

# Petri-Net Model of Multimedia Synchronization in Mobile Environment

Keun-Wang Lee<sup>1</sup>, Jong-Hee Lee<sup>2</sup>, and Hae-Seok Oh<sup>2</sup>

<sup>1</sup> Dept. of Multimedia Science, Chungwoon Univ., Korea  
kwlee@cwunet.ac.kr

<sup>2</sup> School of Computing Soongsil Univ., Korea  
multistar@freechal.com  
oh@computing.soongsil.ac.kr

**Abstract.** As demand for the application in multimedia networks is increasing rapidly, it is important to provide multimedia services in Mobile Environments (ME). Obtaining to multimedia services which satisfy synchronization constraints in ME and improving the delay time and Quality of Service(QoS) between media streams to be presented, new specification model has to be proposed. In this paper we propose Mobile Petri-net Model(MPM) as a new specification model including the QoS parameter that minimizes the transmission delay time. The proposed model has higher the guarantee of QoS such as the loss rate and the playout rate than it of the previous work.

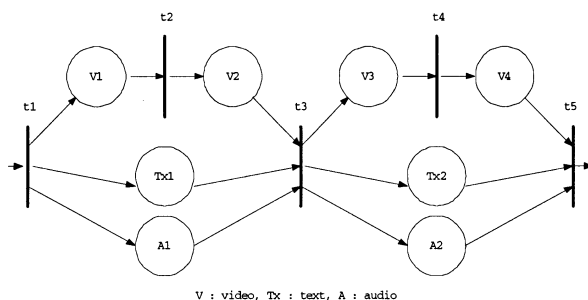
## 1 Introduction

The interest in multimedia services coupled with a growing high-bandwidth structure has been increased in the area of wired communications. Recently, with the advancement in wireless communication systems and portable computing technologies, demand for mobile multimedia services is increasing in the area of wireless communications, such as data transmission, video transmission, www access, and so on[1][2][3]. Mobile multimedia service is complex concept of wireless extension of wired multimedia service and mobile specific multimedia service. That is, mobile multimedia service is to add user mobility to multimedia services in wired networks[4]. The synchronization in wired networks has been continuously studied, while scarcely in wireless networks. In wireless networks, a Mobile Host (MH) has small memory, and a Base Station (BS) supports the limited resources than the sites in wired networks. Moreover, the mobile communication system has low power, and must provide high-quality access with MHs. Consequently, the synchronization in wireless networks is complex than that in wired networks. In particular, because of the small memory and the limited resources, a MH may suffer from either the buffer underflow or the buffer overflow. Therefore, an adaptive synchronization method is needed for mobile multimedia networks. The previous works have been done in describing the synchronization model for multimedia applications[5],[6][7][8][9][10]. Among those, Petri-net

based specification models provide a good method to specify temporal relationships. The integration of various media and the definition of QoS requirements should be supported and easily described. However, previous extended Petri-net based models like as Object Composition Petri-Net (OCPN) and Real Time Synchronization Model (RTSM) have some constraints in modeling the QoS parameters of both intramedia and intermedia. So, we propose MPM model which guarantees the QoS in ME.

## 2 Related Work

At the time of creation of multimedia information, a user needs a model to specify temporal constraints among various data which must be observed at the time of playout[3]. Such specification models are OCPN and RTSM which are a timed Petri-net. OCPN has been proved to have the ability to specify any arbitrary temporal relationship among media and it may be applied to both stored data applications and live applications[3][5]. The OCPN model weakly describes synchronous temporal relationships between media at an object level. A multimedia application which has video, audio and text object can be expressed by OCPN modeling technique as in Figure 1. Generally, audio data is very jitter sensitive and cannot tolerate random delays between frame segments; hence, delay between audio frames may result in unrecognizable audio quality. In contrast, some extent of random delay between video frames may be acceptable in video data. Therefore a late transmission of video frames does not influence the QoS.



**Fig. 1.** An example OCPN model.

When taking the real-time issue of multimedia and the random delay of packet/cell networks into consideration, OCPN and other Petri-net based models are not sufficient to deal with the late transmission of packets. Here, we define late transmission of a packet to be that the packet fails to reach its destination in time and should be dropped. We explain the insufficiency of OCPN by the example shown in Figure 1. A delay of video object in OCPN will give an impact on playing out an audio object. Such a delay of audio object in a real-time

application will cause a significant degradation of its QoS. RTSM is introduced to guarantee the QoS of real-time applications on the delay of intermedia. A jitter-sensitive audio object is defined as key medium and represented as enforced place [6]. However it is possible that a real-time constraint is exceeded due to the delay of key medium itself. If the text object is an one of the key medium in Figure 1, it is assigned as an enforced place just as an audio object. Once one of enforced place is unblocked, each transition of RTSM is fired. When an audio object is unblocked, the transition is fired, even if text object is still activated. Therefore the text object is not guaranteed to display its content. When the transition is fired in RTSM, the remainder part of other medium is dropped. That is, when taking a random delay into account, it is not sufficient to deal with late transmission of packets, and it is also insufficient in modelling the QoS parameters.

3 MPM

The explosive growth of Internet access in parallel with the technological advances has motivated mobile computing and multimedia applications in mobile networks. We assume that Multimedia Servers (MSs) over Internet act as repositories for multimedia applications orchestrated according to the synchronization model, such as OCPN. A MS searches the requested data from its databases and transmits it to the BS. Figure 2 shows the mobile multimedia system structure.

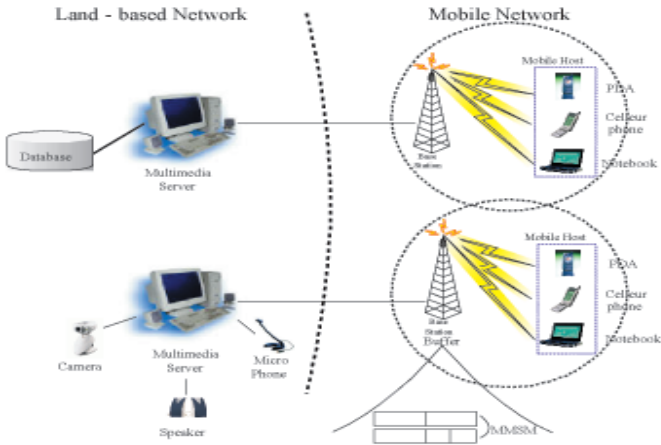


Fig. 2. Mobile multimedia system structure.

Multimedia data consists of many multimedia objects, a unit of synchronization, which are stamped with the current local time to allow the BS to calculate a round trip delay, jitter and inter-arrival time, during the MS transmits multimedia objects. The multimedia objects need to be buffered at the BS for carrying

out synchronization as interface between the Internet and mobile networks. Mobile multimedia network consists of three parts: MS, BS and MH. In addition to the conventional functions of the BS such as frequency resource management, error control and so on, the BSs in the mobile multimedia network have the additional function such as synchronization, rate matching, etc. The MH must access MSs over Internet through the BS. As shown in Figure 2, multimedia data is transmitted from the MS to the BS. After buffering the multimedia data in order to smooth an inter-packet jitter delay which occurs over Internet, the BS transmits the synchronized data to the MH.

### 3.1 Definition of MPM

The definition of MPM to specify Petri-net in any BS is the following. The MPM specified by the tuple  $[P, T, K, A, DIFF, R, D, J, Re, M]$  where

$P = p_1, p_2, \dots, p_n$  ; Regular places(single circles).

$T = t_1, t_2, \dots, t_m$  ; Transitions.

$K = k_1, k_2, \dots, k_i$  ; Key places.

$X = P \cup K$  ; All places.

$A = (X) \cup (T) \rightarrow I, I = 1, 2, 3, \dots$  ; Directed arcs.

$DIFF : X \rightarrow Realnumber$  ; Difference between maximum duration time and actual duration time.

$R : X \rightarrow Realnumber$  ; Relative duration time.

$D : X \rightarrow Realnumber$  ; Maximum duration time of places.

$J : X \rightarrow Realnumber$  ; Maximum delay jitter.

$Re : X \rightarrow r_1, r_2, \dots, r_k$  ; Type of media.

$M : X \rightarrow I, I = 0, 1, 2$  ; State of places.

Each place may be in one of the following states:

0 : no token.

1 : token is blocked Cross in the place.

2 : token is unblocked Dot in the place.

Places are used to represent a medium unit and their corresponding actions. The Place can have tokens. A Place without token means that it is currently inactive. A place with token is active and could be in one of the two states: blocked or unblocked. When a transition is fired and then token is added into a place, the action of the place is executed, and the token is blocked before finishing the action. The token is unblocked after finishing. Each place has some parameter that determines its relative importance compared with other mediums. Two factors should be considered in deciding the time medium with absolute time and the multiple key media parameter. First, the importance of the medium and second, the jitter-sensitivity of the medium.

### 3.2 Firing Rule of MPM

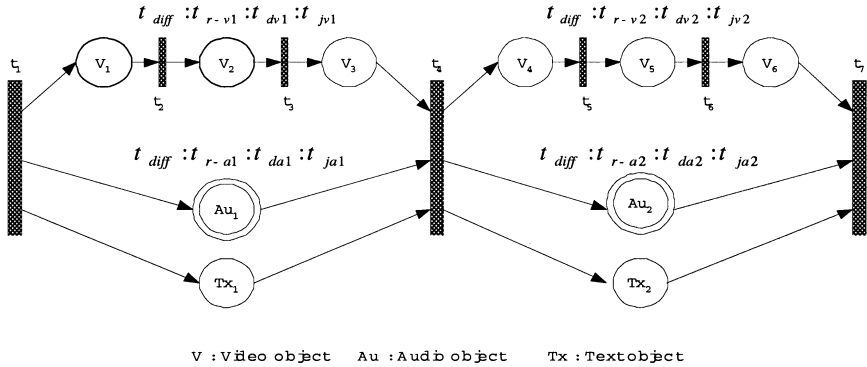
Firing rules of MPM are described as follows:

1) The firing occurs immediately if the end of time medium is done.

- 2) Transition  $t_i$  fires immediately when the arrived key medium places contain an unblocked token.
- 3) Upon firing, a set of backtracking rules is exercised to remove tokens from their input places.
- 4) Transition  $t_i$  removes a token from each of its input places and adds a token to each of its output places.
- 5) After receiving a token, a place  $p_j$  remains in the active state for the interval specified by the duration  $j$ . During this interval, the token is blocked. In MPM, there are two kinds of places as regular places and key places. To differentiate a key place from a regular place, a double circle is drawn for key place as shown in Figure 3 where transition  $t_4$  is fed by two regular places,  $V_3$  and  $Tx_1$  and one key place,  $Au_1$ .

#### 4 MPM Including the QoS Parameter

Each place  $p_i$  has time-related four parameters  $\tau_{diff}$ ,  $\tau_r$ ,  $\tau_d$ , and  $\tau_j$  as well as key medium in order to handle QoS parameters for intermedia synchronization. Figure 3 presents the MPM with QoS parameter.



**Fig. 3.** MPM with the QoS Parameter.

Intermedia skew value 0 means that the two media streaming is completely synchronized. For instance, it can be considered as the synchronized state if it is within 80ms between audio and video streams. If skew value 80ms is applied to key medium of audio, the maximum delay jitter  $\tau_j$  will be destroyed, which will result in degrading the audio quality. Because the maximum delay jitter of audio within media is 10ms, waiting longer than 10ms could be a factor that destroys the audio quality. Therefore,  $\tau_j$  is compensated by means of applying the maximum delay jitter  $\tau_j$ . Maximum delay jitter  $\tau_j$  represents maximum delay jitter of the place  $p_i$ . The delay is a interval between medium  $i$  and medium  $i + 1$ , and it is used to obtain the better quality of service by additionally

playing out as much as the time of maximum delay jitter. Audio medium  $\tau_{diff}$  represents a time difference between duration time and relative duration time. Video medium  $\tau_{diff}$  is a result of jitter-compensatory time algorithm and it is obtained from comparing  $\tau_{diff}$  with  $\tau_{ja1}$  of audio medium. Applying maximum jitter delay not only improves QoS but also has an effect on compensating a time of playing out a place except key medium. A smoothing buffer 125ms is used at the BS to avoid a jitter delay in applications. A delay of a smoothing buffer is acceptable to video phone, video conference, tele-teaching and real-time application. The algorithm that decides the relative duration time is the followings.

**Procedure** Relative Duration Time

Begin {  $\tau_{ma}$  is arrival time of media };

If  $\tau_{ma} \leq \tau_b$  Then {  $\tau_b$  is maximum delay time };

$\tau_r := \tau_b$  ; {  $\tau_r$  represents a relative duration time to playout an arrived media };

Else

Begin

$P_{ma} := P_{ms} \times \tau_b / \tau_{ma}$ ; /\*  $P_{ma}$  is maximum size for media to playout during a duration time.  $P_{ms}$  is arrived size during a smoothing buffer delay \*/;

$P_r := P_{ma} / P_{ms}$ ; /\*  $P_r$  represents a playout rate of  $P_{ma} / P_{ms}$  \*/;

$\tau_r := P_r \times \tau_d$ ; /\*  $\tau_d$  is the duration time for which a playout or a display of a media \*/;

End

End

Jitter-compensatory time algorithm is the following.

**Procedure** Jitter Compensatory Time

Begin /\*  $\tau_{r-a}$  is relative duration time of audio \*/;

$\tau_{r-a} := \text{relative\_duration\_time}(Au)$ ;

$\tau_{diff} := \tau_d - \tau_{r-a}$  ; /\*  $\tau_{diff}$  is the time difference between a duration time and a relative duration time \*/;

While media Then

Begin

$\tau_{r-m} := \text{relative\_duration\_time}(m)$  ; /\*  $\tau_{r-m}$  is relative duration time of media excepts audio \*/;

IF  $\tau_{diff} \leq \tau_j$  Then /\*  $\tau_j$  is the acceptable maximum delay jitter time between each medium \*/;

Begin

wait (  $\tau_{diff}$  ) ;

$\tau_{r-mp} := \tau_{r-a} + \tau_{diff}$  ; /\*  $\tau_{r-mp}$  is playout time of media \*/;

End

Else

Begin

wait (  $\tau_j$  ) ;

$\tau_{r-mp} := \tau_{r-a} + \tau_j$ ;

End

End

End.

Transition  $t_4$  is fired only by  $A_1$  with key medium among input places in Figure 3. Let's assume that it takes 135ms for audio object and video object to reach the destination transit. A relative duration time of audio object will be 115ms according to a relative duration time algorithm and  $t_{diff}$  will be 10ms by a jitter-compensatory time algorithm, which means that audio object will wait as much as 10ms because it does not exceed the maximum delay jitter of 10ms. Video object  $t_{diff}$  will take a smaller one between  $t_{diff}$  and  $t_{ja1}$  of audio object. The relative duration time of video object will be 115ms by the relative duration time algorithm. 10ms from an audio object is added to 115ms of relative duration time of video object. Video object can be compensated in cases that

the total time added with  $t_{diff}$  of video object does not exceed 125ms. That is, video object can be compensated as much as 10ms because it does not exceed 125ms. If jitter is not considered, the delay of audio and video object gives a bad impact on playing a next frame, while a proposed jitter-considerable scheme results in playing a next frame efficiently.

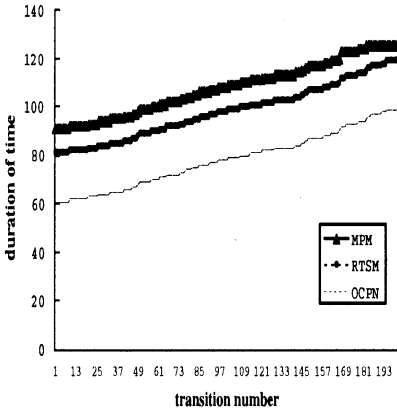
## 5 Performance Evaluation

Media format used in the simulation is in the following. The 1Kbytes audio data is encoded by the PCM encoding scheme, and resolution of the video frame is  $120 \times 120$ . Note that the number of video places corresponding to one audio place is random due to the following reason. At the BS, the application gets one audio packet from the audio device every 125ms, and it gets one, two or third video frames from the video frame grabber, which is determined by the run-time processing overhead of the operating system. In the paper, the simulation environment is set to the MH in ME. The sample data to be used in the simulation are generated by the Poisson distribution and the network jitter delay is equally applied to all media. The number of media is limited to two types for comparing the performance. Notice that the delay value generated by the normal distribution is forced to be within the range  $[0.5 * \text{mean}, 4 * \text{mean}]$  in the program to make the delay values more realistic. The number of transitions is set to 200 and the maximum delay jitter is 10ms. On the purpose to verify the algorithm, the previous OCPN and RTSM are also evaluated in the same simulation environment. The playout rate of the proposed delay jitter scheme based on the maximum delay jitter time is compared to the existing methods. The efficiency of the proposed scheme is observed by being comparing with the playout rate of the different models, in case of the normal and abnormal arrival of audio streams to BS, respectively. It is assumed that the average delay is 100ms and the variance is 20ms in case of audio streams with delay. Figure 4 shows the comparison of the playout time of RTSM and MPM model as functions of the number of transactions in case of audio streams without delay. In Figure 5, the result for the playout rate of RTSM and MPM model is plotted against the number of simulations, in which the playout rate of MPM model is increased to about 8%, as compared to the existing method.

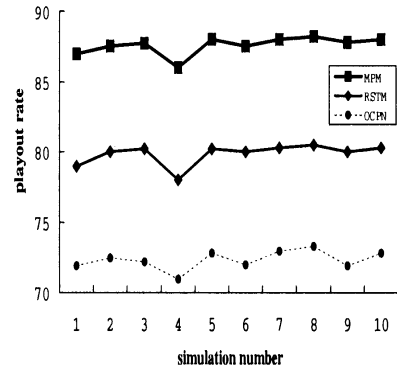
Figure 6 shows the simulation results of the three models in terms of the video duration time. The duration time of OCPN model dramatically decreased however, MPM model shows the better performance, compared with RTSM model.

In Figure 7, the result for the playout rate of the three models is plotted against the number of simulation. Result demonstrate that the playout rate for audio streams in MPM model is increased to about 8 % and 6.3 % in case of without and with delay, respectively. The result here seem to indicate that MPM model increases the playout rate and decreases the loss rate, in terms of audio and video, than OCPN and RTSM model.

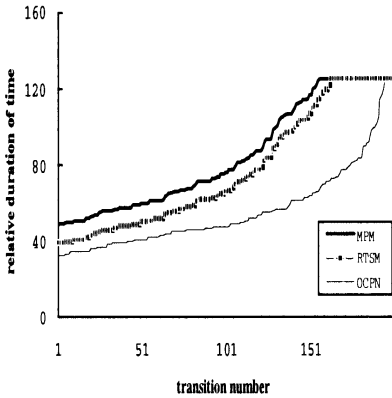




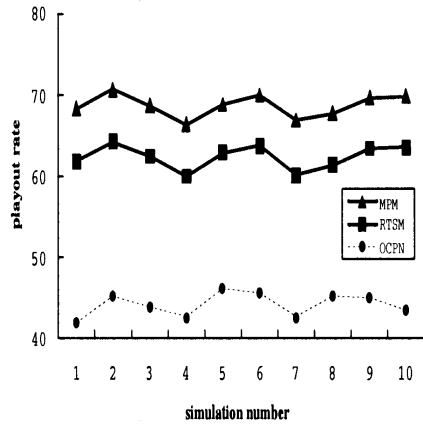
**Fig. 4.** Duration of time for audio without delay



**Fig. 5.** Playout rate for audio without delay



**Fig. 6.** Comparison of playout time with delay



**Fig. 7.** Comparison of playout rate with delay

## 6 Conclusions

In this paper we have proposed a new specification model to describe the QoS parameter and the delay time between media data in ME. The proposed MPM has a distinct property which allows QoS parameter. This is very important since most applications represent without QoS in ME. The contributions of this paper are listed in the following. 1) An MPM to describe the QoS parameter for intermedia synchronization in ME is proposed. 2) The concept of multiple key media in MPM for synchronization control is suggested. 3) The concept of time medium associated with multiple key media to deal with late transmis-

sions of multiple key media is suggested. 4) The proposed method meets the requirements of synchronization between media data by handling dynamically the adaptive waiting time resulted from variations of delay time and the interval adjustment using maximum delay jitter time at BS in ME. The proposed MPM model supplies flexible description of the temporal relationships among various media. This model can provide modeling specifications from the live applications to the pre-orchestrated applications in ME. MPM model adaptively manages the waiting time of smoothing buffer, which leads to minimize the gap from the variation of delay time and it results in naturally playout to MH in ME. Also it is suitable to the system which requires the guarantee of high QoS and the model improves QoS such as decrease of loss rate and increase of playout rate in ME.

## References

1. D. H. Nam and S. K. Park.: Adaptive Multimedia Stream Presentation in Mobile Computing Environment, Proceedings of IEEE TENCON, Cheju Korea.(1999) 966–969
2. A. Boukerche, S. Hong and T. Jacob.: MoSync: A Synchronization Scheme for Cellular Wireless and Mobile Multimedia Systems, Proceedings of the Ninth International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems IEEE. (2001) 89–96
3. M. Woo, N. Prabhu and A. Grafoor.: Dynamic Resource Allocation for Multimedia Services in Mobile Communication Environments, IEEE J. selected Areas in Communications, Vol.13, No.5, June. (1995) 913–922
4. D. H. Nam, S. K. Park.: A Smooth Playback Mechanism of Media Streams in Mobile Computing Environment, ITC-CSCC'98. (1998)
5. P. W. Jardetzky, and C. J. Sreenan, and R. M. Needham.: Storage and synchronization for distributed continuous media, Multimedia Systems/Springer-Verlag. (1995)
6. C.-C. Yang and J.-H. Huang.: A Multimedia Synchronization Model and Its Implementation in Transport Protocols, IEEE J. selected Areas in Communications, Vol.14, No.1, Jan. (1996)
7. R. Steinmetz.: Human Perception of Jitter and Media Synchronization, IEEE J. selected Areas in Communications, Vol.14, No.1, Jan. (1996)
8. D. L. Stone, and K. Jeffay.: An empirical study of delay jitter management policies, Multimedia Systems/Springer-Verlag. (1995)
9. G. Blakowski and R. Steinmetz.: A Media Synchronization Survey: Reference Model, Specification, and Case Studies, IEEE J. selected Areas in Communications, Vol.14, No.1, Jan. (1996)
10. M. J. Perez-Luque and T. D. C. Little.: A Temporal Reference Framework for Multimedia Synchronization, IEEE J. selected Areas in Communications, Vol.14, No.1, Jan. (1996)