A Decidable Temporal Logic of Repeating Values*

Stéphane Demri[1], Deepak D'Souza[2], and Régis Gascon[1]

[1] LSV, ENS Cachan, CNRS, INRIA
{demri,gascon}@lsv.ens-cachan.fr
[2] Dept. of Computer Science & Automation,
Indian Institute of Science, Bangalore, India
deepakd@csa.iisc.ernet.in

Abstract. Various logical formalisms with the freeze quantifier have been recently considered to model computer systems even though this is a powerful mechanism that often leads to undecidability. In this paper, we study a linear-time temporal logic with past-time operators such that the freeze operator is only used to express that some value from an infinite set is repeated in the future or in the past. Such a restriction has been inspired by a recent work on spatio-temporal logics. We show decidability of finitary and infinitary satisfiability by reduction into the verification of temporal properties in Petri nets. This is a surprising result since the logic is closed under negation, contains future-time and past-time temporal operators and can express the nonce property and its negation. These ingredients are known to lead to undecidability with a more liberal use of the freeze quantifier.

1 Introduction

Temporal logic with freeze. In logical languages, the freeze mechanism allows to store a value in a register and to test later the value in the register with a current value. This operator is useful to compare values at distinct states of Kripke-like structures. The freeze quantifier has found applications in real-time logics [Hen90], in hybrid logics [Gor96,ABM01], in modal logics with predicate λ-abstraction [Fit02] and for the specification of computations of systems with unboundedly many locations as resources [LP05]. Although it is known that the freeze operator can lead to undecidability (even with only equality on data [LP05,DLN07]), many decidable temporal logics have a freeze mechanism, sometimes implicitly, see e.g. [AH94,LMS02,KV06]. Recent developments have shown the ubiquity of the freeze operator [LP05,tCF05,DLN07,Laz06,Seg06] and its high expressive power as witnessed by the Σ^1_1-completeness results shown in [DLN07].

* Work supported by the Indo-French project "Timed-DISCOVERI" (P2R/RNP scheme).

The need to design decidable fragments of simple linear-time temporal logic LTL with the freeze quantifier stems from [DLN07,Laz06] and most known decidable fragments in [DLN07,Laz06] does not allow unrestricted use of negation. Still, finitary and infinitary satisfiability for Boolean combinations of safety formulae (with a unique register) is decidable [Laz06]. Potential applications range from the verification of infinite-state systems [Hen90,DLN07] to querying XML documents or more modestly data strings [BMS$^+$06,Seg06]. In the paper, we are interested in studying fragments of LTL with the freeze operator that are decidable in the finitary and infinitary cases, that allow unrestricted use of negation (by contrast to the flat fragments in [DLN07]) and that allow all standard past-time operators (by constrast to what is done in [BMS$^+$06,DL06]). Even in terms of expressive power, the fragment newly shown decidable in the paper can express the "nonce property" and its negation (all the values of a variable are different at every position). Moreover, in [WZ00, Sect. 7], the authors advocate the need to consider infinitary disjunction of the form $\bigvee_{i>0} x = \mathsf{X}^i y$ where $\mathsf{X}^i y$ refers to the value of y at the i^{th} next position. This states that a future value of y is equal to the current value of x. Our fragment can express this property, with the formula $x = \Diamond y$, as well as the dual one: $\bigwedge_{i>0} x = \mathsf{X}^i y$ can be expressed by the formula $\neg(x \neq \Diamond y)$. In the paper we introduce the constraint logic CLTL$(\mathbb{N}, =)$ with atomic formulae $x = \Diamond y$ and past-time operators X^{-1} and S. This logic is denoted by CLTL$^\Diamond$. Hence, in CLTL$^\Diamond$, the freeze quantifier is only used to specify that some values are repeated. Even though CLTL$^\Diamond$ does not enjoy first-order completeness, see e.g. [Rab06], it satisfies interesting computational properties as shown below.

Our contribution. We show that finitary and infinitary satisfiability for CLTL$^\Diamond$ with temporal operators $\{\mathsf{X}, \mathsf{X}^{-1}, \mathsf{S}, \mathsf{U}\}$ is decidable. We provide a uniform proof for the finite and infinite cases based on some substantial extension of the automaton-based approach for (constraint) LTL from [VW94,DD07]. The possibility to compare two values at unbounded distance requires a special class of counter automata for which finitary and infinitary nonemptiness is shown decidable. To do so, we take advantage of a deep result from [Jan90] establishing that verifying fairness properties based on the temporal operator GF ("always eventually") in Petri nets is decidable. By contrast model-checking for full LTL over Petri nets is undecidable [HR89] (see also [Esp94] with linear-time mu-calculi). Observe that infinitary CLTL$^\Diamond$ is the first decidable fragment of CLTL$_1^\downarrow(\mathbb{N}, =)$ [DLN07] with an unrestricted use of negation and that contains all the temporal operators from $\{\mathsf{X}, \mathsf{X}^{-1}, \mathsf{S}, \mathsf{U}\}$. A nice by-product of our technique is that the extensions with temporal operators definable in Monadic Second Order Logic (MSOL) or with $x = \Diamond^{-1} y$ ("a value of y in the past is equal to the current value of x") are also decidable. Finally, we show that finitary and infinitary satisfiability for CLTL$^\Diamond$ restricted to one variable is PSPACE-complete.

Because of lack of space, omitted proofs can be found in [DDG07].

2 Preliminaries

2.1 Temporal Logic with Repeating Values

Let $\mathrm{VAR} = \{x_1, x_2, \ldots\}$ be a countably infinite set of variables. The formulae of the logic $\mathrm{CLTL}^{\Diamond}$ are defined as follows:

$$\phi ::= x = \mathsf{X}^i y \mid x = \Diamond y \mid \phi \wedge \phi \mid \neg\phi \mid \mathsf{X}\phi \mid \phi \mathsf{U} \phi \mid \mathsf{X}^{-1}\phi \mid \phi \mathsf{S} \phi$$

where $x, y \in \mathrm{VAR}$ and $i \in \mathbb{N}$. Formulae of the form either $x = \mathsf{X}^i y$ or $x = \Diamond y$ are said to be atomic and an expression of the form $\mathsf{X}^i x$ (i next symbols followed by a variable) is called a term. Given a set of temporal operators definable from those in $\{\mathsf{X}, \mathsf{X}^{-1}, \mathsf{S}, \mathsf{U}\}$ and $k \geq 0$, we write $\mathrm{CLTL}^{\Diamond}_k(\mathcal{O})$ to denote the fragment of $\mathrm{CLTL}^{\Diamond}$ restricted to formulae with temporal operators from $\mathcal{O}$ and with at most k variables.

A valuation is a map $\mathrm{VAR} \to \mathbb{N}$ and a model σ is a non-empty sequence of valuations either finite or infinite. All the subsequent developments can be equivalently done with the domain $\mathbb{N}$ replaced by an infinite set D since only equality tests are performed. We write $|\sigma|$ to denote the length of σ. The satisfaction relation is defined inductively as follows where σ is a model and $0 \leq i \leq |\sigma| - 1$:

- $\sigma, i \models x = \mathsf{X}^j y$ iff $i + j \leq |\sigma| - 1$ and $\sigma(i)(x) = \sigma(i + j)(y)$,
- $\sigma, i \models x = \Diamond y$ iff there exists $j > 0$ s.t. $i+j \leq |\sigma|-1$ and $\sigma(i)(x) = \sigma(i+j)(y)$,
- $\sigma, i \models \phi \wedge \phi'$ iff $\sigma, i \models \phi$ and $\sigma, i \models \phi'$, $\quad \sigma, i \models \neg\phi$ iff $\sigma, i \not\models \phi$,
- $\sigma, i \models \mathsf{X}\phi$ iff $i + 1 \leq |\sigma| - 1$ and $\sigma, i + 1 \models \phi$,
- $\sigma, i \models \mathsf{X}^{-1}\phi$ iff $i > 0$ and $\sigma, i - 1 \models \phi$,
- $\sigma, i \models \phi \mathsf{U} \phi'$ iff there is $i \leq j \leq |\sigma| - 1$ s.t. $\sigma, j \models \phi'$ and for every $i \leq l < j,\ \sigma, l \models \phi$.
- $\sigma, i \models \phi \mathsf{S} \phi'$ iff there is $0 \leq j \leq i$ s.t. $\sigma, j \models \phi'$ and for $j \leq l < i,\ \sigma, l \models \phi$.

We write $\sigma \models \phi$ if $\sigma, 0 \models \phi$. We shall use the standard abbreviations about the temporal operators (G, F, F^{-1}, ...) and Boolean operators ($\vee$, $\Rightarrow$, ...). We use the notation $\mathsf{X}^i x = \mathsf{X}^j y$ as an abbreviation for $\mathsf{X}^i(x = \mathsf{X}^{j-i} y)$ (when $i \leq j$).

The *finitary [resp. infinitary] satisfiability problem* consists in checking whether given a formula ϕ, there is a finite [resp. infinite] model such that $\sigma \models \phi$. It is known that finitary satisfiability for LTL can be easily reduced in logspace to infinitary satisfiability by introducing for instance an additional propositional variable p and by requiring that $p\mathsf{U}\mathsf{G}\neg p$ holds true. In that way, p holds true at every state of a prefix and p does not hold on the complement suffix. The same principle does not apply to reduce finitary satisfiability for $\mathrm{CLTL}^{\Diamond}$ to infinitary satisfiability even by introducing additional variables in order to simulate a propositional variable. This is due to the additional atomic formulae of the form $x = \Diamond y$. That is why we distinguish the two problems in this paper.

We note that a constraint of the form x *diff* $\Diamond y$ ("the value of x differs from some future value of y") can be expressed in $\mathrm{CLTL}^{\Diamond}$:

$$x \textit{ diff } \Diamond y \;\Leftrightarrow\; (\neg(x = \mathsf{X}y) \wedge \mathsf{X}\top) \vee ((x = \mathsf{X}y) \wedge \mathsf{X}(y = \mathsf{X}y)\mathsf{U}((y \neq \mathsf{X}y) \wedge \mathsf{X}\top))) \quad (1)$$

With infinite models, the conjunct $\mathsf{X}\top$ can be deleted. We could also introduce constraints of the form $x = \mathsf{X}^{-i}y$ but this can be expressed in the language using the equivalence $x = \mathsf{X}^{-i}y \Leftrightarrow \mathsf{X}^{-i}\top \wedge \mathsf{X}^{-i}(y = \mathsf{X}^i x)$. Similarly, CLTL$^{\Diamond}$ can express whether a variable is a nonce by the formula $\mathsf{G}\neg(x = \Diamond x)$. The formula below states a valid property when x and y are nonces:

$$(\mathsf{G}\neg(x = \Diamond x) \wedge \mathsf{G}\neg(y = \Diamond y)) \Rightarrow \mathsf{G}(x = y \Rightarrow \neg(x = \Diamond y)).$$

Other properties witnessing the high expressive power of CLTL$^{\Diamond}$ can be found in [LP05, Sect.3] about systems of pebbles evolving in time.

Apart from the above-mentioned problems, we could also consider standard model-checking problems that can be reduced to the satisfiability problem. Indeed, we can define a suitable class of automata such that the execution of any automaton of this class can be encoded into a CLTL$^{\Diamond}$ formula.

2.2 Known extensions of CLTL$^{\Diamond}$

In this section, we recall the definition of a few known extensions of CLTL$^{\Diamond}$ which is useful for future comparisons. The logic CLTL$^{\Diamond}$ is clearly a fragment of the logic CLTL$^{\downarrow}(\mathbb{N}, =)$ introduced in [DLN07] and restricted to one register. An equivalent logic of CLTL$^{\downarrow}(\mathbb{N}, =)$ is denoted by CLTL$^{\downarrow}$ in this paper and it is defined as follows. We consider an additional set of registers $\mathrm{REG} = \{r_1, r_2, \ldots\}$ and the formulae of CLTL$^{\downarrow}$ are defined as those of CLTL$^{\Diamond}$ except that we allow only atomic formulae of the form $r = x$ where $r \in \mathrm{REG}$ and $x \in \mathrm{VAR}$, and we add the inductive clause $\downarrow_{r=x} \phi$. The satisfaction relation is parameterized by a register assignment $\rho : \mathrm{REG} \to \mathbb{N}$ with $\sigma, i \models_\rho \downarrow_{r=x} \phi$ iff $\sigma, i \models_{\rho[r \mapsto \sigma(i)(x)]} \phi$ and $\sigma, i \models_\rho r = x$ iff $\rho(r) = \sigma(i)(x)$. Consequently, the atomic formula $x = \Diamond y$ in CLTL$^{\Diamond}$ can be naturally encoded in CLTL$^{\downarrow}$ by $\downarrow_{r=x} \mathsf{XF}(r = y)$ and $x = \mathsf{X}^i y$ by $\downarrow_{r=x} \mathsf{X}^i(r = y)$.

We write $\mathrm{CLTL}^{\downarrow}_{(k,k')}(\mathcal{O})$ to denote the fragment of CLTL$^{\downarrow}$ restricted to the temporal operators from $\mathcal{O}$ with at most k variables and k' registers. Following the notation from [DL06], for $\alpha \geq 0$ we write $\mathrm{LTL}^{\downarrow}_{\alpha}(\sim, \mathcal{O})$ to denote the fragment $\mathrm{CLTL}^{\downarrow}_{(1,\alpha)}(\mathcal{O})$ restricted to atomic formulae of the form $r = x$.

2.3 A decidable fragment of finitary satisfiability

It is shown in [DLN07] that CLTL$^{\downarrow}$ is strictly more expressive than its freeze-free fragment. The same argument applies to show that CLTL$^{\Diamond}$ is strictly more expressive than its fragment without atomic formulae of the form $x = \Diamond y$. Observe also that CLTL$^{\Diamond}$ is neither a fragment of the pure-future safety fragment in [Laz06] (where occurrences of U formulae are never in the scope of an even number of negations) nor a fragment of the flat fragment of CLTL$^{\downarrow}$. Unlike these fragments, CLTL$^{\Diamond}$ contains past-time operators and negation can be used without any restriction. Infinitary satisfiability for safety $\mathrm{LTL}^{\downarrow}_1(\sim, \mathsf{X}, \mathsf{U})$ is EXPSPACE-complete [Laz06], for full $\mathrm{LTL}^{\downarrow}_1(\sim, \mathsf{X}, \mathsf{U})$ is Π^0_1-complete and, finitary

and infinitary satisfiability for flat CLTL$^{\downarrow}$ are PSPACE-complete. By contrast, in this paper we show that finitary and infinitary satisfiability for CLTL$^{\Diamond}$ (with full past-time temporal operators) are decidable problems. By taking advantage of [DLN07,DL06], it is already possible to establish decidability of *finitary* satisfiability for *strict* fragments of CLTL$^{\Diamond}$.

Theorem 1. (I) *Finitary satisfiability for* CLTL$^{\Diamond}(\mathsf{X},\mathsf{U})$ *is decidable.*
(II) *Finitary satisfiability for* CLTL$^{\Diamond}(\mathsf{X},\mathsf{X}^{-1},\mathsf{F},\mathsf{F}^{-1})$ *is decidable.*

In the paper we shall show much stronger results: finitary and infinitary satisfiability for full CLTL$^{\Diamond}$ even augmented with MSOL definable temporal operators are decidable. In the rest of the paper, we systematically treat the finitary and infinitary cases simultaneously. However, we provide the full technical details for the infinitary case only and we sketch the main ideas for the finitary case. This latter case cannot be reduced in the obvious way to the infinitary case but its solution is close to the one for the infinitary case.

3 Automata-based Approach with Symbolic Models

In this section we explain how satisfiability can be solved using symbolic models which are abtractions of concrete CLTL$^{\Diamond}$ models. We provide the outline of our automata-based approach, and consider the technical details in Sects. 4 and 5.

3.1 Symbolic Models

Let ϕ be a CLTL$^{\Diamond}$ formula with k variables $\{x_1,\dots x_k\}$ and we write l (the "X-length" of ϕ) to denote the maximal i such that a term of the form $\mathsf{X}^i x$ occurs in ϕ. In order to define the set of atomic formulae that are helpful to determine the satisfiability status of ϕ, we introduce the set of constraints Ω_k^l that contains constraints of the form either $\mathsf{X}^i x = \mathsf{X}^j y$ or $\mathsf{X}^i(x = \Diamond y)$ and their negation with $x, y \in \{x_1,\dots x_k\}$ and $i, j \in \{0,\dots,l\}$. Models are abstracted as sequences of frames that are defined as maximally consistent subsets of Ω_k^l.

An *l-frame* is a set $fr \subseteq \Omega_k^l$ that is maximally consistent in that it satisfies the conditions below:

(F1) For every constraint $\varphi \in \Omega_k^l$, either φ or $\neg\varphi$ belongs to fr but not both.
(F2) For all $i \in \{0,\dots,l\}$ and $x \in \{x_1,\dots x_k\}$, $\mathsf{X}^i x = \mathsf{X}^i x \in fr$.
(F3) For all $i, j \in \{0,\dots,l\}$ and $x, y \in \{x_1,\dots x_k\}$, $\mathsf{X}^i x = \mathsf{X}^j y \in fr$ iff $\mathsf{X}^j y = \mathsf{X}^i x \in fr$.
(F4) For all $i, j, j' \in \{0,\dots,l\}$ and $x, y, z \in \{x_1,\dots,x_k\}$, $\{\mathsf{X}^i x = \mathsf{X}^j y, \mathsf{X}^j y = \mathsf{X}^{j'} z\} \subseteq fr$ implies $\mathsf{X}^i x = \mathsf{X}^{j'} z \in fr$.
(F5) For all $i, j \in \{0,\dots,l\}$ and $x, y \in \{x_1,\dots x_k\}$ such that $\mathsf{X}^i x = \mathsf{X}^j y \in fr$:
- if $i = j$, then for every $z \in \{x_1,\dots,x_k\}$ we have $\mathsf{X}^i(x = \Diamond z) \in fr$ iff $\mathsf{X}^j(y = \Diamond z) \in fr$;
- if $i < j$ then $\mathsf{X}^i(x = \Diamond y) \in fr$, and for $z \in \{x_1,\dots,x_k\}$, $\mathsf{X}^i(x = \Diamond z) \in fr$ iff either $\mathsf{X}^j(y = \Diamond z) \in fr$ or there exists $i < j' \leq j$ such that $\mathsf{X}^i x = \mathsf{X}^{j'} z \in fr$.

Conditions (F2)–(F4) simply encode that equality is an equivalence relation.

For an l-frame fr, $x \in \{x_1, \ldots, x_k\}$ and $i \in \{0, \ldots, l\}$, we define

- the set of future obligations for x at level i in fr as $\Diamond_{fr}(x,i) \stackrel{\text{def}}{=} \{y \mid \mathsf{X}^i(x = \Diamond y) \in fr\}$,
- the equivalence class of x at level-i in fr as $[(x,i)]_{fr} \stackrel{\text{def}}{=} \{y \mid \mathsf{X}^i x = \mathsf{X}^i y \in fr\}$.

An l-frame fr can be represented as an annotated undirected graph G_{fr} which has vertices (x,i) for $x \in \{x_1, \ldots, x_k\}$ and $i \in \{0, \ldots, l\}$, and an edge between (x,i) and (y,j) iff the constraint $\mathsf{X}^i x = \mathsf{X}^j y$ belongs to fr. A vertex (x,i) in the graph is annoted with an "open" arc labelled by the set of future obligations $\Diamond_{fr}(x,i)$ for that vertex. Fig. 1 shows an example of a 3-frame over the variables $\{x, y, z\}$. For convenience we avoid showing transitively inferable edges.

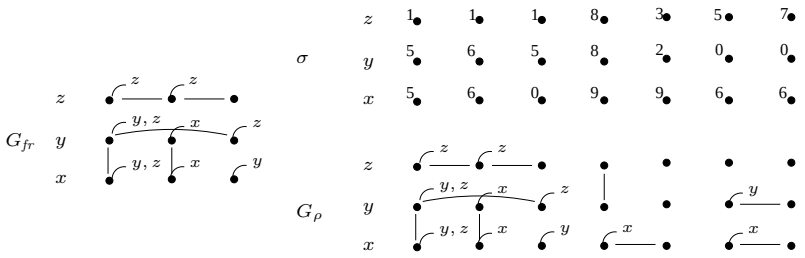

Fig. 1. Example 3-frame graph, concrete model σ, and its induced 3-frame graph G_ρ.

We denote by $\mathtt{Frame}_k^l$ the set of such frames built w.r.t. k and l. We say that a model σ satisfies a frame fr at position i (denoted $\sigma, i \models fr$) iff $\sigma, i \models \varphi$ for every constraint φ in fr.

Since a frame can be viewed as a set of constraints about $l+1$ consecutive positions, for finitary satisfiability we need to add an information about the possibility to end the model before the end of the current window of length $l+1$. This can be done with $\mathcal{O}(l)$ bits and then the conditions (F1)–(F5) need to be updated accordingly in order to take into account this possibility. Note that this method allows to handle the particular case where there exists a model whose size is smaller than the X-length of the formula.

Lemma 1. *For all models σ with k variables and $0 \leq i \leq |\sigma| - 1$, there exists a unique frame $fr \in \mathtt{Frame}_k^l$ such that $\sigma, i \models fr$.*

A pair of l-frames $\langle fr, fr' \rangle$ is said to be *one-step consistent* $\stackrel{\text{def}}{\Leftrightarrow}$

- for all $0 < i, j \leq l$, $\mathsf{X}^i x = \mathsf{X}^j y \in fr$ iff $\mathsf{X}^{i-1} x = \mathsf{X}^{j-1} y \in fr'$,
- for all $0 < i \leq l$, $\mathsf{X}^i(x = \Diamond y) \in fr$ iff $\mathsf{X}^{i-1}(x = \Diamond y) \in fr'$.

A *symbolic model* of X-length l is a (finite or infinite) sequence of l-frames ρ such that for $0 \leq i < |\rho| - 1$, $\langle \rho(i), \rho(i+1) \rangle$ is one-step consistent. We define the symbolic satisfaction relation $\rho, i \models_{\text{symb}} \phi$, for a formula ϕ of X-length l and a symbolic model ρ of X-length l, as done for CLTL$^{\Diamond}$ except that for atomic formulas φ we have: $\rho, i \models_{\text{symb}} \varphi \stackrel{\text{def}}{\Leftrightarrow} \varphi \in \rho(i)$. We say a model σ *realizes* a symbolic model ρ (or equivalently that ρ *admits* a model σ) $\stackrel{\text{def}}{\Leftrightarrow}$ for every $0 \leq i \leq |\sigma| - 1$, we have $\sigma, i \models \rho(i)$.

A symbolic model ρ of X-length l can also be represented as an annotated graph G_ρ in a similar manner to l-frames. Thus the vertices of G_ρ are of the form (x, i) with an edge between (x, i) and (y, j) with $0 \leq j - i \leq l$ iff there was an edge between $(x, 0)$ and $(y, j - i)$ in the frame graph $G_{\rho(i)}$. The annotations for future obligations are added similarly. Fig. 1 shows the graph representation of a symbolic model ρ of X-length 3, and a model it admits. By a *path* p in G_ρ we will mean as usual a (finite or infinite) sequence of vertices $v_0, v_1 \ldots$ in G_ρ such that each v_i, v_{i+1} is connected by an edge in G_ρ. We call p a *forward* path if each v_{i+1} is at a level strictly greater than v_i.

3.2 Automata for Symbolic Models

In order to check whether a CLTL$^{\Diamond}$ formula is satisfiable we use Lemma 2 below based on the approach developed in [DD07].

Lemma 2. *A* CLTL$^{\Diamond}$ *formula ϕ of X-length l is satisfiable iff there exists a symbolic model ρ of X-length l such that $\rho \models_{\text{symb}} \phi$ and ρ admits a model.*

In order to take advantage of Lemma 2, we use the automaton-based approach from [VW94]. We build an automaton $\mathcal{A}_\phi$ as the intersection of two automata $\mathcal{A}_{\text{symb}}$ and $\mathcal{A}_{\text{sat}}$ such that the language recognized by $\mathcal{A}_{\text{symb}}$ is the set of symbolic models satisfying ϕ and the language recognized by $\mathcal{A}_{\text{sat}}$ is the set of symbolic models that are realized by some models.

We define the automaton $\mathcal{A}_{\text{symb}}$ by adapting the construction from [VW94] for LTL. We define $cl(\phi)$ the closure of ϕ as usual, and an atom of ϕ is a maximally consistent subset of $cl(\phi)$. For the infinitary case, $\mathcal{A}_{\text{symb}}$ is the generalized Büchi automaton $(Q, Q_0, \rightarrow, F)$ such that:

- Q is the set of atoms of ϕ and $Q_0 = \{At \in Q \mid \phi \in At,\ \mathsf{X}^{-1}\top \notin At\}$,
- $At \xrightarrow{fr} At'$ iff
  (atomic constraints) for every atomic formula φ in At, $\varphi \in fr$,
  (one step) for every $\mathsf{X}\psi \in cl(\phi)$, $\mathsf{X}\psi \in At$ iff $\psi \in At'$, and for every $\mathsf{X}^{-1}\psi \in cl(\phi)$, $\psi \in At$ iff $\mathsf{X}^{-1}\psi \in At'$
- let $\{\psi_1 \mathsf{U} \phi_1, \ldots, \psi_r \mathsf{U} \phi_r\}$ be the set of until formulae in $cl(\phi)$. We pose $F = \{F_1, \ldots, F_r\}$ where for every $i \in \{1, \ldots, r\}$, $F_i = \{At \in Q : \psi_i \mathsf{U} \phi_i \notin At \text{ or } \phi_i \in At\}$.

For the finitary case, the finite-state automaton $\mathcal{A}_{\text{symb}}$ accepting finite words is defined as above except that F is a set of states At such that no atomic formula

of the form $x = \Diamond y$ occurs in At and no formula of the form either $\mathsf{X}\phi$ or $x = \mathsf{X}^i y$ with $i > 0$ occurs in At. Moreover, such final states can only be reached when the frame labelling the last transition contains proper information about the end of the model.

In the next section, we explain how one can build the automaton $\mathcal{A}_{\text{sat}}$ that recognizes the set of satisfiable symbolic models. Since $\mathcal{A}_\phi$ is the automaton recognizing the intersection of the languages accepted by $\mathcal{A}_{\text{symb}}$ and $\mathcal{A}_{\text{sat}}$, the following result is a direct consequence of Lemma 2.

Theorem 2. *A* CLTL$^\Diamond$ *formula* ϕ *is satisfiable iff the language recognized by* $\mathcal{A}_\phi$ *is nonempty.*

Note that we separate the temporal logic part and the constraint part by defining two different automata. This allows to extend the decidability results to any extension of LTL that induces an ω-regular class of models. We only need to change the definition of $\mathcal{A}_{\text{symb}}$.

4 Characterization of Satisfiable Symbolic Models

In order to determine whether a symbolic model ρ is "satisfiable" (i.e. it admits a model), we introduce counters that remember the satisfaction of constraints $x = \Diamond y$. If $x = \Diamond y_1 \wedge \cdots \wedge x = \Diamond y_n$ needs to be satisfied at the current position, then we shall increment a counter indexed by $\{y_1, \ldots, y_n\}$ that remembers this set of obligations. In a finite model, all the obligations need to be fulfilled before the last position whereas in an infinite model either no more unsatisfied obligations arise after a point, or they are essentially fulfilled infinitely often. The exact conditions will be spelt out soon.

4.1 Counting Sequence

For each $X \in \mathcal{P}^+(\{x_1, \ldots, x_k\})$ (set of non-empty subsets of $\{x_1, \ldots, x_k\}$), we introduce a counter that keeps track of the number of obligations that need to be satisfied by X. We identify the counters with elements of $\mathcal{P}^+(\{x_1, \ldots, x_k\})$. A counter valuation $\boldsymbol{c}$ is a map $\boldsymbol{c} : \mathcal{P}^+(\{x_1, \ldots, x_k\}) \to \mathbb{N}$. For instance, we write $\boldsymbol{c}(\{x, y\})$ to denote the value of the counter $\{x, y\}$, which will stand for the number of obligations to repeat a distinct value in x and y.

We will define a canonical sequence of counter valuations along a symbolic model. We introduce some definitions first. For an l-frame fr and $X \in \mathcal{P}^+(\{x_1, \ldots, x_k\})$, we define a *point of increment* for X in fr to be an equivalence class of the form $[(x, 0)]_{fr}$ such that $\Diamond_{fr}(x, 0) = X$ and $(x, 0)$ is not connected by a forward edge to a node in fr (i.e. there is no edge between $(x, 0)$ and (y, j) for any $j \in \{1, \ldots l\}$). A *point of decrement* for X in fr is defined to be an equivalence class of the form $[(x, l)]_{fr}$ such that $\Diamond_{fr}(x, l) \cup [(x, l)]_{fr} = X$, and (x, l) is not connected by a backward edge to another node in fr (i.e. there is no edge between (x, l) and (y, j) for any $j \in \{0, \ldots l-1\}$). Let u^+_{fr} denote a counter valuation which records the number of points of increment for each counter X,

in fr. Similarly let u^-_{fr} denote the counter valuation which records the number of points of decrement for each counter X in fr.

Now let ρ be a symbolic model of X-length l. We carry over the notations for the set of future obligations and the equivalence class for x at level i to symbolic models as well. Thus $\Diamond_\rho(x,i)$ is equal to $\Diamond_{\rho(i)}(x,0)$ and $[(x,i)]_\rho$ is $[(x,0)]_{\rho(i)}$. For $X \in \mathcal{P}^+(\{x_1,\ldots,x_k\})$, a *point of increment* for X in ρ is an equivalence class of the form $[(x,i)]_\rho$ such that $[(x,0)]_{\rho(i)}$ is a point of increment for X in the frame $\rho(i)$. Similarly, a *point of decrement* for X in ρ is an equivalence class of the form $[(x,i)]_\rho$ such that $i \geq l+1$ and $[(x,l)]$ is a point of decrement for X in the frame $\rho(i-l)$.

We can now define a canonical counter valuation sequence α along ρ, called the *counting sequence* along ρ, which counts the number of "unsatisfied" points of increments for each counter X. Let $\dot{+}$ denote the "proper addition" of integers, defined by $n \dot{+} m = \max(0, n+m)$. We define α inductively for each $X \in \mathcal{P}^+(\{x_1,\ldots,x_k\})$ and $0 \leq i < |\rho|$ as: $\alpha(0)(X) = 0$; and $\alpha(i+1)(X) = \alpha(i)(X) \dot{+} (u^+_{\rho(i)}(X) - u^-_{\rho(i+1)}(X))$.

4.2 Sequences for Satisfiable Symbolic Models

We characterize satisfiable symbolic models using their counting sequences.

Lemma 3. *A finite symbolic model ρ is satisfiable (i.e. admits a model) iff the final value of the counting sequence α along ρ has value 0 for each counter X (i.e. $\alpha(|\rho|-1)(X) = 0$) and in the last frame fr of ρ, there are no "unsatisfied"obligations – i.e. no node (x,i) in G_{fr} and variable $y \in \Diamond_{fr}(x,i)$, with no edge between (x,i) and (y,j) for $j > i$.*

An infinite symbolic model ρ is satisfiable iff the following conditions are satisfied:

(C1) There does not exist an infinite forward path p in ρ and a counter X, such that every node in the path has future obligation X, and there is a variable y in X which is never connected by a forward edge from a node in p (i.e. no node in p is connected by a forward edge to a node of the form (y,i)).

(C2) In the counting sequence along ρ, each counter X satisfies one of the conditions:

- *(a) there is a point after which the value of counter X is always zero and after which we never see a point of increment for X,*
- *(b) infinitely often we see a point of decrement for X of the form $[(x,i)]$ with $\Diamond_\rho(x,i) \subset X$, or,*
- *(c) for each $x \in X$, we infinitely often see a point of decrement for X, which is connected by a forward path to an x node (i.e a node of the form (x,i)).*

In Sect. 5 we show that we can check these conditions on counting sequences using counter automata with a decidable nonemptiness problem.

5 Decidability

We introduce a class of counter automata with a disjunctive variant of generalized Büchi acceptance condition in which, along any run, a zero test is performed at most once for each counter.

5.1 Simple Counter Automata

A simple counter automaton $\mathcal{A}$ is a tuple $\langle \Sigma, C, Q, \mathcal{F}, I, \rightarrow \rangle$ such that

- Σ is a finite alphabet, C is a finite set of counters,
- Q is a finite set of locations, $I \subseteq Q$ is the set of initial locations,
- $\mathcal{F} = \{F_0, F_1, \ldots, F_K\}$ for some $K \geq 0$ where $F_i \subseteq \mathcal{P}(Q)$ for each $i = 1 \ldots K$,
- $\rightarrow$ is a finite subset of $Q \times \mathcal{P}(C) \times \mathbb{Z}^C \times \Sigma \times Q$.

Elements of $\rightarrow$ are also denoted by $q \xrightarrow{Y,up,a} q'$ where Y is interpreted as zero tests on all counters in Y. A configuration $\langle q, \boldsymbol{c} \rangle$ is an element of $Q \times \mathbb{N}^C$ and, $\langle q, \boldsymbol{c} \rangle \rightarrow \langle q', \boldsymbol{c}' \rangle$ iff there is a transition $q \xrightarrow{Y,up,a} q'$ in $\mathcal{A}$ s.t. for $c \in Y$, $\boldsymbol{c}(c) = 0$ and for $c \in C$, $\boldsymbol{c}'(c) = \boldsymbol{c}(c) + up(c)$. As usual, a run is a sequence of configurations ruled by the transitions of $\mathcal{A}$. An infinite run is accepting iff there exists a set $F \in \mathcal{F}$ such that every set $Y \in F$ is visited infinitely often. Elements of Σ^ω labeling accepting runs define the language accepted by $\mathcal{A}$. In order to accept finite words, we suppose that $\mathcal{F}$ defines a single set of final states and a finite run is accepting iff it ends at a final state with all the counters equal to zero.

However, we require additional conditions on the control graph of $\mathcal{A}$ to be declared as simple. We require that there is a partition $\{Q_0, \ldots, Q_K\}$ of Q and corresponding sets of counters $C_0, C_1, \ldots, C_K$ with $C_0 = \emptyset$ such that $I \subseteq Q_0$ and for $i \in \{1, \ldots, K\}$, a transition from a location in Q_0 to a location in Q_i can be fired only if the counters of C_i are equal to zero and all the transitions from a location in Q_i go to another location of Q_i. Moreover every transition from a location of Q_i does not modify the value of the counters in C_i. As a consequence, when we enter in the component made of the locations of Q_i the counters in C_i are equal to zero forever. Finally, for $i \in \{0, \ldots, K\}$, $F_i \subseteq \mathcal{P}(Q_i)$. Let us summarize the conditions:

1. $Q = Q_0 \uplus \cdots \uplus Q_K$ and $I \subseteq Q_0$,
2. $\mathcal{F} = \{F_0, F_1, \ldots, F_K\}$ where each $F_i \subseteq \mathcal{P}(Q_i)$,
3. there exist $K+1$ sets of counters $C_0, \ldots, C_K \subseteq C$ with $C_0 = \emptyset$ such that the transition relation $\rightarrow \subseteq Q \times \mathcal{P}(C) \times \mathbb{Z}^C \times \Sigma \times Q$ verifies the conditions below: for all $i, i' \in \{0, \ldots, K\}$, $q \in Q_i$ and $q' \in Q_{i'}$, the transitions from q to q' are of the form $q \xrightarrow{Y,up,a} q'$ where
   (a) $i \neq i'$ implies $i = 0$ and $Y = C_{i'}$, (b) $i = i'$ implies $Y = \emptyset$,
   (c) for $c \in C_{i'}$, $up(c) = 0$.

In the sequel we consider simple counter automata with $C = \mathcal{P}^+(\{x_1, \ldots, x_k\})$, $K = 2^k - 1$ and each set F_i contains sets of states reached by decrementing the counters in C_i. Lemma 4 below states that simplicity implies decidability of the nonemptiness problem thanks to [Jan90].

Lemma 4. *The nonemptiness problem for simple counter automata is decidable.*

5.2 Automata recognizing satisfiable counting sequences

Now, we can build a simple counter automaton $\mathcal{A}_k^l$ recognizing the set of satisfiable symbolic models of X-length l. We describe the construction for the infinite case and the automaton that recognizes finite satisfiable symbolic models can be defined similarly.

The simple counter automaton $\mathcal{A}_k^l$ is defined to be the intersection of the automata $\mathcal{A}^1$ and $\mathcal{A}^2$ which check conditions (C1) and (C2) respectively. Automaton $\mathcal{A}^1$ is a Büchi automaton and is easy to define. We focus on defining the counter automaton $\mathcal{A}^2$. We define $\mathcal{A}^2 = \langle \Sigma, C, Q, F, s, \rightarrow \rangle$, where $\Sigma = \mathtt{Frame}_k^l$, $C = \mathcal{P}^+(\{x_1, \ldots, x_k\})$, $Q = \{s\} \cup \mathtt{Frame}_k^l \cup \bigcup_{Z \subseteq C} Q_{\mathcal{A}_Z}$, where $Q_{\mathcal{A}_Z}$ is the set of states of the automaton $\mathcal{A}_Z$ which we define below, $\rightarrow$ is given by

- $s \xrightarrow{\emptyset, up_0, fr} fr$
- $fr \xrightarrow{\emptyset, up, fr'} fr'$
- $fr \xrightarrow{Z, up, fr'} s_{\mathcal{A}_Z}$

where up_0 is the zero update (i.e. $up_0(X) = 0$ for each $X \in C$), $u^+_{fr}(X) \leq up(X) \leq u^+_{fr}(X) - u^-_{fr'}(X)$ for each $X \in C$, and $s_{\mathcal{A}_Z}$ is the start state of automaton $\mathcal{A}_Z$. Moreover, we require in the last rule that for every counter $X \in Z$ (i.e. every counter that is tested to zero) we have $up(X) = 0$.

The Büchi automaton $\mathcal{A}_Z$ is given by:

$$\mathcal{A}_Z = \mathcal{A}_Z^{2a} \cap \bigcap_{X \in C \setminus Z} (\mathcal{A}_X^{2b} \cup \mathcal{A}_X^{2c}) \qquad \mathcal{A}_X^{2c} = \bigcup_{x \in X} \mathcal{A}_{X,x}$$

where the automaton $\mathcal{A}_Z^{2a}$ accepts symbolic models in which there are no points of increment for any X in Z; the automaton $\mathcal{A}_X^{2b}$ checks that infinitely often there is a point of decrement for X of the form $[(x, i)]$ such that the set of future obligations of (x, i) is a strict subset of X; and the automaton $\mathcal{A}_{X,x}$ checks condition C2(c) for X and a variable $x \in X$. The automaton $\mathcal{A}_{X,x}$ is the complement of the Büchi automaton $\mathcal{B}_{X,x}$ which accepts symbolic models in which there is a point after which we never see an x-node reachable by a forward path from a point of decrement for X. The automaton $\mathcal{B}_{X,x}$ has states of the form (fr, S) where fr is a frame and S is a subset of nodes in fr. We have a transition from (fr, S) to (fr', S') iff S' is the set of nodes in fr' which are either a point of decrement for X in fr' or are connected by a forward edge to a node in S in fr'. The automaton non-deterministically moves to a second copy where it allows the above transitions only if S' does not contain a node of the form (x, i). All states in the second copy are final.

We can easily check that all the properties of simple counter automata are verified by this construction. For the finite case, $\mathcal{A}_k^l$ is similar to $\mathcal{A}^2$ above, except that a word is accepted when the run ends with all the counters equal to zero.

Lemma 5. *Let ρ be a symbolic model of X-length l. Then ρ is accepted by $\mathcal{A}_k^l$ iff ρ is satisfiable.*

We are now in position to state the main result of the paper.

Theorem 3. *Finitary and infinitary satisfiability for* $\mathrm{CLTL}^{\Diamond}$ *is decidable.*

Proof. Let ϕ be a $\mathrm{CLTL}^{\Diamond}$ formula over k variables with X-length l. Let $\mathcal{A}_\phi$ be the simple counter automaton built as the intersection of $\mathcal{A}_{\mathrm{symb}}$, $\mathcal{A}_k^l$ and $\mathcal{A}_{lc}$ that accepts sequences in which consecutive frames are one-step consistent. We can check whether the language accepted by $\mathcal{A}_\phi$ is non-empty (Lemma 4). Thus, by Theorem 2, checking the satisfiability of ϕ is decidable.

According to the acceptance condition of $\mathcal{A}_k^l$ in the finite case, finitary satisfiability reduces to the reachability problem in Petri nets. □

Theorems 2 and 3 entail that this decidability result can be extended to any extension of LTL as soon as the temporal operators as definable in MSOL, see for instance [GK03].

Corollary 1. *Finitary and infinitary satisfiability for* $\mathrm{CLTL}^{\Diamond}$ *augmented with MSOL definable temporal operators is decidable.*

5.3 A PSPACE fragment of $\mathrm{CLTL}^{\Diamond}$

In this section, we consider the fragment $\mathrm{CLTL}_1^{\Diamond}$ with a unique variable x. The models are sequences of natural numbers and the only counter in counting sequences α is $\{x\}$ (we identify $\alpha(i)(\{x\})$ with $\alpha(i)$). Given a symbolic model ρ over the alphabet $\mathtt{Frame}_1^l$ and the counting sequence α along ρ, for every $0 \leq i < |\rho|$, $\alpha(i+1) = \alpha(i) \dot{+} u_i^+ - u_{i+1}^-$ with $u_i^+, u_{i+1}^- \in \{0,1\}$. By Lemma 3, when ρ is satisfiable, in the counting sequence along ρ either the unique counter remains equal to zero after a finite number of steps or it is decremented infinitely often. Moreover, the value of the unique counter in the counting sequence is nicely bounded unlike in general with strictly more than one variable.

Lemma 6. *Let ρ be a symbolic model and α be the counting sequence along ρ. For $0 \leq i < |\rho|$, $\alpha(i) \leq l$.*

Boundedness entails the possibility to use automata without counters.

Lemma 7. *The set of satisfiable symbolic models over the alphabet $\mathtt{Frame}_1^l$ can be recognized by a standard Büchi automaton $\mathcal{A}_1^l$ for the infinite case, or by a finite-state automaton for the finite case.*

The automaton $\mathcal{A}_1^l$ has an exponential size and can be built in polynomial space in l. Checking nonemptiness for this automaton can be done in non deterministic logarithmic space which allows to establish Theorem 4 below.

Theorem 4. *Finitary and infinitary satisfiability for* $\mathrm{CLTL}_1^{\Diamond}$ *is* PSPACE*-complete.*

The models for $\mathrm{CLTL}_1^{\Diamond}$ corresponds to models of $\mathrm{LTL}^{\downarrow}(\sim, \mathsf{X}, \mathsf{U})$. Therefore, finitary and infinitary satisfiability for $\mathrm{LTL}_1^{\downarrow}(\sim, \mathsf{X}, \mathsf{X}^{-1}, \mathsf{U}, \mathsf{S})$ restricted to formulae such that the freeze operator is restricted to subformulae of the form $\downarrow_{r=x} \mathsf{XF}(r = x)$ and $\downarrow_{r=x} \mathsf{X}^i(r = x)$ is decidable in polynomial space (r is the unique register and x the unique variable).

5.4 Repeating values in the past is still decidable

In this section we explain why we can allow the constraints of the language to state properties about past repetitions of a value without loosing decidability. Let $\mathrm{CLTL}^{\Diamond,\Diamond^{-1}}$ be the extension of $\mathrm{CLTL}^{\Diamond}$ with atomic formulae of the form $x = \Diamond^{-1} y$. The satisfaction relation is extended as follows: $\sigma, i \models x = \Diamond^{-1} y$ iff there is $j > 0$ s.t. $x = \sigma(i-j)(y)$ and $0 \leq i - j$. Similarly to what is done in Section 2.1, *x diff* $\Diamond^{-1} y$ can be omitted since it can be defined from $x = \Diamond^{-1} y$ (a variant of the equivalence (1)).

In order to deal with satisfiability for $\mathrm{CLTL}^{\Diamond,\Diamond^{-1}}$ we need to extend the symbolic representation of models. In addition of the conditions(F1)–(F5) defined in Sect. 3.1, a frame *fr* has to verify the following property (the finitary case admits a similar update):

(F6) for all $i, j \in \{0, \ldots, l\}$ and $x, y \in \{x_1, \ldots x_k\}$, if $\mathsf{X}^i x = \mathsf{X}^j y$ is in *fr* then
- if $i = j$, then for every $z \in \{x_1, \ldots, x_k\}$ we have $\mathsf{X}^i(x = \Diamond^{-1} z) \in fr$ iff $\mathsf{X}^j(y = \Diamond^{-1} z) \in fr$ (we extend the notion of frames);
- if $i > j$ then $\mathsf{X}^i(x = \Diamond^{-1} y) \in fr$, and for every $z \in \{x_1, \ldots, x_k\}$, $\mathsf{X}^i(x = \Diamond^{-1} z) \in fr$ iff either $\mathsf{X}^j(y = \Diamond^{-1} z) \in fr$ or there exists $i > j' \geq j$ such that $\mathsf{X}^i x = \mathsf{X}^{j'} z$ is in *fr*.

We pose $\Diamond^{-}_{fr}(\mathsf{X}^i x) \stackrel{\text{def}}{=} \{y \mid \mathsf{X}^i(x = \Diamond^{-} y) \in fr\}$. Since we need to deal with past obligations, a counter is a pair $\langle X_p, X_f \rangle$ in $\mathcal{P}^+(\{x_1, \ldots, x_k\}) \times \mathcal{P}^+(\{x_1, \ldots, x_k\})$ where X_p is for past obligations and X_f for future obligations. We update the notion of counter valuations accordingly. A value n for $\langle X_p, X_f \rangle$ is the number of values that occurred in a past state of every variable of X_p and that have to be repeated in a future state of every variable in X_f.

We extend some earlier definitions. For an l-frame *fr* and counter $\langle X_p, X_f \rangle$, we define a *point of increment* for $\langle X_p, X_f \rangle$ in *fr* to be an equivalence class of the form $[(x,0)]_{fr}$ such that $\Diamond_{fr}(x,0) = X_f$, $(x,0)$ is not connected by a forward edge to a node in *fr and* $[(x,0)]_{fr} \cup \Diamond^{-1}_{fr}(x,0) = X_p$. A *point of decrement* for $\langle X_p, X_f \rangle$ in *fr* is defined to be an equivalence class of the form $[(x,l)]_{fr}$ such that $\Diamond_{fr}(x,l) \cup [(x,l)]_{fr} = X_f$, (x,l) is not connected by a backward edge to another node in *fr and* $\Diamond^{-1}(x,l) = X_p$. Let u^{+}_{fr} denote a counter valuation which records the number of points of increment for each counter $\langle X_p, X_f \rangle$, in *fr*. Similarly let u^{-}_{fr} denote the counter valuation which records the number of points of decrement for each counter $\langle X_p, X_f \rangle$ in *fr*. We can now define a canonical counter valuation sequence α along ρ, called the *counting sequence* along ρ, which counts the number of "unsatisfied" points of increments for each counter $\langle X_p, X_f \rangle$ with $X_p \neq \emptyset$. We define α inductively: $\alpha(0)(\langle X_p, X_f \rangle) = 0$ and for $0 \leq i < |\rho|$,

$\alpha(i+1)(\langle X_p, X_f\rangle) = \alpha(i)(\langle X_p, X_f\rangle) + (u^+_{\rho(i)}(\langle X_p, X_f\rangle) - u^-_{\rho(i+1)}(\langle X_p, X_f\rangle))$. Note that decrementations are this time compulsory and we allow $\alpha(i)(\langle X_p, X_f\rangle)$ to be negative (but not in the acceptance condition). Though we need more counters, dealing with past repeating values, does not introduce real complications. This is analogous to the passage from LTL to LTL with past-time operators since past is finite and information about past can be accumulated smoothly.

Lemma 8. *A symbolic model ρ for the logic* $\mathrm{CLTL}^{\Diamond,\Diamond^{-1}}$ *is satisfiable iff the counting sequence along ρ satisfies the conditions from Lemma 3 for the future part of the counters and for $0 \leq i < |\rho|$, $\alpha(i)(\langle X_p, X_f\rangle) \geq 0$.*

The proof is similar to the proof of Lemma 3. We just need more registers to store the different values that have to be repeated.

As a consequence, we can easily update the construction of $\mathcal{A}_\phi$ in order to deal with past repeating values. The definition of the automata $\mathcal{A}_{\mathrm{symb}}$ and $\mathcal{A}^l_k$ are just extended by considering the new definition for frames. It is also important to observe that the automaton $\mathcal{A}_\phi$ obtained by synchronization of these automata still belongs to the class of simple counter automata and the decidability result also holds for $\mathrm{CLTL}^{\Diamond,\Diamond^{-1}}$ satisfiability problem.

Theorem 5. *Finitary and infinitary satisfiability for* $\mathrm{CLTL}^{\Diamond,\Diamond^{-1}}$ *[resp.* $\mathrm{CLTL}_1^{\Diamond,\Diamond^{-1}}$*] is decidable [resp.* PSPACE*-complete].*

6 Concluding Remarks

We have shown that satisfiability for $\mathrm{CLTL}^{\Diamond}$ with operators in $\{\mathsf{X}, \mathsf{X}^{-1}, \mathsf{S}, \mathsf{U}\}$ is decidable by reduction into the verification of fairness properties in Petri nets [Jan90]. The proof is uniform for the finitary and infinitary cases and it can be extended to atomic constraints of the form $x = \Diamond^{-1} y$ and to any set of MSOL definable temporal operators. Moreover, satisfiability for $\mathrm{CLTL}^{\Diamond}$ restricted to one variable is shown PSPACE-complete. Hence, we have defined and studied a well-designed decidable fragment of LTL with the freeze quantifier answering some question from [WZ00] and circumventing some undecidability results from [DL06]. Finally, as done also in [DL06,Laz06,BMS$^+$06], we show relationships between fragments of LTL with freeze and counter automata.

The main question left open by our work is the complexity of satisfiability for $\mathrm{CLTL}^{\Diamond}$ and more precisely we do not know whether $\mathrm{CLTL}^{\Diamond}$ satisfiability has elementary complexity. Similarly, are there natural fragments of $\mathrm{CLTL}^{\Diamond}$ that are of lower complexity, for instance the one involved in Theorem 1? Another promising extension consists in considering other concrete domains as $\langle \mathbb{R}, <, = \rangle$ and to allow atomic formulae of the form $x < \Diamond y$. The decidability status of such a variant is still open.

Acknowledgements: We are grateful to Petr Jančar (TU Ostrava) for pointing us to [Jan90] in order to solve the nonemptiness problem for simple counter automata and for suggesting the proof of Lemma 4 and Ranko Lazić (U. of Warwick) for remarks about a preliminary version.

References

[ABM01] C. Areces, P. Blackburn, and M. Marx. Hybrid logics: characterization, interpolation and complexity. *JSL*, 66(3):977–1010, 2001.

[AH94] R. Alur and Th. Henzinger. A really temporal logic. *JACM*, 41(1):181–204, 1994.

[BMS+06] M. Bojańczyk, A. Muscholl, Th. Schwentick, L. Segoufin, and C. David. Two-variable logic on words with data. In *LICS'06*, pages 7–16. IEEE, 2006.

[DD07] S. Demri and D. D'Souza. An automata-theoretic approach to constraint LTL. *I & C*, 205(3):380–415, 2007.

[DDG07] S. Demri, D. D'Souza, and R. Gascon. A decidable temporal logic of repeating values. Technical report, LSV, 2007.

[DL06] S. Demri and R. Lazić. LTL with the freeze quantifier and register automata. In *LICS*, pages 17–26. IEEE, 2006.

[DLN07] S. Demri, R. Lazić, and D. Nowak. On the freeze quantifier in constraint LTL: decidability and complexity. *I & C*, 205(1):2–24, 2007.

[Esp94] J. Esparza. On the decidability of model checking for several μ-calculi and Petri nets. In *CAAP'94*, volume 787 of *LNCS*, pages 115–129. Springer, 1994.

[Fit02] M. Fitting. Modal logic between propositional and first-order. *JLC*, 12(6):1017–1026, 2002.

[GK03] P. Gastin and D. Kuske. Satisfiability and model checking for MSO-definable temporal logics are in PSPACE. In *CONCUR'03*, volume 2761 of *LNCS*, pages 222–236. Springer, 2003.

[Gor96] V. Goranko. Hierarchies of modal and temporal logics with references pointers. *Journal of Logic, Language, and Information*, 5:1–24, 1996.

[Hen90] Th. Henzinger. Half-order modal logic: how to prove real-time properties. In *PODC'90*, pages 281–296. ACM, 1990.

[HR89] R.R. Howell and L.E. Rosier. Problems concerning fairness and temporal logic for conflict-free Petri nets. *TCS*, 64:305–329, 1989.

[Jan90] P. Jančar. Decidability of a temporal logic problem for Petri nets. *TCS*, 74(1):71–93, 1990.

[KV06] O. Kupferman and M. Vardi. Memoryful Branching-Time Logic. In *LICS'06*, pages 265–274. IEEE, 2006.

[Laz06] R. Lazić. Safely freezing LTL. In *FST&TCS'06*, volume 4337, pages 381–392. LNCS, 2006.

[LMS02] F. Laroussinie, N. Markey, and Ph. Schnoebelen. Temporal logic with forgettable past. In *LICS'02*, pages 383–392. IEEE, 2002.

[LP05] A. Lisitsa and I. Potapov. Temporal logic with predicate λ-abstraction. In *TIME'05*, pages 147–155. IEEE, 2005.

[Rab06] A. Rabinovich. Decidability and expressive power of real time logics. In *FORMATS'06*, volume 4202 of *LNCS*, page 32. Springer, 2006. Invited talk.

[Seg06] L. Segoufin. Automata and logics for words and trees over an infinite alphabet. In *CSL'06*, volume 4207 of *LNCS*, pages 41–57. Springer, 2006.

[tCF05] B. ten Cate and M. Franceschet. On the complexity of hybrid logics with binders. In *CSL'05*, volume 3634 of *LNCS*, pages 339–354. Springer, 2005.

[VW94] M. Vardi and P. Wolper. Reasoning about infinite computations. *I & C*, 115:1–37, 1994.

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