Journal of Applied Sciences

ISSN 1812-5654

science alert

ANSInet
an open access publisher
http://ansinet.com

Journal of Applied Sciences 6 (10): 2168-2176, 2006
ISSN 1812-5654
© 2006 Asian Network for Scientific Information

Specification of the Multimedia Documents Scenari by the Petri Networks and the Timed Automata

Zaoui Lynda, Guezzen Zakia, Abed Houaria, Belbachir Hafida
Department of Computer Science, University of Sciences and Technology, Mohamed Boudiaf, Oran, Algeria

Abstract: The multimedia documents design is today an application field which, with the growing complexity of the produced documents (graphic, video, audio animation…), requires an accurate synchronization of the various elements that build up these documents so as they comply with temporal constraints. The present study on multimedia documents rely primarily on specification of the temporal scenari since it is the new characteristic to be taken into account. The suggested techniques can be classified in two approaches: operational or imperative ones and declarative techniques through temporal relations (or with temporal constraints). In this field, one of the most used tools is the p-temporal Petri network. This choice is justified by its manner to model the majority of the temporal constraints intervening in the process of a multimedia documents edition and presentation system. The objective of this work is to study the hybrid application of the p-temporal Petri networks and the timed automata, as tools of temporal modeling of the scenari. The adopted approach consists of: Specifying the temporal scenario, This is done by the expression of the temporal relations between objects by the Allen relations, Modeling this specification by p-temporal Petri network, Modeling the behavior of p-temporal Petri network by a timed automata and analyzing qualitatively and quantitatively this automata

Key words: Multimedia documents, temporal scenario, p-temporal Petri network, timed automata, graph classes of states, translation to timed automata, coherence

INTRODUCTION

The multimedia documents made the object of several studies and research tasks in particularly on the new dimension: temporal dimension. These works can be classified in two categories depending one on the other: the Edition and Presentation.

The edition consists in defining a model to present the document (the p-temporal Petri networks model in our study), to specify how the various objects composing this document are connected in time and to edit it. As what concerns the presentation, it consists in restoring the document to the end-user while respecting the specification made in the edition.

Obviously we can neither present a document without publishing it, nor to publish it without being attentive to the form of its presentation. This dependence is linked to the temporal dimension of the media components. The introduction of this dimension generated new expression needs such as: objects duration, their temporal placements and their synchronization which define the temporal Scenario. Thus, the definition of a presentation format of the multimedia documents must allow the expression of the whole of this information to guarantee a coherent presentation of the document. This presentation requires certain logical, spatial and temporal constraints that must be satisfied. To do this, it will be necessary to find optimal scheduling elements constituting the document which will ensure a correct synchronization of all the document medias. The techniques proposed to specify the temporal scenari can be classified according the two approaches:

Operational techniques: Where the user, through tools, specifies the exact durations of its scenario. Four families of tools were identified: the timeliness (use of an absolute temporal axis), the graphs, the logic design and the programming languages. This approach includes several major disadvantages, like the low portability of documents, the maladjustment of the programming to the incremental nature of the edition process, the difficulty of the documents maintenance thus produced and finally, the problems which have the non_computers authors to master them.

Corresponding Author: Zaoui Lynda, Department of Computer Science, University of Sciences and Technology, Mohamed Boudiaf, Oran, Algeria

Declarative techniques: Based on the temporal constraints: where the author declares the temporal placements wished without being concerned with how they will be obtained. The phase of presentation starts by automatically transforming the declaration into an operational structure. This phase, called temporal formatting, calculates the moments of beginning and the durations of each element to be presented. This approach facilitates the task of the author, but present several problems which are mainly due, to the fact that the constraints considered can be precise, vague or dubious. The major problem is to be able to check temporal coherence: is there a solution, or did the author emit an impossible scenario (incoherent)? Then, a solution should be calculated, possibly optimal, which checks all the constraints. Ideally, it is also necessary to be able to sail between the various possible solutions.

In this context, the complexity of the temporal scenari became such as their design must be assisted by tools and formal methods. Many studies have been already realized in this field and temporal models have been proposed. These models are based on formal tools such as: Petri networks, logic, timed automata, LOTOS, systems conditions/actions, graph theory, operational research....

The approach that we carry out is a hybrid approach: operational and declarative, it consists in using tools of the operational class (temporal Petri network and timed automata) and declarative class (constraints on intervals). Two principal motivations are behind this application: The first is the advantages offered by an approach based on the constraints (Layaida, 1997; Fargier *et al*., 1998). Second is to associate the capacity of modeling of the temporal Petri networks (i.e., those which associate intervals of time to the transitions or to the places) with the power of analysis of the timed automata. Moreover, the tools available for the machine analysis of the timed automata can be used for the checking of the temporal Petri networks. The approach that we propose is divided into several steps and consists of:

- Temporal scenario specification by the expression of the temporal relations between objects of the document by the Allen's relations (relations based on intervals).
- The modelling of this specification by a p-temporal Petri network,
- The modelling of the behavior of p-temporal Petri network by a timed automata. The latter is obtained by an approach of translation of p-temporal Petri network to the timed automata.
- The qualitative and quantitative analysis of this automata.

TEMPORAL DIMENSION IN THE MULTIMEDIA DOCUMENTS

The major characteristic of a multimedia document is its temporal dimension. In this section, we explain, through an example, the various aspects related to this characteristic: course of a scenario, the temporal nature of the objects multimedia and various types of temporal scenari.

Course of a scenario: Whatever their granularity, the objects of a multimedia document are connected temporally so as to define a total order of presentation, i.e. the temporal scenario. The most intuitive and popular method to represent this order is to use a time axis. Figure 1 represents, in fact, the result of projection of a document on temporal dimension.

The course of this scenario is done from the left to the right while following the axis of time. Vertically, each line is reserved for a particular type of objects multimedia.

The temporal nature of the objects multimedia: A multimedia document can contain objects having the following temporal behaviors:

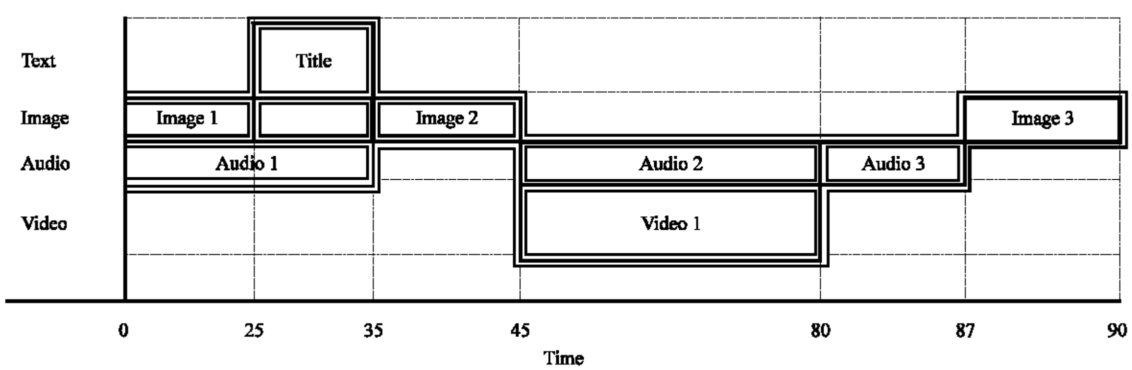


Fig. 1: Temporal scenario of the advertising spot document

- A timeless or discrete object: like the text and fixed images.
- A determinist or controllable temporal object: the presentation system guarantees its duration without loss of direction in a given environment.
- An indeterminist or uncontrollable temporal object: its duration is known only at the time of the presentation.

Types of temporal scenari: The temporal nature of the elements composing the multimedia document allows to determine the type of the scenario associated with this document. Thus, we can distinguish two types of scenari; each scenario has one or more traces of execution and this according to its type (Layaida, 1997):

Deterministic scenario: This type of scenario is associated with documents whose elements are of nature timeless and/or temporal determinist. Its execution trace is single.

Scenario indeterminist: In this case, the document contains at least an element of indeterminist nature. The indeterminist scenario is characterized by several traces of execution.

TEMPORAL MODEL OF THE MULTIMEDIA DOCUMENTS

Introduction: The temporal modeling, which objective is to introduce temporal models, is mainly interested with the specification, the analysis and the synthesis of the temporal scenari. At the specification level, we are interested in temporal modeling of the objects composing a scenario, the relations between these objects, in the interactions with the context of presentation and the problems which result from this (interactions of the user, of another documents, indeterminism…). At the analysis level, we are interested in the design of mechanisms allowing the checking of the specified scenario and the detection of possible temporal inconsistencies with the edition. At the synthesis level, we are interested in the design of mechanisms of temporal formatting.

Temporal models: The temporal model is a central aspect of the multimedia documents model. It allows to define the methods allowing the support of the temporal document structure. It must to report all the temporal dependents between the document components. The presentation and thus the temporal scheduling of the document, will rise from these dependents. It is this structure which determines the synchronization of the textual, sound and visual components of the document. Any temporal model for the edition and the multimedia presentation is composed of three parts:

- A temporal language: (Of the specification mechanisms) which allows the specification of a scenario.
- Mechanisms of analysis: Which allow to prevent the inconsistencies or to detect them while editing the document.
- Mechanisms of synthesis: Which allow statically or dynamically to produce a conform presentation to the specifications of the author.

Modeling of basic temporal information: A multimedia element can be handled through three essential temporal information: its moment of beginning, its duration of presentation and its moment of end. One of these three information is redundant. Indeed, it is possible to calculate one according to both others. But the choice of information selected forms part of the language and thus of the model, because the relationships of the elements is done from this information. The fact of choosing to describe an element with its moments of beginning and end or with its duration of presentation leads us to make a choice between two basic representations of a scenario: the representation based on moments and that based on intervals.

Representation based on moments: In this case, a multimedia element, that it is logical or basic, is described in the scenario by a moment of beginning and a moment of end, as in Firefly (Buchanan and Zellwegel, 1993) and Maestro (Drapeau, 1993).

Representation based on intervals: In this case, the course of the scenario is represented in order to connect the overall activities between them, for example, a video is presented at the same time that an audio sequence. Thus, a multimedia element is regarded as a temporal entity described by its presentation duration, as in OCPN (Little and Ghafoor, 1993) and CMIFED (Bulterman and Harman, 1995).

Modelling of the temporal relations: From the two types of preceding representations, two classes rise from temporal relations:

Relations based on moments: In the relations based on moments (Points Algebra noted PA), the temporal units considered in the relations are the moments of beginning and end of the elements. Being given two moments in a scenario, three relations can exist between them. A moment can precede another (<), succeed (>) to it or to be equal to it (=).

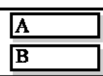
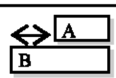
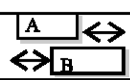
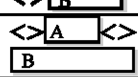
Relations based on intervals: In the relations based on intervals (Intervals Algebra noted IA), the possible relations between two elements multimedia are reduced to all the possible combinations of positioning of two intervals on a directed line. The most general model, suggested by Allen (1983), draws up the exhaustive list of all these relations.

These relations do not take into account the quantitative aspect of time, they are qualified by the qualitative ones. Shih *et al.* (1996) added a numerical parameter representing time to each Allen's relations (Table 1). The numerical parameters allow to define in a precise way the temporal difference between the presentation of two objects. Thus, quantitative relations based on intervals are defined.

With: a_i, $_j$ moment of beginning and of end of A; bi, bj moment of beginning and of end of B; dA (dB): duration of A (B).; t1, t2, t3 temporal parameters > = 0.

Modeling of the causal relations: In certain cases, it is necessary to be able to express scenari in which occurrence of a moment of end of an element causes the stop of another, even if the latter hasn't restored all its contents with the user. In the case of a representation based on intervals, three relevant relations of this type, called causal relations, were identified (Layaida, 1997).

Table 1: Quantitative relations based on intervals

Relation	Graph	Means
A equals B		$a_i = b_i < a_j$ et $a_j <= bj$ $d_A = d_B$
A starts (t) B		$a_i = b_i < a_j$ et $a_j < bj$ $d_B = d_A + t$
A finishes(t) B		$b_i < a_i < a_j$ et $a_j = bj$ $d_B = a\, d_A + t$
A overlaps (t_1, t_2) B		$a_i < b_i < a_j$ et $a_j < bj$
A during (t_1, t_2) B		$b_i < a_i < a_j$ et $a_j = bj$ $d_B = d_A + t_1 + t_{2v}$
A meets B		$a_i < b_i = a_j$, $a_j < bj$
A before(t) B		$< a_j < b_i < bj$

TEMPORAL MODEL ADOPTED OF THE DOCUMENT

In the approach that we carry out, we take as a starting point the Allen's relations (Allen, 1983) to build the model of our scenario (Table 2 and Fig. 2). This scenario is represented by a temporal Petri network. The places represent the media with their tolerated durations of presentation (dimin.: minimal duration and dmax: maximal duration). The transitions represent the passage of a state towards another in the scenario. Thus defined, a scenario can be represented by a whole of crossings of transitions allowing to pass from an initial marking M to a final marking M'.

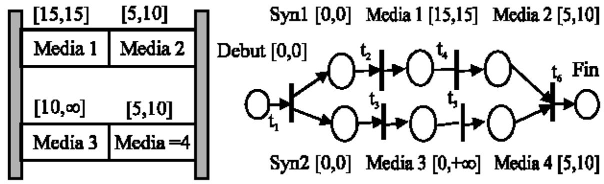


Fig. 2: Multimedia scenario and its p-temporal Petri net

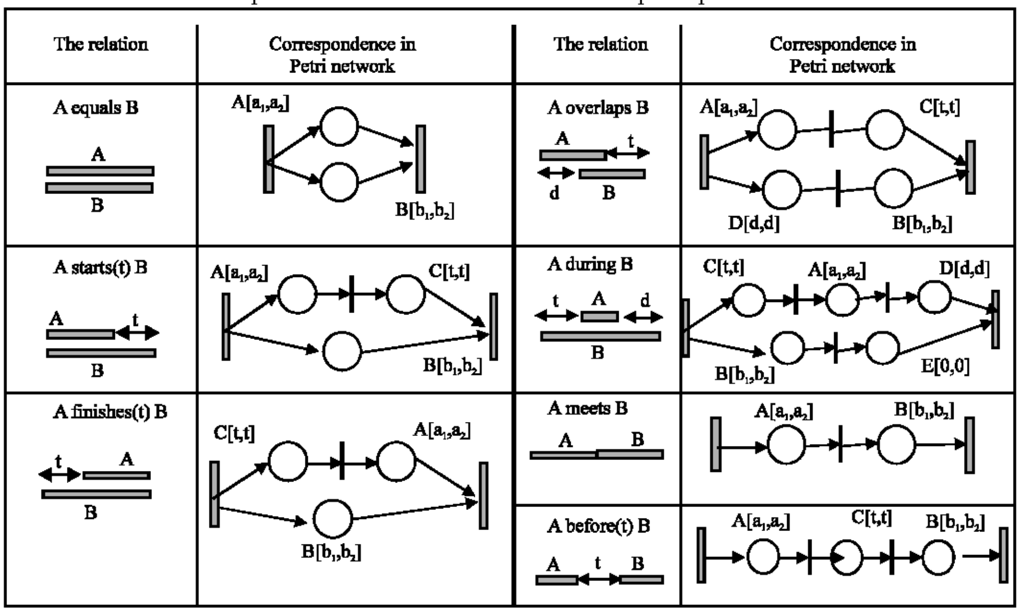
P-TEMPORAL PETRI NETWORKS AND TIMED AUTOMATA

Introduction: The approach, that we propose for the modeling of the scenario is inspired by work of Sava and Alla (2001a,b) and of (El-Kouhen, 1999), it associates the modeling capacity of the Petri network tool to the timed automata analysis power. More precisely, it consists in building temporally Bi-similar timed automata in p-temporal Petri network and in checking its properties with a tool such as kronos. The scenario analysis is carried out on the timed automata modelling the behavior of temporal Petri network obtained.

Construction of the states classes graph: The analysis approach by the states classes graph is a directed state approach, of enumeration analysis, adapted and often used in the analysis of temporal Petri networks (Berthomieul, 2001; Khansas, 1997). The qualitative analysis of temporal Petri networks can be done on the graph of the states classes (Berthomieu, 2001) and the quantitative analysis is done on the graph of the temporal constraints resulting from the graph of the states classes (Berthelot and Boucheneb, 1993). Other approaches were introduced, but directed event, which are based on the analysis of the succession of events (crossing of transitions) (Riviere *et al.*, 2002; Benlalem, 2004).

Definition: <<states classes>>: A state is accessible since the initial state by the shooting of a sequence of transitions (S). For each transition from this sequence, we

Table 2: Table of correspondence of Allen's relations of and p-temporal Petri networks



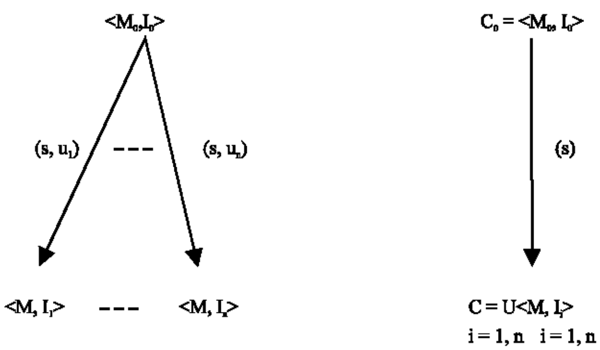


Fig. 3: Passage of the concept of state to the concept of states lass.

associate its moment of shooting (U). Let us consider the whole of the states accessible since the initial state by crossing from all the feasible values (u) corresponding to the same sequence of shooting (s): we then define a whole of states, which all are accessible since the initial state by crossing from (S). This whole of states defines the states class associated with the sequence (S). We can as follows schematize the passage of the concept of state to that of states class (Fig. 3):

Let us note that the C_0 class contains only the initial state and that the value of index N can reach the infinite one.

A states class will be then a pair C = (M, D) where:

M is the marking of the class and D the potential firing domain, the potential firing domain are characterized by a system of in equations of the form: $A\,\theta \geq B$ (1) where A is a matrix of coefficients, B a vector of coefficients and θ the vector of output dates variables of the places marks; θij corresponds to the mark j in place i.

Calculation of the following class: Let us suppose that the transition Ti is triable at the moment θT_i since the class C = (M, D). The Ti shooting at this moment generates a new class It = (M', D'). New marking is obtained to M' by: $\forall p \in P$: M' = M-Pre (p,t)+Post(p, t)

The new field D' is calculated from D by adding to the system (1) the conditions of Ti shooting, while expressing that θTi is lower or equal to all them θ_{ij} associated the other marks not taking part in the crossing of the transition while defining the new potential firing domains from the marks which did not change place during crossing of the transition and by associating a variable for each mark moved or created by crossing of the transition.

Construction of the states classes graph: The shooting rule application between the states classes allows generating a states classes graph starting from a given initial state. The root is the initial class, for each transition triable Ti in the class C a new class C' is created. A directed arc, labeled by Ti and its interval of shooting, east creates C towards C'. The new class will be tested if it is a class dead marks or if no transition is triable since this class, the branch exploration is finished. In addition, in order to decrease the combinative explosion risk, each new class is compared with those already produced to test a possible equality in which case the branch exploration is abandoned: a possible repetitive sequence appears.

Using this principle, the states classes graph will be obtained by regrouping of the equal classes. The existence of a way in the graph connecting the initial class to the class C proves that there is a realizable shooting bill book, supported by this way.

Timed automata: The timed automata were proposed by Alur and Dill (1994) in order to make it possible to model systems which combine evolutions discrete and continuous and where the deadlines separating two actions must be taken into account explicitly. Since the introduction of the model timed automata, several logics were proposed for the study (analyzes) of this model and several teams devoted significant efforts to the model-checking of these logics, which led, in the years 1995-1996, with the development of the temporized model-checkers, they are tools (software) for automatic checking (Yovine, 1998).

One of the main problems of the automatic checking is that of the atteignability of a whole of the system states. Indeed, the majority of the other problems can be reduced to this one.

Description: A timed automata is composed of an automat in finished states and expression of continuous time (allowing specifying systems real time). This formalism makes it possible to introduce temporal constraints via variables (with actual values, positive or null) called clock.

Formal definition: Finally, a timed automata is a 6-uplet $A = (L, L_0, X, \sum, I, \tau)$ (Allur and Diel, 1994), where:
L is the finished whole of the peaks; $L0 \subset L$ is the initial peak; X is the finished whole of the clocks; $\sum$ is a whole of symbols;
I is an application which associates an invariant of the I(L_m peak) each L_m peak $\in S$;
τ is the whole of the transitions.
A transition is a 5-uplet (L_m,a, $G_{m, m+1}$, $A_{m, m+1}$, L_{m+1}), where:
L_m is the peak source;
$a \in \sum$ is a symbol associated with an event;
$G_{m, m+1}$ the condition of crossing, i.e. guard;
$A_{m, m+1}$ is the assignment
L_{m+1} is the peak of destination.

The state of timed automata is defined by the couple (L_m, v), where L_m indicates the peak and v is an evaluation of clock which checks the invariant of I (L_m top).

States attainability analysis: In timed automata, we can remain in a peak, as long as the associated invariant is satisfied by the value of the clocks. Then a peak can not correspond to one state but to a space of states. We call area in a peak L_m and we note it (L_m, Q_n) a space of Q_n clock in a L_m peak. Generally, the study of a system modelled by a timed automata is based on the automata states attainability analysis. To know if an area Q is atteignable since a Q_{init} area, two methods can be used (Sava, 2001).

The first method called the analysis ahead is based on calculation of all the states space which can be reached since of the states belonging to the Q_{init} area. The whole of these states is called successor of the Q_{init} area. If calculated space contains states which also belong to the area Q, then we can conclude that this area is attainable since Q_{init}.

The second method, called behind analysis method is based on calculation of the unit of all the states from which we can reach states of the area Q. The whole of these states is called predecessor of the area Q. If calculated space then contains states which also belong to the Q_{init} area, we can conclude that the area Q is attainable since Q_{init}.

Principle of the p-temporal Petri network passage to the timed automata: One of the difficulties which does raise this approach is the passage of the p-temporal Petri network model to the timed automata model, i.e., how to translate p-temporal Petri network by a timed automata? Many studies were carried out on t-temporal the Petri network passage with the timed automata (Lim, 2003) (Sava, 2001) but no work approached the passage of the p-temporal Petri network model to model timed automata. For our part the principle of the approach that we propose consists to build the states classes graph associated with p-temporal Petri network and to translate this graph into a timed automata.

The algorithm that we will describe is based on the algorithms proposed by Khansa (1997) and Sava (2001). We will use the following notations:

L indicates the whole of the peaks
T indicates the whole of the transitions from the automata
C is the whole of the states classes of p-temporal Petri network

Initially, all these sets are empty and we initialize to zero a meter N. To each transition T from the states classes graph of p-temporal Petri network, we associate a clock X.

Step 1: Initialization: Either C_0 the initial class of p-temporal Petri network and L_0 the associated peak with

C_0. We determine the transitions validated by the class C_0, the active clocks in the L_0 peak. We calculate invariant I(L0) L0 peak from the transitions crossing intervals by C_0. We memorize this visit of the peak of L0 in the pile: C: = {C_0}; L: = {L_0}; N: = 0.

Step 2: Analysis of the last visit memorized in the pile: Let us suppose that it act at obtaining the peak in by the transition crossing $t_{m,n}$, we remove of the pile the element which memorizes this visit (the last element), for each transition tj we determine the class C_{n+1} of the graph reached by its crossing.
If $C_{n+1} \notin C$ then:

- Create an associated peak L_{n+1} with the class C_{n+1} we determine the transitions validated by the class C_{n+1},the active clocks in the peak L_{n+1} and we calculate the invariant $I(L_{n+1})$ peak L_{n+1} starting from the intervals of transitions crossing validated by C_{n+1}.
- Create a transition $t_{n,\ n+1} = (L_n,\ g_{n,\ n+1},\ An_{,\ n+1},\ L_{n+1})$ which models the crossing of tj. We add its guard, identical to the interval of crossing of tj and its assignment $A_{n,\ n+1}$ which puts at zero the clocks associated with the lately validated transitions
- Memorize in the pile this visit of the peak L_{n+1} and
- Bring up to date the sets: C = C U {C_{n+1}}; L = L U {L_{n+1}}; n = n+1.

Else the automata evolution since L_n was already analysed for these values of the clocks.

Step 3: If P ≠ ∅, there remain visits of peaks to be analyzed. Go to Step 2.

Step 4: end

IMPLEMENTATION

We have carried out software which implements the various stages mentioned in the preceding paragraphs. This prototype is developed in DELPHI language 5 under windows 98, (Fig. 4), it implements the following tasks:

- Specification of the temporal scenari by the p-temporal Petri networks.
- Translation of the networks obtained into timed automata.
- The checking of qualitative and quantitative coherence.

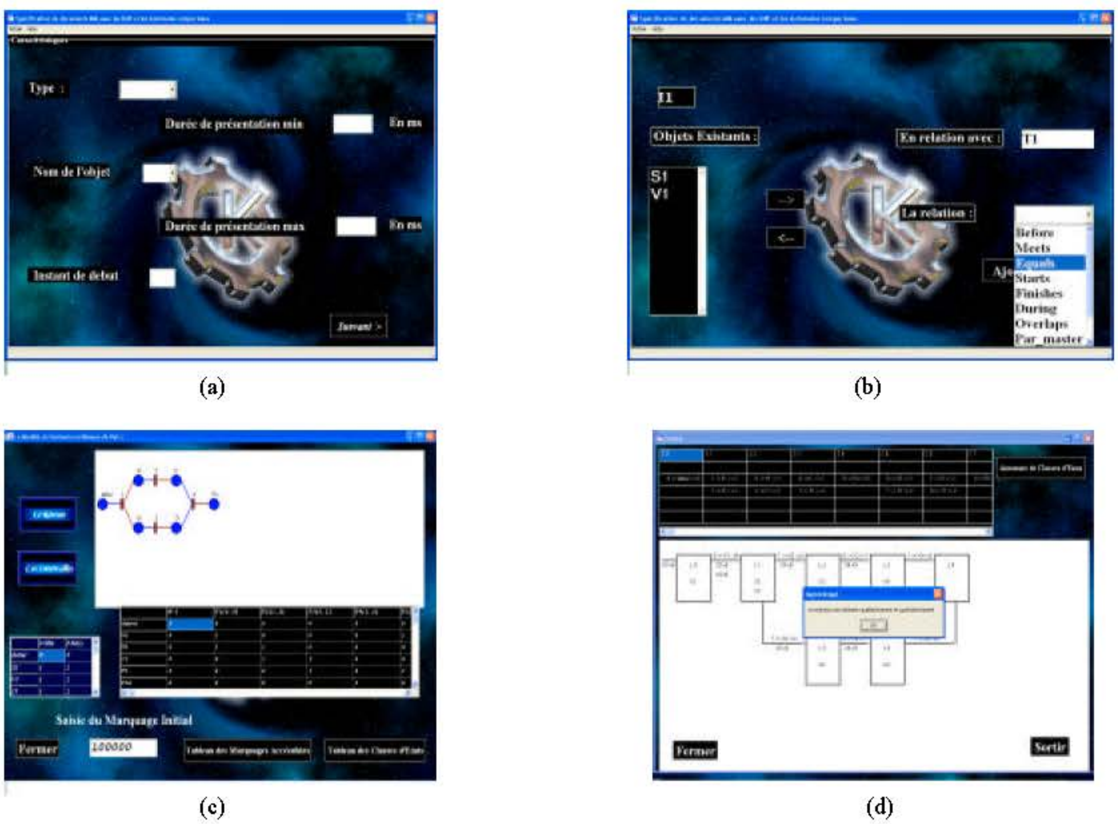


Fig. 4: Interfaces of the software. (a) Window 1: Define each object (name, type, d_{max}, d_{min}, beginning moment) (b) window 2: Input of the temporal relations between two objets (c) window 3: Visualization of the associated p-temporal Petri network (d) window 4: Represent the states classes and the timed automata with its analysis results

Scenario specification: The scenario modelling is done by the construction of the p-temporal Petri network corresponding. By knowing the total number N of the simple objects composing the document like their types, their names, their moments of beginning, their minimal and maximum durations and the temporal relations connecting them, we make a sorting of the objects according to their moments of beginning. For each object, we draw a place by associating its interval to him. Finally, we add the places of synchronization necessary.

Analyze and checking of qualitative and quantitative coherence: The algorithms of construction of the states classes graph and the translation of p-temporal petri network described in states classes graph and principle of the P-temporal petri network passage to the timed automata were implemented. The qualitative and quantitative coherence of the scenario is proven by the course of the graph of the automat by the method of analysis ahead. When the diagram obtained of the translation gives a general treatment of all the cases (not case of failure), this implies that the scenario is coherent.

CONCLUSIONS

The multimedia documents constitute a relevant evolution in the electronic documents field. They are characterized by alternative nature of their contents: statics (text, graph, image) or dynamics (audio, video, animation), also by a temporal organization of their components.

The documents design including multimedia objects (image, sound, video…) organized in this dimension is a complex task because the entity to be built is a dynamic entity of which should be specified the temporal scenario. The current work in this field especially concerns the temporal scenari specification since it is the new characteristic to take into account. Various techniques were proposed to specify a temporal scenario which we can classify in two approaches: operational or imperative and declarative techniques by temporal relations (or with temporal constraints). We draw an important conclusion from these works: the field is still in its stammerings (slowness in the emergence of standards actually accepted by the profession). This is not only due to the youth of the field, but also to the complexity of the authors needs (simple and intuitive specification, iterative and incremental specification…) to be covered and to the problems complexity (associates to these needs and the temporal nature of the treated objects) to be solved. These complexities directed the works towards the use of the declarative approaches based on constraints for the durations and the temporal relations expressions (between the media objects) intervening in a temporal scenario and the use of formal tools for the modeling and the formal analysis of these scenari.

Another remark is that the problems of temporal modeling and analysis set up in the multi-media documents field are found in other fields where time plays a paramount role. The approach suggested in this article is based on this orientation and takes as a starting point the work carried out in other fields. More particularly, those carried out in the field of the synthesis of the order of the SEDT (Systems with Time-lag Discrete Events).

REFERENCES

Allen, J.F., 1983. Maintaining knowledge about temporal intervals. Commun. ACM, 26: 832-843.

Alur, R. and D.L. Dill, 1994. A theory of timed automata, Theoretical Comput. Sci., 126: 183-235.

Benlalem, Z., 2004. Specification of multimedia documents schedulings by the Petri networks and linear logic. M.Sc Thesis, University USTO Oran.

Berthelot, G. and H. Boucheneb, 1993. Towards a Simplified Building of Time Petri Nets Reachability Graph. In 5th International Workshop on Petri Nets and Performance Models (PNPM 93).

Berthomieu. B., 2001. Temporal analysis of Petri nets: The States Classes method of and its implementation. LAAS-CNRS.

Bulterman, D.C.A. and L. Hardman, 1995. Multimedia authoring tools: State of the art and research challenges. In Computer Science Today. Recent Trends and Developements LNCS, Springer-Verlag, 1000: 575-591.

Buchanam, C. and P.T. Zellweger, 1993. Automatic temporal layout mechanisms. Proceedings of the 1st ACM International Conference on Multimedia, ACM Press, Anaheim, Californie, pp: 341-350.

Drapeau, G.D., 1993. Synchronization in the MAEstro multimedia authoring environment. Proceedings of the 1st ACM Conference on Multimedia, ACM Press, Anaheim, Californie, pp: 331-340.

El Kouhen, N., 1999. Order supervised systems with temporized discrete events. Ph.D Thesis, Laboratoire of Automatic and industrial Automatic Ecole Mohammadia of Engineers, Rabat.

Fargier, H., M. Jourdan, N. Layaïda and T. Vidal, 1998. Using temporal constraints networks to manage temporal scenario of multimedia documents. ECAI 98 Workshop on Spatial and Temporal Reasoning, Brighton, UK.

Khansa, W., 1997. P-temporal Petri nets: Contribution to the study of the systems with discrete events. Ph.D Thesis, University of Savoy, Annecy, France.

Layaïda, N., 1997. Madeus: System of edition and presentation of structured multimedia documents. Ph.D Thesis UJF-Grenoble 1.

Lime, D., 2003. T-temporal Petri nets and timed automata. Institute of Research in Communications and Cybernetics of Nantes (IRCCyN). CNRS UMR 6597, Central School.

Little, T.D.C. and A. Ghafoor, 1993. Interval-based conceptual models for time-dependent multimedia data. IEEE Transactions on Knowledge and Data Engineering (Special Issue: Multimedia Information Systems), 5: 551-563.

Riviere, N., B. Pradin-chezalviel and R. Valette, 2002 Propagation of Constraints and Multimedia Documents Scheduling, FAC' 2002.

Sava, A.T., 2001a. On the synthesis of the order of the systems with temporized discrete events. Ph.D Thesis, INP-Grenoble.

Sava, A.T. and H. Alla, 2001b. Order by supervision of the systems with temporized discrete events, Modelling of the reactive systems (MSR 2001), pp: 71-86.

Sava, A.T. and H. Alla, 2001. Combining hybrid Petri Networks and hybrid automata. IEEE Transactions one Robotics and Automation, 17: 670-678.

Shih, T.K., L.J. Hwang and J.Y. Tsai, 1996. Formal Model of Temporal Properties Underlying Multimedia Presentations. In Multimedia Modeling: Towards the Information Superhighway. Courtiat, J.P., M. Diaz and P. Sénac (Eds.), World Scientific, pp: 135-150.

Yovine, S., 1998. Model Cheking Timed Automata. Embedded Systems. Rosenberg, G. and F. Vaandrager Eds., LNCS, 1494.