Synchronization in a Multimedia

A. Ghomari

Université d'Oran Es-Senia
Faculté des Sciences
Département d'Informatique
BP 1524, El-M'Naouer Oran 31000
ALGERIE
ghomari.abdelghani@univ-oran.dz

Abstract. The paper introduces a class of specialized temporal net models, called Multimedia P-Time Petri Net (MP-RdPT), to model the synchronization aspects of multimedia scenarios. Several types of synchronization are introduced, and a number of temporal relations between multimedia objects are presented. After translation of the derived MP-RdPT net to an equivalent t-time Petri net, it is claimed that the proposed approach can be used for verification of specifications by using the tool Tina.

Keywords: Multimedia synchronization, time Petri net, specification, analysis, verification.

1 Introduction

Multimedia refers to the presentation of collections of both static and dynamic data (i.e., data with natural time dependencies e.g., audio or video) in a specified order and time. Therefore, their mutual synchronization must assure a proper temporal order of presentation events. Multimedia synchronization can be defined as a mutual assignment of data items and time instants. These time instants may be known in advance (e.g., standard consumer data players) or they can be also results of some unknown function of time (event driven synchronization) or known with some limited accuracy (e.g., random network delays).

The modeling and the presentation of multimedia scenarios are challenges of multimedia applications. Multimedia scenarios are results of temporal composition and user interactions of multimedia objects in an application domain, and lot of works discussed this notion [18]. Temporal compositions consist in presenting multimedia objects which requires synchronization among different media.

Most of specification models are based on Allen's relations [3]. Allen defined seven basic relations between two temporal intervals. For example, a TV program starts at 9:00 pm, and finishes at 11:00 pm. The TV program can be considered as one of multimedia objects. In addition, "interval" is considered as a range from 9:00 pm to 11:00 pm, and "duration" as two hours. Allen's relations require this duration of the interval. Before designing the specification model, interval duration must be known. This means that multimedia database systems must determine duration of multimedia objects, because presentations are almost dependent of duration [19].

Our work focuses on scenarios that are a natural means of playing and modeling temporal composition relations in an application domain. In our approach, the creation of scenarios is based on extended temporal relations advocated by Allen [3].

The approach presented in this paper is based on the MP-RdPT net for modeling temporal constraints and user interactions where multimedia objects will be represented as places, while transitions of the Petri net will be used for synchronization between the objects. The approach provides the following benefits:

(1) The ability to deal with non-deterministic time intervals, e.g. objects with an unknown duration, objects whose reproduction can fail and objects that represent user interactions.

(2) The possibility of automatic detection of inconsistent synchronization conditions such as "A precedes B, B precedes C, C precedes A".

(3) A graphical notation to describe and simulate the presentation.

(4) An editor which abstracts the internal Petri net representation and allows the user to think in familiar terms such as "precedence" or "overlap".

(5) Automatic generation of a MP-RdPT net based on the previous temporal specification.

(6) Automatic analysis of the MP-RdPT properties: safeness, liveness, reversibility and consistency.

A first version of our approach [12], [13] considers multimedia objects of known or unknown duration and interactive relations, but doesn't consider dependency temporal relations between multimedia objects. This is the main difference between the first version of our approach and the second one that will be described in this paper.

In this paper, we highlight the following points: related work and background (Section 2), our scenario temporal specification (Section 3), a formal definition of the MP-RdPT net (section 4), multimedia scenario representation model (section 5) and analysis of multimedia scenarios using the tool Tina (section 6).

2 Related work and background

Existing temporal models for multimedia may be decomposed into two classes: instant-based and interval-based [26]. In instant-based models, the elementary units are instants in a time space. Each event in the model has its associated time instants. The time instants arranged according to some relations such as precede, simultaneous or after form complex multimedia presentations. An example of the instant-based approach is timeline, in which media objects are placed on several time axes called tracks, one per each media type. All events such as the beginning or the end of a segment are totally ordered on the timeline.

Several approaches support instant-based models such as Hy-Time [16] or [14], [10]. The model is well suited for temporal composition of media segments of known durations; however it falls short for unknown durations. Other authors have proposed to use relations between interval end points for temporal composition of multimedia (temporal point nets [6], MME [8]). However, their use is difficult and results in complicated, unstructured graphs. In addition to that, their use may led to an inconsistent specification in which contradictory conditions are specified for intervals. In this case, a verification algorithm (called sometimes a temporal formatter) must check for inconsistency [6].

Interval-based models consider elementary media entities as time intervals ordered according to some relations. Existing models are mainly based on the relations defined by Allen for expressing the knowledge about time [3]. Giving any two intervals, they can be arranged according to seven relations: before, meets, overlaps, finishes, during, starts, equals. However, using Allen's relations for multimedia composition faces several problems. First, the relations were designed to express existing relationships between intervals of fixed duration and not for specifying relationships that must be always satisfied even when interval durations are changed.

Another problem with the Allen relations is their descriptive character. They allow expression of an existing, a posteriori arrangement of intervals, but they do not express any causal or functional relation between intervals. So, the Allen relations can be useful for characterizing an existing, instantiated presentation (a presentation for which all start and termination instants of media segments are known).

The third problem with the Allen relations is related to inconsistent specifications that can be introduced to a multimedia presentation. Detecting inconsistent specification requires algorithms of complexity [O(N2)], where N is the number of intervals [3].

Many approaches are based on time interval. For example Little and Ghafoor proposes an OCPN model equivalent to Allen's relations [21]. They do not take into account possible unknown durations of intervals and to prepare an instantiated presentation (a presentation in which all interval end points are determined), they must traverse the tree of interval relations to get deadlines used to schedule the presentation.

King proposes a different formalism based on a temporal logic [20]. He shows how the Allen relations can be expressed using temporal logic formula. Although his formalism has solid mathematical bases, composition of multimedia presentations using declarative formula is awkward. Logic formulas do not correspond to the mental image that an author uses in conception. Moreover, to be useful, the formalism must be supported by a consistency checker and an interpreter to execute a given temporal specification. [11] develops a software architecture for multimedia object synchronization and communications called SAMOCS. The object-oriented database management system VODAK [1] supports temporal operations.

Courtiat and De Oliveria [8] presented a synchronization model for the formal description of multimedia documents. This model automatically translates the user formalization into a real-time LOTOS formal specification and verifies a multimedia document aiming to identify potential temporal inconsistencies. Described through a hierarchical model, multimedia documents allow incomplete timing. The model also represents user interaction and expresses a media object as one logical unit. The model provides a set of synchronization patterns, formal semantics, and a verification technique.

Blakowski and Steinmetz [5] recognized an event-based representation of a multimedia scenario as one of the four categories for modeling a multimedia presentation. Events are represented in the Hypermedia/Time-Based Structuring Language (HyTime) and Hypermedia Office Document Architecture (HyperODA). Events are defined in HyTime as presentations of media objects along with the playout specifications and finite coordinate system (FCS) coordinates. HyperODA events happen instantaneously and mainly correspond to start and end of media objects or timers.

All these approaches suffer from poor semantics conveyed by the events. Moreover, they don't provide any scheme for composition and consumption architectures. You'll find and interesting survey on authoring models and approaches elsewhere [17].

The interval-based models face some disadvantages. Firstly, the temporal relations are designed to specify relations between multimedia objects of determined duration, but they are not designed for specifying relations that are not explicitly determined by the user. Secondly, the temporal relations describe existing arrangement of multimedia objects, but do not describe dependency relations between multimedia objects. For example, x meets y means that the end of multimedia object x coincides with the end of multimedia object y, but it does not describe whether multimedia object x starts multimedia object y, or whether multimedia object y stops multimedia object x. So, the majority of current models are interesting for describing presentations in which all start and end instants of multimedia objects are determined and fixed, but they are not appropriate when the duration of multimedia objects is not fixed. Thirdly, the detection of inconsistent specifications that may be introduced into a multimedia presentation requires complex processes.

To resolve theses disadvantages, a recent approach [27], considered in some systems such as STORM [2], is proposed to allow temporal specification of dependency relations between multimedia objects of unknown duration. It defines a set of operators expressing causal relations between multimedia objects. It can be used to form nested multimedia object duration.

One disadvantage of this approach is that not all scenarios can be expressed by means of those operators. For example, the scenario presented below cannot be described, because of interleaved start and stop actions on parallel branches. Temporal point nets, such as [9] and [6] can describe such scenarios, however, the resulting graph become complex and difficult to modify.

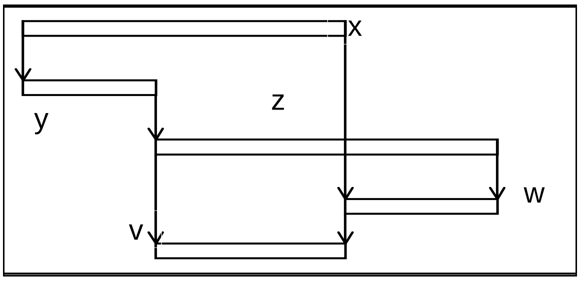

Fig. 1. Example of difficult scenario

Another disadvantage of this approach is its dependency aspects. It allows the expression of causal or dependent relation between multimedia objects. So, if a multimedia object fails, all the multimedia objects that depend on the failed multimedia object fail too. If x fails, the multimedia objects y, z, w, v that depend on the failed multimedia object x, fail too (see Fig. 1). For these reasons, we propose a model based on both time-interval and Weiss causal relations.

3 Our Scenario temporal specification

We will present a model for temporal composition of multimedia objects. The model is based on time-interval [3] and Weiss relations [27]. We consider the seven relations of Allen [3] (*equals, meets, finishes, starts, overlaps, during*) with the following features:

Firstly, the temporal relations are designed to specify relations between multimedia objects of both deterministic and non-deterministic duration. Secondly, the temporal relations describe both existing arrangement of multimedia objects, and dependency relations between multimedia objects. For example, x meets y means that the end of multimedia object x coincides with the end of multimedia object y, or it describes whether multimedia object x starts multimedia object y, or whether multimedia object y stops multimedia object x. Thirdly, the detection of inconsistent specification is not necessary.

3.1 Interval

Our elementary entities are time intervals. Time interval I is defined by the end points ($I_{begin} \leq I_{end}$) as $I = \{t \mid I_{begin} \leq t \leq I_{end}\}$. The duration of interval I is $d = I_{end} - I_{begin}$, and can be constant (e.g. 5 seconds), dependent on the intrinsic playing time of the medium (e.g. playing time of a video segment) or unspecified (e.g. user interaction or live feed). In this paper each interval corresponds to the presentation of one object (e.g. an image or a music selection). In that sense, the beginning and the end of an interval are logical times which will really correspond to physical time during the effective presentation to the user.

In our temporal specification language, an interval I is declared in this way: multimedia-object (min, opt, max): media-type, where min, opt and max are respectively the minimum, optimal and maximum admissible duration of the related interval.

3.2 Temporal relations

Several relationships have been defined on time intervals: *before, meet, equal, overlap, during, start, finish* [3]. Usually, they are binary relationships but can be easily extended to n-ary ones [22]. Sequential relationships combine intervals which share the same timeline (*mutual exclusion*), occurring one after the other with (*before*) or without delay (*meet*) between them. Parallel relationships relate intervals which have their own timeline. In our model these relations are used for composing and synchronizing multimedia objects in presentations.

3.3 Interactive relations specifications

Our approach synchronizes the scenario with the user (i.e. an expert of the application domain). The interaction takes the form of temporal interaction (*start, stop, pause, reverse,* and *forward*) and browsing interactions.

Temporal interactions concern user elementary operations such as *pause/resume, reverse* and *forward*. In *pause/resume* operations, the system records the current state of presentation modeled by a MP-RdPT net, and when resume operation is executed; the system loads the amount of time that the presentation had paused, and starts the presentation again from where it stopped. The *reverse* operation is specified in terms of temporal skip given by the user. Example "*goes back 15 minutes*".

When the *reverse* operation is requested, then the Petri net deals with objects associated with places currently being presented. If the *reverse* operation involves objects that are further behind a place Pi in the presentation graph, the presentation graph is traversed backward until the target object is reached. The *forward* operation is similar to the *reverse* operation.

Other approaches have been implemented for interactive movies by using the hypertext paradigm [7]. The essence of hypertext is a non-linear interconnection of information, unlike the sequential access of conventional text. Data is linked via cross-referencing between keywords to other parts of data. One hypertext called Petri Net-Based-Hypertext (PNBH) [24] describes data units as net places and links as net arcs. Transitions in PNBH indicate the navigation through relations.

In Fig.2 we present the Backus-Naur Form (BNF) of the grammar of our temporal specification language:

```
<Language>    ::=    scenario    <ScenarioName>    declarations
{<Declaration>}*assigns{<Assign>}*                  interactions
{<Interaction>}*  relations {<TemporalRelation>}*  end.
<ScenarioName> ::=  <Identifier>
<Declaration>  ::=  <MultimediaObject>     [(TimeInterval>)]   :
<Type>;
        |<ScenarioObject> : SCENARIO;
        |<ButtonObject> : BUTTON;
<MultimediaObject> ::=  <ObjectIdentifier>   /* OID */
<TimeInterval> ::=  [integer,integer] /* Time */
<Type> :: =audio | video | image | text | animation | button
<Assign> :: = assign (<MultimediaObject> , <PhysicalSupport>)
<Interaction> ::= interaction (ButtonObject, <ScenarioObject>)
<TemporalRelation>   ::=   [<MultimediaVariable>:=]{<Equal>    |
<Meet> | <Before> | <Start> | <During> | <Finis> | <Overlap> |
<par_min> | <par_max> | <par_master>}
<Equal> :: = equals (<MultimediaObjectParameter>);
<Meet> ::=  meets (<MultimediaObjectParameter>);
<Before> ::=  before (<MultimediaObjectParameter>,<Delay>);
<Starts> ::=  starts(<MultimediaObjectParameter>, <Delay>);
<During> ::=  during(<MultimediaObjectParameter>, <Delay>);
<Finish> ::= finishes(<MultimediaObjectParameter>, <Delay>);
<Overlap> ::= overlaps(<MultimediaObjectParameter>, <Delay>);
<par_min> ::= par_min(<MultimediaObjectParameter>);
<par_max> ::= par_max(<MultimediaObjectParameter>);
<par_master> ::= par_master(<MultimediaObjectParameter>);
<MultimediaObjectParameter> ::=<Interval> , <Interval>
<Interval>   ::=   <MultimediaObject>   |   <SenarioObject>   |
<ButtonObject> | <MultimediaVariable>
```

Fig.2. The BNF form of the grammar

4 Formal definition of the MP-RdPT net

Let $\Im$ be a temporal domain. A MP-RdPT on $\Im$ is a tuple (P, T, B, F, M0, IS, SYN, MP, R), where:

- (P, T, B, F, M_0) defines a Petri Net where P is a non empty finite set of places, T is a non empty finite set of transitions, with $P \cap T = \emptyset$, $B : P \times T \rightarrow N$ is the backward function, similarly, $F : R \times T \rightarrow N$ is the forward function, $M0 : P \rightarrow N$ is the initial marking.

As usual, we denote by:

$\{ p \in P \mid F(p, t) \geq 1 \}$ the set $\stackrel{def}{=}$ $\{p \in P \mid B(p, t) \geq 1 \}$ the set of ingoing places and $t\bullet \stackrel{def}{=} \bullet t$
$\{ t \in$ $\stackrel{def}{=}$ $\{ t \in T \mid F(p, t) \geq 1 \}$ and $p\bullet \stackrel{def}{=}$ of outgoing places of a transition t. Similarly, $\bullet p$
$T \mid B(p, t) \geq 1 \}$ are the sets of ingoing transitions and outgoing transitions of a place p.
The set of markings a MP-RdPT can reach from its initial marking Mo will be denoted as R (Mo).

- $\forall\, p \in P$, $\forall\, M \in R(Mo)$, $M(p) \leq 1$ (a MP-RdPT is safe),
- IS: is the static interval function

IS : $P \rightarrow (Q+ \cup 0)\ (Q+ \cup 0) \cup (Q+ \cup \infty)$, Such as : $\forall p \in P$: IS(p_i) = [a, n, b] with $0 \leq a \leq n \leq b$.
The IS function associate with each ingoing place a static validity time interval, where (a, n, b), associated with a place, represents respectively the earliest, nominal, and the latest firing times.

- SYN is the synchronization function that defines the firing rule associated to a transition, SYN: T $\rightarrow$ Rules, with Rules = def {strong_or, weak_and, master}, the set of synchronization rules. This synchronization semantics defines synchronization instants from a place statically or dynamically chosen. -MP is the function which indicates the master place of each transition from which the rule of transition requires a master, defined by : MP : Tmaster =def {t | SYN(t) = master} $\rightarrow \bullet$t,
If we note $\alpha = \{a_i \mid [a_i, b_i] \in IM\}$, $\beta = \{b_i \mid [a_i, b_i] \in IM\}$, then, according to the case of SYN (T), we consider that:

The *strong_or* synchronization rule is driven by the earliest stream. If either one of the two streams finishes, the other one has to stop, and [min (α), min (β)] is the sensibilisation interval.

The *weak-and* synchronization rule is driven by the latest stream. All the streams are presented completely, and [max (α), max (β)] is the sensibilisation interval.

The *master* synchronization rule is driven by the master stream. If two streams are presented simultaneously, when the higher priority stream finishes, the other has to stop. The multimedia continues after that, and [a_m, b_m] is the sensibilisation interval, with pm which indicates the master place. We define a_m, b_m by: let MP (t) = p_m and IM (p_m) = [a_m, b_m].
- R: P $\rightarrow$ {r_1, r_2 …}, a mapping from the set of places to a set of resources.

5 Multimedia scenarios representation model

Our approach is composed of a core and a series of functionalities which revolve around it (see Fig. 3). The core is a formal representation model built on the MP-RdPT model. As for the functionalities, they relate to the management of the temporal non determinism, the editing/creation of the scenarios, the presentation and the properties analysis of the scenarios.

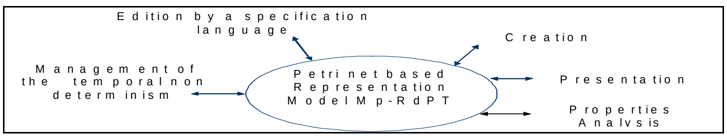

Fig. 3. The various elements of our approach.

5.1 Petri net generation

To create the MP-RdPT net, each temporal relation is associated with a Petri net as illustrated by [15], and modeled in several approaches, such as in OCPN [21]. This mapping is helpful for automatic creation of a MP-RdPT net. In fig 4, Tα, Tβ, Tδ model respectively the duration of places Pα, Pβ and Pδ.

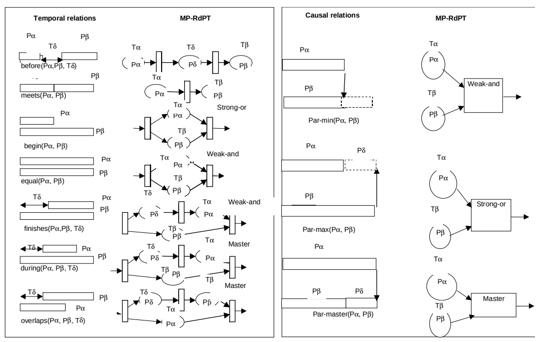

Fig. 4. MP-RdPT associated with temporal and causal relations

Before approaching these two stages, we present the two principles, inspired by those of [15], which guide the process of the Petri net creation. These principles are based on an association diagram between the temporal relations and the Petri nets (see Fig. 4).

5.2 Principles

Principle 1: For each temporal relation between two intervals, there is an equivalent Petri net model.

The Fig. 4 presents associations of temporal relations between intervals and the Petri nets. Each Petri net is composed of places representing the intervals. The delays used in the temporal relations like before, overlaps, during, finishes, are represented by places with validity time interval: [min, opt, max] = [delay, delay, delay].

Principle 2 is a generalization of principle 1.

Principle 2: A complex and arbitrary multimedia scenario, composed of temporal relations, can be built with Petri nets by replacing the temporal relations by the associated Petri nets. Principle 2 has guided to the development of the creation algorithm of the Petri net.

The creation of the Petri net starts at the end of the lexical, syntactic and semantics analysis of the editing program, if no error was detected.

5.3 t-Time Petri net

A t-time Petri net is a tuple (P, T, B, F, M0, IS), where:
(P, T, B, F, M0) defines a Petri net, and IS: $T \rightarrow Q+ \times (Q+ \cup \{\propto\})$ is the static time interval function. The application IS associate with each transition t of the net an interval with rational bounds IS(t) = [min, max] with $0 \leq min \leq max$, and max can be $\propto$. For further details, see [23].

5.4 Rules of translations

The created MP-RdPT net is then translated to an equivalent t-time Petri net for analyzing by the tool Tina [4]. For this, we use three rules of translation (see Fig. 5) inspired from [25]:

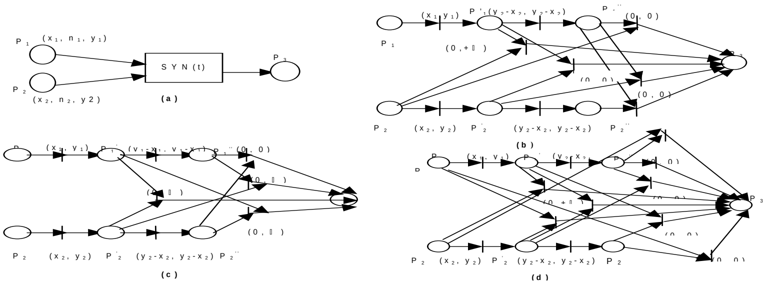

Fig. 5. Translation of an inter-stream synchronization schema of a MP-RdPT net (a) in the form of a t-time Petri net (b, c, d) according to inter-stream synchronization (b) transition of the type «master », (c) transition of the type « weak_and», (d) transition of the type « strong_or ».

6 Analysis of multimedia scenarios using the tool Tina

Tina (Time Petri Net Analyser) [4] is a software environment to edit and analyze Petri Net and t-time Petri Net [23]. In addition to the usual editing and analysis facilities of such environments (computation of marking reachability sets, coverability trees, semi-flows), Tina offers various abstract state spaces constructions that preserve specific classes of properties of the concrete state spaces of the nets. Classes of properties may be general properties (reachability properties, deadlock freeness, liveness), specific properties relying on the linear structure of the concrete space state properties relying on the linear concrete space state (linear time temporal logic properties).

After generating the t-time Petri net, the author investigates the scenario specification before it is delivered to the reader by using the analysis tool Tina. Currently, the following characteristics can be verified by the analysis tool: terminate state existence (i.e., if a state m exists in which non transitions are enabled), safeness (i.e., if every place has only one token), liveness (i.e., if blocking will never occur), reversibility (i.e., if the Petri net come back to its initial state whatever state it reaches), consistency (is a necessary condition for the reversibility that is a difficult property to establish.

7 Conclusion

Many existing specification models of multimedia temporal composition are based on Allen's relations. However, the current implementations of Allen's relations are not appropriate enough for some real world temporal compositions. The multimedia object duration must be known before designing the scenario, and any change in the duration may modify the temporal relations that exist between the multimedia objects. So, we proposed a temporal composition model based on an optional temporal duration. In our temporal specification, the user has the possibility to define a temporal specification which may be either relations depending on multimedia object duration or relations reflecting causal dependency between multimedia objects when the duration is unknown. Finally, a powerful Mp-RdPT model based on temporal specifications is used to specify multimedia scenarios. In addition, MP-RdPT is translated to an equivalent t-time Petri net for the analysis of multimedia scenarios properties by using the tool Tina. Tina allows the author to investigate the document specification before it is delivered to the reader.

References

1. K Aberer., and W Klas.: Supporting Temporal Multimedia Operators in Object- Oriented Database Systems, IEEE Intern Conference on Multimedia Computing and Systems, Boston, USA (1994).
2. M Adiba.: STORM, Structural and Temporal Object-oRiented Multimedia Database System. Blue Mountain Lake, Newark, (1995).
3. J.F Allen.: Maintaining Knowledge About Temporal Intervals, Commun. ACM, Vol. 26, N°. 11, 832-843 (1983)
4. B. Berthomieu B., P.-O. Ribet., F. Vernadat.: The tool TINA – Construction of Abstract State Spaces for Petri Nets and Time Petri Nets. International Journal of Production Research. Vol. 42. N°4, (2004). http://www.laas.fr/tina
5. G Blakowski and R Steinmetz.: A media Synchronization Survey: Reference Model Specification, and Case Studies, IEEE J. on Selected Areas in Comm, 14(1), 5-35. (1996).
6. M.C Buchanan., and P.T Zellweger.: Automatic Temporal Layout Mechanisms. In Proc. First ACM International Conference on Multimedia, Anaheim, CA, 341-350. (1993)
7. J Conklin.: Hypertext: An introduction and Survey, Computer, 17-41 (1987).

8. J-P Courtiat and R.C De Oliveira..: Proving Temporal Consistency in a New Multimedia Synchronization Model, Proc of ACM Multimedia 1996 Conf., ACM Press, New York, 141-152 (1996).
9. D Dingeldein.: Modeling Multimedia-Objects with MME. In Proc. Eurographics Workshop on Object-Oriented Graphics, Sintra, Portugal, (1994).
10. G.D Drapeau.: Synchronization in the MAEstro Multimedia-Authoring Environment. In Proc. First ACM International Conference on Multimedia, 331-339, Anaheim, CA, (1993).
11. C-L Fung.: A Software Architecture for Multimedia Object Communication and Synchronization, PhD Thesis in Computer Science, Hong Kong University, (1993).
12. A Ghomari., C Djeraba.: Towards a Timed-Petri Net Based Approach for the Synchronization of a Multimedia Scenario. In Proceedings of the 5th International Conference on Enterprise Information Systems, Angers, France, (2003).
13. A Ghomari., C Djeraba.: Synchronization of Multimedia Scenarios – A Case Study: Interactive News. In Proceedings of the 3rd International Workshop on Multimedia Data and Document Engineering (VLDB/MDDE) Berlin, Germany, (2003).
14. S Gibbs., C Breiteneder., and D Tsichritzis.: Audio/Video databases: An Object-Oriented Approach. In Proc. 9th IEEE Int.Data Engineering Conference, 381-390 (1993).
15. C.L Hamblin.: Instants and Intervals, In Proc of the 1st Int. Conf. Soc. For the Study Time, pages 324-331, Springer-Verlag, (1972).
16. ISO. Information Technology Hypermedia/Time-based Structuring Language (HyTime). ISO International Standard, (ISO/IEC IS 10744), (1992).
17. M Jourdan., N Layaida, C Roisin.: A Survey on Authoring Techniques for Temporal Scenarios of Multimedia Documents', Hanbook of Internet and Multimedia Systems and Applications, Part 1: Tools and Standars, CRC Press, London, (1998).
18. M.J. Perez-Luque and T.D.C. Little.: A Temporal Reference Framework for Multimedia Synchronization, IEEE Journal on Selected Areas in Communications (Special Issue: Synchronization Issues in Multimedia Communication), Vol. 14, No. 1, 36-51 (1996)
19. M Kaji.: Creating Multimedia Presentation based on Constraint Satisfaction Problems in Multimedia Databases. In Proc of the Intern Symposium on Database Applications in Non-Traditional Environments, Published in IEEE Computer Science, Kyoto, Japan. (1999)
20. P.R King.: Towards a Temporal Logic Based Formalism for Expressing Temporal Constraints in Multimedia Documents. Technical Report 942, LRI, Université de Paris-Sud, Orsay, France, (1994).
21. T.D.C Little., A Ghafoor.: Synchronization and Storage models for Multimedia Objects. IEEE Journal on Selected Areas in Communications, 8(3), 413-427, (1990).
22. T.D.C Little., A Ghafoor.: Interval-based Conceptual Models for Time-dependent Multimedia Data, IEEE Transactions on Knowledge and data Engineering, 5(4), (1993).
23. P.M. Merlin and D.J. Farber.: Recoverability of communication protocols: Implications of a theoretical study. IEEE Trans. Comm. 24(9): 1036 –1043, (1976).
24. P.D Stotts., R Furuta.: Temporal Hyper programming, Journal of Visual Languages and Computing, Vol. 1, 237-253, (1990).
25. P Senac.: Contribution to multimedia and hypermedia systems modeling. PhD thesis, Toulouse, France (1996).
26. T Wahl., K Rothermel.: Representing Time in Multimedia Systems. In Proc IEEE Intl. Conference on Multimedia Computing and Systems, Boston, MA, 538-543. (1994)
27. R Weiss., A Duda., D.K Gifford.: Composition and Search with a Video Algebra. IEEE Multimedia, 2(1): 12-25, (1995).