A Much Compact Abstraction of the State Space of Real time Preemptive Systems

Abdelkrim Abdelli
USTHB University - Computer Science Department- LSI laboratory
Abdelli@lsi-usthb.dz

Abstract

Preemptive Systems are systems whose tasks are timely constrained and which can be suspended for a while and resumed afterwards. In order to check for their reliability, formal methods are applied to model and to analyze their behaviors. This is achieved by computing their state spaces that can be abstracted and encoded as graphs. We present in this paper an algorithm allowing an efficient computation of a DBM over-approximation of the state space of preemptive systems modeled by using Time Petri Nets with inhibitor arcs. For this effect, each class of this graph is expressed as a pair (M, $\mathbb{D}$), where M is a marking and $\mathbb{D}$ is the system of DBM inequalities. In [1] we have defined an algorithm to compute the system $\mathbb{D}$ straightforwardly in its normal form, without requiring computing the intermediary polyhedron. We explore for this abstraction a suitable equivalence relation that contracts yet more the graphs. Experimental results comparing our algorithm with other approaches are reported.

Keywords: *Preemptive systems, Time Petri nets, Inhibitors arcs, State class graph, DBM.*

1. Introduction

Nowadays, the correctness proofs of real-time preemptive systems are demanding much theory regarding their increasing complexity. The execution of such systems admits that a task may be stopped for a while and later resumed at the same point. This notion of suspension requires extending the semantics of traditional timed models so that to handle such behaviors. This has led to introduce the stopwatch mechanism [9] and hence many models have been defined, as for instance, hybrid automata (LHA) [3]. Time Petri nets (TPN) [13] have also been considered in several works including Preemptive-TPN [8], Stopwatch-TPN [6], Inhibitor-TPN [15], and Scheduling-TPN [12].

The verification of qualitative and quantitative properties of such systems on their formal description involves the investigation of a part of or the whole set of their reachable states that determine their state space. As the state space is generally infinite due to dense time semantics, we need therefore to compute finite abstractions of it that preserve properties of interest. In these abstractions, states are grouped together, in order to obtain a finite number of these groups. These groups of states are, for instance, regions and zones for timed automata, or state classes [6] for time Petri nets. Hence, the states pertaining to each group can be described by a system of linear inequalities, noted *D*, whose set of solutions determines the state space of the group. Hence, if the model does not use any stopwatch, then *D* is of a particular form, called *DBM* (Difference Bound Matrix) [10].

However, when dealing with stopwatches, the system *D* becomes more complex and does not fit anymore into a *DBM*. In actual fact, *D* takes a general polyhedral form whose canonical form is given as a conjunction of two subsystems $D = \mathbb{D} \wedge \hat{D}$, where $\mathbb{D}$ is a DBM

system and $\hat{D}$ is a system of general inequalities that cannot be encoded with DBMs. The major shortcoming of manipulating polyhedra is the performance loss in terms of computation speed and memory usage. Indeed, the complexity of solving a general polyhedral system is exponential in the worst case, while it is polynomial for a DBM system.

In order to speed up the state space computation, the idea is to leave out the subsystem $\hat{D}$, to keep only the system $\not{D}$ approximating thus the space of D to the DBM containing it. The obvious consequence of over approximation is that we add states in the computed group that are not reachable indeed. However, since the graph thus computed encompasses the exact one, the properties preserved therein are limited to safety.

For this effect, many algorithms have been proposed to compute the tightest DBM overapproximation [2] [8] [15]. For instance, in [2] the authors have proposed an efficient algorithm that reduces sensibly the computation effort of a class by avoiding the manipulation of the intermediary polyhedron. By the way, the implementation of this algorithm has been proved to be more reliable than that of other approaches [16] [17] [18].

Moreover, they showed in another work [1] that by relaxing a bit in the precision of the constraints of the system $\not{D}$ they achieve to compute smaller graphs and with lesser expenses. We consider in this paper real time preemptive systems modeled by using *ITPN* (Time Petri Nets with inhibitor arcs) [15]. This model extends *TPN* to inhibitor arcs to control the activation and the suspension of stopwatches.

So that to improve yet more this graph construction introduced [1] we provide a new equivalence relation to contract the graph that we prove to be a bisimulation to class equality. However, although this abstraction may derive still more of additional sequences comparatively to the tightest DBM overapproximation [8] [2] [15], it preserves all the firing sequences of the model and may be sufficient to model checking the properties of the *ITPN*.

The remainder of this paper is organized as follows: In Section 2 we present the syntax and the formal semantics of the *ITPN* model. In section 3 we introduce our approach. In Section 4 we discuss some experimental results that compare the performances of our algorithm with those of other approaches.

2. Time Petri nets with Inhibitor Arcs

Time Petri nets with inhibitor arcs (*ITPN*) [15] extends time Petri nets [13] to Stopwatch inhibitor arcs. Formally, an *ITPN* is defined as follows:

Definition .1 *An ITPN is given by the tuple (P,T,B,F,M^0, I, IH) where: P and T are respectively two nonempty sets of places and transitions; B is the backward function: $B:P\times T\rightarrow \mathbb{N}=\{0,1,2,..\}$; F is the forward function $F:P\times T\rightarrow \mathbb{N}$; M^0 is the initial marking mapping[1] $M^0:P\rightarrow \mathbb{N}$; I is the delay mapping $I:T\rightarrow \mathbb{Q}^+\times \mathbb{Q}^+\cup\{\infty\}$, where $\mathbb{Q}^+$ is set of non negative rational. We write $I(t)=[tmin(t),tmax(t)]$ such that $0\leq tmin(t)\leq tmax(t)$; $IH:P\times T\rightarrow \mathbb{N}$ is the inhibitor arc function; there is an inhibitor arc connecting the place p to the transition t, if $IH(p,t)\neq 0$.*

For instance, let us consider the *ITPN* model shown in Figure 1, already presented in [15]. Therein, the inhibitor arc is the arc ended by a circle that connects the place p_7 to the transition t_3. Initially, the place p_3 is marked but not the place p_7; hence t_3 is enabled but not inhibited. Therefore, t_3 is progressing as t_4 which is also enabled for the initial marking.

[1] $\mathbb{N}$ denotes the set of positive integers. In the graphical representation, we represent only arcs of non null valuation, and those valued 1 are implicit.

However, the firing of the transition t_4 consumes the token in the place p_4 and produces one in p_2 and another one in p_7. Therefore, the inhibitor arc becomes activated and hence the clock of t_3 is suspended (t_3 is thus inhibited), and its suspension will last as long as p_7 remains marked.

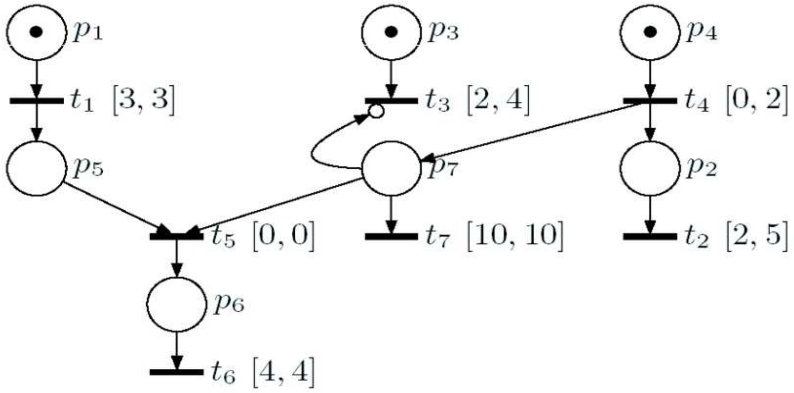

Figure. 1 An ITPN Model.

The formal semantics of the *ITPN* model is introduced next. Let $RT:=(P,T,B,F,M^0,I,IH)$ be an *ITPN*:

- We call a marking the mapping, noted *M*, which associates with each place a number of tokens: $M:P \rightarrow \mathbb{N}$.
- A transition *t* is said to be enabled for the marking M, if $\forall p \in P, B(p,t) \leq M(p)$. We denote by $Te(M)$ the set of transitions enabled for *M*.
- A transition *t* is said to be inhibited for a marking *M*, if it is enabled and if there exists an inhibitor arc connected to *t*, such that the marking satisfies its valuation $(t \in Te(M)) \wedge \exists p \in P,\ 0<IH(p,t) \leq M(p)$. We denote by $Ti(M)$ the set of transitions that are inhibited for the marking *M*.
- A transition *t* is said to be activated for a marking *M*, if it is enabled and not inhibited, $(t \in Te(M)) \wedge (t \notin Ti(M))$; we denote by $Ta(M)$ the set of transitions that are activated for the marking *M*.
- Let *M* be a marking ; two transitions t_i and t_j enabled for *M* are said to be conflicting for *M*, if $\exists p \in P\ B(p,t_i)+B(p,t_j)>M(p)$.We denote by $Conf(M)$ the relation built on $Te(M)^2$ such that $(t_1,t_2) \in Conf(M)$, iff t_1 and t_2 are in conflict for *M*.

For instance, let us consider again the *ITPN* of Figure 1; its initial marking is equal to $M^0:\{p_1, p_3, p_4\} \rightarrow 1;\{p_2, p_5, p_6, p_7\} \rightarrow 0$; the sets of enabled, inhibited, and activated transitions for M^0 are respectively $Te(M^0)=\{t_1\}$, $Ti(M^0)=\varnothing$, and $Ta(M^0)=Te(M^0)$.

Remark: We assume in the sequel that the *ITPN* are safe, which means that no transition can be enabled more than once for any given marking.

We define the semantics of an *ITPN* by as follows:

Definition.2 *The semantics of an ITPN is defined as a LTS (labelled transition system), $ST=(\Gamma, e^0, \rightarrow)$, such that:*

- *Γ is the set of accessible states: Each state, noted e, pertaining to Γ is a pair (M, V) where M is a marking and V is a valuation function that associates with each enabled transition t of Te(M) a time interval that gives the range of relative times within which t can be fired. Formally we have: $\forall t \in Te(M)$, $V(t):=[x(t),y(t)]$*
- *$e^0=(M^0,V^0)$ is the initial state, such that: $\forall t \in Te(M^0)$, $V^0(t):=I(t):=[tmin(t),tmax(t)]$.*
- *$\rightarrow \in \Gamma \times (T \times \mathbb{Q}^+) \times \Gamma$ is a relation between states, such that $((M,V),(t_f, \underline{t_f}),(M',V')) \in \rightarrow$, iff:*
  $(t_f \in Ta(M)) \wedge x(t_f) \leq \underline{t_f} \leq MIN_{t \in Ta(M)}\ y(t)$; and we have:
  $\forall p \in P,\ M'(p):=M(p)-B(p,t_f)+F(p,t_f)$.
  $\forall t \in Te(M')$
  if $t \notin New(M')$:
  $[x'(t), y'(t)]:= [MAX(0, x(t)-\underline{t_f}),\ y(t)-\underline{t_f}]$ $\quad t \in Ta(M)$
  $[x'(t), y'(t)]:= [x(t),\ y(t)]$ $\quad t \in Ti(M)$
  if $t \notin New(M')$
  $[x'(t), y'(t)]:=I(t)=[tmin(t), tmax(t)]$

  Where $New(M')$ denotes the set of transitions newly enabled for the marking M'. These transitions are those enabled for M' and not for M, or those enabled for M' and M but are conflicting with t_f for the marking M. Otherwise, an enabled transition which does not belong to $New(M')$ is said to be persistent.

If t is an enabled transition for a state e, we note $\underline{t}$ the clock associated with t that takes its values in $\mathbb{Q}^+$. $\underline{t}$ measures the residual time of the transition t relatively to the instant where the state e is reached. The time progresses only for activated transitions, whereas it is suspended for inhibited transitions. Therefore, a transition t_f can be fired at relative time $\underline{t_f}$ from an accessible state e, if (i) t_f is activated for the marking M, and if (ii) the time can progress within the firing interval of t_f without overtaking those of other activated transitions. After firing t_f the accessible state, noted $e^{\uparrow}$, is obtained:

- by consuming a number of tokens in each input place p of t_f (given by the value $B(p,t_f)$), and by producing a number of tokens in each output place p of t_f (given by the value $F(p,t_f)$);
- by shifting the interval of a persistent activated transition with the value of the firing time of t_f. However, the interval of persistent inhibited transitions remain unchanged. Finally, a newly enabled transition is assigned its static firing interval.

3. ITPN state space construction

For a *TPN*, the state class graph method [4] allows to compute a graph that preserves chiefly the linear properties of the model. Likewise, this construction can be applied to an *ITPN*. This consists in regrouping in a same class all the states accessible after firing the same sequence of transitions; all the states of a same class have the same marking *M*. Hence, a class is defined by the pair *(M, D)* where *M* is the common marking and *D* is a set of inequalities encoding the firing space of the class. More formally, a class is defined as follows [6]:

Definition.3 *Let $ST=(\Gamma,e^0,\rightarrow)$ be the LTS associated with an ITPN. A class of states of an ITPN, denoted by E, is the set of all the states pertaining to Γ that are accessible after firing the same untimed sequence $S=(t_f^1,..,t_f^n)$ from the initial state e^0. A class E is defined by (M, D), where M is the marking accessible after firing S, and D is the firing space encoded as a set of inequalities. For $Te(M)=\{t_1,..,t_s\}$, we have :* $D=\hat{D}\wedge\not{D}$

$$\not{D}:=\begin{cases}\bigwedge_{i\neq j}\ (\underline{t_i}-\underline{t_i}\leqslant d_{ij})\\ \bigwedge_{i\leqslant s}\ (d_{i\bullet}\leqslant t_i\leqslant d_{\bullet i})\end{cases}$$

with $(t_j,t_i)\in Te(M)^2$ $d_{ij}\in\mathbb{Q}\cup\{\infty\}$, $d_{\bullet i}\in\mathbb{Q}^+\cup\{\infty\}$, $d_{i\bullet}\in\mathbb{Q}^+$

$$\hat{D}:=\bigwedge_{k=1..p}\ (\alpha_{1k}\underline{t_1}+..+\alpha_{sk}\underline{t_s}\leqslant d_k)$$

with $d_k\in\mathbb{Q}\cup\{\infty\}$, $(\alpha_{1k},..,\alpha_{sk})\in\mathbb{Z}^s$ *and Z denotes the set of relative integers.* $\forall k,\exists(i,j),\ (\alpha_{ik},\alpha_{jk})\notin\{(0,0),(1,-1)\}$

We denote by the element $\bullet$ the instant at which the class E is reached. Therefore, the value of the clock $\underline{t_i}$ expresses the time relative to the instant $\bullet$ at which the transition t_i can be fired. Thus, for each valuation ψ satisfying the system D, it corresponds a unique state $e=(M,V)$ accessible in ST after firing the sequence S.

In case of a *TPN*, the system D is reduced to $\not{D}$. The inequalities of the latter are of a particular form, called *DBM* [10]. This form makes it possible to apply an efficient algorithm to compute a class whose overall complexity is $o(m^3)$, where m is the number of enabled transitions.

However, for *ITPNs* the set of valuations pertaining to a given class cannot be encoded anymore with *DBMs*. Actually, inequalities of general form are needed to encode the firing space of a class. The manipulation of these constraints, given by the subsystem $\hat{D}$ induces a higher complexity that can be exponential in the worst case. To tackle this issue, *DBM* over approximation technique has been proposed as an alternative solution to analyse preemptive systems [2][15] [8]. This approach consists in cutting off the inequalities of the subsystem $\hat{D}$ when they appears in D; it thereby keeps only those of the subsystem $\not{D}$ to represent an over approximation of the space of D.

This solution makes it possible to build a less richer graph than the exact one, but nevertheless with lesser expenses in terms of computation time and memory usage.

For a better understanding of how works this technique, we apply the state class graph method to the *ITPN* example of Figure 1. Let $E'=(M',D')$ be the class accessible in the exact graph, noted **GR**, after firing the sequence $S=(t_4, t_1, t_5)$ from the initial class $E^0=(M^0,D^0)$.

$$E^0=\begin{pmatrix}M^0: p_1,p_3,p_4\rightarrow 1\\ D^0:\begin{cases}3\leq\underline{t_1}\leq 3\\ 2\leq\underline{t_3}\leq 4\\ 0\leq\underline{t_4}\leq 2\end{cases}\end{pmatrix}\qquad E'=\begin{pmatrix}M': p_1,p_4,p_6\rightarrow 1\\ D':\begin{cases}0\leq\underline{t_2}\leq 4 & 4\leq\underline{t_6}\leq 4\\ 0\leq\underline{t_3}\leq 4 & 1\leq\underline{t_2}+\underline{t_3}\leq 6\end{cases}\end{pmatrix}$$

We can easily notice that the transition t_6 is not firable from E' since t_2 or t_3 should be fired before. Put in other way, the firing of t_6 requires that the system $D'\wedge(t_6\leqslant t_3)\wedge(t_6\leqslant t_2)$ admits at least one solution; we should check if $t_2=t_3=t_6=4$ and $t_2+t_3\leqslant 6$. As this last inequality does not hold, therefore t_6 cannot fire. The system D' has a polyhedral form that cannot be reduced to a *DBM*. The *DBM* approximation consists in cutting off the polyhedral constraints $1\leqslant t_2+$

$t_3 \leqslant 6$. We thereby obtain a *DBM* system whose manipulation consumes a polynomial time in the number of clocks. However, by doing this, t_6 becomes firable since $t_6=4$ holds. Therefore, the system $D'=\eth'$ denotes an overapproximation of the system D'. In other words, we add new states in the class E' that are not accessible indeed. Nevertheless, this construction, noted ~~GR~~, makes it possible to preserve a subset of properties that deal with safety.

In the sequel, we encode the system ~~Đ~~ as a square matrix where each line and corresponding column, are indexed by an element of $Te(M) \cup \{\bullet\}$. In concrete terms, we have: $\forall t_i,t_j \in Te(M)^2 \wedge (t_i \neq t_j)$, ~~Đ~~$[\bullet,t_i]:=d_{\bullet i}$; ~~Đ~~$[t_i,\bullet]:=-d_{i\bullet}$; ~~Đ~~$[t_i,t_j]:=d_{ij}$; ~~Đ~~$[t_i,t_i]:=0$; ~~Đ~~ $[\bullet,\bullet]:=0$.

These matrix notations are used to represent the coefficients of the system ~~Đ~~. For example, the matrix shown in Table 1 encodes the system $D^0=\eth^0$ associated with the initial class of the exact graph of the *ITPN* of Figure 1..

Table.1. Matrix representation of the system ~~Đ~~^{0}

~~Đ~~^{0}	$\bullet$	t_1	t_3	t_4
$\bullet$	0	3	4	2
t_1	-3	0	1	-1
t_3	-2	1	0	0
t_4	0	3	4	0

Taking on the previous definitions, if $E=(M,D)$ is a class accessible in *GR*, then the class $\ddot{E}=(M,\eth)$ is an over approximation of *E*, if the space of states of *E* is included in that of $\ddot{E}$. Hence, by substituting $\ddot{E}$ for *E* in the graph *GR*, it results that the class $\ddot{E}$ may derive additional sequences that are not firable indeed in *GR* from *E*. We thereby obtain an approximation of the graph *GR* that we can build as defined next:

In other respects, it has been shown in [1] that by relaxing a little bit in the precision of the *DBM* approximation we achieve to reduce yet more the computation effort of the graph while compacting its size. However, as this new abstraction is not as precise as that defined in [2], it may therefore contain additional sequences. Nevertheless, it may also yield, with lesser expenses, smaller graphs that are bisimilar to GR. Formally this construction is defined as follows:

Definition.4 *The contracted approximated graph of an ITPN, denoted by ~~GRC~~, is the tuple $(CEC,Ec^0,\hookrightarrow)$, such that :*

- *CE is the set of approximated classes accessible in ~~GRC~~;*
- $Ec^0=(M^0, \eth c^0) \in CEC$ *is the initial class such that* $\eth c^0 = \eth^0 = D^0$;
- $\hookrightarrow$ *is a transition relation between approximated classes defined on* $CEC \times T \times CEC$, *such that* $((M,\eth c),t_f,(M',\eth c')) \in \hookrightarrow$, *iff :*

  1. $(t_f \in Ta(M)) \wedge (\beta[t_f] \geqslant 0)$ *such that:* $\forall x \in Te(M) \cup \{\bullet\}$, $\beta[x]=MIN_{t \in Ta(M)}\{\eth c[x,t]\}$.

  2. $\forall p \in P$, $M'(p):=M(p)-B(p,t_f)+F(p,t_f)$.

3. The coefficients of the DBM inequalities of the system $\mathbb{D}c'$ *are computed from those of* $\mathbb{D}c$ *by applying the following algorithm:*

$\forall t \in Te(M')$ $\mathbb{D}c'[t,t]:=0;$ $\mathbb{D}c'[\bullet,\bullet]:=0;$

If t is persistent

If $t \in Ti(M)$ $\mathbb{D}c'[t,\bullet]:=\mathbb{D}c[t,\bullet];$ $\mathbb{D}c'[\bullet,t]:=\mathbb{D}c[\bullet,t]$

If $t \notin Ti(M)$ $\mathbb{D}c'[\bullet,t]:=\mathbb{D}c[t_f,t]$; $\mathbb{D}c'[t,\bullet]:=\beta[t]$.

If t is newly enabled.

$\mathbb{D}c'[\bullet,t]:=tmax(t)$; $\mathbb{D}c'[t,\bullet]:=-tmin(t)$.

$\forall (t_1,t_2) \in (Te(M'))^2 \wedge (t_1 \neq t_2)$

If t_1 *or* t_2 *are newly enabled.*

$\mathbb{D}c'[t_1,t_2]:=\mathbb{D}c'[\bullet,t_2]+\mathbb{D}c'[t_1,\bullet]$.

If t_1 *and* t_2 *are persistent.*

If $(t_1,t_2) \notin (Ti(M))^2$ *or* $(t_1,t_2) \in (Ti(M))^2$

$\mathbb{D}c'[t_1,t_2]:=MIN(\mathbb{D}c[t_1,t_2], \mathbb{D}c'[\bullet,t_2]+\mathbb{D}c'[t_1,\bullet])$.

If $(t_1,t_2) \notin (Ti(M))^2 \wedge (t_1 \in Ti(M)) \vee (t_2 \in Ti(M))$

$\mathbb{D}c'[t_1,t_2]:=\mathbb{D}c'[\bullet,t_2]+\mathbb{D}c'[t_1,\bullet]$.

If t is an activated transition, then $\beta[t]$ denotes the minimal time distance between its firing time to any other firable transition. Further, $\beta[\bullet]$ represents the maximal dwelling time in the class Ec. Therefore, an activated transition t_f is not firable from Ec, if $\beta[t_f]<0$. In other words, it does not exist any state accessible in E such that the valuation of the clock associated with t_f can overtake the minimal bound $tmin(t_f)$.

The graph $\mathbb{GRC}$ is computed gradually by enumerating all the classes accessible from the initial class Ec^0. To put an end to the enumeration process, the algorithm is provided with class equivalence conditions. These are based on the equality of markings and systems. We show in the sequel how the construction of the approximated graph $\mathbb{GRC}$, as defined in *Definition.4*, can be improved still more by reducing as well as its size as the effort of its computation. This is achieved by exploring a less stronger equivalence conditions than the equality. This contraction exploits the following concepts:

1. In the construction defined through *Definition 4*, we notice that the firability of a transition is checked by comparing the coefficients $\beta[t']$ which are computed exclusively from $\mathbb{D}c[t,t']$. Besides, the elements $\mathbb{D}c[\bullet,t]$ as well as $\mathbb{D}c[t,\bullet]$ when t' is activated, are not required in the computation of the accessible system $\mathbb{D}c^{\uparrow}$. Therefore, when performing the equivalence test, we can avoid to compare the elements $\mathbb{D}c[\bullet,t]$ and $\mathbb{D}c[t,\bullet]$ when t is activated.

2. The second level of the contraction allows regrouping classes as equivalent when the comparison between some firing distances is useless. In order to discuss this idea, we need before to introduce the following notation:

- A transition t_i is said to be inhibiting t_j, if $\exists p \in P,\ 0<IH(p,t_i) \leq B(p,t_j)$. This means that if the transition t_j is enabled for a given marking, then t_j cannot be activated for this marking. We denote hereafter by *Inhib* the relation defined on T^2, such that $(t_i,t_j) \in$

Inhib, if t_j is inhibiting t_i. Note that the relation *Inhib* is not reflexive, however if (t_i,t_j) and $(t_j,t_i) \in Inhib$, then this means that t_i and t_j are inhibited when they are enabled together.

So, if we consider two transitions that cannot be activated in the same time (namely, *t'* is inhibiting *t*), then it appears that the constraints of the transition *t* have no impact on the firing of *t'*, and conversely, in the graph ~~*GRC*~~. Therefore, the distances ~~$Dc[t',t]$~~ and ~~$Dc[t,t']$~~ are useless since they are not required to decide the firing of the transitions *t'* and *t* since both cannot be activated together. Moreover, we do not need even, to compute these distances and to compare them when running the equivalence test.

3. The third level of the contraction deals with conflicting transitions. Before discussing it, we need before to introduce the following notations:

- We denote by *AT* the set of transitions of *T* that are not connected to any inhibitor arc: $t \in AT$, if $\nexists p \in P, IH(p,t) \neq 0$.
- Two transitions t_i and t_j are said to be twin, if $\forall p \in P$, $IH(p,t_i)=IH(p,t_j)$. This means that if both transitions are enabled for a given marking, then both are either inhibited or activated for this marking. We denote hereafter by *Twin* the relation defined on T^2, such that $(t_i,t_j) \in Twin$, if t_i and t_j are twin. Note that $AT^2 \subseteq Twin$ and we have if $(t_i,t_j) \in Twin$, then $(t_j,t_i) \in Twin$.

The idea is to leave out the comparison of the firing distances between two conflicting transitions *t* and *t'* when its value is positive. More concretely, if two transitions cannot be inhibited, then if ~~Dc~~ $[t, t'] \geqslant 0$, then the transition *t'* has no impact on the firing of *t* as long as both remain persistent. However if *t* is fired, then *t'* will be disabled afterwards since both transitions are conflicting for the same markings. Therefore, there is no need even to re-compute the value of positive distances since the latter preserve their status as long as the conflicting transitions are not disabled in the firing sequence. However, when dealing with inhibited conflicting transitions, this property stands consistent only when the transitions are twin.

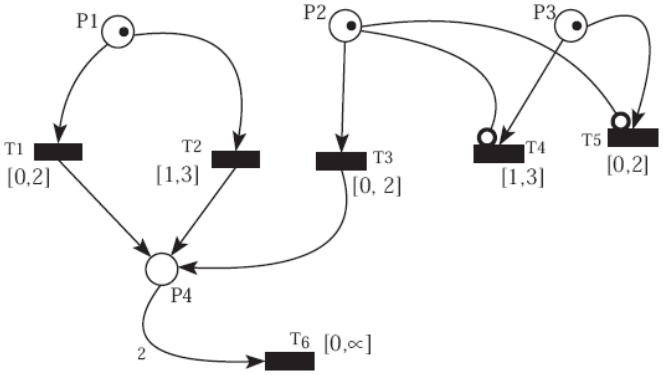

Figure. 2. An ITPN Model with twin and conflicting transitions.

To make clearer these concepts, let us consider the *ITPN* of Figure.2. Hence, we determine that $Inhib=\{(t_4,t_3),(t_5,t_3)\}$. The application of the contraction makes that the distances ~~$Dc[t_4,t_3]$~~, ~~$Dc[t_3,t_4]$~~, ~~$Dc\ [t_3,t_5]$~~ and ~~$Dc[t_5,t_3]$~~ are left out during the computation of a class as well as when performing the equivalence test. Furthermore, we determine that $AT=\{t_1,t_2,t_3,t_6\}$ and $Twin=AT^2 \cup \{(t_4,t_5),(t_5,t_4)\}$. However, among elements of *Twin*, only transitions t_1 and t_2, on a hand, and t_4 and t_5, on the other hand, are in conflict for the initial marking. Therefore, since the distances ~~$Dc^0[t_1,t_2]$~~, ~~$Dc^0[t_2,t_1]$~~, ~~$Dc^0[t_4,t_5]$~~ and ~~$Dc^0[t_5,t_4]$~~ are positive, we do not need

to re compute their value as long as the related transitions remain persistent. Furthermore, as the firing of t_1 (resp, t_4), disables t_2 (resp, t_5), and conversely, these distances stand to be useless for the equivalence test.

Remark The class Ec^i corresponds to the node numbered *(i)* in the graphs.

$E_c^0 =$ $M^0 : p_1, p_2, p_3 \rightarrow 1$

$\not{D}^0$	•	t_1	t_2	t_3	t_4	t_5
•	0	2	3	2	3	2
t_1	0	0	3	2	3	2
t_2	-1	1	0	1	2	1
t_3	0	2	3	0	3	2
t_4	-1	2	2	1	0	1
t_5	0	2	3	2	3	0

$E_c^1 =$ $M1 : p_2, p_3, p_4 \rightarrow 1$

$\not{D}_c^1$	•	t_3	t_4	t_5
•	0	2	3	2
t_3	0	0	3	0
t_4	1	1	0	1
t_5	0	2	3	0

$E_c^6 =$ $M^6 : p_2, p_3, p_4 \rightarrow 1$

$\not{D}_c^6$	•	t_3	t_4	t_5
•	0	1	3	2
t_3	0	0	3	0
t_4	1	0	0	1
t_5	0	1	3	0

$E_c^2 =$ $M^2 : p_4 \rightarrow 1; p_4 \rightarrow 2$

$\not{D}_c^2$	•	t_4	t_5	t_6
•	0	3	2	∞
t_4	-1	0	1	∞
t_5	0	3	0	∞
t_6	0	3	2	0

$E_c^8 =$ $M^8 : p_4 \rightarrow 1; p_4 \rightarrow 2$

$\not{D}_c^8$	•	t_4	t_5	t_6
•	0	3	2	∞
t_4	0	0	1	∞
t_5	0	3	0	∞
t_6	0	3	2	0

$E_c^9 =$ $M^9 : p_1, p_4 \rightarrow 1$

$\not{D}_c^9$	•	t_1	t_2
•	0	1	2
t_1	0	0	2
t_2	0	1	0

$E_c^{10} =$ $M^{10} : p_1, p_4 \rightarrow 1$

$\not{D}_c^{10}$	•	t_1	t_2
•	0	2	3
t_1	0	0	3
t_2	0	1	0

The exact construction *GR* [12], the tightest *DBM* overapproximation ~~*GR*~~ [2] [8] [15] and the abstraction ~~*GRC*~~ produce all the same graph shown in Figure.3.a. However, the application of the last properties makes it possible to contract further the graph ~~*GRC,*~~ as depicted in Figure.3.b. Although it is smaller, we notice that this graph is bisimilar to the former, as it allows gathering classes that derive the same firing sequences. For instance, firing t_1 (resp, t_2), in ~~*GRC*~~ from the initial class Ec^0 leads to the class Ec^1 (resp, Ec^6). These classes are not equal, but they are bisimilar indeed. In actual fact, the distances ~~Dc~~$[\bullet,t_3]$, ~~Dc~~$[t_4,t_3]$ and ~~Dc~~$[t_5,t_3]$ which impede the equality to hold, are useless since t_3 is inhibiting t_4 and t_5. Furthermore, the classes Ec^2 and Ec^8 are equivalent since the coefficient ~~Dc~~$[t_4,\bullet]$ can be left out. Finally, the classes Ec^{10} and Ec^9 are gathered since the positive distances ~~Dc~~$[t_1,t_2]$ can be ignored; t_1 and t_2 are two twin conflicting transitions. Hence we obtain a much compact graph of 8 nodes and 16 edges, whereas the other constructions produce a graph of 11 classes and 22 edges. More formally, we introduce this contraction as an equivalence relation, defined as given next:

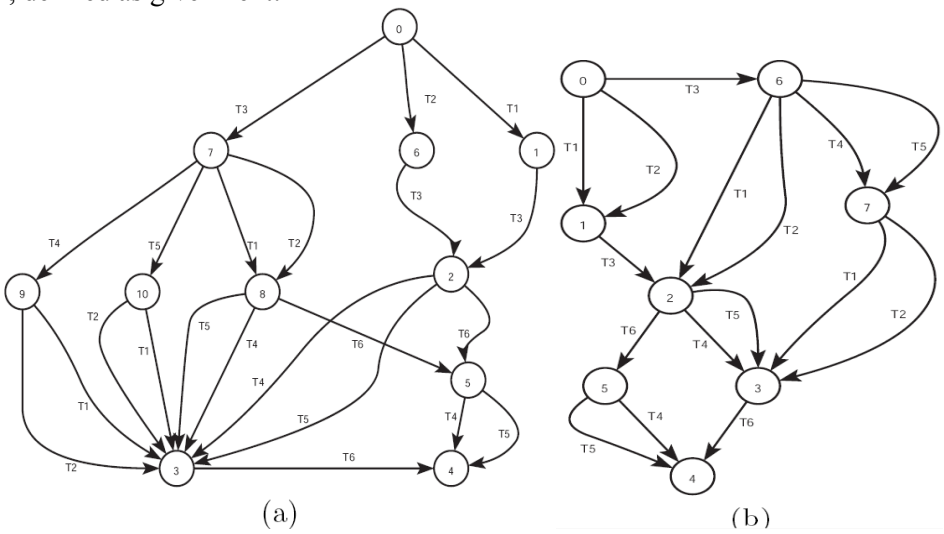

Figure.3. Exact Graph construction and its *DBM* approximations.

Definition.5 *Let $\simeq$ be a relation over state classes of the graph ~~GRC~~, defined by: ((M,~~Dc~~), (M',~~Dc'~~)) $\in \simeq$ if:*

(i) M=M'

(ii) $\forall t \in Ti(M)$ $\quad$ $~~Dc~~[\bullet,t]=~~Dc'~~[\bullet,t]$, $~~Dc~~[t,\bullet]=~~Dc~~'[t,\bullet]$

(iii) $\forall (t,t') \in Twin \cap Conf(M)$

$$\begin{cases} sg(~~Dc~~[t,t'])= sg(~~Dc'~~[t,t']) \\ ~~Dc~~[t,t']= ~~Dc'~~[t,t'] \qquad If\ sg(~~Dc~~[t,t'])=<_0 \end{cases}$$

(iv) $\forall (t,t') \in Te(M)^2$-(Twin $\cap$ Conf(M)) such that (t', t),(t,t') $\notin$ Inhib, $~~Dc~~[t,t']=~~Dc~~'[t,t']$.

where sg(v) is a function which gives the sign of the value v, sg: $\mathbb{Q} \cup \{\infty\} \rightarrow \{\geqslant_0, <_0\}$ such that $\geqslant_0$ (resp, $<_0$), denotes "positive or null" (resp, strictly negative).

In concrete terms, two classes *(M, ~~Dc~~)* and *(M', ~~Dc'~~)* are in the relation $\simeq$, if: (i) they enjoy the same marking; (ii) the maximum and minimum residual time of any inhibited transition is identical in both classes; (iii) for any pair of conflicting twin enabled transitions, the firing distance involving both transitions in both classes holds the same sign, and this distance must be equal in both classes only when it is negative; (iv) For all other pairs of enabled transitions that are not in the relation *Inhib*, the firing distance involving both transitions must be equal. We should prove now that the relation $\simeq$ is a bisimulation over the classes of the graph ~~GRC~~.

Theorem.1 *The relation $\simeq$ is a bisimulation over the graph ~~GRC~~.*

Proof: see Appendix.

By avoiding, on a hand, to compute some distances when working out each accessible class, and on the other hand, to compare them during the equivalence test, we succeed to reduce the computation effort of the approximated graph ~~GRC~~. This construction achieves, in general, to reduce sensibly the size of the graphs, but however while losing a bit of precision in the approximation. The last abstraction over the classes of the graph ~~GRC~~ is the quotient graph of ~~GRC~~ w.r.t the relation $\simeq$. It preserves markings and both firing sequences while it is, in general, smaller. The ~~GRC~~ may be more appropriate than ~~GR~~ to check over linear properties of the model, especially when the number of additional sequences that have been added due to constraint relaxation stands of limited number. However, when the ~~GRC~~ provides a too coarse approximation, it may yield a larger graph than ~~GR~~; the additional sequences are too numerous to be wrapped by the contraction. In actual fact, the abstraction ~~GRC~~ is more convenient to be built when the number of inhibiting and conflicting transitions is important in the net, otherwise the construction of ~~GR~~ should be considered.

4. Experimental Results

The tests have been performed on a Pentium V with a processor speed of 2,7 GHZ and 1,9 GB of memory capacity. They have been carried out while using different tools: *TINA* tool [18], *ROMEO* tool [17] and our tool named *ITPNT*. Their performances are assessed while considering three parameters, the number of classes, the number of edges, and finally in terms

of computation times. It is noteworthy that *ROMEO* does not bring out some parameters; we denote that by the notation *NA* (Not Available).

Through these tests, we intend to advocate the benefits of using the ~~*GRC*~~ construction when dealing with conflicting and inhibiting transitions. For this effect, we have considered the *ITPN* given in Figure.4 while varying the intervals of transitions t_6, t_7 and t_8; the results of these experiments are reported in Table.2.

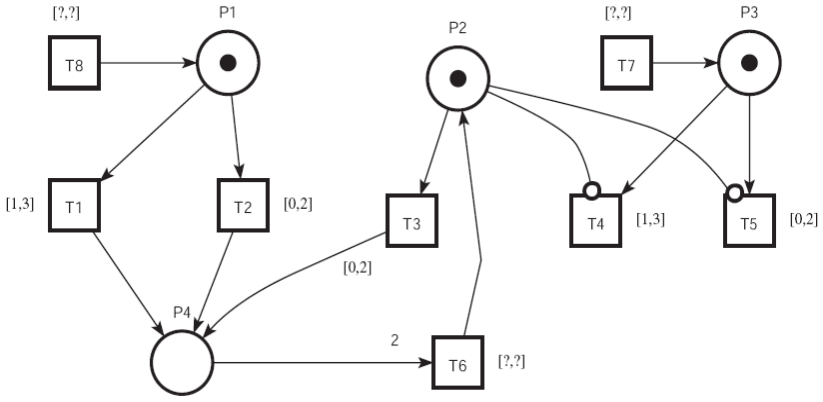

Figure 4. *ITPN* used in the experimentations.

All the experiments show that the graph computation times are in favour of our algorithm. Indeed, the construction of ~~*GRC*~~ achieves to reduce significantly the size of the graphs as well as their computation effort.

Table.2 Results of experiments

	TOOLS	TINA	ROMEO		ITPNT	
Examples	Methods	K-grid	Exact	DBM	~~*GR*~~(=)	~~*GRC*~~($\simeq$)
t_6 [2,5]	Classes	1.035	1.035	1.318	1.035	842
t_7 [22,35]	Edges	1.830	1.830	2.323	1.830	1.471
t_8 [20,30]	Times(ms)	312	NA	NA	8	4
t_6 [2,5]	Classes	750	750	1.685	750	742
t_7 [12,15]	Edges	1.363	1.363	3.106	1.363	1.356
t_8 [10,20]	Times(ms)	219	NA	NA	4	3
t_6 [4,8]	Classes	2.346	2.346	3.398	2.402	1.880
t_7 [14,20]	Edges	4.969	4.969	7.198	5.090	4.118
t_8 [10,20]	Times(ms)	1312	NA	NA	19	12
t_6 [4,10]	Classes	NF	NF	20.638	19.739	16.981
t_7 [16,22]	Edges	NF	NF	46.216	43.378	38.338
t_8 [10,18]	Times(ms)	NA	NA	NA	426	363
t_6 [4,8]	Classes	3.203	3.203	4.451	3.238	2.648
t_7 [16,22]	Edges	6.603	6.603	9.184	6.756	5.758
t_8 [10,15]	Times(ms)	1.594	NA	NA	28	19

5. Conclusion

We have proposed in this paper an efficient algorithm to contract the *DBM* over-approximation of the state class graph of preemptive systems. For this effect, we have shown

in [1] that by relaxing a little bit in the precision of the *DBM* approximation, we can achieve to compute graphs that can be more appropriate, in certain cases, to model-checking the linear properties of the model. We have discussed in this paper how this construction can be improved yet more by leaving out all the distances that are useless for the class computation process. Hence, we have put forward an equivalence relation that reduces sensibly the size of the graphs as well as the effort of their computation. Experimental results have been reported to advocate the benefits of this approach.

References

[1] Abdelli, and D.Yahiatene. A. Efficient computation of state space over approximation of preemptive real time systems. IEEE AICCSA 2008: 726-733.

[2] A.Abdelli. Optimisation de la construction d'une approximation de l'espace d'état des systèmes préemptifs. In Journal Technique et sciences Informatiques, Hermes Lavoisier editions, VOL 28:9 2009. pp.1143-1170.

[3] R. Alur, C. Courcoubetis, N. Halbwachs, T. A. Henzinger, P-H. Ho, X. Nicollin, A. Olivero, J. Sifakis, S. Yovine: The Algorithmic Analysis of Hybrid Systems. Theor. Comput. Sci. 138(1): 3-34 (1995)

[4] B. Berthomieu, M. Menasche: An Enumerative Approach for Analyzing Time Petri Nets. IFIP Congress 1983: 41-46.

[5] B. Berthomieu, and M. Diaz. Modeling and verification of time dependant systems using Time Petri Nets. IEEE TSE, 17(3):(259-273), March 1991.

[6] B. Berthomieu, D. Lime, O. H. Roux, F. Vernadat. Reachability Problems and Abstract State Spaces for Time Petri Nets with Stopwatches. Discrete Event Dynamic Systems 17(2): 133-158 (2007).

[7] H. Boucheneb, H. Rakkay: A More Efficient Time Petri Net State Space Abstraction Useful to Model Checking Timed Linear Properties. Fundam. Inform. 88(4): 469-495 (2008).

[8] G. Bucci, A. Fedeli, L. Sassoli, and E.Vicario. Timed State Space Analysis of Real-Time Preemptive Systems. IEEE TSE, Vol 30, No. 2, Feb 2004.

[9] Cassez : F. Cassez and K.G. Larsen. The Impressive Power of Stopwatches. LNCS, vol. 1877, pp. 138-152, Aug. 2000.

[10] Dill, D.L.: Timing assumptions and verification of finite-state concurrent systems; Workshop Automatic Verification Methods for Finite-State Systems. Vol 407. (1989) 197-212.

[11] Thomas A. Henzinger: The Theory of Hybrid Automata. LICS 1996: 278-292

[12] D.Lime, and O.H.Roux. Expressiveness and analysis of scheduling extended time Petri nets. In 5th IFAC International Conference on Fieldbus Systems and their Applications, (FET'03), Elsevier Science, July, 2003.

[13] P. Merlin. "A study of the recoverability of computer system". PhD thesis Dep. Comp. Science, Uni. California, Irvine, 1974.

[14] M. Magnin, D. Lime, O. H. Roux: An Efficient Method for Computing Exact State Space of Petri Nets With Stopwatches. Electr. Notes Theor. Comput. Sci. 144(3): 59-77 (2006).

[15] O. H. Roux, D. Lime: Time Petri Nets with Inhibitor Hyperarcs. Formal Semantics and State Space Computation. ICATPN 2004: 371-390.

[16] ORIS TOOL:http://www.stlab.dsi.unifi.it/oris/index.html.

[17] ROMEO TOOL http://romeo.rts-software.org.

[18] TINA Tool http://www.laas.fr/tina/.

7. Appendix

Here comes the proof of the Theorem 1.

We should prove that if *(Ec, Ec')* satisfies the hypotheses of *Definition 5*, then we have:

1: If an activated transition t_f can fire from *Ec*, then t_f can fire from *Ec´* too.

2: If $Ec \hookrightarrow Ec^{\uparrow} \wedge Ec' \hookrightarrow Ec'^{\uparrow}$, then $(Ec^{\uparrow}, Ec'^{\uparrow}) \in \simeq$; $Ec^{\uparrow}$ and $Ec'^{\uparrow}$ satisfy *Definition 5*.

1. Let us assume that the transition t_f is firable from $Ec=(M, \not{D}c)$. As Ec and Ec' are in the relation $\simeq$, then the hypotheses of *Definition 5* are satisfied. Basing on the firing condition, we need to prove that **(A1)**: if $\beta c[t_f] \geq 0$, then $\beta c'[t_f] \geq 0$, namely that: $MIN_{t \in Ta(M)}\{\not{D}c'[t_f,t] \geq 0$.
As t_f and t' are both activated, then (t_f, t'), $(t, t_f) \notin Inhib$. Hence, from hypotheses *(iii)* and *(iv)* of *Definition.5*, we determine the property **(A1)**; t_f is firable from Ec'.

2. We have to prove that the hypotheses of *Definition 5* are satisfied for $(Ec^{\uparrow}, Ec'^{\uparrow})$

(a) It is obvious that as $M=M'$, we have $M^{\uparrow}=M'^{\uparrow}$.

(b) Let us prove that $\forall t \in Ti(M^{\uparrow})$, $\not{D}c[\bullet,t] = \not{D}c'^{\uparrow}[\bullet,t]$. Let us replace $\not{D}c^{\uparrow}[\bullet,t]$ with its computation formula according to the status of t, as given in *Definition 4*.

- if $t \in New(M^{\uparrow})$, then we have : $\not{D}c^{\uparrow}[\bullet,t] = \not{D}c'^{\uparrow}[\bullet,t] = tmax(t)$.
- if $t \notin New(M^{\uparrow})$, then we should consider whether t is inhibited for M or not.
  - If $t \notin Ti(M)$, then we need to prove that $\not{D}c[t_f, t] = \not{D}c'[t_f,t]$. This last property holds since $(t_f,t) \notin Twin \cap Conf(M)$, otherwise t should be disabled after firing t_f. Furthermore, $(t_f,t) \notin$ Inhib otherwise t_f should be inhibited for M. Also $(t,t_f) \notin$ Inhib, otherwise t should be inhibited for M.
  - If $t \in Ti(M)$, then the proof is obvious from the hypothesis *(ii)*.

  Notice that likewise we can also prove that $\forall t \in Ta(M^{\uparrow})$, $\not{D}c^{\uparrow}[\bullet, t] = \not{D}c'^{\uparrow}[\bullet, t]$.

(c) We should prove that $\forall t \in Ti(M^{\uparrow})$, $\not{D}c^{\uparrow}[t,\bullet] = \not{D}c'^{\uparrow}[t,\bullet]$.
Let us replace $\not{D}c^{\uparrow}[t,\bullet]$ with the suitable computation formula according to the status of the transition t.

- if $t \in New(M^{\uparrow})$, then we have $\not{D}c^{\uparrow}[t,\bullet] = \not{D}c'^{\uparrow}[t,\bullet] = -tmin(t)$.
- if $t \notin New(M^{\uparrow})$, then we should consider whether t is inhibited for M or not.
  - If $t \notin Ti(M)$, then we need to prove that $\beta c^{\uparrow}[t] = \beta c'^{\uparrow}[t]$, namely that $MIN_{t' \in Ta(M)}\{\not{D}c[t,t']\} = MIN_{t' \in Ta(M)}\{\not{D}c'[t,t']\}$.

    If $(t,t') \notin (Twin \cap Conf(M)) \cup Inhib$, then we have $\not{D}c[t,t'] = \not{D}c'[t,t']$.
    However, $(t',t) \notin$ Inhib (resp, $(t,t') \notin$ Inhib), otherwise t' must be inhibited (resp, t must be inhibited), for M.

    If $(t,t') \in Twin \cap Conf(M)$, then we have $sg(\not{D}e[t,t']) = sg(\not{D}c'[t,t'])$ and yet more $\not{D}c[t,t'] = \not{D}c'[t,t']$ when $sg(\not{D}e[t,t']) = <_0$. As $t \in Ta(M)$ and $\not{D}c[t,t]=0$, then $MIN_{t' \in Ta(M)}\{\not{D}c[t,t']\} \leq 0$. Therefore the value of $\not{D}c[t,t']$ has no effect on the calculation of the minimum when $sg(\not{D}c[t,t']) = \geq_0$; hence the equality holds.
  - If $t \in Ti(M)$, then the proof is stemmed from the hypothesis (ii).
    Notice that likewise we can also prove that $\forall t \in Ta(M^{\uparrow})$, $\not{D}c^{\uparrow}[t,\bullet] = \not{D}c'^{\uparrow}[t,\bullet]$.

(d) We have to prove that **(A2)**: $\forall (t, t') \in Twin \cap Conf(M^{\uparrow})$
$sg(\not{D}c^{\uparrow}[t,t']) = sg(\not{D}c'^{\uparrow}[t,t'])$
$\not{D}c^{\uparrow}[t,t'] = \not{D}e'^{\uparrow}[t,t']$ if $sg(\not{D}c^{\uparrow}[t,t']) = <_0$

As we deal with safe nets, we can easily show that if two persistent transitions are in conflict for M, then they remain in conflict for $M^{\uparrow}$. Hence, $Conf(M^{\uparrow})$ consists of all the pairs

of transitions $(t,t') \in Conf(M)$ that are persistent in $M^{\uparrow}$ to which we add the pairs of conflicting transitions for $M^{\uparrow}$ where at least one transition is newly enabled. First of all, it is obvious that if $t \in New(M^{\uparrow})$ or $t' \in New(M^{\uparrow})$, then (t,t') satisfies the hypothesis (**A2**).

Let us discuss the case where two twin conflicting transitions are persistent, $(t,t') \notin (New(M^{\uparrow}))^2$. Therefore, according to *Definition 4*, we have:

$\not{D}c^{\uparrow}[t,t'] = MIN(\not{D}c[t,t'], \not{D}c^{\uparrow}[t,\bullet] + \not{D}c^{\uparrow}[\bullet,t'])$.

As it is assumed that $sg(\not{D}c[t,t']) = sg(\not{D}c'[t,t'])$, and we have already proved through **(b),(c)** that **(A3)**: $\forall\ t \in Te(M)$, $\not{D}c^{\uparrow}[\bullet,t] = \not{D}c'^{\uparrow}[\bullet,t]$ and $\not{D}c^{\uparrow}[t,\bullet] = \not{D}c'^{\uparrow}[t,\bullet]$, we determine that $sg(\not{D}c^{\uparrow}[t,t']) = sg(\not{D}c'^{\uparrow}[t,t'])$.

Furthermore, if $sg(\not{D}c^{\uparrow}[t,t']) = sg(\not{D}c'^{\uparrow}[t,t']) = <_0$, then we should prove that $\not{D}c^{\uparrow}[t,t'] = \not{D}c'^{\uparrow}[t,t']$; two cases can be seen:

- $\not{D}c^{\uparrow}[t,t'] = \not{D}c[t,t']$: as $\not{D}c[t,t'] = \not{D}'[t,t']$ when $sg(\not{D}c[t,t']) = <_0$, we guarantee that $\not{D}c^{\uparrow}[t,t'] = \not{D}c'^{\uparrow}[t,t']$.
- $\not{D}c^{\uparrow}[t,t'] = \not{D}c^{\uparrow}[t,\bullet] + \not{D}c^{\uparrow}[\bullet,t']$: the property (**A3**) guarantees that $\not{D}c^{\uparrow}[t,t'] = \not{D}c'^{\uparrow}[t,t']$.

(e) We have to prove that **(A4)**:

$\forall (t,t') \in Te(M^{\uparrow})^2 - ((Twin \cap Conf(M^{\uparrow}))$, such that $(t,t'),(t',t) \notin$ Inhib, $\not{D}c^{\uparrow}[t,t'] = \not{D}c'^{\uparrow}[t,t']$.

If $t \in New(M^{\uparrow})$ or $t' \in New(M^{\uparrow})$, then by using property (**A3**) we prove that,

$\not{D}c^{\uparrow}[t,t'] = \not{D}c'^{\uparrow}[t,t'] = \not{D}c^{\uparrow}[t,\bullet] + \not{D}c^{\uparrow}[\bullet,t'] = \not{D}c'^{\uparrow}[t,\bullet] + \not{D}c'^{\uparrow}[\bullet,t']$.

- Let us discuss the case where both transitions are persistent; we have either $\not{D}c^{\uparrow}[t,t'] = MIN(\not{D}c[t,t'], \not{D}c^{\uparrow}[t,\bullet] + \not{D}c^{\uparrow}[\bullet,t'])$ or $\not{D}c^{\uparrow}[t,t'] = \not{D}c^{\uparrow}[t,\bullet] + \not{D}c^{\uparrow}[\bullet,t']$. As $(t,t') \notin (Twin \cap Conf(M^{\uparrow}) \cup$ Inhib, then $\not{D}c[t,t'] = \not{D}c'[t,t']$; hence the property **(A4)** holds.

Authors

Dr Abdelli Abdelkrim is an associate professor at the computer science department of the university USTHB in Algiers. He is a member of the research team MOVES of the LSI laboratory. He received his master degree in 1998 and his PHD in 2007 from the same university. His areas of interests deal with formal methodologies, real time systems, and multimedia applications.