# Cell-Level Multiplex Scheduling to Support Multimedia Applications in Real-Time Channels

Hyung-Ill Kim and Sungyoung Lee
Department of Computer Engineering
Kyung Hee University, Korea
*hikim@oslab.kyunghee.ac.kr, slee@nms.kyunghee.ac.kr*

## Abstract

*A resource reservation mechanism is widely used in multimedia application systems to guarantee quantitatively specified Quality of Services (QoS). A real-time channel, which is one of the network resource reservation mechanisms for the real-time applications introduced by Ferrari [FERRARI], promises to support the different QoS requirements. When determining the channel bandwidth reservation level, a real-time channel however, takes into account the worst case of real-time traffic's burstness which in consequence causes a lower channel utilization. In order to improve the utilization for a given real-time channel as well as to support a different level of QoS requirements, we propose a cell multiplexing scheme in a source node. Since the scheduler in a source can easily recognize the behavior of data traffic, the incoming message can be classified into four different types i.e., real-time, multimedia, aperiodic, and non-real-time message. Then, the algorithm schedules them based on the Fixed Priority and EDD+ (Earliest Due Date+), which is an extension of the EDD scheduling, and multiplexes these four types of cell. Our simulation study shows that the proposed algorithm considerably reduces the missing ratio of real-time traffic within a given channel bandwidth level, compared to the case where multiplexing is not considered.*

## 1. Introduction

A resource reservation mechanism [ANDERSON, MERCER] is widely used in multimedia applications to guarantee quantitatively specified QoS. Multimedia communication involving mixture of continuous and non-continuous media has rather stringent timing requirements. A real-time channel scheme introduced by Ferrari [FERRARI, BANERJEA] is one of the network resource reservation mechanisms [BRADEN, KARLSSON, ZHANG] which can be used for multimedia applications in a high-speed network, such as an ATM [PRYCKER]. A real-time channel guarantees the performance by means of allocating the lower bounds on the bandwidth to a channel and upper bounds on the delays to be expected by a packet on the channel. A real-time channel is a simplex connection between a source and a destination characterized by parameters representing the performance requirement of the user. The data flow on real-time channels is unidirectional, from source to sink via intermediate nodes, with successive messages delivered in the order they were generated. Corrupted, delayed, or lost data is of little value; with a continuous flow of time-sensitive data, there is not sufficient time to recover from errors. Data transfer on real-time channels has unreliable-datagram semantics, i.e., occurs without acknowledgement. The overhead of the real-time channel establishment is to be acceptable in real applications.

When determining the lower bound channel bandwidth, however, a real-time channel uses the worst case of real-time traffic burstness which in consequence causes a lower channel utilization. Obviously, when establishing real-time channel, the foremost concern is how to effectively decide lower bound channel bandwidth with a channel efficiency. Nevertheless, the concern still remains how to improve performance of a real-time channel in an efficient manner, which is the major motivation of this work.

Multiplexing is one of the conventional mechanisms to improve channel utilization [CHOU]. In high-speed networks, a statistical multiplexing is commonly used [ROSBERG] in network switches to improve channel utilization and to support fairness of data traffic. However, that scheme is expensive and inflexible to implement since it is difficult to recognize the characteristics of each type of traffic in detail due to the high-speed. We therefore adopt a cell multiplexing scheduling in a source node which makes it relatively easy to recognize the behavior of traffic a priori. In order to meet the different user-level QoS requirements, we classify data into four categories: real-time, multimedia, aperiodic,



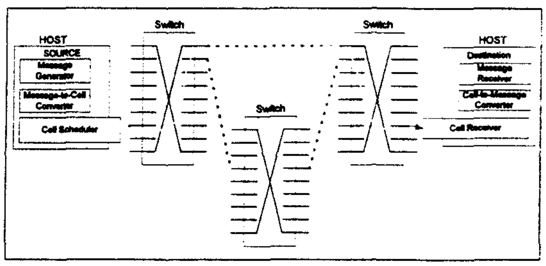
0-8186-8034-2/97 $10.00 © 1997 IEEE

and non-real-time message. We also decompose a physical real-time channel into the corresponding four different types of sub-channel, i.e., a real-time, a multimedia, an aperiodic, and a non-realtime sub-channel, respectively. A sub-channel is informally defined as a stream of messages. Then, cell scheduler schedules them based on the Fixed Priority [LIU] and EDD+ scheduling algorithms and multiplexes the four different types of sub-channels. The EDD+ is an extension of the EDD, the only difference being that EDD+ scheduling has a cell discarding mechanism before the cell is transmitted if it is expected to miss its deadline at a destination when it arrives.

The remainder of this paper is organized as follows: In Section 2 we present our system model. In Section 3 we describe the proposed algorithm. A Section 4 demonstrates our simulation study. Section 5 describes some related works. Finally, Section 6 provides a few concluding remarks.

## 2. System Model

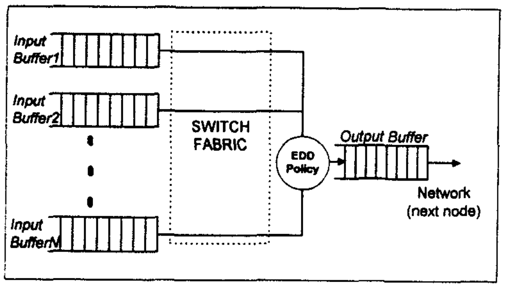
The proposed cell multiplexing performs on a real-time channel which was already established between a source and a destination. An incoming message at a source is chopped up into several message cells through the message-to-cell converter and the scheduler transmits those four different types of cell based on the proposed algorithm, mixture of fixed priority and EDD+ scheduling. Then, the cells arriving at a destination will be reassembled into a message (Figure 1). Figure 2 shows the structure of network switch which appears on the model in Figure 1. At this time, the source already knows the maximum delay for the each incoming message cell.




**[Figure 1] Cell transmission model**

We define different cell types and their corresponding sub-channels as follows:

**Definition 1)** A real-time channel ($C_{RT}$) consists of more than one sub-channel and is defined as follows:




**[Figure 2] Structure of A Network Switch**

$$C_{RT} = \{ C_R(i), C_M(j), C_A(k), C_N(l) \},$$
$$n = i + j + k + l$$

where $C_R$, $C_M$, $C_A$, $C_N$ means real-time, multimedia, aperiodic, and non-real-time sub-channel respectively, and $i$, $j$, $k$, $l$ are the numbers of each sub-channel. The sub-channel means a stream of messages and each sub-channel is defined as follows:

$$C_R(i) = \{ C_{R1}, C_{R2}, \ldots, C_{Ri} \},$$
$$C_M(j) = \{ C_{M1}, C_{M2}, \ldots, C_{Mj} \},$$
$$C_A(k) = \{ C_{A1}, C_{A2}, \ldots, C_{Ak} \},$$
$$C_N(l) = \{ C_{N1}, C_{N2}, \ldots, C_{Nl} \}$$

**Definition 2)** $R_m$ is the real-time message which has period and relative deadline ($D_i$) and it consists of $l_i$ number of cells ($R_m(l)_i = l_i$ cells where is $i$ the $i^{th}$ real-time sub-channel). The $j^{th}$ real-time message on the $i^{th}$ real-time sub-channel can be represented by $R_m(l_i, D_i)_{i,j}$. The relationship of the absolute deadline between adjacent real-time messages can be expressed by $R_m(d)_{i,j+1} = R_m(d)_{i,j} + D_i$.

**Definition 3)** $M_m$ is the multimedia message (continuous message) and it has period and deadline. The $j^{th}$ multimedia message on the $i^{th}$ multimedia sub-channel can be represented by $M_m(l_i, D_i)_{i,j}$. The average length of multimedia message ($l_{ave}i$) varies between $l_{min}i \leq M_m(l)_{i,j} \leq l_{max}i$. The minimum length and maximum length are given in advance. The relationship of the absolute deadline between adjacent multimedia messages can be expressed by $M_m(d)_{i,j+1} = M_m(d)_{i,j} + D_i$. The transmission priority of $M_m$ at source is lower than that of $R_m$.

**Definition 4)** $A_m$ is the aperiodic message and it has no period and no deadline. The $j^{th}$ aperiodic message on the $i^{th}$ aperiodic sub-channel can be represented by $A_m(l_{i,j})_{i,j}$. The average length of aperiodic message ($l_{ave}i$) varies between $l_{min}i \leq A_m(l)_{i,j} \leq l_{max}i$.

**Definition 5)** $N_m$ is the non-real time message and it has no period and no deadline. The $j^{th}$ non real-time message on the $i^{th}$ non real-time sub-channel can be represented by $N_m(l_{i,j})_{i,j}$. The average length of non real-time message ($l_{ave}i$) varies between $l_{min}i \leq N_m(l)_{i,j} \leq l_{max}i$. The transmission priority of $N_m$ at source is lower than that of $A_m$.

Meanwhile, to simplify our multiplexing algorithm, we assume the following:

**A1)** A real-time channel which guarantees maximum message delay and minimum channel bandwidth has already been established between a source and a destination.

**A2)** All of the cell sizes are the same.

**A3)** A preemption is allowed between the messages, but is not allowed between the cells.

## 3. Cell Multiplexing

In this section, we describe the cell-multiplexing algorithm. The major theme of this paper is to study how to maximize the given real-time channel bandwidth utilization, while supporting different QoS requirements. A statistical multiplexing is commonly used for the high-speed network switch. That scheme generally improves the channel utilization and supports the fairness of data traffic. However, in a high speed network switch, it is very expensive to know the characteristics of each kind of traffic such as deadline and user's QoS requirements due to the high-speed. On the other hand, in a source, it is much less expensive to recognize what characteristics incoming messages have. This is the reason for adopting a cell-level multiplex scheduling in a source node.

As we mentioned before, we classify message into four categories, i.e., real-time, multimedia, aperiodic, and non-real-time messages, to support the different QoS requirements. We also decompose a physical real-time channel into the corresponding four different types of sub-channel which is defined as a stream of messages, i.e., a real-time, a multimedia, an aperiodic, and a non-realtime sub-channel, respectively. Then, the proposed algorithm schedules them based on the fixed priority and dynamic priority (EDD+) scheduling algorithms and multiplexes four types of sub-channel. Consequently, when scheduling is to be performed at a source mode, each type of message maps into its corresponding sub-channel respectively. In this case, the scheduler must guarantee the deadline of a real-time message which has a stringent timing requirement. A multimedia message must be discarded when it can not meet its deadline due to exceeding a limitation of the given channel bandwidth. A non-real-time message must be guaranteed to transmit even though it has the lowest transmission priority. If the several non real-time messages are racing, a scheduler must consider their fairness.

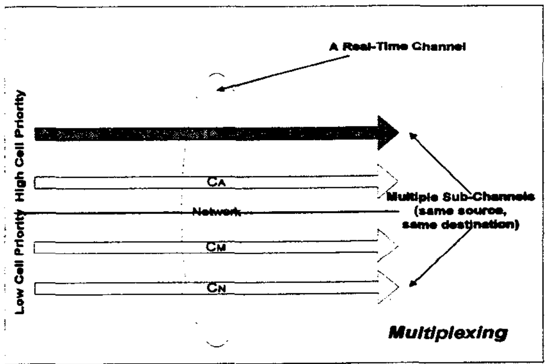
The proposed algorithm works basically on the real-time scheduling paradigm [LIU, CHETTO, BURN, LEE, LECHOCZKY1, LECHOCZKY2]. A real-time scheduling is well suited to handling the different types of task behavior, such as periodic and aperiodic, and with deadline and without deadline. Moreover, a real-time scheduling is generally a preemptive scheduling to meet timing requirements.

The characteristics of messages such as type and applied scheduling policies are depicted in Figure 3.

| message type | period & deadline | cell priority | cell scheduling policy | message size |
|---|---|---|---|---|
| *Rm* | fixed period, fixed deadline | high | fixed-priority | fixed size |
| *Mm* | fixed period, fixed deadline | low | EDD+ | variable size |
| *Am* | stochastical period, no-deadline | high | fixed-priority | unknown size |
| *Nm* | no-period, no-deadline | low | EDD+ | unknown size |

**[Figure 3] Message type and cell scheduling priority**

Figure 4 shows the basic idea of the proposed algorithm in which the scheduler at a source node is to perform cell scheduling based on the two levels of priority; fixed priority and EDD+. In this case, the priority of a real-time message and an aperiodic message cells is higher than that of a multimedia and a non-real-time message.




**[Figure 4] Basic idea of the cell-multiplexing**

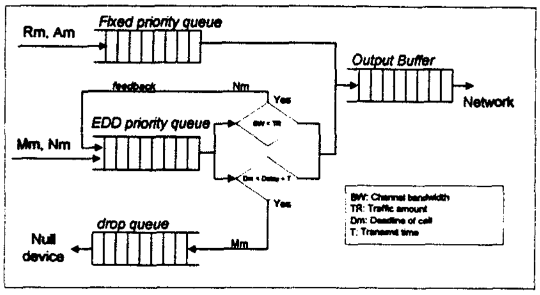
Figure 5 depicts pseudocode of the proposed cell multiplexing algorithm.

```
1 Proc Transfer_Message
2 chop a message up cells and insert the cells
    into a priority_queue
3 pick a cell from the queue header
4 CASE (cell_type)
5   PERIODIC REAL-TIME message:
6     transmit the cell to the channel
      immediately
7   APERIODIC REAL-TIME message:
8     set deadline to unlimited and transmit
      the cell in the channel immediately
9   MULTIMEDIA message:
10    If (deadline of the cell > current time +
         maximum_delay_bound)
      Then transmit the cell to the channel
11    Else insert the cell into a drop_queue
12  NONREALTIME message:
13    If (the channel bandwidth is not over)
         Then transmit the cell in the channel
14    Else insert the cell into a delay_queue
15 EndCASE
```

**[Figure 5] Pseudocode of the cell-multiplexing algorithm**

At line 2 in Figure 5, after the scheduler puts the incoming message cells into a priority queue, it transmits the highest priority message to the network. In the case of a real-time message cell, it is transmitted into the real-time channel immediately. In the case of a multimedia (continuous media) message cell, it is transmitted if the deadline of the message cell is greater than the completion time of that message at a destination node; otherwise it is put into a drop queue. In the case of a non-real-time message cell which should not be lost, a scheduler examines whether the reservation level of the channel bandwidth is enough or not to transmit without any loss. If so, it is transmitted; otherwise put into a delay queue for later transmission.

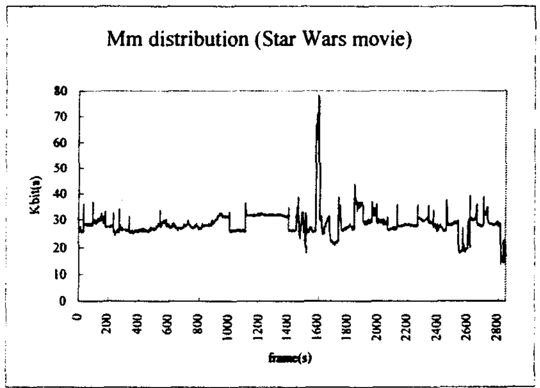
The Figure 6 shows an implementing model of the proposed cell multiplexing algorithm which is self-explanatory.




**[Figure 6] The Implementing Model**

## 4. Simulation

In this section, we demonstrate the result of a simulation study to show what scheduling policy is most appropriate and efficient for cell multiplexing in a real-time channel. Three scheduling policies such as FCFS, EDD, and EDD+ are compared. In the simulation, we use a part of the Star Wars movie as a multimedia traffic which was encoded based on the MPEG-1. Its distribution is shown on the Figure 7. The real-time and multimedia messages occur periodically through a whole simulation period, while aperiodic and non-real-time messages occur sporadically. The distribution of the non-real-time traffic is Poisson and its delay is long enough. Figure 8 shows the characteristics of the initial traffics.



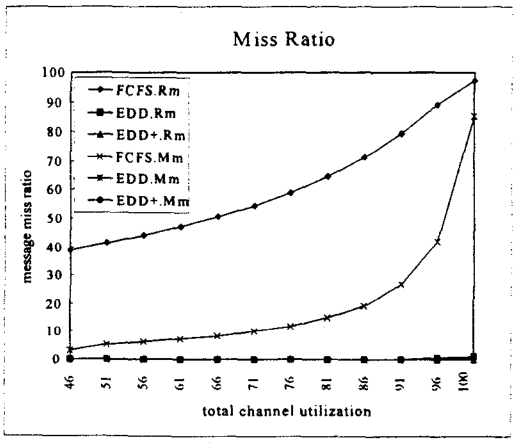

**[Figure 7] Mm traffic simulation**

| No | message type | arrival pattern(period) | average message size | min message size | max message size |
|---|---|---|---|---|---|
| 1 | hard periodic (*Rm*) | periodic (1ms) | 1kb | 1kb | 1kb |
| 2 | hard periodic (*Rm*) | periodic (5ms) | 2kb | 2kb | 2kb |
| 3 | continuous media (*Mm*) | periodic (40ms) | 4.30kb from MPEG-1 movie | 2.59kb | 11.68kb |
| 4 | continuous media (*Mm*) | periodic (40ms) | 4.56kb from MPEG-1 movie | 2.75kb | 12.37kb |
| 5 | aperiodic real-time (*Am*) | poisson distribution average 100ms | exponential 15.36kb | 0b | 49.74kb |
| 6 | non-real-time (*Nm*) | poisson distribution average 1000ms | exponential 119.1kb | 0.57kb | 591.9kb |

**[Figure 8] The initial simulation situation**

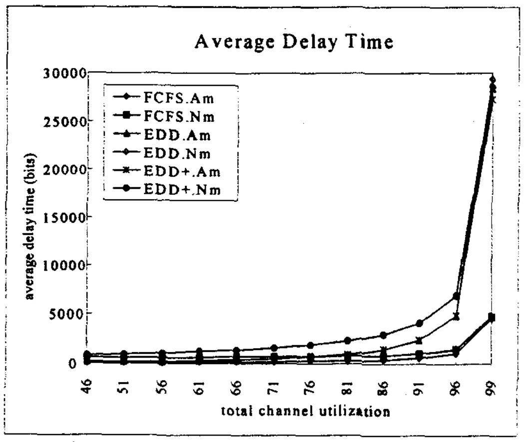
In the simulation, we assumed that the amount of aperiodic message is relatively very small compared to other types of message. In order to have a simple simulation environment, a hard aperiodic message is not considered since it requires an acceptance test [LEHOCZKY1, LEE] which lets the cell multiplexing algorithm be too complex to implement. A non-realtime message has a behavior similar to that of a normal text message which has variable message size with no period and large delay.

The evaluation criteria of the simulation are a miss ratio (MR) and an average delay time (ADT) for each message. Figure 9-A shows the first case of simulation which indicates the miss ratio of traffic when the amount of real-time traffic is increasing approximately 5% whereas other kind of traffic (multimedia, aperiodic, and non-realtime message) are constant. From the results of simulation, we know that the FCFS scheduling policy suffers from the highest miss ratio in almost all messages while EDD or EDD+ scheduling have no effect on the miss ratio. In the case of burst messages, a small amount of real-time messages are missed under the EDD scheduling. The reason is that since some of the real-time traffic which is expected to miss its deadline at destination was transmitted, a delay occurs.



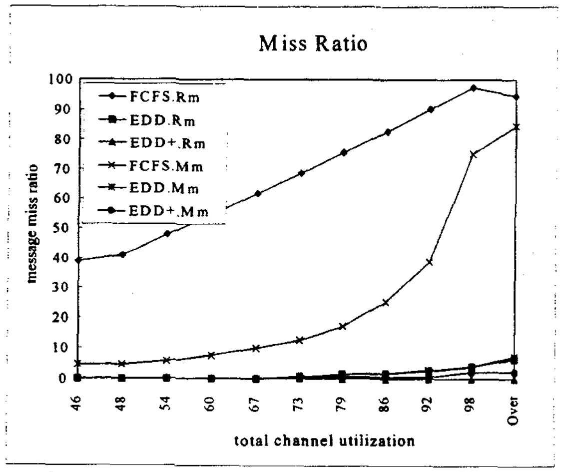

**[Figure 9-A] Miss Ratio as per Rm traffic increasing**

However, in case of EDD+ scheduling, no real-time traffic has been missed. Meanwhile, the delay of aperiodic and non real-time traffic increases exponentially under the EDD and EDD+ scheduling if the channel workload is over 90% (Figure 9-B). This kind of phenomenon is caused by the fact that since the real-time message and multimedia message have deadlines, they are served before the non-realtime message and aperiodic message are.




**[Figure 9-B] ADT as per Rm Traffic increasing**

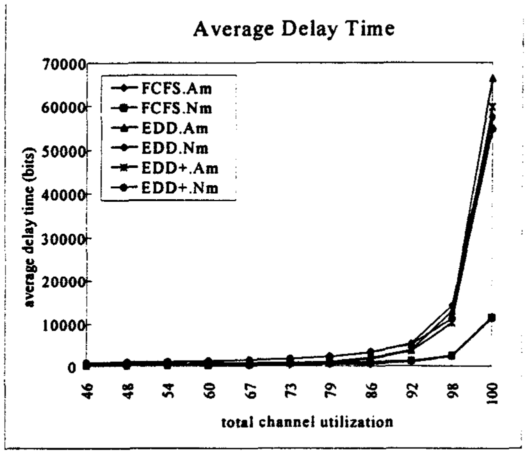
Figure 10-A shows the second case of simulation, which demonstrates the miss ratio when the amount of multimedia traffic is increasing approximately 7% whereas other traffics (real-time, aperiodic, and non-realtime message) are constant. The result of FCFS is almost same as the first case of simulation. In case of EDD, the missing ratio of real-time and multimedia message increases under the heavy traffic. On the other hand, in the case of EDD+, no real-time traffic has been missed. The EDD+ resolves a traffic missing problem which appeared on the EDD by means of giving a high priority to the real-time message.




**[Figure 10-A] Miss Ratio as per Mm traffic increasing**

Meanwhile, similar to the first case, the delay of aperiodic and non real-time traffics increases exponentially under the EDD and EDD+ scheduling if the channel workload is over 90% (Figure 10-B).

From our comparison study, we conclude that the FCFS is not preferred to cell multiplexing in a real-time channel at source node because it suffers from high cell loss in most cases. The EDD scheduling is preferred; however, it suffers from small amount of missing traffic for real-time data when there is burst traffic. Finally, the EDD+ scheduling policy is the most preferred one which guarantees the deadlines of real-time traffic in every case since it has a mechanism for selectively discarding the messages in advance that are expected to miss their deadlines at a destination.




**[Figure 10-B] ADT as per Mm Traffic Increasing**

## 5. Related Works

A few researchers have investigated the field of cell multiplexing at source nodes compared to cell multiplexing at network switches. Rosberg [ROSBERG] studied how various multiplexing schemes impact the cell delay variation of each connection in the absence of a traffic shaper control function. He compared five multiplexing policies; FIFO, Round Robin (RR), Least Time to Reach Bound (LTRB), Most Behind Expected Arrival (MBEA), and Golden Rate (GR) for constant bit rate traffic sources. He concluded that there is no single best multiplexing scheme with respect to all criteria. Nevertheless, the MBEA is the most preferred, if cell delay is not crucial; otherwise, FIFO or RR are preferred. However, his work focused only on multiplexing in ATM network switches which is different from our cell multiplexing scheme at source node.

Chou and Shin [CHOU] introduced a scheme for real-time communication on multi-access networks which can provide performance guarantees for given traffic-generation characteristics and performance requirements. In order to let the system add/delete real-time channels independently, they used multiplexing real-time channels originating from the same node. Their scheme achieved higher network utilization without compromising the capability of independent addition/deletion and the performance guarantees of real-time communication. However, their working environment is a multi-access network which is different from ours multi-hop network environment.

Hansson, Sjodin and Tindell [HANSSON] proposed a method for providing hard real-time guarantees for traffic through an ATM network. They introduced a scheme that always delivers all messages completely and within specified deadlines. By using priority queues in the output buffers, they allow urgent messages to have short end-to-end delays, while still guaranteeing end-to-end delays for low-priority messages. In their scheme, Fixed Priority Scheduling was used to schedule the departure of cells and determined if message deadlines would be met by calculating maximum end-to-end delays and buffer-needs. They assumed that the single switch and the source nodes send cells in priority order and that cells with the same priority are served in FIFO order on ATM network. However, we did not specify certain scheduling policies in a source node. Also, our multiplexing scheme works on a real-time channel which is given a fixed channel bandwidth a priori.

## 6. Conclusion

Multiplexing is an efficient mechanism to improve channel utilization in high-speed networks. However, network switch multiplexing is very expensive since it is hard to recognize the characteristics of each message traffic due to their high speed. Also, a network switch must have a cell traffic control mechanism when a channel suffers from burstness. On the other hand, in a real-time channel which uses the fixed channel bandwidth it is hard to decide the optimal channel bandwidth level. Nevertheless, it promises to support the different QoS requirements as well as to guarantee end-to-end delay for temporal messages such as real-time traffic.

In this paper, we investigated how to improve real-time channel utilization and to support different QoS requirements in distributed multimedia applications. To do so, we classified incoming messages into four categories at source i.e., real-time, multimedia, aperiodic, and non-real-time messages and their corresponding sub-channels, to incorporate different QoS requirements. The classified message cells are basically scheduled by fixed and dynamic priority scheduling. Our simulation study shows that the proposed EDD+ algorithm which selectively discards the transmitting messages in advance considerably reduces the missing ratio of real-time traffic within a given channel.

We did multiplexing when the utilization of a real-time channel is less than one. However, in the future, we will continue our study for the case where channel utilization is greater than one, which is closer to real-world phenomena.

## References


[ANDERSON] D. P. Anderson, S.Tzou, R. Wahbe, R. Govindan, and M. Andrews, "Support for Continuous Media in the DASH System", In Proceedings of the 10th ICDCS (International Conference on Distributed Computing Systems), May, 1990, pp. 54-61.

[BURN] A. Burn, "Scheduling Hard-Real-Time Systems: A Review", Software Engineering Journal, Vol. 6., No.3, pp. 116-128, 1991

[BANERJEA] A. Banerjea, D. Ferrari, B. Mah, M. Moran, C. Verma, and H. Zhanga, "The Tenet Real-Time Protocol Suite: Design, Implementation, and Experiences", IEEE/ACM Transactions on Networking, Vol. 4. No. 1. Feb. 1996, pp.1-10

[BRADEN] R. Braden, L. Zhang, S. Berson, S. Herzog, and S. Jamin, "Resource ReSerVation Protocol (RSVP) Version1 Functional Specification", Version ST2+, August 1995, RFC 1819.

[CHETTO] H. Chetto and M. Chetto, "Some Results of the Earliest Deadline Scheduling Algorithm", IEEE Transactions on Software Engineering, Vol. 15, No. 10, pp. 466-473, 1989.

[CHOU] Chin-Che Chou and Kang G. Shin, "Multiplexing Statistical real-Time Channels in a Multi-Access Network", IEEE Proceedings of the 16th ICDCS, 1996, pp. 133-140.

[FAN] R. Fan, H. Suzuki, K. Yamada, and N. Matsuura, Expandable ATOM Switch Architecture (XATOM) for ATM LANs, white paper, URL http://www.cisco.com/warp/public/614/3.html.

[FERRARI] D. Ferrari and D. Verma, "A Scheme for Real-Time Channel Establishment in Wide-Area Networks", IEEE Journal on Selected Areas in Communications, Vol. 8, No.3., April, 1990, pp. 368-379.

[HANSSON] H. Hansson "Guaranteeing Real-Time Traffic Through an ATM Network", Invited Talk, 3rd. International Workshop on Real-Time Computing Systems and Applications, October 1996, Seoul, Korea.

[KARLSSON] G. Karlsson, "Capacity Reservation in ATM Networks", Computer Communications, Special Issue on Algorithms for ATM, March 1996

[[LEE] J.W. Lee, S.Y. Lee, and H.I Kim, "Scheduling Hard-Aperiodic Tasks in Hybrid Static/Dynamic Priority Systems", ACM SIGPLAN on Languages, Compilers, and Tools for Real-Time Systems, pp. 7-19, La Jolla, CA, June, 1995.

[LEHOCZKY1] J.P. Lehoczky and S. R. Thuel, Scheduling Periodic and Aperiodic Tasks using the Slack Stealing Algorithm (Chapter 8), Advances in Real-Time Systems, (ed. S. Son) Prentice Hall, Englewood Cliffs, NJ, 1995.

[LEHOCZKY2] J.P. Lehoczky and S. Ramos-Thuel, "An Optimal Algorithm for Scheduling Soft-Aperiodic Tasks in Fixed-Priority Preemptive Systems", Proceedings of the IEEE Real-Time Systems Symposium, pp. 110-123, December 1992.

[LIU] C.L. Liu and J.W. Layland, "Scheduling Algorithms for Multi-Programming in a Hard Real-Time Environments", Journal of the Association for Computing Machinery, Vol. 20, No.1, pp. 46-61, January 1973.

[MERCER] C. Mercer, S. Savage, H. Tokuda, "Processor Capacity Reserves: Operating System Support for Multimedia Applications", In Proceedings of the ACM Multimedia, 1994, pp. 90-98.

[PRYCKER] Martin De Prycker, Asynchronous Transfer Mode: Solution for Broadband ISDN, Second Edition, Ellis Horwood, 1993

[ROSBERG] Z. Rosberg, "Cell Multiplexing in ATM Networks, "IEEE/ACM Transactions on Networking, Vol. 4. No. 1. Feb. 1996, pp.112-122

[ZHANG] L.Zhang, S.Deering, D.Estrin, S.Shenker, and D. Zappala, "RSVP: A New Resource ReSerVation Protocol", IEEE Network Magazine, September 1993