An Automata-Theoretic Approach to Constraint LTL

Stéphane Demri[1] and Deepak D'Souza[⋆2]

[1] Lab. Spécification et Vérification, ENS de Cachan & CNRS UMR 8643
61, av. Pdt. Wilson, 94235 Cachan Cedex, France.
email: demri@lsv.ens-cachan.fr
[2] Chennai Mathematical Institute
92 G. N. Chetty Road, Chennai 600 017, India.
email: deepak@cmi.ac.in

Abstract. We consider an extension of linear-time temporal logic (LTL) with constraints interpreted over a concrete domain. We use a new automata-theoretic technique to show PSPACE decidability of the logic for the constraint systems $(\mathbb{Z}, <, =)$ and $(\mathbb{N}, <, =)$. We also give an automata-theoretic proof of a result of Balbiani and Condotta [BC02] for constraint systems that satisfy a "completion" property.

1 Introduction

We consider in this paper an extension of Linear-time Temporal Logic (LTL) [Pnu77] with atomic assertions from a constraint system $\mathcal{D}$. The logic is denoted CLTL($\mathcal{D}$), for Constraint LTL parameterised by the constraint system $\mathcal{D}$, and is obtained from LTL by replacing propositions by atomic constraints in $\mathcal{D}$. In classical LTL variables represent propositions and models are sequences of propositional valuations for these variables. These models can be viewed as having a "spatial" axis (here the elements $\top$ and $\perp$), along which the variables move. In CLTL($\mathcal{D}$), the spatial axis for the models will comprise elements from the domain of $\mathcal{D}$. For example, in CLTL$(\mathbb{N}, <, =)$ one is allowed to use atomic constraints involving $<$ and $=$, and variables ranging over natural numbers. The formula $\Box(x < y)$ in CLTL$(\mathbb{N}, <, =)$ is interpreted over a sequence of $\mathbb{N}$-valuations for the variables x and y, and asserts that at every point in the future, the value of the variable x is less than the value of y. This formula is of course satisfiable, and a candidate satisfying model is $ss\cdots$, where s assigns 1 to x and 2 to y.

Constraint temporal logics have been introduced and studied by logicians in the field of knowledge representation [WZ02]. Spatio-temporal logics, as they are better known there, involve a hybrid of temporal logic and constraint systems, with varying degrees of interaction. For instance, one may be permitted to refer to the value of a variable x in the *next* time instant, leading to constraints of the form $(x > Ox)$. More generously, one may permit $(x > \Diamond y)$, which asserts that the current value of x exceeds the value of *some* future value of y.

⋆ Part of this work was done during this author's visit to LSV, ENS-Cachan.

While research in the area has focused mainly on logics involving constraint systems that have as domains intervals [All83,BC02] and regions [RCC92,WZ00], with a variety of decidability and complexity results, there has been little progress with commonly used constraint systems of the form $(D, <, =)$, with D as the integers $\mathbb{Z}$, or the natural numbers $\mathbb{N}$. Comon and Cortier [CC00] consider a constraint system with the reals $\mathbb{R}$ as the underlying domain and constraints of the form $x < Oy + 2$, which they show is undecidable. They identify a decidable fragment of the logic by restricting the use of the "Until" operator. Balbiani and Condotta [BC02] prove a general decidability result for constraint systems that satisfy a "completion" property – essentially that, given a finite set of consistent constraints S, a valuation for a subset of variables which satisfies the constraints involving only those variables, can be extended to a valuation which satisfies S. The constraint system $(\mathbb{R}, <, =)$ satisfies this property, and the decidability of CLTL$(\mathbb{R}, <, =)$ in PSPACE follows from this result.

In this paper we focus mainly on the constraint systems $(\mathbb{Z},<,=)$ and $(\mathbb{N},<,=)$. As in [BC02], we restrict the interaction between the temporal operators and variables to terms of the form $OO \cdots Ox$. Our approach is automata-theoretic in nature, and we hope that the benefits of such an approach will be evident to the reader. In particular, it facilitates a transparent argument for the result of [BC02] for constraint systems satisfying the completion property. For such constraint systems $\mathcal{D}$ one can further show that the set $L^{\mathcal{D}}(\varphi)$, of models which satisfy a given formula φ, is ω-regular – i.e. it can be described by an automaton on infinite words. This parallels the results for classical LTL [VW86].

The constraint systems $(\mathbb{Z}, <, =)$ and $(\mathbb{N}, <, =)$ are more interesting from a technical point of view. As a typical illustration, the formula $\Box(x > Ox)$ has no (infinite) $\mathbb{N}$-models, while it does have $\mathbb{R}$-models and $\mathbb{Z}$-models. These constraint systems don't satisfy the completion property, and, as we show in Sec. 6, the set of models of a formula in these logics is *not*, in general, ω-regular. Nevertheless, we show that the satisfiability problems for the logics CLTL$(\mathbb{Z}, <, =)$ and CLTL$(\mathbb{N}, <, =)$ are decidable, and in fact PSPACE-complete. Our approach is automata-theoretic, in that we associate with a given formula φ, an automaton $\mathcal{A}^D_\varphi$ which is non-empty iff φ has a D-model (for $D = \mathbb{Z}, \mathbb{N}$). The technique used is as follows: Find an ω-regular superset M of $L^D(\varphi)$, which has the property that all its ultimately periodic words (those of the form $\tau \cdot \delta^\omega$) are also in $L^D(\varphi)$. This property guarantees that M is non-empty iff $L^D(\varphi)$ is. One can then check the emptiness of M using standard automata theory techniques. This lets us decide the D-satisfiability of φ.

The model checking problem for CLTL reduces easily to that of classical LTL. We address this briefly in Sec. 3, and focus on satisfiability in the sequel. Proofs omitted due to lack of space can be found in [DD02].

2 Preliminaries

A *(concrete) constraint system* is of the form $\mathcal{D} = (D, R_1 \ldots, R_n, \mathcal{I})$, where D is a non-empty set called a *domain*, and each R_i is a predicate symbol of arity

a_i, with $\mathcal{I}(R_i) \subseteq D^{a_i}$ being its interpretation. We will suppress the mention of $\mathcal{I}$ whenever it is clear from the context. Let us fix such a constraint system $\mathcal{D}$ for the rest of this section.

An *(atomic) $\mathcal{D}$ constraint* over a finite set of variables U is of the form $R_i(x_1, \ldots, x_{a_i})$ where each $x_i \in U$. A D-valuation over U is a map $s : U \rightarrow D$. Let $c = R_i(x_1, \ldots, x_{a_i})$ be a $\mathcal{D}$ constraint over U, and let s be a D-valuation over U. Then we say s satisfies c, written $s \models_{\mathcal{D}} c$, iff $(s(x_1), \ldots, s(x_{a_i})) \in \mathcal{I}(R_i)$.

The temporal logic we consider will use atomic constraints over terms which refer to the values of variables in succeeding time points. Let V be a finite set of variables which we will use in formulas of our logic, and which we fix in the rest of this paper. An *(atomic) term constraint* in $\mathcal{D}$ is of the form

$$R_i(O^{n_1}x_1, \ldots, O^{n_{a_i}}x_{a_i})$$

where $x_1, \ldots, x_{a_i} \in V$, $n_1, \ldots, n_{a_i} \in \mathbb{N}$. Here O^i stands for the juxtaposition of the next-state operator O i times, with $O^0 x$ representing just x. By the *O-length* of a term constraint c we will mean the value $i+1$ where i is the largest j for which O^j occurs in c. A term constraint in $\mathcal{D}$ is interpreted over a sequence of D-valuations. A D-valuation sequence is represented as a function $\sigma : \mathbb{N} \rightarrow \mathit{val}(D)$, where $\mathit{val}(D)$ denotes the set of D-valuations over V. We say a D-valuation sequence σ satisfies a term constraint $c = R_i(O^{n_1}x_1, \ldots, O^{n_{a_i}}x_{a_i})$, in $\mathcal{D}$, written $\sigma \models_{\mathcal{D}} c$, iff

$$(\sigma(n_1)(x_1), \ldots, \sigma(n_{a_i})(x_{a_i})) \in \mathcal{I}(R_i).$$

For $i, j \in \mathbb{N}$, and $i \leq j$, let us denote by $\sigma^{i,j}$ the finite valuation sequence $\sigma(i), \sigma(i+1), \ldots, \sigma(j)$. The truth of c in σ is thus determined by $\sigma^{0,k-1}$ where k is the O-length of c. A term constraint c can thus be viewed as a constraint over the variables $V \times \{0, \ldots, k-1\}$, where k is the O-length of c.

We now introduce CLTL($\mathcal{D}$), the constraint linear-time temporal logic, parameterised by the constraint system $\mathcal{D}$. The syntax of CLTL($\mathcal{D}$) is given by:

$$\varphi ::= c \mid \neg\varphi \mid (\varphi \vee \varphi) \mid O\varphi \mid (\varphi U \varphi)$$

where c is a term constraint in $\mathcal{D}$.

Models for CLTL($\mathcal{D}$) formulas are D-valuation sequences. Let φ be a CLTL($\mathcal{D}$) formula, and σ a D-valuation sequence. The satisfaction relation $\sigma \models \varphi$ is defined inductively below. We use the notation σ^i to denote the i-th suffix of σ given by $\sigma^i(l) = \sigma(l+i)$ for all $l \in \mathbb{N}$.

$$\begin{array}{ll}
\sigma \models c & \text{iff } \sigma \models_{\mathcal{D}} c \\
\sigma \models \neg\alpha & \text{iff } \sigma \not\models \alpha \\
\sigma \models \alpha \vee \beta & \text{iff } \sigma \models \alpha \text{ or } \sigma \models \beta \\
\sigma \models O\alpha & \text{iff } \sigma^1 \models \alpha \\
\sigma \models \alpha U \beta & \text{iff } \exists k \in \mathbb{N} \text{ such that } \sigma^k \models \beta \text{ and } \forall i : 0 \leq i < k,\ \sigma^i \models \alpha.
\end{array}$$

The semantics $\models_{\mathrm{LTL}}$ of classical LTL are given in a similar manner using propositions instead of constraints and models as propositional valuation sequences. $\Diamond$ and $\Box$ will stand for the usual temporal abbreviations for "sometimes" and "always", with the semantics $\Diamond\alpha \equiv (\top U \alpha)$, and $\Box\alpha \equiv \neg(\top U \neg\alpha)$.

As an example, consider the constraint system $(\mathbb{N}, <, =)$ with the usual interpretation of the symbols '$<$' and '$=$'. Let V be the set of variables $\{x, y\}$ and let us represent a valuation s over V as $\langle s(x), s(y)\rangle$. Then the formula $\Box(x < Oy)$ is satisfied by the $\mathbb{N}$-valuation sequence $\sigma = \langle 1, 2\rangle\langle 2, 4\rangle\langle 3, 6\rangle \cdots$, but not by $\sigma' = \langle 1, 1\rangle\langle 2, 2\rangle\langle 2, 2\rangle \cdots$.

The semantics of CLTL($\mathcal{D}$) can also be given in the sense of Prop. 1 below, in terms of what we call *frame sequences*. A *frame* over a set of variables U, w.r.t. a constraint system $\mathcal{D}$, is essentially a maximally consistent set of $\mathcal{D}$ constraints over U. More precisely, let $\mathit{frame}_{\mathcal{D}}(s, U)$ denote the set of $\mathcal{D}$ constraints over U satisfied by a D-valuation s over U. Then a set S of $\mathcal{D}$ constraints, is a *frame* over the variables U, w.r.t. $\mathcal{D}$, if there exists a D-valuation s over U such that $S = \mathit{frame}_{\mathcal{D}}(s, U)$. The *frame checking problem* for a constraint system $\mathcal{D}$ is to decide whether a given set of $\mathcal{D}$ constraints S, over a finite set of variables U, is a frame over U, w.r.t. $\mathcal{D}$.

In the context of terms in the logic it is convenient to define the notion of a *k-frame* which is essentially a frame over the set of variables $V \times \{0, \ldots, k-1\}$. Let $P_k^{\mathcal{D}}$ denote the set of propositions p_c such that c is a term constraint in $\mathcal{D}$ of O-length at most k. Let σ be a D-valuation sequence. We use $k\text{-}\mathit{frame}_{\mathcal{D}}(\sigma)$ to denote the set of all propositions $p_c \in P_k^{\mathcal{D}}$ such that $\sigma \models_{\mathcal{D}} c$. A *$k$-frame* w.r.t. $\mathcal{D}$ is a propositional valuation v over the set of propositions $P_k^{\mathcal{D}}$ (represented as a subset of $P_k^{\mathcal{D}}$) such that there exists a D-valuation sequence σ with $v = k\text{-}\mathit{frame}_{\mathcal{D}}(\sigma)$.

We say a pair of k-frames (v, v') is *locally consistent* if for all R_i, and for all $n_1, \ldots, n_{a_i} \in \{1, \ldots, k-1\}$:

$$R_i(O^{n_1}x_1, \ldots, O^{n_{a_i}}x_{a_i}) \in v \text{ iff } R_i(O^{n_1-1}x_1, \ldots, O^{n_{a_i}-1}x_{a_i}) \in v'.$$

(For readability, we write c instead of p_c.) Accordingly, a k-frame sequence ρ is locally consistent if for all $l \in \mathbb{N}$, the pairs $(\rho(l), \rho(l+1))$ are locally consistent. A given D-valuation sequence σ induces, for a given k, a canonical locally consistent k-frame sequence ρ, w.r.t. $\mathcal{D}$, denoted $k\text{-}\mathit{fs}_{\mathcal{D}}(\sigma)$, and given by $\rho(i) = k\text{-}\mathit{frame}_{\mathcal{D}}(\sigma^i)$. We extend k-fs to act on sets of valuation sequences in the natural way. A k-frame sequence ρ will be said to *admit* a D-valuation sequence, if there exists a D-valuation sequence σ such that $k\text{-}\mathit{fs}_{\mathcal{D}}(\sigma) = \rho$.

We say a CLTL($\mathcal{D}$) formula φ has O-length k if the largest O-length of constraints in φ is k. Thus the O-length of the formula $\Box(x < Oy)$ is 2. For a CLTL($\mathcal{D}$) formula φ, let $\widehat{\varphi}$ denote the classical LTL formula obtained from φ by replacing each constraint c by p_c. Then the following proposition is a direct consequence of the semantics of CLTL($\mathcal{D}$) and classical LTL.

Proposition 1. *Let φ be a* CLTL($\mathcal{D}$) *formula of O-length k. Let σ be a D-valuation sequence, and let $\rho = k\text{-}\mathit{fs}_{\mathcal{D}}(\sigma)$. Then $\sigma \models \varphi$ iff $\rho \models_{\mathrm{LTL}} \widehat{\varphi}$.* $\Box$

By the above proposition, a CLTL($\mathcal{D}$) formula φ of O-length k cannot differentiate between D-valuations that induce the same k-frame sequences. We define $L_{\mathit{kfs}}^{\mathcal{D}}(\varphi) = \{k\text{-}\mathit{fs}_{\mathcal{D}}(\sigma) \mid \sigma \models \varphi\}$. Stated differently, $L_{\mathit{kfs}}^{\mathcal{D}}(\varphi)$ is the set of all k-frame sequences (w.r.t. $\mathcal{D}$) that satisfy $\widehat{\varphi}$ and admit a D-valuation.

3 Model checking

Let $\mathcal{D} = (D, R_1, \ldots, R_n, \mathcal{I})$ be a given constraint system. A *D-Kripke structure* is of the form $(Q, \longrightarrow, l)$ where $(Q, \longrightarrow)$ is a finite directed graph, and $l : Q \longrightarrow val(D)$ is a labelling of states with D-valuations. For a D-Kripke structure $M = (Q, \longrightarrow, l)$ and a state $q \in Q$, we define $L(M, q)$ to be the set of all D-valuation sequences which appear as labels along an infinite path in M beginning at q.

Let $\varphi \in \mathrm{CLTL}(\mathcal{D})$. We say $M, q \models \varphi$ iff $\sigma \models \varphi$ for each $\sigma \in L(M, q)$. The model checking problem for $\mathrm{CLTL}(\mathcal{D})$ is: given a program modelled as a D-Kripke structure M, a state q in M, and a formula φ in $\mathrm{CLTL}(\mathcal{D})$; does $M, q \models \varphi$?

In the instance of the model checking problem above, let φ have O-length k. Then, by Prop. 1, $M, q \models \varphi$ iff $\rho \models_{\mathrm{LTL}} \widehat{\varphi}$ for every k-frame sequence ρ induced by a D-valuation sequence $\sigma \in L(M, q)$. We can define a classical $P_k^{\mathcal{D}}$-labelled Kripke structure $M_k = (T, \longrightarrow, m)$, which (for suitable choices of initial states) generates precisely the language $k\text{-}fs_{\mathcal{D}}(L(M, q))$. Define T to be the set of k-length paths in M. We define $q_0 \cdots q_{k-1} \longrightarrow r_0 \cdots r_{k-1}$ iff for each $i \in \{1, \ldots, k-1\}$, $q_i = r_{i-1}$. Define $m(q_0 \cdots q_{k-1}) = k\text{-}frame_{\mathcal{D}}(l(q_0) \cdots l(q_{k-1}))$.

The model checking problem for $\mathrm{CLTL}(\mathcal{D})$ now reduces to the model checking problem for LTL in the following sense: $(M, q) \models \varphi$ iff $(M_k, t) \models_{\mathrm{LTL}} \widehat{\varphi}$ for each $t \in T$ of the form $q_0 \cdots q_{k-1}$ with $q_0 = q$. This gives us a way to decide, in PSPACE, the model checking problem for constraint systems that are "well-behaved" in that the satisfiability problem for the predicates R_i are decidable in PSPACE.

Proposition 2. *For every well-behaved constraint system $\mathcal{D}$, the model checking problem for* $\mathrm{CLTL}(\mathcal{D})$ *is in* PSPACE. □

4 A general satisfiability result

For a constraint system $\mathcal{D}$, the satisfiability problem for $\mathrm{CLTL}(\mathcal{D})$ is: given a $\mathrm{CLTL}(\mathcal{D})$ formula φ, does there exist a D-valuation sequence which satisfies φ? Our approach to answering this problem is to decide whether there exists a k-frame sequence ρ (where k is the O-length of φ), which satisfies $\widehat{\varphi}$ *and* admits a D-valuation sequence. Since such a k-frame sequence must be locally consistent, we can restrict our attention to k-frame sequences which are locally consistent.

In general, a locally consistent k-frame sequence may not admit a D-valuation sequence (see for instance the constraint system $(\mathbb{Z}, <, =)$ in Sec. 6). However, when the constraint system satisfies the "completion" property described below, locally consistent k-frame sequences do admit D-valuation sequences.

A constraint system $\mathcal{D}$ is said to satisfy the *completion* property if given

- a frame S over a finite set of variables U w.r.t. $\mathcal{D}$,
- a subset U' of U, and
- a valuation s' over U', such that $S \restriction U' = frame_{\mathcal{D}}(s', U')$ (by $S \restriction U'$ we mean the subset of constraints in S which use only variables in U');

then there exists a valuation s over U which extends s' and satisfies $S = \mathit{frame}_{\mathcal{D}}(s, U)$. The constraint systems $(D, <, =)$ for $D = \mathbb{Q}, \mathbb{R}$ are examples of systems that satisfy the completion property.

Proposition 3. *Let $\mathcal{D}$ be a constraint system which satisfies completion. Then every locally consistent k-frame sequence w.r.t. $\mathcal{D}$ admits a D-valuation sequence.*

Proof. Let $\rho = v_0 v_1 \dots$ be a locally consistent k-frame sequence w.r.t. $\mathcal{D}$. We define a D-valuation sequence $\sigma = s_0 s_1 \cdots$, with the property that $\rho = k\text{-}\mathit{fs}_{\mathcal{D}}(\sigma)$, as follows. We view each v_i as a set of constraints over the variables $U_i = V \times \{i, \dots, i+k-1\}$. Since each v_i is a k-frame, for each i there exists a D-valuation t_i over U_i such that $v_i = \mathit{frame}_{\mathcal{D}}(t_i, U_i)$. Define $s_0(x) = t_0(x, 0), \dots, s_{k-1}(x) = t_0(x, k-1)$, for all $x \in V$. Now consider the set of constraints v_1. We know that v_1 is a frame over U_1. Further consider the subset $U' = V \times \{1, \dots, k-1\}$ of U_1. Clearly the restriction t' of the valuation t_0 to U', is such that $v_1 \restriction U' = \mathit{frame}_{\mathcal{D}}(t', U')$. Thus, by the completion property we have a D-valuation t'' which extends t' to the variables $V \times \{k\}$, and satisfies $v_1 = \mathit{frame}_{\mathcal{D}}(t'', U_1)$. We can now define $s_k(x) = t''(x, k)$ for each $x \in V$. This argument can be extended repeatedly to define the rest of σ. Clearly, σ has the property that $k\text{-}\mathit{fs}_{\mathcal{D}}(\sigma) = \rho$, and hence ρ admits the D-valuation sequence σ. $\square$

We can now state an interesting result regarding the language of k-frames defined by a CLTL($\mathcal{D}$) formula φ. Recall that a Büchi automaton over an alphabet A is simply a classical automaton $\mathcal{A} = (Q, q_0, \longrightarrow, F)$ over A, but with the set of final states F used as a acceptance condition on infinite words $\alpha \in A^\omega$. A run $\rho : \mathbb{N} \to Q$ on α is accepting if $\rho(i) \in F$ for infinitely many $i \in \mathbb{N}$. $L(\mathcal{A})$ is the set of infinite words accepted by $\mathcal{A}$, and $L \subseteq A^\omega$ is termed ω-regular if $L = L(\mathcal{A})$ for some Büchi automaton $\mathcal{A}$ over A (see [Tho90]).

Proposition 4. *Let $\mathcal{D}$ be a constraint system satisfying the completion property. Let φ be a* CLTL($\mathcal{D}$) *formula. Then $L^{\mathcal{D}}_{\mathit{kfs}}(\varphi)$ is ω-regular.*

Proof. Let φ be of O-length k. We define a Büchi automaton $\mathcal{A}^{\mathcal{D}}_{\varphi}$ over the alphabet $2^{(P^{\mathcal{D}}_k)}$, such that $L(\mathcal{A}^{\mathcal{D}}_{\varphi}) = L^{\mathcal{D}}_{\mathit{kfs}}(\varphi)$. Define $\mathcal{A}^{\mathcal{D}}_{\varphi} = \mathcal{A}_{\widehat{\varphi}} \cap \mathcal{A}^{\mathcal{D},k}_{lc}$, where $\mathcal{A}_{\widehat{\varphi}}$ is the Vardi-Wolper automaton [VW86] for the LTL formula $\widehat{\varphi}$, and $\mathcal{A}^{\mathcal{D},k}_{lc}$ is a Büchi automaton over $2^{(P^{\mathcal{D}}_k)}$ which accepts locally consistent k-frame sequences. We can define $\mathcal{A}^{\mathcal{D},k}_{lc} = (Q, q_0, \longrightarrow, F)$ where Q is the set of k-frames w.r.t. $\mathcal{D}$, along with a separate start state q_0; $\longrightarrow$ is given by $q_0 \xrightarrow{v} v$, and $v \xrightarrow{v'} v'$ iff (v, v') is locally consistent (here v and v' range over k-frames w.r.t $\mathcal{D}$); and $F = Q$. $\square$

Prop. 4 gives us a way to decide the satisfiability problem for a constraint system $\mathcal{D}$ which satisfies the completion property, and has a decidable frame checking problem. We can do this by constructing $\mathcal{A}^{\mathcal{D}}_{\varphi}$ and checking for language emptiness. In particular, if $\mathcal{D}$ is such that frame checking can be done in PSPACE, then the satisfiability problem for CLTL($\mathcal{D}$) can be solved in PSPACE. This is because both $\mathcal{A}_{\widehat{\varphi}}$ and $\mathcal{A}^{\mathcal{D},k}_{lc}$ are implicitly defined graphs whose transition relations can be checked (with the above assumption) in PSPACE in the length of φ.

Since the number of states in both $\mathcal{A}_{\widehat{\varphi}}$ and $\mathcal{A}_{lc}^{\mathcal{D},k}$ are exponential in length of φ, reachability of states in $\mathcal{A}_{\varphi}^{\mathcal{D}}$ can be done non-deterministically in PSPACE. Thus, emptiness checking, which essentially involves a couple of reachability checks, can be done non-deterministically in PSPACE in length of φ. This is essentially the result of [BC02].

Theorem 1. *The satisfiability problem for* CLTL($\mathcal{D}$) *when* $\mathcal{D}$ *satisfies completion and allows frame-checking in* PSPACE, *is in* PSPACE. □

5 Constraint systems of the form $(D, <, =)$

In this section we develop some notions which are useful in dealing with constraints systems of the form $(D, <, =)$. The case $D = \mathbb{R}$ is dealt with at the end of the section (the rationals $\mathbb{Q}$ behave very similarly as far as our logic is concerned), while $\mathbb{Z}$ and $\mathbb{N}$ are completed in the next section.

For a constraint system of the form $(D, <, =)$ it is convenient to visualise a frame as a labelled, directed graph. We represent a frame v over a finite set of variables U by a $\{<, =\}$-labelled, directed graph over the vertices U, where we place a '$\sim$'-labelled edge (for $\sim \in \{<, =\}$) from x to y precisely when $x \sim y \in v$. Such a graph clearly satisfies the conditions that:

1. there is an edge between every pair of vertices;
2. if there is '='-labelled edge from x to y then there is also one from y to x;
3. there are no *strict* cycles (i.e. directed cycles with a '<'-labelled edge).

Conversely, given an $\{<, =\}$-labelled graph G over a finite set of vertices U which satisfies the above conditions, one can verify that the set of constraints $v_G = \{x \sim y \mid x \xrightarrow{\sim} y \text{ in } G\}$ constitutes a frame over U. In the sequel we will often make use of this graphical representation of frames. Fig. 1 shows the first two 3-frames of a locally consistent 3-frame sequence over variables x, y.

A locally consistent k-frame sequence ρ can be represented as a single $\{<, =\}$-labelled, directed graph G_ρ, which is essentially the super-imposition of the overlapping k-frames (see Fig. 1). More precisely, G_ρ has $V \times \mathbb{N}$ as its set of vertices, and an edge $(x, i) \xrightarrow{\sim} (y, j)$ iff either $i \leq j$ and $(x \sim O^{j-i}y) \in \rho(i)$, or, $i > j$ and $(O^{i-j}x \sim y) \in \rho(j)$. An important property of G_ρ is that it does not contain any strict cycles (see below).

The frame checking problem for $(D, <, =)$ is essentially the cycle detection problem in a directed graph, which is in NLOGSPACE. It is not difficult to argue that $(\mathbb{R}, <, =)$ satisfies the completion property [DD02]. Thus, from Theorem 1 we conclude that the satisfiability problem for CLTL$(\mathbb{R}, <, =)$ is PSPACE-complete. As a corollary we can also conclude that for a locally consistent k-frame sequence ρ, G_ρ *always* admits an $\mathbb{R}$-valuation and hence contains no strict cycles.

6 Satisfiability for CLTL$(\mathbb{Z}, <, =)$

The constraint system $(\mathbb{Z}, <, =)$ does not satisfy the completion property and hence we cannot appeal to Theorem 1 to solve the satisfiability problem for the

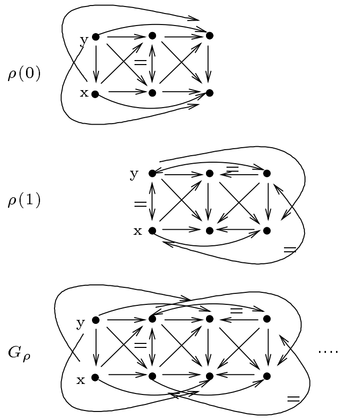

Fig. 1. A 3-frame sequence over the variables $\{x, y\}$

logic CLTL($\mathbb{Z}, <, =$). In fact, unlike $\mathbb{R}$, the language $L^{\mathbb{Z}}_{kfs}(\varphi)$ is *not*, in general, ω-regular (see Cor. 1). However, we can still solve the satisfiability problem for $\mathbb{Z}$ automata-theoretically. The idea is to define a Büchi automaton $\mathcal{A}^{\mathbb{Z}}_{\varphi}$ which accepts a superset of $L^{\mathbb{Z}}_{kfs}(\varphi)$, with the property that all *ultimately periodic* words in it are also in $L^{\mathbb{Z}}_{kfs}(\varphi)$. It then follows that $L(\mathcal{A}^{\mathbb{Z}}_{\varphi})$ is non-empty iff φ is $\mathbb{Z}$-satisfiable.

We begin with a characterisation of locally consistent k-frames which admit a $\mathbb{Z}$-valuation. Let ρ be a locally consistent k-frame sequence. For a directed path p in G_ρ, let $slen(p)$ (the *strict* length of p) denote the number of '$<$'-labelled edges in p if this number is finite, and ω otherwise. For any two vertices u, v in G_ρ, define $slen(u, v)$ to be the supremum of $slen(p)$ over directed paths p from u to v, if it exists, and ω otherwise.

Lemma 1. *Let ρ be a locally consistent k-frame sequence. Then ρ admits a $\mathbb{Z}$-valuation iff for all $u, v \in G_\rho$, $slen(u, v) < \omega$.*

Proof. If ρ admits a $\mathbb{Z}$-valuation, then clearly $slen(u, v) \leq \mid f(v) - f(u) \mid$, where f is a $\mathbb{Z}$-valuation admitted by ρ. Thus, there cannot exist vertices u and v with $slen(u, v) = \omega$. Conversely, suppose G_ρ satisfies the given condition. Then one can verify that the procedure given below produces a $\mathbb{Z}$-labelling f of the vertices of G_ρ that respects the labels on edges of G_ρ. This in turn implies that ρ admits a $\mathbb{Z}$-valuation. We assume an ordering $\prec$ on variables, and use it to define an ordering of vertices in G_ρ given by $(x, i) \prec (y, j)$ iff $i < j$, or $i = j$ and $x \prec y$.

1. Label the vertices in order. Begin by labelling the first, say $(x, 0)$, by 0.
2. In general, if X is the portion of the graph already labelled, and u is the next vertex to be labelled: if there is a directed path from u to a vertex in X, set $f(u) = \min\{f(v) - slen(u, v) \mid v \in X, \exists \text{ a path from } u \text{ to } v\}$, else, set $f(u) = \max\{f(v) + slen(v, u) \mid v \in X, \exists \text{ a path from } v \text{ to } u\}$. $\square$

Let ρ be a locally consistent k-frame sequence. An infinite *forward* increasing (respectively decreasing) chain in G_ρ is a sequence $d : \mathbb{N} \to V \times \mathbb{N}$ satisfying:

1. for all $i \in \mathbb{N}$, there is an edge from $d(i)$ to $d(i+1)$ (respectively, an edge from $d(i+1)$ to $d(i)$),
2. for all $i \in \mathbb{N}$, if $d(i)$ is in level j, then $d(i+1)$ is in a level greater than or equal to $j+1$. By the "level" of a vertex (x, i) we mean i.

Such a chain d is *strict* if there exist infinitely many i for which there is a '$<$'-labelled edge from $d(i)$ to $d(i+1)$ (respectively, from $d(i+1)$ to $d(i)$).

Consider now the condition $(C_\mathbb{Z})$ below on a locally consistent k-frame sequence ρ: there *do not* exist vertices u and v in the same k-frame in G_ρ satisfying:

1. there is an infinite forward increasing chain d from u,
2. there is an infinite forward decreasing chain e from v,
3. either d or e is strict, and
4. for each $i, j \in \mathbb{N}$, whenever $d(i)$ and $e(j)$ belong to the same k-frame there is an edge labelled '$<$' from $d(i)$ to $e(j)$.

It follows from Lemma 1 that condition $(C_\mathbb{Z})$ is necessary for ρ to admit a $\mathbb{Z}$-valuation sequence. But it is not sufficient, as witnessed by the 2-frame sequence $\rho^\mathbb{Z}_{bad}$ in Fig. 2. We have shown only the relevant edges in the figure. $\rho^\mathbb{Z}_{bad}$ clearly satisfies condition $(C_\mathbb{Z})$. However, it cannot admit a $\mathbb{Z}$-valuation since there are paths of unbounded strict length from $(x, 0)$ to $(z, 0)$.

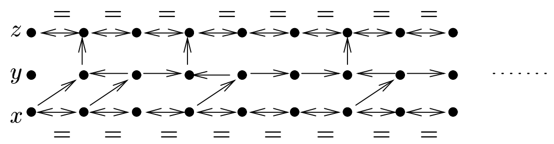

Fig. 2. Frame sequence $\rho^\mathbb{Z}_{bad}$ satisfies $(C_\mathbb{Z})$ but does not admit a $\mathbb{Z}$-valuation.

However, when ρ is an *ultimately periodic* word (i.e. of the form $\tau \cdot \delta^\omega$), this condition is indeed sufficient.

Lemma 2. *Let ρ be an ultimately periodic, locally consistent, k-frame sequence. Then ρ admits a $\mathbb{Z}$-valuation sequence iff ρ satisfies $(C_\mathbb{Z})$.*

Proof. If ρ admits a $\mathbb{Z}$-valuation, then by Lemma 1 ρ must satisfy the condition $(C_\mathbb{Z})$. Conversely, suppose ρ does not admit a $\mathbb{Z}$-valuation. We will show that ρ fails to meet condition $(C_\mathbb{Z})$. By Lemma 1 we know that there must exist two vertices u and v in G_ρ with $slen(u, v) = \omega$.

Now for any $i \in \mathbb{N}$, we can find a path p_i from u to v in G_ρ, of the form $u \xrightarrow{d_i} w \xleftarrow{e_i} v$ where d_i is a forward increasing chain from u to w, e_i is a forward decreasing from v to w, both d_i and e_i have length at least i, and either d_i or e_i has strict length at least i. This must be true, given the bounded width of the graph, and the unbounded strict length of paths from u to v.

We can also assume that u and v are in the same k-frame. Otherwise (say v was ahead of u), using an argument similar to König's Lemma, we can find a vertex u' in the same k-frame as v, with $slen(u', v) = \omega$.

Now let $\rho = \tau \cdot \delta^{\omega}$. Let there be N nodes in δ. Consider p_i for some $i > |\tau| + N$. We assume here that d_i is the one with at least i '$<$' edges (a symmetric argument holds when e_i has this property). Now it must be the case that d_i visits a vertex x in a δ block twice, *with at least one* '$<$' edge between the two occurrences (see Fig. 3). Let this segment of d_i, which begins at x and ends at x (in different δ blocks) be q. Note that the "future" of the vertices x in different δ blocks are exactly the same by the periodic nature of ρ. Now consider the path f from u given by the initial part of d_i to x, then q followed by q, ad infinitum. This is a strict infinite forward increasing chain from u in G_ρ. In a similar manner, the path e_i gives rise to an infinite forward (not necessarily strict) decreasing chain g from v, via an ultimately periodic concatenation of a path segment r between occurrences of y.

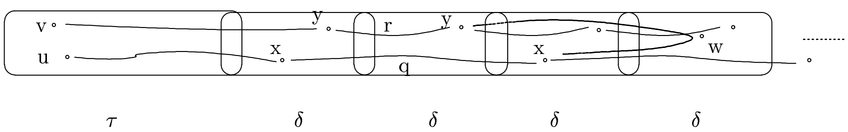

Fig. 3. Infinite paths extracted from p_i

Finally we must argue that for each vertex u' in f and v' in g, whenever they are in the same k-frame, there is an edge from u' to v' labelled $<$. The interesting case is when u' and v' lie on some copies of the path segments q and r (see Fig. 4). The initial occurrences of the segments q and r in f and g were part of the path p_i. Hence there is a path t from x to w to y. Further, this path is strict, since by the choice of i, the path from x to w must contain at least one '$<$'. Now, there must be a point in the future (a multiple of the lcm of the lengths of q and r) which occurs after u' and v', and where the path t is duplicated. Thus we have a path from u' to x, followed by a strict path t, followed by a path from y to v' in G_ρ. Since there must be an edge between u' and v' (being in the same k-frame), the edge must be strict and directed from u' to v' (otherwise G_ρ would have a strict cycle!). $\square$

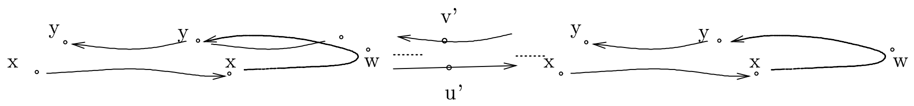

Fig. 4. $u' \xrightarrow{<} v'$

Let us now define $\mathcal{A}_{\varphi}^{\mathbb{Z}} = \mathcal{A}_{\widehat{\varphi}} \cap \mathcal{A}_{lc}^{\mathbb{Z},k} \cap \mathcal{A}^{\mathbb{Z}}$, where $\mathcal{A}_{\widehat{\varphi}}$ and $\mathcal{A}_{lc}^{\mathbb{Z},k}$ are as in Sec. 4, and $\mathcal{A}^{\mathbb{Z}}$, described below, accepts k-frame sequences which satisfy $(C_{\mathbb{Z}})$. Let $\mathcal{B}$

be a Büchi automaton over the alphabet of k-frames, which simply checks the negation of condition $(C_{\mathbb{Z}})$. Thus $\mathcal{B}$ non-deterministically guesses the vertices u and v, and then verifies the conditions (1)–(4). We non-deterministically choose in the beginning, whether to signal (via the Büchi condition) each time the d or e path sees an edge labelled '$<$'. The automaton $\mathcal{A}^{\mathbb{Z}}$ is now just the complement of $\mathcal{B}$.

Corollary 1. *There exist φ in* $\mathrm{CLTL}(\mathbb{Z}, <, =)$ *for which $L^{\mathbb{Z}}_{kfs}(\varphi)$ is not ω-regular.*

Proof. Consider the formula $\varphi = \Diamond((x < Oy) \wedge (y < z))$. The frame sequence $\rho^{\mathbb{Z}}_{bad}$ of Fig. 2 is a model for the LTL formula $\widehat{\varphi}$. Suppose $L^{\mathbb{Z}}_{kfs}(\varphi)$ were ω-regular. Then, $L = ((2^{P^{\mathcal{D}}_k})^{\omega} - L^{\mathbb{Z}}_{kfs}(\varphi)) \cap L(\mathcal{A}^{\mathbb{Z}}_{\varphi})$ is also ω-regular. Now, $\rho^{\mathbb{Z}}_{bad} \notin L^{\mathbb{Z}}_{kfs}(\varphi)$ and $\rho^{\mathbb{Z}}_{bad} \in L(\mathcal{A}^{\mathbb{Z}}_{\varphi})$. Hence $\rho^{\mathbb{Z}}_{bad} \in L$. But L is ω-regular, and hence there must exist an ultimately periodic word ρ in L. But $\rho \in L(\mathcal{A}^{\mathbb{Z}}_{\varphi})$ and hence satisfies $(C_{\mathbb{Z}})$. By Lemma 2, ρ admits a $\mathbb{Z}$-valuation sequence. Since $\rho \in L(\mathcal{A}_{\widehat{\varphi}})$, we can conclude that $\rho \in L^{\mathbb{Z}}_{kfs}(\varphi)$. This is a contradiction. □

Lemma 3. *A* $\mathrm{CLTL}(\mathbb{Z})$ *formula φ is $\mathbb{Z}$-satisfiable iff $L(\mathcal{A}^{\mathbb{Z}}_{\varphi})$ is non-empty.*

Proof. Suppose φ is $\mathbb{Z}$-satisfiable. Let $\sigma \models \varphi$, and let $\rho = k\text{-}fs(\sigma)$, where k is the O-length of φ. By Prop. 1, $\rho \in L(\mathcal{A}_{\widehat{\varphi}})$. We know that $\rho \in L(\mathcal{A}^{\mathbb{Z},k}_{lc})$. Further, by Lemma 1, G_ρ satisfies $(C_{\mathbb{Z}})$ and hence $\rho \in L(\mathcal{A}^{\mathbb{Z}})$. Thus $\rho \in L(\mathcal{A}_{\widehat{\varphi}}) \cap L(\mathcal{A}^{\mathbb{Z},k}_{lc}) \cap L(\mathcal{A}^{\mathbb{Z}})$. Hence $\rho \in L(\mathcal{A}^{\mathbb{Z}}_{\varphi})$.

Conversely, suppose $\mathcal{A}^{\mathbb{Z}}_{\varphi}$ accepts a word ρ. Then it must accept an ultimately periodic word ρ' (by nature of the acceptance condition). Now ρ' is locally consistent and satisfies $(C_{\mathbb{Z}})$. Hence by Lemma 2, ρ' must admit a $\mathbb{Z}$-valuation, say σ. Since $\rho' \models_{\mathrm{LTL}} \widehat{\varphi}$, by Prop. 1, $\sigma \models \varphi$. Thus φ is $\mathbb{Z}$-satisfiable. □

Theorem 2. *The satisfiability problem for* $\mathrm{CLTL}(\mathbb{Z}, <, =)$ *is* PSPACE*-complete.*

Proof. The number of states in $\mathcal{B}$ above can be seen to be polynomial in $|\varphi|$. Hence $\mathcal{A}^{\mathbb{Z}}$, which is the complement of $\mathcal{B}$, is at most exponential in $|\varphi|$ using, say, the construction of Safra [Saf88]. The number of states in both $\mathcal{A}_{\widehat{\varphi}}$ and $\mathcal{A}^{\mathbb{Z},k}_{lc}$ is also exponential in $|\varphi|$. Further, the transition relation of each of these components is implicitly defined in terms of φ and can be computed in polynomial space in $|\varphi|$. It follows that the emptiness of $\mathcal{A}^{\mathbb{Z}}_{\varphi}$ can be decided in PSPACE in the length of φ. PSPACE-hardness follows from that of LTL [SC85]. □

We close this section with an outline of the proof that the satisfiability problem for $\mathrm{CLTL}(\mathbb{N}, <, =)$ is PSPACE-complete. The treatment of this case is very similar to that of $\mathbb{Z}$. We outline the counterparts of the lemmas for the $\mathbb{Z}$ case. Define, for a vertex u in G_ρ, $sdlen(u)$ to be the supremum of $slen(p)$ over directed paths p from some vertex v to u in G_ρ. Then the counterpart of Lemma 1 is that a locally consistent k-frame sequence ρ admits an $\mathbb{N}$-valuation iff for all $u \in G_\rho, sdlen(u) < \omega$.

Let $(C_{\mathbb{N}})$ denote the following condition on a locally consistent k-frame sequence ρ: G_ρ satisfies $(C_{\mathbb{Z}})$ *and* it does not contain a strict infinite forward descending chain. One can now prove, using an argument similar to the proof of Lemma 2, that: an ultimately periodic, locally consistent, k-frame sequence ρ admits an $\mathbb{N}$-valuation sequence iff ρ satisfies $(C_{\mathbb{N}})$. Now define an automaton $\mathcal{A}_{\varphi}^{\mathbb{N}}$ as in the $\mathbb{Z}$ case, so that the formula φ is $\mathbb{N}$-satisfiable iff $L(\mathcal{A}_{\varphi}^{\mathbb{N}})$ is non-empty.

Theorem 3. *The satisfiability problem for* CLTL$(\mathbb{N}, <, =)$ *is* PSPACE-*complete.*

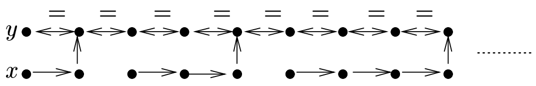

Fig. 5. Frame sequence $\rho_{bad}^{\mathbb{N}}$ satisfies $(C_{\mathbb{N}})$ but does not admit an $\mathbb{N}$-valuation.

Finally, the frame sequence $\rho_{bad}^{\mathbb{N}}$ of Fig. 5 allows us to argue that there is a CLTL$(\mathbb{N}, <, =)$ formula φ for which $L_{kfs}^{\mathbb{N}}(\varphi)$ is not ω-regular.

References

[All83] J. F. Allen. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11):832–846, 1983.

[BC02] Ph. Balbiani and J.F. Condotta. Computational complexity of propositional linear temporal logics based on qualitative spatial or temporal reasoning. In *Frontiers of Combining Systems (FroCoS'02)*, pages 162–176, 2002.

[CC00] H. Comon and V. Cortier. Flatness is not a weakness. In *14 Int. Workshop Computer Science Logic*, pages 262–276. LNCS, vol. 1862. Springer, 2000.

[DD02] S. Demri and D. D'Souza. An automata-theoretic approach to constraint LTL. *CMI Internal Report TCS-02-01*, 2002. `http://www.cmi.ac.in`.

[Pnu77] A. Pnueli. The temporal logic of programs. In *Proc. 18th IEEE Symposium on Foundation of Computer Science*, pages 46–57, 1977.

[RCC92] D. Randell, Z. Cui, and A. Cohn. A spatial logic based on regions and connection. In *3rd Conference on Knowledge Representation and Reasoning*, pages 165–176. Morgan Kaufman, 1992.

[Saf88] S. Safra. On the complexity of ω-automata. *Proc. 29th IEEE Symp. on Foundations of Computer Science*, pages 319–327, 1988.

[SC85] P. Sistla and E. M. Clarke. The complexity of propositional linear temporal logics. *JACM*, 32(3):733–749, 1985.

[Tho90] W. Thomas. Automata on infinite objects. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, volume B. Elsevier, 1990.

[VW86] M. Vardi and P. Wolper. An automata theoretic approach to automatic program verification. In *Logic in Computer Science*, pages 332–334. IEEE, 1986.

[WZ00] F. Wolter and M. Zakharyaschev. Spatio-temporal representation and reasoning based on RCC-8. In *KR'00*, pages 3–14. Morgan Kaufmann, 2000.

[WZ02] F. Wolter and M. Zakharyaschev. Qualitative spatio-temporal representation and reasoning: a computational perspective. In *Exploring Artificial Intelligence in the New Millenium*, 2002. To appear.