Timed Automata with Periodic Clock Constraints

Christian Choffrut
L.I.A.F.A., Université Paris 7, Tour 55-56, 1 er étage,
2 Pl. Jussieu – 75 251 Paris Cedex – France
Christian.Choffrut@liafa.jussieu.fr

Massimiliano Goldwurm
Dipartimento di Scienze dell'Informazione, Università degli Studi di Milano
via Comelico, 39 – 20135 Milano – Italia
goldwurm@dsi.unimi.it

July 1998

Abstract

The traditional constraints on the clocks of a timed automaton are based on real intervals, e. g., the value of a clock belongs to the interval $(0, 1)$. Here, we introduce a new set of constraints, which we call "periodic", and which are based on regularly repeated real intervals, e. g., the value modulo 2 of a clock belongs to the interval $(0, 1)$ which means that it belongs to $(0, 1)$ or $(2, 3)$ or $(4, 5)$ ....

Automata with these new constraints have greater expressive power than the automata with traditional sets while satisfiability remains decidable. We address questions concerning ϵ-moves: simulation of automata with periodic constraints by automata with traditional constraints and removal of ϵ-moves under certain conditions. Then, we enrich our model by introducing "count-down" clocks and show that the expressive power is not increased. Finally, we study three special cases: 1) all transitions reset clocks, 2) no transition reset clocks, and 3) the time domain is discrete and prove the decidability of the inclusion problem under each of these hypotheses.

Keywords: model checking, real-time, synchronized relations, timed automata.

1 Introduction

Emphasis on concrete time, i. e., on *when* events occur and not only in *which order* they occur, is a vivid concern of the ongoing research on how real-time systems should be modelled or verified. Among the most popular models to be found in the literature are different kinds of real-time temporal logics that are extensions of the classical time or branching temporal logics, [18], [4], [19] and different types of timed finite automata obtained by providing traditional automata with clocks controlling the triggering of transitions. A third approach is based on fragments of monadic second order logics by resorting to a restricted use of distance between two elements on the real line, [3], [19]. Under some hypotheses, the expressive equivalence of these models can be established. The great merit of these extensions of "untimed" notions is that they are very powerful while maintainig decidable many interesting questions.

The present paper deals more specifically with the "automata" model as exposed in [1] for example. More precisely, a fixed set of clocks is given along with an automaton. These clocks share the same unit of time but can be reset independently when a transition is traversed. A transition is enabled provided certain conditions on the values of the clocks are met. This initial model has been extended in various directions to accomodate precision of computation [14], ϵ-transitions (also called *silent* transitions, ϵ-moves or *silent* moves) [6, 9], convergent sequences of time [7], hybrid models, [15] etc,....

The original model of timed automata is specified in terms of conditions envolving intervals only: e. g., a transition is enabled when $1 < x < 2$ and $y > 5$ holds. In a certain sense it allows counting up to or from a certain threshold value. It also allows conditions of the previous form where the difference of two clocks is substituted for one single clock, but it is well known that the expressive power does not increase. On the other hand, with more general conditions such as $x - 2y > 3$ the emptiness problem becomes undecidable. The notion of periodic clock constraints stems from the theory of rational relations on $\mathbb{N}$. It carries over to $\mathbb{R}_+$ easily but we explain it in the case of $\mathbb{N}$. Every set of linear equalities and inequalities envolving k integer variables, defines a rational subset of $\mathbb{N}^k$, i., e., a subset obtained from finite subsets by applying finitely many times the operations of set union, componentwise sum and Kleene star, [13]. The relations associated with the traditional clock constraints such as above, define on the other hand very specific rational relations (they define a "small" subfamily of the recognizable relations). Here we make use of a family of relations that lies between these two extreme cases in such a way as to extend the expressive power of the traditional constraints while keeping the emptiness problem decidable. This is done by allowing "modulo" counting primitives, e. g., by considering for a given clock x the values $2, 5, \ldots, 3k+2, \ldots$. For this reason we call the constraints "periodic" as opposed to the traditional "aperiodic" ones. This is a natural extension since, regardless of the previous theoretical reason, many processes in different fields (physical, biologic, social, etc..) have a periodic behaviour. To our knowledge this formalism was mentioned in [5] only for the discrete time [1]. The use of periodic constraints has the merit of increasing the expressive power of classical timed automata and of reducing significantly the number of transitions needed to produce a given behaviour (see example of subsection 3.2). At the same time it keeps the relevant properties of the classical model like the decidability of the satisfiability problem.

The main results of this work are the following. We start by comparing the expressive powers of timed automata using periodic and aperiodic constraints. We give a construction that preserves determinism and that transforms automata with periodic clock constraints into automata with aperiodic clock constraints at the cost of introducing some possible ϵ-transitions. In fact the expressive power of the family of automata with periodic constraints and no ϵ-transitions lies strictly between the family of automata with aperiodic constraints and no ϵ-transitions and the family of automata with aperiodic constraints and possible ϵ-transitions.

Next we tackle the problem of removing ϵ-transitions. For finite (whether one- or multi-tape) untimed automata, such moves can be eliminated easily. For timed automata this does not hold any longer. It was observed in [6] that ϵ-transitions without resets can be removed in automata using aperiodic constraints. The same result holds with the

[1] Writing the final version of this work we came to know that periodic clocks on dense time have been considered in [8] as a special case of a more general model called "control timed automaton".

new set of constraints. Some words of explanations are here in order. We are able to establish this result by making an excursion into non-commutativity and more specifically into "synchronous" relations, [12]. Indeed, the condition stated in [6] guaranteeing the possibility of removing the silent moves is a direct consequence of two simple closure properties of synchronous relations. This observation spares us the trouble of making a tiresome case study. With traditional constraints, further ϵ-transitions may be eliminated, to wit those that do not lie on a loop. This no longer holds in our case for a reason that is closely related to some Zeno property.

Dually to clocks that measure the time elapsed since an event occurred, we may consider "count-down" clocks measuring the time left before an event occurs. Many examples of some type of count-down clocks can be found in the literature, whether in the framework of automata or temporal logics, [2], [4], [19]. We show that for automata with periodic clock constraints, "count-down" clocks can be eliminated without introducing new ϵ-moves.

In section 6 we investigate two special cases, reset-free and pure reset automata with periodic constraints. Under these particular hypotheses a stronger result holds since inclusion of languages recognized by timed automata is decidable. We also show that when the domain is the set of integers (with possible ϵ-transitions and periodic clock constraints), all clocks can be replaced by a unique pure reset clock, which extends some results of [5] and [16].

2 Preliminaries

Our model of time $\mathbb{T}$ is either the set $\mathbb{N}$ of integers or the set $\mathbb{R}_+$ of nonnegative reals. Given a finite alphabet Σ, a timed sequence is a finite or infinite sequence of the form $(\sigma_1, t_1)(\sigma_2, t_2)\ldots$, where the σ_i's belong to Σ and the sequence $t_1, t_2, \ldots,$ is strictly increasing and divergent. In order to specify subsets of timed words, the standard notion of finite automaton is modified by introducing enabling times for transitions. These times are controlled by predicates which determine when the transition may be executed. There is no universal time of reference given with the automaton, only a fixed number of clocks sharing the same unit of time but that can be reset independently. We start by giving a general definition of clock constraints.

Definition 1 *Given a set X of n clocks, i. e., of real variables, a set $\Phi(X)$ of* clock constraints *is a family of subsets of $\mathbb{R}_+^n$ that is a closed under the Boolean set operations.*

The following is the usual definition of an automaton that recognizes finite and infinite timed strings at once.

Definition 2 *A* timed (Büchi) automaton *on $\Phi(X)$ is a tuple $\mathcal{A} = (\Sigma, Q, I, F, T, R, X)$ where*

Σ *is a finite set of* events*,*
Q *is a finite set of* states*,*
I *is a finite set of* initial states*,*
F *is a finite set of* final states*,*
R *is a finite set of* repeated states*,*

X is a finite set of real valued clocks,
$T \subseteq Q \times \Phi(X) \times (\Sigma \cup \{\epsilon\}) \times 2^X \times Q$ *is a finite set of* transitions.

We recall briefly the meaning of the transitions. An *assignment* of the clocks is a function of $\nu : X \to \mathbb{T}$. A finite (resp. infinite) timed sequence $(\sigma_1, t_1)(\sigma_2, t_2)\ldots, (\sigma_i, t_i)\ldots$ is *accepted* by $\mathcal{A}$ if there exist

1) a sequence of assignments ν_i of the clocks, $i = 0, 1, 2, \ldots$, with $\nu_0(x) = 0$ for every x,

2) a sequence of states q_i, $i = 0, 1, 2, \ldots$, with $q_0 \in I$, ending in F (resp. visiting infinitely many times R),

such that for all $i \geq 1$ there exists a transition $(q_{i-1}, \phi_i, \sigma_i, X_i, q_i)$ such that the set of values $\nu_{i-1}(x) + t_i - t_{i-1}$ satisfy the constraints ϕ_i (with the initial condition $t_0 = 0$) and the following holds for every $x \in X$:

$$\nu_i(x) = \begin{cases} 0 & \text{if } x \in X_i \\ \nu_{i-1}(x) + t_i - t_{i-1} & \text{otherwise} \end{cases}$$

Whenever ϵ is taken as the third component of a transition, we say it is an *ϵ-transition* or a *silent move*. Each sequence with ϵ-transitions defines a timed sequence by simply ignoring the ϵ-transitions.

Moreover, we recall that a timed Büchi automaton with possible ϵ-transitions is *deterministic* whenever for two different transitions with the same first component, the following holds

$$\begin{gathered} \text{for all } \sigma_1, \sigma_2 \in \Sigma \cup \{\epsilon\}, \text{ for all } (q, \phi_1, \sigma_1, X_1, p_1), (q, \phi_2, \sigma_2, X_2, p_2) \in T \\ \sigma_1 = \sigma_2 \text{ or } \sigma_1 = \epsilon \text{ or } \sigma_2 = \epsilon \text{ implies } \phi_1 \wedge \phi_2 = \texttt{false} \end{gathered}$$

The *language accepted* by the time automaton $\mathcal{A}$ is the set of all the finite and infinite timed sequences accepted by $\mathcal{A}$. Two timed automata are *equivalent* if they accept the same language.

3 New clock constraints

By an *interval* of $\mathbb{R}_+$ we mean an open (resp. left semi-open, right semi-open, closed) interval of the form (a, b) (resp. $(a, b]$, $[a, b)$, $[a, b]$) with $a, b \in \mathbb{R}_+$. The interval is *rational* (resp. *integer*) if further $a, b \in \mathbb{Q}_+$ (resp. $a, b \in \mathbb{N}$).

We are now in condition to define the clock constraints whose study is the main purpose of this paper.

DEFINITION **3** *Given a set X of clocks, the set $\Phi(X)$ of* periodic clock constraints *is inductively defined by*

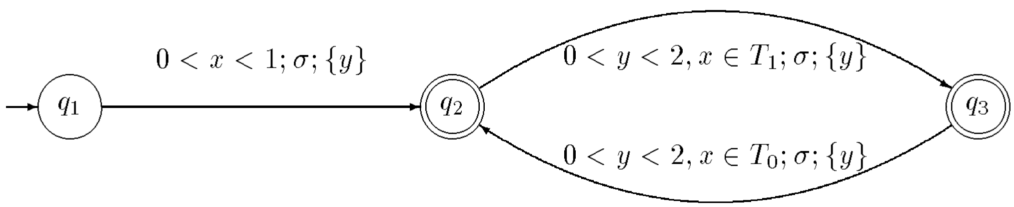
$$\phi := \exists k \in \mathbb{N} : x \in [a + k\lambda, b + k\lambda] \mid \exists k \in \mathbb{N} : x - y \in [a + k\lambda, b + k\lambda] \mid \neg\phi \mid \phi_1 \wedge \phi_2$$

where $a, b, \lambda \in \mathbb{Q}$, and $x, y \in X$. A constraint involving atoms with only one clock is called non-diagonal.

We recall that the traditional constraints, call them *aperiodic*, have $x \leq a$ and $x \geq a$ as unique atomic formulae. The latter formula, e. g., is expressed in our language by $\exists k \in \mathbb{N} : x \in [a + k\lambda, b + k\lambda]$ with $\lambda = b - a$.

When ϵ-transitions are allowed, periodic and aperiodic clock constraints have the same expressive power as shown in paragrah 3.2. However, when no ϵ-moves are allowed, the new clock constraints are strictly more expressive than the traditional ones: e. g., the languages of timed automata $\mathcal{A}_{Delay(k,p)}$ ($t_i - t_{i-1} = k \bmod p$), $\mathcal{A}_{Even}$ ($t_i \in 2\mathbb{N}$) and $\mathcal{A}_{Int}$ ($t_i \in \mathbb{N}$), introduced in [6], can be recognized by timed automata on periodic clock constraints without ϵ-transitions. A more significant example is the following.

Example **3.1** The set of all strings $(\sigma, t_1)(\sigma, t_2)\ldots(\sigma, t_i)\ldots$ such that $i - 1 < t_i < i$ for all $i \geq 1$, is not recognized by any ϵ-free timed automaton on traditional clock constraints [17]. Define $T_0 = \{t \in \mathbb{R} \mid \exists n \in \mathbb{N}, 2n < t < 2n + 1\}$ and $T_1 = \{t \in \mathbb{R} \mid \exists n \in \mathbb{N}, 2n + 1 < t < 2n + 2\}$. Then, an ϵ-free automaton with periodic constraints is described by the following picture:



Now, let us modify the condition on the t_i's by requiring that there exist a real $\alpha > 0$ such that $\alpha + i - 1 < t_i < \alpha + i$ holds for all $i \geq 1$. An automaton with periodic constraints recognizing this set can be obtained from the previous one by transforming q_1 into an ordinary state and by adding a new initial state q_0 and the ϵ-transition

$$q_0 \xrightarrow{x > 0;\ \epsilon;\ \{x, y\}} q_1$$

On the other hand, this language is no longer recognizable by any timed automaton on periodic constraints without ϵ-transitions. Indeed, assume to the contrary that this is the case and let δ be the inverse of the least common multiple of the denominators of the coefficients involved in the constraints. Then, assuming $X = \{x_1, x_2, \ldots, x_m\}$ the set of clocks, any clock predicate occurring in the automaton has a constant value over the open intervals of the form $\prod_{1 \leq i \leq m} (k_i\delta, (k_i + 1)\delta)$, where $k_i \in \mathbb{N}$ for each i.

Let now $k > 0$ and consider the sequences $\{t_n\}$ and $\{t'_n\}$ defined by $t_1 = t'_1 = k$, $t_n = k + n - \frac{\delta}{2} - \frac{\delta}{2^{n+1}}$ and $t'_n = k + n - \frac{\delta}{2^{n+1}}$ for all $n > 1$. Then, we have $t_n, t'_n \in (k + n - \delta, k + n)$ and $t_n - t_1, t'_n - t'_1 \in (n - \delta, n)$ for all $n > 1$, while, for all $1 < m < n$, we have $t_n - t_m, t'_n - t'_m \in (n - m, n - m + \delta)$. By the form of the constraints this means that the behaviour of the automaton over the two sequences is the same. However, the timed sequence associated with $\{t_n\}$ belongs to the language while that one associated with $\{t'_n\}$ does not.

□

3.1 Canonical form

In order to reduce the complexity of the proofs we show that every timed automaton with periodic constraints is equivalent to a timed automaton with simpler constraints.

First, we make the usual assumption that the constants are integers. Indeed, if $\mathcal{A}$ is a timed automaton it is possible to multiply all the constants occurring in the constraints by the least common multiple m of their denominators. Then all constants have integral values and $\mathcal{A}$ recognizes the sequence $(\sigma_1, t_1)(\sigma_2, t_2)\ldots$ if and only if the new automaton $\mathcal{A}'$ recognizes the sequence $(\sigma_1, mt_1)(\sigma_2, mt_2)\ldots$. However the most significant simplification is due to the following result whose proof is given in the appendix. Observe that strictly speaking it does not yield a canonical form, just a sort of simplified form. Given a subset $H \subseteq \mathbb{R}$ and an integer $m \in \mathbb{N}$, the notation $T_m(H)$ stands for the set $\{x = h + km \mid h \in H, k \in \mathbb{N}\}$.

THEOREM 1 *Let $\mathcal{A}$ be a timed automaton on periodic clock constraints with integer constants. Then, $\mathcal{A}$ is equivalent to a timed automaton $\mathcal{A}'$ for which there exists an integer m such that each constraint is a conjunction of conditions of the following form:*

$$i) x = i, \quad ii) x \in (i, i+1), \quad iii) x \in T_m[i], \quad iv) x \in T_m(i, i+1) \tag{1}$$

where $0 \leq i < m$. Furthermore, $\mathcal{A}'$ is deterministic if so $\mathcal{A}$ is.

3.2 Simulation of periodic clock constraints

We apply the previous results for establishing the equivalence of periodic and aperiodic clock constraints when ϵ-transitions are allowed.

THEOREM 2 *For each timed (resp. deterministic timed) automaton with periodic clock constraints, there exists an equivalent timed (resp. deterministic timed) automaton with aperiodic clock constraints.*

Proof. The following construction is valid under both deterministic and nondeterministic hypotheses.

We assume the clock constraints of all transitions of the given automaton $\mathcal{A}$ to be as in Theorem 1. The idea of the proof is to associate with every clock x a twin clock $\overline{x}$ which records the value of x modulo $m\mathbb{Z}$, where m is defined as in Theorem 1. In other words, denoting by $x(t)$ the value of a clock $x \in X$ at time $t \in \mathbb{R}_+$, any two twin clocks x and $\overline{x}$ satisfy the relation

$$\overline{x}(t) = x(t) - \lfloor \frac{x(t)}{m} \rfloor m$$

for every $t \in \mathbb{R}_+$.

Associate with each clock $x \in X$ its twin clock $\bar{x}$ and extend this notation to subsets by setting $\bar{Y} = \{\bar{y} \mid y \in Y\}$ for every $Y \subseteq X$. We construct a new automaton on the same set of states of $\mathcal{A}$, whose transitions are subject to new constraints on the clocks $\bar{x}$'s.

In order to explain the construction we show how a given transition $q \xrightarrow{H,\sigma,Y} p$ in $\mathcal{A}$, with $\sigma \in \Sigma \cup \{\epsilon\}$ and $Y \subseteq X$, is modified. By the previous result, H is a conjunction of constraints of the form (1). The condition $\bar{H}$ is obtained from H by replacing each occurrence of $x \in T_m[i]$ and $x \in T_m(i, i+1)$ by $\bar{x} = i$ and $\bar{x} \in (i, i+1)$, respectively. Note that $\bar{H}$ keeps unchanged the possible aperiodic constraints of H. Moreover, for all subsets $Z \subseteq X$, we denote by $\psi(Z)$ the predicate $\bigwedge_{x\in Z} \bar{x} = m \wedge \bigwedge_{x\notin Z} \bar{x} < m$.

Then, the following transitions are in the new automaton:

$$q \xrightarrow{\bar{H} \wedge \bigwedge_{x\in X} \bar{x} < m;\ \sigma;\ Y \cup \bar{Y}} p$$

and, for all nonempty $Z \subseteq X$,

$$q \xrightarrow{\neg\bar{H} \wedge \psi(Z);\ \epsilon;\ \bar{Z}} q$$

$$q \xrightarrow{\bar{H} \wedge \psi(Z);\ \sigma;\ Y \cup \bar{Y} \cup Z} p$$

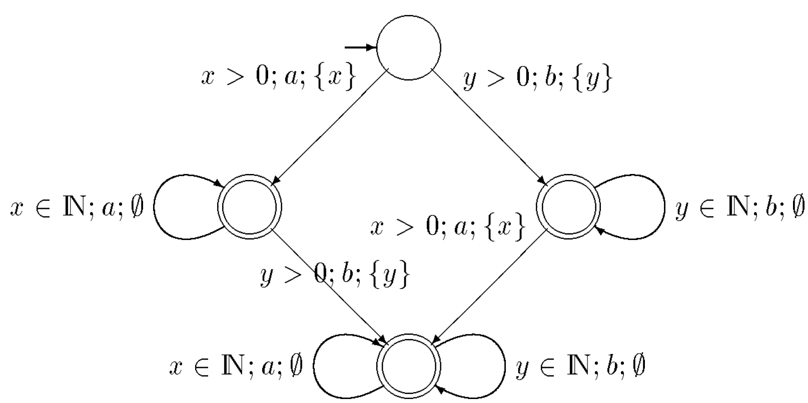
■

The construction described in the previous proof shows that eliminating periodic constraints may increase exponentially the number of transitions of the automaton relative to the number of clocks. The following example shows an automaton with two clocks. The reader may try to figure out by himself how to design an automaton that is equivalent to the given one and that uses aperiodic constraints and ϵ-transitions. It should also give some evidence of how the use of periodic constraints can reduce the size of the automaton recognizing a given timed language.

EXAMPLE **3.2** Let Σ be the alphabet $\{a, b\}$ and consider the set of timed sequences given by

$$\{\{\sigma_i, t_i\}_{i\geq 1} \mid \forall i \quad \sigma_i \in \Sigma, \exists u, v > 0 : \sigma_i = a \Rightarrow t_i - u \in \mathbb{N}, \sigma_i = b \Rightarrow t_i - v \in \mathbb{N}\}$$

Such a language is recognized by the following timed automaton.



□

4 Removal of ϵ-moves without clock-resets

In [6] the question is raised to determine under which conditions ϵ-transitions can be removed. Examples are given where ϵ-transitions are necessary. We address this issue now.

Observe that for finite (untimed) automata, whether one or multi-tape, such moves can be eliminated. For timed automata this is no longer true, but we show in the sequel that, exactly as in the mentioned paper, ϵ-transitions with no clock resets can be suppressed.

These conditions rely on closure properties stated in Theorem 3. In the case of aperiodic clock constraints they can be verified through a tedious process of case study. Instead we prefer to transform them into properties of formal languages. Indeed, we are able to interpret the periodic clock constraints as synchronized relations over $\mathbb{N}^k$ which enjoy these properties. This explains why we make a detour through relations on strings, however far away from timed automata it may seem.

4.1 Forward and backward closures

We recall a notion introduced in [6] in order to show that certain silent moves may be eliminated. With every subset $A \subseteq \mathbb{R}_+^n$ and every real $t \in \mathbb{R}$ we define

$$A + t = \{(x_1 + t, x_2 + t, \ldots, x_n + t) \mid (x_1, x_2, \ldots, x_n) \in A\}$$

Given a subset $A \subseteq \mathbb{R}_+^n$ we define the unary operation of *forward closure* $\overrightarrow{A}$ of A by setting

$$\overrightarrow{A} = A + \mathbb{R}_+$$

Equivalently, given a subset $A \subseteq \mathbb{R}_+^n$ and a subset $I \subseteq \{1, \ldots, n\}$, we define the *backward* closure $\overleftarrow{A}^I$ of A relative to I as the subset

$$\overleftarrow{A}^I = \pi_I^{-1}(A - \mathbb{R}_+)_I \tag{2}$$

where for all $B \subseteq \mathbb{R}_+^n$, B_I is the set of elements $(x_1, \ldots, x_n) \in B$ with $x_i = 0$ for all $i \in I$, and π_I is the projection of $\mathbb{R}_+^n$ onto the subspace determined by the equations $x_i = 0$ for all $i \in I$, cf. [6].

The following general result was established in [6, Theorem 22].

THEOREM **3** *Given a family of clock constraints that is a closed under backward and forward closures, every restricted automaton is equivalent to a timed automaton with no ϵ-transition.*

Since $(A - \mathbb{R})_I \cap (A - \mathbb{R})_J = (A - \mathbb{R})_{I \cup J}$ holds for any pair of subsets $I, J \subseteq X$, a set Φ of constraints is closed under backward closure if it is closed under the inverse projections π_I^{-1} and it contains the sets $\overleftarrow{A}^{\{x\}}$ for all $x \in X$ and every A in Φ.

4.2 Synchronized relations

Let us recall the notion of synchronized relation over a direct product of n free monoids $\Sigma_1^* \times \ldots \times \Sigma_n^*$. For all $i = 1, \ldots, n$ set $\Sigma_i^\sharp = \Sigma_i \cup \{\sharp\}$ and set $\Delta = \Sigma_1^\sharp \times \ldots \times \Sigma_n^\sharp - (\sharp, \ldots, \sharp)$. With every n-tuple $(u_1, \ldots, u_n) \in \Sigma_1^* \times \ldots \times \Sigma_n^*$ associate the tuple $(u_1, \ldots, u_n)^\sharp = (v_1, \ldots, v_n)$ as follows: let ℓ_i be the length of u_i and $\ell = \max\{\ell_i \mid 1 \leq i \leq n\}$. Then $v_i = u_i \sharp^{\ell - \ell_i}$. Extend this notation to subsets $H \subseteq \Sigma_1^* \times \ldots \times \Sigma_n^*$ by setting $H^\sharp = \{(u_1, \ldots, u_n)^\sharp \mid (u_1, \ldots, u_n) \in H\}$. An n-tuple such as $(v_1, \ldots, v_n) = (u_1, \ldots, u_n)^\sharp$

can be viewed as a string of length ℓ over the alphabet Δ. Indeed, writing for all $i = 1 \ldots, n$ $v_i = v_i^{\{1\}} \ldots v_i^{\{\ell\}}$ we obtain $(v_1, \ldots, v_n) = (v_1^{\{1\}} \ldots v_n^{\{1\}}) \ldots (v_1^{\{\ell\}} \ldots v_n^{\{\ell\}})$. Then H is a *synchronized relation* if the relation $H^\sharp$, viewed as a subset of Δ^* is recognized by some finite automaton. Furthermore, it is an *aperiodic synchronized relation* if it can be recognized by some finite automaton where no loop is labelled by a word w^k with $k > 1$ and w is not the empty word. It is well known that the family of synchronized relations is closed under Boolean operations, cf. [10]. Observe that whenever $n = 1$ and Σ_1 is a singleton, a synchronized relation R is just a regular subset of $\mathbb{N}$, i. e., there exist an integer $p \geq 0$ and two finite subsets $A \subseteq \mathbb{N}$, $B \subseteq \{1, \ldots, p\}$, such that

$$R = A \cup \bigcup_{i \in B} \{i + kp \mid \text{ for some } k \in \mathbb{N}\} \tag{3}$$

The relation is aperiodic if further $p = 1$.

Define the *support* of a relation H as the subset $Supp(H)$ of integers $i \in \{1, \ldots, n\}$ for which there exists a n-tuple $(x_1, \ldots, x_n) \in H$ where x_i is not the empty word. A synchronized relation is *length-preserving* if $(x_1, \ldots, x_n) \in H$, $i, j \in Supp(H)$ implies x_i and x_j have the same length.

Furthermore, if $K \subseteq \Sigma_1^* \times \ldots \times \Sigma_n^*$ is a length-preserving synchronized relation and if H is synchronized, then the following subsets are also synchronized:

$$KH = \{(u_1 v_1, \ldots, u_n v_n) \mid (u_1, \ldots, u_n) \in K \text{ and } (v_1, \ldots, v_n) \in H\} \tag{4}$$

$$K^{-1}H = \{(v_1, \ldots, v_n) \mid \exists (u_1, \ldots, u_n) \in K \text{ and } (u_1 v_1, \ldots, u_n v_n) \in H\} \tag{5}$$

We will apply the present notions when all free monoids Σ_i^*'s are isomorphic to the additive monoid $\mathbb{N}$. Applied to clock constraints, the previous two operations are closely related to the forward and backward closures of a relation.

The following can be easily established by resorting to standard automata-theoretic methods.

PROPOSITION **1** *Each synchronized relation (resp. aperiodic synchronized relation) on* $\Sigma_1^* \times \ldots \times \Sigma_n^*$ *is a finite union of subsets of the form* $H_1 H_2 \ldots H_k$ *for some* $k > 0$ *where* $\{1, \ldots, n\} \supseteq Supp(H_1) \supset Supp(H_2) \ldots \supset Supp(H_k)$ *and the* H_i*'s are synchronized (resp. aperiodic synchronized) length-preserving relations.*

4.3 Logical characterization of synchronized relations

The logical characterization of synchronized relations of a direct product of free monoids was established in [11]. In the case where all alphabets Σ_i's are unary which is the case of clock constraints, we can find a logical characterization by very simple formulae.

We recall that a subset R is *defined* by a formula ϕ of the previous logic if R is the set of n-tuples $(a_1, \ldots, a_n) \in \mathbb{N}^n$ such that ϕ is true whenever each x_i is substituted for a_i in ϕ.

PROPOSITION **2** *A relation* R *on* $\mathbb{N}^n$ *is synchronized if and only if it is defined by a formula of the logic given in Definition 3, where however the constants* a, b, λ *are integers and the variables are interpreted over the set of positive integers.*

Proof. Since the family of synchronized relations is a Boolean algebra it suffices to verify that the atoms of the logic define synchronized relations but this is trivial.

Conversely, let R be a synchronized relation. Let us deal with a simple case first: $R = \{a + k\lambda \mid k \geq 0\} \subseteq \mathbb{N}$ with $a, \lambda \in \mathbb{N}$. Then, R is defined by the formula

$$\phi_{a,\lambda}(y) \equiv (\exists k \geq 0, a + k\lambda = y)$$

Let us now turn to the general case. By Proposition 1 and by closure under finite union, we may assume without loss of generality that R is of the form $H_1 H_2 \ldots H_k$ for some $k > 0$ where $\{1, \ldots, n\} \supseteq Supp(H_1) \supset Supp(H_2) \ldots \supset Supp(H_k)$ and each projection of H_i onto any component is as in the above special case. Define $I_i = Supp(H_i) - Supp(H_{i-1})$, $i = 1, \ldots, k-1$ and $I_k = Supp(H_k)$. Choose arbitrarily an element $r_i \in I_i$, $i = 1, \ldots, k$. Then R can be expressed by the formula

$$(\phi_{a_1,p_1}(y_{r_1}) \wedge \bigwedge_{i \in I_1} y_i = y_{r_1}) \wedge \bigwedge_{j=2}^{k} \{\phi_{a_j,p_j}(y_{r_j} - y_{r_{j-1}}) \wedge \bigwedge_{i \in I_j} y_i = y_{r_j})\}$$

for some integers $a_1, a_2, \ldots, a_k, p_1, p_2, \ldots, p_k$.

■

Remark 4.1 It is not difficult to show that aperiodic synchronized relations in $\mathbb{N}^n$ are characterized by the set of propositional formulae whose atoms are of the form $a \leq x \leq b$ and $a \leq x - y \leq b$ with $a, b, \lambda \in \mathbb{N}$, x, y are variables.

4.4 Application to ϵ-transitions

In order to prove that ϵ-transitions can be removed with our constraints when no clock reset is performed, it suffices to prove that the family of our constraints is closed under the Boolean operations, backward and forward closures.

THEOREM 4 *For each timed automaton with periodic clock constraints and whose ϵ-moves are reset-free, there exists an equivalent automaton with periodic clock constraints and no ϵ-moves.*

Proof. By equations (4) and (5), if $A \subseteq \mathbb{R}_+^n$ is defined by periodic clock contraints, then its forward closure $\overrightarrow{A}$ (resp. its backward closure $\overleftarrow{A}^{\{x\}}$) can be defined by periodic clock constraints (the inverse image operator involved in (5) is actually a direct product of free monoids and therefore the inverse image is synchronized). The result follows from the Theorem 3.

■

Observe that, contrarily to the case of aperiodic constraints [9], Example 3.1 shows that the removal of ϵ-moves with clock resets not lying on a loop is not always possible.

5 Count down clocks

5.1 A closure property

References to the future in the timed automata model exist in the literature. In [2], the authors aim at defining a class of timed languages on finite sequences that is closed with respect to the Boolean operations: the distance between the present time and the next occurrence of each input letter is "predicted". In [4] a different approach is proposed by allowing the input timed sequences to be scanned back and forth. Here, we introduce count-down clocks, i. e., clocks that decrease though running at the same speed as normal clocks. Also, each clock constraint is the disjunct of two constraints that apply to the "count-up" and the "count-down" clocks separately. Here is a formal definition where $\Phi(\overrightarrow{X})$ and $\Phi(\overleftarrow{X})$ are two disjoint families of periodic constraints.

DEFINITION **4** *A* timed (Büchi) automaton with count-down clocks *is a tuple* $\mathcal{A} = (\Sigma, Q, I, F, T, R, \overleftarrow{X}, \overrightarrow{X})$ *where* Σ, Q, I, F, R *are as in Definition 2,* $\overleftarrow{X}$ *is a finite set of real valued* count-up clocks, $\overrightarrow{X}$ *is a finite set of real valued* count-down clocks, $\overleftarrow{X} \cap \overrightarrow{X} = \emptyset$ *and* $T \subseteq Q \times \Phi(\overleftarrow{X}) \times \Phi(\overrightarrow{X}) \times (\Sigma \cup \{\epsilon\}) \times 2^{\overleftarrow{X}} \times 2^{\overrightarrow{X}} \times Q$ *is a finite set of* transitions.

The meaning of the transitions is a direct extension of the standard case. An *assignment* of the clocks is a function of $\nu : \overleftarrow{X} \cup \overrightarrow{X} \to \mathbb{R}_+$. A finite (resp. infinite) timed word $(\sigma_1, t_1)(\sigma_2, t_2)\ldots(\sigma_n, t_n)$ (resp. $(\sigma_1, t_1)(\sigma_2, t_2)\ldots(\sigma_i, t_i), \ldots$) is accepted by $\mathcal{A}$ if there exist

– a sequence of assignments ν_i, $i \geq 0$, where $\nu_0(x) = 0$ for all $x \in \overleftarrow{X}$ while $\nu_0(x)$ is an arbitrary value in $\mathbb{R}_+$ for every $x \in \overrightarrow{X}$,

– a sequence of states q_i, $i \geq 0$, with $q_0 \in I$, ending in $q_n \in F$ (resp. $q_i \in R$ for infinitely many i's),

– a sequence of transitions $(q_{i-1}, \phi_i, \sigma_i, \overleftarrow{X}_i, \overrightarrow{X}_i, q_i)$, $i = 1, 2, \ldots$,

such that the following properties hold for every $i = 1, 2, \ldots$:

– for all $x \in \overleftarrow{X}$

$\nu_{i-1}(x) + t_i - t_{i-1}$ satisfies ϕ_i (with the initial condition $t_0 = 0$),

$$\text{and } \nu_i(x) = \begin{cases} 0 & \text{if } x \in \overleftarrow{X}_i \\ \nu_{i-1}(x) + t_i - t_{i-1} & \text{otherwise;} \end{cases}$$

– for all $x \in \overrightarrow{X}$

$\nu_i(x)$ satisfies ϕ_i,

$\nu_{i-1}(x) = t_i - t_{i-1}$ if $x \in \overrightarrow{X}_i$,

$$\text{and } \nu_i(x) = \begin{cases} \text{arbitrary} & \text{if } x \in \overrightarrow{X}_i \\ \nu_{i-1}(x) - (t_i - t_{i-1}) & \text{otherwise.} \end{cases}$$

THEOREM **5** *The family of timed languages recognized by timed automata with count down clocks and no ϵ-moves (resp. and possible ϵ-moves) equals the family of timed languages recognized by timed automata with no count down clocks and no ϵ-moves (resp. and possible ϵ-moves).*

Proof. In order to simplify notations, we assume there exists only one count down clock (the proof can be easily extended to the general case by induction).

Let $\mathcal{A}$ be a timed automaton, let z be its count-down clock and let X be the set of ordinary count-up clocks. As usual, it is possible to assume that there exists an integer m such that all z-constraints are of the form $z = i$, $z \in (i, i+1)$, $z \in T_m[i]$ or $z \in T_m(i, i+1)$, where $0 \leq i < m$ and $T_m[i]$, $T_m(i, i+1)$ are defined as in Theorem 1. We omit the cases where the clock z satisfies an exact condition of the form $z = i$ or $z \in T_m[i]$, and consider only the z-constraints

$$\phi_i(z) \equiv z \in (i, i+1),$$
$$\phi_{i+m}(z) \equiv z \in T_m(i, i+1).$$

Then, all clock constraints of $\mathcal{A}$ are of the form $\phi \equiv \psi(X)$ or $\phi \equiv \psi(X) \wedge \phi_i(z)$ for some $0 \leq i < 2m$. We call *i-transition* a transition that bears a condition of the form $\psi(X) \wedge \phi_i(z)$.

Let us first explain intuitively how we proceed. Consider the traversal of the r-th transition θ_r by some path in the automaton. Assume $z \in \overrightarrow{X}$ occurs in the penultimate component of the previous transition θ_{r-1} and let θ_s, $(r \leq s)$ be the next transition where $z \in \overrightarrow{X}$ occurs. The idea is to collect information about the traversal times of the different transitions between θ_r and θ_s and to postpone the verification $z = 0$ until θ_s is reached. To this purpose, for all integers $0 \leq i < 2m$ for which there exists an i-transition between θ_r and θ_s we record

– the earliest and latest time e_i, l_i of a traversal of an i-transition associated with a condition $z \in (i, i+1)$ for some $0 \leq i < m$;

– the first time f_i and the latest and earliest "modulo m" time l_i, e_i of a traversal of an i-transition for some $m \leq i < 2m$. By latest time "modulo m" we mean the traversal time l_i of an i-transition such that $0 \leq l_i - f_i \leq \frac{m}{2}$ mod m and for all traversal times l of an i-transition such that $0 \leq l - f_i \leq \frac{m}{2}$ mod m, the inequalities $0 \leq l - f_i \leq l_i - f_i$ mod m hold. Equally, by earliest time "modulo m" we mean the traversal time e_i of an i-transition such that $-\frac{m}{2} < e_i - f_i \leq 0$ mod m and for all traversal times e of an i-transition such that $-\frac{m}{2} < e - f_i \leq 0$ mod m, the inequalities $-\frac{m}{2} < e_i - f_i \leq e - f_i \leq 0$ mod m hold.

Then, in order to simulate the clock z, we introduce a new set of ordinary clocks denoted by e_i, l_i (for $i = 0, 1, \ldots, 2m-1$) and f_i (for $i = m, \ldots, 2m-1$). During the run between transition θ_r and θ_s these clocks are reset according with the description above. When θ_s is traversed, it suffices to verify that $e_i, l_i \in (i, i+1)$ holds for all $0 \leq i < m$ and that $e_i, l_i, f_i \in T_m(i-m, i+1-m)$ holds for all $m \leq i < 2m$.

We now define more rigorously the construction of an equivalent automaton with ordinary clocks only. Its states are the pairs (q, I), where q is a state of $\mathcal{A}$ and I is a subset of $\{0, \ldots, 2m-1\}$ used to record the possible i-transitions encountered. Each run is now split in two parts: in the first one the informations about i-transitions are collected, in the second step we perform the test $z = 0$.

Collecting step

If $0 \leq i < m$ then every transition $q \xrightarrow{\psi \wedge z \in (i,i+1);\sigma;X'} p$ with $z \notin X'$ gives rise to the following transitions

$$(q, I) \xrightarrow{\psi;\sigma;X''} (p, I \cup \{i\})$$

where $I \subseteq \{0, \ldots, 2m-1\}$ and $X'' = \begin{cases} X' \cup \{l_i\} & \text{if } i \in I \\ X' \cup \{e_i, l_i\} & \text{if } i \notin I. \end{cases}$

If $m \leq i < 2m$ then every transition $q \xrightarrow{\psi \wedge z \in T_m(i-m,i+1-m);\sigma;X'} p$ with $z \notin X'$ gives rise to the transitions

$$(q, I) \xrightarrow{\psi;\sigma;X' \cup \{f_i, e_i, l_i\}} (p, I \cup \{i\})$$

for all $I \subseteq \{0, \ldots, 2m-1\}$ such that $i \notin I$, and to the transitions

$$(q, I) \xrightarrow{\psi \wedge f_i \in T_m(0,\frac{m}{2}] \wedge l_i \in T_m(0,\frac{m}{2}];\sigma;X' \cup \{l_i\}} (p, I)$$

and

$$(q, I) \xrightarrow{\psi \wedge f_i \in T_m(-\frac{m}{2},0) \wedge e_i \in T_m(-\frac{m}{2},0);\sigma;X'\cup\{e_i\}} (p, I)$$

for all $I \subseteq \{0, \ldots, 2m-1\}$ such that $i \in I$.

Testing step

Every transition $q \xrightarrow{\psi\wedge\phi;\sigma;X\cup\{z\}} p$ with $\phi \equiv z \in (i, i+1)$ (resp. $\phi \equiv z \in T_m(i, i+1)$) gives rise to the transitions

$$(q, I) \xrightarrow{\psi\wedge\alpha;\sigma;X'} (p, \{i\}) \text{ (resp. } (q, I) \xrightarrow{\psi\wedge\alpha;\sigma;X'} (p, \{i+m\}) \text{)}$$

where

$$\alpha \equiv \bigwedge_{j \in I \cap \{0,\ldots,m-1\}} e_j, l_j \in (j, j+1) \quad \bigwedge_{j \in I \cap \{m,\ldots,2m-1\}} f_j, e_j, l_j \in T_m(j-m, j+1-m)$$

and $X' = X \cup \bigcup_{0 \leq j \leq 2m-1} \{f_j, e_j, l_j\}$.

In a similar way we modify the transitions of the form $q \xrightarrow{\psi;\sigma;X\cup\{z\}} p$ where ψ is a constraint on ordinary clocks.

5.2 Relative modulo distance theory

Future operations were also introduced in temporal logics such as in $\mathrm{TL}_\Gamma(P)$ (cf. [18]), MITL_P (cf. [3]) and EMITL_P (cf. [19]). In [5] the monadic logic of distance was introduced by enriching the usual "Büchi-like" signature ($<$ and for all symbols $a \in \Sigma$ the first order predicate Q_a) with a binary first order *distance predicate* $d(.,.) \sim c$ where $\sim \in \{<, \leq, =, \neq, >, \geq\}$. The authors proved that the theory of timed sequences in the first order fragment is undecidable.

In [19] the predicates $d(.,.) \sim c$ are replaced by two types of *relative distance* predicates $\overleftarrow{d(X,x)} \sim c$ and $\overrightarrow{d(x,X)} \sim c$, where $\sim \in \{<, =, >\}$, $c \in \mathbb{N}$, x is an individual variable and X is a subset variable. The predicate $\overleftarrow{d(X,x)} \sim c$ (resp. $\overrightarrow{d(x,X)} \sim c$) is equivalent to $d(y,x) \sim c$ (resp. $d(x,y) \sim c$), where y is the greatest element of X less than x (resp. the smallest element of X greater than x). Given a finite set P of propositions, the signature Sign(P) of the language $\mathcal{L}\overleftrightarrow{d}$(P) of *monadic logic of relative distance over P* comprises the symbol $<$, a unary predicate Q_p for each $p \in P$ and the predicates $\overleftarrow{d(X,x)} \sim c$ and $\overrightarrow{d(X,x)} \sim c$ as above. The formulae of the language are of the form $\exists X_0 \ldots X_{m-1}\psi$, where ψ is built from the atomic formulae $Q_p x$, $x < y$, Xx (meaning $x \in X$), $\overleftarrow{d(X_i,x)} \sim c$, $\overrightarrow{d(x,X_i)} \sim c$, with $i < m$ by using Boolean connectives, quantification of individual and set variables except for $X_0 \ldots X_{m-1}$.

The language $\mathcal{L}\overleftrightarrow{md}(\Sigma)$ of the *monadic logic of relative modulo distance* over Σ is obtained by dentifying the power set of P with Σ and by substituting the *relative periodic distance* predicates for the relative distance predicates in the previous definition. More precisely, these predicates are $\overleftarrow{d_\rho(X,x)}$ and $\overrightarrow{d_\rho(x,X)}$, where ρ is a quintuple $(a, b, \lambda, \prec_1, \prec_2)$ consisting of three rational numbers a, b, λ and two relation symbols $\prec_1, \prec_2 \in \{<, \leq, \}$. The predicate $\overleftarrow{d_\rho(X,x)}$ (resp. $\overrightarrow{d_\rho(x,X)}$) is equivalent to $\exists k > 0, a + k\lambda \prec_1 d(y,x) \prec_2 b + k\lambda$, where y is the greatest element of X less than x (resp. $\exists k > 0, a+k\lambda \prec_1 d(x,y) \prec_2 b+k\lambda$, where y is the smallest element of X greater than x).

The proof of expressive completeness of [19] carries over to our logic and we get the following

THEOREM **6** *A timed language over Σ is recognized by some timed automaton with periodic constraints if and only if it is defined by some $\mathcal{L}\overleftrightarrow{md}(\Sigma)$-sentence.*

6 Special cases

In this section we study special cases concerning the reset of clocks on one side and the discrete time on the other.

6.1 Reset-free and pure reset automata

By reset-free we understand that no transition performs a reset and by pure reset we mean that all transitions reset its clocks. We show that inclusion (and therefore the equivalence of automata) is decidable under either hypothesis. Observe that these conditions imply that the automaton uses one single clock.

PROPOSITION **3** *The inclusion problem of two timed languages defined by reset-free timed automata with periodic constraints and no ϵ-transitions is decidable.*

Proof. We assume there exists one clock and by modifying slightly conditions (1), we may suppose that all constraints are of the form: i) $x = i$, ii) $x \in (i, i+1)$, iii) $x \in T_m[i+m]$, iv) $x \in T_m(i+m, i+1+m)$, where $0 \leq i < m$; note that these constraints are disjoint. Our purpose is to construct from the timed automaton, an ordinary Büchi automaton $\mathcal{A}'$ (on infinite strings) conveying the same information.

With every letter $\sigma \in \Sigma$ we associate the symbols $\sigma_{i,j}$, $\sigma_{\hat{i},j}$, $\sigma_{i,\hat{j}}$ and $\sigma_{\hat{i},\hat{j}}$, where $0 \leq i, j < 2m$. The idea is to replace in the infinite sequence $(\sigma^{\{1\}}, t_1)(\sigma^{\{2\}}, t_2) \ldots (\sigma^{\{\ell-1\}}, t_{\ell-1})$ $(\sigma^{\{\ell\}}, t_\ell) \ldots$, the ℓ-th occurrence $(\sigma^{\{\ell\}}, t_\ell)$ by one of the previous symbols according to the values of $t_{\ell-1}$ and t_ℓ. This is illustrated by the following table where the rows are labelled by the value of the clock before the transition and the columns by its value after the transition.

	j	$(j, j+1)$	$T_m[j+m]$	$T_m(j+m, j+1+m)$
i	$\sigma_{\hat{i},\hat{j}}$	$\sigma_{\hat{i},j}$	$\sigma_{\hat{i},\widehat{j+m}}$	$\sigma_{\hat{i},j+m}$
$(i, i+1)$	$\sigma_{i,\hat{j}}$	$\sigma_{i,j}$	$\sigma_{i,\widehat{j+m}}$	$\sigma_{i,j+m}$
$T_m[i+m]$	$\sigma_{\widehat{i+m},\hat{j}}$	$\sigma_{\widehat{i+m},j}$	$\sigma_{\widehat{i+m},\widehat{j+m}}$	$\sigma_{\widehat{i+m},j+m}$
$T_m(i+m, i+1+m)$	$\sigma_{i+m,\hat{j}}$	$\sigma_{i+m,j}$	$\sigma_{i+m,\widehat{j+m}}$	$\sigma_{i+m,j+m}$

Table 1: Meaning of the symbols $\sigma_{i,j}$, etc...

With every transition $q \xrightarrow{A,\sigma} p$ of $\mathcal{A}$, we associate the following set of transitions of $\mathcal{A}'$: if $A \equiv x = j$ then for all $i < j$

$$(q,B) \xrightarrow{\sigma_{\hat{i},\hat{j}}} (p,A) \text{ for all } B \equiv x = i$$
and
$$(q,B) \xrightarrow{\sigma_{i,\hat{j}}} (p,A) \text{ for all } B \equiv x \in (i, i+1);$$

if $A \equiv x \in (j, j+1)$ then for all $i \leq j$

$$(q,B) \xrightarrow{\sigma_{\hat{i},j}} (p,A) \text{ for all } B \equiv x = i$$
and
$$(q,B) \xrightarrow{\sigma_{i,j}} (p,A) \text{ for all } B \equiv x \in (i, i+1);$$

if $A \equiv x \in T_m[j+m]$ then for all $0 \leq i < m$

$$\begin{aligned}
(q,B) &\xrightarrow{\sigma_{i,\widehat{m+j}}} (p,A) \text{ if } B \equiv x \in (i, i+1)\\
(q,B) &\xrightarrow{\sigma_{\hat{i},\widehat{m+j}}} (p,A) \text{ if } B \equiv x = i\\
(q,B) &\xrightarrow{\sigma_{\widehat{m+i},\widehat{m+j}}} (p,A) \text{ if } B \equiv x \in T_m[i+m]\\
(q,B) &\xrightarrow{\sigma_{m+i,\widehat{m+j}}} (p,A) \text{ if } B \equiv x \in T_m(i+m, i+m+1);
\end{aligned}$$

if $A \equiv x \in T_m(j+m, j+m+1)$ then for all $0 \leq i < m$

$$\begin{aligned}
(q,B) &\xrightarrow{\sigma_{i,m+j}} (p,A) \text{ if } B \equiv x \in (i, i+1)\\
(q,B) &\xrightarrow{\sigma_{\hat{i},m+j}} (p,A) \text{ if } B \equiv x = i\\
(q,B) &\xrightarrow{\sigma_{m+i,m+j}} (p,A) \text{ if } B \equiv x \in T_m[i+m]\\
(q,B) &\xrightarrow{\sigma_{\widehat{m+i},m+j}} (p,A) \text{ if } B \equiv x \in T_m(i+m, i+m+1).
\end{aligned}$$

The Büchi automaton $\mathcal{A}'$ carries all the information about $\mathcal{A}$. Indeed, let Δ be the set of all the $\sigma_{i,j}$'s, $\sigma_{\hat{i},j}$'s, $\sigma_{i,\hat{j}}$'s and $\sigma_{\hat{i},\hat{j}}$'s as introduced above. Define a mapping $f : (\Sigma \times \mathbb{R})^\omega \longrightarrow \Delta^\omega$ by replacing in each timed sequence

$$(\sigma^{\{1\}}, t_1)(\sigma^{\{2\}}, t_2) \ldots (\sigma^{\{\ell-1\}}, t_{\ell-1})(\sigma^{\{\ell\}}, t_\ell) \ldots$$

the ℓ-th element $(\sigma^{\{\ell\}}, t_\ell)$ by a symbol in Δ according to table 1. Then given two timed sequences $w, w' \in (\Sigma \times \mathbb{R}_+)^\omega$ we have: if w is recognized by $\mathcal{A}$ and if $f(w) = f(w')$ then w' is recognized by $\mathcal{A}$.

Now given two reset-free timed automata $\mathcal{A}$ and $\mathcal{B}$, let m be a common integer for which all constraints of $\mathcal{A}$ and $\mathcal{B}$ are of the form (1). Then construct as above the two Büchi automata $\mathcal{A}'$ and $\mathcal{B}'$: $\mathcal{A}$ is included in $\mathcal{B}$ if and only if $\mathcal{A}'$ is included in $\mathcal{B}'$. ■

Proposition 4 *The inclusion problem of two timed languages defined by pure reset timed automata with periodic constraints and no ϵ-transitions is decidable.*

Proof. We proceed in the same vein as in Proposition 3. In particular the constraints of the automata are assumed to be of the same form as in the previous proof. We associate with every letter $\sigma \in \Sigma$ the symbols σ_i and $\sigma_{\hat{i}}$ for $0 \leq i < 2m$. In the infinite sequence $(\sigma^{\{1\}}, t_1)(\sigma^{\{2\}}, t_2) \ldots (\sigma^{\{\ell-1\}}, t_{\ell-1})(\sigma^{\{\ell\}}, t_\ell) \ldots$, we replace the ℓ-th occurrence $(\sigma^{\{\ell\}}, t_\ell)$ by $\sigma_{\hat{i}}$ with $0 \leq i < m$ or σ_i with $0 \leq i < m$ or $\sigma_{\hat{i}}$ with $m \leq i < 2m$ or σ_i with $m \leq i < 2m$), according as to whether $t_\ell - t_{\ell-1}$ satisfies condition i), ii), iii) or iv).

■

6.2 The discrete time

The results presented in this paragraph are extensions of some results in [5] and [16]. The novelty is that they consider periodic constraints on one hand and ϵ-moves on the other.

PROPOSITION 5 *Whenever* $\mathbb{T} = \mathbb{N}$*, each timed automaton with periodic clock contraints and possible* ϵ*-transitions is equivalent to a pure reset timed automaton with periodic clock contraints and no* ϵ*-transition.*

Proof. We will first show that the original automaton $\mathcal{A}$ can be transformed into an automaton $\mathcal{A}'$ with one single clock x that is reset after each transition. Number the clocks from 1 to n and assume further that the constraint associated with a transition is of the form $\bigwedge_{1\leq i\leq n} \psi_i(x_i)$ where each condition ψ_i defines a rational subset $[\![\psi_i]\!]$ of $\mathbb{N}$. Consider the intersection $\sim$ of the right invariant equivalences associated with all the $[\![\psi_i]\!]$'s in all transitions. It exists and has finite index. Denote by $[m]$ the class of the integer m relative to $\sim$, by $[m]^{-1}[p]$ the set of integers $r > 0$ such that $m + r \sim p$.

Consider a transition $q \xrightarrow{\phi;\sigma;Y} p$ with $\sigma \in \Sigma \cup \{\epsilon\}$ and $\phi \equiv \bigwedge_{1\leq i\leq n} \psi_i(x_i)$. Then the transition gives rise to the following transitions of $\mathcal{A}'$ with $a_1, \ldots, a_n,\ b_1, \ldots, b_n \in \mathbb{N}$ and $[b_i] \subseteq [\![\psi_i]\!]$ for all i:

$$(q, [a_1], \ldots, [a_n]) \xrightarrow{\theta(x);\sigma;\{x\}} (p, \widehat{[b_1]}, \ldots, \widehat{[b_n]})$$

where

$$[\![\theta]\!] = \bigcap_{1\leq i\leq n} [a_i]^{-1}[b_i] \text{ and, for } i = 1, \ldots, n, \quad \widehat{[b_i]} = \begin{cases} [0] & \text{if } x_i \in Y \\ [b_i] & \text{otherwise} \end{cases}$$

At this point, we may assume the automaton is pure reset and we prove that its ϵ-moves may be eliminated. An ϵ-path is a path taking ϵ-transitions only. The idea is to view an infinite path as an infinite sequence of ϵ-paths followed by a σ-transition where $\sigma \in \Sigma$. This means that we group together the ϵ-transitions preceding a given occurrence of a letter σ and that we add the duration of the ϵ-path to the duration of the σ-transition. More formally, let $q, p \in Q$ be two states. We distinguish the ϵ-paths according to whether or not they visit some repeated state. The set of all paths taking q to p and not visiting any repeated state is rational over the alphabet of the transitions of the automaton. Now the set of durations of these paths is the image of this set by a rational substitution which assigns the clock constraint with each transition. Denote this last rational subset of $\mathbb{N}$ by $A_{q,p}$. Similarly, the subset $B_{q,p}$ of all durations of paths labelled by ϵ and taking q to p and visiting some repeated state is a rational subset of $\mathbb{N}$.

Now we modify the transitions of the automaton and we add a component to the state in order to remember whether or not the ϵ-path preceding a given occurrence $\sigma \in \Sigma$ has visited a repeated state. More specifically, the new set of states is $Q \times \{0, 1\}$ and the set of repeated states is $Q \times \{1\}$. We denote by $\phi^{\sigma}_{q,p}$ the time constraint associated with the σ-transition taking q to p. We define

$$(q, i) \xrightarrow{\psi;\sigma;\{x\}} (p, 0) \text{ where } [\![\psi]\!] = \bigcup_{r\in Q} (A_{q,r} + [\![\phi^{\sigma}_{r,p}]\!])$$

and

$$(q,i) \xrightarrow{\psi;\sigma;\{x\}} (p,1) \text{ where } [\![\psi]\!] = \bigcup_{r\in Q} (B_{q,r} + [\![\phi^{\sigma}_{r,p}]\!])$$

(here the sum of two sets is to be understood as the set of all possible sums of their elements). ∎

Since all transitions reset the clock, the same technique as in the previous pure-reset case applies and we can establish the following.

Corollary **6** *Whenever $\mathbb{T} = \mathbb{N}$, the inclusion (and therefore the equality) of two timed languages defined by timed automata with periodic clock contraints and possible ϵ-transitions is decidable.*

We recall that the logic $\mathcal{L}^2_T$ introduced in [5, p. 39], is an extension of the second order monadic logic of one successor enriched with predicates to accomodate time in $\mathbb{N}$. These predicates are $x \leq y$ and $x \equiv_i y$ the latter being the usual "modulo i" operation. Here is therefore a logical characterization of the class of timed languages recognized by some automaton with periodic constraints and possible ϵ-transitions (the result without ϵ-transitions is Theorem 1, p. 44, [5]).

Corollary **7** *Assume $\mathbb{T} = \mathbb{N}$. Then a timed language is recognized by some automaton with periodic constraints and possible ϵ-transitions, if and only if it is the set of models of a $\mathcal{L}^2_T$-formula.*

7 Appendix

In this section we give the proof of Theorem 1. To this purpose, for each $\lambda = \{\lambda_i\}_{i=1,\ldots,n} \in \mathbb{R}^n$ and each subset $H \subseteq \mathbb{R}^n_+$, we define

$$T_\lambda(H) = \{(k_1\lambda_1 + x_1, \ldots, k_n\lambda_n + x_n) \mid k_1, \ldots, k_n \in \mathbb{N} \text{ and } (x_1, \ldots, x_n) \in H\}$$

We start with an easy property.

Proposition **8** *Consider the subsets of the form $T_\lambda(H)$, where $\lambda \in \mathbb{Q}^n_+$, $H = H_1 \times \ldots \times H_n$, and H_i is a rational interval for each $i = 1, 2, \ldots, n$. Then, the family of the finite unions of these sets forms a Boolean algebra.*

Proof. By a pure set-theoretic properties, we can restrict ourselves to the case $n = 1$. We first prove the closure under intersection. Since intersection distributes over union, it suffices to consider the expression $T_\lambda(H) \cap T_\mu(K)$: multiplying by the least common multiple of the denominators of the different constants, we may assume without loss of generality that all coefficients are integers. Let ν be the least common multiple of λ and μ. Then, there exist $p \in \mathbb{N}$ and four finite unions of integer intervals U, V, H', K' such that $U \cup V \subseteq \{x \mid x < p\}$ and $H' \cup K' \subseteq \{x \mid x \geq p\}$ and

$$T_\lambda(H) = U \cup T_\nu(H') \text{ and } T_\lambda(K) = V \cup T_\nu(K')$$

thus

$$T_\lambda(H) \cap T_\mu(K) = (U \cap V) \cup T_\nu(H' \cap K')$$

Concerning the complement, it suffices to deal with the sets $T_\lambda(H)$ such that $H = [a, b]$ and $a, b, \lambda \in \mathbb{Q}_+$ (the other cases can be treated similarly). Then, the complement satisfies the equality

$$\overline{T_\lambda(H)} = [0, a) \cup T_\lambda((b, a + \lambda))$$

which completes the proof. ■

Let us say that the clock constraint $\phi(x_1, \ldots, x_n)$ *defines* the subset $[\![\phi]\!] \subseteq \mathbb{R}_+^n$, if $[\![\phi]\!]$ is the set of all $(a_1, \ldots, a_n) \in \mathbb{R}_+^n$ such that ϕ is true whenever each x_i is substituted for a_i in ϕ. The following proposition characterizes the subsets defined by non-diagonal clock constraints.

PROPOSITION **9** *A non-diagonal clock constraint in $\Phi(X)$ defines a subset $K \subseteq \mathbb{R}_+^n$ if and only if K is a finite union of subsets of the form $T_\lambda(H)$, with $\lambda \in \mathbb{Q}_+^n$ and $H = H_1 \times \ldots \times H_n$, where H_i is a rational interval for each $i = 1, 2, \ldots, n$.*

Proof. Indeed, in one direction the proof is easy because a subset of the form $T_\lambda(H)$ as above is clearly defined by a non-diagonal clock constraint. Conversely, the atoms of $\Phi(X)$ define subsets of the given type $T_\lambda(H)$ and, since the finite unions of such sets form a Boolean algebra as shown in Proposition 8, the claim is proved. ■

The following property further simplifies the form of the periodic constraints of a timed automaton. Its proof mimics that of the traditional clock constraints and is therefore folklore.

PROPOSITION **10** *For each timed automaton $\mathcal{A}$ with clock constraints $\Phi(X)$ there exists an equivalent timed automaton $\mathcal{A}'$ using non-diagonal atoms of $\Phi(X)$ and the logical connectives.*

Now, we are able to give the proof of the theorem. In view of the previous proposition, we consider a timed automaton with non-diagonal clock constraints. By Proposition 9 and via possible further decomposition, we may assume that each constraint defines a subset of $\mathbb{R}^n$ of the form

$$T_\lambda(H) \tag{6}$$

where $\lambda \in \mathbb{N}^n$, $H = H_1 \times \ldots \times H_n$ and $H_i = [a_i]$ or $H_i = (a_i, a_i + 1)$, with $a_i \in \mathbb{N}$ for each $i = 1, 2, \ldots, n$.

Let p be a multiple of all components λ_i of the λ's appearing in the constraints. Consider a costraint $T_\lambda(H)$ defined as above and let $I \subseteq \{1, \ldots, n\}$ be the subset of indices i for which $\lambda_i \neq 0$. Moreover, denote by Λ_I the vector satisfying $\Lambda_I(i) = p$ if $i \in I$ and $\Lambda_I(i) = 0$ otherwise. Then, $T_\lambda(H)$ is the union of the subsets

$$(\mu_1\lambda_1, \ldots, \mu_n\lambda_n) + T_{\Lambda_I}(H) = T_{\Lambda_I}((\mu_1\lambda_1, \ldots, \mu_n\lambda_n) + H)$$

where the integer μ_i equals 0 if $\lambda_i = 0$ and $0 \leq \mu_i < \frac{p}{\lambda_i}$ otherwise.

At this point all constraints are of the form (6) where all non-zero components of λ are equal to p. Let $m = kp$ be a multiple of p greater than all a_i's. Then $T_\lambda(H_1 \times \ldots \times H_n)$ is a finite union of subsets of the form $T_{k\lambda}(K_1 \times \ldots \times K_n)$ with $K_i = \{j\}$ or $K_i = (j, j + 1)$ for $0 \leq j < m$, which is precisely the claim. Observe that all decompositions performed in the construction are disjoint. Hence, if the original automaton is deterministic, so is the final one.

References

[1] R. Alur and D.L. Dill. A theory of timed automata. *Theoret. Comput. Sci.*, 126:183–235, 1994.

[2] R. Alur, L. Fix, and T.A. Henzinger. A determinizable class of timed automata. In *Proceedings of CAV'94*, number 818 in Lecture Notes in Computer Science, pages 1–13. Springer Verlag, 1994.

[3] R. Alur and T.A. Henzinger. Real-time logics: complexity and expressiveness. In *Fifth Annual IEEE Symposium on Logic in Computer Science*, pages 390–401, Philadelphia, Pennsylvania, 1990. IEEE Computer Science Press.

[4] R. Alur and T.A. Henzinger. Back to the future: towards a theory of timed regular languages. In *Proceedings of FOCS'92*, pages 177–186, 1992.

[5] R. Alur and T.A. Henzinger. Real-time logics: Complexity and Expressiveness. *Inform. and Comput.*, 104:35–77, 1993.

[6] B. Bérard, P. Gastin, and A. Petit. Timed automata with non observable actions: expressive power and refinement. Technical report, LITP 97/23, Université Paris 7, 1997.

[7] B. Bérard and C. Picaronny. Accepting Zeno words without stopping time. In *Proceedings of MFCS'97*, number 1295 in Lecture Notes in Computer Science, pages 149–158. Springer Verlag, 1997.

[8] F. Demichelis and W. Zielonka. Control timed automata. Technical report, LaBRI, Université Bordeaux 1, 1998.

[9] V. Diekert, P. Gastin, and A. Petit. Removing ϵ-transitions in timed automata. In *Proceedings of STACS'97*, number 1200 in Lecture Notes in Computer Science, pages 583–594. Springer Verlag, 1997.

[10] S. Eilenberg. *Automata, Languages and Machines*, volume A. Academic Press, 1974.

[11] S. Eilenberg, C.C. Elgot, and J.C. Shepherdson. Sets recognized by n-tape automata. *Journal of Algebra*, 3:447–464, 1969.

[12] C. Frougny and J. Sakarovitch. Synchronized rational relations of finite and infinite words. *Theoret. Comput. Sci.*, 108:45–82, 1993.

[13] S. Ginsburg and E. Spanier. Presburger formulas and languages. *Pacific Journal of Mathematics*, 16:285–296, 1966.

[14] V. Gupta, T. A. Henzinger, and R. Jagadeesan. Robust timed automata. In *Proceedings of HART'97*, number 1201 in Lecture Notes in Computer Science, pages 331–345. Springer Verlag, 1997.

[15] T.A. Henzinger. The theory of hybrid automata. In *Proceedings of LICS'96*, pages 278 –292, New Brunswick, New Jersey, 1996. Invited tutorial.

[16] T.A. Henzinger, P.W. Kopke, and H Wong-Toi. The expressive power of clocks. In *Proceedings of ICALP'95*, number 944 in Lecture Notes in Computer Science, pages 335–346. Springer Verlag, 1995.

[17] P. Hermann. Automates temporisés et reconnaissabilité. *Inform. Process. Lett.*, 65:313–318, 1998.

[18] Z. Manna and A. Pnueli. Models for reactivity. *Acta Informatica*, 30(2):609–678, 1993.

[19] T. Wilke. Specifying timed state sequences in powerful decidable logics and timed automata. In H. Langmaack, W.-P. de Roever, and J. Vytopil, editors, *Formal Techniques in Real-Time and Fault-Tolerant Systems*, volume 863 of *Lecture Notes in Computer Science*. Springer Verlag, 1994.