The light side of Interval Temporal Logic: the Bernays-Schönfinkel's fragment of CDT

Davide Bresolin*, Dario Della Monica†‡, Angelo Montanari‡, and Guido Sciavicco§¶
*Dept. of Computer Science, University of Verona, Italy
Email: davide.bresolin@univr.it
†Dept. of Computer Science, University of Salerno, Italy
Email: ddellamonica@unisa.it
‡Dept. of Mathematics and Computer Science, University of Udine, Italy
Email: angelo.montanari@uniud.it
§Dept. of Information and Communication Engineering, University of Murcia, Spain
Email: guido@um.es
¶UIST Ohrid, Macedonia

***Abstract*—Decidability and complexity of the satisfiability problem for the logics of time intervals have been extensively studied in the last years. Even though most interval logics turn out to be undecidable, meaningful exceptions exist, such as the logics of temporal neighborhood and (some of) the logics of the subinterval relation. In this paper, we explore a different path to decidability: instead of restricting the set of modalities or imposing suitable semantic restrictions, we take the most expressive interval temporal logic studied so far, namely, Venema's CDT, and we suitably limit the nesting degree of modalities. The decidability of the satisfiability problem for the resulting CDT fragment is proved by embedding it into a well-known decidable prefix quantifier class of first-order logic, namely, the Bernays-Schönfinkel's class. In addition, we show that such a fragment is in fact NP-complete (the Bernays-Schönfinkel's class is NEXPTIME-complete), and that any natural extension of it is undecidable.**

***Keywords*-interval temporal logic; tableaux methods; decidability; complexity.**

I. Introduction

In the last years, the study of interval-based temporal reasoning and logic has been very intensive. Since the seminal work by Halpern and Shoham on the interval logic HS [1], that features a modal operator for each Allen's relation [2] between a pair of intervals over a linear order, and Venema's work on the quite expressive interval logic CDT [3], a series of papers on interval temporal logics has been published, e.g., [4], [5], [6], [7], [8], [9], [10], [11]. In the context of this research line, the problem of classifying all "natural", genuinely interval-based logics (that is, logics that take into consideration all intervals over the given linear order, and assume no projection principle [12]) with respect to their expressive and computational power has been systematically studied and almost completely solved.

One of the most significant decidable fragments of HS is Propositional Neighborhood Logic (PNL for short), whose two modalities correspond to Allen's relations *meets* and *met by*. PNL has been introduced in [13], and further studied in [14], where it has been shown to be expressively complete with respect to the two-variable fragment of first-order logic interpreted over a number of classes of linearly-ordered sets. Recently, some decidable extensions of PNL have been identified. In [15], Montanari et al. have proved that the pair of modalities corresponding to Allen's relations *starts* and *started by* (or, equivalently, *ends* and *ended by*) can be added to PNL preserving decidability over finite linear orders, while a decidable metric extension of PNL over natural numbers has been investigated in [16]. Unfortunately, it is possible to show that the addition of quite simple hybrid (resp., first-order) constructs to PNL immediately leads to undecidability [17], (resp., [18]). The D fragment of HS, which features a single modality for the Allen's relation *during*, is a meaningful example of how easy it is to fall into undecidability: D is decidable over dense linear orders and undecidable over finite and (weakly) discrete linear orders. Moreover, the extension of D with modalities for the inverse relation *contains*, the pair of relations *starts* and *started by* (or, equivalently, *ends*and *ended by*), and the pair of relations *before* and *later* is still decidable over dense linear orders [19], but the extension of D with, for instance, any of the PNL modalities turns out to be undecidable over all interesting classes of linear orders, including that of dense linear orders [20].

The case with classical first-order logic is somehow similar. Ever since it has been shown that the satisfiability problem for the full language is undecidable, a great effort has been made in order to identify more and more expressive decidable fragments. At least three different strategies have been explored: (i) to limit the number of variables of the language, (ii) to limit the allowed types of formula by relativizing quantification (*guarded fragments*), and (iii) to limit the structure and the shape of the quantifier prefix. First-order logics with a limited number of variables have been already explored in connection with interval temporal logics; most notably, as already mentioned, the equivalence in expressive power between the two-variable fragment of first-order logic over linear orders (shown to be NEXPTIME-complete in [21]) has made it possible to prove decidability of PNL before specific decision procedures were devised for it [14]. Guarded fragments of first-order logics have been shown to be extremely useful to understand and to explain the good computational properties of modal logics [22]. However, to the best of our knowledge, they turned out to be almost useless to tackle interval-based temporal logics, the main reason being the fact that transitive guards, necessary to force the linearity of the structures, preserve decidability only when at most two variables are allowed, while interval properties (when intervals are interpreted as pairs of points) are mostly three-variable.

In this paper, we explore a different path to decidability, based on the third strategy: we analyze the relationships between interval-based temporal logics and quantifier prefix decidable first-order logics. The decidability of the latter family of logics does not depend only on the shape of the quantifier prefixes, but also on the number and the arity of predicate and function symbols that are allowed in the formulas, and on the presence/absence of equality. Seven intrinsically different decidable classes have been identified in the literature. A comprehensive survey on prefix classes of first-order logic can be found in [23]. For the purpose of this paper, we focus our attention on the prefix vocabulary class identified in 1928 by Bernays and Schönfinkel [23] (also known as Bernays, Schönfinkel, and Ramsey's class). It features all and only formulas in prenex form, where the quantifier prefix is of the form $\exists x_1 \ldots \exists x_n \forall y_1 \ldots \forall y_m$ and predicate symbols of any arity, and, possibly, equality (but no function symbols), may occur in the matrix. We will consider the most expressive (undecidable) interval temporal logic studied so far, namely, Venema's CDT [3], and we will define a syntactically-defined fragment of it, called CDT_{BS}, whose standard translation fits into the above-mentioned prefix vocabulary class. It is well known that Bernays-Schönfinkel's fragment of first-order logic is expressive enough to model a linear order devoid of specific properties like discreteness or denseness. Simpler frame properties commonly studied in interval temporal logic literature, such as boundedness, can also be expressed. Moreover, we will take into consideration a non-terminating tableau-based deduction system for CDT developed in [24], and we will show that, when tailored to CDT_{BS}, it actually terminates. As a by-product, we will prove that the satisfiability problem for CDT_{BS} is NP-complete, in contrast with that of the Bernays-Schönfinkel's fragment, which is NEXPTIME-complete. An example of application of CDT_{BS} in the area of robot motion planning will also be given. We will conclude the paper by showing that any natural extension of CDT_{BS} immediately yields undecidability.

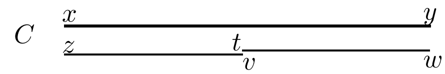

Figure 1. The ternary relation *chop*.

II. Preliminaries

The Bernays-Schönfinkel's prefix vocabulary class is defined by all and only those first-order formulas built using any relational symbol of any arity and such that they can be put in prenex form by using a quantifier prefix of the form $\exists\vec{x}\forall\vec{y}$, where $\vec{x} = x_1 \ldots x_n$ and $\vec{y} = y_1 \ldots y_m$ are (possibly empty) vectors of first-order variables. Throughout the paper, we denote this class by FO_{BS}, and its extension with equality by $\text{FO}_{\text{BS}}[=]$. It is well known that the satisfiability problem for both classes is NEXPTIME-complete [23]. It is worth noticing that, in general, the class is not closed by negation; nevertheless, since all our formulas can be thought of as sentences (free variables can be existentially quantified), the class turns out to be closed under conjunctions and disjunctions.

Interval temporal logics are usually interpreted over a linearly ordered set $\mathbb{D} = \langle D, < \rangle$. In this setting, an *interval* on $\mathbb{D}$ is an ordered pair $[d_i, d_j]$ with $d_i \leq d_j$ (this is usually referred to as the *non-strict* semantics, in contrast with the *strict* one that excludes intervals of the form $[d_i, d_i]$). The set of all intervals on $\mathbb{D}$ is denoted by $\mathbb{I}(\mathbb{D})$. The variety of all possible relations between any two intervals has been studied by Allen [2], who identified 12 different binary ordering relations plus the equality. Halpern and Shoham's HS is one of the first interval logics proposed in the literature, and it can be seen as the modal logic that features exactly one modalility for each Allen's relation. HS is undecidable over most classes of linearly ordered sets [1]. In [3], the ternary relation *chop*, depicted in Fig. 1, has been considered. The corresponding binary modality C, together with its two conjugated D and T, and the modal constant π for point-intervals, defines the interval logic CDT, that turns out to be undecidable whenever HS is, as the former is strictly more expressive than the latter. Recently, Hodkinson et al. studied in detail the properties of the three one-modality fragments $C, D,$ and T of CDT, and they showed that all of them are undecidable [25].

III. The logic CDT_{BS} over all linear orders and its standard translation

Formulas of CDT are built over a set of propositional letters $\mathcal{AP} = \{p, q, \ldots\}$, the Boolean connectives $\neg, \vee$, the three binary modalities $C, D,$ and T, and a modal constant π, according to the following abstract grammar [3]:

$$\varphi ::= p \mid \pi \mid \neg\varphi \mid \varphi \vee \varphi \mid \varphi\, C\, \varphi \mid \varphi\, D\, \varphi \mid \varphi\, T\, \varphi.$$

The other Boolean connectives can be thought of as shortcuts, as usual. $R \in \{C, D, T\}$. Universal modalities have not a specific notation, and can be defined using negation. The semantics of CDT-formulas can be given in terms of concrete *models* of the form $M = \langle \mathbb{D}, V \rangle$, where $V : \mathcal{AP} \to 2^{\mathbb{I}(\mathbb{D})}$ is a *valuation function*:

- $M, [d_i, d_j] \Vdash p$ if and only if $[d_i, d_j] \in V(p)$,
- $M, [d_i, d_j] \Vdash \pi$ if and only if $d_i = d_j$,
- $M, [d_i, d_j] \Vdash \neg\varphi$ if and only if $M, [d_i, d_j] \nVdash \varphi$,
- $M, [d_i, d_j] \Vdash \varphi \vee \psi$ if and only if $M, [d_i, d_j] \Vdash \varphi$ or $M, [d_i, d_j] \Vdash \psi$,
- $M, [d_i, d_j] \Vdash \varphi C \psi$ if and only if there exists $d_i \leq d_k \leq d_j$ such that $M, [d_i, d_k] \Vdash \varphi$ and that $M, [d_k, d_j] \Vdash \psi$,
- $M, [d_i, d_j] \Vdash \varphi \, D \, \psi$ if and only if there exists $d_k \leq d_i$ such that $M, [d_k, d_i] \Vdash \varphi$ and that $M, [d_k, d_j] \Vdash \psi$,
- $M, [d_i, d_j] \Vdash \varphi \, T \, \psi$ if and only if there exists $d_k \geq d_j$ such that $M, [d_j, d_k] \Vdash \varphi$ and that $M, [d_i, d_k] \Vdash \psi$.

The *standard translation* is the classical tool used to express the semantics of a modal/temporal formula in first-order logic. The clauses for the standard translation $ST(\varphi)[x, y]$, over a pair of points x, y, are defined as follows:

- $ST(p)[x, y] = p(x, y)$,
- $ST(\pi)[x, y] = (x = y)$,
- $ST(\neg\varphi)[x, y] = \neg ST(\varphi)[x, y]$,
- $ST(\varphi \vee \psi)[x, y] = ST(\varphi)[x, y] \vee ST(\psi)[x, y]$,
- $ST(\varphi \; C \; \psi)[x, y] = \exists z(x \leq z \leq y \wedge ST(\varphi)[x, z] \wedge ST(\psi)[z, y])$,
- $ST(\varphi \; D \; \psi)[x, y] = \exists z(z \leq x \wedge ST(\varphi)[z, x] \wedge ST(\psi)[z, y])$,
- $ST(\varphi \; T \; \psi)[x, y] = \exists z(y \leq z \wedge ST(\varphi)[y, z] \wedge ST(\psi)[x, z])$.

In such a way, the satisfiability problem for a generic modal logic is reduced to a first-order satisfiability problem: a CDT-formula φ is satisfiable if and only if there exist a model M and a pair of points x, y such that the standard translation of φ, evaluated on x, y, is (first-order) satisfiable. Now, we can ask ourselves the following question: which CDT-formulas are such that their satisfiability problem is a first-order problem in Bernays-Schönfinkel's class? To answer this question, we devise an abstract grammar that produces only CDT-formulas suitably limited in the nesting of modalities:

$$\begin{aligned} \varphi ::= \; & \pi \mid \neg\pi \mid p \mid \neg p \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \\ & \varphi \, C \, \varphi \mid \varphi \, D \, \varphi \mid \varphi \, T \, \varphi \mid \\ & \neg(\varphi_\exists \, C \, \varphi_\exists) \mid \neg(\varphi_\exists \, D \, \varphi_\exists) \mid \neg(\varphi_\exists \, T \, \varphi_\exists) \end{aligned} \tag{1}$$

$$\begin{aligned} \varphi_\exists ::= \; & \pi \mid \neg\pi \mid p \mid \neg p \mid \varphi_\exists \wedge \varphi_\exists \mid \varphi_\exists \vee \varphi_\exists \mid \\ & \varphi_\exists \, C \, \varphi_\exists \mid \varphi_\exists \, D \, \varphi_\exists \mid \varphi_\exists \, T \, \varphi_\exists \end{aligned} \tag{2}$$

The above grammar generates a fragment of CDT, that we call CDT_{BS}, and it is characterized by the fact that the modalities C, D, and T can occur in the scope of *at most one negation*. By exploiting this limitation, we will show that the standard translation of every formula of CDT_{BS} is in Bernays-Schönfinkel's first-order class. It is easy to see that the syntactic limitations of CDT_{BS} do not prevent the logic to simulate every modality corresponding to an Allen's relation. For instance, we have that: $\langle B \rangle \varphi \equiv \varphi \; C \; \neg\pi$, and similarly for the other cases [3].

In order to prove our main theorem, we need the following observation: the linear ordering relation $<$ can be axiomatized in Bernays-Schönfinkel's class with equality [23]. Let Φ be the conjunction of the following four classical properties:

1) $\forall x \neg(x < x)$;
2) $\forall x, y(x < y \to \neg y < x)$;
3) $\forall x, y, z(x < y \wedge y < z \to x < z)$;
4) $\forall x, y(x = y \vee x < y \vee y < x)$.

It is immediate to see that Φ is in $\text{FO}_{\text{BS}}[=]$. From now on, for a first-order formula α, we write $\alpha(x, y)$ to indicate that x, y are among the free variables of α.

Lemma 1. *For every formula φ of CDT_{BS}, its standard translation is a formula of the form $\exists \vec{z} \forall \vec{w} \, \alpha(x, y)$, where $\alpha(x, y)$ is quantifier-free and $x, y \notin \vec{z} \cup \vec{w}$.*

Proof: We proceed by structural induction. We start with the set of formulas generated by the sub-grammar for $\varphi_\exists$, and we show that their standard translations are of the form $\exists \vec{z} \, \alpha(x, y)$ with $x, y \notin \vec{z}$. As base case, assume $\varphi_\exists = p$ for some propositional letter p. Then $ST(p)[x, y] = p(x, y)$ and the claim trivially holds. The case $\neg p$ is similar, as well as the cases for π and $\neg\pi$. Consider now the case of $\varphi_\exists \wedge \varphi'_\exists$. By definition, $ST(\varphi_\exists \wedge \varphi'_\exists)[x, y] = ST(\varphi_\exists)[x, y] \wedge ST(\varphi'_\exists)[x, y]$. By the inductive hypothesis, we have that $ST(\varphi_\exists)[x, y] = \exists \vec{z} \alpha(x, y)$ and $ST(\varphi'_\exists)[x, y] = \exists \vec{w} \beta(x, y)$, for some α and β quantifier-free and such that $x, y \notin \vec{z} \cup \vec{w}$. We can assume that $\vec{z} \cap \vec{w} = \emptyset$ (otherwise, we make a suitable variables substitution), and therefore we have that $ST(\varphi_\exists \wedge \varphi'_\exists)[x, y] = \exists \vec{z} \vec{w}(\alpha(x, y) \wedge \beta(x, y))$. The case of disjunction is similar. Consider now the case of $\varphi_\exists C \varphi'_\exists$. By definition, $ST(\varphi_\exists \; C \; \varphi'_\exists)[x, y] = \exists z(x \leq z \leq y \wedge ST(\varphi_\exists)[x, z] \wedge ST(\varphi'_\exists)[z, y])$. By inductive hypothesis, we have that $ST(\varphi_\exists)[x, z] = \exists \vec{w} \alpha(x, z)$ and $ST(\varphi'_\exists)[z, y] = \exists \vec{t} \beta(z, y)$, with α and β quantifier-free and $x, y, z \notin \vec{w} \cup \vec{t}$. As in the previous case, we can assume $\vec{w} \cap \vec{t} = \emptyset$ and we can conclude that $ST(\varphi_\exists \; C \; \varphi'_\exists)[x, y] = \exists z(x \leq z \leq y \wedge \exists \vec{w} \alpha(x, y) \wedge \exists \vec{t} \beta(z, y)) = \exists z \exists \vec{w} \exists \vec{t}(x \leq z \leq y \wedge \alpha(x, y) \wedge \beta(z, y)$. The other two cases are dealt with in a similar way.

We can now consider a generic formula generated by the grammar. The only interesting cases are those corresponding to negation of modalities. Therefore, consider the case of $\neg(\varphi_\exists \; C \; \varphi'_\exists)$. By definition, we have that $ST(\neg(\varphi_\exists \; C \; \varphi'_\exists))[x, y] = \neg \exists z(x \leq z \leq y \wedge ST(\varphi_\exists)[x, z] \wedge ST(\varphi'_\exists)[z, y])$. By the previous argument, we can assume that $ST(\neg(\varphi_\exists \, C \, \varphi'_\exists))[x,y] = \neg\exists z(x \le z \le y \wedge \exists\vec{w}\alpha(x,z) \wedge \exists\vec{t}\beta(z,y))$, that is equivalent to the formula in prenex form $\forall z \forall\vec{w}\forall\vec{t}(\neg(x \le z \le y) \vee \neg\alpha(x,z) \vee \neg\beta(z,y))$. The two remaining cases can be proved in a similar way. ■

Theorem 1. *The satisfiability problem for* CDT$_{\mathrm{BS}}$ *in the class of all linear orders is decidable.*

Proof: By the above lemma, if φ is a CDT$_{\mathrm{BS}}$-formula, then $ST(\varphi)[x,y]$ is such that the formula $\exists x,y ST(\varphi)[x,y]$ is in the Bernays-Schönfinkel's class. Therefore, satisfiability of φ can be reduced to satisfiability of $\Phi \wedge \exists x, y ST(\varphi)[x,y]$, where, possibly, we have changed the variables in such a way that Φ and $ST(\varphi)[x,y]$ have no variables in common. Since the satisfiability problem for FO$_{\mathrm{BS}}[=]$ is decidable, decidability of CDT$_{\mathrm{BS}}$ trivially follows. ■

IV. A TABLEAU METHOD FOR CDT$_{\mathrm{BS}}$

In [24], Goranko et al. propose a tableau method for BCDT$^+$, a generalization of Venema's CDT logic to partial orders with linear interval property (i.e., partial orders where every interval is supposed to be linear). Since the considered logic is undecidable, the method is not guaranteed to terminate, and it is only a semi-decision procedure. In this section we show how to tailor it to CDT$_{\mathrm{BS}}$, and how to exploit the syntactic restriction of this logic to guarantee termination and obtain an NP decision procedure for it; the original rules for modal operators are adapted to restrict the search for possible models to linear orders only.

The tableau construction generates a tree, whose nodes are decorated with $\langle\psi, [d_i, d_j], \mathbb{D}, p, u\rangle$, where $\mathbb{D} = \langle D, <\rangle$ is a finite partial order (with linear interval property), $[d_i, d_j] \in \mathbb{I}(\mathbb{D})$, $p \in \{0, 1\}$, and u is a *local flag function* which associates the values 0 or 1 with every branch B containing the node. Intuitively, the value 1 for a node n in a branch B means that n can be expanded on B. The auxiliary flag p is added to simplify the termination and complexity proofs; $p = 0$ if ψ has the syntax ((2)) and $p = 1$ otherwise. If B is a branch, then $B \cdot n$ is the result of expanding B with the node n, while $B \cdot n_1 | \ldots | n_k$ is the result of expanding B with k immediate successor nodes $n_1, \ldots, n_k$. With $\mathbb{D}_B$ we denote the finite partial ordering in the decoration of the leaf of B. Since in CDT$_{\mathrm{BS}}$ negation can occur only in front of propositional letters or modal operators, we need to introduce the notion of *dual formula* of a formula φ, denoted by $\overline{\varphi}$ and inductively defined as follows:

- $\overline{p} = \neg p$ and $\overline{\neg p} = p$, for every $p \in \mathcal{AP} \cup \{\pi\}$;
- $\overline{\varphi \vee \psi} = \overline{\varphi} \wedge \overline{\psi}$;
- $\overline{\varphi \wedge \psi} = \overline{\varphi} \vee \overline{\psi}$;
- $\overline{\varphi \, R \, \psi} = \neg(\varphi \, R \, \psi)$, for $R \in \{C, D, T\}$;
- $\overline{\neg(\varphi \, R \, \psi)} = \varphi \, R \, \psi$, for $R \in \{C, D, T\}$.

Notice that the dual of a generic formula of CDT$_{\mathrm{BS}}$ does not necessarily belong to CDT$_{\mathrm{BS}}$: this is the case, for instance, of the formula $pC\neg(qCr)$. However, it can be easily proved (by induction on the above production rules) that the dual of a formula generated by the sub-grammar for $\varphi_\exists$ is always a formula of CDT$_{\mathrm{BS}}$. This observation will be crucial for the correctness of the tableau method.

The construction of a tableau for CDT$_{\mathrm{BS}}$ starts from a three-node *initial tree* built up from an empty-decorated root and two leaves with decorations $\langle\varphi, [d_0, d_0], \{d_0\}, 1, 1\rangle$ and $\langle\varphi, [d_0, d_1], \{d_0 < d_1\}, 1, 1\rangle$, respectively, where φ is the formula to check for satisfiability. The procedure exploits a set of expansion rules to add new nodes to the tree. Notice that the rules and other concepts are very similar to [24], which we refer to for further explanations.

Definition 1. *Given a tree $\mathcal{T}$, a branch B in $\mathcal{T}$, and a node $n \in B$ decorated with $\langle\psi, [d_i, d_j], \mathbb{D}, p_n, u_n\rangle$ such that $u_n(B) = 1$, the* branch expansion rule *for B and n is defined as follows. In all considered cases, $u_{n'}(B') = 1$ for all new nodes n' and branches B'.*

- **R1** *If $\psi = \xi_0 \wedge \xi_1$, then expand B to $B \cdot n_0 \cdot n_1$, where n_0 is decorated with $\langle\xi_0, [d_i, d_j], \mathbb{D}_B, p_n, u_{n_0}\rangle$ and n_1 is decorated with $\langle\xi_1, [d_i, d_j], \mathbb{D}_B, p_n, u_{n_1}\rangle$.*
- **R2** *If $\psi = \xi_0 \vee \xi_1$, then expand B to $B \cdot n_0 \mid n_1$, where n_0 is decorated with $\langle\xi_0, [d_i, d_j], \mathbb{D}_B, p_n, u_{n_0}\rangle$ and n_1 is decorated with $\langle\xi_1, [d_i, d_j], \mathbb{D}_B, p_n, u_{n_1}\rangle$.*
- **R3** *If $\psi = \neg(\xi_0 \, C \, \xi_1)$ and d is an element of $\mathbb{D}_B$ such that $d_i \le d \le d_j$ and d has not been used yet to expand n in B, then expand B to $B \cdot n_0 | n_1$, where n_0 is decorated with $\langle\overline{\xi_0}, [d_i, d], \mathbb{D}_B, 0, u_{n_0}\rangle$ and n_1 is decorated with $\langle\overline{\xi_1}, [d, d_j], \mathbb{D}_B, 0, u_{n_1}\rangle$.*
- **R4** *If $\psi = \neg(\xi_0 \, D \, \xi_1)$, the rule is analogous to **R3**.*
- **R5** *If $\psi = \neg(\xi_0 \, T \, \xi_1)$, the rule is analogous to **R3**.*
- **R6** *If $\psi = \xi_0 \, C \, \xi_1$, then expand the branch B to $B \cdot (n_i \cdot m_i) | \ldots | (n_j \cdot m_j) | (n'_i \cdot m'_i) | \ldots | (n'_{j-1} \cdot m'_{j-1})$, where:*
  - *a) for all $i \le k \le j$, n_k is decorated with $\langle\xi_0, [d_i, d_k], \mathbb{D}_B, p_n, u_{n_k}\rangle$ and m_k is decorated with $\langle\xi_1, [d_k, d_j], \mathbb{D}_B, p_n, u_{m_k}\rangle$;*
  - *b) for all $i \le k \le j-1$, $\mathbb{D}_k$ is the linear ordering obtained from $\mathbb{D}_B$ by inserting a new element d between d_k and d_{k+1}, n'_k is decorated with $\langle\xi_0, [d_i, d], \mathbb{D}_k, p_n, u_{n'_k}\rangle$ and m'_k is decorated with $\langle\xi_1, [d, d_j], \mathbb{D}_k, p_n, u_{m'_k}\rangle$.*
- **R7** *If $\psi = \xi_0 \, D \, \xi_1$, the rule is analogous to **R6**.*
- **R8** *If $\psi = \xi_0 \, T \, \xi_1$, the rule is analogous to **R6**.*

Finally, let $u_n(B) = 0$ and, for every node $m \ne n$ in B and any branch B' extending B, let $u_m(B') = u_m(B)$, while for every branch B' extending B, $u_n(B') = 0$, unless $\psi = \neg(\xi_0 C \xi_1)$, $\psi = \neg(\xi_0 D \xi_1)$, or $\psi = \neg(\xi_0 T \xi_1)$ (in such cases $u_n(B') = 1$).

We briefly explain the expansion rules for $\xi_0 \, C \, \xi_1$ and $\neg(\xi_0 C \xi_1)$ (similar considerations can be made for the cases of the temporal operators D and T). The rule for the formula

$\xi_0 \; C \; \xi_1$ deals with two possible cases: either there exists $d_k \in \mathbb{D}_B$ such that ξ_0 holds over $[d_i, d_k]$ and ξ_1 holds over $[d_k, d_j]$, or such an element must be added to $\mathbb{D}_B$. On the converse, the formula $\neg(\xi_0 \; C \; \xi_1)$ states that, for all $d_i \leq d \leq d_j$, either $\overline{\xi_0}$ holds over $[d_i, d]$ or $\overline{\xi_1}$ holds over $[d, d_j]$. The expansion rule imposes such a condition for a single element d and keeps the flag equal to 1. In this way, all elements of $\mathbb{D}_B$ are eventually considered, including those elements that will be added in some subsequent steps of the tableau construction.

Intuitively, a branch is *closed* if there are two nodes n, n' in B such that n is decorated with $\langle p, [d_i, d_j], \mathbb{D}, p_n, u_n \rangle$ and n' is decorated with $\langle \neg p, [d_i, d_j], \mathbb{D}', p_{n'}, u_{n'} \rangle$, for some $p \in \mathcal{AP}$, or if π (resp., $\neg \pi$) is in the decoration of a node where $d_i \neq d_j$ (resp., $d_i = d_j$). Otherwise, the branch is *open*. The expansion strategy for the tableau expands a branch B only if it is open and it applies the branch expansion rule to the closest-to-the-root node for which the branch expansion rule is applicable (i.e., whose local flag is 1) and generates at least one node with a new decoration. In order to establish the satisfiability of a $\mathrm{CDT}_{\mathrm{BS}}$-formula φ, we start with the initial tableau for φ, and we keep expanding it until it is possible; if at least one branch of the resulting tree is open, then it is satisfiable, otherwise it is not. It is easy to adapt the result in [24] to obtain the following theorem.

Theorem 2. *The tableau method for* $\mathrm{CDT}_{\mathrm{BS}}$ *is sound and complete.*

To prove that the method is terminating, and to establish its computational complexity, we need to fix some preliminary results. First of all, we define the *counting function* on B as follows:

$$\mathrm{Count}(B) = \sum_{n \in B} |\psi_n| \cdot p_n \cdot u_n(B),$$

where ψ_n and p_n are the formula and the p-flag in the decoration of n, respectively. The following lemma proves that $\mathrm{Count}(B)$ is non-increasing.

Lemma 2. *Let B be a branch in a tableau for φ, and let B' be an expansion of B that respects the expansion strategy. Then* $\mathrm{Count}(B') \leq \mathrm{Count}(B)$. *Moreover, if the expansion strategy applied either* ***R1***, ***R2***, ***R6***, ***R7***, *or* ***R8*** *rule to a node n such that $p_n = 1$, then* $\mathrm{Count}(B') < \mathrm{Count}(B)$.

Proof: Let $\mathcal{T}$ be a tableau for φ, B a branch on it and n the closest to the root node for which the branch expansion rule is applicable. Let B' a branch obtained by applying the expansion strategy on B. We proceed by induction on the expansion rule applied to n. The missing rules are similar to other cases, and skipped.

- Rule **R1** is applied to n. Then, n is decorated with $\langle \xi_0 \wedge \xi_1, [d_i, d_j], \mathbb{D}, p_n, u_n \rangle$ and $B' = B \cdot n' \cdot m'$ is such that n' is decorated with ξ_0 and m' is decorated with ξ_1. Since $p_{n'} = p_{m'} = p_n$, $u_{n'}(B') = u_{m'}(B') = 1$, $u_n(B') = 0$, and $u_m(B') = u_m(B)$ for every $m \notin \{n, n', m'\}$, we have that $\mathrm{Count}(B') = \mathrm{Count}(B) - |\xi_0 \wedge \xi_1| + |\xi_0| + |\xi_1| < \mathrm{Count}(B)$, when $p_n = 1$, and that $\mathrm{Count}(B') = \mathrm{Count}(B)$ when $p_n = 0$.
- Rule **R3** is applied to B. Then, n is decorated with $\langle \neg(\xi_0 \, C \, \xi_1), [d_i, d_j], \mathbb{D}, p_n, u_n \rangle$ and $B' = B \cdot n'$ is such that either n' is decorated with $\overline{\xi_0}$ or with $\overline{\xi_1}$. In both cases n' is decorated with $p_{n'} = 0$. This implies that $\mathrm{Count}(B') = \mathrm{Count}(B)$.
- Rule **R6** is applied to B. Then, n is decorated with $\langle \xi_0 \, C \, \xi_1, [d_i, d_j], \mathbb{D}, p_n, u_n \rangle$ and $B' = B \cdot n' \cdot m'$ where n' is decorated with ξ_0 and m' with ξ_1. Since $p_{n'} = p_{m'} = p_n$, $u_{n'}(B') = u_{m'}(B') = 1$, $u_n(B') = 0$, and $u_m(B') = u_m(B)$ for every $m \notin \{n, n', m'\}$, we have that $\mathrm{Count}(B') = \mathrm{Count}(B) - |\xi_0 \, C \, \xi_1| + |\xi_0| + |\xi_1| < \mathrm{Count}(B)$ when $p_n = 1$, and that $\mathrm{Count}(B') = \mathrm{Count}(B)$ when $p_n = 0$.

In every case, we have that $\mathrm{Count}(B') \leq \mathrm{Count}(B)$. Moreover, in the cases of rules **R1**, **R2**, **R6**, **R7**, and **R8**, applied to a node n such that $p_n = 1$, we have that $\mathrm{Count}(B') < \mathrm{Count}(B)$. Hence, the thesis is proved. ■

Let $\mathcal{L}(\varphi_\exists)$ be the language of all formulas that can be generated by the grammar rule (2) for $\mathrm{CDT}_{\mathrm{BS}}$. Another crucial property of the tableau method is that the p flag in the decoration correctly marks formulas that belong to $\mathcal{L}(\varphi_\exists)$.

Lemma 3. *Let $\mathcal{T}$ be a tableau for φ and let n be a node in $\mathcal{T}$ decorated with $\langle \psi, [d_i, d_j], \mathbb{D}, p_n, u_n \rangle$. Then, $p_n = 0$ implies that $\overline{\psi} \in \mathcal{L}(\varphi_\exists)$.*

Proof: Let n be a node on a branch B in $\mathcal{T}$, decorated with $\langle \psi, [d_i, d_j], \mathbb{D}, p_n, u_n \rangle$. We prove the claim by induction on the height $h(n)$ of n (i.e, the number of nodes on B from n to the root, including the root itself). If $h(n) \leq 2$ then n is either the root or one of the leaves of the initial tableau. In both cases the claim follows trivially. Now, let $h(n) > 2$ and suppose by inductive hypothesis that the claim holds for every ancestor of n in B. Consider the node n' to which the expansion rule has been applied in the construction of $\mathcal{T}$ to obtain the node n; as before, only conceptually different cases are shown.

- Rule **R1** was applied to n'. Then, n' is decorated with $\langle \xi_0 \wedge \xi_1, [d_i, d_j], \mathbb{D}, p_{n'}, u_{n'} \rangle$ and n is decorated either with ξ_0 or with ξ_1. Suppose n decorated with ξ_0 (the opposite case is analogous) and $p_n = 0$. By rule **R1** we have that $p_{n'} = 0$. By inductive hypothesis we have that $\overline{\xi_0 \wedge \xi_1} = \overline{\xi_0} \vee \overline{\xi_1} \in \mathcal{L}(\varphi_\exists)$. From the grammar rules it follows that $\overline{\xi_0} \in \mathcal{L}(\varphi_\exists)$.
- Rule **R3** is applied to B. Then, n' is decorated with $\langle \neg(\xi_0 \; C \; \xi_1), [d_i, d_j], \mathbb{D}, p_{n'}, u_{n'} \rangle$ and n is decorated either with $\overline{\xi_0}$ or with $\overline{\xi_1}$. Suppose n decorated with $\overline{\xi_0}$

(the opposite case is analogous). By rule **R3** we have that $p_n = 0$. From the grammar for CDT_{BS} it follows that $\xi_0 \in \mathcal{L}(\varphi_\exists)$. Since $\overline{\overline{\xi_0}} = \xi_0$, the claim is proved.

- Rule **R6** is applied to B. Then, n' is decorated with $\langle \xi_0 \, C \, \xi_1, [d_i, d_j], \mathbb{D}, p_{n'}, u_{n'} \rangle$ and n is decorated either with ξ_0 or with ξ_1. Suppose $p_{n'} = 0$: by inductive hypothesis this implies that $\overline{\xi_0 C \xi_1} = \neg(\xi_0 C \xi_1) \in \mathcal{L}(\varphi_\exists)$, a contradiction. Hence, $p_{n'} = 1$ and thus, by rule **R6**, $p_n = 1$.

In all considered cases the claim is proved, and this completes the proof. ∎

By exploiting Lemma 2 and Lemma 3 we can prove that the length of any branch B of any tableau for φ is polynomially bounded by the length of the formula.

Theorem 3. *Let $\mathcal{T}$ be a tableau for φ, and let B be a branch in $\mathcal{T}$. Then $|B| \leq 2|\varphi|^3 + 8|\varphi|^2 + 8|\varphi|$.*

Proof: Let B be a branch in a tableau $\mathcal{T}$ for φ. By the expansion rules and the expansion strategy, we have that there cannot be two nodes in B decorated with the same formula and the same interval. Since the formula in the decoration of a node is either a subformula of φ or the dual of a subformula of φ, we have that $|B| \leq 2 \cdot |\varphi| \cdot (|\mathbb{D}_B|)^2$.

To give a bound on the number of points in $\mathbb{D}_B$, it is sufficient to notice that:

1) only rules **R6**, **R7**, and **R8** adds new points to $\mathbb{D}_B$,
2) by Lemma 3, they can be applied only to nodes where the p flag is equal to 1, and
3) by Lemma 2, every application of them strictly decreases the value of $\text{Count}(B)$.

Now, let B_0 be the two-node prefix of B made by the root and one of its successor labelled with φ. Since $\text{Count}(B_0) = |\varphi|$, we have that $|\mathbb{D}_B| \leq |\varphi| + 2$ and thus we can conclude that $|B| \leq 2 \cdot |\varphi| \cdot (|\varphi| + 2)^2 \leq 2|\varphi|^3 + 8|\varphi|^2 + 8|\varphi|$. ∎

From Theorem 3 it follows that the tableau method for CDT_{BS} is terminating and that its computational complexity is in NP. Since satisfiability for propositional logic is NP-hard, the following result trivially holds.

Corollary 1. *The satisfiability problem for* CDT_{BS} *is NP-complete.*

V. Motion Planning with CDT_{BS}

To motivate the study of CDT_{BS}, in this section we explore its application to an interesting class of planning problems, showing how it is possible to benefit both from its low computational complexity and its high expressivity. Given an autonomous robot, the *motion planning problem* focuses on generating trajectories which reach a given goal while avoiding obstacles. The applicability of temporal logic in robotics dates back to the '90s, and it has mostly concentrated on point-based temporal logics. One recent example is [26], where LTL is used as the specification language and model checking techniques are used to generate robot trajectories satisfying the requirements.

We claim that CDT_{BS} can be used as an alternative specification language for motion planning problems. First of all, we state the problem as a satisfiability problem. Given a formula Sys describing all trajectories of the robot, and a formula Req describing the requirements, we have that the formula $Sys \wedge Req$ is satisfiable if and only if there exists a *feasible trajectory* for the robot, that is a trajectory respecting the requirements. Moreover, since the tableau method given in Section IV provides an example of a model satisfying the formula, we can use it to construct a plan respecting the requirements.

In [26], the robot motion is defined by a finite transition system $D = (Q, q_0, \rightarrow_D, h_D)$ where Q is a finite set of cells where the robot can be, q_0 is the initial cell, $\rightarrow_D \subseteq Q \times Q$ is the dynamics, and $h_D : Q \mapsto \Pi$ is an *observation map* that assigns a set of propositional letters in Π to every cell. To define it in CDT_{BS} we first need to introduce the "somewhere in the future" operator F, defined as follows:

$$F\varphi ::= \top \, T \, (\varphi \, T \, \top). \tag{3}$$

The formula Sys is then defined as the conjunction of the following formulas:

$$\pi \wedge q_0 \tag{4}$$

$$\bigwedge_{q \in Q} \neg F \, (q \wedge \top \, C \, (\neg q \, C \, \top)) \tag{5}$$

$$\bigwedge_{(q,q') \notin \rightarrow_D} \neg F \, (q \, C \, q') \tag{6}$$

$$\bigwedge_{q \in Q} \neg F \Big(q \wedge \neg \bigwedge_{p \in h_C(q)} p \Big) \tag{7}$$

It is easy to check that all formulas belong to CDT_{BS}, and that they describe the finite transition system D.

As an example of a possible specification, consider [26, Example 1]: *"Visit area r_2, then area r_3, then area r_4 and, finally, return to area r_1 while avoiding areas r_2 and r_3"*. It can be formally expressed as follows:

$$\begin{aligned} Req ::= {} & F(r_2 \wedge F(r_3 \wedge F(r_4 \wedge \\ & (\top \, T \, (r_1 \, T \, (\neg(\top \, C \, (r_2 \vee r_3) \, C \, \top))))))) \end{aligned} \tag{8}$$

In contrast with the LTL formalization given in [26], formula (8) has an interval interpretation. This means that the subformula Fr_2 should be read as "there is a subtrajectory completely inside area r_2", rather than "there is a point inside area r_2". Hence, more complex requirements, like "the trajectory is almost completely inside area r_2" can be expressed. Moreover, since satisfiability of our logic is NP-complete, while LTL model checking is PSPACE-complete, using CDT_{BS} is more efficient from a computational point of view. A drawback of our approach is that $\forall\exists$-properties, like reactivity properties, cannot be expressed.

VI. Undecidable extensions of CDT_{BS}

In the previous sections we have shown that the satisfiability problem for CDT_{BS} is decidable and, more precisely, it is NP-complete. Since it fits into the Bernays-Schönfinkel's class, which is NEXPTIME-complete, one may ask whether we can extend CDT_{BS} preserving decidability. In this section, we show that the most natural extension of CDT_{BS} is already undecidable.

As we have seen, CDT_{BS} allows one to build formulas in modal prenex form and such that modal operators can occur in the scope of at most one negation. Therefore, the most natural extension is to allow one more nesting of negations and modal operators, obtaining formulas of the type $\neg(\neg(pCq)Cq)$ or $\neg(pC\neg(qCr))$. In [25] it has been shown that CDT is undecidable over the class of all linearly ordered sets even if only one binary modal operator is allowed in the formulas. Undecidability has been proven by reducing the problem of finding a solution to the *Octant Tiling Problem* to the satisfiability problem of the logic. The following theorem is based on the simple observation that the entire construction exploits formulas where modal operators occur in the scope of at most *two* negations.

Theorem 4. *The satisfiability problem for any extension of* CDT_{BS} *where modal operators occur in the scope of two negations is undecidable.*

Proof: (*Sketch*) The octant tiling problem is the problem of establishing whether a given finite set of tile types $\mathcal{T} = \{t_1, \dots, t_k\}$ can tile the second octant of the integer plane $\mathcal{O} = \{(i, j) : i, j \in \mathbb{N} \wedge 0 \leq i \leq j\}$. For every tile type $t_i \in \mathcal{T}$, let $ri(t_i)$ (resp., $le(t_i)$, $up(t_i)$, $do(t_i)$) be the color of the right (resp., left, up, down) side of t_i. To solve the problem, one must find a function $f : \mathcal{O} \rightarrow \mathcal{T}$ such that $ri(f(n, m)) = le(f(n + 1, m))$ and $up(f(n, m)) = do(f(n, m + 1))$. Given an instance $\mathcal{T} = \{t_1, \dots, t_k\}$ of the octant tiling problem, we will assume that $\mathcal{AP}$ contains the following propositional letters: u, t_1,…, t_k. G is the *universal* operator, defined as the dual of the F operator (3): $G\varphi ::= \neg F \neg\varphi$. Consider now the following formulas:

$$uT\top \wedge G(u \rightarrow uT\neg u), \tag{9}$$

$$G(u \rightarrow \bigvee_{t_i \in \mathcal{T}} t_i), \tag{10}$$

$$G \bigwedge_{t_i \neq t_j} \neg(t_i \wedge t_j), \tag{11}$$

$$G \bigwedge_{t_i \in \mathcal{T}} (t_i \rightarrow \neg(uT\neg \bigvee_{t_j \in \mathcal{T}, up(t_i) = do(t_j)} t_j)), \tag{12}$$

$$G\Big(u \rightarrow \bigwedge_{t_i, t_j \in \mathcal{T},\, ri(t_j) \neq le(t_i)} \neg(t_i T t_j)\Big). \tag{13}$$

It is easy to see that, in (9), (10), and (11), modal operators occur in the scope of at most two negations. Also, formulas (12) and (13) can be rewritten in such a way that modal operators occur in the scope of at most two negations. In [25] it has been shown that $\varphi_{\mathcal{T}}$ is satisfiable if and only if $\mathcal{T}$ tiles the second octant. Using König's lemma, one can prove that a tiling system tiles the second octant if and only if it tiles arbitrarily large squares if and only if it tiles $\mathbb{N} \times \mathbb{N}$ if and only if it tiles $\mathbb{Z} \times \mathbb{Z}$. Undecidability of the first problem immediately follows from that of the last one [23]. ∎

VII. Conclusions and future work

In this paper, we studied a syntactic fragment of Venema's CDT logic, that we called CDT_{BS}, whose standard translation to first-order logic fits into the Bernays-Schönfinkel's class of prefix quantified formulas. Decidability of CDT_{BS} is thus a direct consequence of the one of the Bernays-Schönfinkel's class. We analyzed the computational complexity of the logic by developing a terminating tableau method, and proving the NP-completeness of the satisfiability problem for CDT_{BS}. Finally, we showed that any natural relaxation of the syntactic restrictions we imposed on CDT_{BS} leads to an interval logic which is expressive enough to encode the octant tiling problem, and thus turns out to be undecidable.

A question that naturally arises from our work is "Can every formula in Bernays-Schönfinkel's class of first-order logic, interpreted over linear orders and limited to binary predicates, be turned into a CDT_{BS}-formula?" Similar expressive completeness issues have been dealt with in [27], [28] (for point-based temporal logics) and in [14], [3] (for interval-based temporal logics). In particular, Venema showed that, at least in the case of dense models, there is no modal logic expressively complete for first-order logic with an unlimited number of variables, even when predicates are limited to binary ones [3]. It is not difficult to show that a similar result can be stated also for CDT_{BS} when compared with the Bernays-Schönfinkel's class. This means that, if an expressive completeness result for CDT_{BS} exists at all, it will be necessary to further limit the first-order fragment itself, probably by focusing on the fragment obtained by using at most three variables.

VIII. Acknowledgements

We would like to thank the anonymous reviewers for their useful comments and suggestions. This research has been partially supported by the EU project *FP7-ICT-223844 CON4COORD*, the Italian PRIN project *Innovative and multi-disciplinary approaches for constraint and preference reasoning*, and the Spanish MEC project *TIN2009-14372-C03-01*.

References

[1] J. Halpern and Y. Shoham, "A propositional modal logic of time intervals (short version)," in *Proc. of LICS'86*. IEEE Comp. Society, 1986, pp. 279–292.

[2] J. Allen, “Maintaining knowledge about temporal intervals,” *Communications of the ACM*, vol. 26, no. 11, pp. 832–843, 1983.

[3] Y. Venema, “A modal logic for chopping intervals,” *Journal of Logic and Computation*, vol. 1, no. 4, pp. 453–476, 1991.

[4] D. Bresolin, D. Della Monica, V. Goranko, A. Montanari, and G. Sciavicco, “Decidable and undecidable fragments of Halpern and Shoham’s interval temporal logic: towards a complete classification,” in *Proc. of LPAR’08*, ser. LNCS, vol. 5330, 2008, pp. 590–604.

[5] ——, “Undecidability of interval temporal logics with the overlap modality,” in *Proc. of TIME’09*. IEEE Comp. Society, 2009, pp. 88–95.

[6] D. Bresolin, A. Montanari, P. Sala, and G. Sciavicco, “Optimal tableaux for right propositional neighborhood logic over linear orders,” in *Proc. of JELIA’08*, ser. LNAI, vol. 5293. Springer, 2008, pp. 62–75.

[7] ——, “What’s decidable about Halpern and Shoham’s interval logic? The maximal fragment $\mathsf{AB\overline{B}\overline{L}}$,” 2011, Proc. of LICS’11.

[8] D. Della Monica, V. Goranko, A. Montanari, and G. Sciavicco, “Expressiveness of the Interval Logics of Allen’s Relations on the Class of all Linear Orders: Complete Classification,” 2011, Proc. of IJCAI’11.

[9] J. Marcinkowski and J. Michaliszyn, “The ultimate undecidability result for the Halpern-Shoham logic,” 2011, Proc. of LICS’11.

[10] J. Marcinkowski, J. Michaliszyn, and E. Kieronski, “B and D are enough to make the Halpern-Shoham logic undecidable,” in *Proc. of ICALP’10 - Part II*, ser. LNCS, vol. 6199, 2010, pp. 357–368.

[11] A. Montanari, G. Puppis, P. Sala, and G. Sciavicco, “Decidability of the interval temporal logic $AB\bar{B}$ on natural numbers,” in *Proc. of STACS’10*, 2010, pp. 597–608.

[12] V. Goranko, A. Montanari, and G. Sciavicco, “A road map of interval temporal logics and duration calculi,” *Applied Non-classical Logics*, vol. 14, no. 1-2, pp. 9–54, 2004.

[13] ——, “Propositional interval neighborhood temporal logics,” *Journal of Universal Computer Science*, vol. 9, no. 9, pp. 1137–1167, 2003.

[14] D. Bresolin, V. Goranko, A. Montanari, and G. Sciavicco, “Propositional interval neighborhood logics: expressiveness, decidability, and undecidable extensions,” *Annals of Pure and Applied Logic*, vol. 161, no. 3, pp. 289–304, 2009.

[15] A. Montanari, G. Puppis, and P. Sala, “Maximal decidable fragments of Halpern and Shoham’s modal logic of intervals,” in *Proc. of ICALP’10 - Part II*, ser. LNCS, vol. 6199, 2010, pp. 345–356.

[16] D. Bresolin, D. D. Monica, V. Goranko, A. Montanari, and G. Sciavicco, “Metric Propositional Neighborhood Logics,” *Journal of Software and System Modeling*, 2011, dOI: 10.1007/s10270-011-0195-y.

[17] D. Della Monica, V. Goranko, and G. Sciavicco, “Hybrid Metric Propositional Neighborhood Logics with Interval Length Binders,” in *Proc. of HyLo’10*, Edinburgh, Scotland, UK, July 2010, to appear in ENTCS.

[18] D. Della Monica and G. Sciavicco, “On First-Order Propositional Neighborhood Logics: a First Attempt,” in *Proc. of STeDY’10*, Lisbon, Portugal, August 2010, pp. 43–48.

[19] A. Montanari, G. Puppis, and P. Sala, “A decidable spatial logic with cone-shape cardinal directions (extended version of [29]),” Dipartimento di Matematica ed Informatica, Università di Udine, Research Report 3, 2010.

[20] D. Della Monica, “Expressiveness, decidability, and undecidability of interval temporal logic,” Ph.D. dissertation, University of Udine, 2011, available on: http://www.dia.unisa.it/dottorandi/dario.dellamonica/pubs/thesis/phd_thesis.pdf.

[21] M. Otto, “Two variable first-order logic over ordered domains,” *Journal of Symbolic Logic*, vol. 66, no. 2, pp. 685–702, 2001.

[22] E. Grädel, “Why are modal logics so robustly decidable?” *Bulletin of the EATCS*, vol. 68, pp. 90–103, 1999.

[23] E. Börger, E. Grädel, and Y. Gurevich, *The Classical Decision Problem*. Springer, 1999.

[24] V. Goranko, A. Montanari, P. Sala, and G. Sciavicco, “A general tableau method for propositional interval temporal logics: theory and implementation,” *Journal of Applied Logic*, vol. 4, no. 3, pp. 305–330, 2006.

[25] I. Hodkinson, A. Montanari, and G. Sciavicco, “Non-finite axiomatizability and undecidability of interval temporal logics with C, D, and T,” in *Proc. of CSL’08*, ser. LNCS, vol. 5213, 2008, pp. 308–322.

[26] G. Fainekos, H. Kress-Gazit, and G. Pappas, “Temporal logic motion planning for mobile robots,” in *Proc. of ICRA’05*. IEEE Comp. Society Press, 2005, pp. 2020 – 2025.

[27] K. Etessami, M. Y. Vardi, and T. Wilke, “First-order logic with two variables and unary temporal logic.” *Information and Computation*, vol. 179, no. 2, pp. 279–295, 2002.

[28] H. Kamp, “Events, instants and temporal reference,” in *Semantics from different points of view*. de Gruyter, 1979, pp. 376–417.

[29] A. Montanari, G. Puppis, and P. Sala, “A decidable spatial logic with cone-shaped cardinal directions,” in *Proc. of CSL’09*, ser. LNCS, vol. 5771, 2009, pp. 394–408.