H. Comon and Y. Jurski

Timed automata and the theory of real numbers

Research Report LSV−99−6, July 1999

Laboratoire
Spécification
et
Vérification

CENTRE NATIONAL
DE LA RECHERCHE
SCIENTIFIQUE

Ecole Normale Supérieure de Cachan
61, avenue du Président Wilson
94235 Cachan Cedex France

Short version published in *Proc. 10th Int. Conf. Concurrency Theory (CONCUR'99), Eindhoven, NL, Aug. 1999, volume 1664 of Lecture Notes in Computer Science, pages 242−257. Springer, 1999.*

Timed automata and the theory of real numbers

Hubert Comon Yan Jurski
LSV, ENS Cachan, 94235 Cachan cedex, France
{comon,jurski}@lsv.ens-cachan.fr
http://www.lsv.ens-cachan.fr/∼comon
Tel: 01 47 40 24 30. Fax: 01 47 40 24 64

Abstract

A configuration of a timed automaton is given by a control state and finitely many clock (real) values. We show here that the binary reachability relation between configurations of a timed automaton is definable in an additive theory of real numbers, which is decidable. This result implies the decidability of model checking for some properties which cannot be expressed in timed temporal logics and provide with alternative proofs of some known decidable properties. Our proof is based on two intermediate results: 1. Every timed automaton can be effectively emulated by a timed automaton which does not contain nested loops. 2. The binary reachability relation for counter automata without nested loops (called here *flat automata*) is expressible in the additive theory of integers (resp. real numbers). The second result can be derived from [10].

1 Introduction

Timed automata have been introduced in [4] to model real time systems and became quickly a standard. They roughly consist in adding to finite state automata a finite number of clocks which grow at the same speed. Each transition comes together with some clock resets and an enabling condition, whose satisfaction depends on the current clock values. Temporal properties of real time systems have been expressed and studied through temporal logics such as TPTL [7], TCTL [2, 14, 20], MITL [6], timed μ-calculi [14, 15]. These logics are in general undecidable, with the notable exception of MITL. On the other hand, the model-checking is decidable for the (real-time) branching time logics, though hard in general.

Timed models are harder than untimed ones since they can be seen as infinite state systems in which every configuration consists of a pair of a control state (out of a finite set) and a vector of real clock values. Reasoning about possible clocks values in each state is the core of the difficulty. In this paper, we adopt the following point of view: infinite sets of configurations can be finitely described using *constraints*. For instance, "$(q, x \geq y + z)$" is the set of configurations in control state q and such that the clock x is larger than the sum of clocks y and z. This point of view is not new, as the *regions* of [2], which are used in a crucial way in the verification algorithm, are a representation of sets of configurations indeed. Here, we go one step further: we express not only sets of configurations, but also *relations* between configurations in a (decidable) constraint system. Then temporal properties of the model are described through the binary reachability relation $\xrightarrow{*}$ relating clock values, which is expressible in the constraint system. Since we may always assume that there is a clock τ which is never reset by the automaton (and hence is a witness of the total elapsed time), we may express for instance some delay conditions such as "d is a delay between q and q'" as a constraint: $\exists \vec{x}, \vec{x'}, \tau.(q, \vec{x}, \tau) \xrightarrow{*} (q', \vec{x'}, \tau + d)$. Now it is possible to analyse delays between some events such as finding minimal or maximal delays. There are already algorithms which find such extremal delays [11], but we may also decide properties such as: "the delay between event a and event b is never larger than twice the delay between event a' and event b'" (which is, up to our knowledge, a new decidability result). More generally, our main result is that the binary reachability relation between clocks values, which is defined by a timed automaton, is effectively expressible in the additive theory of real numbers. Since the additive theory of real numbers is decidable, any property which can be expressed in this theory using the reachability relation, can be decided. In particular, we can compute reachable configurations from a definable set of configurations as well as the set of configurations from which we can reach a definable set. Hence we have forward and backward model-checking algorithms of safety properties as simple instances of our result. But we may also check properties which express relations between the original and final clock values. Also, some parametric verification is possible as we may keep free variables in the description of original and final configurations: for safety properties, the results of [19] can be derived from our main result.

On the negative side, not all timed temporal properties can be expressed in the first-order theory of $\xrightarrow{*}$. Typically, unavoidability is not expressible. This is not surprising since our logic is decidable, whereas the timed temporal logics are not in general.

Our main result is proved in two steps: first we show that any timed automaton can be emulated by an automaton without nested loops, hereafter called *flat automaton*. The notion of emulation will be precised, but keep only in mind that it preserves the reachability relation. Hence, in some sense, timed automata with a star height n are not more expressive than timed automata with star height 1. (This is not true, of course, if we consider the accepted language instead of the reachability relation as an equivalence on automata). The second step consists of applying one of our former results, which shows that the reachability relation is effectively expressible in the additive theory of real (resp. integer) numbers for flat counter automata [10]. We go from timed automata to automata with counters using an encoding due to L. Fribourg [12].

The emulation result itself is proved in three steps: first we define an equivalence relation on transition sequences, which we show to be a right compatible equivalence of finite index. This is similar to a region construction, though the equivalence is rather on pairs of configurations than on configurations. Second, we show some commutation properties of equivalent transition sequences: roughly, equivalent transition sequences can be performed in any order, without affecting the reachability relation. The third (and last) step consists in using combinatorial arguments on words and proving that there is a flat automaton whose language contains a set of representatives for the congruence generated by the commutation properties. (This result can be stated as a formal language property which is independent from the rest of the paper).

From this proof, we can also derive some other decidability results. For instance, we can decide whether a sequence of transitions can be iterated.

In section 2, we recall the basic definitions of timed automata and we introduce our constraint system. Next we will prove the emulation result in section 3 and derive in section 4 the definability of the reachability relation. In section 5, we show some examples of temporal properties which can be expressed in the theory of real numbers and conclude in section 6.

2 Timed Automata

We start with a classical notion of timed automaton, which includes invariants in the states and guarded transitions. The syntax and semantics of timed transition systems we use here is not important: we can switch from the following definitions to others (such as [5]) without changing our main result. The events and the accepted language are also irrelevant here, as we are interested in reachability.

2.1 Syntax and semantics

Let B be a finite set of real numbers (we will assume later that these constants are in $\mathbb{Z}$) and C a finite number of variables called *clocks*. $\Phi(B,C)$ is the set of conjunctions[1] of atomic formulas of the form $x \leq c, x \geq c, x < c, x > c, x = c$ where $x \in \mathcal{X}$ and $c \in B$.

Definition 1 ([1]) *A* timed automaton*) is a tuple* $< \Sigma, Q, Q_0, C, I, E >$ *where*

- Σ *is a finite alphabet*
- Q *is a finite set of states (and* $Q_0 \subseteq Q$ *is a set of start states, irrelevant here)*
- C *is a finite set of clocks*
- I *is a mapping from* Q *to* $\Phi(B,C)$ *(the* invariant *associated with each state).*
- $E \subseteq Q \times Q \times \Sigma \times 2^C \times \Phi(B,C)$ *gives the set of transitions. In each transition* $< q, q', a, \lambda, \phi >$*,* λ *is a set of* clock resets *and* ϕ *is a* clock constraint*.*

A *configuration* of the automaton is a pair $(q, \vec{V})$ where $q \in Q$ and $\vec{V} \in \mathbb{R}_+^{|C|}$ is a clock value. There is a *move* of a timed automaton $\mathcal{A}$ from a configuration $(q, \vec{V})$ to $(q', \vec{V'})$, which we write $(q, \vec{V}) \underset{\mathcal{A}}{\longrightarrow} (q', \vec{V'})$, iff

- Either $q = q'$ and $\vec{V} \models I(q), \vec{V'} \models I(q)$ and there is a positive real number t such that, for every component i, $v'_i = v_i + t$.
- Or else there is a transition $< q, q', a, \lambda, \phi >$ and a positive real number t such that $\vec{V} \models \phi$ and for every component i, either $v_i \in \lambda$ and then $v'_i = t$ or else $v_i \notin \lambda$ and $v'_i = v_i + t$. Moreover, $\vec{V'} \models I(q')$.

$\overset{*}{\underset{\mathcal{A}}{\longrightarrow}}$ is the reflexive transitive closure of $\underset{\mathcal{A}}{\longrightarrow}$. We also write $\xrightarrow[q_1,q_2]{*} \subseteq \mathbb{R}^{|C|} \times \mathbb{R}^{|C|}$ the relation on clocks vectors defined by $\vec{V} \xrightarrow[q_1,q_2]{*} \vec{V'}$ iff $(q_1, \vec{V}) \overset{*}{\underset{\mathcal{A}}{\longrightarrow}} (q_2, \vec{V'})$. We will always assume without loss of generality that there is a clock τ which is never reset.

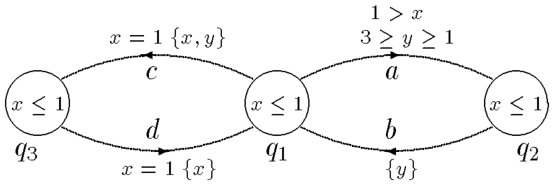

Figure 1: A timed automaton

Example 2 An example of a timed automaton is displayed on figure 1. As usual, invariants are written in the states and enabling conditions label the edges. The variables which are reset by a transition are written inside brackets. We assume that there are three clocks x, y, τ.

If we consider for instance transitions c, d only, we can express $\xrightarrow[q_1,q_1]{*}$ using the formula:

$$\begin{array}{ll} \exists \tau_1, x_1, y_1, t_1, t_2. & \tau_1 = \tau + t_1 \wedge x_1 = x + t_1 \wedge y_1 = y + t_1 \wedge x_1 \leq 1 \\ \wedge & \exists n. \tau' = \tau_1 + 2n + 1 - t_2 \wedge x' = t_2 \wedge x' \leq 1 \wedge y' = 1 + t_2 \end{array}$$

t_1 is the time spent before the first transition c is fired and is specified on the first line. τ_1, x_1, y_1 are the values of the clocks at that date. Then n is the number of times the loop cd is executed and t_2 is the time spent in q_1 after the last transition d has been fired.

This is typically what we will get from our formula computation.

A *flat automaton* is a timed automaton which does not contain nested loops: for every state q there is at most one non-empty path from q to itself.

Example 3 The automaton of figure 1 is not flat. If we remove any of the four transitions, we have a flat automaton.

2.2 Emulation

Definition 4 *A timed automaton $\mathcal{A}'$* emulates *$\mathcal{A}$ if there is a mapping ϕ from the set of states of $\mathcal{A}'$ into the set of states of $\mathcal{A}$ such that, for every states q, q' of $\mathcal{A}$ and every clock vectors $\vec{V}, \vec{V'}$, $(q, \vec{V}) \xrightarrow[\mathcal{A}]{*} (q', \vec{V'})$ iff there are states $q_1 \in \phi^{-1}(q)$, $q_1' \in \phi^{-1}(q')$ such that $(q_1, \vec{V}) \xrightarrow[\mathcal{A}']{*} (q_1', \vec{V'})$.*

Example 5 The automaton of figure 2 emulates the automaton of figure 1. The states q_i^j are mapped to q_i. It is a flat automaton. It is not straightforward that this automaton indeed emulates the original one. Note for instance that the possible event sequences are different as *abcdcdabcd* is a possible sequence in the automaton of figure 1 and is not a possible sequence in the automaton of figure 2. However, this sequence yields the same binary relation between configurations as the sequence *abcdabcdcd* which is possible in the automaton of figure 2.

[1] Having arbitrary Boolean combination does not increase the expressive power. We choose to consider conjunctions only since they guarantee a convexity property for the invariant constraints.

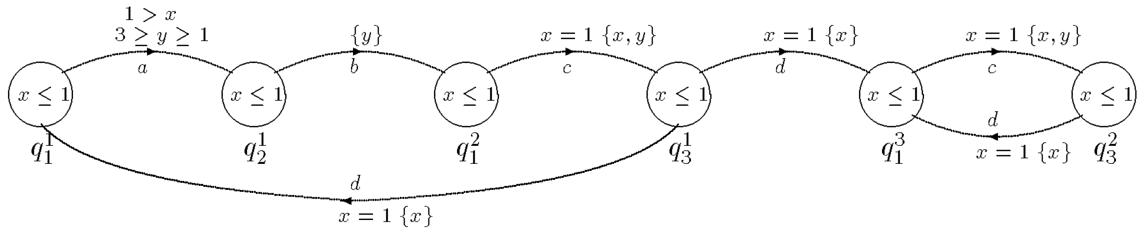

Figure 2: An emulation automaton

The automaton of figure 2 is typically what we want to compute from the automaton of figure 1.

If $\mathcal{A}'$ emulates $\mathcal{A}$ then $\xrightarrow[q_1,q_2]{*} = \bigcup_{\substack{q_1' \in \phi^{-1}(q_1) \\ q_2' \in \phi^{-1}(q_2)}} \xrightarrow[q_1',q_2']{*}$ hence, as far as $\xrightarrow{*}$ is concerned, we may consider any automaton emulating $\mathcal{A}$ instead of $\mathcal{A}$ itself.

2.3 The additive theory of real numbers

The theory $\mathcal{T}$ we consider here is defined as follows. Terms are built from variables, the constants $0, 1$ and the function symbol $+$. Formulas are built using first-order quantifiers and the usual logical connectives on atomic formulas which are either equations $u = v$ between terms or predicates $Int(u)$ where u is a term.

The domain of interpretation of such formulas is the set of non-negative real numbers, with the usual interpretation of function symbols. Int is the set of natural numbers.

This theory can be encoded in S1S using (infinite) binary representation of real numbers. Hence it is a decidable theory.[2]

Example 6 The formula of example 2 is a formula of this theory.

3 Every timed automaton can be emulated by a flat timed automaton

The automaton of figure 1 is not flat because there are two loops ab and cd on state q_1. If the order of the two loops was irrelevant to the reachability relation, we could switch them and assume that all sequences ab are performed before sequences cd. Then we would get a flat automaton, first considering the loop ab and then the loop cd. However, in this example, we cannot switch the two sequences because, for instance, $abcd$ and $cdab$ do not induce the same relations on the initial values of the clocks. Then, the question is: when can we switch two sequences of transitions w and w' without altering the reachability relation? Let us look first at some necessary conditions.

If w and w' do not induce the same relations on initial clock values, then their order is relevant since, for instance w may occur after some other transition sequence, whereas

[2] We do not know the complexity of this theory, nor of the fragment of $\mathcal{T}$ which is used here.

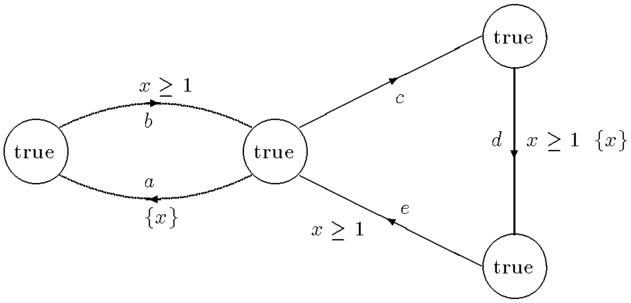

Figure 3: Another example of non-commuting loops

w' cannot. This is the case in our example: ab and cd do not induce the same relations on initial clock values and ab cannot occur after another ab, whereas cd can occur after ab. $ababcd$ is impossible and $abcdab$ is possible. Hence a first necessary condition is that w and w' induce the same constraints on initial clock values.

Similarly, w and w' should enable the same transitions: whereas w or w' has been executed last should not be relevant for further transitions. This means that w and w' should induce the same constraints on final clock values, or at least constraints that can be met by the same enabling conditions of further transitions.

There are further necessary conditions for two transition sequences to commute.

Example 7 For instance consider the automaton displayed on figure 3. On this example, let $w = ab$ and $w' = cde$. Executing ww' yields a constraint $x' - x \geq \tau' - \tau - 1$ on the final and initial x's values. Whereas $w'w$ yields a strictly weaker constraint $x' - x \geq \tau' - \tau - 2$. (To see this, start with $\tau = 1$ and $x = 0$. $w'w$ may yield a configuration in which $\tau' = 4$ and $x' = 1$, which is not reachable using ww'). On the other hand both w and w' induce the same constraints on initial clock values on one hand and on final clock values on the other hand. Hence another necessary condition for switching w and w' is that they induce the same discrepancy between $x - x'$ and $\tau - \tau'$.

We will see that, very roughly, these three necessary conditions are also sufficient.

We are going to define a right compatible equivalence $\sim$ on transition sequences such that, in particular, the above situations cannot occur when $w \sim w'$. $\sim$ will be a right compatible equivalence of finite index. Hence we will be able to split the states according to its congruence classes, ensuring that two sequences starting from the initial state and which have the same final extended state can be switched, without changing the reachability relation.

3.1 A right compatible equivalence on transition sequences

This is the analog of *regions*, considering pairs of configurations instead of single configurations. Roughly, in the regions construction, two configurations $(q, \vec{v})$ and $(q, \vec{v'})$ are considered as equivalent, if they satisfy the same constraints $x \geq y + c$, $x > y + c$ where x, y are clocks and c is a constant (which is bounded by the largest constant of the model). Here, we define a right compatible equivalence on pairs of configurations $(q_1, \vec{v_1})$, $(q_2, \vec{v_2})$ and $(q_1, \vec{v_1'})$, $(q_2, \vec{v_2'})$. Two such pairs are equivalent, roughly, if they satisfy the same constraints $x \geq y' + c, x \geq y + c \ldots$, i.e. not only constraints relating clock values at a given time, but also constraints relating clock values before and after a sequence of transitions. The situation is not as simple as in the region case, however. Indeed, the relevant constants c now range over an a priori infinite set. Hence, we will need some approximations. We show however that such approximations are faithful: they are sufficient to ensure the desirable commutation properties.

Let E be the set of transitions and w, w' be transition words. Let moreover $\phi(w)$ be the formula with free variables $\vec{X}, \vec{X'}$ which expresses the relationship between clock values before and after w:

$$(q, \vec{V}) \xrightarrow[\mathcal{A}]{w} (q', \vec{V'}) \text{ iff } \vec{V}, \vec{V'} \models \phi(w)$$

Note that this formula is independent of the states q, q' since we gave a different name for each transition of $\mathcal{A}$. Hence, given w there is only one starting and one target state for w.

In the following paragraphs, we show the technical details of our construction of the equivalence relations on transition sequences. First, we show how to represent the constraints as weighted graphs in section 3.1.1. This is a standard representation, once we have have performed a change of variables. Then we state in section 3.1.2 a number of properties of the particular graphs that are generated by transition sequences. In section 3.1.3 we define the equivalence relation on transition sequences and prove its finite index property and right compatibility in section 3.1.4. Finally, in section 3.2 we will derive the commutation property we were looking for.

3.1.1 The graph representation of constraints

For the following proofs, it will be convenient to represent the constraints $\phi(w)$ using a finite directed weighted graph. To explain this representation, we use a trick due to L. Fribourg [12]. There are mainly two ingredients:

1. First, with each clock x different from τ is associated the variable $X \overset{\text{def}}{=} \tau - x$.
2. Second, we decompose the constraint $\phi(w)$ ($w = t_1 \cdots t_n$) into time sections: the first section consists in the time spent before the first transition t_1 of w is fired; the next sections consist in the intervals between to consecutive transitions.

The advantage of the change of variables is that the clocks (if we except τ) are no longer varying continuously, but only when they are reset, i.e. at the beginning of each time section. Moreover, as we will see, the relations between them can easily be represented as a weighted graph.

More precisely, we associate with each transition $t = (q_1, q_2, a, R, \phi) \in Q \times Q \times \Sigma \times 2^C \times \Phi(B, C)$ the following constraint g_t:

$$\begin{aligned} g_t(\vec{X}, \vec{X'}) \overset{\text{def}}{=} & \\ & \bigwedge_{i=1}^{m} (\tau \geq X_i \wedge \tau' \geq X_i') \wedge \tau' \geq \tau \\ & \wedge \bigwedge_{x_i \in R} X_i' = \tau \wedge \bigwedge_{x_i \notin R} X_i' = X_i \\ & \wedge \overline{\phi}(\vec{X}) \wedge \overline{I(q_2)}(\vec{X'}) \end{aligned}$$

where $\overline{\phi}$ is the formula ϕ where each variable x (except τ) is replaced with $X - \tau$.

Example 8 Consider the example of figure 1. We have:

$$\begin{array}{rcl} g_a & \overset{\mathrm{def}}{=} & \tau \geq X \wedge \tau \geq Y \wedge \tau' \geq X' \wedge \tau' \geq Y' \wedge \tau' \geq \tau \\ & & \wedge X' = X \wedge Y' = Y \wedge X + 1 > \tau \wedge Y + 3 \geq \tau \geq Y + 1 \wedge X' \geq \tau' - 1 \\ g_b & \overset{\mathrm{def}}{=} & \tau \geq X \wedge \tau \geq Y \wedge \tau' \geq X' \wedge \tau' \geq Y' \wedge \tau' \geq \tau \\ & & \wedge X' = X \wedge Y' = \tau \wedge X' \geq \tau' - 1 \\ g_c & \overset{\mathrm{def}}{=} & \tau \geq X \wedge \tau \geq Y \wedge \tau' \geq X' \wedge \tau' \geq Y' \wedge \tau' \geq \tau \\ & & \wedge X' = \tau \wedge Y' = \tau \wedge \tau = X + 1 \wedge X' \geq \tau' - 1 \\ g_d & \overset{\mathrm{def}}{=} & \tau \geq X \wedge \tau \geq Y \wedge \tau' \geq X' \wedge \tau' \geq Y' \wedge \tau' \geq \tau \\ & & \wedge X' = \tau \wedge Y' = Y \wedge \tau = X + 1 \wedge X' \geq \tau' - 1 \end{array}$$

If $w = t_1 \cdots t_n$ is a sequence of transitions, then g_w is the formula

$$g_w(\vec{X}, \vec{X'}) \overset{\mathrm{def}}{=} \exists \vec{X_0}, \ldots \vec{X_n}. \vec{X_0} = \vec{X} \wedge \vec{X'} = \vec{X_n} \wedge \bigwedge_{i=1}^{n} g_{t_i}(\vec{X_{i-1}}, \vec{X_i})$$

The following lemma states that the guards g_w, indeed correspond to the formulas $\phi(w)$ if the automaton does not "wait in the initial state" before firing the first transition ($\vec{\tau}$ is the vector whose all components are equal to τ):

Lemma 9 ([12]) *Let $w = t_1 \cdots t_n$ be a sequence of transitions and q_0 be the origin state of t_1. $\phi_w(\tau, \vec{X}, \tau', \vec{X'})$ is logically equivalent to*

$$\exists \tau_1, \vec{X_1}. I(q_0)(\tau, \vec{X}) \wedge I(q_0)(\tau_1, \vec{X_1}) \wedge g_w(\tau_1, \vec{\tau} - \vec{X_1}, \tau', \vec{\tau'} - \vec{X'})$$

Proof : Assume $(q_0, \vec{V}) \xrightarrow[\mathcal{A}]{w} (q', \vec{V'})$. We decompose the transition sequence into sections corresponding to moves of the automaton: there are intermediate vectors $\vec{V_1}, \ldots, \vec{V+1}$ and intermediate states $q_1, \ldots, q_{n+1}$ such that $q_0 = q_1$, $q_{n+1} = q'$, $\vec{V_{n+1}} = \vec{V'}$ and

$$(q_0, \vec{V}) \xrightarrow[\mathcal{A}]{} (q_1, \vec{V_1}) \xrightarrow[\mathcal{A}]{t_1} (q_2, \vec{V_2}) \ldots \xrightarrow[\mathcal{A}]{t_{n-1}} (q_n, \vec{V_n}) \xrightarrow[\mathcal{A}]{t_n} (q_{n+1}, \vec{V_{n+1}})$$

Moreover,

- $\vec{V_1} = \vec{V} + \lambda \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}$ for some $\lambda \in \mathbb{R}_+$ and $\vec{V} \models I(q_1)$, $\vec{V_1} \models I(q_1)$
- for every i, $t_i =< q_i, q_{i+1}, a_i, R_i, \phi_i >$, there is some $\lambda \in \mathbb{R}_+$ with $V^j_{i+1} = \lambda if x_j \in R_i$, $V^j_{i+1} = V^j_i + \lambda$ if $x_j \notin R_i$ and $\vec{V_i} \models \phi_i$. Moreover, $\vec{V_{i+1}} \models I(q_{i+1})$.

Now, it suffices to prove that $(q_i, \vec{V_i}) \xrightarrow[\mathcal{A}]{t_i} (q_{i+1}, \vec{V_{i+1}})$ iff $t\vec{a}u - \vec{V_i}, \vec{\tau} - \vec{V_{i+1}} \models g_{t_i}$ and we conclude by a simple induction on the length of w.

For a single transition t, it suffices to notice that

$$\exists \lambda \geq 0. x'_i = \lambda \wedge \tau' = \lambda + \tau \Leftrightarrow \tau' - x'_i = \tau \wedge \tau' \geq \tau \Leftrightarrow X'_i = \tau \wedge \tau' \geq \tau$$

for clocks which are reset, and

$$\exists \lambda \geq 0. x'_i = x_i + \lambda \wedge \tau' = \tau + \lambda \Leftrightarrow \tau' - x'_i = \tau - x_i \wedge \tau' \geq \tau \Leftrightarrow X'_i = X_i \wedge \tau' \geq \tau$$

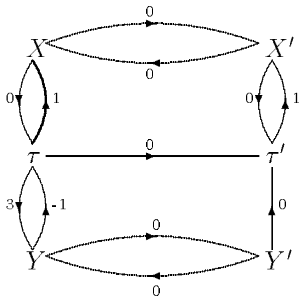

Figure 4: The graph for g_a

□

Second, we represent the constraints g_t as weighted directed (coloured) graphs as follows. The vertices are the clocks $\tau, x_1, \ldots, x_m$ before the transition and the clocks $\tau', x'_1, \ldots, x'_m$ after the transition. There are two kinds of edges: the edges corresponding to strict inequalities and the edges corresponding to non strict inequalities. If g_t contains an inequality $x \leq y + c$, then there is an edge from x to y whose weight is c. Similarly for inequalities between primed variables, for inequalities between primed and unprimed variables and for strict inequalities.

Example 10 The graphs corresponding to transitions a, b, c, d are respectively depicted on figures 4, 5, 6, 7 where the thick edges correspond to strict inequalities and the thin edges correspond to non-strict inequalities.

For instance, on figure 4 there is one thick edge from τ to X. Also, when there are several edges which have the same origin and the same target, we only keep the edge of smallest weight. This is relevant since, when the weight p is smaller or equal to p' then the corresponding constraint c_p implies the constraint $c_{p'}$. On our examples, this occurs in figures 4, 6, 7: there are a priori two edges from Y to τ in figure 4. One has weight 0 and one has weight -1. We only keep the latter.

Now the graphs for g_w are obtained simply by concatenation of the graphs of each transition in w: vertices are now all intermediate clock values and the edges are the union of edges of each single transition.

More formally, if $w = t_1 \cdots t_n$, the graph $G(w)$ associated with g_w has a set of vertices $\bigcup_{i=0}^{n} \mathcal{S}_i$ where each $\mathcal{S}_i = \{\tau^i, X_1^i, \ldots, X_m^i\}$ ($X_1, \ldots, X_m$ are the clocks) and edges and their weights are defined according to the definition of g_w.

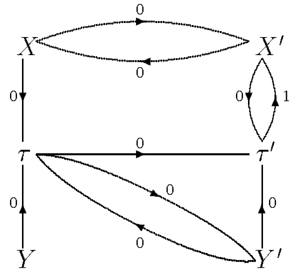

Figure 5: The graph for g_b

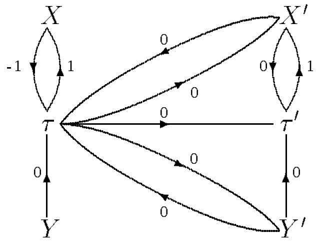

Figure 6: The graph for g_c

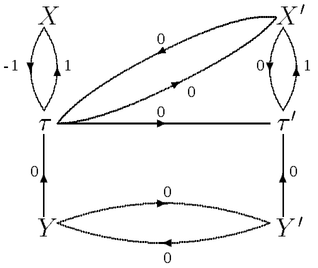

Figure 7: The graph for g_d

Example 11 Coming back to our example, the graph of the constraint g_{abcd} is depicted on figure 8.

As before, when there are two edges with the same source and the same target, we only keep the relevant one.

We can also define concatenation of graph as a morphism on concatenation of words. For any path γ, we write $p(\gamma)$ the total weight of the path.

The first property is a standard property in the theory of one successor:

Property 12 *$\phi(w)$ is satisfiable if and only if there is no cycle γ in $G(w)$ of negative weight.*

If $\phi(w)$ is satisfiable, we define ? (w) as the set of minimal weighted paths between vertices in $\mathcal{S}_0 \cup \mathcal{S}_{|w|}$. With every $\gamma \in$? (w) with origin X_i^j and target X_k^l we associate the constraint $\mathcal{C}(\gamma, w)$ defined by: $x \leq y + p(\gamma)$ if $j = l = 0$, $x' \leq y + p(\gamma)$ if $j = |w|, l = 0$, and similarly for $l = |w|$. We have the property :

Property 13 *If $\phi(w)$ is satisfiable then $\phi(w) \models\!\!\dashv \bigwedge_{\gamma \in \Gamma(w)}(\mathcal{C}(\gamma, w))$*

This allows to give another (equivalent) definition of $\phi(w)$ (with free variables $\vec{X}, \vec{X}'$) :

Definition 14 *If there is a cycle of negative weight in $G(w)$ then $\phi(w) \stackrel{def}{=} -$*
Otherwise $\phi(w) \stackrel{def}{=} \bigwedge_{\gamma \in \Gamma(w)} \mathcal{C}(\gamma, w)$

Given a word w on the transitions of the timed automata, and given two clock names $x, y \in C$, we will denote $(xy)_w$ the class of paths of minimal weight in $G(w)$ between the two vertices X^0 and Y^0. Similarly $(x'y')_w$ will denote any representative of a path of minimal weight in $G(w)$ between the two vertices $X^{|w|}$ and $Y^{|w|}$. $(xy')_w$ and $(x'y)_w$ are defined in a similar way.

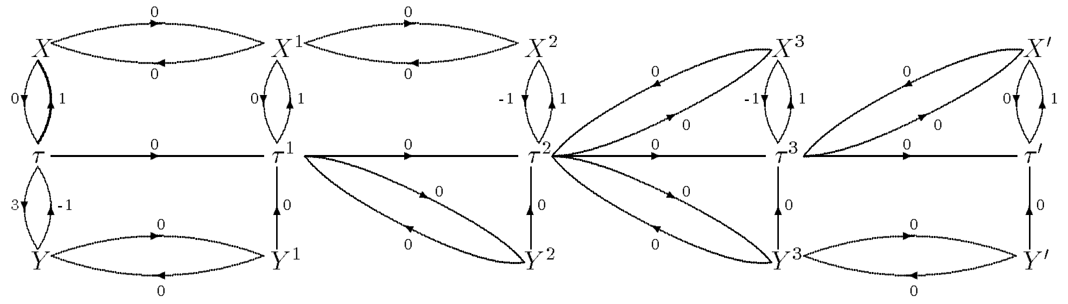

Figure 8: The graph for g_{abcd}

3.1.2 Properties

This section studies the properties induced by the particular form of the constraints resulting from timed automata. Such properties will also support the faithfulness of the approximations which will be used in the definitions of equivalences.

We start with simple properties that hold for paths γ in any given graph $G(w)$, and can be shown simply by induction on the length of γ.

The first property states that infering a relation between two distinct clocks requires at least one relation with τ. (In the graphs $G(w)$ there are no edges relating distinct clocks, except if one of them is τ).

Property 15 *$\forall\gamma$ whose origin is $X_{i_1}^{j_1}$ and target is $X_{i_2}^{j_2}$, with $i_1 \neq i_2$, $\exists j_3$ such that γ contains a τ^{j_3} vertex.*

More precisely:

Property 16 *$\forall\gamma$ whose origin is $X_{i_1}^{j_1}$ and target is$X_{i_2}^{j_2}$, if γ doesn't contain any τ^k vertex, then $i_1 = i_2$ and $p(\gamma) = 0$*

τ is always larger than any other clock and τ is increasing. It follows that:

Property 17 *$\forall i, j_1, j_2, j_1 \leq j_2, \exists\gamma$ from $X_i^{j_1}$ to τ^{j_2}, with $p(\gamma) \leq 0$*

As a consequence of this last property, since the existence of a cycle of negative weight prevents the satisfiability :

Property 18 *If $\phi(w)$ is satisfiable, then any γ from τ^{j_2} to $X_i^{j_1}, j_1 \leq j_2$, is such that $p(\gamma) \geq 0$*

Let K be the maximal constant that appears in the timed automaton.

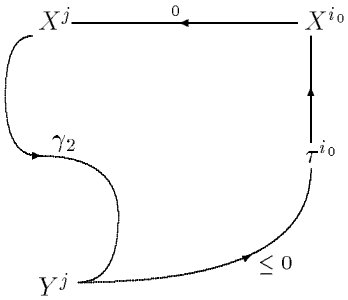

Figure 9: A cycle through τ

Property 19 *For all w such that $\phi(w)$ is satisfiable, for all $\gamma = \gamma_1 \cdot \gamma_2$ in $G(w)$. If γ_1 contains some τ^i, and γ_2 is a path between X^j and Y^j with $j \leq i$, then $p(\gamma_2) \geq -K$*

Proof : Let i_0 be the minimal index i satisfying the hypothesis of the property. By property 17 there is a negative path γ_3 from Y^j to τ^{i_0} and there is a suffix γ_4 of γ_1 which is a path from τ^{i_0} to X^j. (See figure 9).

By minimality of i_0, γ_4 does not contain any τ^i (but its origin τ^{i_0}). By property 16, γ_4 consists in an edge from τ^{i_0} to X^{i_0}, whose weight is smaller than K and of a path of 0 weight: $p(\gamma_4) \leq K$.

Thus we have a cycle $\gamma_4 \cdot \gamma_2 \cdot \gamma_3$ whose weight satisfies:

$$p(\gamma_4 \cdot \gamma_2 \cdot \gamma_3) = p(\gamma_4) + p(\gamma_2) + p(\gamma_3) \leq K + p(\gamma_2) + 0$$

Since $\phi(w)$ is assumed to be satisfiable, the weight of the cycle is not negative. It follows that $p(\gamma_2) \geq -K$ $\Box$

A similar proof gives :

Property 20 *For all w such that $\phi(w)$ is satisfiable, for all $\gamma = \gamma_1 \cdot \gamma_2$ in $G(w)$. If γ_2 contains some τ^j, and γ_1 is a path from X^i to Y^i with $i \leq j$, then $p(\gamma_1) \leq K$.*

We define here a set of particular transitions $\mathcal{T}_s = \bigcup_{X \neq Y, \theta \in [-K,K]} \mathcal{T}(\theta, X, Y)$ which is intended to play the role of an extention of an execution. For every $\alpha \in [-K, 0], \beta \in [0, K]$ and for every clocks X, Y which are both distinct from τ, the transition $t_{X,Y,\alpha,\beta}$ is defined as a reset-free transition with exactly the two additional constraints: $X' \leq \tau' + \alpha$ and $\tau' \leq Y' + \beta$. $\mathcal{T}(\theta, X, Y) = \{t_{X,Y,\alpha,\beta} \mid \alpha + \beta = \theta\}$. When $X = \tau$, we consider transitions built only with the constraint $\tau' \leq Y' + \beta$, and when $Y = \tau$, transitions with only $X' \leq \tau' + \alpha$.

The *decomposition* of a path γ in $G(u \cdot v)$ according to (u, v) is defined as the path concatenation $\gamma = \delta_1 \cdots \delta_p$ with either δ_{2i} in $G(u)$ and δ_{2i+1} in $G(v)$ or δ_{2i} in $G(v)$ and δ_{2i+1} in $G(u)$ (vertices of the paths are implicitly renamed).

Property 21 *Let v,w be words such that $\phi(v \cdot w)$ is satisfiable. If γ, a path of minimal weight, between two vertices $X^{|v|}, Y^{|v|}$ can be decomposed according to (v, w) into $\gamma_0 \cdot \gamma_1 \cdots \gamma_n \cdot \gamma_{n+1}$ such that γ_1 and γ_n have vertices in $\bigcup_{i \in [|v|..|v \cdot w|]} \mathcal{S}_i$ (i.e. in $G(w)$, up to renaming), then $\eta = p(\gamma_1 \cdots \gamma_n) \in [-K, K]$. Moreover, if $\gamma_1 \cdots \gamma_n$ is a path from clock $Z^{|v|}$ to $U^{|v|}$, then there exists $t_\eta \in \mathcal{T}(\eta, Z, U)$ such that $\phi(v \cdot t)$ is satisfiable, and the minimal weight of a path between $X^{|v|}$ and $Y^{|v|}$ in $G(v \cdot t)$ is $p(\gamma)$*

<u>Proof</u> : If $Z = U$, then in particular the weight of the empty path is 0. As $\phi(w)$ is satisfiable, $p(\gamma_1 \cdots \gamma_n) \geq 0$ (by property 12). By minimality $p(\gamma_1 \cdots \gamma_n) = 0 \in [-K, K]$.

If $Z \neq U$, by property 15 we can decompose $\gamma_1 \dots \gamma_n$ into $\delta_1 \cdot t_1 \cdot \delta_2 \cdot t_2 \cdot \delta_3$ in such a way that t_1 (resp. t_2) is an edge whose target (resp. source) is τ^i (resp. τ^j) and δ_1 (resp. δ_3) is the longest prefix (resp. suffix) of γ which does not contain any τ^k. By minimality of the weight of δ_1, and by property 16 we have $p(\delta_1) = p(\delta_3) = 0$. There are now three cases:

If $i < j$, we let $\alpha = p(t_1)$, and $\beta = p(\delta_2 \cdot t_2)$. We then have $\alpha \in [-K, 0]$, and since there is a cycle on τ^i of weight β, we have $\beta \geq 0$. Moreover, since γ is a minimal weighted path, and by property 17, we have $p(\delta_2) \leq 0$. So $\beta \leq p(t_2) \leq K$, and $\beta \in [0, K]$.

If $i > j$, then we let $\alpha = p(t_1 \cdot \delta_2)$ and $\beta = p(t_2)$: we have $\beta \in [0, K]$. By lemma 18 $p(\delta_2) \geq 0$, which implies $\alpha \geq p(t_1) \geq -K$. Since, by property 17, there is a path of negative weight between $X^{|v|}$ and τ^i, and γ is of minimal weight, then $\alpha \leq 0$, and $\alpha \in [-K, 0]$.

If $i = j$, then $\gamma_1 \cdots \gamma_n = \delta_1 \cdot t_1 \cdot t_2 \cdot \delta_3$ (otherwise, there is a cycle on τ^i in γ, which contradicts either the satisfiability of $\phi(v \cdot w)$ or the minimality of γ). We let then $\alpha = p(t_1)$ and $\beta = p(t_2)$. Then $\alpha \in [-K, 0]$ and $\beta \in [0, K]$ and $\eta = \alpha + \beta$.

In all cases, we see that $\eta = \alpha + \beta \in [-K, K]$, hence the first part of the property.

The decomposition $\alpha + \beta = \eta$ showed that there is a t_η such that there is in $G(v \cdot t_\eta)$ a path of weight $p(\gamma)$ between $X^{|v|}$ and $Y^{|v|}$.

Now consider a minimal weighted path γ' between $X^{|v|}$ and $Y^{|v|}$ in $G(v \cdot t_\eta)$. There are again three cases:

If γ' is in $G(v)$, then it was already in $G(v \cdot w)$, hence its weight is larger than $p(\gamma)$ by minimality of γ.

If γ' contains the edge labeled by α , then it also contains the one labeled by β (otherwise it is impossible to return to $U^{|v|}$). Then its weight equals $p(\gamma)$.

If γ' contains the edge labeled by β and not the edge labeled by α , then we decompose γ' into a path $\gamma'_1 \cdot \gamma'_2 \cdot \gamma'_3$ where γ'_1 and γ'_3 are paths of $G(v)$ and γ'_2 is a path in t_η, whose weight is β. Let $T^{|v|}$ be the source of γ' (its target is $U^{|v|}$). We construct the path γ'' from $X^{|v|}$ to $Y^{|v|}$ in $G(v \cdot w)$ as follows: $\gamma'' = \gamma'_1 \cdot \delta_0 \cdot \delta' \cdot \delta'' \cdot \gamma'_3$ where δ_0 is any path from $T^{|v|}$ to $\tau^{|v|}$ (such a path does exist and has a negative weight), δ' is any path from $\tau^{|v|}$ to $\tau^{\max(i,j)}$ and δ'' is the path $\delta_2 \cdot t_2$ in the case where $i > j$ and the path t_2 otherwise. δ' does exist and has a negative weight since $i, j \geq |v|$, because γ_1 and γ_n are in $G(w)$.

$$p(\gamma'') = p(\gamma'_1) + p(\delta_0) + p(\delta') + p(\delta'') + p(\gamma'_3) \leq p(\gamma'_1) + 0 + 0 + \beta + p(\gamma'_3) = p(\gamma')$$

Then, by minimality of γ, $p(\gamma'') \geq p(\gamma)$, hence $p(\gamma) \leq p(\gamma')$.

In any case, $p(\gamma) \leq p(\gamma')$. On the other hand, by definition of α, β, there is a path in $G(v \cdot t_\eta)$ whose weight equals $p(\gamma)$. It follows that $p(\gamma) = p(\gamma')$ by minimality of $p(\gamma')$.

This shows in particular that $\phi(v \cdot t_\eta)$ is satisfiable, thanks to property 12. □

As a consequence of the first part of the above property, if we consider v as the empty word, we have :

Property 22 *If $\phi(w)$ is satisfiable, then any minimal weighted path γ between any X_i^0, X_j^0 is such that $p(\gamma) \in [-K, K]$*

The following property explains the faithfulness of the approximations (truncatures to K) in the definition of equivalences.

Property 23 *Let u, v, be two words such that $\phi(u)$ and $\phi(v)$ are satisfiable. If there is a cycle of negative weight in $G(u \cdot v)$ corresponding to a decomposition with respect to (u, v) : $\gamma_0 \cdot \delta_0 \cdots \gamma_n \delta_n$ with $p(\gamma_0) < -K$, then there is a cycle γ' of negative weight whose decomposition with respect to (u, v) contains two components: $\gamma' = \gamma_0 \cdot \delta'$.*

Proof : Let $X_i^{|u|}$ and $X_j^{|u|}$ be respectively the source and the target of γ_0. Then, $(x_j x_i)_v$ exists, and by property 22, belongs to $[-K, K]$. Choose for δ' any minimal path from X_j^0 to X_i^0 in $G(v)$. □

The property 21 allows to "anticipate" possible relations between clocks values. Because of the symmetry between past and future, we would also like to have an analog property which "guesses" what were the possible relations in the past. This is the purpose of the next property.

We define $\mathcal{P} = \bigcup_{\eta \in [-K,K], x,y \in C} \mathcal{P}(\eta, X, Y)$ where $\mathcal{P}(\eta, X, Y)$ is defined as follows:

If $X \neq \tau$ **and** $Y \neq \tau$, then $\mathcal{P}(\eta, X, Y) \stackrel{\text{def}}{=} \{X \leq \tau + \alpha \wedge \tau \leq Y + \beta \mid \alpha \in [-K, 0], \beta \in [0, K], \alpha + \beta = \eta\}$

If $X = \tau$, then $\mathcal{P}(\eta, \tau, Y) \stackrel{\text{def}}{=} \{\tau \leq Y + \beta \mid \beta \in [0, K], \exists \alpha \in [-K, 0], \alpha + \beta = \eta\}$

If $Y = \tau$, then $\mathcal{P}(\eta, X, \tau) \stackrel{\text{def}}{=} \{X \leq \tau + \alpha \mid \alpha \in [-K, 0], \exists \beta \in [0, K], \alpha + \beta = \eta\}$

The operator $\circ$ takes as arguments a constraint $c \in \mathcal{P}$ and a transition sequence w. $\circ$ is first defined for a single transition t: $c \circ t$ is a transition which is identical to t except that there is an additional precondition c. $c \circ t_1 \cdots c_n$ is defined as $(c \circ t_1) \cdots t_n$.

Property 24 *Let u, v be two transition sequences such that $\phi(u \cdot v)$ is satisfiable. Let γ be a minimal weighted path in $G(u \cdot v)$ from X_1^i to X_2^j where $i, j \geq |u|$. Assume that $\gamma = \gamma_1 \cdot \delta_1 \cdots \delta_q \cdot \gamma_2$ is the decomposition of γ with respect to (u, v). If γ_1, γ_2 both contain some τ^k vertices, then $\eta = p(\delta_1 \cdots \delta_q) \in [-K, K]$, and if δ_1 is a path from X, and δ_q is a path to Y, then there exists $t_\eta \in \mathcal{P}(\eta, X, Y)$ such that in $G(t_\eta \circ v)$, $p(\gamma_1 \cdot t_1) + \eta + p(t_2 \cdot \gamma_4)$ is the weight of the minimal paths between $X_1^{i-|u|}$ and $X_2^{j-|u|}$, and $\phi(t_\eta \circ v)$ is satisfiable.*

Proof : We decompose $\delta_1 \cdots \delta_q$ in $\theta_1 \cdot t_1 \cdot \theta_2 \cdot t_2 \cdot \theta_3$ with θ_1, θ_3 the longest paths which do not contain any τ^k. Let τ^i be the destination of t_1, and τ^j be the source of t_2.

If $i \leq j$ we define $\beta = p(t_2)$ ($\in [0, K]$) and $\alpha = p(t_1 \cdot \theta_2)$. Since $i \leq j$, and γ is a minimal weighted path, we have $p(\theta_2) \leq 0$, since $p(t_1) \in [0, -K]$ we have $\alpha \leq 0$. If a is the weight of the last transition of γ_1 that contains a τ^k, we have a cycle of weight $a + p(t_1 \cdot \theta_2) \geq 0$, so $\alpha \geq -K$. Moreover $p(\delta_1 \cdots \delta_q) \in [-K, K]$.

Otherwise, if $i > j$, then we let $\alpha = p(t_1)$, and $\beta = p(\theta_2 \cdot t_2)$.

It is easy to see that a cycle in $G(t_\eta \circ v)$ should also be a cycle in $G(u \cdot v)$. □

3.1.3 Definition of the equivalence relation

In this section we will introduce the definitions which are sufficient to built the right-compatible equivalence relation $\sim$. The proofs of the properties of $\sim$ are delayed until the next sections

$\sim$ is defined as a finite union of relations $\sim_i$ on Σ^* (Σ is the set of transitions).

$w_1 \sim_0 w_2$ ensures that reachability relations induced by the graphs $G(w_1)$ and $G(w_2)$ (regardless to their weights) are identical.

Definition 25 *We will say that two words u, v are $\sim_0$-equivalent when for any clocks x, y, if a path $(xy)_u$, $(xy')_u$, $(x'y)_u$, or $(x'y')_u$ exists in $G(u)$, then it is the case in $G(v)$. Moreover, if a clock x is never reset in u then it is the case in v.*

$\sim_1$ guarantees that the weights of the edges relating initial clock values in the graphs $G(u)$ and $G(v)$ are identical.

Definition 26 *We will say that two words u, v are $\sim_1$-equivalent when, for all clock $x, y, p((xy)_u) = p((xy)_v)$*

Similarly we will define an equivalence for paths $(x'y')$. However, in order to preserve the finite index property, we have to truncate the value of $(x'y')$ when its absolute value is greater than the largest constant in the automaton K.

Definition 27 *We will say that two words u, v are $\sim_2$-equivalent when, for all clock x, y, if $|p((x'y')_u)| \leq K$ then $p((x'y')_u) = p((x'y')_v)$, and $p((x'y')_u) > K \Leftrightarrow p((x'y')_v) > K$ and $p((x'y')_u) < -K \Leftrightarrow p((x'y')_v) < -K$.*

$\sim_2$ is an equivalence relation.

$\sim_1 \cup \sim_2$ is not right compatible (see the examples in the introduction of section 3). Hence we define two more equivalence relations which will restore this property.

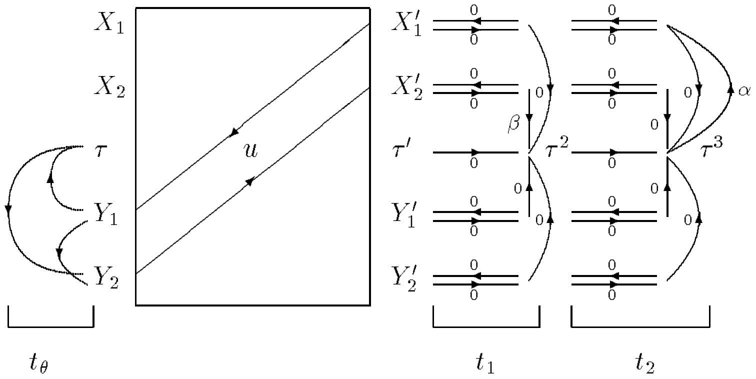

Figure 10: The additional edges in the definition of $\sim_5$

Informally, these two relations are intended to anticipate the impact of possible future transitions on $\sim_1, \sim_2$: $\sim_3$ will restore the right-compatibility of $\sim_2$, and $\sim_4$ restores the right-compatibility of $\sim_1$.

Definition 28 *We will say that two words u, v are $\sim_3$-equivalent when, for all clock $x, y, z_1, z_2, \forall\theta \in [-K, K]$, if there is t in $\mathcal{T}(\gamma, Z_1, Z_2)$ such that $\phi(u \cdot t)$ is satisfiable, then we have :*

1. *$p((x'z'_1)_u) + \theta + p((z'_2y')_u) \leq ((x'y')_u)$ and $p((x'z'_1)_u) + \theta + p((z'_2y')_u) \in [-K, K]$ iff the same property holds replacing u with v and $p((x'z'_1)_u) + \theta + p((z'_2y')_u) = p((x'z'_1)_v) + \theta + p((z'_2y')_v)$*
2. *$p((x'z'_1)_u) + \theta + p((z'_2y')_u) < -K \Leftrightarrow p((x'z'_1)_v) + \theta + p((z'_2y')_v) < -K$*

Definition 29 *We will say that two words u, v are $\sim_4$-equivalent when, for all clocks $x_1, y_1, x_2, y_2, \forall\theta \in [-K, K]$, if there is a t in $\mathcal{T}(\theta, Y_1, Y_2)$ such that $\phi(u \cdot t)$ is satisfiable, and $p((x_1y'_1)_u)+\theta+p((y'_2x_2)_u) \leq p((x_1x_2)_u)$ [3], then $p((x_1y'_1)_v)+\theta+p((y'_2x_2)_v) \leq p((x_1x_2)_v)$ and $p((x_1y'_1)_u) + p((y'_2x_2)_u) = p((x_1y'_1)_v) + p((y'_2x_2)_v)$*

The union of $\sim_i$ for $i = 0..4$ is already a finite index right-compatible equivalence relation (this will be proved in section 3.1.4). This is however not sufficient, as shown by example 7. We need three more equivalence relations which relate now the initial and final clock values.

Definition 30 *Let $Q(u, x_1, x_2, y_1, y_2, \theta, \alpha, \beta)$ be the following property of a word $u \in \Sigma^*$, clocks x_1, x_2, y_1, y_2 such that $x_1 \neq x_2$ and $y_1 \neq y_2$, $\alpha \in [0, K]$, $\beta \in [-K, 0]$ and $\theta \in [-K; K]$:*

[3] if $(x_1x_2)_u$ doesn't exist, then its value is considered as being $+\infty$

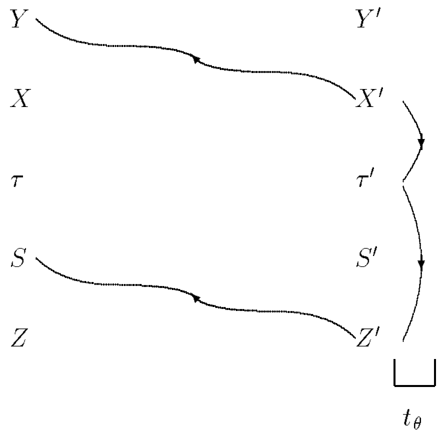

Figure 11: The difference of backwards paths is preserved under $\sim_6$.

> *$\mathcal{Q}(u, x_1, x_2, y_1, y_2, \theta, \alpha, \beta)$ if there is t_θ in $\mathcal{P}(\theta, Y_1, Y_2)$ and two reset-free transitions t_1, t_2 containing at most one additional constraint which belongs to $\{\tau' \leq X_1' + \alpha, X_2' \leq \tau' + \beta\}$, such that $\phi(t_\theta \circ u \cdot t_1 \cdot t_2)$ is satisfiable and there is a minimal path of weight $\alpha + p((x_1'y_1)_u) + \theta + p((y_2x_2')_u) + \beta$ using the edges of t_1, t_2in $G(t_\theta \circ u \cdot t_1 \cdot t_2)$.*

We will say that two words u, v are $\sim_5$-equivalent when, for all clocks $x_1, y_1, x_2, y_2, x_1 \neq x_2, y_1 \neq y_2, \forall \theta \in [-K, K], \forall \alpha \in [0, K], \forall \beta \in [-K, 0]$,

1. *$\mathcal{Q}(u, x_1, x_2, y_1, y_2, \theta, \alpha, \beta)$ iff $\mathcal{Q}(v, x_1, x_2, y_1, y_2, \theta, \alpha, \beta)$*
2. *when $\mathcal{Q}$ holds for both u and v, then $p((x_1'y_1)_u) + p((y_2x_2')_u) = p((x_1'y_1)_v) + p((y_2x_2')_v)$.*

In other words, $u \sim_5 v$ if, adding to u (resp. v) the transitions as depicted on figure 10, we get the same weights for the minimal paths from τ^3 to τ^2 in u, v (through the additional edges).

The relations $\sim_i$ defined so far already allow to derive some commutation properties: given a transition sequence $u \cdot v$ such that $\phi(u \cdot t)$ is satisfiable and $u \sim_i v$, for $i = 0..5$, then $\phi(v \cdot u)$ is satisfiable, and for all clocks x, y, if γ is a representative of $(xy)_{u \cdot v}$, then there is a representative γ' of $(xy)_{v \cdot u}$ such that $p(\gamma) = p(\gamma')$. And, roughly, the same property holds for $(x'y')_{u \cdot v}$ and $(x'y')_{v \cdot u}$. However, we are also interested in constraints (xy') and $(x'y)$. Therefore, we introduce two more refinements of the equivalence relation.

Definition 31 *We will say that two words u, v are $\sim_6$-equivalent when,*

1. *for all clocks x, y, z such that $p((x'\tau')_u) \geq -K$, if there is $t_\theta \in \mathcal{T}(\theta, X, Z)$ such that $p((x'y)_u) = p((x'y)_{u \cdot t_\theta})$ then the same property holds true, replacing u with v.*

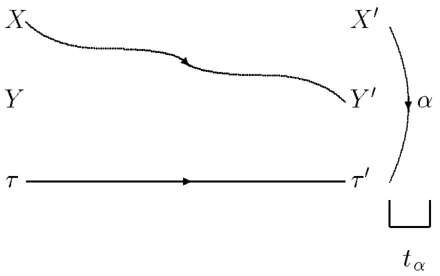

Figure 12: The difference of forward paths with $(\tau\tau')$ is preserved under $\sim_7$.

2. *for all clocks x, y, z, s such that $p((x'\tau')_u) \geq -K$ and $p((z'\tau')_s) \geq -K$, if there a transition $t_\theta \in \mathcal{T}(\theta, X, Z)$ and a transition $t_\gamma \in \mathcal{T}(\gamma, Z, X)$ such that $p((x'y)_{u \cdot t_\theta}) = p((x'y)_u)$ and $p((z's)_{u \cdot t_\gamma}) = p((z's)_u)$, then*

$$p((x'y)_u) - p((z's)_u) = p((x'y)_v) - p((z's)_v)$$

This definition is depicted on figure 11.

Definition 32 *We will say that two words u, v are $\sim_7$-equivalent when, for all clocks x, y, if there is $\alpha \in [-K; K]$ and $t_\alpha \in \mathcal{T}(\alpha, Y, \tau)$ such that $p((x\tau')_{u \cdot t_\alpha}) = p((xy')_u) + \alpha$, then $p((x\tau')_{v \cdot t_\alpha}) = p((xy')_v) + \alpha$ and $p((xy')_u) - p((\tau\tau')_u) = p((xy')_v) - p((\tau\tau')_v)$*

The notations for $\sim_7$ are displayed on figure 12.

Definition 33 *$\sim$ is the union for $i \in [0..7]$ of $\sim_i$.*

3.1.4 $\sim$ is a finite index right-compatible equivalence on transition sequences

This section is entirely devoted to the proof of the following lemma:

Lemma 34 *$\sim$ is an equivalence relation of index $O(K^{n^4})$ where n is the number of clocks and, for every $u, v, w \in \Sigma^*$, if $u \sim v$, then $u \cdot w \sim v \cdot w$.*

Property 35 *For every $u, v, t \in \Sigma^*$, if $u \sim v$ and $\phi(u \cdot t)$ is satisfiable, then $u \cdot w \sim_0 v \cdot w$*

The proof is straightforward.

Property 36 *For every $t, u, v \in \Sigma^*$, if $\phi(u \cdot t)$ is satisfiable and $u \sim v$, then $u \cdot t \sim_1 v \cdot t$.*

<u>Proof</u> : Let x, y be two clock names. And let $\mu = p((xy)_{u \cdot t})$ and $\nu = p((xy)_{v \cdot t})$, we show that $\nu \leq \mu$, then by symmetry we will have $\nu = \mu$, meaning that $u \cdot t \sim_1 v \cdot t$.

Let γ be a representative of $(xy)_{u \cdot t}$. If γ is a path in $G(u)$ it is also a representative of $(xy)_u$, and since $u \sim_1 v$ there is a γ', representative of $(xy)_v$ with the same weight. This path is also in $G(v \cdot t)$, so $\nu \leq \mu$

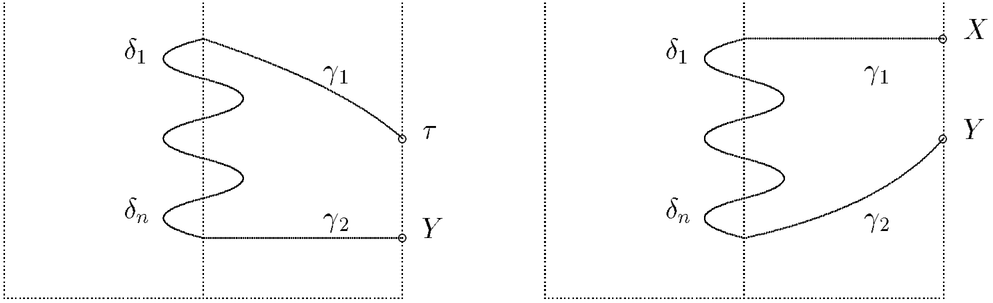

Figure 13:

If γ can be decomposed in $\gamma_1 \cdot \delta_1 \cdots \delta_n \cdot \gamma_2$ where γ_1 is a representative of some $(xz'_1)_u$ and γ_2 of some $(z'_2y)_u$, then by property 21, and since $u \sim_4 v$, the sum $(xz'_1)_u + (z'_2y)_u$ is equal to $(xz'_1)_v + (z'_2y)_v$. Now, by properties 19, 20, and since $u \sim_1 v$ and $u \sim_2 v$, then there are δ'_i of same weight than δ_i such that for a representative γ'_1 of $(xz'_1)_v$ and γ'_2 of $(z'_2y)_v$ the path $\gamma'_1 \cdot \delta'_0 \cdots \delta'_n \cdot \gamma'_2$ is a path in $G(v \cdot t)$ with origin X^0 and destination Y^0. It follows again that $\nu \leq \mu$. □

Property 37 *For every $u, v \in \Sigma^*$, for every $t \in \Sigma$, if $\phi(u \cdot t)$ is satisfiable and $u \sim v$, then $u \cdot t \sim_2 v \cdot t$.*

Proof : Let x, y be two clock names. And let $\mu = p((x'y')_{u \cdot t})$ and $\nu = p((x'y')_{v \cdot t})$. If $\mu \in [-K, K]$, we show, as before, that $\nu \leq \mu$. If $\mu \leq -K$, we show that $\nu \leq -K$. If $\mu > K$, then $\nu > K$, as a consequence of property 35.

So, let γ be a path in $G(u \cdot t)$ with origin $X^{|u \cdot t|}$ and target $Y^{|u \cdot t|}$. We decompose $\gamma = \gamma_1 \cdot \delta_1 \cdots \delta_n \cdot \gamma_2$ according to (u, t). If all the δ_i are in $[-K, K]$, then since $u \sim_2 v$, we can find δ'_i of same weight in $G(v \cdot t)$ and paths γ'_1, γ'_2 such that $\gamma'_1 \cdot \delta'_1 \cdots \delta'_n \cdot \gamma'_2$ is a path of weight $p(\gamma)$ in $G(v \cdot t)$ between $X^{|v \cdot t|}$ and $Y^{|v \cdot t|}$. By properties 19 and 20, this occurs each time γ_1 and γ_2 contain some τ^i.

Assume now that either γ_1 or γ_2 does not contain any τ^i. If x and y are both distinct from τ, then considering the fact that in t a clock is either reset or constant, and thanks to property 21, we can conclude using the fact that $u \sim_3 v$. Only the two cases depicted in figure 13 remain to be considered.

Consider first the case where $x = \tau$. Then every $\delta_i, i \in [1..n-1]$ is in $[-K, K]$ and either $\delta_n \in [-K, K]$ or $\delta_n > K$. We have already studied the first case. For the second case, as $\phi(u \cdot t)$ is satisfiable, the path $\gamma_1 \cdot \delta_1 \cdots \delta_{n-1}$ extended with a zero path to $X^{|u \cdot t|}$ is a cycle. Hence its weight is positive. It follows that $\mu > K$ when $\delta_n > K$.

It remains only to consider the case where $y = \tau$ and γ_1 does not contain any τ^i, and $\delta_1 \leq -K$. Let z be the clock such that δ_1 is a path between $X^{|u|}$ and $Z^{|u|}$. As there is a zero path between $Z^{|u|}$ and $Y^{|u \cdot v|}$, and because γ_1 is a minimal weighted path, we have $\Sigma_{i=2..n} p(\delta_i) + p(\gamma_2) \leq 0$. In particular $\mu \leq -K$. Using $u \sim_2 v$, we can exhibit a path $\gamma'_1 \cdot \delta'_1 \cdots \delta'_n \cdot \gamma'_2$ of weight $< -K$ between $X^{|v \cdot t|}$ and $Y^{|v \cdot t|}$. It follows that $\nu < -K$. □

Property 38 *For every $u, v \in \Sigma^*$, for every $t \in \Sigma$, if $u \sim v$ and $\phi(u \cdot t)$ is satisfiable, then $u \cdot t \sim_3 v \cdot t$*

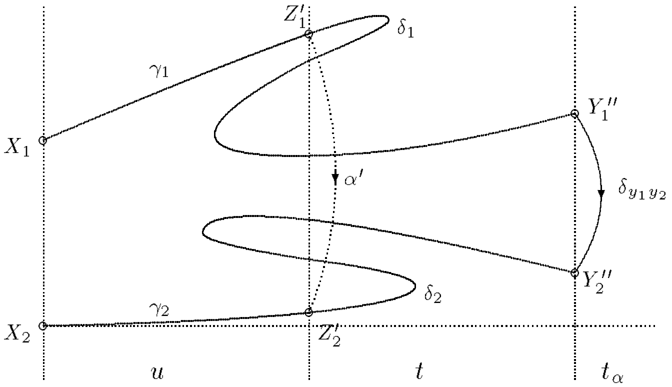

Figure 14: The proof of property 39

Proof : It is essentially the same proof as for $\sim_2$, replacing t by $t \cdot t_\gamma$ for some $t_\gamma \in \mathcal{T}(\gamma, Z_1, Z_2)$. The comparison with $p((x'y')_u)$ holds because we have already shown $u \cdot t \sim_2 v \cdot t$. □

Property 39 *For every $u, v \in \Sigma^*$, for every $t \in \Sigma$, if $u \sim v$ and $\phi(u \cdot t)$ is satisfiable, then $u \cdot t \sim_4 v \cdot t$.*

Proof :

Let x_1, x_2, y_1, y_2 be clocks and $t_\alpha \in \mathcal{T}(\alpha, Y_1, Y_2)$ such that the property of $\sim_4$ holds for $u \cdot t$. Let $\mu = p((x_1x_2)_{u\cdot t\cdot t_\alpha})$ and $\nu = p((x_1x_2)_{v\cdot t\cdot t_\alpha})$. As before we show that $\nu \leq \mu$. Let γ be a representative of $(x_1x_2)_{u\cdot t\cdot t_\alpha}$. γ can be decomposed according to $(u, t \cdot t_\alpha)$ into $\gamma_1 \cdot \delta_1 \cdot \delta_{y_1y_2} \cdot \delta_2 \cdot \gamma_2$ (see figure 14). By property 21, if γ_1 is a path relating X_1 and Z'_1 and γ_2 a path relating Z'_2 and X_2, there is a $t_{\alpha'} \in \mathcal{T}(\alpha', Z_1, Z_2)$ such that $\alpha' = p(\delta_1) + \alpha + p(\delta_2) \in [-K, K]$ and $p((x_1z'_1)_u) + \alpha' + p((z'_2x_2)_u) \leq p((x_1x_2)_u)$. Now, since $u \sim_4 v$, $p((x_1z'_1)_v) + \alpha' + p((z'_2x_2)_v) \leq p((x_1x_2)_v)$ and

$$p((x_1z'_1)_v)+p((z'_2x_2)_v) = p((x_1z'_1)_u)+p((z'_2x_2)_u) = p((x_1y'_1)_{u\cdot t})-p(\delta_1)+p((y'_2x_2)_{u\cdot t})-p(\delta_2)$$

Now, since $u \sim_2 v$ and by property 21, there are paths δ'_1 and δ'_2 in $G(v \cdot t)$ relating respectively Z'_1 and Y''_1 on one hand and Y''_2 and Z'_2 on the other hand, such that $p(\delta'_1) = p(\delta_1)$ and $p(\delta'_2) = p(\delta_2)$. Therefore

$$\begin{aligned} \nu &\leq \alpha + p(\delta'_1) + p(\delta'_2) + p((x_1z'_1)_v) + p((z'_2x_2)_v) \\ &= \alpha + p(\delta_1) + p(\delta_2) + p((x_1y'_1)_{u\cdot t}) - p(\delta_1) + p((y'_2x_2)_{u\cdot t}) - p(\delta_2) \\ &= \mu \end{aligned}$$

Which, by symmetry, proves $\mu = \nu$.

Finally, a best path from X_1 to X_2 goes through t_α in $G(u \cdot t)$ iff the same property holds true for $G(v \cdot t)$, because $u \sim_1 v$.

□

This already shows that $\bigcup_{i=0}^{4} \sim_i$ is right-compatible:

Property 40 *For every $u, v, w \in \Sigma^*$, if $u \sim v$, then for every $i \in [0..4]$, $u \cdot w \sim_i v \cdot w$.*

Proof : Inspecting the proof of properties 35, 36, 37, 38, 39, we only used $u \sim_i v$ for $i \in [0..4]$. Hence we can conclude by an induction on $|w|$ using these properties. □

Property 41 *For all $u \sim v$, and all $t \in \Sigma$, such that $\phi(u \cdot t)$ is satisfiable, we have $u \cdot t \sim_5 v \cdot t$*

Proof : The proof can be followed on figure 15. Let $x_1, x_2, y_1, y_2, t_\eta, t_1, t_2$ be satisfying $\mathcal{Q}$. We also define $\gamma_\eta, \gamma_\alpha$ and γ_β as being the paths allowed by the introductions of the constraints in t_η, and the constraints labeled by α, β. Let γ be the minimal weighted path : $\gamma_\alpha \cdot (x'_1 y_1)_{u\cdot t} \cdot \gamma_\eta \cdot (y_2 x'_2)_{u \cdot t} \cdot \gamma_\beta$. We decompose $\gamma = \delta_1 \cdot (z'_1 y_1)_u \cdot \gamma_\eta \cdot (y_2 z'_2)_u \cdot \delta_2$, with δ_1, δ_2. δ_1 and δ_2 contain at least one τ vertex. Let t'_1 be the transition which consists only (besides the mandatory constraints) of the constraint $\tau \leq Z_1 + \alpha'$ where α' is the label of the last edge of δ_1 whose source is some τ vertex (then, necessarily, its target is some Z_1 vertex). Similarly, let t'_2 be the transition which only consists of the constraint $Z_2 \leq \tau + \beta'$ where β' is the label of the first edge of δ_2 whose target is some τ vertex (then, necessarily, its source is some Z_2 vertex). $\mathcal{Q}$ is satisfied for $u, y_1, y_2, z_1, z_2, \eta, \alpha', \beta'$, hence, since $u \sim_5 v$, it is also satisfied replacing u with v and, moreover,

$$p((z'_1 y_1)_u) + p((y_2 z'_2)_u) = p((z'_1 y_1)_v) + p((y_2 z'_2)_v)$$

Now δ_1 (resp. δ_2) can be decomposed as $\delta_1 = \theta_1 \cdot \pi_1$ (resp. $\delta_2 = \pi_2 \cdot \theta_2$) where the θ_i are maximal length paths within $G(t)$. Moreover, for each i, π_i can be decomposed according to (u, t) and, thanks to properties 20 and 19, each piece of the path within $G(u)$ has a weight in $[-K, K]$. Now, since $u \sim_2 v$, there are paths in $G(v)$ with the same sources, targets and weights as these pieces. This shows that there are paths π'_i in $G(v \cdot t)$ which have the same sources, targets and weights as their corresponding π_i.

Now, putting everything together, we can construct a path γ' in $G(t_\eta \circ v \cdot t_1 \cdot t_2)$ with the same source, target and weight as γ. If γ'' is the minimal weight path in $G(t_\eta \circ v \cdot t_1 \cdot t_2)$ between $\tau^{|v|+|t|+1}$ and $\tau^{|v|+|t|+2}$ (or the converse, depending on the ordering between t_1 and t_2), then $p(\gamma'') \leq p(\gamma)$. By symmetry, this inequality is an equality. □

Property 42 *When the difference $\delta = p((x'y)_u) - p((z's)_u)$ has to be preserved according to $\sim_6$ (i.e. we assume that there a transition $t_\theta \in \mathcal{T}(\theta, X, Z)$ such that $p((x'y)_u) = p((x'y)_{u \cdot t_\theta})$ and a transition $t_\gamma \in \mathcal{T}(\gamma, Z, X)$ such that $p((z's)_u) = p((z's)_{u \cdot t_\gamma}))$ then it belongs to $[-2K, 2K]$.*

Proof :

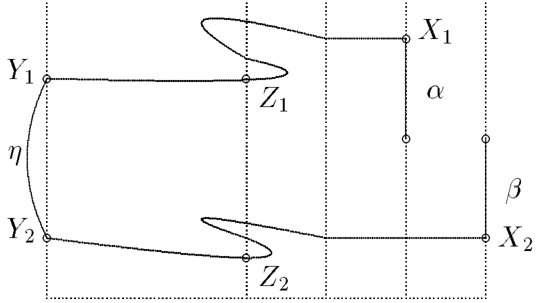

Figure 15:

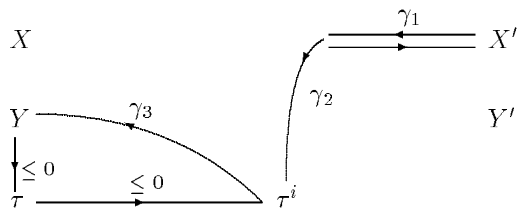

Figure 16: Case 1 in the proof of property 46.

1.We have always $p((x'y)_u) \geq -K$: If x is never reset in u, then $y = x$ and we have $p((x'y)_u) = 0 \geq -K$.

If it is not the case, we can decompose a representative of $(x'y)_u$ into $\gamma_1 \cdot \gamma_2 \cdot \gamma_3$ where γ_2 consists of the single edge, which is the first whose target is some τ^i in $G(u)$. Then we have $p(\gamma_1) = 0, p(\gamma_2) \in [-K, 0]$, and $p(\gamma_3) \geq 0$ since there is always a path of weight 0 between Y and τ^i. Hence $p((x'y)_u) \geq -K$ (see figure 16).

2. We have $p((x'y)_u) - p((z's)_u) \leq 2K$:

If there is a representative of $(x'y)_u$ that contains no vertex τ^i, then we use the fact that $p((z's)_u) \geq -K$: $p((x'y)_u) = 0$.

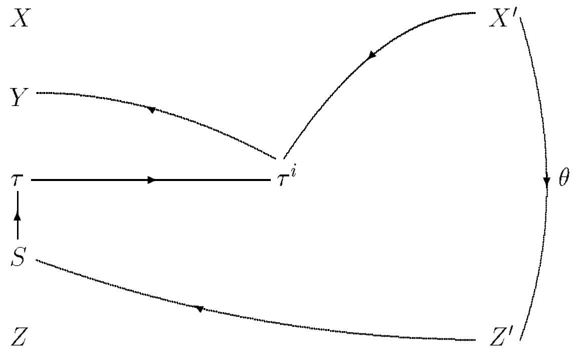

Figure 17: Case 2 in the proof of property 46.

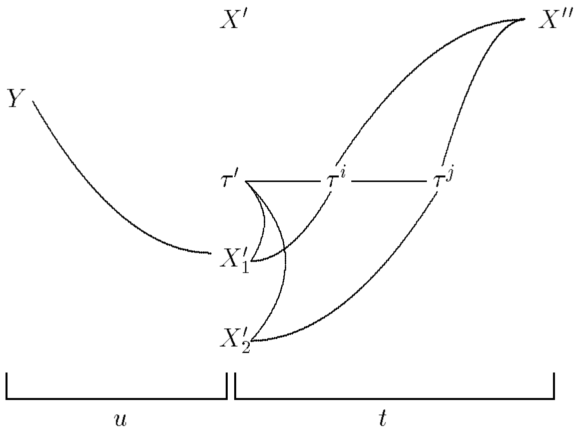

Figure 18: Notations in the proof of property 43.

In the other case, there is a path in u from S to Y (at least the one which goes through τ and τ^i). Then there is a path from X' to Y in $u \cdot t_\theta$ which goes through Z' and S (see figure 17). Moreover, by hypothesis, the weight of the best path from X' to Y in $G(u \cdot t_\theta)$ is equal to the weight of the best path from X' to Y in $G(u)$. Therefore

$$p((x'y)_u) \leq \theta + p((z's)_u) + p((sy)_u)$$

Hence $(x'y)_u - (z's)_u \leq 2K$.

As x, y and z, s play a symmetrical role, we have $(x'y)_u - (z's)_u \in [-2K, 2K]$. □

Property 43 *Let $u \sim v$, and $t \in \Sigma^*$ such that $\phi(u \cdot t)$ is satisfiable. Then, $\forall x, y$, such that $(x'\tau')_{u \cdot t} \geq -K$, every representative γ of $(x'y)_{u \cdot t}$ can be decomposed in $\delta \cdot \gamma_1$ with γ_1 a representative of $(x'_1 y)_u$ in such a way that there is a representative γ'_1 of $(x'_1 y)_v$, and a δ' of same weight as δ such that $\delta' \cdot \gamma'_1$ is a representative of $(x'y)_{v \cdot t}$.*

Proof : First, by property 40, if there is a path from X'' to Y in $u \cdot t$, then this is the case in $v \cdot t$. Moreover, if $p((x'y)_{u \cdot t}) \geq -K$ then $p((x'y)_{v \cdot t}) \geq -K$.

We decompose γ and a representative γ_v of $(x'y)_{v \cdot t}$ as follows:

- $\gamma = \delta \cdot \gamma_1$ and $\gamma_v = \delta_v \cdot \gamma'_2$ where γ_1 (resp. γ'_2) are paths in $G(u)$ (resp. $G(v)$)
- the last edges of δ, δ_v do not belong to $G(t)$ (i.e. γ_1, γ'_2 have been chosen maximal among the paths satisfying the above condition)

Then γ_1 and γ'_2 are respectively representatives of $(x'_1 y)_u$ and $(x'_2 y)_v$ for some x'_1, x'_2.

If δ and δ_v do not contain any τ^i , then $x = x_1 = x_2 = y$ and we have the property indeed.

If δ contains some τ^i and δ_v contains some τ^j , then we may assume without loss of generality that τ^i and τ^j do not belong to $G(u)$ (indeed, consider the longest suffix of δ which is in t; if such a suffix, which is not empty, does not contain any τ^i, then it must be a null path and can be removed).

Now, by property 19, $p((x_1'\tau')_u) \geq -K$ and $p((x_2'\tau')_u) \geq -K$. Moreover, there is a path from X_1' to X_2' in t: consider the path going through τ', τ^i, τ^j (see figure 18). Symmetrically, there is a path from X_2' to X_1' in t.

By property 22, $p((x_1x_2)_t) \in [-K, K]$ and $p((x_2x_1)_t) \in [-K, K]$.

Let $t_\theta \in \mathcal{T}(p((x_1x_2)_t), X_1', X_2')$ (resp. $t_\gamma \in \mathcal{T}(p((x_2x_1)_t), X_2', X_1')$). $p((x_1'y)_{u\cdot t}) = p((x_1'y)_u)$ by definition of the decomposition of $(x'y)_{u\cdot t}$ (and similarly $p((x_2'y)_{v\cdot t}) = p(x_2'y)_v)$. It follows that $p((x_1'y)_{u\cdot t_\theta}) = p((x_1'y)_u)$ (resp. $p((x_2'y)_{v\cdot t_\gamma}) = p(x_2'y)_v$, which implies, by the first part of the definition of $\sim_6$ that $p((x_2'y)_{u\cdot t_\gamma}) = p((x_2'y)_u)$). The conditions of $\sim_6$ are fulfilled: $u \sim_6 v$ implies that

$$(P1) \qquad p((x_1'y)_u) - p((x_2'y)_u) = p((x_1'y)_v) - p((x_2'y)_v)$$

Now, let $\zeta_0 \cdot \zeta_1 \cdots \zeta_n$ be the decomposition of δ according to (u, t). Consider a path ζ_i in $G(u)$. Either $i = 1$ and ζ_0 is a null-weight path, in which case, the property $p((x'\tau')_{u\cdot t}) \geq -K$ implies that $p(\zeta_i) \geq -K$, or else $\zeta_0 \cdots \zeta_{i-1}$ contains a τ^j vertex which does not belong to $G(u)$. Then, by property 19, we have again $p(\zeta_i) \geq -K$. Similarly, since the last edge of δ is not in u, for ζ_n is a path in $G(t)$ and it must contain some τ^j vertex. Then, by property 20, $p(\zeta_i) \leq K$.

Now, since $u \sim_2 v$, there are paths ζ_i' in $G(v \cdot t)$ such that, for all i, ζ_i and ζ_i' have the same target, the same origin and the same weight. It follows that there is a path δ' in $G(v \cdot t)$ from X'' to X_1' and such that $p(\delta') = p(\delta)$.

Symmetrically, we also have $p((x'\tau')_{v\cdot t}) \geq -K$ since $u \sim v$ and by property 40. Hence there is a path δ_v' in $G(u \cdot t)$ from X'' to X_2' and such that $p(\delta_v') = p(\delta_v)$.

Summing up what we have:

$$\begin{aligned}
p((x'y)_{v\cdot t}) &\leq p(\delta') + p((x_1'y)_v) && \text{(best path)}\\
&= p(\delta) + p((x_1'y)_v) && \text{(By construction)}\\
&= p(\delta) + p((x_2'y)_v) + p((x_1'y)_u) - p((x_2'y)_u) && \text{(By } (P1))\\
&= p((x'y)_{u\cdot t}) + p((x_2'y)_v) - p((x_2'y)_u) && \text{(By definition of } \delta)\\
&\leq p(\delta_v') + p((x_2'y)_v) && \text{(best path)}\\
&= p(\delta_v) + p((x_2'y)_v) && \text{(By construction)}\\
&= p((x'y)_{v\cdot t}) && \text{(By definition of } \delta_v)
\end{aligned}$$

It follows that all these expressions are equal. In particular $p((x'y)_{v\cdot t}) = p(\delta') + p((x_1'y)_v)$, which is the desired property.

δ contains some τ^i but δ_v does not contain any τ^j. Then $x_2' = x'$ and x is not reset by t. Let δ_v^{-1} the 0-weight path in t from X to X'. $\delta_v^{-1} \cdot \delta$ is a path in $u \cdot t$ from $X^{|u|}$ to $X_1^{|u|}$ of weight $p(\delta)$. Moreover, since δ is a path of minimal weight between $X^{|u\cdot t|}$ and $X_1^{|u|}$, $p(\delta) \leq p(\delta_v) + p(\gamma)$ for any path from $X^{|u|}$ to $X_1^{|u|}$ in $u \cdot t$. Since $p(\delta_v) = p(\delta_v^{-1}) = 0$, it follows that $\delta_v^{-1} \cdot \delta$ is one minimal weight path in $u \cdot t$ between $X^{|u|}$ and $X_1^{|u|}$. Now, by property 21, and since the first and the last edge of $\delta_v^{-1} \cdot \delta$ belong to $G(t)$, we have $p(\delta) \in [-K, K]$.

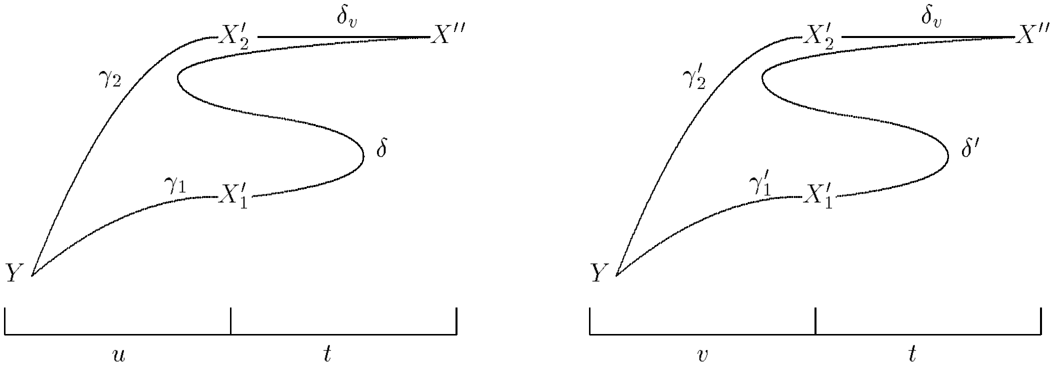

Figure 19: The paths in the second case of the proof of property 43

Let $t_\delta \in \mathcal{T}(p(\delta), x, x_1)$. $p((x'y)_v) = p((x'y)_{v\cdot t_\delta})$, by definition of δ_v. Hence, since $u \sim_6 v$, we must have also $p((x'y)_u) = p((x'y)_{u\cdot t_\delta})$. This implies in particular that $p(\delta) = 0$. On the other hand, as in the previous case, since δ contains some τ^i with $i \geq |u|$, we can construct a path δ' in $v \cdot t$, whose source is $X^{|v\cdot t|}$, whose target is $X_1^{|v|}$ and whose weight is $p(\delta) = 0$. If γ'_1 is any representative of $(x'_1y)_v$ and γ_2 is any representative of $(x'_2y)_u$ (see figure 19 for a graphical representation of the paths), we get

$$p(\gamma'_1) = p(\gamma'_1) + p(\delta') \geq p((x'y)_{v\cdot t}) = p(\gamma'_2)$$

and symmetrically,

$$p(\gamma_2) = p(\gamma_2) + p(\delta_v) \geq p((x'y)_{u\cdot t}) = p(\gamma_1)$$

On the other hand, if $t_\delta^{-1} \in \mathcal{T}(0, X_1, X)$, then $p((x'_1y)_{u\cdot t_\delta^{-1}}) = p((x'_1y)_u)$ (because $p(\gamma_1) \leq p(\gamma_2)$) and $p((x'y)_u) = p((x'y)_{u\cdot t_\delta})$ as we have seen before. Hence, using $u \sim_6 v$, we must have $p(\gamma'_2) - p(\gamma'_1) = p(\gamma_2) - p(\gamma_1)$. The differences are both positive and negative, hence they must be 0.

Therefore $\delta' \cdot \gamma'_1$ is a representative of $(x'y)_{v\cdot t}$.

δ_v contains some τ^i and δ does not contain any τ^j. This case is similar to the previous one. It is actually simpler, since we can choose $\delta' = \delta$ instead of the above construction. Except for this, the proof is the same.

□

Property 44 *Let $u \in \Sigma^*$. If $p((x'\tau')_u) \geq -K$, then for every $t_\theta \in \mathcal{T}(\theta, X, Z)$, $p((x'\tau')_{u\cdot t_\theta}) \geq -K$.*

<u>Proof</u> : Let γ be a representative of $(x'\tau')_{u\cdot t_\theta}$. If The first edge γ is an edge from $X^{|u|+1}$ to $\tau^{|u|+1}$, then we conclude immediately. Now assume that the first edge of γ is an edge from $X^{|u|+1}$ to $X^{|u|}$ (whose weight is 0).

Since γ cannot contain $\tau^{|u|+1}$, except as a last vertex, it can be decomposed as $\gamma = \gamma_1 \cdot \gamma_2 \cdot \gamma_3$, where γ_1 and γ_3 are 0-weight paths and γ_2 is a path in $G(u)$ between $X^{|u|}$ and some $Y^{|u|}$. Now, $p((x'\tau')_u) \leq p((x'y')_u) + p((y'\tau')_u) \leq p((x'y')_u) = p((x'\tau')_{u\cdot t_\theta}$, hence the result. □

Property 45 *Let $u \sim v$, and $t \in \Sigma^*$ such that $\phi(u \cdot t)$ is satisfiable. $\forall x, y, z$, such that $(x'\tau')_{u\cdot t} \geq -K$, if there is $t_\theta \in \mathcal{T}(\theta, x, z)$ such that $p((x'y)_{u\cdot t}) = p((x'y)_{u\cdot t\cdot t_\theta})$ then $p((x'y)_{v\cdot t}) = p((x'y)_{v\cdot t\cdot t_\theta})$.*

Proof : Let γ be a representative of $(x'\tau')_{u\cdot t}$. Now, since $p(\gamma) = p((x'\tau')_{u\cdot t\cdot t_\theta})$, the path γ preceded by the 0-weight edge δ_0 from $X^{|u\cdot t|+1}$ to $X^{|u\cdot t|}$ is a representative of $(x'\tau')_{u\cdot t\cdot t_\theta}$.

Using property 44 (twice) and property 43 (four times) we can decompose representatives of respectively $((x'y)_{u\cdot t\cdot t_\theta}), ((x'y)_{u\cdot t}), ((x'y)_{v\cdot t\cdot t_\theta}), ((x'y)_{v\cdot t})$ into $\delta_0 \cdot \delta_1 \cdot \gamma_1, \delta_1 \cdot \gamma_1, \delta_2' \cdot \gamma_1', \delta_1' \cdot \gamma_1'$ where γ_1, γ_1' are respectively representatives for $(x_1'y)_u, (x_1'y)_v$ and $p(\delta_1) = p(\delta_1')$, $p(\delta_0 \cdot \delta_1) = p(\delta_2')$. In particular, $p(\delta_1') = p(\delta_2')$, which shows $p((x'y)_{v\cdot t}) = p((x'y)_{v\cdot t\cdot t_\theta})$. □

Property 46 *If $u \sim v$ and $t \in \Sigma^*$, then $u \cdot t \sim_6 v \cdot t$.*

Proof : By property 45, the first condition in the definition of $\sim_6$ is satisfied for $u \cdot t$, $v \cdot t$.

Now, assume that $p((x'\tau')_{u\cdot t}) \geq -K$, $p((z'\tau')_{u\cdot t}) \geq -K$ and $t_\theta \in \mathcal{T}(\theta, X, Z), t_\gamma \in \mathcal{T}(\gamma, Z, X)$ are such that $p((x'y)_{u\cdot t\cdot t_\theta}) = p(x'y)_{u\cdot t})$ and $p((z's)_{u\cdot t\cdot t_\gamma}) = p((z's)_{u\cdot t})$. By property 45, we also have $p((x'y)_{v\cdot t\cdot t_\theta}) = p(x'y)_{v\cdot t})$ and $p((z's)_{v\cdot t\cdot t_\gamma}) = p((z's)_{v\cdot t})$.

We decompose a representative of $(x'y)_{u\cdot t}$ and a representative of $(z's)_{u\cdot t}$ into respectively $\delta_1 \cdot (x_1'y)_u$ and $\delta_2 \cdot (z_1'y)_u$ in such a way that the last edges of δ_1, δ_2 do not belong to $G(u)$.

There is a path from $X_1^{|u|}$ to $Z_1^{|u|}$ in $G(t \cdot t_\theta)$: either δ_2 is a path within t and we may travel through $\tau^{|u|}, \tau^{|u|+|t|+1}, Z^{|u|+|t|}$ and then follow δ_2, or else there is a τ^i, $i \geq |u|$, in δ_2. In the latter case, let us decompose δ_2 into a path from $X_1^{|u|}$ to $\tau^{|u|}$ (which always exists), a path between $\tau^{|u|}$ and τ^i (which always exists) and the path between τ^i and $Z_1^{|u|}$. (See figure 20).

Similarly, there is a path in $G(t \cdot t_\gamma)$ between $Z_1^{|u|}$ and $X_1^{|u|}$.

By property 22, $p((x_1z_1)_{t\cdot t_\theta}), p((z_1x_1)_{t\cdot t_\gamma}) \in [-K, K]$. Therefore, considering transitions $t_1 \in \mathcal{T}(p((x_1z_1)_{t\cdot t_\theta}), X_1, Z_1)$ and $t_2 \in \mathcal{T}(p((z_1x_1)_{t\cdot t_\gamma}), Z_1, X_1)$, $p((x_1y)_{u\cdot t_1}) = p((x_1y)_u)$ and $p((z_1s)_{u\cdot t_2}) = p((z_1s)_u)$. Since $u \sim_6 v$, we get

$$(P2) \qquad\qquad p((x_1'y)_u) - p((z_1's)_u) = p((x_1'y)_v) - p((z_1's)_v)$$

On the other hand, by property 43 (twice), there are representatives of $((x'y)_{v\cdot t})$ and $((z's)_{v\cdot t})$ respectively which can be decomposed into $\delta_1' \cdot \gamma_1'$ and $\delta_2' \cdot \gamma_2'$ in such a way that $p(\delta_1) = p(\delta_1')$, $p(\delta_2) = p(\delta_2')$, γ_1' is a representative of $(x_1'y)_v$ and γ_2' is a representative of $(z_1's)_v$. By $(P2)$ we then get:

$$\begin{aligned} p((x'y)_{u\cdot t}) - p((z's)_{u\cdot t} &= p(\delta_1) + p((x_1'y)_u) - p(\delta_2) - p((z_1's)_u) \\ &= p(\delta_1') + p((x_1'y)_v) - p(\delta_2') - p((z_1's)_v) \\ &= p((x'y)_{v\cdot t}) - p((z's)_{v\cdot t}) \end{aligned}$$

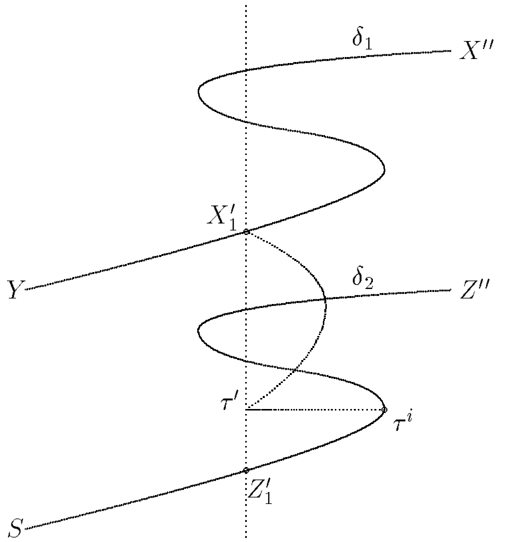

Figure 20: Paths from $X_1^{|u|}$ to $Z_1^{|u|}$

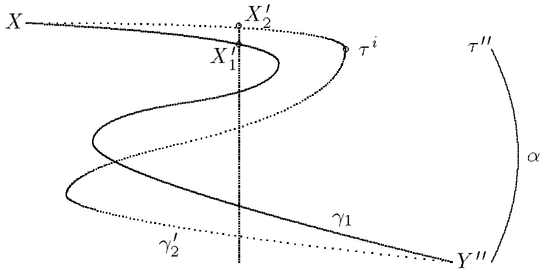

Figure 21: Proof of the property 47

□

Property 47 *Let $u \sim v$, and $t \in \Sigma$ such that $\phi(u \cdot t)$ is satisfiable, then $u \cdot t \sim_7 v \cdot t$*

<u>Proof</u> : Let x, y be two clock names, and $t_\alpha \in \mathcal{T}(\alpha, Y, \tau)$ be such that $p((x\tau')_{u \cdot t \cdot t_\alpha}) = p((xy')_{u \cdot t}) + \alpha$. Let δ_1, γ_1 be paths such that $\delta_1 \cdot \gamma_1$ is a representative of $(x\tau')_{u \cdot t \cdot t_\alpha}$, δ_1 is a representative of $(x, x'_1)_u$ and the first edge of γ_1 is not in $G(u)$. Let $\delta_2 \cdot \gamma_2$ be a similar decomposition of a representative of $(x\tau')_{v \cdot t \cdot t_\alpha}$: γ_2 is a representative of $(xx'_2)_v$. We first show that there are paths γ'_1, γ'_2 in $G(v \cdot t \cdot t_\alpha)$ and $G(u \cdot t \cdot t_\alpha)$ respectively, such that $p(\gamma'_1) = p(\gamma_1)$, $p(\gamma'_2) = p(\gamma_2)$ and γ'_1 is a path from $X_1^{|v|}$ to $\tau^{|v|+|t|}$, γ'_2 is a path from $X_2^{|u|}$ to $\tau^{|u|+|t|}$ (see figure 21). By symmetry, let us only show how γ'_2 can be constructed.

Let $\zeta_0 \cdot \ldots \cdot \zeta_n$ be a decomposition of γ_2 according to $v \cdot (t \cdot t_\alpha)$. We proceed as in the proof of property 43. Either $n = 0$ and γ_2 is a path of $G(t \cdot t_\alpha)$, in which case we choose $\gamma'_2 = \gamma_2$. Or else ζ_0 contains some τ^i. In any case, ζ_n contains some τ^i. If ζ_j is a path in $G(t \cdot t_\alpha)$, we let $\zeta'_j = \zeta_j$. Otherwise, ζ_j is a path in $G(v)$. By property 20, $p(\zeta_j) \leq K$

and by property 19, $p(\zeta_j) \geq -K$. Now, since $u \sim_2 v$ and $p(\zeta_j) \in [-K, K]$, there is a path ζ'_j of $G(u)$ which has the same target and the same origin as ζ_j and whose weight is $p(\zeta'_j) = p(\zeta_j)$. Finally, we define $\gamma'_2 = \zeta'_0 \cdot \dots \cdot \zeta'_n$.

If γ'_2 does not contain any τ^i, then $x_2 = y = x_1$ and $p((xx'_2)_v \cdot \gamma_2) = p((xx'_1)_v \cdot \gamma'_1)$ (and symmetrically $p((xx'_2)_u \cdot \gamma'_2) = p((xx'_1)_u \cdot \gamma_1))$.

Otherwise, let τ^i the first τ vertex of γ'_2. The weight α_1 of the path from $X_2^{|u|}$ to τ^i belongs to $[-K, 0]$. Let $t_{\alpha_1} \in \mathcal{T}(\alpha_1, X_2, \tau)$. By construction 9minimal path property), $p((x\tau')_{v \cdot t_{\alpha_1}}) = p((x\tau')_v) + \alpha_1$. Now, since $u \sim_7 v$, $p((xx'_2)_u) - p((\tau\tau')_u) = p((xx'_2)_v) - p((\tau\tau')_v)$. Symmetrically, we have also $p((xx'_1)_u) - p((\tau\tau')_u) = p((xx'_1)_v) - p((\tau\tau')_v)$. By difference, we get:

$$(P3) \qquad p((xx'_1)_u) - p((xx'_2)_u) = p((xx'_1)_v) - p((xx'_2)_v)$$

Now, by definition of the paths (minimal weight properties) we have

$$\begin{cases} p((xx'_1)_u) + p(\gamma_1) & \leq & p((xx'_2)_u) + p(\gamma'_2) \\ p((xx'_2)_v) + p(\gamma_2) & \leq & p((xx'_1)_v) + p(\gamma'_1) \end{cases}$$

Together with $(P3)$ we get:

$$\begin{array}{rcl} p((xx'_2)_v + p(\gamma_2) & \leq & p((xx'_1)_v) + p(\gamma'_1) \\ & = & p(\gamma'_1) + p((xx'_2)_v) + p((xx'_1)_u) - p((xx'_2)_u) \\ & \leq & p(\gamma'_1) - p(\gamma_1) + p(\gamma'_2) + p((xx'_2)_v) \\ & = & p((xx'_2)_v) + p(\gamma_2) \end{array}$$

which means that all inequalities above are equalities. In particular, if $p((x\tau')_{u \cdot t \cdot t_\alpha}) = p((xy')_{u \cdot t}) + \alpha$, then

$$p((x\tau')_{v \cdot t \cdot t_\alpha}) = p((xx'_2)_v) + p(\gamma_2) = p((xx'_1)_v) + p(\gamma'_1)$$

and since the last edge of γ'_1 is the same as the last edge of γ_1 (by construction), we get $p((x\tau')_{v \cdot t \cdot t_\alpha}) = p((x\tau')_{v \cdot t}) + \alpha$.

It remains to show that $p((xy')_{u \cdot t}) - p((\tau\tau')_{u \cdot t}) = p((xy')_{v \cdot t}) - p((\tau\tau')_{v \cdot t})$.

Let $(xx'_1)_u \cdot \delta_1$ be a decomposition of a re presentative of $(xy')_{u \cdot t}$, and $(\tau x'_2)_u \cdot \delta_2$ be one of a representative of $(\tau\tau')_{u \cdot t}$. We have

$$p((xy')_{u \cdot t}) - p((\tau\tau')_{u \cdot t}) = p((xx'_1)_u) - p((\tau\tau')_u) - (p((\tau x'_2)_u) - p((\tau\tau')_u)) + p(\delta_1) - p(\delta_2)$$

. And, since $u \sim_7 v$,

$$p((xy')_{u \cdot t}) - p((\tau\tau')_{u \cdot t}) = p((xx'_1)_v) - p((\tau\tau')_v) - (p((\tau x'_2)_v) - p((\tau\tau')_v)) + p(\delta_1) - p(\delta_2)$$

Since $u \sim_7 v$ and $u \sim_2 v$, as we saw above, there are paths δ'_1, δ'_2 in $G(v \cdot t)$ with respectively the same source, the same target and the same weight as $\delta_1 \delta_2$. It follows that

$$\begin{array}{rcl} p((xy')_{u \cdot t}) - p((\tau\tau')_{u \cdot t}) & = & p((xx'_1)_v) - p((\tau\tau')_v) - (p((\tau x'_2)_v) - p((\tau\tau')_v)) + p(\delta'_1) - p(\delta'_2) \\ & \geq & p((xy')_{v \cdot t}) - p((\tau\tau')_{v \cdot t}) \end{array}$$

And we get the equality by symmetry. □

Proof of lemma 34 .

The right compatibility is a consequence of properties 40, 41, 46, 47.

The finite index properties follows from bounds on the number of classes for each component of the relation (these bounds are rough):

$\sim_0$: the number of classes is bounded by the number of Boolean functions whose domain is the set of pairs of clocks; there are at most 2^{n^2} classes

$\sim_1$: each class is characterised by a mapping which associates each pair of clocks x, y with $p((xy)_u)$. By property 22, $p((xy)_u)$ takes its values in the interval $[-K, K]$, hence the number of classes for $\sim_1$ is bounded by $(2K+1)^{n^2} + 1$ (for unsatisfiable formulas).

$\sim_2$: Similarly, there are at most $(2K+3)^{n^2} + 1$ classes.

$\sim_3$: For each 4-tuple of clocks and each $\theta \in [-K, K]$, the sum $p((x'_1 z'_1)_u) + p((z'_2 y')_u)$ may take only $4K+3$ distinct significant values. Hence the number of classes is bounded by $(4K+3)^{n^4}$.

$\sim_4$: The inequalities imply that $p((x_1 y'_1)_u) + p((y'_2 x_2)_u)$ can only take $4K+3$ significant values, thanks to property 22. Then there are at most $(4K+3)^{n^4}$ distinct classes.

$\sim_5$: The minimal path weight $\alpha + p((x'_1 y_1)_u) + \theta + p((y_2 x'_2)_u) + \beta$ is the weight of a path between τ^i and τ^{i+1} (or between τ^{i+1} and τ^i). Hence, thanks to properties 19 and 20, it belongs to $[-K, K]$. Then an analysis similar to the above one shows that there are at most $(6K+3)^{n^4}$ classes.

$\sim_6$: By property 42, $p((x'y)_u) - p((z's)_u)$ can take at most $4K+1$ distinct values. Hence there are at most $(4K+1)^{n^4} + 1$ distinct classes

$\sim_7$: If $p((x\tau')_{u \cdot t_\alpha}) = \alpha + p((xy')_u)$, $\alpha + p((xy')_u) \leq p((x\tau)_u) + p((\tau\tau')_u) \leq p((\tau\tau')_u)$. Since α can only take $K+1$ distinct values, there are at most $(K+1)^{n^2} + 1$ distinct classes.

3.2 A commutation property

Lemma 48 *Let $u, v, w \in \Sigma^*$ such that $u \sim v \sim w$. Then $\phi(u \cdot v \cdot w) \models\!\!\dashv \phi(v \cdot u \cdot w)$*

This property shows that, if we have two sequences of transitions w and w' from q to itself and such that $w \sim w'$, then the iteration of both loops i.e. the set of transitions $(w + w')^*$ has the same effect on clocks values as $w^* w'^* (w + \epsilon)$. This shows a flattening operation on regular expressions: $(w + w')^*$ is not flat whereas $w^* w'^* (w + \epsilon)$ is flat.

However, we cannot conclude yet since it is not always possible to compute an automaton emulating $\mathcal{A}$ and such that any two loops on the same state are equivalent for $\sim$. We need a more complex construction which proves that the automaton can be flattened when we have such commutation properties. This is the subject of the next section.

To prove the lemma, we use again the graphs associated with the transition sequences (see the properties 13,12). The easy part is the case where $\phi(u \cdot v \cdot w)$ is unsatisfiable. Then by property 23, $\phi(v \cdot u \cdot w)$ is also unsatisfiable. In the other case, we consider paths in ?$(u \cdot v \cdot w)$, and ?$(v \cdot u \cdot w)$ (see property 13).

We consider in the following 4 properties the four kinds of paths:

1. between two initial clock values (property 49)
2. between two final clock values (property 52)
3. from a final to an initial clock value (property 50)
4. from an initial to a final clock value (property 51

In each case, we show that the weight of the minimal path is the same in $u \cdot v \cdot w$ and in $v \cdot u \cdot w$, which proves lemma 48.

Property 49 *Let $u \sim v \sim w$ be words such that $\phi(u \cdot v \cdot w)$ is satisfiable. For all x, y we have $p((xy)_{u \cdot v \cdot w}) = p((xy)_{v \cdot u \cdot w})$*

Proof :
If $p((xy)_u) = p((xy)_{u \cdot v \cdot w})$ then since $u \sim_1 v$ we have $p((xy)_{v \cdot u \cdot w}) \leq p((xy)_{u \cdot v \cdot w})$.

Otherwise, let us consider a decomposition $(xx'_1)_u \cdot \Pi_i(\delta_i \cdot \gamma_i) \cdot \delta_n \cdot (y'_1 y)_u$ such that every δ_i is in $G(v \cdot w)$ and every $\gamma_i$is in $G(u)$. For every i, $p(\gamma_i) \in [-K, K]$ by properties 20 and 19 (and since there must be τ-vertices in every δ_i. Since $u \sim_2 v$, for every i, there is a path γ'_i in v which has the same source, the same target and the same weight as γ_i. Similarly, since $u \sim v$, $u \cdot w \sim v \cdot w$, and, in particular, $u \cdot w \sim_1 v \cdot w$. Therefore, for every i, there is a path δ'_i in $G(u \cdot w)$ which has the same source, the same target and the same weight as δ_i. Finally, we may use property 21: there is a $t_\eta \in \mathcal{T}(p(\Pi_i(\delta_i \cdot \gamma_i) \cdot \delta_n), X_1, X_2)$ such that the weight of minimal path between $X_1^{|u|}$ and $Y_1^{|u|}$ in $G(u \cdot t_\eta)$ is $\eta = p(\Pi_i(\delta_i \cdot \gamma_i) \cdot \delta_n)$. Now, since $u \sim_4 v$, $p((xx'_1)_u) + p((y'_1 y)_u) = p((xx'_1)_v) + p((y'_1 y)_v)$.

Now, we can construct a path from X^0 to Y^0 in $G(v \cdot u \cdot w)$ as follows: $(xx'_1)_v \cdot \Pi_i(\delta'_i \cdot \gamma'_i) \cdot \delta'_n \cdot (y'_1 y)_v$. According to the properties above, the weight of this path is equal to $p((xy)_{u \cdot v \cdot w})$. This proves that $p((xy)_{u \cdot v \cdot w}) \geq p((xy)_{v \cdot u \cdot w})$, hence the equality, by symmetry.

□

Property 50 *Let $u \sim v \sim w$ such that $\phi(u \cdot v \cdot w)$ is satisfiable. For all x, y we have $p((x'y)_{u \cdot v \cdot w}) = p((x'y)_{v \cdot u \cdot w})$*

Proof : In case x is not reset in w, then $p((x'y)_{u \cdot v \cdot w}) = p((xy)_{u \cdot v \cdot w}) = p((xy)_{v \cdot u \cdot w})$ by the property 49, and since x is not reset in $v \cdot u \cdot w$, it is also equal to $p((x'y)_{v \cdot u \cdot w})$.

So we consider the case where x is reset in w. We decompose a representative γ of $(x'y)_{u \cdot v \cdot w}$ into $\gamma_0 \cdot \delta_1 \cdot \epsilon_1 \cdot \eta_1 \cdot \theta_1 \cdots \delta_n \cdot \gamma_1 \cdot \eta_k \cdot \gamma_2$ (see figure 22) in such a way that paths δ_i belong to $G(v \cdot w)$ and relate some $Z^{|u|+|v|}$ to some $U^{|u|+|v|}$, paths $\epsilon_i, theta_i$ belong to $G(v)$, paths η_i belong to $G(u \cdot v)$ and relate two vertices $Z^{|u|}, U^{|u|}$. Moreover γ_0 is a pure $G(w)$ path, γ_1 is a pure $G(v)$ path and γ_2 is a pure $G(u)$ path.

Now, for each piece of this path, we construct another path in $G(v \cdot u \cdot w)$ with the same sources, targets and weights as the original path in $G(u \cdot v \cdot w)$:

- The path γ_0 is unchanged

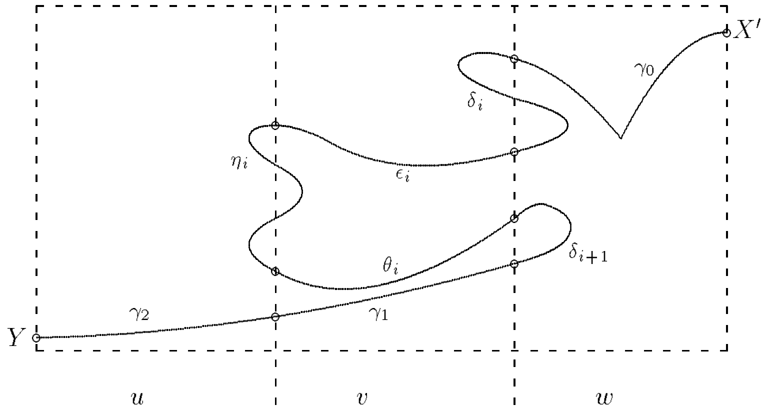

Figure 22: The proof of property 50

- The paths δ_i can be decomposed according to (v, w). Since there is a τ vertex in the path before δ_i, each piece of δ_i which belongs to $G(v)$ has a weight in $[-K, K]$ thanks to properties 20, 19, 21. Then, since $u \sim_2 v$, there is, for each piece of the decomposition belonging to $G(v)$ a path in $G(u)$ with the same source, same target and same weight. It follows that there is a path δ'_i in $G(u \cdot w)$ with the same source, the same target and the same weight as δ_i.

- A similar reasoning, using $u \sim_1 v$ and $u \sim_2 v$ shows that there are paths η'_i in $G(v \cdot u)$ which have the same source, the same target and the same weight as η_i.

- Concerning the paths ϵ_i and θ_i, we know that there is a τ vertex in the path before ϵ_i (at least in γ_0) and that there is a τ vertex in the path δ_{i+1} which follows θ_i. Moreover, by property 21, the intermediate path η_i has a weight in $[-K, K]$. Considering now $t_\theta \in \mathcal{P}(X_i, Y_i, p(\eta_i))$ (for appropriate X_i, Y_i), t_1 as being the transitions which consist of the last edge in γ whose source is a τ vertex and which is before ϵ_i in γ (let α be its weight), t_2 as the transition which consists of the first edge in γ whose destination is a τ vertex and which follows θ_i in γ (let β be its weight), $\mathcal{Q}$ is satisfied for $x_i, y_i, z_i, u_i, \theta, \alpha, \beta$ and, since $u \sim_5 v$, there are paths ϵ'_i, θ'_i in $G(u)$ with respectively the same sources and the same targets as ϵ_i, θ_i and such that $p(\epsilon_i) + p(\theta_i) = p(\epsilon'_i) + p(\theta'_i)$.

- Consider now the paths γ_1, γ_2. Let $X_1^{|u|+|v|}$ be the source of γ_1 and $Z_1^{|u|}$ its destination. Let $Z_2^{|u|}$ be the source of γ_2. We want to show that $p((x'_1 z_1)_u) - p((z'_2 y)_u) = p((x'_1 z_1)_v) - p((z'_2 y)_v)$.

  δ_i, when decomposed according to (v, w), has a last component in $G(w)$ which contains a τ vertex. Since moreover x is reset in w, in all cases, there is a path from τ to X_1 in $G(w)$. Since $v \sim_1 w$, there is also a path in $G(v)$ from τ to X_1. Now, since $G(u \cdot v \cdot w)$ does not contain any cycle of negative weight and thanks to property 22, $p((x'_1 \tau')_u) \geq -K$. A similar argument shows that there is a path from τ to Z_2 in $G(v)$, hence that $p((z'_2 \tau')_u) \geq -K$.

  As a consequence, we now also that there are paths in $G(v)$ respectively from X_1 to

Z_2 and from Z_2 to X_1. The weights of these paths belong to $[-K, K]$ thanks again to property 22. Let $t_\theta \in \mathcal{T}(p((x_1z_2)_v), X_1, Z_2)$ and $t_\gamma \in \mathcal{T}(p((z_2x_1)_v), Z_2, X_1)$. Since the path γ_2 is a path of $G(u)$, $p((z'_2y)_{u\cdot t_\gamma}) = p((z'_2y)_u)$. Similarly, $p((x'_1z_1)_{v\cdot t_\theta}) = p((x'_1z_1)_v)$. Then, since $u \sim_6 v$ (and thanks to the above proved property that $p((x'_1\tau')_v) \geq -K)$, $p((x'_1z_1)_{u\cdot t_\theta}) = p((x'_1z_1)_u)$.

Now, the hypotheses of definition 31 are satisfied and we can conclude from $u \sim_6 v$ that

$$p((x'_1z_1)_u) - p((z'_2y)_u) = p((x'_1z_1)_v) - p((z'_2y)_v)$$

Let γ'_1 be a representative of $(x'_1z_1)_u$ and γ'_2 be a representative of $(z'_2y)_v$. From the above equality, we derive

$$p(\gamma_1) + p(\gamma_2) = p(\gamma'_1) + p(\gamma'_2)$$

Finally, we constructed a path in $G(v\cdot u\cdot w)$: $\gamma_0 \cdot \delta'_1 \cdot \epsilon'_1 \cdot \eta'_1 \cdot \theta'_1 \cdots \delta'_n \cdot \gamma'_1 \cdot \gamma'_2$ whose source, target and weight are identical to those of γ. It follows that $p((x'y)_{v\cdot u\cdot w}) \leq p((x'y)_{u\cdot v\cdot w})$, hence the equality by symmetry. □

Property 51 *Let $u \sim v \sim w$ such that $\phi(u \cdot v \cdot w)$ is satisfiable. For all x, y we have $p((xy')_{u\cdot v\cdot w}) = p((xy')_{v\cdot u\cdot w})$*

Proof : Let γ be a representative of $(xy')_{u\cdot v\cdot w}$. If y is not reset in w, since $u \sim_0 v \sim_0 w$, $x = y$ and it is not reset in any of u, v, w. Then $p((xy')_{u\cdot v\cdot w}) = 0 = p((xy')_{v\cdot u\cdot w})$. Now, we assume that y is reset in w. Let $\gamma = \gamma_0 \cdot \gamma_1$ where γ_0 is a maximal path contained in $G(w)$. Let Y_1 be the target of γ_0. If y_1 is not reset in v, then, again, it should not be reset in u and w, which would imply $y_1 = y$ and contradict the above hypothesis. hence we may also assume that y_1 is reset in v.

As in the proof of property 50, we decompose γ. The proof is then similar: only the parts corresponding to γ_1 and γ_2 in the previous proof have to be modified. More precisely, if γ is decomposed as depicted on figure 23, we prove, as above, that there is a path δ' in $G(v \cdot u \cdot w)$ which has the same source, same target and same weight as the path between $Y_2^{|u|+|v|}$ and $Y_1^{|u|+|v|}$ in $G(u \cdot v \cdot w)$. There is also a path in $G(v \cdot u)$ whose source is $X_1^{|v|}$, whose target is $X_2^{|v|}$ and which has the same weight as the path between $X_1^{|u|}$ and $X_2^{|u|}$ in $G(u \cdot v \cdot w)$.

It only remains to show that $p((xx'_1)_u) + p((x_2y'_2)_v) \geq p((xx'_1)_v) + p((x_2y'_2)_u)$: then we will have constructed a path in $G(v \cdot u \cdot w)$ from X to Y' and whose weight is smaller or equal to $p(\gamma)$ and we can conclude by symmetry.

As before, if we consider the first edge in γ whose target is τ and which is in v (such an edge does exist as we have seen), we let t_α be the transition consisting of only that edge (of weight α) besides the mandatory edges. Then $t_\alpha \in \mathcal{T}(\alpha, X_1, \tau)$ and $p((x\tau')_{u\cdot t_\alpha}) = p((x\tau')_u) + \alpha$. Now, since $u \sim_7 v$, $p((x\tau')_{v\cdot t_\alpha}) = p((x\tau')_v) + \alpha$ and $p((xx'_1)_u) - p((\tau\tau')_u) = p((xx'_1)_v) - p((\tau\tau')_v)$. Similarly, we can construct a $t_\beta \in \mathcal{T}(\beta, Y_2, \tau)$ such that $p((x_2\tau')_{v\cdot t_\beta}) = p((x_2\tau')_v) + \beta$ and (again by $\sim_7$), $p((x_2\tau')_{u\cdot t_\beta}) = p((x_2\tau')_u) + \beta$, $p((x_2y'_2)_u) - p((\tau\tau')_u) = p((x_2y'_2)_v) - p((\tau\tau')_v)$. Summing and simplifying the equalities we have the desired identity:

$$p((xx'_1)_u) + p((x_2y'_2)_v) = p((xx'_1)_v) + p((x_2y'_2)_u)$$

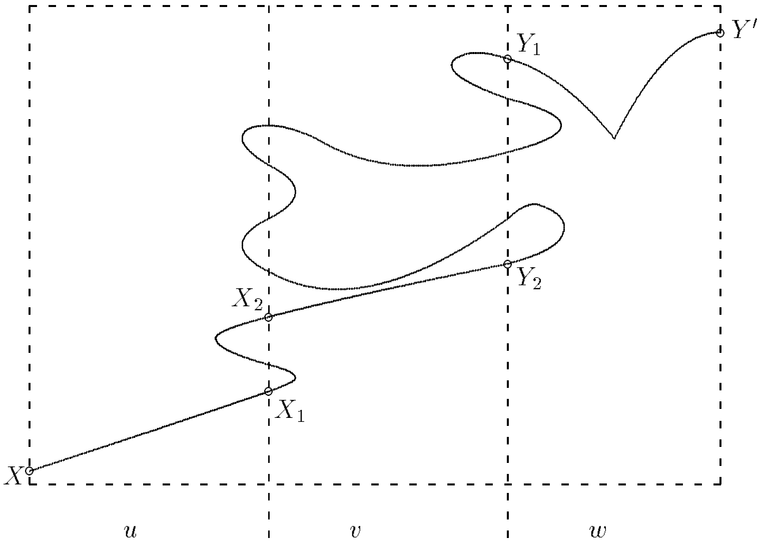

Figure 23: The proof of property 51

□

Property 52 *Let $u \sim v \sim w$ such that $\phi(u \cdot v \cdot w)$ is satisfiable. For all x, y we have $p((x'y')_{u\cdot v\cdot w}) = p((x'y')_{v\cdot u\cdot w})$*

<u>Proof</u> : If x or y is not reset in w, then since $u \sim v \sim w$ we have either $p((x'y')_{u\cdot v\cdot w}) = p((xy')_{u\cdot v\cdot w})$ or $p((x'y')_{u\cdot v\cdot w}) = p((x'y)_{u\cdot v\cdot w})$ and the lemma is a consequence of properties 51 and 50.

We assume now that x and y are reset in w. If $p((x'y')_w) = p((x'y')_{u\cdot v\cdot w})$, then we have $p((x'y')_{v\cdot u\cdot w}) \leq p((x'y')_{u\cdot v\cdot w})$.

Otherwise, let us consider a decomposition $\delta_0 \cdot \gamma_1 \cdot \delta_1 \cdots \delta_{n-1} \cdot \gamma_n \cdot \delta_n$ of a representative of $(x'y')_{u\cdot v\cdot w}$, where δ_0 is a representative of $(x'y_1)_w$, δ_n is a representative of $(x_ny')_w$ and δ_i, $1 \leq i < n$ is a representative of $(x_iy_{i+1})_w$ and γ_i, $1 \leq i \leq n$ is a representative of $(y_ix_i)_{u\cdot v}$. The key of the proof is that, for every k, every path δ_k contains a τ vertex. We fix now any k. Let τ^{i_k} (resp. τ^{j_k}) be the last (resp. the first) τ vertex in δ_k. $i_k \neq j_{k+1}$, since there is no cyclic path in $(x'y')_{u\cdot v\cdot w}$. Let us assume that $i_k < j_{k+1}$ (the case where $j_{k+1} < i_k$ is similar). This is depicted on figure 24.

Then let t_1 (resp. t_2) be the transition which contains, besides the constraints which are present in any transition, a single constraint corresponding to the edge of δ_k (resp. δ_{k+1}) whose source (resp. target) is τ^{i_k} (resp. $\tau^{j_{k+1}}$).

Moreover, we decompose γ_{k+1} according to (u, v): $\gamma_{k+1} = \gamma'_{k+1} \cdot \eta_{k+1} \cdot \gamma''_{k+1}$ where γ'_{k+1} and γ''_{k+1} are maximal paths in $G(v)$. Thanks to property 24, $p(\eta_{k+1}) \in [-K, K]$. Let then $\theta = p(\eta_{k+1})$ and $t_\theta \in \mathcal{P}(\theta, Z_1, Z_2)$ (Z_1, Z_2 are respectively the target and the source of γ'_{k+1} and γ''_{k+1}).

The property $\mathcal{Q}$ is satisfied with appropriate clock names, t_1, t_2 and t_θ. Since $u \sim_5 v$, there are paths π'_{k+1} and π''_{k+1} in $G(u)$ such that π'_{k+1} (resp. π''_{k+1}) has the same source

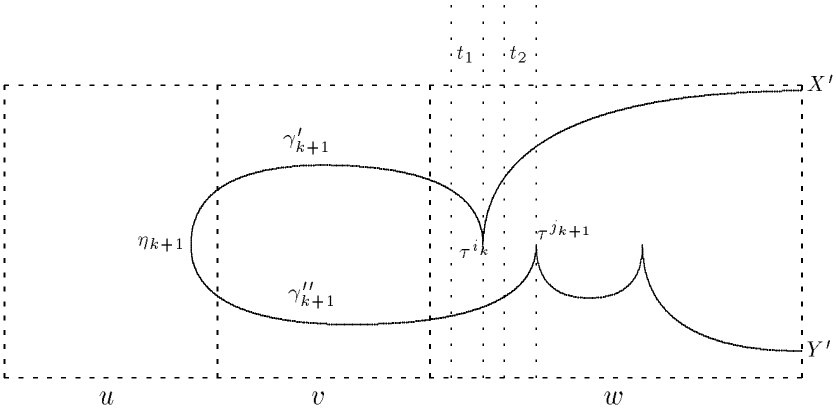

Figure 24: The proof of property 52

and the same target as γ'_{k+1} (resp. γ''_{k+1}) and $p(\gamma'_{k+1}) + p(\gamma''_{k+1}) = p(\pi'_{k+1}) + p(\pi''_{k+1})$. Now, since $u \sim_2 v$ and $u \sim_1 v$ and thanks to properties 24 and 21, there is a path η'_{k+1} in $G(v \cdot u)$ whose source is $Z_1^{|v|}$ and whose target is $Z_2^{|v|}$ such that $p(\eta'_{k+1}) = p(\eta_{k+1})$.

Putting everything together, there is a path ρ_{k+1} in $G(v \cdot u)$ with the same source, same target and same weight as γ_{k+1}.

Since this result holds for every k, we find a path in $G(v \cdot u \cdot w)$: $\delta_0 \cdot \rho_1 \cdots \rho_n \cdot \delta_n$ whose source, target and weight are identical to the source, target and weight of a representative of $((x'y')_{u\cdot v\cdot w})$. This proves that

$$p((x'y')_{v\cdot u\cdot w}) \leq p((x'y')_{u\cdot v\cdot w})$$

By symmetry, this inequality is an equality, hence the property. □

3.3 A formal language property

In this section, we prove a formal language property which relates commutation properties and flat automata:

Theorem 53 *Let $\sim$ be a finite index right-compatible equivalence relation on A^* and let $\approx$ be the least congruence on A^* such that $w \sim w' \sim w''$ implies $w \cdot w' \cdot w'' \approx w' \cdot w \cdot w''$. Then there is a flat automaton (whose all states are final states) which accepts a language containing a set of representatives for $\approx$.*

The proof of this result is a consequence of a series of lemmas which we state and prove below. In what follows, if $w \in A^*$, we write $w(i)$ for the ith symbol of w and $w[i, j]$ for the factor of w starting at position i (included) and ending at position j (included).

Lemma 54 *Let k be the index of $\sim$. There is a constant $n(k)$ such that, for any increasing sequence of positions $p_1 < \ldots < p_{n(k)}$ of w, w can be factorised into $w = w_0 \cdot w_1 \cdot w_2 \cdot w_3 \cdot w_4$ with $w_1 \sim w_2 \sim w_3$ and $w_1 = w[p_{i_1}, p_{i_2} - 1], w_2 = w[p_{i_2}, p_{i_3} - 1], w_3 = w[p_{i_3}, p_{i_4} - 1]$ for some $p_{i_1} < p_{i_2} < p_{i_3} < p_{i_4}$.*

This states that any long enough word contains a factor to which the commutation property can be applied.

Proof : We let $n(k)$ be $2^{2k}+1$, We associate with each position p_i in w a pair of sets as follows:

$$E(i) = \{c \in A^*/\sim \ \mid \exists j < i, w[p_j, p_i - 1] \in c\}$$
$$F(i) = \{c \in A^*/\sim \ \mid \exists j \geq i, w[p_i, p_j] \in c\}$$

The pairs $(E(i), F(i))$ take at most 2^{2k} distinct values, hence there are at least two distinct indices i_1, i_2 such that $E(i_1) = E(i_2)$ and $F(i_1) = F(i_2)$. Let $w_2 = w[p_{i_1}, p_{i_2} - 1]$ and c be such that $w_2 \in c$. $c \in F_{i_1} \cap E_{i_2}$. Hence $c \in E_{i_1} \cap F_{i_2}$, which means that there is a factor $w_1 = w[p_{j_1}, p_{i_1} - 1] \in c$ and a factor $w_3 = w[p_{i_2}, p_{j_2}] \in c$. The result follows. □

Now, consider the word rewrite system defined by

$$R \overset{\text{def}}{=} \{uxvyz \ \rightarrow \ vyuxz \mid u >_{lex} v; ux \sim vy \sim z; u, v \in A^{k_1}\}$$

This rewrite system compares lexicographically prefixes of length k_1 of equivalent words and commutes them according to the ordering.

Lemma 55 *R is a terminating rewrite system. The set of irreducible words $NF(R)$ w.r.t R is recognisable. Moreover, if $u \xrightarrow[R]{*} v$ then $u \approx v$.*

Proof : The termination follows from the well-foundedness of the lexicographic comparison on words of the same length.

For each class $c \in A^*/\sim$, the set of words ux belonging to c is recognisable (each c is recognisable). By closure properties of recognisable languages (under concatenation, intersection and union) the set of words matching an expression $uxvyz$ with $ux \sim vy \sim z$ is recognisable, hence the set of reducible words is also recognisable, since there are only finitely many possible u, v. Finally, by closure of recognisable languages under complement, the set of irreducible words is recognisable.

Now, by definition of $\approx$, each single rewrite step reduces a word to an equivalent one w.r.t. $\approx$. □

Now, we claim that $NF(R)$ is accepted by a flat automaton for some well-chosen k_1. We first recall some results in word combinatorics.

Lemma 56 ([18],[17] chapter 1) *If u is an infinite word which contains at most n distinct factors of length n, then u is ultimately periodic with a period of length at most n.*

Lemma 57 *If $w \in A^*$ and w^ω contains at most n distinct factors of length n, then there is a word u of length at most n and an integer m such that $w = u^m$.*

Proof : We apply lemma 56 to w^ω: w is ultimately periodic with a period $p \leq n$. Then it is ultimately periodic with a period $q = gcd(p, |w|)$. It follows that there is a $v \in A^*$ such that $w = w_0 v^m w_1$ with $w_1 w_0 = v$ and $|v| \leq n$. Now, it suffices to choose $u = w_0 w_1$. □

Lemma 58 *If $w^* \subseteq NF(R)$ for some $w \in A^*$, then there is a word u whose length is smaller than $n(k)-1$ and an integer m satisfying $w = u^m$.*

Proof : Let $w_{i_1} \geq_{lex} w_{i_2} \dots \geq_{lex} w_{i_r}$ be the lexicographically ordered sequence of factors $w^N[i, i+k_1-1]$ for a large enough N. Thanks to lemma 54, in any sequence $s = z_0 w_{i_{p_1}} z_1 w_{i_{p_2}} z_2 \dots w_{i_{p_{n(k)}}} z_{n(k)} \in w^*$ such that $p_1 < \dots < p_{n(k)}$, there are indices l, m, n such that $w_{i_{p_l}} y_l \sim w_{i_{p_m}} y_m \sim w_{i_{p_n}} y_n$ (for appropriate factors y_l, y_m, y_n of s). Now, by construction, $w_{i_{p_l}} \geq_{lex} w_{i_{p_m}} \geq_{lex} w_{i_{p_n}}$. Since we assumed that all words in w^* are irreducible by R, we must have $w_{i_{p_l}} = w_{i_{p_m}}$, by definition of R. This implies that w^N (for any large enough N) contains at most $n(k)-1$ distinct factors of length $n(k)-1$.

Now, by lemma 57, w must be a power of u, for some u whose length is smaller than $n(k)-1$.

□

Using this lemma together with standard combinatorial arguments [16], we get:

Lemma 59 *If $(w_1 + \dots + w_n)^* \subseteq NF(R)$ for some words $w_1, \dots, w_n \in A^*$, then there is a word u whose length is smaller than $n(k)-1$ and integers $m_1, \dots, m_n$ satisfying $w_i = u^{m_i}$ for all i.*

Proof : For each w_i we may apply lemma 58 and there are primitive words $u_1, \dots, u_n$ of length at most $n(k)-1$ such that $w_i = u_i^{n_i}$. Now, consider any two indices i, j and an integer q. $(w_i w_j^q)^* \subseteq NF(R)$, hence we may apply again lemma 58: there is a v_q of length at most $n(k)-1$ and an integer α_q such that $w_i w_j^q = v_q^{\alpha_q}$. Now, since v_q has only finitely many possible values, there is an infinite subsequence $\sigma(q)$ for which v_q is a constant v:

$$w_i w_j^{\sigma(q)} = v^{\alpha_{\sigma(q)}} = v^{\alpha(0)} w_2^{\sigma(q)-\sigma(0)} = v^{\sigma(0)} (u_j^{n_j})^{\sigma(q)-\sigma(0)}$$

It follows that

$$v^{\alpha_{\sigma(q)} - \alpha_{\sigma(0)}} = u_j^{m_j \times (\sigma(q)-\sigma(0))}$$

Hence, since v and u_j are primitive, they are identical: $v = u_j$. Now,

$$w_i w_j^{\sigma(q)} = u_j^{\alpha_{\sigma(q)}} = u_i^{n_i} u_j^{n_j \times \sigma(q)}$$

Again, by primitiveness of u_i, u_j, this implies $u_i = u_j$. □

Lemma 60 *If $L^* \subseteq NF(R)$ for some regular subset L of A^*, then there are finitely many pairs of words $(u_i, v_i) \in A^*$ such that $L^* = \sum_{i=1}^{n} u_i v_i^*$.*

Proof : By lemma 59, there is a word u such that, for any $w \in L$, there is an integer m_w such that $w = u^{m_w}$. Hence $L^* \subseteq u^*$ and, moreover, the set of lengths of words in L^* is a recognisable subset of $\mathbb{N}$, hence a finite union of arithmetic progressions $S = \bigcup_{i=1}^{n} \{a_i + b_i z \mid z \in \mathbb{N}\}$ for some $a_i, b_i \in \mathbb{N}$. It follows that

$$L^* = \{u^k \mid k \in S\} = \sum_{i=1}^{n} u^{a_i} (u^{b_i})^*$$

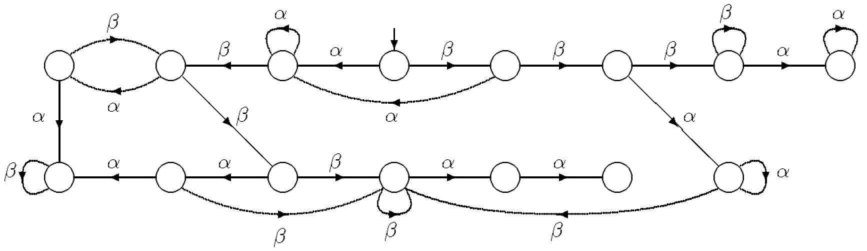

Figure 25: The resulting flat automaton

□

Which, altogether, allows to prove theorem 53.

Proof : (theorem 53) $NF(R)$ is a regular language (lemma 55). Let e be a regular expression for $NF(R)$. By lemma 60, we can replace any subexpression s^* of e with a sum of $u_i v_i^*$. Then we have another regular expression for $NF(R)$ which has the form $\sum_{i=1}^n u_{i,1} v_{i,1}^* \dots v_{i,n_i}^* u_{i,n_i}$, from which a flat automaton can be derived directly. Moreover, by lemma 55, $NF(R)$ contains a set of representatives for $\approx$, which completes the proof of the theorem. □

Example 61 Let us continue our running example. If we apply the algorithm described this section and if we let $\alpha = ST(= abcd)$ and $\beta = T(= cd)$, then the rewrite system is

$$R = \{\alpha\beta x\alpha\alpha y\alpha \rightarrow \alpha\alpha y\alpha\beta x\alpha;\ \beta\beta x\beta\alpha y\beta \rightarrow \beta\alpha y\beta\beta x\beta \mid x, y \in (\alpha+\beta)^*\}$$

and the flat automaton (or rather its abstraction using the letters α, β instead of a, b, c, d: the whole flat automaton for $NF(R)$ would contain 76 states) is displayed on figure 25. All states are final.

3.4 The main theorem

Summing up what we have so far, thanks to lemma 34 and theorem 53, we can compute a flat automaton $\mathcal{A}_1$ which accepts at least one sequence of transitions for each equivalence class of $\approx$. We may moreover restrict the transition sequences accepted by $\mathcal{A}_1$ to transition sequences which are possible in the original automaton $\mathcal{A}$, thanks to a closure property for flat automata:

Lemma 62 *Let $\mathcal{A}$ be any finite automaton and $\mathcal{A}_1$ be a flat automaton. Then there is a flat automaton $\mathcal{A}'$ which accepts $L(\mathcal{A}) \cap L(\mathcal{A}_1)$.*

Moreover, there are two mappings f_1, f_2 from states of $\mathcal{A}'$ into respectively the states of $\mathcal{A}$ and the states of $\mathcal{A}_1$ such that, for every w and every states q, q', q_1, q_1', $\left.\begin{array}{c} q \xrightarrow[\mathcal{A}]{w} q' \\ q_1 \xrightarrow[\mathcal{A}_1]{w} q_1' \end{array}\right\} \Leftrightarrow$

$$\exists q_2, q_2'. \left\{\begin{array}{l} q_2 \xrightarrow[\mathcal{A}']{w} q_2' \\ f_1(q_2) = q, f_2(q_2) = q_1 \\ f_1(q_2') = q', f_2(q_2') = q_1' \end{array}\right.$$

The proof of this lemma is similar to that of lemma 59: we construct $\mathcal{A}'$ from the product automaton $\mathcal{A} \times \mathcal{A}_1$, then all loops on a state are power of a same word, thanks to the flatness of $\mathcal{A}_1$. Now, we are able to prove our main theorem:

Theorem 63 *Every timed automaton can be emulated by a flat timed automaton.*

<u>Proof</u> : Let $\mathcal{A}$ be any timed automaton and let $\mathcal{A}_1$ be a flat automaton which accepts at least one transition sequence for each class modulo $\approx$. Such an automaton does exist thanks to lemmas 34, 48 and theorem 53 ($w \approx w'$ implies $\phi_w \models\!\!\!\dashv \phi_{w'}$ as $\approx$ is the least congruence relation which satisfies the commutation properties of lemma 48). $\mathcal{A}$ can also be seen as a finite automaton on transition sequences and we construct $\mathcal{A}'$ as in lemma 62. $\mathcal{A}'$ can be seen as a timed automaton.

Now $\mathcal{A}'$ emulates $\mathcal{A}$: let f_1, f_2 be as in lemma 62. Then if $(q, \vec{V}) \xrightarrow[\mathcal{A}]{w} (q', \vec{V'})$, let $w' \approx w$. $(q, \vec{V}) \xrightarrow[\mathcal{A}]{w'} (q', \vec{V'})$, thanks to lemma 48. In particular if w' is the representative of the class of w w.r.t. $\approx$ which is accepted by $\mathcal{A}_1$ in state q_1', then $q_1 \xrightarrow[\mathcal{A}_1]{w'} q_1'$ where q_1 is the initial state of $\mathcal{A}_1$. It follows, by lemma 62, that there are states q_2, q_2' of $\mathcal{A}'$ such that $q_2 \xrightarrow[\mathcal{A}']{w'} q_2'$. Then $(q_2, \vec{V}) \xrightarrow[\mathcal{A}']{w'} (q_2', \vec{V'})$. Conversely, if $(q_2, \vec{V}) \xrightarrow[\mathcal{A}']{w} (q_2', \vec{V'})$, then $(f_1(q_2), \vec{V}) \xrightarrow[\mathcal{A}]{w} (f_1(q_2'), \vec{V'})$. Then it suffices to choose f_1 for the emulation function ϕ of definition 4. □

Example 64 Considering the automaton of figure 1, our automatic construction will not yield the automaton of figure 2 (unfortunately). We actually obtain an automaton which is isomorphic to the automaton of figure 25.

4 Expressibility of the reachability relation

We use here a transformation of timed automata into automata with counters and use a result on flat automata with counters.

Definition 65 ([10]) *An* automaton with (real valued) counters *is a tuple* $(Q, q_i, C, \delta \subseteq Q \times G(C, C') \times Q)$ *where*

- Q *is a finite set of* states
- $q_i \in Q$ *is an* initial state
- C *is a finite set of* counter names; C' *is the set of primed counter names.*
- $G(C, C')$ *is the set of* guards *built on the alphabets* C, C'. *A member of* $G(C, C')$ *is a conjunction of atomic formulas of one of the forms* $x \# y + c$, $x \# c$ *where* $x, y \in C \cup C'$, $\# \in \{\geq, \leq, =, >, <\}$. *and* $c \in \mathbb{R}$.

The automaton may *move* from a configuration $(q, \vec{v})$ to a configuration $(q', \vec{v'})$, which we write $(q, \vec{v}) \rightarrow (q', \vec{v'})$ if there is a triple $(q, g, q') \in \delta$ such that $\vec{v}, \vec{v'} \models g$, with the standard interpretation of relational symbols.

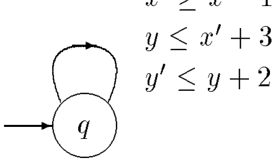

Figure 26: A flat counter automaton

Example 66 Consider the automaton of figure 26. The automaton may move from $(q, \begin{pmatrix} 1 \\ 1 \end{pmatrix})$ to $(q, \begin{pmatrix} 0 \\ 2 \end{pmatrix})$. We also have $(q, \begin{pmatrix} 1 \\ 1 \end{pmatrix}) \to (q, \begin{pmatrix} 3 \\ 3 \end{pmatrix})$ for instance.

Following [12], timed automata can be seen as a particular class of automata with counters: as explained in section 3.1.1, we use the variable transformation $x \mapsto \tau - x$. This yields a transformation on clocks valuations from $\vec{V}$ to $\vec{V_c}$. Then, if $< q, q', a, \phi, \lambda > \in E$, we translate it into a transition $\delta =< q, q', g >$ where g is the translation of ϕ together with the constraints $c' = \tau$ for each $c \in \lambda$ and $c' = c$ for each $c \notin \lambda$, plus the constraints on time positiveness: $\tau' \geq \tau$ and $c \leq \tau$ for every c. In this way each timed automaton $\mathcal{A}$ can be translated into an automaton with counters $\mathcal{A}_c$:

Theorem 67 ([12]) $(q, \vec{V}) \xrightarrow[\mathcal{A}]{*} (q', \vec{V'})$ *iff there is a vector* $\vec{V^1}$ *such that* $(q, \vec{V^1_c}) \xrightarrow[\mathcal{A}_c]{*} (q', \vec{V'_c})$ *and* $\vec{V^1} \models I(q), \vec{V} \models I(q), \vec{V^1} = \vec{V} + t \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}$ *for some* $t \geq 0$.

In other words, if we start a computation of $\mathcal{A}$ by firing a transition, then $\xrightarrow[\mathcal{A}]{*}$ is identical to $\xrightarrow[\mathcal{A}_c]{*}$, modulo the variable change. This result follows directly from lemma 9.

On the other hand, we have the following result on flat automata:

Theorem 68 ([10]) *For every flat real (resp. integer) counter automata, the relations* $\xrightarrow[q,q']{*}$ *are effectively definable in* $\mathcal{T}$ *(resp. Presburger arithmetic).*

Now, putting everything together, we have:

Theorem 69 *For every timed automata, the binary reachability relations* $\xrightarrow[q,q']{*}$ *are effectively definable in the additive theory of real numbers.*

5 Examples of properties which can be decided on timed automata

Using the reachability relation, it is possible to express that some "good" (resp. "bad") thing may (resp. may not) happen. This is typical for safety properties. However it is not (at least in an obvious way) possible to express that something *must* happen. Typically, we cannot express inevitability. Hence, only a fragment of timed temporal logics can be expressed in the first-order theory of the reachability relation. However, our formalism offers other possibilities.

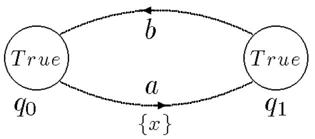

Figure 27: A very simple timed automaton

Example 70 On the simple automaton of figure 27, using the binary reachability relation, we can check properties of configurations, that are not properties of regions. This points out a difference in nature between our result and [19].

Consider the automaton of figure 27 which contains only one clock x, and no constant. Assume that we are interested in the following property: " After firing a in configuration c, what is the minimal delay spent before reaching c again ?"
For example, from state q_0 with $x = 0$ we can fire a and then b without waiting in q_1, then we are again in q_0 with $x = 0$. In this case the minimal delay is 0.

If the initial value x_0 of x is a parameter, then the minimal delay d is a function of x_0. In our example, we have $d(x_0) = x_0$. This result can be obtained using the binary reachability relation, since the set of possible delays is $\mathcal{D}(x_0) = \{\tau' - \tau \mid (q_0, x_0, \tau) \xrightarrow{ab} (q_0, x_0, \tau')\}$ and $d(x_0)$ is such that $d(x_0) \in \mathcal{D}(x_0)$ and $\forall d' \in \mathcal{D}(x_0), d(x_0) \leq d'$.

Such a minimal delay property cannot be obtained using classical methods since, usually, the computed delays cannot depend on the initial configuration. For instance, for the automaton of figure 27, there are two clock regions (depending on $x = 0$ or not). The minimal delay between two configurations of the region automaton is always 0.

More generally, it is possible to express sets of configurations which are not necessary unions of regions; any first-order definable set of clock values can be used in the logic. For instance we could express "each time the clock x is the double of clock y, we can reach a state in which y is a third of z, within a delay which is the half of x":

$$\forall \vec{X}, \exists \vec{X}', (x = 2y) \Rightarrow (\vec{X} \xrightarrow[\mathcal{A}]{*} \vec{X}' \wedge z' = 3y' \wedge 2(\tau' - \tau) = x)$$

Not only such sets of configurations are expressible, but also relations between configurations. For instance we can express that "each time we are in state q, we can reach a configuration in which the clock x has doubled". This corresponds to a relation $x' = 2x$.

Using free variables, it is also possible to define values of clocks (or delays). For instance, the minimal delay between configurations c_1, c_2 can be defined by:

$$\phi(x) \overset{\text{def}}{=} \exists \tau, (c_1, \tau) \xrightarrow[\mathcal{A}]{*} (c_2, \tau + x) \wedge \forall y < x, \forall \tau, \neg(c_1, \tau) \xrightarrow[\mathcal{A}]{*} (c_2, \tau + x)$$

and hence can be computed or used in further verifications.

We can take advantage of the *binary* relation: it is as easy to express properties about the past as about the future. For instance: "each time we reach a state q, then the clock x was never larger than $y + z$ in the past".

We conclude this section with an example which looks more relevant: assume we have a server which receives requests from several users. Assume that the server receives requests from two users at time t and $t + \epsilon$ and that these requests are granted at time $t + \delta_1$ and $t + \delta_2$. We may want to check that the server is "fair" and that the delay δ_2 is always smaller than $2 \times \delta_1 + \epsilon$, for instance. This is again a typical property which can be expressed and checked in our theory.

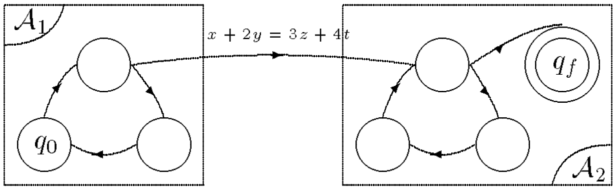

Figure 28: An extended timed automaton

6 Conclusion and perspectives

We have shown that, for timed automata, the binary reachability relation is definable in a decidable theory. The formula which results from our computation may be, in principle, quite huge since for instance the number of equivalence classes for $\sim$ is exponential. Hence our result is, so far, more of theoretical nature. One possible future research direction is then to have more precise informations on the complexity of the method (both theoretically and practically). However, beyond an hypothetical practical verification technique for timed automata, we believe that our result shows another possible direction of research. It may suggest other (more tractable) real-time computation models, starting from the logical side (the theory of real numbers). For instance, we could start from flat timed automata (without loosing expressiveness in the clock valuations sense). It also separates the expressiveness of the properties to be checked (in which it is possible to express much more relationships between clocks) from the expressiveness of the model. There are also several side effects of our proof. For instance we can decide whether or not a loop (or several loops) can be iterated.

The ability to express the reachability relation as a constraint between initial and final clock values allows to replace a whole automaton (or a piece of it corresponding to a timed automaton) with a single *meta-transition* [9], hence faithfully abstracting complex models. This can be used in verifying complex systems.

Conversely, we can mechanically check properties of models which are more expressive than timed automata.

Example 71 We consider two timed automata which are connected by a single transition whose enabling condition is an arbitrary first-order formula ψ over clock values (see figure 28). Properties of such a network can be verified as easily (or as hardly) as properties of a single timed automaton: it suffices to compute the binary reachability relation for each individual automaton and then to connect the two formulas with the enabling condition ψ. This can be extended of course to any network of timed automata, provided that there is no cycle through such general transitions.

Adding a stop watch to the model yields undecidability of the reachability [13]. However, as shown in [3], this does not imply that we cannot check properties involving accumulated delays. It is not possible, at least in an obvious way, to express accumulated delay constraints using the first-order theory of the reachability relation. We believe that it is still worth to study more deeply what can (and what cannot) be automatically checked using a similar approach.

Another interesting possible investigation consists in considering *parametrised* timed automata, as in [8]. Though the authors show that emptiness of such automata is undecidable as soon as there are at least three clocks, our method seems to be well-suited for parametric reasoning and, for instance, we may derive conditions on the control instead of on the number of clocks, which yield decision techniques for such parametrised automata.

Finally, our method seems to be well-suited for models which combine timed automata with additional global variables: assume we add registers to the model, with simple operations on them such as in definition 65, then, for flat timed automata with counters, we expect to get again a decision procedure.

References

[1] R. Alur. Timed automata. In *Verification of Digital and Hybrid Systems, Proc. NATO-ASI Summer School, Antalya, Turkey*, 1997. To appear.

[2] R. Alur, C. Courcoubetis, and D. Dill. Model-checking in dense real time. *Information and Computation*, 104(1):2–24, 1993.

[3] R. Alur, C. Courcoubetis, and T. Henzinger. Computing accumulated delays in real-time systems. In *Proc. 5th Conf. on Computer Aided Verification*, volume 818 of *Lecture Notes in Computer Science*, pages 181–193. Springer-Verlag, 1993.

[4] R. Alur and D. Dill. Automata for modeling real-time systems. In *Proc. 17th Int. Coll. on Automata, Languages and Programming, Warwick, LNCS 443*, pages 322–335. Springer-Verlag, 1990.

[5] R. Alur and D. Dill. A theory of timed automata. *Theoretical Computer Science*, 126:183–235, 1994.

[6] R. Alur, T. Feder, and T. Henzinger. The benefits of relaxing punctuallity. *J. ACM*, 43:116–146, 1996.

[7] R. Alur and T. Henzinger. A really temporal logic. *J. ACM*, 41:181–204, 1994.

[8] R. alur, T. Henzinger, and M. Vardi. Parametric real-time reasoning. In *Proc. 25th Annual ACM Symposium on Theory of Computing*, 1993.

[9] B. Boigelot and P. Wolper. Symbolic verification with periodic sets. In *Computer Aided Verification, Proc. 6th Int. Conerence*, LNCS, Stanford, June 1994. Springer-Verlag.

[10] H. Comon and Y. Jurski. Multiple counters automata, safety analysis and presburger arithmetic. In A. Hu and M. Vardi, editors, *Proc. Computer Aided Verification*, volume 1427 of *LNCS*, pages 268–279, Vancouver, 1998. Springer-Verlag.

[11] C. Courcoubetis and M. Yannakakis. Minimal and maximal delay problems in real time systems. In K. Larsen and A. Skou, editors, *Proc. CAV 91: Computer Aided Verification*, volume 575 of *Lecture Notes in Computer Science*, pages 399–409. Springer-Verlag, 1991.

[12] L. Fribourg. A closed form evaluation for extending timed automata. Technical Report 1998-02, Laboratoire Spécification et Vérification, ENS Cachan, Mar. 1998.

[13] T. Henzinger, P. Kopke, A. Puri, and P. Varaiya. What is decidable about hybrid automata ? In *Proc. 27th Symposium on Theory of Computing*, pages 373–382. ACM Press, 1995.

[14] T. Henzinger, X. Nicollin, J. Sifakis, and S. Yovine. Symbolic model checking for real time systems. *Information and Computation*, 111(2):193–244, 1994.

[15] F. Laroussinie, K. Larsen, and C. Weise. From timed automata to logic – and back. In *Proc. 20th Conf. on Foundations of Computer Science*, volume 969 of *Lecture Notes in Computer Science*, Prag, 1995. Springer-Verlag.

[16] M. Lothaire. *Combinatorics on words*, volume 17 of *Encyclopedia of mathematics and its applications*. Cambridge University Press, 1982.

[17] M. Lothaire. *Combinatorics on words.* Draft, 1999. The second version of the book is available at http://www-igm.univ-mlv.fr/ berstel/Lothaire/index.html.

[18] Morse and Hedlund. Symbolic dynamics ii. sturmian trajectories. *American J. Math.*, 62:1–42, 1940.

[19] F. Wang. Timing behaviour analysis for real time systems. In *Tenth Annual IEEE Symposium on Logic in Computer Science*, San Diego, CA, June 1995. IEEE Comp. Soc. Press.

[20] T. Wilke and M. Dickhöfer. The automata-theoretic method works for tctl model checking. Technical Report 9811, Inst. f. Informatik u. Prakt. Math., CAU Kiel, 1998.