A SYNCHRONIZATION MODEL FOR PRESENTATION OF MULTIMEDIA OBJECTS

In-Ho Lin, Bih-Hwang Lee* and Chwan-Chia Wu
Department of Electrical Engineering
National Taiwan University of Science and Technology
Taipei, Taiwan 106, R.O.C.

Key Words: multimedia synchronization, temporal relations, media group, resource scheduling.

ABSTRACT

To support the presentation requirements of distributed multimedia information, synchronization of multimedia objects must be achieved. To this end, system resource scheduling and resource reservation for object pre-fetch, network bandwidth and buffer occupancy must be determined prior to the time the presentation is initiated. This paper proposes an object-oriented model to handle the temporal relationship for all of the multimedia objects at the presentation platform and study the related problems of resource allocation. Synchronization of the composite media objects is achieved by ensuring that all objects presented in the upcoming “manageable” period must be ready for execution. To this end, the nature of overlap is first examined for various types of objects. The importance of critical overlap and critical point that are vital to synchronization is addressed and taken into account in this research. The concept of manageable presentation interval and the irreducible media group are also introduced and defined. Analysis of resource allocation among pre-fetch time of media object, network bandwidth and buffer occupancy is also examined. Accordingly, a new model called group cascade object composition Petri-net (GCOCPN) is proposed and an algorithm to implement this temporal synchronization scheme is presented.

I. INTRODUCTION

With the advance of computer, network and multimedia technologies in the past decade, the demands for multimedia information services from complex environments are rapidly growing in different fields including education, entertainment, CAE, Web browsing, and internet telephony. It is an emerging trend that the integration of various multimedia objects, which can include text, images, audio, video, graphics and so forth, over heterogeneous and distributed environments, to furnish a particular application service is essential and inevitable. Presentation of distributed multimedia information involves temporal organization, spatial organization, prefetch, transformation and delivery of components, which compose the multimedia information for the user and allow the user to interact with the presentation sequence as well. As to the issues of temporal synchronization and resource allocation,

*Correspondence addressee

the media characteristics, temporal dependence and resource utilization must be clearly established to ensure proper scheduling of the synchronized presentation.

Multimedia data can be classified into two categories: discrete media and continuous media (Allen, 1983; Blakowski and Stenmetz, 1996; Little and Ghafoor, 1990; Nicolaou, 1990). Real-time data, such as video, audio and animation that require time-ordered presentation to the user are basically classified as continuous media. On the other hand, text, images and graphics media are basically time-independent and are normally presented on a page or frame basis. They are therefore classified as discrete or static data. The specification of temporal composition is required to describe these objects completely by taking all of the temporal relationships into account.

Synchronization in a multimedia presentation system refers to the temporal relations between media objects; it makes multimedia presentation take place in the desired time-ordered sequence exactly at the predetermined starting and ending time instants. There are two basic types of synchronization, intra-media and inter-media synchronization: intra-media synchronization refers to the time relations between various presentation units of one continuous media object; and a sequence of media units such as video frames should be played back continuously and smoothly for ensuring the serial media synchroniza-tion. Inter-media synchronization refers to the synchronization among different media objects, which may be retrieved and transferred from different places and be presented in parallel. In order to offer a better performance and quality of services for multimedia applications, synchronization constraints for both the inter- and intra- media objects must be specified and maintained.

For supporting distributed multimedia applications, many researchers (Little and Ghafoor, 1990; Little and Ghafoor, 1993; Iino *et al*., 1994; Qazi *et al*., 1993; Raghavan *et al*., 1996; Yang and Huang, 1996) adopt the concept of Petri-net (Murata, 1989) to construct a reference model for archiving multimedia synchronization. The object composition Petri-net (OCPN) model (Little and Ghafoor, 1990) specifies temporal requirements at the presentation level. Qazi *et al*. propose the XOCPN model (Qazi *et al*., 1993) that makes some improvement of OCPN by introducing a synchronization interval unit (SIU) to resolve the network communication delay problem. There also has been intensive research on the subject of temporal synchronization (Qazi *et al*., 1993; Raghavan *et al*., 1996; Yang and Huang, 1996). Several models have been designed and presented to cope with the problem of latency resulting from resource allocation, data generation, packet assembly, network communication, etc. (Qazi *et al*., 1993; Raghavan *et al*., 1996). Media derived from different sources may introduce different delays to the data transmission path and thus result in lip-sync or delays jitters problems during presentation. However, up to this moment, none of the prior works presents a simple yet comprehensive model to resolve the aforementioned problems.

In addition, resource allocation and scheduling are essential to ensure achievement of the intra- and inter- objects synchronization requirements for multimedia presentation. To facilitate a smooth playback of media objects with satisfactory QoS, it is required that objects should be presented to the memory before they are delivered for playback.

The purpose of this paper is to present an object-oriented model that specifies and manipulates the temporal relationships of all multimedia objects at the presentation platform. Synchronization of the composed media objects is achieved by ensuring that all of the media objects to be presented in the upcoming “manageable” period must be available for execution. The basic concept is therefore to analyze the temporal relationship and to partition the whole of the composite media objects into several “manageable” groups that can be presented in sequence. To this end, we first investigate the nature of overlays for various types of objects. Based on our observation, we define the manageable presentation interval and introduce the concept of presentation groups. The resource scheduling of each presentation group for buffer occupancy versus pre-fetch time of media object is also examined. We then develop a pre-fetch scheme for describing the requirement of buffer occupancy versus pre-fetch time. Accordingly, we present a new model called group cascade object composition Petri-net (GCOCPN) to cope with the synchronization problems and an experimental result to compare the required buffer occupancy from the original multimedia system to the proposed GCOCPN model.

This paper is organized as follows: Section 2 outlines the related works in this field and the background of multimedia synchronization models; Section 3 introduces the concepts of multimedia information group and irreducible media group which are to be used in this paper; Section 4 provides the analysis of system parameters which include buffer occupancy, network bandwidth and pre-fetch time; section 5 provides an experimental result by comparing the resource utilization from the original systems to the proposed GCOCPN model. Finally, conclusions are stated in section 6.

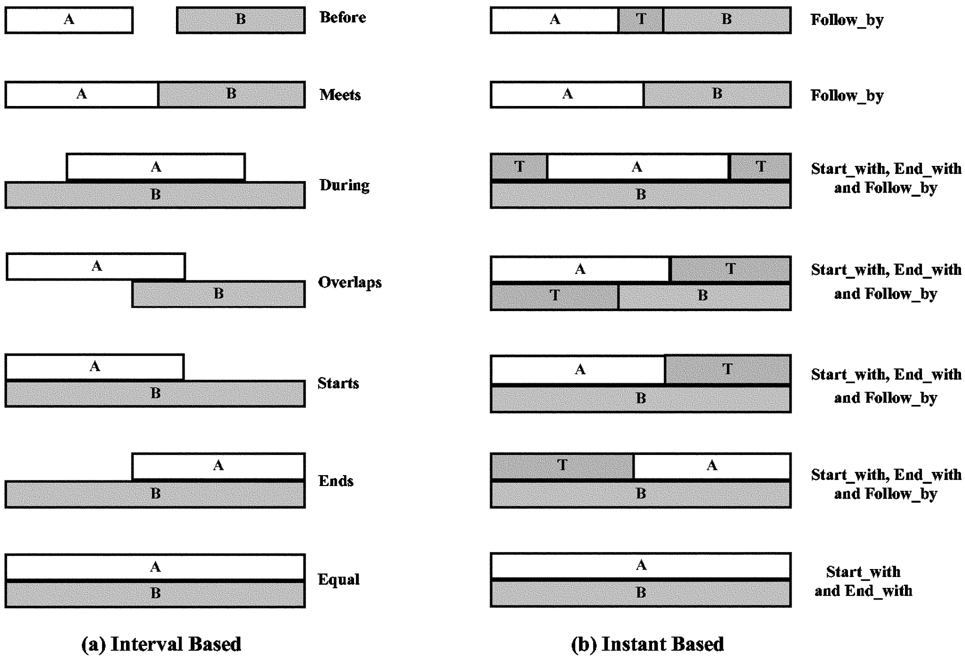


Fig. 1 Temporal relations for representation

II. FUNDAMENTALS

1. Temporal Relations

Temporal relationships describe how media objects are simultaneously acquired, formulated, and presented in time ordered sequence to produce multimedia information. Many researchers have addressed the theories of action and time based on intervals and points (Allen, 1983; Jixin and Knight, 1994a; Jixin and Knight, 1994b). Two of them take intervals as primitive objects and define temporal relations among intervals as temporal quantities (Allen, 1983; Jixin and Knight, 1994a). The other one deals with temporal relations among points based on an instant-based approach (Jixin and Knight, 1994b), where temporal relations among intervals are defined in terms of points and the corresponding ordered relations among points. Temporal relations and temporal intervals have been widely used for solving multimedia synchronization by many researchers (Little and Ghafoor, 1990; Little and Ghafoor, 1993; Iino *et al.*, 1994; Qazi *et al.*, 1993); and those relations are basically used to describe how two intervals relate to each other in time domain.

In the research (Little and Ghafoor, 1990), on classification of all possible temporal relations, Little and Ghafoor found that a total of thirteen temporal relations can be used to represent the relationships among any two possible intervals. These thirteen categories can be further simplified to seven since each of the remaining six relations is an inverse of one of the seven simplified relations. These seven relations are *before*, *meets*, *during*, *overlaps*, *starts*, *ends* and *equal* as shown in Fig. 1(a). Five of those are used to describe the parallelism among media objects and to model the specifications of the inter-media synchronization. The rest of two describe the intra-media synchronization among various presentation units of the same media object represented in sequential order of time. A common representation of these media objects is based on temporal intervals, defined by a temporal-interval-based (TIB) model (Little and Ghafoor, 1993).

In order to facilitate our research, temporal interval is represented by its starting time and ending time via instant-based modeling. Namely, for a sequence of media objects, we can describe their time dependency as a sequence of starting and ending time instants. A time object is also introduced here to construct our model; it is a virtual media object and normally contains a duration of time. A time object is nothing but a shift of time on its playback action. It is inserted into the vacant time interval between any two objects that are segregated by a given time interval and thereby the result will not affect the synchronization requirement of multimedia presentation. By the introduction of time object, the temporal relations of media objects classified by the aforementioned seven relations can be further simplified in terms of three primitive relations: start_with, end_with, and follow_by (Fung and Pong,

1994) as depicted in Fig. 1(b), where objects A and B can be any types of media objects, while T represents a time object with a duration of time T. They also comply with the extended set of definitions with parallel first, parallel last and sequential in (Hoepner, 1992).

2. Temporal Overlap

When multiple objects with different media types are presented at the same platform simultaneously at some specific time intervals, temporal overlap results. Without loss of generality, a temporal overlap can be defined as follows:

Definition 2.1: *Let A be an object of a certain media type with the starting playback instant $t_{cs}(a)$ and the temporal interval $T_c(a)$; B be another object with the starting playback instant $t_{cs}(b)$ and the temporal interval $T_c(b)$, then object A and B are temporally overlapping if (1) $t_{cs}(a) \leq t_{cs}(b)$ and $(t_{cs}(a)+T_c(a)) \geq t_{cs}(b)$; or (2) $t_{cs}(b) \leq t_{cs}(a)$ and $(t_{cs}(b)+T_c(b)) \geq t_{cs}(a)$.*

Note that in Definition 2.1, there is no constraint of media type on either object A or object B. That is, when object A is temporally overlapping with object B, object A can be either continuous or discrete data, and so can object B.

3. Critical Overlap and Critical Point

Our synchronization problem is to ensure media objects are to be presented, not only in the desired time-ordered sequence, but also following the timing requirements imposed on the relevant media objects involved in the presentation. Temporal overlap is therefore a vital factor for multimedia synchronization. In particular, if media object A temporally overlaps with media object B with $t_{cs}(a) \leq t_{cs}(b)$, then a loss of synchronization may result if media object B arrives and starts its playback later than the pre-scheduled time instant $t_{cs}(b)$. By taking the characteristics of continuous and discrete media types into account, one may observe that the aforementioned example will result in un-synchronization of the presentation if media object A is a continuous object disregarding the media type of object B, whilst this example will not affect the QoS of multimedia presentation if media object A is a discrete data object. In other words, when presentation of two objects A and B are temporally overlapping and $t_{cs}(a) \leq t_{cs}(b)$, then availability of object B at $t_{cs}(b)$ for presentation plays the key role for synchronization if object A is continuous data. Such overlap is therefore "critical" from the synchronization point of view.

Figure 2(a) shows the temporal relation of two

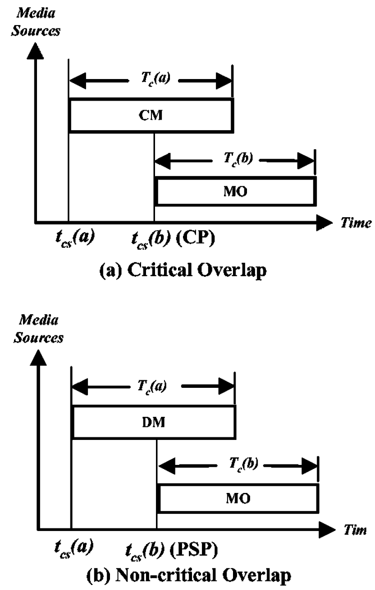


Fig. 2 Temporal overlap

critically overlapped objects in which time instant $t_{cs(b)}$ is the critical point (CP) of such critical overlap where CM refers to a continuous media object while MO refers to a media object of any type.

4. Pseudo Separation Point (PSP) and Separation Point (SP)

A temporal overlay which is not critical is referred to as non-critical overlap and is illustrated in Fig. 2(b), where a discrete media (DM) object is presented prior to the other media object (MO). Since a DM object is a time-independent data type and is presented statically, it can be temporally decomposed into any number of smaller (in duration) objects of the same type without violating the requirement of synchronization. We will therefore focus our discussion on resolving synchronization problems of critical overlap by adapting temporal properties of this non-critical overlap. It is clear that any playback delays of an MO object in a non-critical overlap as shown in Fig. 2(b) will not degrade QoS of presentation. This means that the temporal overlap may broken into two separate non-overlapped information groups due to the following property: the discrete media objects of a non-critical overlap can be partitioned at the instance of starting playback time of MO object while keeping the criteria of synchronization unchanged. The time instance that the two objects intersect is called a pseudo separation point (PSP). A temporal relation of non-critical overlap is illustrated in Fig. 2(b) where the $t_{cs}(b)$ is the PSP of this non-critical overlap. If media object MO is a continuous media object type, then the PSP is further called as a separation point (SP).

III. CONCEPT OF MULTIMEDIA GROUPING

1. Multimedia Information Group

Multimedia information is composed of various media objects of continuous or discrete data type and is organized in any temporal relationships among objects. It is clear that critical overlaps may result in any particular temporal relationship. Note that critical points may induce problems of synchronization during presentation, such as lip-sync, jitters and loss of information representations. In a distributed multimedia information system, critical point is particularly sensitive to latency resulting from network delay, resource allocation, etc. In order to make sure that the media objects are ready at the critical point when the CM object is scheduled to playback, the concept of pre-fetch is proposed (Akyildiz and Yen, 1996). To facilitate pre-fetch of media objects to achieve synchronization at the critical points, requires allocation of memory buffer and communication bandwidth to store the pre-fetched data for later playback. However, since the availability of resources can not be predicted in advanced, especially in a distributed computing environment, it is difficult to determine the optimal time to activate a pre-fetch prior to presentation of the particular media object actually taking place. Clearly, it is totally impractical to pre-fetch all media objects at the beginning of the presentation as this will occupy too much of the system. It is therefore desirable to find a systematic way to activate pre-fetches that will guarantee synchronization to be achieved at the critical points and, in the mean time, to keep the occupation of system resources to the minimum. Since an SP can facilitate partition of a DM object into two identical objects without making any violations to present, multimedia information can thus be divided into several smaller information groups at each SP without affecting their representations of the whole of the multimedia information.

An SP is a separation point from which multimedia information can be separated into two groups. Each group composed of various media objects with continuous or discrete data type is indeed a completed information unit with a group starting instant t_{gs} and a group interval T_g as that characterizing a media object. The advantages of dividing multimedia information into smaller groups are three formats: first, combining the diverse media types with temporal overlapping relations in a uniting information; secondly, providing the features of being easily accessed, shared, and retrieved of media objects over the computer network and being smoothly presented on the presentation platform; thirdly, reducing the overhead of resources allocation at the presentation level. We therefore introduce the concept of media group. For simplicity, we use the notation $O(i, j)$ to represent the ith media object of the jth media source type of a multimedia information group. Here a media source is a collection of data with similar formats and characteristics for storage and presentation. For instance, a multimedia information may be a combination of video, audio, text, graphics, and so on, where video and audio are different media sources. Now, the definition of media group is given as follows:

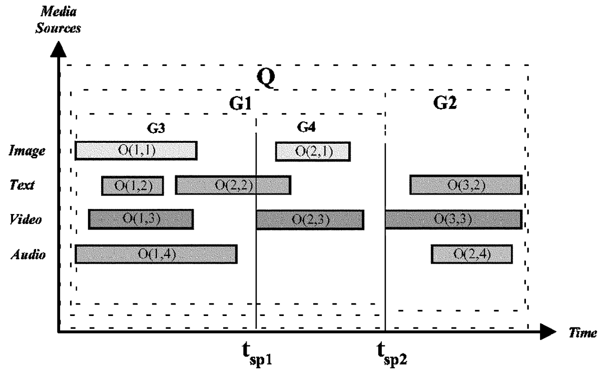


Fig. 3 Media group partitioning

Definition 3.1: *Let $Q=\{O(i, j)|1\le i<n, 1\le j<m\}$ be the set of all media objects of a multimedia information program and $G_k \subset Q$. Then G_k is defined as a group if no such temporal relation $t_{cs}(p, l)\le t_{cs}(i, j)+T_c(i, j)$ exits for all $O(i, j)\in G_k$ and $O(p, l)\in Q-G_k$, where $t_{cs}(p, l)$ denotes the starting instant of object $O(p, l)$, $t_{cs}(i, j)$ and $T_c(i, j)$ denotes the starting instant and the time interval of object $O(i, j)$, respectively. In other words, any object in a group must not temporally overlap with an object not in this group.*

In Fig. 3, Q represents the set of all media objects of a multimedia information program. According to definition 3.1, G1 and G2 are two separate media groups; and G1 can be further partitioned into G3 and G4 at a particular separation point t_{sp2}.

It is clear that the multimedia information program itself is a media group. Ideally, all media objects in an information group can be stored in a single storage as a hybrid object with a special file format, such as MPEG, AVI, and CD-I. However, in real world applications, the media objects may come from different storage locations. They are difficult and not economical to pre-compound in a single hybrid object. Nevertheless, it can be logically used to represent a complete uniting multimedia information. The starting playback time of a media group can be treated as a group synchronization point with other groups. In order to achieve temporal synchronization, all media objects in one group should be available to play-out at this instant.

2. Irreducible media group

As explained above, a media group can be treated as an independent presentation unit. All media objects of this group must be available for presentation in some sense before the presentation is activated. This may require resource allocation for all these media objects. As a result, a "larger" group may require more system resources such as storage buffer, communication bandwidth, etc.. It is therefore conceivable that the smaller a media group is, the better occupancy of resources for presentation will be. Therefore in the example of Fig. 3, the group partition {G3, G4, G2} is better than {G1, G2} as G1 is partitioned into G3 and G4 resulting in more economical resource allocation for presentation. This comes up with the requirement of searching for the irreducible media groups. An irreducible media group is a multimedia information unit, which is comprised of media objects within some deterministic time interval; it will begin at some SP and terminate at the next ordered SP. An irreducible media group with a starting playback instance of time and temporal interval is a non-separable synchronization unit. In other words, no other SP can be found in an irreducible media group. An irreducible media group can therefore be defined as:

Definition 3.2: *Let* $G=\{O(i, j) | \text{ for all } i, j;\ 1 \le i < \overline{n},\ 1 \le j < \overline{m}\} \subset Q$ *be a media group. If no SP can be found within G, then G is an irreducible media group.*

In Fig. 3, G2, G3, and G4 can not be further decomposed into smaller groups and thus are all irreducible media groups. Based on the aforementioned definitions and discussions, the algorithm to find out separation points SPs in a time line based multimedia presentation program is presented as follows:

Algorithm 3.1 Search for *separation points* in the time-line based multimedia presentation program.

FindoutSP(Multimedia Information Q)

SP:={}
For each CM type object $O(i, j)$ in Multimedia Information program Q, where i and j represent the ith object in jth media type
do
if (the staring playback time $t_{cs}(i, j)$ of object $O(i, j)$ is located within the interval of any DM type object meanwhile $t_{cs}(i, j)$ do not overlapped with any others CM type media object)
then SP:=SP$\cup t_{cs}(i, j)$

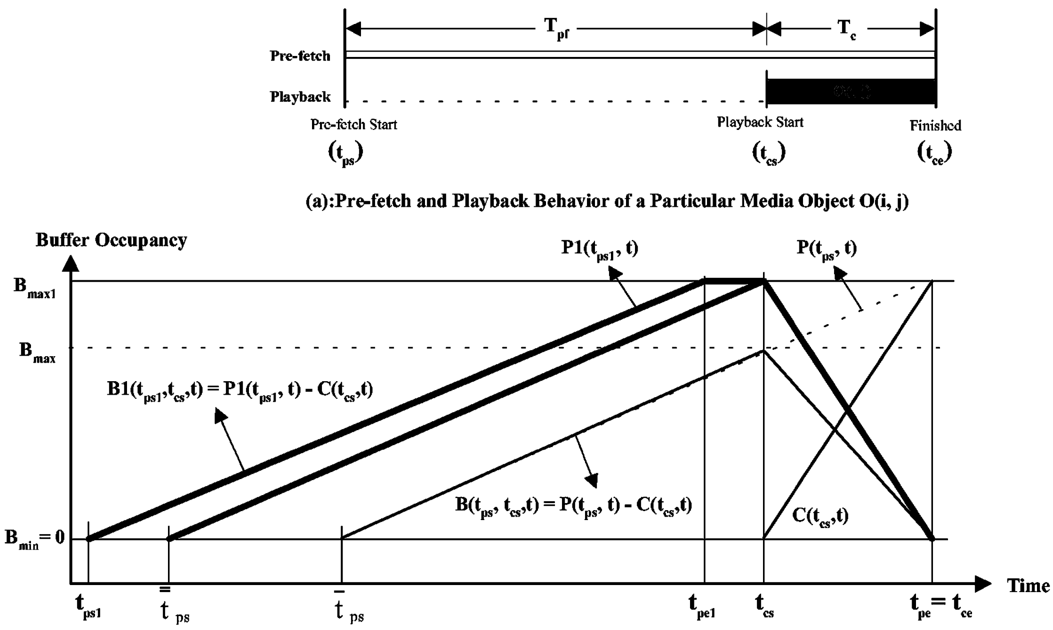
IV. ANALYSIS OF SYSTEM PARAMETERS

The presentation of a distributed multimedia information system must not only support fast data transfer by which a guaranteed delivery of data stream can be achieved through a bandwidth-constrained network, but also provide sufficient system resource to ensure synchronization among multiple independent data sources. Several networks can support transmission of multimedia applications, such as FDDI, high-speed Ethernet, or ATM networks with bandwidth of 100 Mbps and above. However, multimedia applications require communication services to satisfy the QoS requirement at each specific layered architecture. In addition, in a distributed multimedia presentation system, media object streams are continuously retrieved from various servers through reserved network channels, added to an intermediate buffer at the presentation platform, and continuously consumed from the buffer by the output device. It is very similar to the behavior of multimedia data storage and retrieval described in (Gemmell and Christodoularkis, 1992; Rangan and Vin, 1993). To support the presentation of diverse media data, resources scheduling disciplines associated with real-time adaptive retrieval schemes are required. Although the system latencies in a distributed system are problematic due to the fact that several streams are originating from different sources, synchronization satisfying QoS requirements can still be achieved by proper resources allocation and adaptive pre-fetch schemes.

In this section we present the basic requirements for successful retrieval of data streams to facilitate playback of media objects with the desired QoS. In particular, we investigate the minimal buffer requirements and the minimal pre-fetch time for data retrieval through an allocated bandwidth to ensure continuous playback of multimedia data streams on a pre-designated time schedule. We first examine the behavior of pre-fetch and playback of a particular media object to construct the relationship between buffer occupancy and pre-fetch starting time. Based on the result, a generalized resource scheduling and pre-fetch scheme for multiple objects within an irreducible media group is then investigated. Finally, we formulate the basic requirements for resource scheduling and pre-fetch schemes of a distributed multimedia information model so that inter and intra media synchronization can be achieved.

1. Relationship between Buffer Occupancy and Communication Bandwidth

In order to simplify our discussion, we assume throughout this paper that the media object stream is



(a):Pre-fetch and Playback Behavior of a Particular Media Object O(i, j)

(b): Relationships between Production Function $P(t_{ps}, t)$, Consumption Function $C(t_{cs}, t)$ and the Buffer Occupancy Function $B(t_{ps}, t_{cs}, t)$ of a particular Media Object O(i,j)

Fig. 4 Buffer occupancy function for an individual media object retrieval and playback

transferred from a remote server via an allocated network channel with bandwidth Bw. Furthermore, the media data is stored in a non-compressed state, the consuming function of each object can be approximated as linear and thus is presented at a constant consumption rate Ω. We also assume that $\Omega > Bw$ throughout the discussion stated below.

During the playback of a media object, data streams are requested from the presentation platform, retrieved from the server station, transferred via a network with the pre-allocated bandwidth, stored in an intermediate buffer and then continuously processed for playback by the particular output device. Since a guaranteed timely availability of a media stream for continuous playback is required, once the media object is fired for playback, one must ensure that data streams received at the presentation platform and stored in the buffer are always sufficient for data consumption. Since data consumption rate Ω is presumably greater than the allocated bandwidth Bw, as a result, pre-fetch must be employed prior to playback taking place. Fig. 4(a) is an illustration of pre-fetch and playback behavior of a particular CM object $O(i, j)$ at the presentation platform with starting playback instant t_{cs}, playback interval T_c (playback finished at time t_{ce}). Observing that the total amount of data consumed equals $\Omega \bullet T_c$ which must be delivered before t_{ce} provided that the processing time of the data stream for playback is negligible. This means that the data fetch cycle must be ended before t_{ce}, namely, $t_{pe} \leq t_{ce}$, where t_{pe} is the pre-fetch end time. Furthermore, since data transfer rate is assumed to be constant and equals Bw and $\Omega > Bw$, therefore, the pre-fetch starting time must not be later than $\overline{t}_{ps}$, where

$$\overline{t}_{ps} = t_{ce} - \Omega \bullet T_c / Bw \tag{1}$$

That is $\overline{t}_{ps}$ is the maximum pre-fetch starting time.

To consider the retrieval of a media stream from the network, pre-fetch of data is started from server station at t_{ps}. Let $P(t_{ps}, t)$ be the production function at time t, and denote the total amount of data transmitted to the presentation platform, this function can be expressed as:

$$p(t_{ps}, t) = \begin{cases} Bw \bullet (t - t_{ps}) & \text{if } t_{ps} \leq t \leq t_{pe} \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

Meanwhile, the total number of data having been consumed for playback at time t is defined as consumption function $C(t_{cs}, t)$, which is represented by:

$$C(t_{cs}, t) = \begin{cases} \Omega \bullet (t - t_{cs}) & \text{if } t_{cs} \leq t \leq t_{ce} \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

Since data in buffer B is accumulated by a production function of $P(t_{ps}, t)$ and is consumed by a consumption function of $C(t_{cs}, t)$, the total buffer size occupied by the data streams at time t is then denoted by a buffer occupancy function $B(t_{ps}, t_{cs}, t)$ and is represented by:

$$B(t_{ps}, t_{cs}, t)=P(t_{ps}, t)-C(t_{cs}, t) \tag{4}$$

In order to avoid running out of data for playback during presentation, the buffer occupancy function must be always positive during playback session, $t_{cs} \le t \le t_{ce}$. Let the pre-fetch time T_{pf} be defined as the time interval between pre-fetch starting time t_{ps} and playback starting time t_{cs}, that is $T_{pf}=t_{cs}-t_{ps}$. The minimum pre-fetch time is therefore defined as:

$$\overline{T}_{pf} = t_{cs} - \overline{t}_{ps} \tag{5}$$

Note that data is increasingly accumulated in the buffer after pre-fetch is performed and up until playback of the media stream begins, at which the buffer occupation starts decline for $\Omega > Bw$. Therefore the maximum buffer requirement $B_{\max}$ occurs at t_{cs}. However, this is not always true if we move the starting time of pre-fetch ahead a step further until $\overline{\overline{T}}_{pf}=\Omega \bullet T_c/Bw$, or equivalently $\overline{\overline{t}}_{ps}=t_{cs}-\overline{\overline{T}}_{pf}$. It is clear that any pre-fetch starting time earlier than $\overline{\overline{t}}_{ps}$ does not result in a greater buffer requirement than $B_{\max}(\overline{\overline{T}}_{pf})=\Omega \bullet T_c$. Therefore, $B_{\max}(\overline{\overline{T}}_{pf})$ is the worst case for buffer requirement. Fig. 4(b) illustrates the variation of maximum buffer requirements for each pre-fetch starting time. Observing that the buffer occupancy is always the maximum at time t_{cs}. The buffer requirement associated with a given pre-fetch time t_{pf} can therefore be summarized as:

$$B_{req}(T_{pf})=B_{max}(T_{pf})=\mathrm{Min}(Bw \bullet T_{pf}, \Omega \bullet T_c) \quad \text{for } T_{pf} \ge (t_{cs} \ge \overline{t}_{ps}) \tag{6}$$

Note that the buffer requirement for ensuring continuous playback is in between $(\Omega - Bw) \bullet T_c$ and $\Omega \bullet T_c$, since $\overline{t}_{ps}$ is the pre-fetch starting time that requires the minimum buffer occupancy for a given communication bandwidth Bw. Accordingly, the buffer requirement can be given by the following:

$$B_{req}(Bw)=(\Omega - Bw) \bullet T_c \tag{7}$$

2. Resource Scheduling for Multiple Objects of the Same Media Type

In the previous section, we have derived the resource requirements and setup the relationship of buffer occupancy versus pre-fetch time for a single particular media object. Now, we continue to examine the case that multiple time-dependent objects with the same media source type are in the same irreducible media group and each is specified by a particular temporal specification. These media objects are retrieved in sequence from the same server station and through the same network channel with a preserved bandwidth, and then played back to the same output device. Based on the result obtained in section 4.1, buffer requirements for each media object can be derived by the corresponding pre-fetch time. Since pre-fetch time must start before playback, when performing pre-scheduling to ensure inter and intra-synchronization, it is possible to result in overlap of time interval for data retrieval of two media objects as illustrated in Fig. 5. Namely, even if data retrieval of the previous object is still being processed, pre-fetch of the next object has to be started due to the deadline constraint. It is clear that unless the bandwidth of the reserved channel is doubled for this overlapped duration, the pre-allocated bandwidth can not support sufficient data transfer to ensure jitter-free playback of these media objects. Since every data stream has to be prepared for playback before deadline, and further, a constant bandwidth is pre-allocated via a QoS negotiation during system initialization, the only way to resolve this problem is to start the pre-fetch of the proceeding object earlier so that it can complete data retrieval before the starting of pre-fetch for the following object.

An irreducible media group may be composed of objects of more than one media source type, each media source type may have several objects for playback in sequence. Fig. 5(a) illustrates two video objects to be played back one by one at pre-scheduled time durations in one group. Let $\Delta T_{i,i+1}(j)$ be the time difference between the pre-fetch ending time of object $O(i, j)$ and the pre-fetch start time of object $O(i+1, j)$:

$$\Delta T_{i,i+1}(j)=t_{pe}(i, j)-t_{ps}(i+1, j) \tag{8}$$

When $\Delta T_{i,i+1}(j)>0$, there is an overlap of data retrieval for object $O(i, j)$ and $O(i+1, j)$ during the time interval of $[t_{ps}(i+1, j), t_{pe}(i, j)]$ as shown in Fig. 5(b). In order to prevent both objects from using the same network channel at the same time, the pre-fetch time of object $O(i, j)$ should be started earlier as shown in Fig. 5(c), so that data retrieval of $O(i, j)$ can be completed before $t_{ps}(i+1, j)$. The same procedure must be carried out repeatedly until an object $O(l, j)$ is obtained, where $\Delta T_{l,\, l+1}(j) \ge 0$. Algorithm 4.1 is to find out the pre-fetch time and maximum buffer requirement for a specific media source type $R(j)$ in an irreducible media group k, where j represents the jth media source type in this irreducible media group.

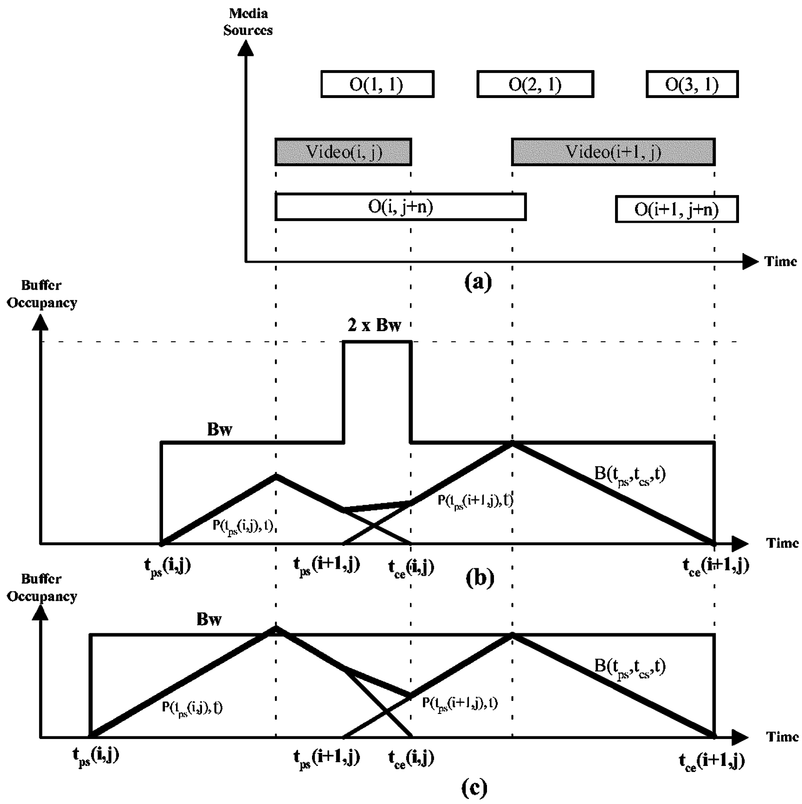


Fig. 5 Relationship between buffer occupancy function, network bandwidth and pre-fetch time for multiple CM objects

Algorithm 4.1: Calculation of pre-fetch time and maximum buffer requirement for a specific media type $R(j)$ in an irreducible media group k

FindMaxBuff (all the media objects in medai type $R(j)$)

For each media object $O(i, j)$ of media type $R(j)$ in media group k, where i represent ith object in jth media type
do begin from the last object of media type $R(j)$
if (the pre-fetch starting time $t_{ps}(i+1, j)$ of object $O(i+1, j)$ is earlier than the pre-fetch ending time of $t_{ps}(i, j)$ of object $O(i, j)$)
then
Update the pre-fetch starting time $t_{ps}(i, j)$ and pre-fetch ending time $t_{ps}(i, j)$ of object $O(i, j)$
calculate the maximum buffer occupancy $B_{\max}$ of $O(i, j)$according to Eq. 6
Update the maximum buffer requirement $B_{\max}(j)$ of media type $R(j)$
Until all the objects are examined

Let t_{gs} and t_{ge} be the starting and ending time of an irreducible media group, an associated buffer occupancy function B_j for each media source type $R(j)$ in group can be expressed as:

$$B_j(t) = \sum_{i=1}^{I} B_i(t_{ps}, t_{cs}, t) \quad t_{gs} \le t \le t_{ge} \tag{9}$$

Here I is the number of data objects of media source type $R(j)$. The total required buffer occupancy function $\Theta(t)$ in an irreducible media group can thus be derived as the summation of all the buffer occupancy function B_j of each media source type $R(j)$, and can be expressed as a function of time:

$$\Theta(t) = \sum_{j=1}^{J} B_j(t) \quad t_{gs} \le t \le t_{ge} \tag{10}$$

Note that, the maximum buffer occupancy for each media source type may not occur at the same time. The buffer requirement for this particular irreducible media group is therefore given by:

$$\Theta_{buf} = \mathrm{Max}(\Theta(t)) \text{ where } \Theta_{buf} \le \sum_{j=1}^{J} B_{\max}(j) \tag{11}$$

3. Waiting Time for the Firing of an Irreducible Media Group

From the discussion given above, it is clear that the starting time of pre-fetch of a particular media object can be determined when network bandwidth and buffer space is allocated. Similar to the discussion addressed in section 4.2, pre-fetch of media objects of a given irreducible media group can only start after completion of data retrieval of the same type of media objects in the previous irreducible media group. Further, this given irreducible media group can only be fired after retrieval of data streams has taken place for this particular pre-fetch time. As a result, some waiting time may be introduced so that presentation of this irreducible media group may leave behind the schedule specified by the temporal relation. This is acceptable since the introduction of group waiting time does not degrade the QoS for multimedia presentation.

Let $\Delta\tau_{k-1,\,k}(j)$ be the time difference between the pre-fetch ending time of the last object in group $k-1$ and the pre-fetch starting time of the first object with the same media source type $R(j)$ in group k, denote as $t_{pe}(I, j, k-1)$ the pre-fetch ending time of the last object in group $k-1$ and $t_{ps}(1, j, k)$ the pre-fetch starting time of the first object in group k:

$$\Delta\tau_{k-1,\,k}(j)=t_{pe}(I, j, k-1)-t_{ps}(1, j, k) \tag{12}$$

When $\Delta\tau_{k-1,\,k}(j)>0$, there is an overlap of data retrieval for the last object in group $k-1$ and the first object in group k during the time interval of $[t_{ps}(1, j, k), t_{pe}(I, j, k-1)]$. In this case, firing of group k should delay at least $\Delta\tau_{k-1,\,k}(j)$ for preventing the in-sufficiency of network bandwidth, such delay is the group waiting time introduced during presentation. The same procedure must be performed repeatedly until every two contiguous groups are scanned.

V. GROUP CASCADE OCPN MODEL

We are now presenting a new synchronization model for multimedia presentation in distributed computing systems. The objective is to ensure precise firing of media objects at the prescheduled critical points disregarding where these objects are originally located. To achieve this goal, some mechanism must be furnished for pre-allocation of resources before playback of the object actually takes place. As any pre-allocation of the resource will result in reservation of system resources to a certain degree and thus decrease the utilization rate if the period of reservation takes too long. It is therefore desirable to reduce the requirement of resource pre-allocation to the largest extent. The concept of irreducible media group is therefore adopted in our proposed modeling through which resource allocation is performed for each irreducible media group individually. In particular, we propose a synchronization model called group cascade object composition Petri-net (GCOCPN) (Lin *et al*., 1998a; Lin *et al*., 1998b) by which an entire multimedia presentation unit is decomposed into a number of irreducible media groups cascading one another. Each irreducible media group is modeled by any OCPN based model and thus is considered as a closed presentation unit. The firing rule of an irreducible media group requires completion of presentation of the previous media groups and availability of all objects in this media group for presentation. In other words, in addition to the firing rule specified by OCPN (Blakowski and Steinmetz, 1996), which basically focuses on the status of presentation of the objects preceding the transition, the firing rule of a media group in our GCOCPN model requires some "lookahead" mechanism to ensure that resource allocation is completed.

The group cascade OCPN model, which incorporates the mechanisms of resource allocation, resource scheduling, group pre-fetch and group waiting time for group synchronization is specified by the tuple $\{\Gamma, G, R, \Psi, D\}$ where:

- $\Gamma=\{T_1, T_2, ...T_n\}$, is a set of group transitions (bars).
- $G=\{G_1, G_2, ...G_n\}$, is a set of irreducible media groups. Each group G_i is defined as an OCPN based model (Little and Ghafoor, 1990) (circles).
- $R=\{R_1, R_2, ...R_j\}$, is a set of media source types for the entire presentation program.
- Ψ: $G\rightarrow$ {real number}, represents group waiting time as a mapping from the set of groups.
- D: $T\rightarrow$ {pointers of function} represents a set of functions to perform the algorithm of resource scheduling, allocation and pre-fetch as a mapping from group transition to a set of algorithm execution functions.

Group cascade OCPN model structurally forms a set of paired objects (T_i, G_i), which represents a group transition T_i followed by a group object G_i. Group transition T_i, for $i\neq 1$, provides a set of control functions for group synchronization at the boundary, including pre-fetch of media objects, resource allocation and scheduling for presentation, and firing of group i, whilst group object G_i can be any OCPN based model. An OCPN is a model of multimedia composition with respect to inter-media synchronization based on the logic of temporal intervals and timed Petri-nets, and is in particular composed of media objects (places) and transitions to furnish execution of a Petri-net. Fig. 6 depicts the proposed GCOCPN model.

Initially, T_i $(1\leq i\leq n)$ should be based on the data consumption rate of each media object of group G_i and the desired time schedule for media presentation to figure out a relationship between the bandwidth of communication channels and the buffer requirements to achieve the required QoS for presentation. Note that the available storage capacity of the presentation platform constrains the upper bound of buffer

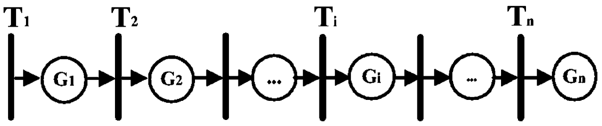


Fig. 6 A Group cascade object composition petri-net model

requirement of Eqs. (4-10). This indeed sets a bound for the communication bandwidth for data transmission from the remote servers. Negotiation for acquiring the resource of communication channel should be performed prior to the firing of the media group. In order to avoid possible loss of resource of communication channel during inter-group transition, the communication channel should be requested and reserved for the entire presentation and thus for all groups. It is therefore the responsibility of the initial transition, T_1, to negotiate and acquire the communication channel for the entire group. Once all the required resources, which may include possible pre-fetch of data streams from the remote servers, are available, T_1 fires group G_1 and therefore starts presentation. After firing group G_1, T_1 sends a group transition token to T_2. This ends process of T_1 activates T_2 to monitor the execution of group G_1 and in the mean time prepare data availability for group G_2, which include possible execution of pre-fetch of certain data objects. When all the tokens from group G_1 arrive and data streams are available to ensure inter-media and intra-media synchronization for group G_2, T_2 fires G_2 and then passes the group transition token to T_3. The same procedure repeats until the final group G_n is fired. User participation for backwarding or forwarding is allowed at the group level which requires the current active group transition T_i to quit all the preparation process and sends the group transition token to the group transition T_j of the designed group G_j, where $i \neq j$.

VI. EXPERIMENTAL RESULTS

In this section, the experimental results based on the approaches proposed in the previous sections are presented. We have implemented a time-line model based editing tool to construct a multimedia presentation program. Calculations of buffer occupancy versus a specified network bandwidth for resource allocation and reservation scheme for one particular media group as well as the whole multimedia information were also examined. A multimedia information system available in our experiment included three video clips and three audio clips as shown in Fig. 7, where the temporal information for each media object was pre-recorded. For simplicity, we specified the consumption rate of data streams

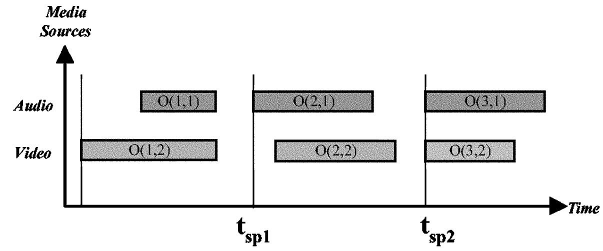


Fig. 7 Example of multimedia presentation group for experiment

with 176KB/s (11 KHz with 16-bit samples) for audio and 250KB/s (JPEG-encoded stream displayed at 24 frames per second) for video. The whole multimedia information system was then partitioned into three irreducible media groups at separation point t_{sp1} and t_{sp2} based on algorithm 4-1.

In our experiment, we fixed the network bandwidth allocated for audio to be 132 KB/s, and changed the bandwidth for video to a range between 140 KB/s to 180 KB/s. We then calculate the maximum buffer occupancy resulting from the approach with GCOCPN modeling and the approach without group partitioning implementation. Comparison of the results shows that our proposed GCOCPN scheme requires less buffer resource than non-GCOCPN schemes when the same communication resources are reserved as shown in Table 1. Further, when a given buffer size is reserved, the GCOCPN scheme requires less communication bandwidth than that required by non-GCOCPN schemes, if synchronization at the critical points is to be resolved.

VII. CONCLUSIONS

This paper presents a new model to incorporate characteristics of various media to facilitate multimedia presentation synchronization. The ultimate goal of this work is to resolve all possible delay problems at the presentation level such that each media object is always available and executable at the scheduled time instant. It ensures that the multimedia information can be presented by the specific QoS requirement. We first investigate a temporal relation called critical overlap resulting generally from a continuous media object overlapped with another media object. We then define the separation point by exploring the non-critical overlap from a discrete media object overlapped with another continuous media object. An algorithm for searching separation points is then developed to facilitate partition of a multimedia presentation program into manageable irreducible media groups that ensure the synchronization can be achieved. A modified OCPN model, called group cascade object composition Petri-net (GCOCPN), is thereafter proposed to characterize the special firing

Table 1 The xcomparison of maximum buffer occupancy using GCOCPN model

Network bandwidth (KB/s)		Maximum buffer occupancy	Maximum buffer occupancy (KB) without Group Partitioning	Maximum buffer occupancy (KB) with partitioned GCOCPN Model	Percentage of Buffer Saving
1	Video Audio	180.0 132.0	6140.0	6140.0	0.00
2	Video Audio	178.5 132.0	6211.4	6211.4	0.00
3	Video Audio	170.0 132.0	7160.0	6560.0	0.83
4	Video Audio	160.0 132.0	8460.0	7160.0	15.37
5	Video Audio	150.0 132.0	9760.0	7760.0	20.49
6	Video Audio	140.0 132.0	11060.0	8360.0	24.41

requirement for those objects with temporal relations of critical overlap.

A resources scheduling scheme for supporting the synchronous presentation of GCOCPN model is also introduced. The relations among buffer occupancy at the presentation platform, network bandwidth and object pre-fetch time for a single media object are examined. In particular, the result shows that lower network bandwidth requires higher buffer occupancy and earlier pre-fetch time. For multiple objects, a temporally overlapped pre-fetching in the same media source type requires that the pre-fetching of the preceding object should be shifted a step ahead and started earlier for avoiding the double bandwidth requirement in the overlapped interval, this will result in a higher requirement for buffer occupancy. A resource allocation taking buffer occupancy, network bandwidth and pre-fetch time into account is then examined for both irreducible media groups and the entire presentation program. Experiments are carried out to observe the performance of the proposed resource allocation algorithm. The experimental results show a better performance assessment of buffer utilization by using the GCOCPN model.

Further research on resource allocation is conducted to work out an efficient yet dynamic method to perform resource allocation and scheduling, especially when the traffic is heavy and thus communication resources are limited.

NOMENCLATURE

Bw	allocated network bandwidth
$B(t_{ps}, t_{cs}, t)$	buffer occupancy function
$B_{\max}$	maximum buffer requirement
$C(t_{cs}, t)$	consumption function
CM	continuous media
CP	critical point
D	a set of functions
DM	discrete media
$GCOCPN$	group cascade object composition Petri-net
G_k	multimedia Group k
MO	media object
$OCPN$	object composition Petri-net
$O(i, j)$	ith media object of jth media source type of a multimedia information group
$P(t_{ps}, t)$	production function
PSP	pseudo separation point
QoS	quality of services
Q	the set of all media objects of a multimedia information program
R	a set of media source type for the entire presentation program
SP	separation point
t_{cs}	consumption start time
t_{ce}	consumption end time
t_{ps}	pre-fetch start time
t_{pe}	pre-fetch end time
T_c	object playback interval
T_{pf}	obejct pre-fetch time
T_j	group transition
$\overline{t}_{ps}$	the minimum pre-fetch starting time
$\overline{T}_{pf}$	the minimum pre-fetch time
Γ	as set of group transition
Ψ	group waiting time
Ω	data consumption rate
$\Delta T_{i,i+1(j)}$	time difference between the pre-fetch ending time of object $O(i, j)$ and the pre-fetch start time of object $O(i+1, j)$
$\Theta_{k(t)}$	total required buffer occupancy function for media group k
$\Delta\tau_{k+1,\, k(j)}$	time difference between the pre-fetch ending time of the last object in group

$k-1$ and the pre-fetch starting time of the first object with the same media source type R_j in group k

REFERENCES

1. Akyildiz, I.F., and Yen, W., 1996, "Multimedia Group Synchronization Protocol for Integrated Services Networks," *IEEE Journal on Selected Areas in Communications*, Vol. 14, No. 1, pp. 162-173.
2. Allen, J.F., 1983, "Maintaining Knowledge about Temporal Intervals," *Communication of ACM*, Vol. 26, No. 11, pp. 832-843.
3. Blakowski, G., and Steinmetz, R., 1996, "A Media Synchronization Survey: Reference Model, Specification, and Case Studies," *IEEE Journal on Selected Areas in Communications*, Vol. 14, No. 1, pp. 5-35.
4. Fung, C.L., and Pong, M.C., 1994, "MOCS: an Object-Oriented Programming Model for Multimedia Object Communication and Synchroniza-tion," *Proceedings of the 14th International Conference on Distributed Computing Systems*, pp. 494-501.
5. Gemmell, J., and Christodoularkis, S., 1992, "Principles of Delay-Sensitive Multimedia Data Storage and Retrieval," *ACM Trans. On Inform. Syst.*, Vol. 10, pp. 51-99.
6. Hoepner, P., 1992, "Synchronizing the presentation of multimedia objects," *Computer Communications*, Vol. 15, No. 9, pp. 557-562.
7. Iino, M., Young, F.D., and Ghafoor, A., 1994, "An Object-Oriented Model for Spatio-Temporal Synchronization of Multimedia Information," *Proceedings of the International Conference on Multimedia Computing and Systems*, pp. 110-119.
8. Jixin, M., and Knight, B., 1994a, "A General Temporal Theory," *The Computer Journal*, Vol. 37, No. 2, pp. 114-123.
9. Jixin, M., Knight, B., and Petridis, M., 1994b, "A Revised Theory of Action and Time based on Intervals and Points," *The Computer Journal*, Vol. 37, No. 10, pp. 847-857.
10. Lin, I.H., Lee, B.H., and Wu, C.C., 1998a, "A Synchronization Model for Multimedia Presentation with Critical Overlap Avoidance," *Proceedings of the International Workshop on Multimedia Software Engineering*, pp. 40-47.
11. Lin, I.H., Lee, B.H., and Wu, C.C., 1998b, "An Object-Oriented Synchronization Model for Distributed Multimedia Presentation Systems," *Proceedings of the Fifth International Workshop on Distributed Multimedia Systems*, pp. 127-130.
12. Little, T.D.C., and Ghafoor A., 1990, "Synchronization and Storage Models for Multimedia Objects," *IEEE Journal on Selected Areas in Communications*, Vol. 8, No. 3, pp. 413-427.
13. Little, T.D.C., and Ghafoor, A., 1993, "Interval-Based Conceptual Models for Time-Dependent Multimedia Data," *IEEE Trans. On Knowledge and Data Engineering*, Vol. 5, No. 4, pp. 551-563.
14. Murata Tadao, 1989, "Petri Nets: Properties, Analysis and Applications," *Proceedings of the IEEE*, Vol. 77, No. 4, pp. 541-580.
15. Nicolaou, C., 1990, "An Architecture for Real-time Multimedia Communication Systems," *IEEE Journal on Selected Areas in Communications*, Vol. 8, No. 3, pp. 391-400.
16. Qazi, N.U., Woo, M., and Ghafoor, A., 1993, "A Synchronization and Communication Model for Distributed Multimedia Objects," *Proc. ACM Multimedia 93*, pp. 147-155.
17. Rangan, P.V., and Vin, H.M., 1993, "Efficient Storage Techniques for Digital Continuous Multimedia," *IEEE Trans. On Knowledge and Data Engineering*, Vol. 5, No. 4 , pp. 564-573.
18. Raghavan, S.V., Prabhakaran, B., and Tripathi, S.K., 1996, "Synchronization Representation and Traffic Source Modeling in Orchestrated Presentation," *IEEE Journal on Selected Areas in Communications*, Vol. 14, No. 1, pp. 104-113.
19. Yang, C.C., and Huang, J.H., 1996, "A Multimedia Synchronization Model and Its Implementation in Transport Protocols," *IEEE Journal on Selected Areas in Communications*, Vol. 14, No. 1, pp. 212-225.

Discussions of this paper may appear in the discussion section of a future issue. All discussions should be submitted to the Editor-in-Chief.

Manuscript Received: Apr. 16, 1999
Revision Received: Jun. 23, 2000
and Accepted: Jul. 13, 2000

多媒體物件展示之同步模式

林銀河　黎碧煌　吳傳嘉

國立臺灣科技大學電機工程系

摘　要

為提供分散式多媒體展示，除了媒體物件的同步現象必須維持外，物件預取、網路頻寬取得及緩衝器配置的相互關係，必須讓系統資源的配置獲得適當的協調。本論文提出在多媒體展示平台中，以物件導向模式處理媒體物件的時間性關係及研究相關資源配置的問題，使多媒體的展示，能經由組合媒體物件本身的特質及其時間性的重疊關係而分割為“可管理的”區段並至可備準執行狀態達成同步的需求。本文將首先探討各種媒體物件型態間的重疊現象，並定義同步問題中棘手的關鍵重疊（critical overlaps），其目標為確保關鍵重疊的物件能精確的在關鍵重疊點（critical point）同步啓動展示·文中接著引導組合媒體物件依媒體重疊現象分割為可管理區段的觀念，將多媒體展示資料分割至多個串接的不可分割的媒體群組（irreducible media group），同時探討每個展示媒體群組的資源排程、資料預取（pre-fetch）對緩衝區（buffer occupancy）的關係·據此本文建立一個群組串接的物件組合派區網路（group cascade object composition Petri-net, GCOCPN）模式，實現多媒體時間性同步展示的方案，並以實例說明緩衝器資源配置的效能。

關鍵詞：多媒體同步，時間性關係，多媒體群組，資源排程。