We also observe that, as the load increases, the discrepancy between the analytic and simulation results becomes smaller. At a load of 0.9, for example, the difference is less 2 percent.

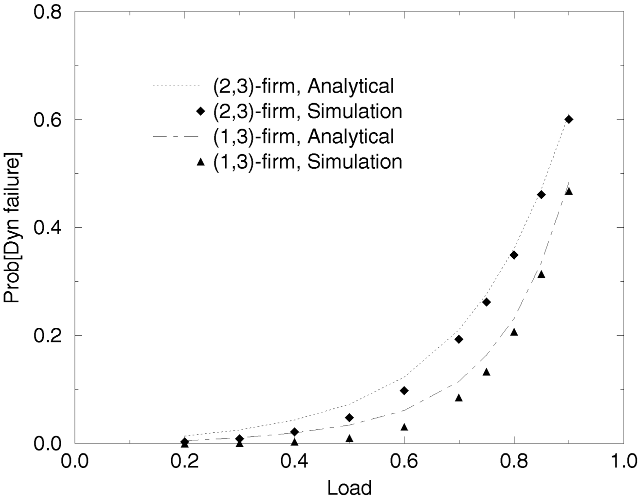

Fig. 3. The probability of dynamic failure as computed analytically and as obtained through simulation for the conventional single priority scheme.

Fig. 4 shows plots of the probability of dynamic failure in a system like the one examined above in Fig. 3, but with only one stream. Note that, in this case, since there is only one stream, (10) is an exact expression of the system time of customer $i + 1$. The only source of error in the analytic model is assumption A2, i.e., the probability that a customer misses its deadline depends on the status of the previous customer, but not on earlier customers. As the plots show, the predicted probability of dynamic failure closely matches the simulation results at all loads. This shows that, in this case, the effect of earlier customers is indeed negligible if one accounts for the status of the last customer, and, therefore, A2 is reasonable.

3.2 Distance-Based Priority Scheme

As described in Section 2, a customer from a stream with (m, k)-firm deadlines is assigned a priority l, where l is the minimum number of consecutive misses required to take the stream from its current state to a failing state. Recall that, for a stream with (m, k)-firm deadlines, the maximum distance to a failing state is $k - m + 1$, and, therefore, its customers are assigned a priority $l \in \{0, 1, \ldots, k - m + 1\}$. When the server becomes idle, it selects for service a customer with the highest priority, i.e., one with the lowest priority value.

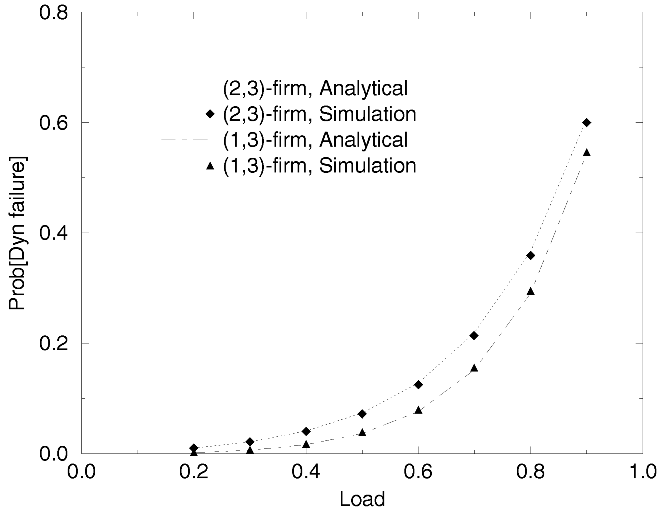

Fig. 4. The probability of dynamic failure as computed analytically and as obtained through simulation for the conventional single priority scheme in a system with one stream.

Let S^j be the set of all possible states of stream R_j. Let $S_l^j = \{s \in S^j : d^j(s) = l\}$, where $d^j(s)$ is the minimum number of consecutive misses required to take stream R_j from state s to a failing state. For example, if stream R_j has (1, 3)-firm deadlines, then

$$S_0^j = \{\text{mmm}\},$$
$$S_1^j = \{\text{Mmm}\},$$
$$S_2^j = \{\text{mMm, MMm}\},$$
$$S_3^j = \{\text{mmM, mMM, MmM, MMM}\}.$$

Note that S_0^j is the set of failing states of stream R_j. Let the customer arrivals from stream R_j form a Poisson process with rate λ_j and let $\lambda = \sum_{j=1}^{N} \lambda_j$ be the overall customer arrival rate. In the DBP scheme, a customer is assigned a priority l if the stream is in a state $s \in S_l^j$. The arrival rate of customers with priority l (from all streams) is, therefore,

$$\lambda^{(l)} = \sum_{j=1}^{N} \lambda_j \sum_{s \in S_l^j} \pi^j(s). \quad (24)$$

We assume that customer arrivals at priority level l forms a Poisson process with rate $\lambda^{(l)}$.

Let the customer service times be exponentially distributed with rate μ. Then the mean waiting time of a priority l customer is (Cobham's formula [8])

$$W^{(l)} = \frac{\lambda / \mu^2}{(1 - \sigma_l)(1 - \sigma_{l-1})}, \quad (25)$$

where $\sigma_l = \sum_{i=0}^{l} \frac{\lambda^{(i)}}{\mu}$.The mean system time of a priority l customer is, therefore,

$$X^{(l)} = W^{(l)} + \frac{1}{\mu} = \frac{\lambda/\mu^2}{(1-\sigma_l)(1-\sigma_{l-1})} + \frac{1}{\mu}. \tag{26}$$

However, because of the prioritized service, the exact probability distributions of customer system times are difficult to compute. We approximate the probability distribution function of priority l customer system times with an exponential distribution with a mean $X^{(l)}$. That is, given that customer i has priority l, we have

$$F_{X_i}(t \mid l) = \Pr[X_i \le t] = 1 - e^{-t/X^{(l)}}. \tag{27}$$

Again, we model each stream in the system as a Markov chain and compute the probability of dynamic failure from the steady state distribution, using (7). Consider stream R_j. Recall that, from a given state s, the stream transits to one of two possible states, depending on whether its next customer misses or meets the deadline. The transition probabilities to these two states are, therefore, p_s and $1 - p_s$, respectively, where p_s is the probability that the next customer from stream R_j misses its deadline, given that stream R_j is currently in state s. More formally, the transition probability from state $s \in \mathcal{S}^j$ to state $s' \in \mathcal{S}^j$ is given by

$$P^j_{(s,s')} = \begin{cases} p_s & \text{if } s' = s \leftarrow \mathtt{m} \\ 1 - p_s & \text{if } s' = s \leftarrow \mathtt{M} \\ 0 & \text{otherwise.} \end{cases} \tag{28}$$

Because of the prioritized service, the probability that a customer misses its deadline, p_s, depends on the priority at which the customer is serviced. Also, as discussed earlier for the case of the single priority scheme, this probability depends on the status of the previous customers. However, the procedure for computing the probability of miss developed for the single priority scheme is not always applicable here, since, in general, consecutive customers from the same stream may be serviced at *different* priority levels. The degree of correlation between the system times of consecutive customers depends on the relative priorities of the customers. For example, consider a customer with a priority l. If the previous customer had the same priority and it met its deadline, then the current customer is likely to meet its deadline. On the other hand, suppose that the previous customer had a much higher priority. Then, the fact that it met its deadline does not say as much about whether the current customer will meet its deadline. The probability that a customer misses its deadline, therefore, depends on the status of the previous customer <u>and</u> on the priority at which the previous customer was serviced. Recall that the priority at which the previous customer is serviced is a function of the previous state of the stream. The probability that a customer misses its deadline, p_s, must, therefore, be conditioned on the previous state of the stream. For convenience, let CS and PS stand for current state and previous state, respectively. The probability p_s can be written as

$$\begin{aligned} p_s &= \Pr[\mathtt{miss} \mid \mathrm{CS} = s] \\ &= \sum_{s_p \in \mathcal{S}^j} \Pr[\mathtt{miss} \mid \mathrm{CS} = s,\ \mathrm{PS} = s_p] \cdot \Pr[\mathrm{PS} = s_p \mid \mathrm{CS} = s], \end{aligned} \tag{29}$$

and therefore, for all $s, s' \in \mathcal{S}^j$,

$$P^j_{(s,s')} = \begin{cases} \sum_{s_p \in \mathcal{S}^j} \Pr[\mathtt{miss} \mid \mathrm{CS} = s,\ \mathrm{PS} = s_p] \cdot \Pr[\mathrm{PS} = s_p \mid \mathrm{CS} = s] & \text{if } s' = s \leftarrow \mathtt{m} \\ 1 - \sum_{s_p \in \mathcal{S}^j} \Pr[\mathtt{miss} \mid \mathrm{CS} = s,\ \mathrm{PS} = s_p] \cdot \Pr[\mathrm{PS} = s_p \mid \mathrm{CS} = s] & \text{if } s' = s \leftarrow \mathtt{M} \\ 0 & \text{otherwise.} \end{cases} \tag{30}$$

The conditional probability $\Pr[\mathrm{PS} = s_p | \mathrm{CS} = s]$ can be rewritten as

$$\begin{aligned} \Pr[\mathrm{PS} = s_p \mid \mathrm{CS} = s] &= \frac{\Pr[\mathrm{PS} = s_p,\ \mathrm{CS} = s]}{\pi^j(s)} \\ &= \frac{\Pr[\mathrm{CS} = s \mid \mathrm{PS} = s_p] \cdot \pi^j(s_p)}{\pi^j(s)} \\ &= P^j_{(s_p,s)} \cdot \frac{\pi^j(s_p)}{\pi^j(s)}. \end{aligned} \tag{31}$$

Substituting the above expression in (30), we get the following system of linear equations[5]:

$$P^j_{(s,s')} = \begin{cases} \sum_{s_p \in \mathcal{S}^j} \Pr[\mathtt{miss} \mid \mathrm{CS} = s,\ \mathrm{PS} = s_p] \cdot \left(\pi^j(s_p)/\pi^j(s)\right) \cdot P^j_{(s_p,s)} & \text{if } s' = s \leftarrow \mathtt{m} \\ 1 - \sum_{s_p \in \mathcal{S}^j} \Pr[\mathtt{miss} \mid \mathrm{CS} = s,\ \mathrm{PS} = s_p] \cdot \left(\pi^j(s_p)/\pi^j(s)\right) \cdot P^j_{(s_p,s)} & \text{if } s' = s \leftarrow \mathtt{M} \\ 0 & \text{otherwise} \end{cases} \tag{32}$$

for all $s, s' \in \mathcal{S}^j$. Given the probabilities $\Pr[\mathtt{miss} \mid \mathrm{CS} = s, \mathrm{PS} = s_p]$ (a procedure for evaluating these probabilities is presented later in this section) and the steady state distribution, the above system can be solved for the transition probabilities. However, the steady state distribution, in turn, depends on the transition probabilities.[6] Because of this interdependence, an iterative approach is needed to solve for the steady state distribution. Such a procedure is shown in Fig. 5. The leading superscript indicates the iteration number; ${}^{(i)}P^j_{(s,s')}$ and ${}^{(i)}\pi^j$ are the transition probabilities and the steady state distributions as computed in the ith iteration. Initially, we assume that all customers are serviced at the same priority level, in which case, the procedure described in Section 3.1 can be used to compute the transition probabilities ${}^{(0)}P^j_{(s,s')}$. The resulting steady state distributions, ${}^{(0)}\pi^j$, are then used as an initial solution for

5. Note that, given a current state s, there are only two possible previous states. Let s_1 and s_2 be these two possible states. We have $\Pr[\mathrm{PS} = s_p \mid \mathrm{CS} = s] = 0$ for all $s_p \notin \{s_1, s_2\}$, and $\Pr[\mathrm{PS} = s_1 \mid \mathrm{CS} = s] + \Pr[\mathrm{PS} = s_2 \mid \mathrm{CS} = s] = 1$. Furthermore, in most cases, we have $\Pr[\mathtt{miss} \mid \mathrm{CS} = s, \mathrm{PS} = s_1] = \Pr[\mathtt{miss} \mid \mathrm{CS} = s, \mathrm{PS} = s_2]$, in which case, we have $p_s = \Pr[\mathtt{miss} \mid \mathrm{CS} = s, \mathrm{PS} = s_1]$. In these cases, the transition probabilities from state s can be directly computed from $\Pr[\mathtt{miss} \mid \mathrm{CS} = s, \mathrm{PS} = s_1]$.

6. In fact, the probability $\Pr[\mathtt{miss} \mid \mathrm{CS} = s, \mathrm{PS} = s_p]$ also depends on the steady state distribution. The probability that a customer misses its deadline depends on the priority level at which the customer is serviced and on the effective load at each priority level. However, since the priority assigned to a customer depends on the state of the stream, the effective load at each priority level depends on the steady state distribution.

```
/* find an initial solution */
```
for $j = 1, 2, \ldots, N$,

- Compute the transition probabilities assuming all customers serviced at the same priority level, i.e., **for all** $s, s' \in S^j$,

$$ {}^{(0)}P^j_{(s,s')} = \begin{cases} p_{\mathtt{m}} & \text{if } (s' = s \leftarrow \mathtt{m}) \text{ and } \left(\exists s_p \in S^j \text{ such that } s = s_p \leftarrow \mathtt{m}\right) \\ p_{\mathtt{M}} & \text{if } (s' = s \leftarrow \mathtt{m}) \text{ and } \left(\exists s_p \in S^j \text{ such that } s = s_p \leftarrow \mathtt{M}\right) \\ 1 - p_{\mathtt{m}} & \text{if } (s' = s \leftarrow \mathtt{M}) \text{ and } \left(\exists s_p \in S^j \text{ such that } s = s_p \leftarrow \mathtt{m}\right) \\ 1 - p_{\mathtt{M}} & \text{if } (s' = s \leftarrow \mathtt{M}) \text{ and } \left(\exists s_p \in S^j \text{ such that } s = s_p \leftarrow \mathtt{M}\right) \\ 0 & \text{otherwise} \end{cases} $$

- Solve for the steady state distribution ${}^{(0)}\pi^j$

```
/* iterate */
```
$i \leftarrow 0$ `/* iteration number */`

Repeat

$i \leftarrow i + 1$

for $j = 1, 2, \ldots, N$,

- Compute the transition probabilities **for all** $s, s' \in S^j$,

$$ {}^{(i)}P^j_{(s,s')} = \begin{cases} \sum_{s_p \in S^j} T\left(s, s_p\right) & \text{if } s' = s \leftarrow \mathtt{m} \\ 1 - \sum_{s_p \in S^j} T\left(s, s_p\right) & \text{if } s' = s \leftarrow \mathtt{M} \\ 0 & \text{otherwise,} \end{cases} $$

where $T\left(s, s_p\right) = \Pr\left[\mathtt{miss} \middle| \text{ CS} = s,\ \text{PS} = s_p\right] \cdot \frac{{}^{(i-1)}\pi^j\left(s_p\right)}{{}^{(i-1)}\pi^j(s)} \cdot {}^{(i-1)}P^j_{\left(s_p, s\right)}$

- Solve for the steady state distribution ${}^{(i)}\pi^j$

Until $\left(\forall j, \forall s \in S^j, \left|\frac{{}^{(i-1)}\pi^j(s) - {}^{(i)}\pi^j(s)}{{}^{(i)}\pi^j(s)}\right| \le \epsilon\right)$

Fig. 5. Procedure for computing the steady state distributions.

the iterative procedure. The transition probabilities ${}^{(1)}P^j_{(s,s')}$ are then recomputed assuming the steady state distributions ${}^{(0)}\pi^j$. The steady state distributions ${}^{(1)}\pi^j$ are then computed from the transition probabilities ${}^{(1)}P^j_{(s,s')}$. This procedure is repeated until, for every state s, the percent difference between $\pi(s)$ from the current iteration and $\pi(s)$ from the previous iteration is below some tolerance ϵ. In practice, the procedure converges quickly and is terminated within a few iterations.

3.2.1 Computing Pr[*miss* | CS = *s*, PS = s_p]

Suppose the stream of interest is currently in state s. Its customer will therefore be serviced at priority level $d(s)$. Also, suppose that the previous state of the stream is s_p. The previous customer from the stream was therefore serviced at priority level $d(s_p)$. As discussed earlier, the degree of correlation between the system times of the previous and current customers depends on the relative order of their priorities, $d(s_p)$ and $d(s)$. Without loss of generality, let i and $i + 1$ be the indices of the previous and current customer, respectively, and let l_i and l_{i+1} be the priorities at which they are serviced, i.e., $l_i = d(s_p)$ and $l_{i+1} = d(s)$. We identify the following three cases based on the relative order of l_i and l_{i+1} ($l_i < l_{i+1}$, $l_i = l_{i+1}$, and $l_i > l_{i+1}$) and consider each separately.

<u>*Priority dropped*</u>: The priority assigned to the $(i + 1)$th customer is strictly lower than the priority assigned to the ith customer, i.e., $l_{i+1} > l_i$. Note that this may occur only when the previous customer meets its deadline. Also, the difference between l_i and l_{i+1} can be up to $k_1 - m_1 + 1$. In general, the fact that a customer with a *high* priority met its deadline tells us very little about whether or not the next customer, with a possibly *much lower* priority, will meet its deadline. In this case, the probability that the $(i + 1)$th customer misses its deadline is computed as the unconditional probability that a customer with priority l_{i+1} misses its deadline. In other words, we assume that X_{i+1} depends only on the priority at which the $(i + 1)$th customer is serviced, but not on X_i. The probability that customer i+1 misses its deadline is computed as

$$ p^{(l_{i+1})} = \Pr\left[X_{i+1} > D_{i+1} \middle| l_{i+1}\right] = 1 - F_{X_{i+1}}\left(D_{i+1} \middle| l_{i+1}\right), \quad (33) $$

where $F_{X_{i+1}}\left(D_{i+1} \middle| l_{i+1}\right)$ is as given by (26) and (27). Since a dropped priority implies that the previous customer *met* its deadline, one would expect that the probability that the next customer misses its deadline is lower than the unconditional probability of miss. Therefore, the expected effect of this assumption is to over-estimate the probability of dynamic failure.

<u>Same priority</u>: The $(i + 1)$th customer is assigned a priority equal to that of the ith customer, i.e., $l_{i+1} = l_i = l$. This occurs, for example, when the ith customer has the lowest priority and it meets its deadline. We denote the probability that the $(i + 1)$th customer misses its deadline by $p_{\mathtt{m}}^{(l)}$ or $p_{\mathtt{M}}^{(l)}$, depending on whether the ith customer missed or met its deadline, respectively. Since both customers have the same priority, the probabilities $p_{\mathtt{m}}^{(l)}$ and $p_{\mathtt{M}}^{(l)}$ can be computed using the procedure developed for the single priority scheme in Section 3.1. The procedure requires knowledge of the distribution of Y_i. All the customers with priority $u \in \{0, 1, \ldots, l\}$ that arrive between customers i and $i + 1$ are serviced before customer $i + 1$, and, therefore, the effective arrival rate for computing Y_i is

$$\Lambda_l = \sum_{u=0}^{l} \lambda^{(u)}, \tag{34}$$

where $\lambda^{(u)}$ is the arrival rate of priority u customers.

<u>Priority raised</u>: The priority assigned to the $(i + 1)$th customer is one higher than the priority assigned to the ith customer, i.e., $l_{i+1} = l_i - 1$. This occurs when the ith customer, with a priority $l_i > 0$, misses its deadline. Note that, in the DBP scheme, the priority can be raised by at most one level from one customer to the next consecutive customer. We denote the conditional probability that a customer misses its deadline at priority $l_i - 1$ given that the previous customer missed its deadline at priority l_i by $p_{\mathtt{M}}^{(l_i, l_i-1)}$. We have

$$\begin{aligned} p_{\mathtt{M}}^{(l_i, l_i-1)} &= \Pr\left[X_{i+1} > D_{i+1} \,\middle|\, X_i > D_i, l_{i+1}, l_i\right] \\ &= \frac{\Pr\left[X_{i+1} > D_{i+1}, X_i > D_i \,\middle|\, l_{i+1}, l_i\right]}{\Pr\left[X_i > D_i, \,\middle|\, l_i\right]}. \end{aligned} \tag{35}$$

We now derive an approximate expression for $\Pr[X_{i+1} > D_{i+1} \mid X_i > D_i, l_{i+1}, l_i]$. In all the expressions that follow, it is implied that customers i and $i + 1$ are serviced at priorities l_i and l_{i+1}, respectively. For compactness of notation, this will not be explicitly indicated.

Consider the case when customer $i + 1$ arrives after customer i is serviced,[7] i.e., $C_i > X_i$. In this case, the interarrival time is large (X_i is large, since customer i missed its deadline) and, therefore, we neglect the correlation between the system times of customers i and $i + 1$ and assume that the system time of customer $i + 1$ (X_{i+1}) is independent of X_i. Now, consider the case when customer $i + 1$ arrives before customer i is serviced, i.e., $C_i \le X_i$. Let the time $t = 0$ denote the arrival time of customer i. (Customer $i + 1$, therefore, arrived at time C_i and customer i received service at time X_i.) Because of the prioritized service, customer i, with a priority l_i, waits behind customers in priority queues 0, 1, …, l_i. Since $C_i \le X_i$, customer i receives service $X_i - C_i$ time units after customer $i + 1$ arrives. Assuming that no other customers arrived to queues 0, 1, …, $l_i - 1$ during the time period $[C_i, X_i]$, the cumulative length of queues 0, 1, …, l_i seen by customer i at time C_i is $X_i - C_i$. At this time, customer $i + 1$ joins queue $l_{i+1} = l_i - 1$ and has to wait behind customers in queues 0, 1, …, $l_i - 1$. The cumulative length of queues 0, 1, …, $l_i - 1$ is a fraction of that of queues 0, 1, …, l_i. We estimate this fraction as the ratio of the average system time a priority $l_{i+1} = l_i - 1$ customer to that of a priority l_i customer. In this case ($C_i \le X_i$), the system time of customer $i + 1$ is approximated as

$$X_{i+1} = \frac{X^{(l_{i+1})}}{X^{(l_i)}} \cdot \left(X_i - C_i\right) + S_{i+1}, \tag{36}$$

where $X^{(l)}$ is the average system time of a priority l customer, as given by (26). The probability $\Pr[X_{i+1} > D_{i+1}, X_i > D_i]$ can, therefore, be written as

$$\begin{aligned} \Pr\left[X_{i+1} > D_{i+1}, X_i > D_i\right] &= \Pr\left[X_{i+1} > D_{i+1}, X_i > D_i, C_i > X_i\right] + \\ &\quad \Pr\left[X_{i+1} > D_{i+1}, X_i > D_i, C_i \le X_i\right] \\ &= \Pr\left[X_{i+1} > D_{i+1}\right] \cdot \Pr\left[X_i > D_i, C_i > X_i\right] + \\ &\quad \Pr\left[r\left(X_i - C_i\right) + S_{i+1} > D_{i+1}, X_i > D_i, C_i \le X_i\right], \end{aligned} \tag{37}$$

where $r = \frac{X^{(l_{i+1})}}{X^{(l_i)}}$. Assuming that X_i and D_i are mutually independent,[8] we have

$$\Pr\left[X_i > D_i, C_i > X_i\right] = \int_{u=0}^{\infty} \int_{c=u}^{\infty} \int_{x=u}^{c} f_{X_i}(x) f_{C_i}(c) f_{D_i}(u) dx\, dc\, du.$$

Similarly, we have

$$\begin{aligned} &\Pr\left[r\left(X_i - C_i\right) + S_{i+1} > D_{i+1}, X_i > D_i, C_i \le X_i\right] \\ &= \int_{u=0}^{\infty} \int_{x=u}^{\infty} \Pr\left[r\left(x - C_i\right) + S_{i+1} > D_{i+1}, C_i \le x\right] f_{X_i}(x) f_{D_i}(u) dx\, du \\ &= \int_{u=0}^{\infty} \int_{x=u}^{\infty} \int_{c=0}^{x} \Pr\left[S_{i+1} > D_{i+1} - r(x - c)\right] f_{C_i}(c) f_{X_i}(x) f_{D_i}(u) dc\, dx\, du \\ &= \int_{u=0}^{\infty} \int_{x=u}^{\infty} \int_{c=0}^{x} \int_{v=0}^{\infty} \Pr\left[S_{i+1} > v - r(x - c)\right] \cdot \\ &\qquad f_{D_{i+1}}(v) f_{C_i}(c) f_{X_i}(x) f_{D_i}(u) dv\, dc\, dx\, du. \end{aligned} \tag{38}$$

The probability $\Pr[X_{i+1} > D_{i+1}, X_i > D_i]$ is evaluated by substituting the above two equations in (37). The probability $p_{\mathtt{m}}^{(l_i, l_i-1)}$ can then be evaluated using (35).

The probability $\Pr[\mathtt{miss} \mid \mathrm{CS} = s, \mathrm{PS} = s_p]$ can now be computed using one of the above three cases. Namely, if s_p is a possible previous state to state s, i.e., $s = s_p \leftarrow \mathtt{m}$ or $s = s_p \leftarrow \mathtt{M}$, then

$$\Pr\left[\mathtt{miss} \,\middle|\, \mathrm{CS} = s,\ \mathrm{PS} = s_p\right] = \begin{cases} p^{(d(s))} & \text{if } d(s) > d\left(s_p\right) \\ p_{\mathtt{m}}^{(d(s))} & \text{if } d(s) = d\left(s_p\right) \text{ and } s = s_p \leftarrow \mathtt{m} \\ p_{\mathtt{M}}^{(d(s))} & \text{if } d(s) = d\left(s_p\right) \text{ and } s = s_p \leftarrow \mathtt{M} \\ p_{\mathtt{m}}^{\left(d\left(s_p\right), d(s)\right)} & \text{if } d(s) = d\left(s_p\right) - 1. \end{cases} \tag{39}$$

7. Ideally, customer i should be dropped without service as soon as its deadline expires. However, for convenience, we will still refer to X_i as its system time, whether or not customer i is actually serviced. If customer i is dropped, X_i can be thought of as customer i's system time had it not been dropped.

8. It should be noted here that customers from the same stream usually have the same, fixed, deadline, in which case X_i and D_i are independent.

The transition probabilities can now be computed. For example, consider a stream with (1, 3)-firm deadlines and consider the state s = MMm. This state is two misses away from a failing state, i.e., d(MMm) = 2. The state s_p = MMM is a possible previous state with d(MMM) = 3. Since d(MMm) = d(MMM) – 1, we have

$$\Pr\left[\mathtt{miss}\middle|\ \mathrm{CS} = \mathtt{Mmm},\ \mathrm{PS} = \mathtt{MMM}\right] = p_{\mathtt{m}}^{(3,2)}. \tag{40}$$

The state mMM is the other possible previous state. We also have d(mMM) = d(MMM) – 1, and therefore,

$$\Pr\left[\mathtt{miss}\middle|\ \mathrm{CS} = \mathtt{MMm},\ \mathrm{PS} = \mathtt{mMM}\right] = p_{\mathtt{m}}^{(3,2)}. \tag{41}$$

Since mMM and MMM are the only two possible previous states, we have Pr[PS = s_p | CS = MMm] = 0 for all $s_p \notin$ {mMM, MMM}. Since Mmm = MMm $\leftarrow$ m, the transition probability from state MMm to state Mmm is (given by (30))

$$\begin{aligned}
&P_{(\mathtt{MMm},\mathtt{Mmm})} = \\
&\Pr\left[\mathtt{miss}\middle|\ \mathrm{CS} = \mathtt{MMm},\ \mathrm{PS} = \mathtt{MMM}\right] \cdot \Pr\left[\mathrm{PS} = \mathtt{MMM}\middle|\ \mathrm{CS} = \mathtt{MMm}\right] + \\
&\Pr\left[\mathtt{miss}\middle|\ \mathrm{CS} = \mathtt{MMm},\ \mathrm{PS} = \mathtt{mMM}\right] \cdot \Pr\left[\mathrm{PS} = \mathtt{mMM}\middle|\ \mathrm{CS} = \mathtt{MMm}\right] \\
&= p_{\mathtt{m}}^{(3,2)} \cdot \left(\Pr\left[\mathrm{PS} = \mathtt{MMM}\middle|\ \mathrm{CS} = \mathtt{MMm}\right] + \Pr\left[\mathrm{Ps} = \mathtt{mMM}\middle|\ \mathrm{CS} = \mathtt{MMm}\right]\right) \\
&= p_{\mathtt{m}}^{(3,2)}.
\end{aligned} \tag{42}$$

The other transition probabilities can be computed similarly.

Numerical Example 2. Consider the system of Example 1 in which there are seven streams with (1, 3)-firm deadlines. We have $\lambda_1 = \cdots = \lambda_7 = 0.8/7$, $\mu = 1.0$, and $D_i = 5$ for all i. Using the iterative procedure described earlier, we compute the probability of dynamic failure at steady state ($i \gg 1$) when the DBP scheme is used.

We first compute the steady state distribution of the Markov chain, assuming that all customers are serviced at the same priority level. This distribution is then used as an initial solution for the iterative procedure. We have (see Example 1)

$$\begin{gathered}
\pi(\mathtt{mmm}) = 0.2318,\ \pi(\mathtt{mmM}) = 0.0602,\ \pi(\mathtt{mMm}) = 0.0091, \\
\pi(\mathtt{mMM}) = 0.0668,\ \pi(\mathtt{Mmm}) = 0.0602,\ \pi(\mathtt{MmM}) = 0.0156, \\
\pi(\mathtt{MMm}) = 0.0668,\ \pi(\mathtt{MMM}) = 0.4895.
\end{gathered} \tag{43}$$

From the steady state distribution, we have

$$\lambda^{(0)} = 0.1854,\ \lambda^{(1)} = 0.0481,\ \lambda^{(2)} = 0.0607,\ \lambda^{(3)} = 0.5056. \tag{44}$$

We have

$$\begin{gathered}
p_{\mathtt{m}}^{(1,0)} = 0.2739,\ p_{\mathtt{m}}^{(2,1)} = 0.3210,\ p_{\mathtt{m}}^{(3,2)} = 0.2394,\ p^{(3)} = 0.4724, \\
p_{\mathtt{m}}^{(0)} = 0.8858,\ \text{and}\ p_{\mathtt{M}}^{(3)} = 0.1239
\end{gathered},$$

and the transition probabilities can now be computed. The steady state distribution is then recomputed using the new values for the transition probabilities. After five iterations, the distribution converges to

$$\begin{gathered}
\pi(\mathtt{mmm}) = 0.0129,\ \pi(\mathtt{mmM}) = 0.0236,\ \pi(\mathtt{mMm}) = 0.0582, \\
\pi(\mathtt{mMM}) = 0.0825,\ \pi(\mathtt{Mmm}) = 0.0236,\ \pi(\mathtt{MmM}) = 0.1171, \\
\pi(\mathtt{MMm}) = 0.0825,\ \pi(\mathtt{MMM}) = 0.5996,
\end{gathered} \tag{45}$$

and we have

$$P_{fail} = \pi(\mathtt{mmm}) = 0.0129. \tag{46}$$

Fig. 6 shows plots of the probability of dynamic failure as computed analytically and as obtained through simulation for the system described in Example 2 when the streams have

1) (1, 3)-firm deadlines and
2) (2, 3)-firm deadlines.

The dotted curves show the probability of dynamic failure as computed analytically (the iterative procedure is terminated when the change in the distributions (ϵ) in less than 1 percent.) For all the cases studied, the iterative procedure converges very quickly and is terminated within four to five iterations. The model underestimates the probability of dynamic failure for most loads. At higher loads, however, the probability of dynamic failure is overestimated.

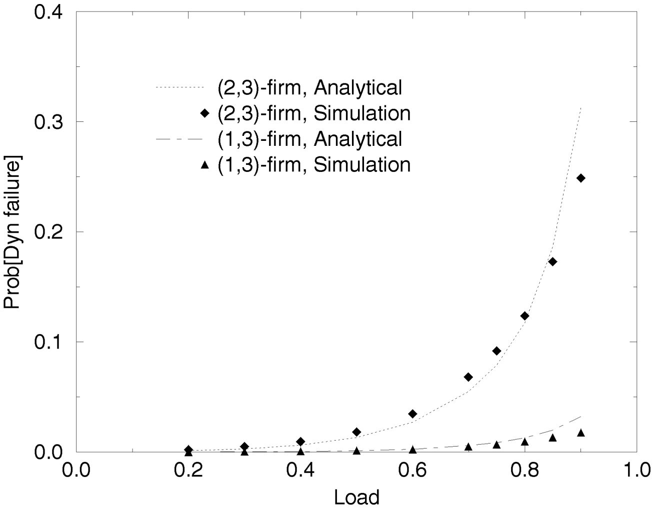

Fig. 6. The probability of dynamic failure as computed analytically and as obtained through simulation for the DBP scheme.

4 Conclusion

The rate at which a real-time customer stream experiences dynamic failure is a measure of how often the minimum requirements are violated. In [5], we compared several service policies in terms of the resulting rate of dynamic failure. It was shown, through simulation, that the distance-based priority scheme results in a substantially lower probability of dynamic failure compared to the conventional single priority scheme. In this paper, we developed an analytic model to compute the probability of dynamic failure in both the single priority scheme and the distance-based priority scheme. The model specifically deals with the correlation between the system times of consecutive customers. The predicted probability of dynamic failure is compared to that obtained through simulation for an M/M/1 system and the two results are shown to be close for low and moderate loads. The analytic results again show that the distance-based priority scheme is superior to the single priority scheme.

APPENDIX

A Derivation of $\Pr[Y_i \le t \mid C_i = c]$

Y_i is the overall service time of all the customers that arrive during a time period c. The interarrival times are exponentially distributed and the arrival rate is λ_r. The probability that n customers arrive during the time c is given by

$$\Pr\left[n \text{ arrivals} \mid C_i = c\right] = \frac{(\lambda_r c)^n}{n!} e^{-\lambda_r c} \tag{47}$$

for n, $c \ge 0$. The service times of these customers are i.i.d. and are exponentially distributed with rate μ. The probability distribution function of the overall service time given that $n > 0$ customers arrived is therefore (n-Erlang)

$$\Pr\left[Y_i \le t \mid n \text{ arrivals}\right] = 1 - \sum_{k=0}^{n-1} \frac{(\mu t)^k}{k!} e^{-\mu t} \tag{48}$$

for $t \ge 0$. If no customers arrive (i.e., $n = 0$), then $Y_i = 0$. Combining the above two equations, we have

$$\begin{aligned}
&\Pr\left[Y_i \le t \mid C_i = c\right] = \\
&\sum_{n=0}^{\infty} \Pr\left[Y_i \le t \mid n \text{ arrivals}\right] \cdot \Pr\left[n \text{ arrivals} \mid C_i = c\right] = \\
&e^{-\lambda_r c} + \sum_{n=1}^{\infty} \left[\frac{(\lambda_r c)^n}{n!} e^{-\lambda_r c} \left(1 - \sum_{k=0}^{n-1} \frac{(\mu t)^k}{k!} e^{-\mu t} \right) \right] = \\
&e^{-\lambda_r c} + e^{-\lambda_r c} \sum_{n=1}^{\infty} \frac{(\lambda_r c)^n}{n!} - e^{-\lambda_r c} e^{-\mu t} \sum_{n=1}^{\infty} \left[\frac{(\lambda_r c)^n}{n!} \sum_{k=0}^{n-1} \frac{(\mu t)^k}{k!} \right] = \\
&e^{-\lambda_r c} \sum_{n=0}^{\infty} \frac{(\lambda_r c)^n}{n!} - e^{-\lambda_r c} e^{-\mu t} \sum_{n=1}^{\infty} \left[\frac{(\lambda_r c)^n}{n!} \sum_{k=0}^{n-1} \frac{(\mu t)^k}{k!} \right] = \\
&1 - e^{-\lambda_r c} e^{-\mu t} \sum_{n=1}^{\infty} \left[\frac{(\lambda_r c)^n}{n!} \sum_{k=0}^{n-1} \frac{(\mu t)^k}{k!} \right]
\end{aligned} \tag{49}$$

for t, $c \ge 0$, and $\Pr[Y_i \le t \mid C_i = c] = 0$ for $t \le 0$. For convenience, we denote the probability $\Pr[Y_i \le t \mid C_i = c]$ by $F_{Y_i|c}(t)$.

B Evaluation of $\Pr[X_i \le t_i, X_{i+1} \le t_{i+1}]$ for an M/M/1 System

We have $f_{S_{i+1}}(t) = \mu e^{-\mu t}$ and $F_{S_{i+1}}(t) = \Pr[S_{i+1} \le t] = 1 - e^{-\mu t}$. Also, recall that, for $t \ge 0$, $f_{C_i}(t) = \lambda_1 e^{-\lambda_1 t}$ and $f_{X_i}(t) = \mu(1-\rho)e^{-\mu(1-\rho)t}$. From (12), we have

$$\begin{aligned}
&\Pr\left[X_i \le t_i, S_{i+1} \le t_{i+1}, X_i + Y_i \le C_i\right] = \\
&\Pr\left[S_{i+1} \le t_{i+1}\right] \cdot \int_{x=0}^{t_1} \int_{c=x}^{\infty} \Pr\left[Y_i \le c - x\right] f_{C_i}(c) f_{X_i}(x)\, dc\, dx = \\
&\Pr\left[S_{i+1} \le t_{i+1}\right] \cdot \int_{x=0}^{t_1} \int_{c=x}^{\infty} F_{Y_i|c}(c - x) f_{C_i}(c) f_{X_i}(x)\, dc\, dx.
\end{aligned} \tag{50}$$

Since $c \ge x$, the term $c - x$ in the integrand is positive and (49) can be substituted for $\Pr[Y_i \le c - x]$.

From (13), we have

$$\begin{aligned}
&\Pr\left[X_i \le t_i, X_i + Y_i - C_i + S_{i+1} \le t_{i+1}, C_i < X_i + Y_i\right] = \\
&\int_{c=0}^{\infty} \int_{x=0}^{t_i} \int_{s=0}^{t_{i+1}} \Pr\left[c - x < Y_i \le t_{i+1} - s + c - x\right] \cdot \\
&\qquad f_{S_{i+1}}(s) f_{X_i}(x) f_{C_i}(c)\, ds\, dx\, dc,
\end{aligned} \tag{51}$$

where the upper limit in the third integral is set to t_{i+1}, since, when $s > t_{i+1}$, we have $c - x > t_{i+1} - s + c - x$ and, therefore, $\Pr[c - x < Y_i \le t_{i+1} - s + c - x] = 0$. The above integral can be rewritten as:

$$\begin{aligned}
&\Pr\left[X_i \le t_i, X_i + Y_i - C_i + S_{i+1} \le t_{i+1}, C_i < X_i + Y_i\right] = \\
&\int_{c=0}^{t_i} \int_{x=0}^{c} \int_{s=0}^{t_{i+1}} \Pr\left[c - x < Y_i \le t_{i+1} - s + c - x\right] \cdot \\
&\qquad f_{S_{i+1}}(s) f_{X_i}(x) f_{C_i}(c)\, ds\, dx\, dc + \\
&\int_{c=0}^{t_i} \int_{x=c}^{t_i} \int_{s=0}^{t_{i+1}} \Pr\left[c - x < Y_i \le t_{i+1} - s + c - x\right] \cdot \\
&\qquad f_{S_{i+1}}(s) f_{X_i}(x) f_{C_i}(c)\, ds\, dx\, dc + \\
&\int_{c=t_i}^{\infty} \int_{x=0}^{t_i} \int_{s=0}^{t_{i+1}} \Pr\left[c - x < Y_i \le t_{i+1} - s + c - x\right] \cdot \\
&\qquad f_{S_{i+1}}(s) f_{X_i}(x) f_{C_i}(c)\, ds\, dx\, dc = \\
&\int_{c=0}^{t_i} \int_{x=0}^{c} \int_{s=0}^{t_{i+1}} \left(F_{Y_i|c}\left(t_{i+1} - s + c - x\right) - F_{Y_i|c}(c - x) \right) \cdot \\
&\qquad f_{S_{i+1}}(s) f_{X_i}(x) f_{C_i}(c)\, ds\, dx\, dc + \\
&\int_{c=0}^{t_i} \int_{x=0}^{t_i} \int_{s=0}^{t_{i+1}} \left(F_{Y_i|c}\left(t_{i+1} - s + c - x\right) - F_{Y_i|c}(c - x) \right) \cdot \\
&\qquad f_{S_{i+1}}(s) f_{X_i}(x) f_{C_i}(c)\, ds\, dx\, dc + \\
&\int_{c=t_i}^{\infty} \int_{x=0}^{t_i} \int_{s=0}^{t_{i+1}} \left(F_{Y_i|c}\left(t_{i+1} - s + c - x\right) - F_{Y_i|c}(c - x) \right) \cdot \\
&\qquad f_{S_{i+1}}(s) f_{X_i}(x) f_{C_i}(c)\, ds\, dx\, dc.
\end{aligned} \tag{52}$$

In the second term, $c - x \le 0$ and, therefore, $F_{Y_i|c}(c - x) = 0$. Furthermore, $F_{Y_i|c}(t_{i+1} - s + c - x)$ is nonzero only when $s \le t_{i+1} + c - x$. Therefore,

$$\begin{aligned}
&\Pr\left[X_i \le t_i, X_i + Y_i - C_i + S_{i+1} \le t_{i+1}, C_i < X_i + Y_i\right] = \\
&\int_{c=0}^{t_i} \int_{x=0}^{c} \int_{s=0}^{t_{i+1}} \left(F_{Y_i|c}\left(t_{i+1} - s + c - x\right) - F_{Y_i|c}(c - x) \right) \cdot \\
&\qquad f_{S_{i+1}}(s) f_{X_i}(x) f_{C_i}(c)\, ds\, dx\, dc + \\
&\int_{c=0}^{t_i} \int_{x=c}^{t_i} \int_{s=0}^{\max\{0, t_{i+1} + c - x\}} F_{Y_i|c}\left(t_{i+1} - s + c - x\right) \cdot \\
&\qquad f_{S_{i+1}}(s) f_{X_i}(x) f_{C_i}(c)\, ds\, dx\, dc + \\
&\int_{c=t_i}^{\infty} \int_{x=0}^{t_i} \int_{s=0}^{t_{i+1}} \left(F_{Y_i|c}\left(t_{i+1} - s + c - x\right) - F_{Y_i|c}(c - x) \right) \cdot \\
&\qquad f_{S_{i+1}}(s) f_{X_i}(x) f_{C_i}(c)\, ds\, dx\, dc.
\end{aligned} \tag{53}$$

Note that, if $t_i \le t_{i+1}$, then, in the second term, $t_{i+1} + c - x \ge 0$ and the upper limit becomes $t_{i+1} + c - x$. Both $c - x$ and $t_{i+1} - s + c - x$ are greater than 0 in the above expression and, so, (49) can be substituted for $F_{Y_i|c}$ and the above integrals can be evaluated numerically. Adding (50) and (53) gives the joint probability distribution $\Pr[X_{i+1} \le t_{i+1}, X_i \le t_i]$ needed to compute the conditional probabilities p_m and p_M.

C Evaluation of the Steady State Distribution

For a stream with (1, 3)-firm deadlines, the state space is

$$S = \{\mathrm{mmm, mmM, mMm, mMM, Mmm, MmM, MMm, MMM}\}. \tag{54}$$

The transition matrix of the Markov chain of Fig. 1 is

$$P = \begin{bmatrix} p_m & 1-p_m & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & p_M & 1-p_M & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & p_m & 1-p_m & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & p_M & 1-p_M \\ p_m & 1-p_m & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & p_M & 1-p_M & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & p_m & 1-p_m & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & p_M & 1-p_M \end{bmatrix} \tag{55}$$

and the steady state distribution π must satisfy

$$\begin{cases} \pi = \pi P \\ \sum_{s \in S} \pi(s) = 1. \end{cases} \tag{56}$$

Letting $\pi' = [\pi\, 0]$ and

$$P' = \begin{bmatrix} & & 1 \\ & P & \vdots \\ 1 & \cdots & 1 \end{bmatrix}, \tag{57}$$

the above conditions ((56)) can be written as

$$\pi' = \pi' \, \mathrm{P}' - [0 \cdots 0\ 1]. \tag{58}$$

Solving for π', we get

$$\pi' = [0 \cdots 0\ 1] \cdot (\mathrm{P}' - \mathrm{I})^{-1}, \tag{59}$$

where I is the 9×9 identity matrix.

ACKNOWLEDGMENTS

The work reported here is supported in part by the U.S. National Science Foundation under Grant MIP-9213716.

REFERENCES

[1] C.M. Aras, J.F. Kurose, D.S. Reeves, and H. Schulzrinne, “Real-Time Communication in Packet-Switched Networks,” *IEEE Proc.*, vol. 82, pp. 122-139, Jan. 1994.

[2] J.J. Bae and T. Suda, “Survey of Traffic Control Schemes and Protocols in ATM Networks,” *IEEE Proc.*, vol. 79, pp. 170-189, Feb. 1991.

[3] R. Chipalkatti, J.F. Kurose, and D. Towsley, “Scheduling Policies for Real-Time and Non-Real-Time Traffic in a Statistical Multiplexer,” *Proc. IEEE INFOCOM*, pp. 774-783, Apr. 1989.

[4] J.Y. Chung, J.W.S. Liu, and K.-J. Lin, “Scheduling Periodic Jobs That Allow Imprecise Results,” *IEEE Trans. Computers*, vol. 39, no. 9, pp. 1,156-1,174, Sept. 1990.

[5] M. Hamdaoui and P. Ramanathan, “A Dynamic Priority Assignment Technique for Streams with (*m*, *k*)-Firm Deadlines,” *IEEE Trans. Computers*, vol. 44, no. 12, pp. 1,443-1,451, Dec. 1995.

[6] J. Hong, X. Tan, and D. Towsley, “A Performance Analysis of Minimum Laxity and Earliest Deadline Scheduling in a Real-Time System,” *IEEE Trans. Computers*, vol. 38, no. 12, pp. 1,736-1,744, Dec. 1989.

[7] L. Kant and W.H. Sanders, “Loss Process Analysis of the Knockout Switch Using Stochastic Activity Networks,” *Proc. Int’l Conf. Computer Comm. and Network*, pp. 344-349, Sept. 1995.

[8] L. Kleinrock, *Queueing Systems*, vols. I and II. John Wiley and Sons, 1976.

[9] K. Kubota, M. Murata, H. Miyahara, and Y. Oie, “Congestion Control for Bursty Video Traffic in ATM Networks,” *Electronics and Comm. in Japan*, vol. 175, pp. 13-19, Apr. 1992.

[10] J.W.S. Liu, K.-J. Lin, C.L. Liu, W.K. Shih, and J.Y. Chung, “Imprecise Computations: A Means to Provide Scheduling Flexibility and Enhance Dependability,” *Readings in Real-Time Systems*, Y.-H. Lee and C.M. Krishna, eds., chapter 2, pp. 81-97. IEEE CS Press, 1993.

[11] J.W.S. Liu, K.-J. Lin, W.-K. Shih, A.C. Yu, J.-Y. Chung, and W. Zhao, “Algorithms for Scheduling Imprecise Computations,” *Computer*, vol. 24, no. 5, pp. 58-68, May 1991.

[12] R. Nagarajan, “Quality-of-Service Issues in High-Speed Networks,” PhD thesis, Univ. of Massachusetts, Amherst, Sept.1993.

[13] H. Schulzrinne, J.F. Kurose, and D. Towsley, “Congestion Control for Real-Time Traffic in High-Speed Networks,” *Proc. IEEE INFOCOM*, pp. 543-550, June 1990.

Moncef Hamdaoui receved the BS (with highest distinction), MS, and PhD degrees in electrical and computer engineering from the University of Wisconsin, Madison, in 1988, 1990, and 1994, respectively. He is currently working for Northern Telecom in Tunis, Tunisia.

Dr. Hamdaoui’s research interests include real-time systems, communication networks, and fault-tolerant systems. He is a member of the IEEE and the IEEE Computer Society.

Parameswaran Ramanathan received the BTech degree from the Indian Institute of Technology, Bombay, India, in 1984, and the MSE and PhD degrees from the University of Michigan, Ann Arbor, in 1986 and 1989, respectively.

From 1984 to 1989, he was a research assistant in the Department of Electrical Engineering and Computer Science at the University of Michigan, Ann Arbor. He is currently an associate professor in the Department of Computer and Electrical Engineering at the University of Wisconsin, Madison. His research interests include the areas of real-time systems, fault tolerant computing, distributed systems, and parallel algorithms. He is a member of the IEEE.