Journal of Computer Science 3 (9): 673-684, 2007
ISSN 1549-3636
© 2007 Science Publications

Analysis of Hierarchical Diff-EDF Schedulability over Heterogeneous Real-Time Packet Networks

Moutaz Saleh and Zulaiha Ali Othman
Faculty of Information Science and Technology, University Kebangsaan Malaysia

Abstract: Packet networks are currently enabling the integration of traffic with a wide range of characteristics that extend from video traffic with stringent QoS requirements to the best-effort traffic requiring no guarantees. QoS guarantees can be provided in conventional packet networks by the use of proper packet scheduling algorithms. As a computer revolution, many scheduling algorithms have been proposed to provide different schemes of QoS guarantees with Earliest Deadline First (EDF) as the most popular one. With EDF scheduling, all flows receive the same miss rate regardless of their traffic characteristics and deadlines. This makes the standard EDF algorithm unsuitable for situations in which the different flows have different miss rate requirements since in order to meet all miss rate requirements it is necessary to limit admissions so as to satisfy the flow with the most stringent miss rate requirements. In this paper, we propose a new priority assignment scheduling algorithm, Hierarchal Diff-EDF (Differentiate Earliest Deadline First), which can meet the real-time needs of these applications while continuing to provide best effort service to non-real time traffic. The Hierarchal Diff-EDF features a feedback control mechanism that detects overload conditions and modifies packet priority assignments accordingly. To examine our proposed scheduler model, we introduced our attempt to provide an exact analytical solution. The attempt showed that the solution was apparently very complicated due to the high interdependence between the system queues' service. Hence, the use of simulation techniques seems inevitable. The simulation results showed that the Hierarchical Diff-EDF achieved the minimum packet average delay when compared with both EDF and Diff-EDF schedulers.

Key words: Performance Analysis, Hierarchical Scheduling, EDF, Real-Time, Packet Networks

INTRODUCTION

Recently, many applications of computer networks rely on the ability of the network to provide Quality of Service (QoS) guarantees. These guarantees are usually bounded in the form of delay, bandwidth, packet loss rate and buffer utilization or a combination of these parameters. Furthermore, packet networks are currently enabling the integration of traffic with a wide range of characteristics that extend from video traffic with stringent QoS requirements to ``best-effort'' traffic requiring no guarantees. QoS guarantees can be provided in conventional packet networks by the use of proper packet scheduling algorithms. The function of a scheduling algorithm is to select the packet to be transmitted in the next cycle from the available arrived packets.

Network traffic can be categorized into two types: real-time traffic, such as video and audio and non-real-time traffic such as http data. Recently, there has been a significant increase in the amount of multimedia services transmitted over networks. These multimedia applications, due to the stringent delay constraints, have to meet certain QoS guarantees. Since scheduling has a direct impact on the system capacity and delay as well as throughput, it is therefore necessary to investigate the suitable scheduling algorithms for multimedia traffic.

The distinguishing characteristic of real-time traffic is that it requires bounded delay while it can tolerates some packet losses. The delay can be bounded by associating a deadline for each packet. Once a packet misses its deadline, it will be dropped as it is no longer useful. Therefore the main goal for any scheduling scheme for real-time traffic is to deliver packets in a timely manner.

As a computer revolution, many scheduling algorithms have been proposed to meet this goal. The First-Come-First-Serve (FCFS) scheduling algorithm, which is mostly used in conventional networks, is

Corresponding Author: Moutaz Saleh, Faculty of Information Science and Technology, University Kebangsaan Malaysia, 43600 UKM Bangi, Selangor

widely adopted for best-effort traffic. On the other hand, many scheduling algorithms have been proposed to provide different schemes of QoS guarantees, with Earliest Deadline First (EDF) as the most popular one.

Real-time systems: A real-time system has two notions of correctness: logical and temporal[1]. In particular, in addition to producing correct outputs (logical correctness), such a system needs to ensure that these outputs are produced at the correct time (temporal correctness). However, selecting appropriate methods for scheduling activities is one of the important considerations in the design of a real-time system[2]; such methods are essential to ensure that all activities are able to meet their timing constraints. These timing constraints are usually specified using a deadline, which corresponds to the time by which a specific operation must complete.

Real-time systems can be broadly classified as hard or soft depending on the criticality of deadlines[3]. In hard real-time systems, all deadlines must be met; equivalently, a deadline miss results in an incorrect system. On the other hand, in a soft real-time system, timing constraints are less stringent; occasional deadline misses do not affect the correctness of the system.

A real-time system is typically composed of several or sequential tasks with timing constraints. In most real time systems, tasks are invocated repeatedly: each invocation of a task is referred as a job; and the corresponding time of invocation is referred as the job's release time or job's deadline[1]. Thus, the relative deadline parameter is used to specify the timing constraints of the jobs.

Related work: Many real-time systems rely on the earliest deadline first (EDF) scheduling algorithm. This algorithm has been shown to be optimal under many different conditions. For example, for independent, preemptable tasks, on uni-processor EDF is optimal in the sense that if any algorithm can find a schedule where all tasks meet their deadline then EDF can meet the deadline[4]. Also, Jackson's rule[5] says that ordering a set of tasks by deadline will minimize the maximum lateness. Further, it also has been shown that EDF is optimal under certain stochastic conditions[6].

In spite of these advantageous properties, EDF has one major negative aspect. That is, when using EDF in a dynamic system, if overload occurs, tasks may miss deadlines in an unpredictable manner and in the worst case, the performance of the system can approach zero effective throughput. This is due to the fact that EDF gives highest priority to those processes that are close to missing their deadlines. In such situations, EDF does not provide any type of guarantee on which tasks will meet their timing constrains. This is a very undesirable behavior in practical systems, since in real-world applications intermittent overloads may occur due to exceptional situations, such as modifications in the environment, arrival of a burst of tasks, or cascades of system failures. In that case, matters may be improved by introducing some congestion control mechanism.

A robust earliest deadline scheduling algorithm for dealing with sporadic tasks under overloads in hard real-time environment was proposed by[7]. The algorithm synergistically combines many features including a very minimum level of guarantee, dynamic guarantees, graceful degradation in overloads, deadline tolerance and resource reclaiming. Also, in 1995[8] presented a comparative study among scheduling algorithms, which use different priority, assignments and different guarantee mechanisms to improve the performance of a real-time system during overload conditions. Their results showed that EDF scheduling performs best if admission control is used along with a reclaiming mechanism that's takes advantage of early completions. In 1997[9] introduced algorithms for flow admission control at an EDF link scheduler. Their results showed that these algorithms have very low computational complexity and are easily applicable in practice.

While real-time system designers try to design the system with sufficient resources, because of cost and highly unpredictable environments, it is sometimes impossible to guarantee that the system resources are sufficient; in this case EDF's performance degrades rapidly in overload situations. However, it is worthy to say that in the year of 1998, EDF was a major paradigm for real-time scheduling[10].

EDF is a widely used algorithm for online deadline scheduling. It has been known for long that EDF is optimal for scheduling an underloaded, single processor system; recent results on the extra-resource analysis of EDF further revealed that EDF when using moderately faster processors can achieve optimal performance in the under loaded, multi-processor setting[11], initiated the extra resource analysis of EDF for overloaded systems, showing that EDF supplemented with simple form of admission control can provide a similar performance guarantee in both single and multi-processor settings.

Also, EDF is widely used in scheduling real-time database transactions[12]. By using EDF, database transactions are classified into two categories, those that have missed their deadlines and those that have not. The latter category can be scheduled using the EDF algorithm, while the former can be kept in background and executed whenever there are no transactions that have not missed their deadlines awaiting services.

A major problem with EDF, when scheduling network traffic, is that all flows receive the same miss rate regardless of their traffic characteristics and deadlines[13]. This makes the standard EDF algorithm unsuitable for situations in which the different flows have different miss rate requirements since in order to meet all miss rate requirements it is necessary to limit admissions so as to satisfy the flow with the most stringent miss rate requirements.

MATERIALS AND METHODS

The research methodology is based on breaking up the simulation into three stages:

Decomposition: Breaks up the real problem into a set of autonomous elements that interact between them and whose interactions reproduce the real problem.

Modeling: That models each element by defining its functional capabilities, its behaviors and its interaction modes with other elements. This is done through building different java classes for the different elements.

Description: Description of possible actions between the elements by defining the laws which control them.

The performance evaluation technique used for this research is the simulation technique. Simulation is one of the most widely used operation-research and management science techniques. It is defined as the imitation of the operation of a real-world process or system over time. Thus, simulation modelling can be used both as an analysis tool for predicting the effect of changes to existing systems and as a design tool to predict the performance of new systems under varying sets of circumstances.

The specific nature of the developed simulator is considered as a Dynamic Stochastic Discrete event simulation. A dynamic simulation model represent a system as it evolves over time, stochastic is defined as the simulation model contains some probabilistic such as random input components. Finally, discrete event system is one for which the state variables change instantaneously at separated points of time.

System structure of the hierarchical Diff-EDF: The goal of the Hierarchical Diff-EDF scheduling algorithm is to guarantee that a flow's deadline miss rates meet its pre-specified QoS requirements and achieve the high utilization. In EDF scheduler, low priority flows, such as Non-Real-Time traffic, can starve as it is characterized with long lead-times. Despite EDF provides stable QoS guarantees to high priority flows, such as Real-Time traffic, the deadline miss rates of the low priority flows can be unacceptably high. Figure 1 models the EDF scheduler.

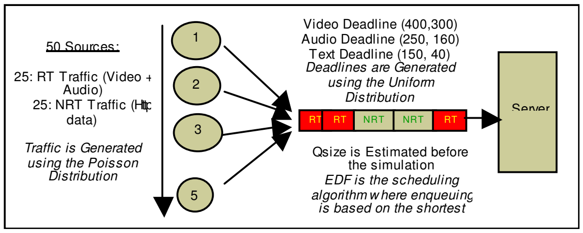
50 Sources:
25: RT Traffic (Video + Audio)
25: NRT Traffic (Http data)
Traffic is Generated using the Poisson Distribution
1
2
3
5
Video Deadline (400,300)
Audio Deadline (250, 160)
Text Deadline (150, 40)
Deadlines are Generated using the Uniform Distribution
RT
RT
NRT
NRT
RT
Server
Qsize is Estimated before the simulation
EDF is the scheduling algorithm where enqueuing is based on the shortest

Fig. 1: EDF scheduler

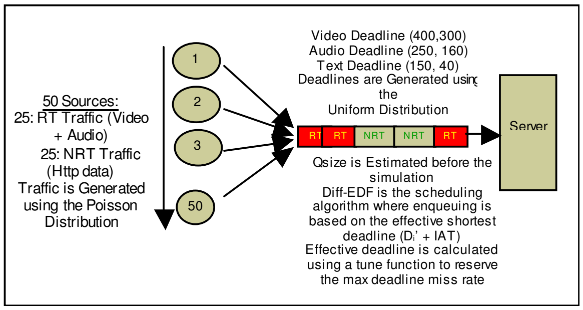
50 Sources:
25: RT Traffic (Video + Audio)
25: NRT Traffic (Http data)
Traffic is Generated using the Poisson Distribution
1
2
3
50
Video Deadline (400,300)
Audio Deadline (250, 160)
Text Deadline (150, 40)
Deadlines are Generated using the Uniform Distribution
RT
RT
NRT
NRT
RT
Server
Qsize is Estimated before the simulation
Diff-EDF is the scheduling algorithm where enqueuing is based on the effective shortest deadline (Di' + IAT)
Effective deadline is calculated using a tune function to reserve the max deadline miss rate

Fig. 2: Diff-EDF scheduler

By analyzing the previous figure, three main drawbacks were discovered in using the EDF to schedule real-time packet network traffic:

* All flows receive the same miss rate regardless of their traffic characteristics and deadlines. This makes the standard EDF algorithm unsuitable for situations in which the different flows have different miss rate requirements since in order to meet all miss rate requirements it is necessary to limit admissions so as to satisfy the flow with the most stringent miss rate requirements.
* Packet Starving for the Non-Real-Time traffic. Since Real-Time Traffic is characterized with short lead-times (time until their deadline expires), then it receives high priority comparing to the Non-Real-Time Traffic, which leads to packet starving.
* A random Assignment of Network traffic (Real-Time and Non-Real-Time). As mentioned before, the FCFS scheduling algorithm is widely adopted for best-effort traffic. Having only one service discipline forces all traffic, regardless of their characteristics, to follow the same scheduling algorithm, in our case the EDF.

To overcome the first drawback of the EDF, a new Diff-EDF priority assignment algorithm is proposed[13]. The Diff-EDF scheduling algorithm considers each flow as having stochastic traffic characteristic, a stochastic deadline and a maximum allowable miss rate. Figure 2 shows a representative model for the Diff-EDF scheduler.

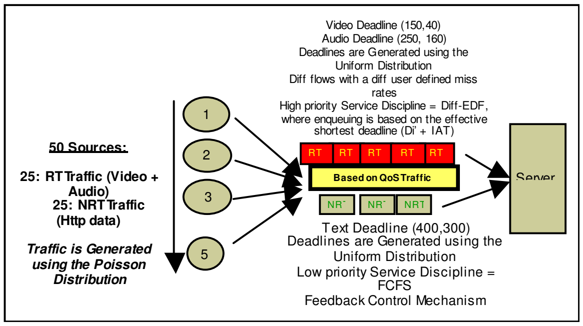
Video Deadline (150,40)
Audio Deadline (250, 160)
Deadlines are Generated using the Uniform Distribution
Diff flows with a diff user defined miss rates
High priority Service Discipline = Diff-EDF, where enqueuing is based on the effective shortest deadline (Di' + IAT)
50 Sources:
25: RT Traffic (Video + Audio)
25: NRT Traffic (Http data)
Traffic is Generated using the Poisson Distribution
1
2
3
5
RT
RT
RT
RT
RT
Based on QoS Traffic
NRT
NRT
NRT
Server
Text Deadline (400,300)
Deadlines are Generated using the Uniform Distribution
Low priority Service Discipline = FCFS
Feedback Control Mechanism

Fig. 3: Hierarchical Diff-EDF scheduler

Again by analyzing the Fig. 2, it is obvious that the last two drawbacks are still discovered when Diff-EDF is used to schedule non-real-time network traffic. As a result, it would be desirable to have a scheme which allows the individual deadline miss rates of different flows to be distinct and controllable. Our proposed Hierarchical Diff-EDF scheme satisfies this objective. It can meet the real-time needs of such applications, by using the Diff-EDF scheduler, while continuing to provide best effort service to non-real time traffic through depending on the strength of the FCFS scheduler. The Hierarchical Diff-EDF features a feedback control mechanism that detects overload conditions and modifies packet priority assignments accordingly. Figure 3 shows a representative model for the Hierarchical Diff-EDF scheduler.

The Hierarchical Diff-EDF scheduler uses a tuning method, or marker, that adjusts the deadlines of the incoming packets by adding a constant to the relative deadlines before the packets are placed into different queues based on the traffic type. Different constants are added to different flows and the modified deadlines are known as "Effective Deadlines".

Model Formal Description: Let $P_{n_1,n_2,n_3}(t)$ be the probability that there are n_1, n_2 and n_3 customers of types Text (T), Audio (A) and Video (V) respectively in the system at time t. Let δt be some small number, and then the probability $P_{n_1,n_2,n_3}(t+\delta t)$ can be expressed as:

$P_{n_1,n_2,n_3}(t+\delta t)$ = P(There are (n_1-1) customers of type T, n_2 of type A, n_3 of type V in the system at time t and a T arrival in $(t,t+\delta t)$ and the service is for the (T) queue) + P(There are (n_1+1) customers of type T, n_2 of type A, n_3 of type V in the system at time t and a T customer is served in $(t,t+\delta t)$ and the service is for the (T) queue) + P(There are (n_1) customers of type T, (n_2-1) of type A, n_3 of type V in the system at time t and an A arrival in $(t,t+\delta t)$ and the service is for the (A,V) queue) + P(There are (n_1) customers of type T, (n_2) of type A, n_3-1 of type V in the system at time t and a V arrival in $(t,t+\delta t)$ and the service is for the (A,V) queue) + P(There are (n_1) customers of type T, (n_2) of type A, n_3 of type V in the system at time t and no arrivals or departures in $(t,t+\delta t)$ and the service is for the (A,V) queue) + P(There are (n_1-1) customers of type T, (n_2) of type A, n_3 of type V in the system at time t and an A arrival in $(t,t+\delta t)$ and the service is for the (A,V) queue) + P(There are (n_1) customers of type T, (n_2-1) of type A, n_3 of type V in the system at time t and an A arrival in $(t,t+\delta t)$ and the service is for the (T) queue) + P(There are (n_1) customers of type T, (n_2+1) of type A, n_3 of type V in the system at time t and the priority is for A customers and an A customer is served in $(t,t+\delta t)$ and the service is for the (A,V) queue) + P(There are (n_1) customers of type T, (n_2) of type A, (n_3-1) of type V in the system at time t and a V arrival in $(t,t+\delta t)$ and the service is for the (T) queue) + P(There are (n_1) customers of type T, (n_2) of type A, (n_3+1) of type V in the system at time t and the priority Is for V customers and a V customer is served in $(t,t+\delta t)$ and the service is for the (A,V) queue) + P(There are (n_1) customers of type T, (n_2) of type A, n_3 of type V in the system at time t and no arrivals or departures in $(t,t+\delta t)$ and the service is for the (T) queue).

Now, knowing that the DiffEDF Queue will serve the Video and Audio traffic while the FCFS Queue will serve the Text traffic, the following conditions should be satisfied:

```
1.  If ( !EmptyDiffEdfQueue && !thresholdOn )
        Serve the Diff-EDF Queue

2.  Else If ( !EmptyDiffEdfQueue && thresholdOn )
        If ( FCFSQueue > (0.7*pgr/3) )
                Serve the FCFS Queue
        Else    Serve the Diff-EDF Queue
```

3. Else Serve the FCFS Queue

Rewriting the above event probabilities symbolically we obtain:

$$
\begin{aligned}
P_{n_1,n_2,n_3}(t+\delta t) = {} & \\
& \theta \Big[P_{n_1-1,n_2,n_3}(t)\ \lambda\delta t + P_{n_1+1,n_2,n_3}(t)\ \mu\delta t \\
& + P_{n_1,n_2-1,n_3}(t)\ \lambda\delta t + P_{n_1,n_2,n_3-1}(t)\ \lambda\delta t + \\
& + P_{n_1,n_2,n_3}(t)(1-3\lambda\delta t-\mu\delta t)+o(\delta t) \Big] \\
+(1-\theta) & \Big[P_{n_1-1,n_2,n_3}(t)\ \lambda\delta t \\
& + P_{n_1,n_2-1,n_3}(t)\ \lambda\delta t + P_{n_1,n_2+1,n_3}(t)\ \mu\delta t\ \beta \\
+ P_{n_1,n_2,n_3-1}(t)\ \lambda\delta t & + P_{n_1,n_2,n_3+1}(t)\ \mu\delta t\ (1-\beta) \\
& + P_{n_1,n_2,n_3}(t)(1-3\lambda\delta t-\mu\delta t)+o(\delta t) \Big] \qquad (1)
\end{aligned}
$$

Where θ is the probability that the Text queue has reached the threshold *k*. This probability is for the text queue size equals or exceeds *k*, and where β is the probability that priority condition was satisfied by a member of the n_2 queue (Audio). Accordingly, we will show our attempt to solve the equation (1) as an analytical solution for our mathematical model.

Attempting Analytical Solution:

Let $Y_i = IAT_i + QoS_i, \qquad i = 1,\ldots,n_2$

Where IAT_i is $Exp(\lambda)$, and QoS_i is $U(a_A, b_A)$

Also, let $Z_j = IAT_j + QoS_j, \qquad j = 1,\ldots,n_3$

Where IAT_j is $Exp(\lambda)$, and QoS_j is $U(a_V, b_V)$

Let $R = \begin{cases} 1, \text{ if } \min\limits_{\text{i,j}}(Y_i, Z_j) \text{is a Y} \\ 0, \text{ Otherwise} \end{cases}$

Hence, $\beta = P(R=1)$

Where $\beta = P(R=1) = P\left(\min\limits_i Y_i \le \min\limits_j Z_j\right)$

Now we want to find the PDF of $MY = \min\limits_i Y_i$ and $MZ = \min\limits_j Z_j$

$P(MY \le u) = 1 - P(MY > u) = 1 - P(Y > u)^{n_1}$

Now let $Y = IAT + QoS$

Where IAT is $Exp(\lambda)$, QoS is $U(a,b)$

The Probability Density Function (PDF) of Y can be found using the distribution function technique in [14] as follows:

Let $G(y) = P(Y \le y)$

Substituting $Y = IAT + QoS$

$G(y) = P(IAT + QoS \le y)$

The sample space here is a strip in the Cartesian plane where IAT extends horizontally from 0 to ∞ and QoS extends vertically from a to b. This leads us to consider two cases: Case 1: $a \le y \le b$, Case 2: $y > b$

For Case 1:

$$G(y) = \int_a^b \int_0^{y-v} \frac{1}{b-a} \lambda e^{-\lambda u}\, du dv$$

$$= 1 - \frac{\mathrm{e}^{-\lambda(\mathrm{y-b})} - e^{-\lambda(y-a)}}{\lambda(b-a)}$$

For Case 2:

$$G(y) = \int_a^y \int_0^{y-v} \frac{1}{b-a} \lambda e^{-\lambda u}\, du dv$$

$$= \frac{y-a}{b-a} - \frac{1-e^{-\lambda(y-a)}}{\lambda(b-a)}$$

Now, the PDF of Y can be found by differentiating $G(y)$, this gives:

$$g(y) = \begin{cases} \dfrac{e^{-\lambda(b-a)} - e^{-\lambda(y-a)}}{(b-a)}, a \le y \le b \\ \dfrac{1-e^{-\lambda(y-a)}}{(b-a)} \qquad\qquad , y > b \end{cases}$$

Hence, the PDF of $MY = \min\limits_i Y_i$ can be found using the distribution function technique as follows:

Let $F(t)$ be the distribution function of MY, then

$$F(t) = P(MY \le t) = P\left(\min\limits_i Y_i \le t\right) = 1 - P\left(\min\limits_i Y_i > t\right) = 1 - (P(Y > t))^{n_2}$$

$$= 1 - (1 - G(t))^{n_2}$$

Therefore, the PDF of MY is given by

$$f(t) = n_2 (1 - G_Y(t))^{n_2 - 1} g_Y(t) \qquad (2)$$

Where

$$g_Y(t)=\begin{cases}\dfrac{e^{-\lambda(b_A-a_A)}-e^{-\lambda(t-a_A)}}{(b_A-a_A)}, a_A \le t \le b_A\\ \dfrac{1-e^{-\lambda(t-a_A)}}{(b_A-a_A)} \qquad ,t>b_A\end{cases}$$

Similarly,
Let $H(t)$ be the distribution function of MZ, then

$$H(t)=P(MZ\le t)=P\left(\min_i Z_i \le t\right)=1-P\left(\min_i Z_i > t\right)=1-(P(Z>t))^{n_3}$$
$$=1-(1-H(t))^{n_3}$$

Hence, the PDF of MZ is given by

$$h(t)=n_3(1-G_Z(t))^{n_2-1}g_Z(t) \qquad (3)$$

Where

$$g_Z(t)=\begin{cases}\dfrac{e^{-\lambda(b_V-a_V)}-e^{-\lambda(t-a_V)}}{(b_V-a_V)}, a_V \le t \le b_V\\ \dfrac{1-e^{-\lambda(t-a_V)}}{(b_V-a_V)} \qquad ,t>b_V\end{cases}$$

The probability we need now is $P(MY<MZ)$. The analytical evaluation of this probability requires the joint PDF of MY and MZ then the integration of some complicated functions over some restricted regions.

Now, for solving equation (1)some manipulations with the transformation techniques of random variables is used. This could result in the following boundary conditions:

$$P_{n_1,0,0}(t+\delta t)=P_{n_1-1,0,0}(t)\ \lambda\delta t+P_{n_1+1,0,0}(t)\ \mu\delta t+$$
$$P_{n_1,0,0}(t)(1-\lambda\delta t-\mu\delta t)+o(\delta t)$$

$$P_{k,n_2,n_3}(t+\delta t)=P_{k-1,n_2,n_3}(t)\ \lambda\delta t+P_{k,n_2-1,n_3}(t)\ \lambda\delta t+$$
$$P_{k,n_2,n_3-1}(t)\ \lambda\delta t+P_{k,n_2,n_3}(t)(1-3\lambda\delta t)+o(\delta t)$$

$$P_{s,n_2,n_3}(t+\delta t)=P_{s,n_2-1,n_3}(t)\ \lambda\delta t+P_{s,n_2+1,n_3}(t)\ \mu\delta t\ \ \beta+$$
$$P_{s,n_2,n_3-1}(t)\ \lambda\delta t+P_{s,n_2,n_3+1}(t)\ \mu\delta t\ (1-\beta)$$
$$P_{s,n_2,n_3}(t)(1-2\lambda\delta t-\mu\delta t)+o(\delta t)$$

Rearranging terms and taking limit as $\delta t\to 0$ we get:

$$\frac{d}{dt}P_{n_1,n_2,n_3}(t)=$$
$$\theta\ \Big[\ P_{n_1-1,n_2,n_3}(t)\ \lambda+P_{n_1+1,n_2,n_3}(t)\ \mu$$
$$+P_{n_1,n_2-1,n_3}(t)\ \lambda+P_{n_1,n_2,n_3-1}(t)\ \lambda+$$
$$+P_{n_1,n_2,n_3}(t)(-3\lambda-\mu)\ \Big]$$
$$+(1-\theta)\ \Big[\ P_{n_1-1,n_2,n_3}(t)\ \lambda$$
$$+P_{n_1,n_2-1,n_3}(t)\ \lambda+P_{n_1,n_2+1,n_3}(t)\ \mu\beta$$
$$+P_{n_1,n_2,n_3-1}(t)\ \lambda+P_{n_1,n_2,n_3+1}(t)\ \mu(1-\beta)$$
$$+P_{n_1,n_2,n_3}(t)(-3\lambda-2\mu)\ \Big]$$

$$\frac{d}{dt}P_{n_1,0,0}(t)=P_{n_1-1,0,0}(t)\ \lambda+P_{n_1+1,0,0}(t)\ \mu+$$
$$P_{n_1,0,0}(t)(-\lambda-\mu)$$

$$\frac{d}{dt}P_{k,n_2,n_3}(t)=P_{k-1,n_2,n_3}(t)\ \lambda\ \text{-}\ P_{k,n_2,n_3}(t)\lambda$$

$$\frac{d}{dt}P_{s,n_2,n_3}(t)=P_{s,n_2-1,n_3}(t)\ \lambda+P_{s,n_2+1,n_3}(t)\ \mu\beta+$$
$$P_{s,n_2,n_3-1}(t)\ \lambda+P_{s,n_2,n_3+1}(t)\ \mu(1-\beta)+$$
$$P_{s,n_2,n_3}(t)(-2\lambda-\mu)$$

In the steady state, all derivatives are zero and the probabilities are independent of time. So, we obtain:

$$0=\ \theta\ \Big[\ P_{n_1-1,n_2,n_3}\ \lambda+P_{n_1+1,n_2,n_3}\ \mu\ +$$
$$P_{n_1,n_2-1,n_3}\ \lambda+P_{n_1,n_2,n_3-1}\ \lambda+$$
$$P_{n_1,n_2,n_3}(-3\lambda-\mu)\ \Big]$$
$$+(1-\theta)\ \Big[\ P_{n_1-1,n_2,n_3}\ \lambda\ +$$
$$P_{n_1,n_2-1,n_3}\ \lambda+P_{n_1,n_2+1,n_3}\ \mu\beta\ +$$
$$P_{n_1,n_2,n_3-1}\ \lambda+P_{n_1,n_2,n_3+1}\ \mu(1-\beta)$$
$$+P_{n_1,n_2,n_3}(-3\lambda-2\mu)\ \Big] \qquad (4)$$

$$0=P_{n_1-1,0,0}\ \lambda+P_{n_1+1,0,0}\ \mu+P_{n_1,0,0}(-\lambda-\mu) \qquad (5)$$

$$0=P_{k-1,n_2,n_3}\ \lambda\ \text{-}\ P_{k,n_2,n_3}\lambda \qquad (6)$$

$$0 = P_{s,n_2-1,n_3}\ \lambda + P_{s,n_2+1,n_3}\ \mu\beta + P_{s,n_2,n_3-1}\ \lambda\ + P_{s,n_2,n_3+1}\ \mu(1-\beta) + P_{s,n_2,n_3}\left(-2\lambda-\mu\right) \qquad (7)$$

The evaluation of performance measures for the queuing system under consideration depends on the solution to the above system of difference equations (4, 5, 6, and 7). The above system of equations is apparently very complicated. What makes things worse is the interdependence between the different queues as the service for one of them begins after certain conditions concerning the other queues are satisfied. This interdependence presents great difficulties in obtaining a closed form analytical solution. In fact, even for two-queue systems with such interdependence the analytical solution is not available [15, 16]. The use of simulation techniques seems inevitable.

Analysis of the Hierarchical Diff-EDF

Assumptions and notations: Assume we have K packet flows and we want to determine whether they can be scheduled so that their QoS requirements are met. Each flow j is characterized by:

* A packet inter-arrival distribution (Exponential Distribution), with a mean of $1/\lambda_j$. Let $\Lambda = \sum_{j=1}^{k}\lambda_j$, the total arrival rate.
* A packet service requirement distribution (Exponential Distribution), with a mean of $1/\mu_j$.
* A soft deadline $D_j > 0$. For each flow j that is randomly drawn from a distribution G_j (in our case the uniform distribution), then D_j represents the mean of G_j. Let $D_j = \sum_{j=1}^{k}\lambda_j D_j/\Lambda$, the mean packet deadline averaged over all flows.
* Define $\rho_j = \lambda_j/\mu_j$ the traffic intensity of flow j and $\rho = \sum_{j=1}^{k}\rho_j$. In addition, we define $\alpha_j = \rho_j/\rho$, the faction of the traffic intensity that is attributed to flow j.
* A QoS requirement ϕ_j, interpreted as the requirement that the long run average fraction of flow j's packets missing their deadlines D_j must not exceed ϕ_j.
* The RTQT analysis used in this work models the workload process as a Brownian motion with drift $-\theta$, where:

$$\theta = \frac{2(1-\rho)}{\sum_{j=1}^{k}\lambda_j(\lambda_j^2+\mu_j^2)/\mu_j^2}$$

Packet processing: On receiving each packet from a certain packet flow (assume packet flow j), Diff-EDF performs the following operations:

* Identify the associated flow for the packet using Packet Type as an ID label (we define 1 for http data, 2 for audio and 3 for video) and lookup up the adjustment constant B_j. The relative deadline is then changed to the effective deadline according to the following equation: $D_{e,j} = D_j + B_j$
* Perform the ordinary EDF scheduling using the effective deadline. That is, the packet's absolute deadline is now given by:
$D'_{e,j} = D_{e,j} + t_a$
Where t_a is the packet's arrival time and insert the packet into the Diff-EDF queue in the order of increasing absolute deadline (smallest absolute deadline is at queue head thus served first).

Deadline miss rate prediction: Prediction of miss rate, per each flow j, is based on the Real-Time Queuing Theory (RTQT) analysis of the EDF algorithm. The basic methodology can be found in [17], [18], [19], or [20]. It had been found that when all flows have the same deadline miss rate, then it can be computed by:

$$\phi_j = e^{-\theta\overline{D}}$$

For Hierarchical Diff-EDF, we will adjust the deadlines of each flow j by adding a constant B_j to the deadline. (Constant B_j can be either positive or negative value). Using the above equation, it is obvious that when using Hierarchical Diff-EDF, the deadline miss rate for each flow can be computed by:

$$\phi_j = e^{-\theta(\overline{D'}-B_j)}$$

Where: $\overline{D'} = \sum_{j=1}^{k}\alpha_j(D_j + B_j)$

Determination of B_j's: As we mentioned before, one of the Hierarchical Diff-EDF system components is the Marker. A Marker adjusts the deadlines of incoming packets by adding a constant Bj , different constants are added to different flows, to the relative deadlines before the packets are placed into the queue. After tuning the Diff-EDF system to achieve the best QoS flow's requirements, it had been found that the constant values of B_j could be computed by:

$$B_j = \frac{1}{\theta}\log\frac{\phi_j}{\phi_1}, 1 \le j \le k$$

Where $1/\theta$ is the mean of the exponential stationary distribution of the workload process, ϕ_j is the deadline miss rate of the flow j, ϕ_1 is the smallest deadline miss rate among the flows and k is the number of the flows to be serviced.

If we assume B_1 is the constant to be added for the deadlines in the video flow and that the video flow has the smallest deadline miss rate, then by applying the above equation: $B_1 = 1/\theta \log 1 = 0$. Hence, once the B_j for the high priority flow is determined, the Diff-EDF system will select a much larger values of B_j's for the flows to be run at low priority.

Generating arrival packets: In order to generate the arrival packets, a number of arguments must be determined as the following:

* Number of sources in the system as s.
* Number of flows in the system as k.
* Total Number of Arrival rate for all flows j as $\lambda_T, 1 \le j \le k$.
* Arrival rate per flow j as $\lambda_j = \lambda_T / k$.
* Inter-arrival mean per flow j as $1/\lambda_j$
* Relative Deadline range per flow j as (QoS_{max}, Qos_{min}).

It is worthy to mention here that the total number of arrival rate in the system should equal the summation of all the arrival rates per each flow j as in the following equation: $\lambda_T = \sum_{j=1}^{k} \lambda_j$.

Now, after determine the arguments we start generating the packets. Two Steps were carried out:

1. Using the Exponential Distribution, with a mean of $1/\lambda_j$, to generate the inter-arrival time of the different packets.
2. Using the Uniform Distribution, with a mean of (QoS_{max}, Qos_{min}), to generate the relative deadlines for the different packets.

An important issue that been taken into consideration when generating the arrival packets was to ensure that the Traffic Generator always obtain an initial seeds for the different streams. This mechanism will ensure that each flow is following a specified seed, when its packets are generated, to reflect the real-world traffic and leads for high accuracy. To do that, a Seed Initializer method is used to initialize seeds for the variety streams based on the defined mathematical techniques in[21].

Determination of effective deadlines range: As we mentioned earlier, the Uniform Distribution with a mean of (QoS_{max}, Qos_{min}) is used to generate the relative deadlines for the different packets. QoS_{max} and QoS_{min} are the Effective Deadlines range for each flow *j*. Calyam and Lee in[22] had built a voice and video traffic measurement testbed to determine their effective deadline ranges. The Good range corresponds to delay values of (0-150) ms, the Acceptable range corresponds to delay values of (150-300) ms, while the Poor range corresponds to delay values > 300ms. Now, by observing Fig. 4 and 5, we can conclude:

* The Real Good Range for the video traffic between (40, 150).
* The Real Good Range for the audio traffic between (60, 160).
* The Acceptable Range for both video and audio traffic between (150, 300).
* The poor Range for both video and audio traffic is greater than 300.

Based on the above real results, collected in large-scale Internet, we choose our effective deadline, to achieve the highest system throughput, in the following manner:

* Effective Video Deadline in the Good range (40, 150).
* Effective Audio Deadline in the Acceptable range (250, 160).
* Effective Text Deadline in the Poor range (assume 400, 300).

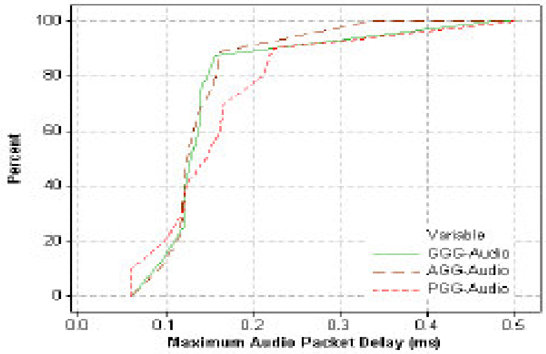

Fig. 4: Maximum audio packet delay

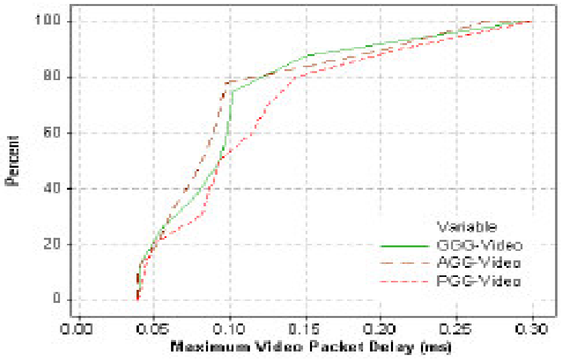

Fig. 5: Maximum video packet delay

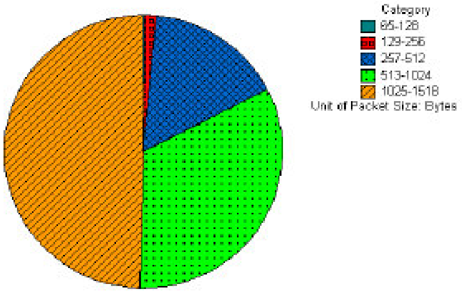

Fig. 6: Packet distribution

- The reason to choose the Effective deadline for the text traffic in the poor range is coming from the fact that it is less sensitive to the delay compared with the multimedia traffic.

System queues implementation: In Hierarchical Diff-EDF system, two different queues were implemented as the following:

1. Diff-EDF queue: This queue is implemented using Sorted Linked List, where the sorting is based on the min value of $(D'+i_{at})$. The reason to choose this type of Linked List is to reduce scheduler complexity, so that rather than the scheduler will spend the time in picking up the shortest lead time packet to get serve, the queue is ready to be served starting from the queue head. This type of queue will serve all real-time flows (video and audio).
2. FCFS Queue: This queue is implemented using Linked List. The events are queued based on their Inter-arrival time (i_{at}), with smallest i_{at} at the head of the queue. This type of queue will serve the non-real-time traffic (http data).

Feedback control mechanism: The Hierarchical Diff-EDF scheduling algorithm features a feedback control mechanism that detects overload conditions and modifies packet priority assignment accordingly. To do that, the algorithm is implemented with a feedback control mechanism (Threshold limitation). In other words, the server always serves the packets in the Diff-EDF queue (high priority) and serves the FCFS queue (low priority) if either the Diff-EDF queue is empty or the FCFS queue reaches its threshold value. Now, after tuning the system to achieve the highest performance, through meeting all the flows deadline miss rates ϕ_j, it has been found that the threshold value is when $\text{FCFSQsize} >= 0.9\lambda_T / k$. On the other hand, the stopping case was found when the value approaches to $0.7\lambda_T / k$.

System parameters: For this system a number of parameters were set as the following:

- Packet Size: the packet size was chosen to be of 1500 Byte. The reason to choose this value is that almost more than 50% of the traffic being propagated has a packet size of 1500 Byte (Fig. 6).
- Bandwidth B_w: the bandwidth was chosen to be 3 Mbps. The reason to choose this value is based on the following facts:
  - Video traffic consumes a highest bandwidth with a value close to 5 Mbps.
  - Audio traffic roughly consumes 3 Mbps.
  - Low traffic consumes about 1 Mbps.
- The Aggregate Average Bandwidth = (5 Mbps + 3 Mbps + 1 Mbps) / 3 = 3 Mbps
- Total Arrival Rate λ_T: the simulation was carried out for λ_T start at 5000 packet up to 60000 packet with a 5000 packet simulation step.
- Mean Service μ: the mean service was calculated with the following equation:
- Mean Service $(\mu) = 8 * \text{PacketSize} / B_w$.
- Number of Sources s: the simulation was carried out with a 50 generated sources.
- The Experiments were carried out with three different flows; two of them are real-time traffic (video and voice) while the third flow is non real-time traffic (http data or text).

Comparative analysis: The simulation has been run for arrival rates (λ_T) of 10000 – 60000 packets with an increment step of 5000 packets. The bandwidth is assumed to be 3 Mbps while the packet size 1500 Byte. The analysis elaborates different performance metrics with focus on the average delay values per each flow j.

Here, four graphs were plotted to evaluate the efficiency of the Hierarchical Diff-EDF against both EDF and Diff-EDF with respect to the average delay. Figure 7 shows the packet average delay of the video flow. The results show that the average delay of the Hierarchical Diff-EDF is of linear shape with difference of 1.6 ms before it reaches the steady state when the system is extremely overloaded with difference of < 1ms. It is obvious that the peak improvement of the Hierarchical Diff-EDF over the EDF is achieved when the system is overloaded with 45000 packets; at this point the average delay difference reaches its maximum value with 31.6ms. When compared with Diff-EDF, the peak improvement is achieved when the system reaches its equilibrium point; at this point the average delay difference reaches its maximum value with 29.7 ms.

In Figure 8, we compare the average delay of the audio flow for the three schedulers. Again, the results show that the Hierarchical Diff-EDF average delay is still of linear shape at a smaller difference of 2.2ms before reaching system stability when it is extremely overloaded. It is noticeable here that the Hierarchical Diff-EDF achieves a remarkable performance over both EDF and Diff-EDF with a peak improvement of 124 ms and 101 ms respectively detected when the system is overloaded.

For measuring the average delay of the text flow, figure 9 was plotted. In this figure, we can observe that the Hierarchical Diff-EDF achieves a much lower average delay compared to both EDF and Diff-EDF. The Hierarchical Diff-EDF still keeps its linearity with 3.8 ms. The peak improvement is achieved at the equilibrium point of a moderate load with 40000 packets. This superior improvement is attributed for deploying a feedback control mechanism which enforces the desired service guarantees and prevent text flow packet starving.

Finally, to compare the total average delay for both schedulers, we plotted figure 10. The results show that Hierarchical Diff-EDF achieves the minimum delay, with 2.5 ms linear difference, compared to the EDF with 12 ms average difference and Diff-EDF with 11 ms. The peak improvement for the Hierarchical Diff-EDF over both EDF and Diff-EDF was 93 ms and 94 ms delay respectively. Hence, we can conclude that the Hierarchical Diff-EDF shows a remarkable scheduling performance over heterogonous network traffic by achieving the minimum total average delay.

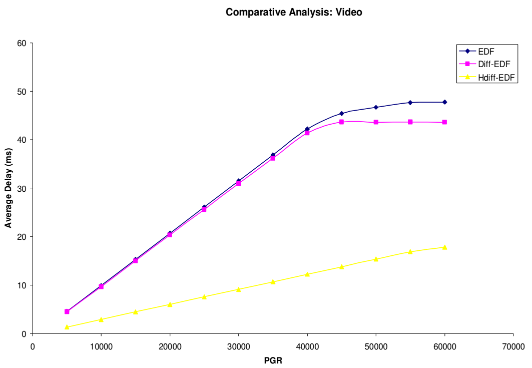

Fig. 7: Average delay – video traffic

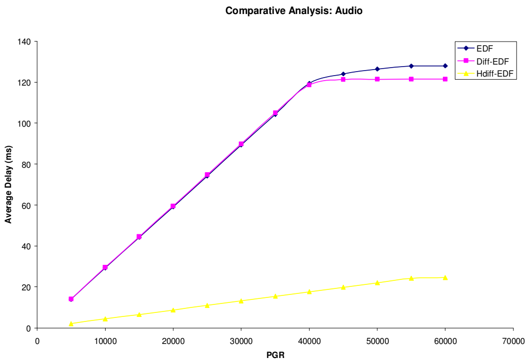

Fig. 8: Average delay – voice traffic

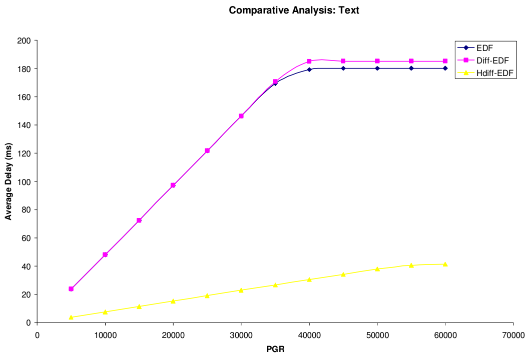

Fig. 9: Average delay – text traffic

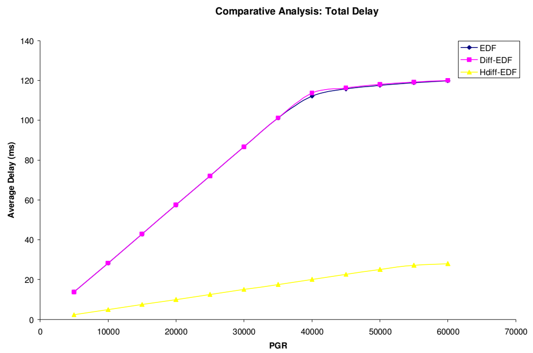

Fig. 10: Total Average delay

CONCLUSION

For the past decade, a significant amount of research in data networks and telecommunications has been devoted to providing different levels of service guarantees. In this study, we have presented the new priority assignment scheduling algorithm Hierarchical Diff-EDF (Differentiate Earliest Deadline First), which can meet the real-time needs while continuing to provide best effort service to the non-real time traffic, over heterogeneous real-time network traffic. This Hierarchical Diff-EDF features a feedback control mechanism that detects overload conditions and modifies packet priority assignments accordingly. Also, our scheduler considers each flow as having stochastic traffic characteristic, a stochastic deadline and a maximum allowable miss rate. The Hierarchical Diff-EDF service meets the flow miss rate requirements through the combination of the hierarchal scheduling for the different network traffic classes (real-time and non real-time) and the admission control mechanism that detects the overload conditions to modify packet's priorities.

For examining the proposed scheduler model, we presented our attempt for providing an exact analytical solution. However, due to the high interdependency between the different queues' service, the solution was very complicated. Consequently, the use of simulation techniques seems inevitable. The simulation results show that the Hierarchical Diff-EDF scheduler produces a better performance of packet serving over heterogeneous network traffic through achieving the minimum miss ratio. This improvement is attributed to the use of the QoS priority based packet assignment.

Future research: Proposed future research is as follows:

* To evaluate the performance of the Hierarchical Diff-EDF scheduler under a multiprocessor environment by using the parallel simulation technique.
* To compare the performance of the Hierarchical Diff-EDF scheduler with other different scheduling algorithms such as Round Robin RR, Weighted Fair Queuing (WFQ), Worst-Case Fair Weighted Fair Queuing (WF^{2}Q) and Largest Processing Time First (LPT) under same operating environments.
* To design a more flexible and friendly user interface in a way that allows a user to specify more system parameters with different input data ranges such as bandwidth, packet size, arrival rate (λ), service rate (μ), number of sources, threshold value, type of traffic and size of multimedia stream.
* To test the simulation with other different probability distribution functions rather than using the Poisson distribution and compare the obtained graphs with the collected one.
* To carry out thorough performance evaluation studies between the Hierarchical Diff-EFD and GPS scheduling algorithms in situations where the GPS is preferable than EDF.

REFERENCES

1. Srinivasan, A., 2003. Efficient and Flexible Fair Scheduling of Real-time Tasks on Multiprocessors. Chapel Hill.
2. Zhang, H, and D. Ferrari, 1993. Rate-controlled static-priority queuing. In: Proc. of IEEE INFOCOM ‘93, 1: 227-236.
3. Lu, C.Y., A. Stankovic, G. Tao and H.S. Son, 2002. Feedback control real-time scheduling: framework, modeling and algorithms. Special issue of Real-Time Systems J. on Control-Theoretic Approaches to Real-Time Computing, 23: 85-126.
4. Dertouzos, M., 1974. Control robotics: The procedural control of physical processes. In: Proc. of IFIP Congress, pp: 807-813.
5. Jackson, J.R., 1955. Scheduling a production line to minimize maximum tardiness. Research Report 43, Management Science, University of California, Los Angeles, CA.
6. Towsley, D. and S.S. Panwar, 1991. Optimality of the stochastic earliest deadline policy for the G/M/c queue serving customers with deadlines. COINS Technical Report 91-61, Univ. Massachusetts.
7. Giorgio, C.B. and J. Stankovic, 1993. RED: Robust earliest deadline scheduling. In: Proc. 3rd Intl. Workshop on Responsive Computing Systems, pp: 100-111.

8. Giorgio, B., M. Spuri and F. Sensini, 1995. Values vs. deadline scheduling in overload conditions. IEEE Real-Time Systems Symposium.
9. Firoiu, V., J. Kurose and D. Towsley, 1997. Efficient admission control for EDF schedulers. In: Proc. of IEEE INFOCOM ‘97, 1: 310.
10. Stankovic, A., M. Spuri, K. Ramamritham and G.C. Buttazzo, 1998. Deadline scheduling for real-time systems – EDF and related algorithms. Kluwer Academic Publishers, Boston, Massachusetts, USA.
11. Tak-Wah, L., T.W. Johnny Ngan and K.K. To, 2004. Performance guarantee for EDF under overload. J. Algorithms, 52: 193-206.
12. Harista, J.R., M. Livny and M.J. Carey, 1991. Earliest deadline scheduling for real-time database systems. IEEE Real-Time Systems Symp.
13. Zhu, H.F., J.P. Lechoczky, J.P. Hansan and R. Rajkumar, 2005. Diff-EDF: A simple mechanism for differentiated EDF service. Proc. of the 11th IEEE Real-Time and Embedded Technology and Applications Symp.
14. Hogg, R.V. & Craig, A.T. 1978. Introduction to Mathematical Statistics. New York, Macmillan Publishing.
15. Resing, J.A.C. 1993. Polling Systems and Multitype Branching Processes. Journal of Queueing Systems. 13: 409-426.
16. Winands, E.M.M., Adan, I.J.B.F. & Houtum, G.J. 2005. A Two-Queue Model with Alternating Limited Service and State-Dependent Setups. Proceedings of Analysis of Manufacturing Systems - Production Management, pp. 200-208.
17. Zhu, F., Lehoczky, J.P., Hansen J.P. & Rajkumar, R. 2005. Diff-EDF: A Simple Mechanism for Differentiated EDF Services. Proceedings of the 11th IEEE Real-Time and Embedded Technology and Applications Symposium, pp. 268-277.
18. Lehoczky, J.P. 1996. Real-Time Queuing Theory. In Proceedings of the 17th IEEE Real-Time Systems Symposium, pp. 186-195.
19. Lehoczky, J.P. 1998. Scheduling Communication Networks Carrying Real-Time Traffic. In Proceedings of the IEEE Real-Time Systems Symposium, pp. 470-479.
20. Zhu, H.F., Hansen, J.P., Lehoczky, J.P. & Rajkumar, R. 2002. Optimal Partitioning for Quantized EDF Scheduling. In Proceedings of the 23rd IEEE Real-Time Systems Symposium, pp. 212-222.
21. Marse, K. & Roberts, S.D. 1983. Implementing a Portable FORTRAN Uniform (0,1) Generator. Simulation. 41: 135-139.
22. Calyam, P. and Lee, C. Characterizing voice and video traffic behavior over the Internet, International Symposium on Computer and Information Sciences (ISCIS), 2005.