pp. 83 – 90, Communication Networks and Services Research, IEEE 2004.

[12] Anderw S. Tanenbaum, "Computer networks", Prentice-Hall, Inc. 1996.

[13] S. Minezewa, and Y.Ishibashi, "Quality assessment of media synchronization of preventive control schemes in video and voice communications," in Proc. SPIE Optics East, Multimedia Systems and Applications IX, vol. 6391, no. 6, Oct. 2006.

Appendix A

Queuing delay for priority and non-priority classes.

(a)- Delay for non-priority traffic:

$$D_i = 1/2(\lambda\sigma^2)/(1-\lambda\sigma)$$

(b)- Delay for high-priority part of traffic:

$$D^h_i = 1/2(\lambda\sigma^2)/(1-\lambda_h\sigma)$$

(c)- Delay for low-priority part of traffic:

$$D^l_i = 1/2(\lambda\sigma^2)/(1-\lambda_h\sigma)(1-\lambda\sigma)$$

where D_i is queuing delay in link i, D^h_i is delay for high priority traffic, and D^l_i is delay for low priority traffic. Also $\sigma = b/v$ is service time, b is packet size, v is data rate, and $\lambda = \lambda_l + \lambda_h$.

multimedia environments, sum of normalization path delays is constant, and based on this property we can minimize the playtime interval for delay-sensitive media. It is shown that, in transmission media the multi-class queuing is a possible solution, where in the receiver side, an adaptive playtime scheduling is an important issue. Furthermore, the queuing discipline in both criteria is an important factor. We have proposed an optimization procedure for transmission and four algorithms for adaptive playtime scheduling in the receiving buffer. The successful algorithm has been evaluated through simulations.

8- References:

[1] M.R.Ahmadi, S. Yoshinori, "Priority Assignment for Convergence Type Multimedia Communication", Transaction of IEICE, Vol. J81-B-I, No.4, pp.235-242, April., 1998.

[2] M. Charalambides, P. Flegkas, G. Pavlou, and A. Bandara, "Dynamic Policy Analysis and Conflict Resolution for DiffServ Quality of Service Management", Proc. of the IEEE/IFIP Network Operations and Management Symposium (NOMS'2006), Vancouver, Canada, IEEE, April 2006.

[3] S. Minezewa, Y.Ishibashi, and T. Asano, "An experiment on preventive control for media synchronization of video and voice in a distributed haptic museum", (in Japanese) in Proc. IEICE General Conference, B-11-4, Mar. 2006.

[4] K. E. Psannis and Y. Ishibashi, "Efficient flexible macroblock ordering technique,"

IEICE Trans. Commun., vol. E91-B, no. 8, pp. 2692-2701, Aug. 2008.

[5] S. Georgoulas, and P. Trimintzios, " Measurement-based Admission Control for Real-time Traffic in IP Differentiated Services Networks", Proceedings of IEEE International Conference on Telecommunications, Capetown, South Africa, IEEE, May 2005.

[6] Sue B. Moon, "Measurement And Analysis Of End-To-End Delay And Loss In The Internet", 2000.

[7] Fernando Boronat, Jaime Lloret, Miguel García, "Multimedia group and inter-stream synchronization techniques: A comparative study", Information Systems, v.34 n.1, p.108-131, March, 2009.

[8] L. Kleinrock, "Queuing Systems," Jone Wiley, vol 1 and 2, 1975.

[9] M. Garey and D. Johnson, "Computer and Intractability," A guide to theory of NP-Completeness. New York: W.H. Freeman and Co.,1979.

[10] Jiang Li, "End-to-End Multicast Congestion Control and Avoidance", Ph.D. Dissertation, August 2003.

[11] R. McGorman, J. Almhana, and V. Choulakian, "Similarities between voice and high speed Internet traffic provisioning", Proceedings. Second Annual Conference on 19-21 May 2004,

Transport protocol is UDP

We consider four algorithms for playtime assignment as follows (in algorithms (*a-d*) the mean delay value is calculated during message (*i*-1), and α =0.998002):

$$A \lg orithm \quad (a): Min \ (\hat{d}_{i-1}) \qquad (20)$$

$$A \lg orithm (b): \alpha \times \hat{d}_{i-1} + (1-\alpha) \hat{var}_{i-1} \quad (21)$$

$$A \lg orithm \quad (c): \alpha \times \hat{d}_{i-1} + \hat{var}_{i-1} \qquad (22)$$

$$A \lg orithm (d): \alpha \times \hat{d}_{i-1} + 1.2 \times \hat{var}_{i-1} \qquad (23)$$

In simulations, packets are generated in the source node, experience network delay and sequentially enter to the receiver buffer. In receiver buffer the initial playtime is calculated based on the following equation:

$$Playtime \quad (1) = \alpha \times \hat{d}_1 + 4 \times \hat{var}_1 \qquad (24)$$

The next coming playtimes are based on the proposed algorithms. The packet loss is calculated for each algorithm and compared with other algorithms as shown in Figure 8. Figure8 shows that algorithm (*d*) has the minimum packet loss rate among the others; because in this algorithm the weighing function related to variance has the maximum effect. As a result, the variance is a key factor for packet loss control where it should be limited; otherwise, it will experience long transmission delay.
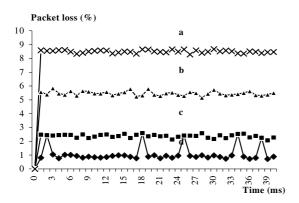

Fig. 8 Comparison of different adaptive playtime algorithms

On the other hand, to show the effect of proposed algorithms on data recovery in burst traffics, which cause sudden increase in the incoming traffic, we have prepared another simulation program. The network and the buffer prototype are the same as the previous model. Generally, a burst in the traffic is equal to a high level of packet loss or large delays in the receiver. Once this condition oceures, the algorithms follow the new behavior instantly, and adjust the initial value by receiving the first packet to calculate the playtime for the next incoming packets. In practice, the detection of bursty traffic is not so difficult and we declare a situation as being bursty when the delay between two consecutive packets goes beyong a certain threshold level. This situation depends on the network behavior, and may change quickly or stay unchanged for a long period of time. The main point is that how an algorithm can follow the network behavior instantly and define a suitable playtime for each packet. As it is shown in Figure 9, algorithms *a,b,c*, and *d* has same packet loss rate, where the recovery time in a burst traffic in algorithm *d* is around 0.05 ms. As a result, algorithm d not only minimizes the packet loss rate, but also it is successful in bursty traffic.
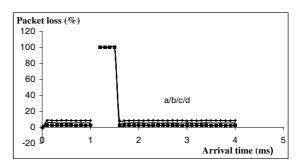

Fig. 9 Data recovery for bursty traffic

7- Conclusions

This paper considers delay problem in multimedia environments. We analyzed playtime delay for each source-destination delivery and considered the network environments and receiver buffer as the two optimization criteria. We proved that in a

scenario, we consider the process in figure 7. In this figure two consecutive audio messages are generated in the source node, and experience a non-uniform network delay. As a result, they may arrive in The receiving buffer simultaneously, out of sequence or with different delays.
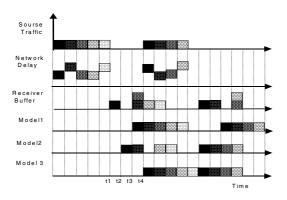

Fig. 7 Example adaptive playtime model

In Fig.7 the top graph shows a sequence of source traffic. The second graph shows the effect of non-uniform network delay experienced by each packet. In the third graph, the arrival time of the packets is compared. Here, we consider three different playtime algorithms and compare the effect of those algorithms on packet loss together with the message interval time. In model 1, the playtime is at t_3 with enough waiting time. Here, all the packets are ready at the playtime point with three unit message interval time. In model 2, the playtime is at t_2, in this case one packet is lost and the message interval time is two units. Finally, in model 3, the playtime is two intervals unit later than t_2, in this case all the packets are available; but, the message interval time is omitted. As a result, based on playtime scheduling, we can get all the packets successfully, we may loss some part of the messages, or we may loss message interval synchronization. Since, a mechanism with fixed playtime scheduling can not follow the network behavior; it is not a good solution for playtime assignment. For this problem, we consider a combined set of network parameters as an estimation value for the playtime scheduling. It should be noted that the estimation value can

determine the playtime point for the first packet in each message, and rest of the packets can be obtained by adding an offset to the first estimation. We have considered different algorithms that have differences in the way they combine parameters and calculate the departure time. These parameters are, average and variance of delay with suitable weighting factors. In these algorithms, delay for the ith packet and a measure for the variance are calculated based on the following equation [9]:

$$\hat{d}_i = \alpha \times \hat{d}_{i-1} + (1 - \alpha) \times r_i \qquad (18)$$

$$\hat{v}_i = \alpha \hat{v}_{i-1} + (1 - \alpha) \left| \hat{d}_i - r_1 \right| \qquad (19)$$

where, $\hat{d}_i$ and $\hat{v}_i$ are two estimations for mean and variance of the point-to-point delay during the message, and r refers to the arrival time of a packet in the destination host. Figure 3 shows more details for these assumptions. In our calculation, we consider values of 0.998002 and 0.78 for α [10]-[13].

6- Evaluation of Adaptive Scheduling in the Simulation Program

In this section we evaluate the effect of adaptive playtime scheduling on packet loss rate. In this simulation, we apply an adaptive mechanism for playtime scheduling in the receiver buffer and compare the effect of different parameters on packet loss. It should be noted that loss rate is a result of either late arrival or extremely premature arrival of packets. In the latter case, the length of the receiver buffer is an important factor. If i- the packet is currently departed and the buffer has room for m packets, any packet arriving with length $m+i$ or greater will be discarded; having arrived too far in advance to be buffered. Here, the simulation environments has the following specifications:

Length of receiver buffer: 10,000 bytes
Packet rate=1000~9000 packets/$_s$
Network delay= A random value with exponential distribution
Evaluation factor= packet loss rate, message length=1000 bytes

will tolerate the service time delay based on the service discipline.

t_w = waiting for scheduled playtime (queue time) + service time.

Service time = depart system - depart queue.

Since our problem in the network side is variance of the delay, as a complementary process for this optimization, we consider the FCFS (first-come, first-service) discipline for all the queues. We have proved that among other disciplines, the FCFS can minimize variance of the delay.

Theorem 2: In receiver buffer, the FCFS discipline minimizes waiting time variance when queuing discipline is service time independent.

Proof: Assume that expected waiting time is the same for all queue disciplines; the variance is minimized when the expectation of the squared waiting time is minimized. We consider the following definitions:

t_n = Arrival time of n- th packet.

σ_n = Service time of n- th packet.

$\sum W_q'^2$ = Sum of the squared waiting time in queue with non-FCFC discipline.

$\sum W_q^2$ = Sum of the squared waiting time in queue with FCFC discipline.

We consider the change in squared time when two packets are served in a sequence other than FCFS. In this case, at time T, suppose the server is servicing either packet n or packet $n+1$. To calculate the difference in squared time in the queue we have:

$$\sum W_q'^2 - \sum W_q^2 = (T - t_2)^2 + (T - t_1 + \sigma_2)^2 -$$
$$[(T - t_1)^2 + (T - t_2 + \sigma_1)^2] =$$
$$2(T - t_1)\sigma_2 + \sigma_2^2 - [2(T - t_1)\sigma_1 + \sigma_1^2] \qquad (14)$$

Since both σ_1 and σ_2 have identical distribution, $E(\sigma_1) = E(\sigma_2)$, and $E(\sigma_1^2) = E(\sigma_2^2)$. Thus, we have:

$$E[\sum W_q'^2 - \sum W_q^2] = 2(t_2 - t_1)E(\sigma) > 0 \quad (15)$$

Since $t_2 > t_1$ by definition, equation (15) must be greater than zero. As a result, processing the packets in any order other than FCFS increases the waiting-time variance.■

5- Adaptive Playtime Analysis

One approach to deal with the unknown nature of delay distribution and synchronizing the generaion and departure of real time messages (i.e. voice or video messages), is to estimate the delay and adaptively adjust playtime scheduling. Practically, to compensate for the queuing delay in the network, we buffer packets in the receiver side. Since the queuing delay is not unifor, we need an estimation mechanism to adaptively determine the network delay and select the playtime according to the network behavior. Here to deal with this problem, we adjust the playtime adaptively according to its variation using dynamic delay estimation. For our estimation, we separate the delay into two parts. One is playtime delay for the first packet (i.e. a talkspurt in an audio message) that is based on the following formula:

$$Playtime \quad (1) = t_1 + \hat{d}_i + n \times \hat{var}_i \qquad (16)$$

And the second for the rest of the packets in the message that is:

$$Playtime \ (j) = Playtime \ (1) + t_j - t_1 \qquad (17)$$

Where, $\hat{d}_i$ and $\hat{v}_i$ are estimation values for the mean and variance of the point-to-point delay during the message, and t refers to the time. The playtime for any subsequent packets in a message is considered as an offset value from the time where the first packet in the message was played out. The term of, $n * \hat{v}$, is used to set the playtime far enough beyond the estimated delay so that a small fraction of packets be rejected because of long delay. For this synchronization we need an adaptive playtime mechanism which has low loss rate and low playtime delay. For more details, we compare the effect of playtime interval on packet loss in receiver buffer for UDP audio packets in a public network. In this

Where p is the normalized path delay, $\bar{p}$ is average of normalized delay, and N is the number of paths. It should be noted that the problem of priority assignment which minimizes one parameter of delay (i.e. variance) is NP-hard [9].

4-1- Evaluation of Delay in the Network Environments

In a given network, we consider a backbone sub-network that covers all the nodes with minimum propagation delay. Then, we assign a sensitivity factor for each media, and define two classes of priority for each queue. Here we define P_i as the normalized value for the delay in path i, and P_{ik} as the priority of P_i in link k. As a result, we have a set of virtual paths in the network which arrange our communication model. In addition, each link in a path can support traffic with two priority classes. The evaluation parameter is path delay and we minimize the variance of delay. In order to show the effect of priority assignment we have prepared a simulation environments with the following parameters

Message length=1000 bytes. Link capacity = 1~10 Mbps. Packet rate=1000~9000 packets/s. Evaluation factor = Normalized path delay. Optimization factor = Variance of the path delay. Variable = Priority of traffic in each link.

Figure 5 shows the network model that we have used as a prototype with 10 source nodes and one destination address. This prototype is selected for simplicity and it can be applied to any arbitrary topology. All the multi-point nodes send the traffic to a destination node through different virtual paths.
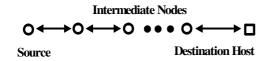
Intermediate Nodes

Fig. 5 Prototype network in the simulation environments

For each packet rate, we generate different random patterns related to priority of the traffic in each link. This pattern covers all the

combinations of link-traffic in all the virtual paths. The procedure tries to select the best pattern that can minimize the value of variance for all the network traffic. Figure 6 shows the result of simulation and compares the improvement rate for maximum path delay. Note that, since we have normalized all the path-delay based on its sensitivity factor, minimizing the variance will decrease the delay in sensitive media where as increase the delay in the low sensitivity part. This concept is based on the tradeoff between decreasing the delay in sensitive media with the expense of increasing delay in the low sensitivity media. Since these values never pass the average value, this mechanism is an effective method for priority assignment.
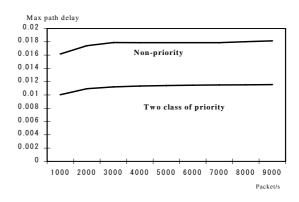

Fig. 6 Effect of priority assignment on maximum path delay

As a result, this heuristic algorithm is successful for minimizing the variance of the normalized path delay in all the virtual paths. The algorithm minimizes the variance by minimizing path delay in the time-sensitive media where increasing the delay in low sensitiving traffic and the average value determines the optimum solution.

4-2- Evaluation of Delay in Receiving Host

The receiving host has two impacts on incoming traffic. First, in order to compensate for the variable network delay, we buffer packets and assign proper playtime delayed for playout so that most of the packets will be received before their scheduled playtime. Second, each packet

Link n: $\sum_{j=1}^{l} \sigma_n \lambda_j T_n^{\ j} = c_n$ $\qquad$ (5)

If we assume one sample path with l links we have,

$$\sum_{j=1}^{l} \sigma_j \lambda_1 T_1^{\ j} + \sum_{j=1}^{l} \sigma_j \lambda_2 T_2^{\ j} + ... + \sum_{j=1}^{l} \sigma_j \lambda_k T_k^{\ j} = cte \quad (6)$$

where $T_i^{\ j}$ is delay of path j in link i, λ_j is traffic in path j, σ_i is service time, and $c_1.....c_n$ are constant values. Note that, if a virtual path does not traverse trough a link, delay of that path in the link is zero. If we calculate total delay in all the paths in media i, and assume $\lambda_j = \lambda$:

$$t_1 + t_2 + ... + t_{n_i} = n_i \times T = cte \qquad (7)$$

where t_j is delay of path j in media i. Also for average delay in m media we have:

$$n_1 T_1 + n_2 T_2 + ...+ n_m T_m = m \times \overline{T} = cte \qquad (8)$$

where T_j is average path delay in media i, and $\overline{T}$ is average total path delay in m media. On the other hand, in media k with n_k nodes and sensitivity factor s_k, normalization of path delay is done as follows:

$$p_j = t_j \times (s_k / S) \qquad (9)$$

where $S = 1/m \sum_{k=1}^{m} s_k$ is the average value of time-sensitivity in m media, and s_k is time-sensitivity in media k. If we consider average normalization of delay in media k with n_k nodes, we have:

$$p_1 + p_2 + ... + p_{n_k} = \sum_{j=1}^{n_k} p_j / n_k \qquad (10)$$

Also if we consider the normalized path delay in m media, based on Eq. 7 and 8, we have:

$$P_1 + P_2 + ...+ P_m = 1/m \sum_{i=1}^{m} \sum_{j=1}^{n_k} p_j / n_i = cte \qquad (11)$$

∎

As a result, in m media environments, sum of the normalized path delay for all the virtual paths is constant. Thus, by using a suitable priority assignment, we can control the delay in highey time-sensitive traffic with the expense of low sensitive media where the total delay is invariant. It should be noted that we consider time sensitivity as an important factor in the optimization procedure.

4 - Network Parameters and the Optimization Function

In order to control the queuing delay, we consider two classes of priority: low and high priority queue (we can extend the problem to more classes). For each packet, there are two possibilities: being serviced through the low or high priority level. This decision is made based on the path delay and the link traffic. We consider normalized path delay as the evaluation parameter:

$$p_i = t_i \times s_i / S \qquad (12)$$

where $t_i = f(c_1,....c_k, p_i^1,....p_i^k)$, c_i is link capacity and p_i^k is the priority of path i in link k. Here the problem is to determine the priority for traffic in each queue, where it can minimize variance of the normalized path-delay. This method has the advantage that we can decrease delay in the set of paths with maximum delay value while increasing delay in the set of paths with minimum delay where based on Conservation Law the total delay is invariant. For this problem, we propose a heuristic optimization procedure that can solve the problem even though for the set of paths with equal delay values (i.e. loop networks or set of sources with equal distances from the destination). The procedure minimizes the following equation:

$$Var_{min} = Min \sum_{i=1}^{N} \left[(p_i - \overline{p_i})^2 / N \right] \qquad (13)$$

allowable delays), put them in their respected queues, and based on its queuing discipline, forward them toward the destination. In order to decrease queuing delay, priority queuing is a possible solution, especially for time sensitive media. Since in a multimedia environments each media can tolerate certain level of delay, which is different from other media, minimizing the delay in sensitive media with the expense of low sensitive media is a challenging issue. Here, for simplicity, we assume each node as M/D/1 queuing system. In this case, for a non-priority and priority classes, we can calculate the queuing delay based on Appendix A. With this assumption, according to priority of the traffic in each link, we can change the delay in each path (i.e. the longest path delay or delay sensitive media.) where based on Conservation Law the total delay is invariant.

3-1- Delay Normalization and Media Sensitivity
In order to minimize the maximum path delay in a multimedia environments, we introduce a sensitivity factor for each media according to characteristic of a media. Sensitivity factor normalizes the function of delay in one media to a uniform value among the other media. Indeed, we assume one closed bound between zero and one [0,1], where the maximum value specifies the media with the maximum sensitivity or minimum allowable delay, and the minimum sensitivity factor for non-sensitive media (i.e. non real-time applications). To make this definition clear, in two media applications, let the allowable delay in media a and b be T_a and T_b respectively where $T_b > T_a$. It gives the sensitivity value of s_a and s_b to media a and b where $s_a > s_b$ (media with less allowable delay has more sensitivity).

Normalized delay for media a = a function of path delay $\times s_a / (s_a + s_b)$.

Normalized delay for media b = a function of path delay $\times s_b / (s_a + s_b)$.

In this case different characteristic of each media comes to the account and we have a uniform value with the same priority in the optimization procedure. To use this factor for priority assignment, we need to prove the following theorem.

Theorem 1: In m media environments, sum of the normalize path delay for all the virtual paths is constant, i.e. :

$$1/m\sum_{i=1}^{m}\ \sum_{j=1}^{n_k} P_j\ /\ n_i = cte \qquad (2)$$

where P_j is the normalize path delay in media i, n_i is number of nodes in media i, and m is number of media.

Proof: In a multimedia environments with m media, we assume n individual virtual paths from a set of source nodes to a destination address. Assume m different media such that media i contains n_i source nodes where $\sum_{i=1}^{m} n_i = n$. As it is shown in figure 4.
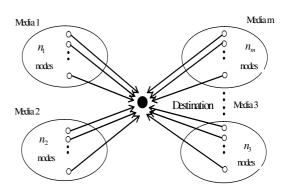

Fig.4 m media communication model

Based on Conservation Law in a given network traffic, the weighted sum of all the stream delays is constant. To calculate the delay in each link we have:

Link 1: $\sum_{j=1}^{l} \sigma_1 \lambda_j T_1^{\ j}\ =\ c_1$ \qquad (3)

Link 2: $\sum_{j=1}^{l} \sigma_2 \lambda_j T_2^{\ j}\ =\ c_2$ \qquad (4)
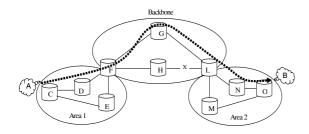
Fig. 2 Multimedia transmission with store and forward mechanism

In this process any packet experiences propagation delay in each link and processing plus queuing delay in each node. Those items may cause the packets to wait, put them out of sequence, or may cause them to be dropped. Fig. 3 Shows playtime interval for one point-to-point delivery.
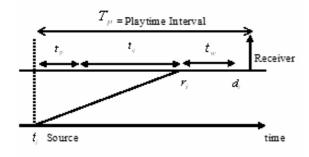

Fig.3 Playtime of a point-to-point delivery

The main parameters in transmission media are propagation, processing, and queuing delay where in the receiver buffer; the problem is waiting time for playtime schedule. Here, we separate the playtime period into three main parts. Propagation delay that is assumed to be constant. Queuing delay which depends on service discipline, buffer size as well as traffic load. Waiting time delay in the receiver buffer which depends on playtime schedule and service discipline.

Playtime delay = Propagation delay + Queuing delay + Waiting-time delay.

t_i is the generating time of the packet x.

r_i is the arrival time of packet x in the destination host.

d_i is the played-out of the packet x at the destination host.

t_p is the propagation delay from source to the destination for packet x and is assumed to be constant.

t_q is the queuing delay from the source to the destination host for packet x.

t_w is the waiting time of packet x in the receiver side for its playtime schedule.

T_p is the playtime delay that refers to amount of time from source until play out in destination for packet x.

To minimize the playtime interval, we can employ different strategies. In one hand, we can apply a suitable queuing mechanism to decrease the network delay; on the other hand, we can control waiting time in the receiving buffer.

To analyze this problem, we model each node as M/D/1 queuing system [8]. It should be noted that for compatibility of the model we consider a multi-point to point communication as N individual connections between N points to one point. N is the number of source nodes with one destination. In this case, the value of each path delay can be obtained using the following formula

$$T_p = \sum_{i=1}^{k} t_{pi} + 1/2 \sum_{i=1}^{k} \lambda_i \sigma_i /(1 - \lambda_i \sigma_i) + t_w \qquad (1)$$

where $\sigma = b/v$ is service time, b is packet size, v is data rate, t_{pi} is propagation delay in link i, k is the number of links in path p, and t_w is waiting time in receiver buffer. We can analyze the path delay in two separate parts: delay in the network environments and waiting time in the receiving host.

3- Delay in the Network Environment

Major factors making quality degradation in the network environments are propagation and queuing delay. Each intermediate node, receives the packets from different media (with different

any resources such as bandwidth or performance measures (i.e. maximum delay or maximum loss rate) [5]. In this scenario, the challenge for supporting the real-time data over the network is the need to provide a synchronous playtime of packets where we have a stochastic end-to-end network delay [6,7]. Packets experience transmission and queuing delay between the source and the destination, where in the receive side, they may wait for their scheduled playtime. In order to control the delay from one source to a destination address in each media, we divide the delay to two parts. One includes the propagation, processing, and queuing delay in the network side, where the another part refers to waiting time for the scheduled playtime in the receiving host. In this paper we investigate the problem of packet delay during the playtime from a source to a destination, and evaluate the performance of priority assignment as well as queuing discipline in network side together with adaptive playtime scheduling in the receive buffers.

This paper is organized as follows: Section 2 gives a brief introduction to some principal concepts in real time transmission. Section 3 analyzes network delay and sensitivity factor. Section 4 proposes an optimization technique for queuing delay in network environments and receiving host. Section 5 introduces an adaptive playtime analysis. In section 6, evaluation of adaptive scheduling is given. And finally, we conclude the paper in section 7.

2- Delay Analysis in Packet Networks

In a multimedia environments, we can consider two qualities for services. One is the network quality of service that refers to bandwidth, end-to-end delay, inter/intra stream synchronization, delivery order of data, error recovery, packet loss, jitter, echo, and so on. The effect of these parameters on quality of service depends on media and application models. For example, in a VoIP application, the round-trip delay, jitter, packet loss, and delivery order of data are critical issues where as in video conferencing, in addition to those items, the quality of video signals is also an

essential factor. On the contrary, in text transmission, packet loss is a major concern. The second quality of service refers to the requirements for perception of multimedia information at the user interface. Here, we discuss some characteristics and behavior of data transmission in store and forward packet switching environments. Figure.1. shows the process of real time packet transmission over Wide Area Networks. The packets are generated in the source node and experience a random delay in the network. In order to smooth out such delay jitter, the receiving host can delay the initiation of periodic played-out of received packets for some time interval.
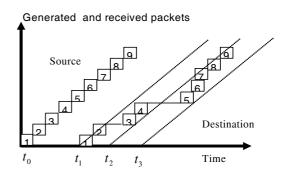

Fig.1 Process of packet transmission in public networks

In Fig. 1, if the receiver delay is in the beginning of playtime t_3, all the packets will have been received before the schedule. On the other hand, if the play out began at t_2, delay is less, but the packets (5-9) will loss having arrived after their scheduled playtime. This illustrates the tradeoff between the delay and packet loss that is a critical issue in transmission media. To explain this scenario with more details, we consider the process of data transmission from an arbitrary source node to a destination as is shown in Fig.2. In this figure data from the source node arrives in the first intermediate node and based on its routing table will be forwarded to the next intermediate node until it can reach the destination host.

‹‹Research Note››

Delay Control Using Media Sensitivity in Multimedia Environments

Mohammad Reza Ahmadi*

Assistant Professor, of Iran Telecom Research Center

*P. O. Box: 14155-3961
m.ahmadi@itrc.ac.ir

Abstract- Increasing real time services in multimedia environments has initiated a new phenomenon in data communication. This paper analyzes delay in multimedia environments focusing on multi-point to point communication [1]. We separate the playtime delay in a point-to-point transmission and propose an optimization scenario for each part. It is proved that sum of the normalized path delay for point-to-point connections is invariant and based on this property, the playtime for delay-sensitive media has been minimized. We have shown that in transmission media, priority queuing is an effective solution where in the receiver side, waiting time for playtime scheduling and queuing discipline are two main factors. It is shown that there is a compromise between packet loss and packet departure time in the receiver side where the acceptable packet loss can adjust the playtime delay adaptively. Theoretical analysis for priority assignment, queuing technique and performance evaluation in different classes of queuing with different playtime scheduling are given.

Key Words: Multimedia Services, Network Delay, Playtime Scheduling.

1- Introduction

Progress of new technologies and services in data communications, creates an opportunity for using the continuous media (e.g. video and audio) in the existing packet networks. In attemt to support the real-time applications (e.g. video conferencing, VoIP service, real-time fax services, and so on) over the public networks which are not designed for this purpose, the services are suffering from quality degradation which results from limitations of available network resources. One of the important characteristics in multimedia environments is supporting the variety of services with different requirements. Some media need to get real-time services, where some may not be so time sensitive. Moreover, based on different behavior of network characteristics in different geographical points, different traffic may experience different network delays. These characteristics give us the opportunity to manage a reasonable tradeoff between different media and network resources. As a good candidate, we can apply a priority technique for time sensitive media with the expense of media with low time sensitivity. There are extensive literature for classification of traffic and using different techniques based on service level agreement (SLA) between the users and network providers [2,3,4]. The users follow the agreement level and never concern about characteristics of different services to optimize the network parameters and resources. In this research we address some of the main problems which are involved in obtaining the quasi optimal solution in multi-point to point data communications. Practically, in multimedia applications the public network does not guarantee