Journal of Industrial and Systems Engineering
Vol. 1, No. 1, pp 37-55
Spring 2007

A Multiprocessor System with Non-Preemptive Earliest-Deadline-First Scheduling Policy: A Performability Study

Mehdi Kargahi[1,*,], Ali Movaghar[2]**
[1]Department of Electrical and Computer Engineering, University of Tehran,Tehran, Iran
(*kargahi@ipm.ir*)
[2]Department of Computer Engineering, Sharif University of Technology, Tehran,Iran
(movaghar@sharif.edu)

ABSTRACT

This paper introduces an analytical method for approximating the performability of a firm real-time system modeled by a multi-server queue. The service discipline in the queue is earliest-deadline-first (EDF), which is an optimal scheduling algorithm. Real-time jobs with exponentially distributed relative deadlines arrive according to a Poisson process. All jobs have deadlines until the end of service and are served non-preemptively. An important performance measure to calculate is the loss probability. The performance of the system is approximated by a Markovian model in the long run. A key parameter, namely, the loss rate when there are n jobs in the system is used in the model, which is estimated by partitioning the system into two subsystems. The resulting model can then be solved analytically using standard Markovian solution techniques. The number of servers in the system may change due to failure or repair. The performability of the system is evaluated in the presence of such structural changes. The latter measure is approximated by a Markov reward model, considering the loss probability as the reward rate. Comparing numerical and simulation results, we find that the existing errors are relatively small.

Keywords: Analytical methods, Earliest-deadline-first (EDF), Firm real-time systems, Multiprocessor systems, Non-preemptive scheduling, Performability modeling

1. INTRODUCTION

Assessing the impact of scheduling algorithms as well as the failures and repairs of processors in multi-server real-time systems are very important issues in the real-time research community. Furthermore, the concept of scheduling a job that is released and must be completed within some timing constraints, namely *deadline,* is central to both the design and analysis of real-time systems. According to some recent classification of real-time systems as in AlEnavy and Aydin (2005) and Bernat et al.(2001), such systems are categorized as *hard real-time* (HRT), *soft real-time* (SRT), and *firm real-time* (FRT). Contrary to HRT systems, it is not required for SRT and FRT systems, such as mobile devices, to meet all their deadlines. Deadlines in such systems can be met statistically with an upper bound on the fraction of permitted deadline misses, where the ability to respect this bound is quite affected by the scheduling algorithm being used. Some examples with such properties are multimedia-related applications in mobile devices as described in Qiu et al. (2001) or target tracking applications in sensor networks as in Raghunathan et al. (2002). However,

* Corresponding author. Tel.: +98-21-66164628; Fax: +98-21-66019246.
** This work was done in part in the period that the first author was visiting University of Victoria.

systems wherein jobs missing their deadlines can continue their execution with a degraded value are categorized as SRT, while systems wherein such jobs are of no value and are usually thrown away are called FRT.

In general, scheduling policies can be classified into two broad categories: *preemptive* and *non-preemptive*. In a preemptive scheduling policy, processing of the currently running job can be interrupted by a higher priority job, whereas in a non-preemptive scheduling policy, an arriving higher priority job is scheduled only after the completion of the current job execution. Though preemptive scheduling policies can guarantee higher system utilization, there are scenarios where the properties of some hardware or software devices make preemption either impossible or prohibitively expensive. For example, in high speed packet switching networks, preemption requires the retransmission of the preempted packet. Scheduling over a shared media such as LAN, WLAN and field buses described in EN 50170 (1996) such as CAN bus which is discussed in CAN-CIA (1992) and Livani and Kaiser (1998) is inherently non-preemptive, because each node in the network has to ensure that the shared channel is free before it can begin transmission. Besides its extensive use in communication systems, non-preemptive processor scheduling is also used in light weight multi-tasking kernels and is beneficial in multimedia applications as depicted in Dolev and Keizelman (1999). Non-preemptive scheduling policies for real-time embedded systems have also the benefits of more accurate response time analysis, ease of implementation, reduced run-time overhead, and guaranteeing exclusive access to shared resources and data which eliminate both the need for synchronization and its associated overheads.

During the past decades, there has been a growing interest in developing models for evaluating the performance of real-time scheduling algorithms as described in the following paragraph. In such models, an accurate estimation of performance measures such as the fraction of real-time jobs violating their timing constraints, namely, *loss probability* is quite important. We consider FRT systems and the model of deadlines until the end of service, wherein a job is thrown away and considered *lost* if it does not complete execution before its deadline. As many FRT systems such as sensor networks may work in harsh environments with the potential of fault occurrences, their performance can somehow be affected by permanent faults (failures) that may occur in the system processors. Such failures and the possible respective repairs can have quite a profound effect on the dependability of an FRT system. Since the performance level of such a system is changed with failures and/or repairs of system processors, the behavior of the system can be described by a random variable called *performance variable* in Meyer (1982). However, performability evaluation entails a complete probabilistic description of the performance variable (its probability distribution function), and the evaluation of combined dependability and performance measures is referred to as performability evaluation as indicated in Meyer (1982) and Meyer (1992). Moreover, one common method for such an evaluation is using a Markov reward model (MRM) as described in Smith et al. (1988) and Trivedi et al. (1992), and also considered in this paper.

The analysis of queueing systems handling jobs with deadlines has been addressed in numerous papers such as Baccelli et al. (1984), Barrer (1957), Brandt and Brandt (1999), Brandt and Brandt (2002), Cohen (1968), Daley (1965), Doytchinov et al. (2001), Hong et al. (1989), Kargahi and Movaghar (2005), Kargahi and Movaghar (2006), Kruk et al. (2004), Lehoczky (1996), Lehoczky (1997), Movaghar (1998), Movaghar (2006), Palm (1953), Takacs (1974), and Zhao and Stankovic (1989), most of which are focused on FCFS scheduling algorithm. The analysis of scheduling policies that use information on timing constraints is increasingly being of interest in the literature as e.g., in Doytchinov et al. (2001), Hong et al. (1989), Kargahi and Movaghar (2005), Kargahi and Movaghar (2006), Kruk et al. (2004), Lehoczky (1996), Lehoczky (1997), and Zhao and Stankovic (1989). Among such policies, it has been shown that preemptive and non-preemptive EDF are optimal policies within the classes of non-idling service time independent preemptive (see George et al. (1996) and Liu and Layland (1973)) and non-preemptive (see George et al. (1995)) scheduling policies, respectively. Moreover, it has also been shown that EDF stochastically minimizes the fraction of lost jobs in both preemptive and non-preemptive models in FRT systems as indicated in Panwar et al. (1988), Towsley and Panwar (1990), and Towsley and Panwar (1992). Based on the EDF scheduling algorithm (and as its name indicates), the job with the earliest deadline is the one with the highest priority. Because of the optimality of EDF, its analysis is particularly valuable. Despite its importance, the evaluation of the EDF scheduling algorithm has not been able to keep pace with many other recent successful attempts to analyze real-time queueing systems. This may be mainly due to the complexity of such analysis. Hong et al. (1989) first introduced upper and lower bounds on the performance of an *M/M/m/EDF+M* queue in FRT systems (The last *M* determines that the *relative deadlines*, which are the interval of time between the arrival of jobs and their deadlines, follow an exponential distribution). Their results were later improved by Kargahi and Movaghar (2006) and extended to general distribution of relative deadlines. Some approximation methods to analyze the performance of EDF in *heavy traffic* conditions (in which the system has high average-case utilization) in SRT systems have also been proposed in Doytchinov et al. (2001), Lehoczky (1996), and Lehoczky (1997). All of the above studies are only for the performance analysis of preemptive EDF.

This paper generalizes a preliminary analytical method proposed by the same authors in Kargahi and Movaghar (2005) for *M/M/*1/EDF+*M* queues with the model of deadlines until the end of service and the *non-preemptive* EDF scheduling algorithm in two manners: one is that it evaluates the performance of multi-server systems, namely *M/M/m/*EDF+*M* models, and the other is that the performability of the system in the presence of processor failure/repair is also studied. The method to be introduced in this paper is based on Markov chain models and covers a wide range of input rates while it is simple to use. It gives a relatively good approximation for an important measure of performance in firm real-time systems, namely, the loss probability. Assuming the loss probability as the reward rate, the accuracy of the approximation method has also been evaluated for some performance variables and performability measures, which show the ability of an FRT system to complete its mission in some specified time durations. A key parameter used in this method is γ_n, which is the rate of missing deadline when there are n jobs in the system. To the best of our knowledge, no other analytical or approximation method exists for a similar problem, i.e., a multi-server system with the model of deadlines until the end of service. Comparison of the analytical and simulation results will show that the proposed method is relatively accurate.

The paper is organized as follows. Section 2 introduces the system model and solution including the performance and performability models, and the proposed analytical method. This is followed in Section 3 by explaining our method of estimating the loss rate for the multi-server system. Section 4 provides the numerical results and the comparison of the analytical and simulation results for both performance and performability measures. Concluding remarks and future works are finally presented in Section 5.

2. SYSTEM MODEL AND SOLUTION

This section initially uses a continuous-time Markov chain (CTMC) to describe the performance model of a multi-server system and solves it with respect to an important performance measure, namely loss probability. Afterwards, the performability model of the system with the possibility of variations in the number of servers (due to failure and/or repair) will be presented using a Markov reward model (MRM). In the latter model, the loss probability is assumed as the reward rates, and some performability measures will be studied.

2.1. Performance Model

We consider a multiprocessor system with m servers which serve the jobs waiting in a single infinite-capacity queue. The jobs arrive according to a Poisson process with rate λ, where an exponentially distributed service time with rate μ is assigned to each arriving job. A relative deadline, which is the interval of time between the arrival of a job and its deadline, is also associated with each job. We assume that the relative deadlines are random variables of an exponential distribution with mean value θ. If the deadline of a job is missed, it is thrown away and considered lost. The scheduling algorithm in the system is earliest-deadline-first (EDF). As specified in the definition of the EDF scheduling algorithm, the job closest to its deadline is to be served. Since the serving mechanism in the system is non-preemptive, no other job can preempt the serving job(s). Two models of deadlines are used together in our formulation: deadlines until the beginning of service (DBS) and deadlines until the end of service (DES). In the former model, a job is thrown away if it cannot begin execution before its deadline; otherwise, it is executed. In the latter model, a job is thrown away and considered lost if it does not complete its execution before its deadline. However, all deadlines follow one of the above models. Whereas if the model of deadlines is DBS, the serving jobs will eventually complete their services, if it is DES, they will either complete their services or miss their deadlines. It is proved in Panwar et al. (1988) and Towsley and Panwar (1990) that the EDF scheduling algorithm stochastically maximizes the fraction of jobs meeting their deadlines for both DBS and DES models within the class of non-idling service time independent non-preemptive scheduling policies.

The approach presented in this paper is based on finding a state-dependent loss rate function γ_n to be defined below. Let N be the set of natural numbers and R^+ the set of positive real numbers. For $t,\varepsilon \in R^+$ and $n \in N$, let

$\Psi_n(t,\varepsilon) \equiv$ the probability that a job misses its deadline during $[t,t+\varepsilon)$, given there are n jobs in the system at time t.

Define

$$\gamma_n(t) = \lim_{\varepsilon \to 0} \frac{\Psi_n(t,\varepsilon)}{\varepsilon} \tag{1}$$

Assuming statistical equilibrium, let

$$\gamma_n = \lim_{t \to \infty} \gamma_n(t) \tag{2}$$

γ_n is the (steady-state) rate of missing deadlines when there are n jobs in the system (including the one(s) being served). Consequently, the resulting Markov chain model of the system, $\mathbf{X}_m$, may be shown as in Figure 1. When the population of the system is n, it can be decreased to n-1 because of either completing the service requirements of a job (with rate $\mathrm{Min}(n,m)\mu$) or missing a job's deadline (with rate γ_n).

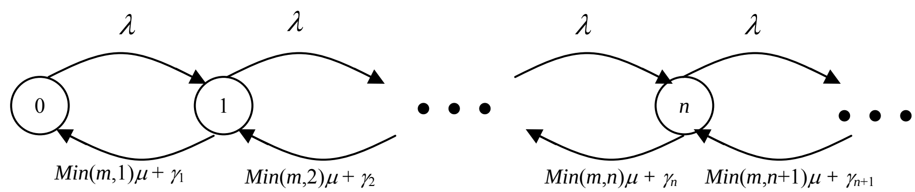


Figure 1. State-transition-rate diagram for Markov chain $\mathbf{X}_m$.

Barrer (1957) was the first to introduce the idea of γ_n for deterministic relative deadlines of real-time jobs in a single-server queue. The idea was extended in Brandt and Brandt (2002), Movaghar (1998), and Movaghar (2006) to a larger class of models when relative deadlines have a general distribution and jobs arrive according to a *state-dependent* Poisson process. These latter results assume the FCFS scheduling algorithm, and show that γ_n is independent of the input rate. In Brandt and Brandt (2002) and Movaghar (1998), the description of calculating γ_n for DBS in a multi-server system is presented. The calculation of γ_n for the case of DES in a single-server system is presented in Movaghar (2006). Moreover, a method for estimating γ_n for an $M/M/m/$EDF+G system with preemptive services for DES (with m=1) and non-preemptive services for DBS is presented in Kargahi and Movaghar (2006). Furthermore, a method for estimating this parameter for an $M/M/1/$EDF+M queue with non-preemptive services for DES is proposed in Kargahi and Movaghar (2005).

In the following, the required equations for solving the system model $\mathbf{X}_m$ are presented and the equilibrium state probabilities will be obtained. Using such information, the target performance measure, namely, the loss probability of the system will be calculated. Let

$p_n \equiv$ the (steady-state) probability that there are n jobs in $\mathbf{X}_m$. (3)

The balance equations for the system, in equilibrium, can be written as follows:

$$0 = -\lambda p_0 + (\mu + \gamma_1) p_1, \qquad \text{if } n = 0$$
$$0 = \lambda p_{n-1} - (\lambda + Min(m,n)\mu + \gamma_n) p_n + (Min(m,n+1)\mu + \gamma_{n+1}) p_{n+1}. \quad \text{if } n > 0 \tag{4}$$

Solving the equilibrium, we obtain

$$p_n = \frac{\lambda^n}{\prod_{i=1}^{n} (\gamma_i + Min(m,i)\mu)} p_0 \tag{5}$$

The normalizing condition is

$$\sum_{n=0}^{\infty} p_n = 1. \tag{6}$$

From (5) and (6), we find

$$p_0 = \left(1 + \sum_{n=1}^{\infty} \frac{\lambda^n}{\prod_{i=1}^{n} (\gamma_i + Min(m,i)\mu)} \right)^{-1} \tag{7}$$

The probability of missing deadlines in the system may then be obtained as

$$\alpha_{d,m} = \frac{\sum_{n=1}^{\infty} p_n \gamma_n}{\sum_{n=0}^{\infty} p_n \lambda} = \frac{\sum_{n=1}^{\infty} p_n \gamma_n}{\lambda}, \tag{8}$$

which is the average rate of missing deadlines divided by the average rate of job arrivals. To analyze the system with the EDF scheduling algorithm, we need to have a formulation of γ_n (for EDF) as defined in (2). In Section 3, we describe a method for estimating γ_n of non-preemptive EDF for the case of DBS in an infinite-capacity multiprocessor system. Afterwards, such estimation will be used in some formulations to find a generalization to a similar system with non-preemptive EDF for the case of DES.

2.2. Performability Model

In this section, we assume a multiprocessor system with an infinite-capacity queue and a maximum of M servers with the possibility of variations in the number of servers due to failure and repair. Such variation can affect the performance level of the system. This system is modeled by a structure state process **X**, as shown in Figure 2. Each structure state m of **X**, $0\leq m\leq$M, corresponds to $\mathbf{X}_m$, showing that m fault-free servers are available in the system. Similarly, $\mathbf{X}_0$ indicates that the system is completely failed. We assume that one failure can occur in the servers at each instant of time. The rate of such events is assumed to be λ_F. When a server fails, a recovery action must be taken (e.g., shutting down the failed server to prevent of damaging the operation of the other system components), or the whole system will fail and enter the structure state 0. The probability that the recovery action is successfully completed is known as the coverage, denoted by c. On the other hand, we consider two kinds of repair actions, global repair with rate μ_G which restores the system from structure state 0 to structure state M, and local repair with rate μ_F which can be thought of as a repairman beginning to fix a server as soon as it fails, assuming that one repair person is available for the system.

The performability evaluation of a system may characterize its behavior in steady-state or in some periods of time such as $[0,t]$. Because the time-scale of the performance-related events (e.g., inter-arrival times, service times, and relative deadlines) is at least two orders of magnitude less than the time-scale of the dependability-related events and the utilization time under study (i.e., t, $1/\lambda_F, 1/\mu_F, 1/\mu_G >> 1/\lambda, 1/\mu, \theta$), the steady-state values of performance measures can be used to specify the performance levels or *reward rates* for each structure state of **X**. We consider that the reward rate associated with structure state m of **X**, denoted by r_m, is the loss probability of $\mathbf{X}_m$, namely $r_m=\alpha_{d,m}$. Accordingly, the structure state process **X** and the reward structure ($\mathbf{X}_m$, m=0,…,M) determine a MRM.

According to the above discussion, the evolution of **X** in time can be represented by the finite-state stochastic process $\{S(t),\ t\geq 0\}$, which characterizes the dynamics of the system structure, where $S(t)\in\{0,1,\ldots,\mathrm{M}\}$ is the structure state of the system at time t. The holding times in the structure states are exponentially distributed, and hence $S(t)$ is a homogeneous CTMC. Even in situations where such times are generally distributed, they may often be acceptably approximated using a finite number of exponential phases as indicated in Hsueh et al. (1988). Let Q be the M+1 by M+1 generator matrix of **X** and $\pi_m(t)$ denote Prob $[S(t)=m]$, i.e., the probability that **X** is in structure-state

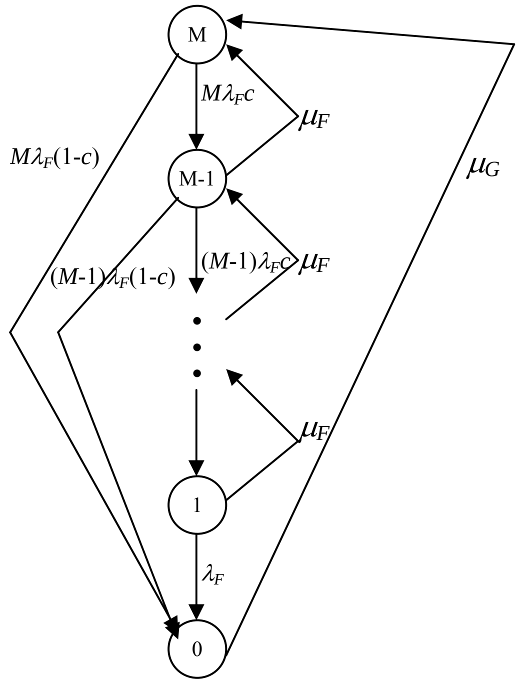


Figure 2. State-transition-rate diagram for the structure state process **X**

m at time *t*. The column vector $\boldsymbol{\pi}(t)$ of the state probabilities may be computed by solving a matrix differential equation as demonstrated in Reibman and Trivedi (1988)

$$\frac{d}{dt}\boldsymbol{\pi}(t) = Q^T\boldsymbol{\pi}(t) \tag{9}$$

The steady state probability vector $\boldsymbol{\pi}$ of the Markov chain is the solution for the linear system

$$Q^T\boldsymbol{\pi}=0, \quad \sum_{i=0}^{\mathrm{M}} \pi_i = 1 \tag{10}$$

Accordingly, we can have the reward rate of the system at time *t*, $R(t)=r_{S(t)}$ as a performance variable. More performance variables can also be defined as follows. We let $A(t)$ to be the accumulated reward until time *t*, namely, the area under the $R(t)$ curve

$$A(t) = \int_0^t R(\tau)d\tau \tag{11}$$

and $B(t)$ to be the corresponding time-averaged accumulated reward

$$B(t) = \frac{A(t)}{t} = \frac{1}{t}\int_0^t R(\tau)d\tau \tag{12}$$

By interpreting rewards as loss probabilities, we see that the distribution of time-averaged loss probability ($B(t)$) is at the heart of characterizing the quality of service (QoS) of firm real-time systems that evolve through states with different reward rates. (More discussion on this matter is presented in Section 4.) We denote the probability distribution function (PDF) of time-averaged loss probability by time *t* evaluated at *x* as

$$\mathfrak{B}(x,t) \equiv \text{Prob}\ [B(t) \le x]. \qquad (13)$$

Some interesting studies can be done on the expected values of the above parameters, e.g. $E[R(t)]$ or $E[B(t)]$. To find the steady-state behavior of the above measures, we should calculate their limit value when $t \to \infty$. (It is interesting to note that $\lim_{t\to\infty} E[B(t)] = \lim_{t\to\infty} E[R(t)]$.) Moreover, we write $\mathfrak{B}(x,\infty) \equiv \lim_{t\to\infty} \mathfrak{B}(x,t)$.

In the following section, we investigate a method to estimate γ_n to be able to calculate the reward rates (loss probabilities) associated to different structure states of **X** with different number of servers. Subsequently, we can use the above discussions to study the performability of **X**.

3. DETERMINATION OF LOSS RATES

In this section, we present methods for estimating γ_n^{EDF} in the cases of DBS and DES. First, we will have an overview on a method to estimate γ_n^{EDF} for the case of DBS, namely, $\gamma_n^{\text{EDF-DBS}}$. Then, we will use some ideas to present a method for estimating the same parameter for DES, namely, $\gamma_n^{\text{EDF-DES}}$. To do so, some bounds for γ_n^{EDF} will be defined. Combining the bounds for the former case will result in an estimation of the required parameter for DBS. Afterwards, the resulting formulation besides a different view to the system will be used to estimate the required parameter for DES.

As indicated in Movaghar (2006), for a specified mean relative deadline (θ) in a FCFS system, deterministic relative deadlines generate the minimum loss probability among all distributions of relative deadlines. Accordingly, we assume that such a property is also valid for the EDF scheduling algorithm. Since for deterministic relative deadlines, EDF is the same as FCFS, we can assume the loss probability of FCFS scheduling algorithm for deterministic relative deadlines as the lower bound of the loss probability of EDF scheduling algorithm for exponentially distributed relative deadlines. On the other hand, since EDF is an optimal scheduling algorithm for both deadline models (see Panwar et al. (1988), Towsley and Panwar (1990), and Towsley and Panwar (1992)), it can minimize the loss probability among all other scheduling algorithms, especially FCFS. Therefore, we will have

$$\alpha_d^{\text{FCFS-det}} \le \alpha_d^{\text{EDF-exp}} \le \alpha_d^{\text{FCFS-exp}}, \qquad (14)$$

where $\alpha_d^{\text{FCFS-det}}$ and $\alpha_d^{\text{FCFS-exp}}$ represent the loss probabilities of the system with deterministic and exponential relative deadlines for the FCFS scheduling algorithm, respectively. We also assume that such ordering is valid for loss rates in the FCFS and EDF scheduling algorithms. Such validity is strongly confirmed by simulation results presented in part in Kargahi and Movaghar (2006). Therefore, we will have

$$\gamma_n^{\text{FCFS-det}} \le \gamma_n^{\text{EDF-exp}} \le \gamma_n^{\text{FCFS-exp}}, \qquad (15)$$

where the functions describing the above two bounds of $\gamma_n^{\text{EDF-exp}}$ are given in Movaghar (1998) for a multi-sever system with DBS and in Movaghar (2006) for a single-server system with DES.

The above two bounds are linearly combined using a multiplier to obtain an appropriate estimation of $\gamma_n^{\text{EDF-DBS}}$. (If the exact values of $\gamma_n^{\text{EDF-DBS}}$ were to be known, then solving $\mathbf{X}_m$ would result in an exact analysis of EDF for that model of deadlines). More explanation of this approach for an infinite-capacity queue with multiple servers and the DBS model is given in the following section. Consequently, such estimation will be used in a different manner to estimate the loss rates of a multi-server system with non-preemptive EDF scheduling algorithm and DES model, namely, $\gamma_n^{\text{EDF-DES}}$.

3.1. Non-preemptive EDF with DBS

In this section, we propose a multiplier to linearly combine the two bounds indicated above in the case of DBS to estimate $\gamma_n^{\text{EDF-DBS}}$.

As defined in Kargahi and Movaghar (2006), contrary to the FCFS scheduling algorithm, the simulation results strongly indicate that the state-dependent loss rates depend on λ for the EDF scheduling algorithm. Accordingly, advantages of some properties of EDF and some simulation results can be used to make a multiplier which linearly combines the bounds defined previously. The multiplier must be adjusted to a function of λ to get a more accurate estimation of $\gamma_n^{\text{EDF-DBS}}$. The multiplier, $\xi_{\text{DBS}}(.)$, combines the bounds as follows:

$$\gamma_n^{\text{EDF-DBS}} = \frac{\left(\xi_{\text{DBS}}(.)\gamma_n^{\text{FCFS-exp-DBS}} + \gamma_n^{\text{FCFS-det-DBS}}\right)}{\xi_{\text{DBS}}(.)+1} \tag{16}$$

where $\xi_{\text{DBS}}(.)$, which defines the effective ratio of each of the bounds on $\gamma_n^{\text{EDF-DBS}}$, is to be specified.

As discussed previously, it has been shown that for the FCFS scheduling algorithm, the loss rate is independent of λ (see Brandt and Brandt (2002), Movaghar (1998), and Movaghar (2006)). Therefore, such parameters can be calculated as

$$\gamma_n^{\text{FCFS}-\text{exp}-\text{DBS}} = \begin{cases} 0, & n \le m \\ \dfrac{n-m}{\theta}, & n > m \end{cases} \tag{17}$$

and

$$\gamma_n^{\text{FCFS}-\text{det}-\text{DBS}} = \begin{cases} 0, & n \le m \\ m\mu\left(\dfrac{F_{E_{n-m-1}}(\theta)}{F_{E_{n-m}}(\theta)} - 1\right), & n > m \end{cases} \tag{18}$$

where

$$F_{E_n}(\theta) = 1 - e^{-m\mu\theta}\sum_{i=0}^{n-1}\frac{(m\mu\theta)^i}{i!} \tag{19}$$

for exponential and deterministic relative deadlines until the beginning of service, respectively, which are obtained from Movaghar (1998). As defined in Kargahi and Movaghar (2006), $\xi_{\text{DBS}}(.)$ is a function of three parameters, namely, *n-m*, $\rho=\lambda/m\mu$, and $m\mu\theta$ for exponential relative deadlines, where *n-m* is the number of waiting jobs in the queue, ρ is the normalized arrival rate (normalized λ with respect to $m\mu$), and $m\mu\theta$ is the normalized mean relative deadline with respect to the mean service time $1/m\mu$. The function describing the behavior of $\xi_{\text{DBS}}(.)$ with respect to the above three parameters, i.e., $\xi_{\text{DBS}}(n-m,\rho,m\mu\theta)$, is as follows (obtained from Kargahi and Movaghar (2006)):

$$\xi_{\text{DBS}}(n-m,\rho,m\mu\theta)=\frac{6.7}{(n-m+1)\rho^{1.25}\sqrt{m\mu\theta}}, \tag{20}$$

Substituting $\xi_{\text{DBS}}(.)$ above in (16), we can find $\gamma_n^{\text{EDF-DBS}}$.

The way that $\xi_{\text{DBS}}(.)$ depends on the normalized arrival rate (ρ) can be explained by some properties of EDF. Due to the dynamics of the EDF scheduling algorithm with respect to different values of ρ, for very small values of ρ where $\rho\to 0$, $\gamma_n^{\text{EDF-DBS}}$ converges to $\gamma_n^{\text{FCFS-exp-DBS}}$; therefore, $\xi_{\text{DBS}}(.)$ tends to be very large as $\xi_{\text{DBS}}(.)\to+\infty$. The reason is that for very light traffic intensities (where the average population is very low), EDF behaves very similar to FCFS and the improvements of EDF over FCFS are quite limited. On the other hand, for large values of ρ, the behavior of EDF becomes more similar to that of FCFS with deterministic relative deadlines, where $\gamma_n^{\text{EDF-DBS}}$ converges to the lower bound; therefore, $\xi_{\text{DBS}}(.)$ tends to be very small as $\xi_{\text{DBS}}(.)\to 0$. Next, we use the recent formulations of $\gamma_n^{\text{EDF-DBS}}$, as obtained from (20) and (16), to estimate $\gamma_n^{\text{EDF-DES}}$ for non-preemptive EDF scheduling algorithm.

3.2. Non-preemptive EDF with DES

In spite of the DBS model, the serving jobs may also miss their deadlines in the DES model. However, although the deadlines are until the end of service, due to the non-preemptive nature of the scheduling algorithm, even if the deadline of an arriving job is earlier than that of a job in one of the servers, the serving job will not be preempted and continues to get service. It has been proven in Towsley and Panwar (1990) and Towsley and Panwar (1992) that the non-preemptive EDF scheduling algorithm also stochastically minimizes the fraction of lost jobs in the class of non-idling service time independent non-preemptive scheduling algorithms. In spite of its optimality, to the best of our knowledge, no other analytical or approximation method for the probabilistic analysis of this algorithm exists (other than the one proposed in Kargahi and Movaghar (2005) for a single-server queue by the same authors). In the following, we present a method for estimating γ_n^{EDF} for non-preemptive EDF with DES model, namely $\gamma_n^{\text{EDF-DES}}$, which results in approximating the performance of non-preemptive EDF scheduling algorithm. To do so, we propose another view to the system as in the following paragraphs.

Since we do not have any formulation for calculating the bounds of $\gamma_n^{\text{EDF-DES}}$, we should use a different method for estimating this parameter. The main idea of the proposing estimation method is to break the multi-server system into two subsystems. Afterwards, two loss rates will be calculated for the subsystems, which adding them together will result in an estimation of the desired parameter, namely, $\gamma_n^{\text{EDF-DES}}$.

Due to the fact that the serving jobs are non-preemptive, after starting service, the behavior of the system with respect to these jobs is similar to that of a system with a number of parallel servers (with no waiting rooms) and the FCFS scheduling algorithm. On the other hand, if the number of available jobs in the system (n) is greater than the number of existing servers (m), the remaining n-m job(s) in the system follow the EDF scheduling algorithm. Therefore, the system can be broken into two subsystems (see Figure 3): the first one (Subsystem-1) containing the non-preemptive servers with rate μ, which can be considered as m FCFS queues, each with capacity 1 (no waiting room), and the second one (Subsystem-2) which can virtually be assumed as a multi-server non-preemptive EDF queue with DBS and m servers, each with a virtual service rate μ', to be determined.

First, we study Subsystem-1. Since the available k jobs in Subsystem-1 ($k \leq m$) can be assumed as k parallel FCFS queues (each with capacity 1), the total loss rate of this subsystem will be simply $\gamma'_k = k\gamma_1^{\text{FCFS-exp-DES}}$, where we have $\gamma_1^{\text{FCFS-exp-DES}} = \gamma_2^{\text{FCFS-exp-DBS}}$ as can be found in Movaghar (2006).

Second, we consider Subsystem-2. As defined previously, we can assume of this subsystem as a virtual multi-server queue with the DBS model. According to such view to Subsystem-2 and due to the fact that the loss rates in the servers are taken into account in Subsystem-1, it can virtually be assumed that no jobs of Subsystem-2 will miss their deadlines after starting service. Now, we use the method presented in Section 3.1 to calculate the loss rate of Subsystem-2. First, the lower and upper bounds should be specified (since this subsystem is assumed as a virtual system with DBS, we have the respective bounds for the required calculations). As indicated previously, deterministic relative deadlines construct the lower bound and exponentially distributed relative deadlines construct the upper bound. Since the serving jobs leave the servers due to service completion or deadline miss, the virtual service rate of the servers for the lower bound can be assumed as $\mu_{\text{L}} = \mu + \gamma_1^{\text{FCFS-det-DES}}$, where we have $\gamma_1^{\text{FCFS-det-DES}} = \gamma_2^{\text{FCFS-det-DBS}}$ as can be found in Movaghar (2006). Similarly, the virtual service rate of the servers for the upper bound can be assumed as $\mu_{\text{U}} = \mu + \gamma_1^{\text{FCFS-exp-DES}}$. Due to the fact that $\gamma_n^{\text{FCFS-exp-DBS}}$ (and therefore $\gamma_n^{\text{FCFS-exp-DES}}$) is independent of the service rate, μ_{U} does not affect the upper bound. Substituting μ_{L} for μ in (18) and (19), we obtain the lower bound. Moreover, (17) simply gives the upper bound. On the other hand, since the distribution of relative deadlines is exponential for the EDF scheduling algorithm, the virtual service rate of the servers of Subsystem-2 can also be assumed as $\mu' = \mu + \gamma_1^{\text{FCFS-exp-DES}}$. Accordingly, substituting μ' for μ and $\rho' = \lambda / m\mu'$ for ρ in (20), and then using (16) with the bounds specified above, we obtain the loss rate for Subsystem-2, namely, γ''_n. Consequently, we have

$$\gamma_n^{\text{EDF-DES}} = \begin{cases} \gamma'_n, & \text{if } n \leq m \\ \gamma'_m + \gamma''_n, & \text{if } n > m \end{cases} \tag{21}$$

Substituting $\gamma_n^{\text{EDF-DES}}$ above for γ_n in $\mathbf{X}_m$ and solving the resulting Markov chain using the method presented in Section 2.1, we find $\alpha_{d,m}$ for the multi-server system with the non-preemptive EDF scheduling algorithm.

4. NUMERICAL EXAMPLES

In this section, we study some examples to illustrate the accuracy of our approximation method. The examples are first presented for evaluating the performance of $\mathbf{X}_m$ and then to do so for the performability of $\mathbf{X}$. The results obtained from the analytical method are compared with respect to the simulation results in both performance and performability studies. To model $\mathbf{X}$ and calculate the *expected value of performance variables* and also the *performability measures*, we have used the Mobius tool (discussions on this tool can be found in Deavours et al. (2002) and Sanders (2005)).

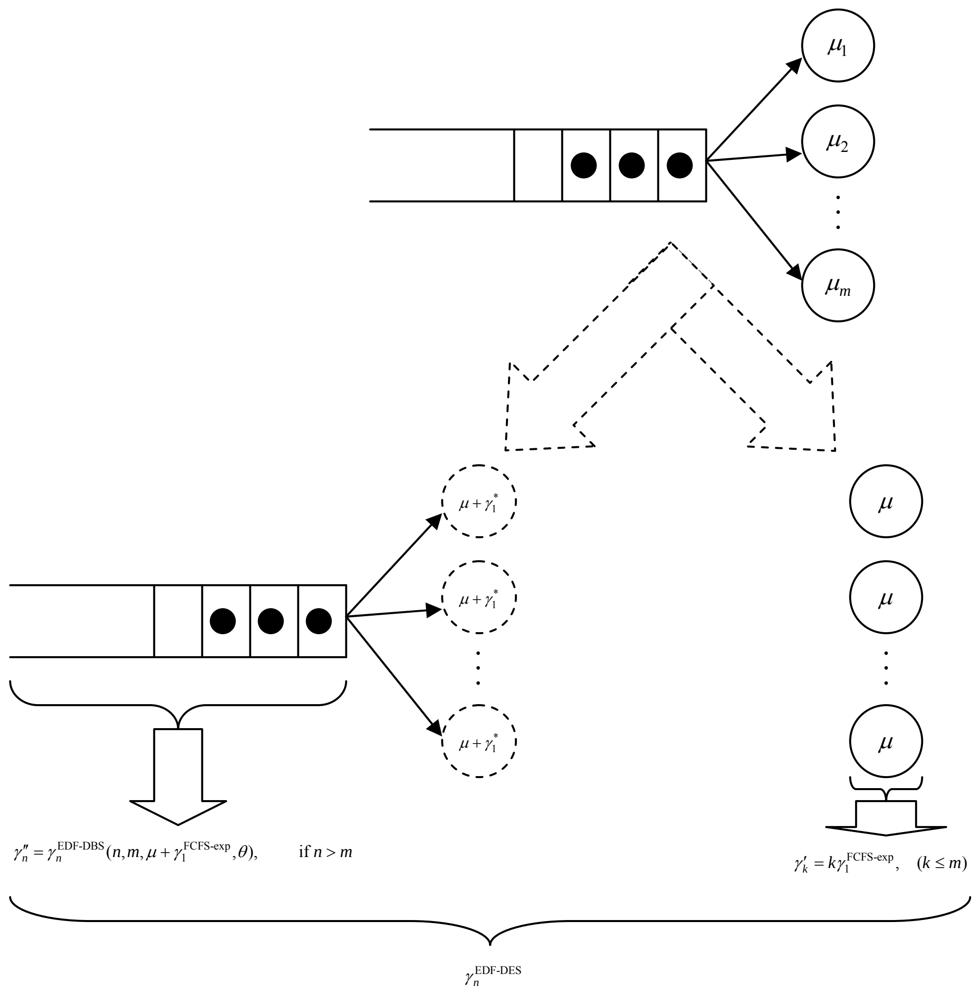
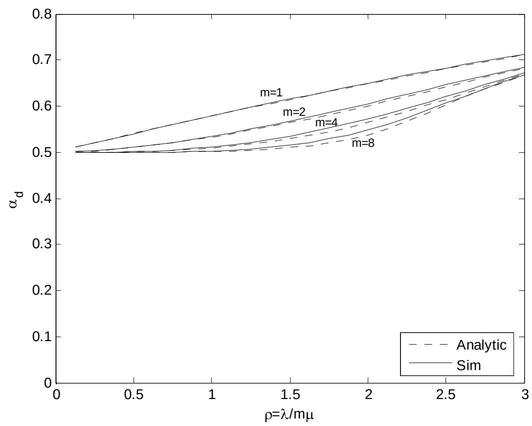
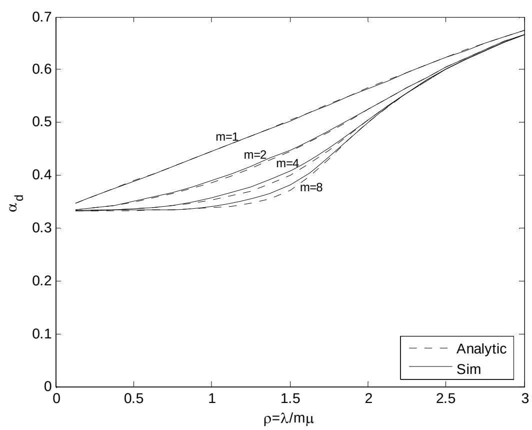


Figure 3. The modified view to the multi-server non-preemptive EDF queue with DES.

First, we study the accuracy of the proposed method for approximating the performance of $\mathbf{X}_m$. The performance measure considered is the loss probability ($\alpha_{d,m}$). The respective results are presented for a broad range of normalized input rates, $\rho \in (0,3]$, and two values of mean relative deadline θ, namely 1 and 2, where θ is normalized with respect to $1/\mu$. In this regards, we investigate the respective probabilities of missing deadlines from the analytical modeling and simulation and a wide range of normalized input rates (ρ) from almost no traffic to very heavy traffic intensity, i.e., for the interval (0,3]. The capacity of the queue in the system is taken to be large enough to be approximated as infinite. The experiments have been done for a multi-server system with m=1, 2, 4, and 8 and non-preemptive EDF scheduling algorithm. Figure 4 illustrates such information graphically showing that the analytical and simulation results are very close together. As can be observed, the less the number of servers, the higher the accuracy of the approximation method. The probabilities of missing deadlines obtained from the analytical modeling as well as simulation and their respective relative errors for both values of θ are presented in Tables 1 and 2. Moreover, their respective maximum relative errors and the *root mean square error* (RMSE[1]) of the approximated results with respect to the real (simulation) results have been shown in Table 3. As can be seen, the worst relative error is about 2.78 %, which happens when m=8, θ=2, and ρ is about 1.375 (not shown in Table 2). This error is in the acceptable range for most FRT applications. The worst error for the single-server case (m=1) is about 0.29 % (occurring for $\theta = 2$ and ρ=2.125) which is almost half of its respective maximum relative error obtained by the method introduced in Kargahi and Movaghar (2005) for a single-server system.



(a)



(b)

Figure 4. Comparison of the analytical and simulation results for a multi-server queue with non-preemptive EDF scheduling algorithm and (a) $\theta = 1$, (b) $\theta = 2$.

Second, we study the accuracy of our method for approximating the performability of **X**. The failure rate has been considered to be λ_F=0.001 and the coverage probability is assumed to be c=0.95. Moreover, we have considered a multiprocessor system with M=4, θ=2, and both non-repairable and repairable models with μ_F=μ_G=0 and μ_F=10μ_G=0.5, respectively. The reward rate assigned to each structure state m of **X,** namely r_m can also be calculated using (8). Such reward rates for some

[1] - To calculate RMSE, the square root of the mean value of the squares of relative errors is calculated.

values of *m* have been shown in Figure 3(b) and partially in Table 2 (the reward rate for structure state 0 is considered as r_0=1).

Table 1. Probability of missing deadlines obtained from the analytical method and simulation and the respective relative errors for a multi-server non-preemptive system with DES and $\theta = 1$.

X_m	α_d											
	m=1			*m*=2			*m*=4			*m*=8		
ρ	Sim.	Anal.	Err.%	Sim.	Anal.	Err.%	Sim.	Anal.	Err.%	Sim.	Anal.	Err.%
0.25	0.5204	0.5206	0.04	0.5029	0.5028	-0.02	0.4999	0.5001	0.04	0.5001	0.5000	-0.02
0.50	0.5403	0.5406	0.06	0.5101	0.5100	-0.02	0.5010	0.5012	0.04	0.5003	0.5000	-0.06
0.75	0.5596	0.5598	0.04	0.5208	0.5205	-0.06	0.5043	0.5042	-0.02	0.5006	0.5004	-0.04
1.00	0.5792	0.5786	-0.10	0.5349	0.5332	-0.32	0.5113	0.5097	-0.31	0.5022	0.5016	-0.12
1.25	0.5974	0.5968	-0.10	0.5509	0.5478	-0.56	0.5211	0.5180	-0.59	0.5065	0.5046	-0.38
1.50	0.6156	0.6146	-0.16	0.5676	0.5641	-0.62	0.5353	0.5293	-1.12	0.5151	0.5100	-0.99
1.75	0.6336	0.6319	-0.27	0.5859	0.5820	-0.67	0.5521	0.5445	-1.38	0.5288	0.5197	-1.73
2.00	0.6500	0.6488	-0.18	0.6054	0.6011	-0.71	0.5721	0.5641	-1.40	0.5490	0.5376	-2.08
2.25	0.6668	0.6653	-0.22	0.6255	0.6212	-0.69	0.5955	0.5882	-1.23	0.5747	0.5668	-1.37
2.50	0.6822	0.6812	-0.15	0.6461	0.6419	-0.65	0.6211	0.6151	-0.97	0.6061	0.6025	-0.59
2.75	0.6977	0.6966	-0.16	0.6656	0.6626	-0.45	0.6460	0.6429	-0.48	0.6383	0.6368	-0.24
3.00	0.7129	0.7114	-0.21	0.6853	0.6828	-0.36	0.6714	0.6693	-0.31	0.6671	0.6667	-0.06

Table 2. Probability of missing deadlines obtained from the analytical method and simulation and the respective relative errors for a multi-server non-preemptive system with DES and $\theta = 2$.

X_m	α_d											
	m=1			*m*=2			*m*=4			*m*=8		
ρ	Sim.	Anal.	Err.%	Sim.	Anal.	Err.%	Sim.	Anal.	Err.%	Sim.	Anal.	Err.%
0.25	0.3610	0.3612	0.06	0.3378	0.3380	0.06	0.3337	0.3336	-0.03	0.3336	0.3333	-0.09
0.50	0.3884	0.3890	0.15	0.3503	0.3495	-0.23	0.3361	0.3360	-0.03	0.3339	0.3335	-0.12
0.75	0.4165	0.4168	0.07	0.3678	0.3659	-0.52	0.3432	0.3421	-0.32	0.3356	0.3347	-0.27
1.00	0.4448	0.4451	0.07	0.3902	0.3867	-0.90	0.3570	0.3527	-1.20	0.3414	0.3385	-0.85
1.25	0.4738	0.4742	0.08	0.4165	0.4126	-0.94	0.3775	0.3701	-1.96	0.3542	0.3465	-2.17
1.50	0.5032	0.5041	0.18	0.4482	0.4447	-0.78	0.4088	0.4004	-2.05	0.3809	0.3706	-2.70
1.75	0.5338	0.5347	0.17	0.4851	0.4828	-0.47	0.4523	0.4481	-0.93	0.4350	0.4319	-0.71
2.00	0.5642	0.5653	0.19	0.5251	0.5243	-0.15	0.5052	0.5042	-0.20	0.5007	0.5001	-0.12
2.25	0.5944	0.5952	0.13	0.5663	0.5656	-0.12	0.5567	0.5563	-0.07	0.5554	0.5556	0.04
2.50	0.6226	0.6237	0.18	0.6045	0.6039	-0.10	0.6003	0.6001	-0.03	0.5999	0.6000	0.02
2.75	0.6498	0.6504	0.09	0.6379	0.6378	-0.02	0.6364	0.6363	-0.02	0.6362	0.6363	0.02
3.00	0.6741	0.6749	0.12	0.6672	0.6671	-0.02	0.6670	0.6667	-0.04	0.6666	0.6663	-0.05

Table 3. Maximum relative error and root mean square error (RMSE) of loss probabilities for a multi-server non-preemptive EDF queue.

Non-Preemptive EDF	α_d			
	$\theta = 1$		$\theta = 2$	
Number of servers (*m*)	**Max. Rel. Err. (%)**	**RMSE (%)**	**Max. Rel. Err. (%)**	**RMSE (%)**
1	0.27	0.16	0.29	0.13
2	0.76	0.49	0.95	0.49
4	1.39	0.82	2.15	0.94
8	2.08	0.94	2.78	1.04

Since many FRT systems should provide a predetermined QoS in a bounded time (their mission time), the accuracy of the approximations of some performance variables such as the time-averaged loss probability ($B(t)$) can be very important to design such systems. Accordingly, the expected values of $B(t)$ for the time duration of [0, 300] and different arrival rates are shown in Figure 5, showing the accuracy of the approximation method. The results have been presented for both analytically calculated rewards (obtained using (8), as described above) and rewards obtained from simulation. Comparing these results, we find that the existing errors are relatively small (the relative errors are less than 2% in the worst case, which occurs for λ=5.5). Figure 6 also shows similar information for λ=5.5 and different time durations, namely [0, t]. As can be observed, the performance variable $B(t)$ (showing the QoS) of the non-repairable system degrades as the time progresses, while that of the repairable system becomes stable after a short period of time.

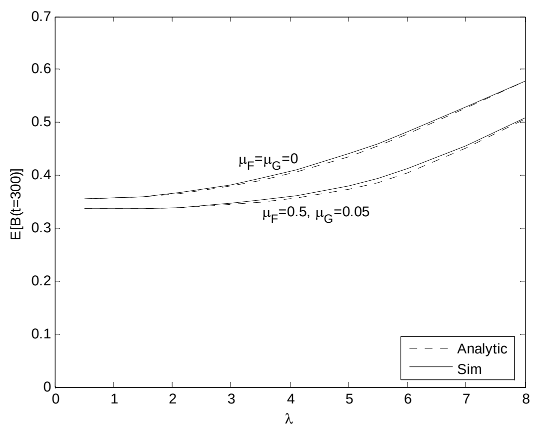


Figure 5. Comparison of the expected values of time-averaged loss probability for analytically calculated rewards and rewards obtained from simulation for $\theta = 2$ in the time interval [0, 300], and both repairable and non-repairable models.

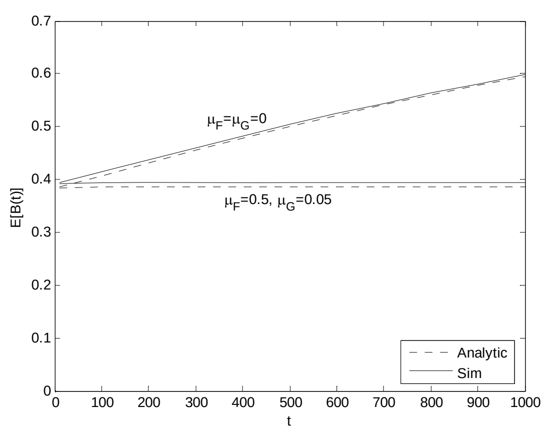


Figure 6. Comparison of the expected values of time-averaged loss probability for analytically calculated rewards and rewards obtained from simulation for $\theta = 2$ in the time interval [0, t], and both repairable and non-repairable models.

An important problem in FRT systems is to find the *degree of certainty* that the system, with some failure and repair rates, remains functional and delivers an acceptable performance level (QoS) during a well-defined *operation/mission time*. As discussed in Section 2.2, the PDF of the performance variables can be used to derive this information. The PDF of $B(t)$, namely $\mathfrak{B}(x,t)$ as indicated in (13), obtained for both analytically calculated rewards and rewards obtained from simulation of a non-repairable system have been shown in Figure 7 (Similar results can be obtained for the repairable system, which due to the overlapping of the curves have not been shown in here). As can be observed, the analytical and simulation results almost overlap in all cases. As an example of the information which can be derived from this figure, we can note that the probability of having the time-averaged loss probability of the system in a duration of 300 time units to be less than 50% is about 80%, i.e., $\mathfrak{B}(50\%,300)=\text{Prob}(B(300)\leq 50\%)=80\%$. According to such information, we can realize if the system satisfies the desired QoS in a predetermined mission time or not.

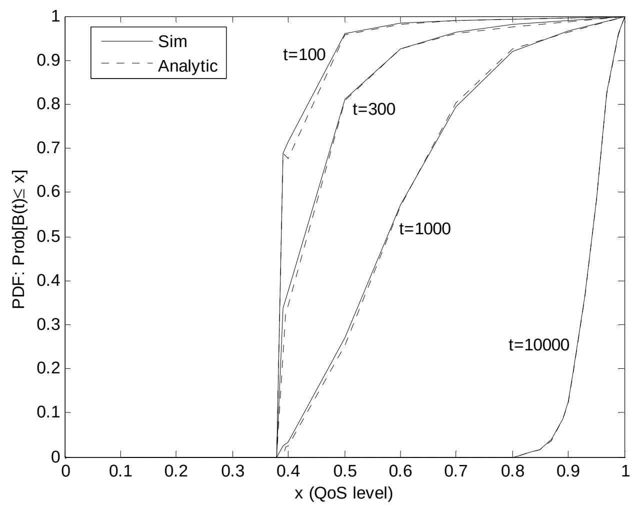


Figure 7. Comparison of the PDF of time-averaged loss probability by time t evaluated at x, namely $\mathfrak{B}(x,t)$, for analytically calculated rewards and rewards obtained from simulation for $\theta = 2$ in the time intervals [0, 100], [0, 300], [0, 1000], and [0, 10000] for the non-repairable model.

6. CONCLUDING REMARKS AND FUTURE WORK

In this paper, we have considered the problem of performance and performability analysis of a multi-server firm real-time system with the non-preemptive EDF scheduling algorithm. All jobs in the system are considered to have deadlines until the end of service. The performance model of the system is presented using a CTMC and its performability model is presented using a MRM. The main performance measure to calculate is the loss probability. Assuming this measure as the reward rate in the respective MRM, an important performance variable, namely the time-averaged loss probability and its PDF can also be approximated with good accuracies. Such a PDF indicates the probability of presenting a predetermined QoS level during the mission-time of a firm real-time system. To the best of our knowledge, there has been no other analytical or approximation method for the performance/performability evaluation of a similar system. We have presented an approximation method for this problem, which is relatively accurate. The main idea for such approximations is the estimation of an important parameter, γ_n^{EDF}, which is the loss rate when there are n jobs in the system.

Our future work will include extending the current approach to parallel queues as well as using such analysis in the design of more realistic QoS-based firm real-time systems.

REFERENCES

[1] AlEnavy T.A., Aydin H. (2005), Energy-constrained scheduling for weakly-hard real-time systems; *Proceedings of the 26th IEEE Real-Time Systems Symposium (RTSS'05)*; 376-385.

[2] Baccelli F., Boyer P., Hebuterne G. (1984),Single-server queues with impatient customers; *Advanced Applied Probability* 16; 887-905.

[3] Barrer D.Y. (1957), Queueing with impatient customers and ordered service; *Operational Research* 5; 650- 656.

[4] Bernat G., Burns A., Llamosi A. (2001), Weakly hard real-time systems; *IEEE Transactions on Computers* 50 (4); 308-321.

[5] Brandt A., Brandt M. (1999), On the M(n)/M(n)/s queue with impatient calls; *Performance Evaluation* 35; 1-18.

[6] Brandt A., Brandt M. (2002), Asymptotic results and a Markovian approximation for the M(n)/M(n)/s+GI system; *Queueing Systems* 41; 73-94.

[7] Cohen J.W. (1968), Single server queue with uniformly bounded virtual waiting time; *Journal of Applied Probability* 5; 93-122.

[8] Daley D.J. (1965), General customer impatience in queue GI/G/1; *Journal of Applied Probability* 2; 186-205.

[9] Deavours D.D., Clark G., Courtney T., Daly D., Derisavi S., Doyle J.M., Sanders W.H., Webster P.G. (2002), The Mobius framework and its implementation; *IEEE Transactions on Software Engineering* 28(10); 956-969.

[10] Dolev S., Keizelman A. (1999), Non-preemptive real-time scheduling of multimedia tasks; *Real-Time Systems* 17(1); 23–39.

[11] Doytchinov B., Lehoczky J., Shreve S. (2001), Real-time queues in heavy traffic with earliest-deadline-first queue discipline; *Annals of Applied Probability* 11; 332-379.

[12] EN 50170 (1996), General purpose field communication system; CENELEC, In European Standard.

[13] George L., Muhlethaler P., Rivierre N. (1995), Optimality and non-preemptive real-time scheduling revisited; Rapport de Recherche RR-2516, INRIA, Le Chesnay Cedex, France.

[14] George L., Rivierre N., Spuri M. (1996), Preemptive and non-preemptive real-time uni-processor scheduling; Rapport de Recherche RR-2966, INRIA, Le Chesnay Cedex, France.

[15] Hong J., Tan X., Towsley D. (1989), A performance analysis of minimum laxity and earliest deadline scheduling in a real-time system; *IEEE Transactions on Computers* 38(12); 1736-1744.

[16] Hsueh M.C., Iyer R. K., Trivedi K.S. (1988), Performability modeling based on real data: A case study; *IEEE Transactions on Computers* 37(4); 478-484.

[17] Kargahi M., Movaghar A. (2005), Non-preemptive earliest-deadline-first scheduling policy: A performance study; *Proceedings of IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS'05)*; 201-210.

[18] Kargahi M., Movaghar A. (2006), A method for performance analysis of earliest-deadline-first scheduling policy; *Journal of Supercomputing* 37(2); 197-222.

[19] Kruk L., Lehoczky J., Shreve S., Yeung S.N. (2004), Earliest-deadline-first service in heavy-traffic acyclic networks; *Annals of Applied Probability* 14(3); 1306-1352.

[20] Lehoczky J.P. (1996), Real-time queueing theory; *Proceedings of IEEE Real-Time Systems Symposium*; 186-195.

[21] Lehoczky J.P. (1997), Using real-time queueing theory to control lateness in real-time systems; *Performance Evaluation Review* 25(1); 158-168.

[22] Liu C.L., Layland J.W. (1973). Scheduling algorithms for multiprogramming in a hard real-time environment; *Journal of Associative Computational Machinery* 20; 46-61.

[23] Livani M.A., Kaiser J. (1998), EDF consensus on CAN bus access for dynamic real-time applications; *Proceedings of IEEE Workshop on Parallel and Distributed Computing Systems in conjunction with 12th International Parallel Processing Symposium / 9th Symposium on Parallel and Distributed Processing (IPPS/SPDP'98)*; 1088-1097.

[24] Meyer J.F. (1982), Closed-form solutions of performability; *IEEE Transactions on Computers* C-31(7); 648-657.

[25] Meyer J.F. (1992), Performability: A retrospective and some pointers to the future; *Performance Evaluation* 14(3-4); 139-156.

[26] Movaghar A. (1998), On queueing with customer impatience until the beginning of service; *Queueing Systems* 29; 337-350.

[27] Movaghar A. (2006), On queueing with customer impatience until the end of service; *Stochastic Models* 22; 149-173.

[28] Palm C. (1953), Methods for judging the annoyance caused by congestion; *Tele* 2; 1-20.

[29] Panwar S.S., Towsley D., Wolf J.K. (1988), Optimal scheduling policies for a class of queues with customer deadlines to the beginning of service; *Journal of Associative Computational Machinery* 35(4); 832-844.

[30] Qiu Q., Wu Q., Pedram M. (2001), Dynamic power management in a mobile multimedia system with guaranteed quality-of-service; *Proceedings of the 38th Conference on Design Automation* (*DAC'01*); 834-839.

[31] Raghunathan V., Schurgers C., Park S., Srivastava M. B. (2002), Energy aware wireless microsensor networks; *IEEE Signal Processing Magazine* 19 (2); 40-50.

[32] Reibman A.R., Trivedi K.S. (1988), Numerical transient analysis of Markov dependability models; *Computers and Operations Research* 15; 19-36.

[33] Smith R.M., Trivedi K.S., Ramesh A.V. (1988), Performability analysis: Measures, an algorithm, and a case study; *IEEE Transactions on Computers* 37(4); 406-417.

[34] Takacs L. (1974), A single-server queue with limited virtual waiting time; *Journal of Applied Probability* 11; 612-617.

[35] Towsley D., Panwar S.S. (1990), On the optimality of minimum laxity and earliest deadline scheduling for real-time multiprocessors; *Proceedings of IEEE EUROMICRO-90 Workshop on Real-Time*; 17-24.

[36] Towsley D., Panwar S.S. (1992), Optimality of the stochastic earliest deadline policy for the G/M/c queue serving customers with deadlines; *Second ORSA Telecommunications Conference*.

[37] Trivedi K.S., Muppala J.K., Woolet S.P., Haverkort B.R. (1992), Composite performance and dependability modeling; *Performance Evaluation* 14; 197-215.

[38] Zhao W., Stankovic J.A. (1989), Performance analysis of FCFS and improved FCFS scheduling algorithms for dynamic real-time computer systems; *Proceedings of IEEE Real-Time Systems Symposium;* 156-165.

[39] CAN-CIA, ″CAN specification 2.0 Part B″, http://www.cancia.org/downloads/ciaspecificatios, 1992.

[40] Sanders W.H., ″Mobius user manual″, http://www.mobius.uiuc.edu, 2005.