

A Decidable Spatial Logic with Cone-Shaped Cardinal Directions

Angelo Montanari¹, Gabriele Puppis², and Pietro Sala¹

¹ Department of Mathematics and Computer Science, Udine University, Italy
 {angelo.montanari,pietro.sala}@dimi.uniud.it

² Computing Laboratory, Oxford University, England
 gabriele.puppis@comlab.ox.ac.uk

Abstract. We introduce a spatial modal logic based on cone-shaped cardinal directions over the rational plane and we prove that, unlike projection-based ones, such as, for instance, Compass Logic, its satisfiability problem is decidable (PSPACE-complete). We also show that it is expressive enough to subsume meaningful interval temporal logics, thus generalizing previous results in the literature, e.g., its decidability implies that of the subinterval/superinterval temporal logic interpreted over the rational line.

1 Introduction

Spatial reasoning has both a strong theoretical relevance and applications in many areas of computer science, including robotics, natural language processing, geographical information systems [1, 5, 12]. However, despite the widespread interest in the topic, few techniques have been developed to automatically (and efficiently) reason about spatial relations over infinite structures. As a matter of fact, spatial reasoning has been mainly investigated in quite restricted algebraic settings.

In this paper, we introduce a novel spatial modal logic, called *Cone Logic*, which allows one to reason about cone-shaped directional relations between points in the rational plane. While the satisfiability problem for spatial modal logics with projection modalities turns out to be highly undecidable [7, 9], we prove that Cone Logic enjoys a decidable satisfiability problem (in fact, PSPACE-complete) by taking advantage of a suitable filtration technique. We also show that Cone Logic subsumes interesting interval temporal logics such as the temporal logic of subintervals/superintervals, thus generalizing previous results in the literature [3] and basically disproving a conjecture by Lodaya [6].

2 Syntax and semantics of Cone Logic

In this section, we introduce syntax and semantics of Cone Logic. Let $\mathbb{P} = \mathbb{Q} \times \mathbb{Q}$ denote the *rational plane* and let $\mathbf{p} = (x, y)$ be one of its points. We denote by



Fig. 1. The four quadrants and the cone-shaped cardinal directions.

$LL(p)$, $LR(p)$, $UL(p)$, and $UR(p)$ the open *lower-left*, *lower-right*, *upper-left*, and *upper-right quadrants* of p , respectively, which are defined as follows:

$$\begin{aligned} LL(p) &= \{(x', y') : x' < x, y' < y\} & LR(p) &= \{(x', y') : x' > x, y' < y\} \\ UL(p) &= \{(x', y') : x' < x, y' > y\} & UR(p) &= \{(x', y') : x' > x, y' > y\}. \end{aligned}$$

Note that, up to a rotation of the axes, these open quadrants can be viewed as the Frank's cone-shaped cardinal directions ‘North’, ‘West’, ‘East’, ‘South’ [4] (see Figure 1). Similarly, one can denote by $LL^+(p)$, $LR^+(p)$, $UL^+(p)$, and $UR^+(p)$ the semi-closed quadrants of p , which are defined in the natural way, e.g., $LL^+(p) = \{(x', y') : x' \leq x, y' \leq y\} \setminus \{p\}$.

Given a set \mathbf{Prop} of propositional variables, formulas of Cone Logic are built up from \mathbf{Prop} using the boolean connectives \neg and \vee and eight modal operators \blacklozenge , \blacktriangleright , \blacktriangleleft , \blacktriangleright^+ , \blacktriangleleft^+ , \blacklozenge^+ , \blacktriangleright^+ , and \blacktriangleleft^+ . The *size* $|\varphi|$ of a formula φ is given by the number of its subformulas (for instance, $\blacklozenge a \vee \neg \blacklozenge \neg b$ is a formula of size 7). Formulas of Cone Logic are evaluated over (labeled regions of) the rational plane. Precisely, let $\mathcal{P} = (P, \sigma)$ be a *labeled region*, where $P \subseteq \mathbb{P}$ is a non-empty subset of the rational plane and $\sigma : P \rightarrow \mathcal{P}(\mathbf{Prop})$ is a *labeling function*. We define the semantics of a formula with respect to a distinguished initial point $p \in P$ as follows:

- $\mathcal{P}, p \models a$ iff $a \in \sigma(p)$,
- $\mathcal{P}, p \models \neg \varphi$ iff $\mathcal{P}, p \not\models \varphi$,
- $\mathcal{P}, p \models \varphi_1 \vee \varphi_2$ iff $\mathcal{P}, p \models \varphi_1$ or $\mathcal{P}, p \models \varphi_2$,
- $\mathcal{P}, p \models \blacklozenge \varphi$ (resp., $\mathcal{P}, p \models \blacklozenge^+ \varphi$) iff P contains a point q such that $q \in LL(p)$ (resp., $q \in LL^+(p)$) and $\mathcal{P}, q \models \varphi$ (and similarly for the other modal operators \blacktriangleright , \blacktriangleleft , \blacktriangleright^+ , \blacktriangleleft^+ , and \blacklozenge^+).

We further use shorthands such as $\varphi_1 \wedge \varphi_2 = \neg(\neg\varphi_1 \vee \neg\varphi_2)$, $\blacksquare \varphi = \neg \blacklozenge \neg \varphi$, $\blacklozenge \varphi = \blacklozenge \blacklozenge \varphi$, $\blacksquare \varphi = \blacksquare \blacksquare \varphi$, $\blacklozenge \varphi = \blacklozenge \blacklozenge \varphi$, $\blacksquare \varphi = \blacksquare \blacksquare \varphi$, etc.

Cone Logic is well-suited for expressing spatial relationships between points, curves, and regions inside the rational plane. Below, we give an intuitive account of its expressiveness through a couple of examples. To begin with, we show how to define an a -labeled open rectangular region, whose edges are aligned with the x - and y -axes, by means of a Cone Logic formula:

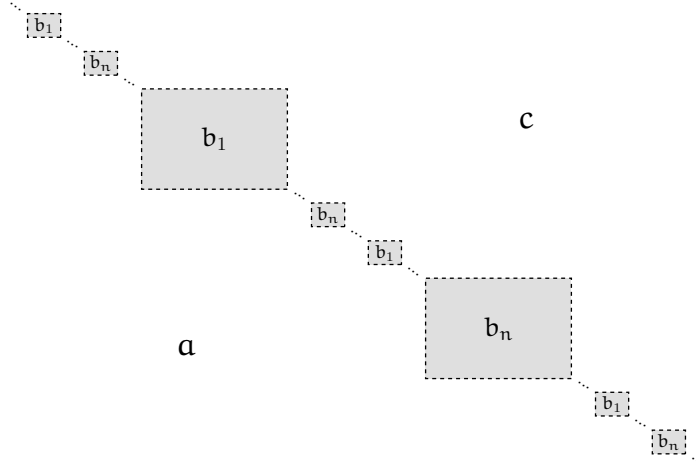


Fig. 2. A labeled rational plane satisfying $\varphi_{<}$.

$$\begin{aligned} \varphi = & \blacklozenge a \wedge \blacklozenge b \wedge \blacklozenge c \wedge \blacklozenge d \wedge \blacklozenge e \\ & \wedge \blacksquare (a \rightarrow \blacklozenge a \wedge \blacklozenge a \wedge \blacklozenge a \wedge \blacklozenge a) \wedge \blacksquare (\neg a \leftrightarrow b \vee c \vee d \vee e) \\ & \wedge \blacksquare (b \rightarrow \blacksquare b) \wedge \blacksquare (c \rightarrow \blacksquare c) \wedge \blacksquare (d \rightarrow \blacksquare d) \wedge \blacksquare (e \rightarrow \blacksquare e). \end{aligned}$$

The second example uses two symmetric modal operators, namely, \blacklozenge and \blacklozenge , to enforce non-trivial spatial relationships between labeled regions of the rational plane. Let \mathbf{Prop} be a signature containing $n + 2$ propositional variables a, b_1, \dots, b_n, c and let $<$ be the partial order over \mathbf{Prop} such that $a < b_i < c$, for every $1 \leq i \leq n$, and $b_i \not< b_j$, for every pair of distinct indices $1 \leq i, j \leq n$. We shortly write $p \leq q$ (resp., $p \geq q$) whenever $p = q$ or $p < q$ (resp., $p > q$). Consider now the (Hintikka-like) formula induced by the partial order $<$:

$$\begin{aligned} \varphi_{<} = & \blacksquare \bigvee_{p \in \mathbf{Prop}} p \wedge \blacksquare \bigwedge_{p \neq q} \neg(p \wedge q) \\ & \wedge \blacksquare \bigwedge_{p \in \mathbf{Prop}} \left(p \rightarrow \bigwedge_{q \leq p} \blacklozenge q \wedge \bigwedge_{q \geq p} \blacklozenge q \wedge \blacksquare \bigvee_{q \leq p} q \wedge \blacksquare \bigvee_{q \geq p} q \right). \end{aligned}$$

The unique (up to homeomorphism) labeled rational plane that satisfies $\varphi_{<}$ is depicted in Figure 2. Notice that (i) for every propositional variable b_i , with $1 \leq i \leq n$, the b_i -labeled region is an (infinite) union of disjoint open rectangles (in fact, the coordinates of their corners are given by pairs of irrational numbers) and (ii) the b_i -labeled open rectangles are arranged densely in the rational plane, that is, for all indices $1 \leq i, j, k \leq n$, with $i \neq j$, all b_i -labeled points (x_1, y_1) , and all b_j -labeled points (x_2, y_2) , with $x_1 < x_2$ and $y_1 > y_2$, there is a b_k -labeled point (x, y) such that $x_1 < x < x_2$ and $y_1 > y > y_2$.

The *satisfiability problem* for Cone Logic consists of deciding whether a given formula φ holds at some point of a labeled region of the rational plane. In particular, we are interested in satisfiability problems under interpretation over (open or closed) rectangular regions, namely, regions of the form $I \times J$, with I and J

being two fixed (open or closed) intervals of the rational line³. As a matter of fact, note that the whole rational plane \mathbb{P} is homeomorphic to any open rectangular region of the form $I \times J$, with $I = (x_0, x_1)$ and $J = (y_0, y_1)$. Moreover, any formula φ interpreted over an open rectangular region of the form $I \times \mathbb{Q}$, with $I = (x_0, x_1)$, can be rewritten into an equi-satisfiable formula φ' interpreted over the semi-closed rectangular region $I' \times \mathbb{Q}$, where $I' = [x_0, x_1]$. Taking advantage of the reducibility of the satisfiability problem over open rectangular regions to that over semi-closed rectangular regions, we can restrict our attention to labeled regions of the form $\mathcal{P} = (I \times \mathbb{Q}, \sigma)$, where I is a closed (non-singleton) interval (hereafter, we call such structures *labeled stripes*).

The relationships between Cone Logic and spatial logics with projection modalities deserve a little discussion. Projection-based spatial logics (most notably, Compass Logic [13]) are two-dimensional modal logics whose accessibility relations allow one to move along one of the two coordinates while keeping the other coordinate constant. On the one hand, Cone Logic inherits from projection-based modal logics some of their desirable features. For instance, it allows one to write suitable formulas that constrain labels to occur along some distinguished axes, e.g., the formula $\blacklozenge a \wedge \blacksquare \neg a \wedge \blacksquare \neg a \wedge \blacksquare \neg a$ forces a to hold at the origin or at some point over the positive x -axis. On the other hand, unlike projection-based modal logics, only a *bounded* number of constraints ‘along the axes’ can be enforced by Cone Logic. We will see that such a limitation can be traded for a positive decidability result.

Hereafter, for the sake of simplicity, we constrain Cone Logic formulas to quantify over open quadrants only, that is, to make use of the modal operators \blacklozenge , \blacktriangleright , \blacktriangleleft , and \blacklozenge only. However, the achieved results (in particular, the tree pseudo-model property proved in Section 4 and the PSPACE decision procedure described in Section 5) can be naturally generalized to the case of unrestricted Cone Logic formulas.

3 Basic machinery: types, dependencies, and shadings

Let us fix a labeled region $\mathcal{P} = (P, \sigma)$ and a formula φ of Cone Logic. In the sequel, we compare points in \mathcal{P} with respect to the set of subformulas of φ they satisfy. To do that, we introduce the key notions of φ -atom, φ -type, φ -cluster, and φ -shading.

First of all, we denote by $\mathcal{Cl}(\varphi)$ the set of all subformulas of φ and of their negations (we identify $\neg\neg\alpha$ with α , $\neg\blacklozenge\alpha$ with $\blacksquare\neg\alpha$, etc.). A φ -atom is any non-empty set $A \subseteq \mathcal{Cl}(\varphi)$ such that (i) for every formula $\alpha \in \mathcal{Cl}(\varphi)$, $\alpha \in A$ iff $\neg\alpha \notin A$ and (ii) for every formula $\gamma = \alpha \vee \beta \in \mathcal{Cl}(\varphi)$, $\gamma \in A$ iff $\alpha \in A$ or $\beta \in A$ (intuitively, a φ -atom is a maximal *locally consistent* set of subformulas of φ). The cardinality of $\mathcal{Cl}(\varphi)$ is *linear* in $|\varphi|$, while the number of φ -atoms is

³ Hereafter, square brackets are used to denote closed intervals, e.g., $[0, 1]$, while round brackets are used to denote open intervals, e.g., $(0, 1)$. Semi-open intervals are represented by mixing the two notations, e.g., $[0, 1)$.

(at most) *exponential* in $|\varphi|$. We then associate with each point $p \in \mathcal{P}$ the set of all formulas $\alpha \in \mathcal{Cl}(\varphi)$ such that $\mathcal{P}, p \models \alpha$. Such a set is called φ -*type* of p and it is denoted by $\mathcal{T}ype_{\mathcal{P}}(p)$. We have that every φ -type is a φ -atom, but not vice versa.

Given an atom A , we denote by $\mathcal{R}eq_{LL}(A)$ (resp., $\mathcal{R}eq_{LR}(A)$, $\mathcal{R}eq_{UL}(A)$, $\mathcal{R}eq_{UR}(A)$) the set of all formulas $\alpha \in \mathcal{Cl}(\varphi)$ such that $\blacklozenge \alpha \in A$ (resp., $\blacklozenge \alpha \in A$, $\blacktriangleright \alpha \in A$, $\blacktriangleleft \alpha \in A$); similarly, we denote by $\mathcal{O}bs_{LL}(A)$ (resp., $\mathcal{O}bs_{LR}(A)$, $\mathcal{O}bs_{UL}(A)$, $\mathcal{O}bs_{UR}(A)$) the set of all formulas α such that $\alpha \in A$ and $\blacklozenge \alpha \in \mathcal{Cl}(\varphi)$ (resp., $\blacklozenge \alpha \in \mathcal{Cl}(\varphi)$, $\blacktriangleright \alpha \in \mathcal{Cl}(\varphi)$, $\blacktriangleleft \alpha \in \mathcal{Cl}(\varphi)$). We call formulas belonging to one of the first (resp., last) four sets *requests* (resp., *observables*). Taking advantage of these sets, for each direction $D \in \{LL, LR, UL, UR\}$, we define two *transitive* relations \xrightarrow{D} and $\xrightarrow{\bar{D}}$ between atoms as follows:

$$\begin{aligned} A \xrightarrow{D} A' & \text{ iff } \begin{cases} \mathcal{R}eq_D(A) \supseteq \mathcal{R}eq_D(A') \cup \mathcal{O}bs_D(A') \\ \mathcal{R}eq_{\bar{D}}(A') \supseteq \mathcal{R}eq_{\bar{D}}(A) \cup \mathcal{O}bs_{\bar{D}}(A) \end{cases} \\ A \xrightarrow{\bar{D}} A' & \text{ iff } \begin{cases} \mathcal{R}eq_D(A) \supseteq \mathcal{R}eq_D(A') \\ \mathcal{R}eq_{\bar{D}}(A') \supseteq \mathcal{R}eq_{\bar{D}}(A) \end{cases} \end{aligned}$$

where \bar{D} denotes the direction opposite to D (e.g., $\overline{LL} = UR$). Note that $A \xrightarrow{D} A'$ (resp., $A \xrightarrow{\bar{D}} A'$) iff $A' \xrightarrow{\bar{D}} A$ (resp., $A' \xrightarrow{D} A$). Moreover, the relations \xrightarrow{D} and $\xrightarrow{\bar{D}}$ satisfy the *view-to-type dependency* property, namely, for every pair of points p, q in \mathcal{P} and every direction $D \in \{LL, LR, UL, UR\}$,

$$\begin{aligned} q \in D(p) & \text{ implies } \mathcal{T}ype_{\mathcal{P}}(p) \xrightarrow{D} \mathcal{T}ype_{\mathcal{P}}(q) \\ D(q) \subseteq D(p) & \text{ implies } \mathcal{T}ype_{\mathcal{P}}(p) \xrightarrow{\bar{D}} \mathcal{T}ype_{\mathcal{P}}(q). \end{aligned}$$

Below, we introduce analogous notions for regions. First, we define a φ -*cluster* as any non-empty set C of atoms. Then, for a cluster C and a direction $D \in \{LL, LR, UL, UR\}$, we denote by $\mathcal{R}eq_D(C)$ and $\mathcal{O}bs_D(C)$, respectively, the set $\bigcup_{A \in C} \mathcal{R}eq_D(A)$ and the set $\bigcup_{A \in C} \mathcal{O}bs_D(A)$. Moreover, given a pair of clusters C, C' , we write $C \xrightarrow{D} C'$ (resp., $C \xrightarrow{\bar{D}} C'$) whenever $A \xrightarrow{D} A'$ (resp., $A \xrightarrow{\bar{D}} A'$) holds for all pairs of atoms $A \in C$ and $A' \in C'$. Finally, we associate with each non-empty region \mathcal{P}' of \mathcal{P} its φ -*shading*, which is defined as the set $\mathcal{T}ype_{\mathcal{P}}(\mathcal{P}') = \{\mathcal{T}ype_{\mathcal{P}}(p) : p \in \mathcal{P}'\}$ of the φ -types of all points of \mathcal{P}' .

Note that, for every labeled region $\mathcal{P} = (P, \sigma)$, the formula φ holds at some point p of \mathcal{P} iff the shading $\mathcal{T}ype_{\mathcal{P}}(\mathcal{P})$ contains an atom A such that $\varphi \in A$. Hereafter, we shall omit the argument φ , thus calling a φ -atom (resp., a φ -type, a φ -cluster, etc.) simply an atom (resp., a type, a cluster, etc.).

4 From the rational plane to the infinite binary tree

In this section, we aim at establishing a tree (pseudo-)model property for satisfiable formulas of Cone Logic. This is done by introducing a suitable notion of decomposition of a labeled region (more precisely, of a labeled stripe) and by iteratively applying it in order to obtain an infinite decomposition tree structure that correctly represents the original model.

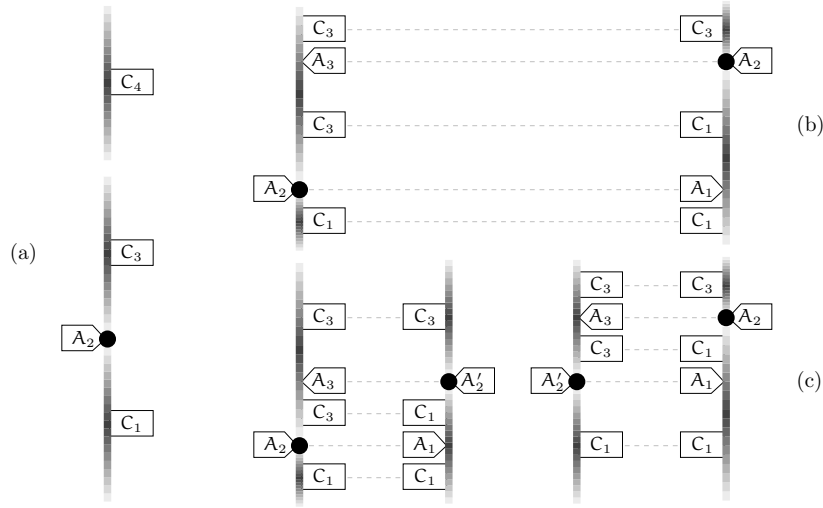


Fig. 3. Shading sequences (a), stripe expressions (b), and decompositions (c).

4.1 Shading sequences and stripe expressions

To start with, we consider the shadings of the vertical straight lines inside a labeled rational plane. A *shading sequence* is a non-empty finite sequence S of atoms and clusters such that, for every $1 \leq i \leq |S|$, if $S(i)$ is an atom, then $1 < i < |S|$ and both $S(i-1)$ and $S(i+1)$ are clusters. Shading sequences allow one to represent the types that appear along some vertical straight lines of a labeled rational plane. As an example, Figure 3(a) depicts a labeled vertical line with an associated shading sequence $S = C_1 A_2 C_3 C_4$.

To represent the shadings of the two vertical borders of a labeled stripe, we introduce the notion of *stripe expression*, which is a pair $E = (L, R)$ of (*left* and *right*) shading sequences having equal length ($|L| = |R|$) and such that, for every $1 \leq i \leq |E|$ ($= |L| = |R|$), $L(i)$ is an atom (resp., a cluster) iff $R(i)$ is an atom (resp., a cluster). We call any pair of the form $(L(i), R(i))$, with $1 \leq i \leq |E|$, a *matched pair*. As an example, Figure 3(b) depicts the left border and the right border of a labeled stripe, together with the associated stripe expression $E = (L, R)$, where $L = C_1 A_2 C_3 A_3 C_3$ and $R = C_1 A_1 C_1 A_2 C_3$. We say that an atom A is *featured* by the left (resp., right) sequence of a stripe expression $E = (L, R)$ if there is an index $1 \leq i \leq |E|$ such that $A = L(i)$ (resp., $A = R(i)$) or $A \in L(i)$ (resp., $A \in R(i)$), depending on whether $L(i)$ (resp., $R(i)$) is an atom or a cluster. By a slight abuse of notation, we denote by $\bigcup_{1 \leq i \leq |E|} L(i)$ (resp., $\bigcup_{1 \leq i \leq |E|} R(i)$) the set of all atoms featured by the left (resp., right) sequence of the stripe expression $E = (L, R)$.

For every labeled stripe \mathcal{P} , there is a stripe expression E whose left (resp., right) sequence features all and only the types of the points along the left (resp., right) border of \mathcal{P} . However, for a given stripe expression E , there might exist no labeled

stripe \mathcal{P} such that the shading of the left (resp., right) border of \mathcal{P} coincides with the set of atoms featured by the left (resp., right) shading sequences of E . In the following, we show how to get rid of such a problem. As a first step, we enforce suitable consistency conditions on any stripe expression $E = (L, R)$:

- (C1) for every $1 \leq i < j \leq |E|$, $L(i) \xrightarrow{-D} L(j)$ and $R(i) \xrightarrow{-D} R(j)$ hold for both $D = \text{UL}$ and $D = \text{UR}$;
- (C2) for every $1 \leq i \leq |E|$, if $L(i)$ and $R(i)$ are clusters, then $L(i) \xrightarrow{-D} L(i)$ and $R(i) \xrightarrow{-D} R(i)$ hold for both $D = \text{UL}$ and $D = \text{UR}$;
- (C3) for every $1 \leq i \leq |E|$, $L(i) \xrightarrow{-D} R(i)$ (and hence $R(i) \xrightarrow{-D} L(i)$) holds for both $D = \text{LR}$ and $D = \text{UR}$;
- (C4) for every $1 \leq i \leq |E|$, if $L(i)$ and $R(i)$ are atoms (resp., clusters), then $L(i) \xrightarrow{\text{LR}} R(j)$ and $L(i) \xrightarrow{\text{UR}} R(k)$ (and hence $R(j) \xrightarrow{\text{UL}} L(i)$ and $R(k) \xrightarrow{\text{UL}} L(i)$) hold for all $1 \leq j < i$ (resp., $1 \leq j \leq i$) and all $i < k \leq |L|$ (resp., $i \leq k \leq |L|$).

We compare stripe expressions with respect to their generality by introducing a suitable partial order \leq . Given two stripe expressions $E = (L, R)$ and $E' = (L', R')$, we write $E \leq E'$ if $|E| = |E'|$ and, for every index $1 \leq i \leq |E|$, we have either $L(i) = L'(i)$ and $R(i) = R'(i)$, or $L(i) \subseteq L'(i)$ and $R(i) \subseteq R'(i)$, depending on whether $L(i), R(i), L'(i), R'(i)$ are atoms or clusters. Unless otherwise stated, we tacitly assume that a stripe expression is maximal with respect to the above-defined partial order \leq . Note that a cluster appearing in a (maximal) stripe expression may contain an exponential number of distinct atoms; however, thanks to consistency conditions, the set of all its atoms can be characterized in terms of the sets of its requests and observables, namely, for every (maximal) stripe expression $E = (L, R)$, every index $1 \leq i \leq |E|$, and every atom A , we have that A belongs to the cluster $C = L(i)$ (resp., $C = R(i)$) if and only if $\text{Req}_D(A) = \text{Req}_D(C)$ and $\text{Obs}_D(A) \subseteq \text{Obs}_D(C)$ hold for all directions $D \in \{\text{LL}, \text{LR}, \text{UL}, \text{UR}\}$. This allows us to succinctly represent the two clusters of a matched pair of a (maximal) stripe expression by the sets of their requests and observables, whose size is linear in $|\varphi|$. In addition, we can assume every (maximal) stripe expression $E = (L, R)$ to contain pairwise distinct matched pairs $(L(i), R(i))$. From the above, it follows that the length $|E|$ of any (maximal) stripe expression $E = (L, R)$ is at most $4 \cdot |\varphi|$. At worst, for every pair of distinct indices $1 \leq i < j \leq |E|$, if $L(i), R(i), L(j)$, and $R(j)$ are clusters, then, for both $D = \text{UR}$ and $D = \text{UL}$, we have $\text{Req}_D(L(j)) \subseteq \text{Req}_D(L(i))$, $\text{Req}_D(R(j)) \subseteq \text{Req}_D(R(i))$, and either $\text{Req}_D(L(j)) \subsetneq \text{Req}_D(L(i))$ or $\text{Req}_D(R(j)) \subsetneq \text{Req}_D(R(i))$, and in both shading sequences there exist an atom between any pair of consecutive clusters. Hence, every (maximal) stripe expression can be represented using *polynomial space* with respect to $|\varphi|$.

4.2 Recursive decompositions of stripes

Roughly speaking, conditions C1-C4 above provide us with a guarantee that the natural spatial interpretation of a stripe expression E is *consistent* with view-to-type dependencies. To enforce the *fulfillment* of the existential requests of the

atoms featured by the two shading sequences of E , we further need to introduce a suitable notion of decomposition. We start by dividing a given labeled stripe into a pair of (thinner) adjacent labeled sub-stripes and then we recursively apply the decomposition to every emerging sub-stripe. This yields an infinite tree-shaped decomposition of the initial structure, where each vertex of the tree represents a labeled (sub-)stripe (and, thus, it is associated with a stripe expression) and each edge represents a containment relationship between two labeled (sub-)stripes.

To start with, we introduce a suitable equivalence relation between shading sequences. Two shading sequences S and S' are said to be *equivalent* if

- i) every cluster $S(i)$ of S (resp., $S'(i')$ of S') is also a cluster of S' (resp., S);
- ii) every atom $S(i)$ of S (resp., $S'(i')$ of S') either is an atom of S' (resp., S) or it belongs to the two adjacent clusters $S(i-1) = S(i+1)$ in S (resp., $S'(i'-1) = S'(i'+1)$ in S').

As an example, the shading sequences $S = C_1 A_1 C_1 C_2$ and $S' = C_1 C_2 C_2$ are equivalent, provided that $A_1 \in C_1$, while the shading sequences $S = C_1 A_1 C_2 C_2$ and $S' = C_1 C_2 C_2$ are not equivalent (unless $A_1 \in C_1$ and $C_1 = C_2$).

Decompositions of stripe expressions are defined as follows. Let $E = (L, R)$ be a stripe expression. A *decomposition* of E is any pair of stripe expressions (E_1, E_2) , with $E_1 = (L_1, R_1)$ and $E_2 = (L_2, R_2)$, such that the following *matching* conditions hold:

- (M1) L_1 and L are equivalent;
- (M2) R_2 and R are equivalent;
- (M3) R_1 and L_2 are equivalent.

We say that a matched pair $(L(i), R(i))$ of the stripe expression E *corresponds* to a matched pair $(L_1(i_1), R_1(i_1))$ (resp., $(L_2(i_2), R_2(i_2))$) of the stripe expression E_1 (resp., E_2) under the decomposition (E_1, E_2) of E if there exists an index $1 \leq i_2 \leq |E_2|$ (resp., $1 \leq i_1 \leq |E_1|$) such that (i) $L(i) \stackrel{\subseteq}{=} L_1(i_1)$, (ii) $R(i) \stackrel{\subseteq}{=} R_2(i_2)$, and (iii) $R_1(i_1) \stackrel{\subseteq}{=} L_2(i_2)$, where $\stackrel{\subseteq}{=}$ denotes either the equality relation $=$, the membership relation \in , or the containment relation \ni depending on the form of its left and right arguments (namely, whether they are atoms or clusters). As an example, Figure 3(c) depicts a decomposition of the stripe expression $E = (L, R)$, where $L = C_1 A_2 C_3 A_3 C_3$ and $R = C_1 A_1 C_1 A_2 C_3$. Note that, under such a decomposition, the matched pair (C_3, C_1) of E corresponds to the matched pairs (C_3, C_1) , (A_3, A'_2) , (C_3, C_3) of E_1 and to the matched pairs (C_1, C_1) , (A'_2, A_1) , (C_3, C_1) of E_2 . By iteratively applying decompositions, starting from a given stripe expression, one obtains an infinite tree-shaped structure, called decomposition tree.

Definition 1. A decomposition tree is an infinite complete binary labeled tree $\mathcal{T} = (V, E, \downarrow_1, \downarrow_2)$, where

- V is the set of tree vertices;
- \downarrow_1 and \downarrow_2 are the two successor relations;
- E is a labeling function associating a stripe expression $E(v)$ with each $v \in V$ such that the pair $(E(\downarrow_1(v)), E(\downarrow_2(v)))$ is a decomposition of $E(v)$.

Note that, for every pair of vertices v and v' at the same level of a decomposition tree $\mathcal{T} = (\mathbf{V}, \mathbf{E}, \downarrow_1, \downarrow_2)$, if v' is right-adjacent to v (even without being its sibling) and $\mathbf{E}(v) = (\mathbf{L}_v, \mathbf{R}_v)$ and $\mathbf{E}(v') = (\mathbf{L}_{v'}, \mathbf{R}_{v'})$ are the associated stripe expressions, then the sequence \mathbf{R}_v turns out to be equivalent to the sequence $\mathbf{L}_{v'}$.

Let $\mathcal{T} = (\mathbf{V}, \mathbf{E}, \downarrow_1, \downarrow_2)$ be a decomposition tree. We impose suitable conditions on \mathcal{T} which guarantee that every existential request of every atom featured by a stripe expression $\mathbf{E}(v)$ is eventually fulfilled by an observable of an atom featured by a (possibly different) stripe expression $\mathbf{E}(v')$. Given a stripe expression $\mathbf{E}(v) = (\mathbf{L}, \mathbf{R})$, let us denote by $\mathbf{E}(v)[\mathbf{L}]$ (resp., $\mathbf{E}(v)[\mathbf{R}]$) its left shading sequence \mathbf{L} (resp., right shading sequence \mathbf{R}). In the following, we consider a generic vertex v of \mathcal{T} and we look at the *right-oriented* (i.e., LR- and UR-oriented) requests of the atoms featured by $\mathbf{E}(v)[\mathbf{L}]$; symmetrically, we look at the *left-oriented* (i.e., LL- and UL-oriented) requests of the atoms featured by $\mathbf{E}(v)[\mathbf{R}]$.

Let us consider the UR-requests of a left shading sequence $\mathbf{E}(v)[\mathbf{L}]$. Given a vertex v of \mathcal{T} , an index $1 \leq i \leq |\mathbf{E}(v)|$, and a subformula $\alpha \in \mathcal{R}eq_{\text{UR}}(\mathbf{E}(v)[\mathbf{L}](i))$, we say that the UR-request α is

- (F1) *postponed at position i of vertex v* , if we have $\alpha \in \mathcal{R}eq_{\text{UR}}(\mathbf{E}(v)[\mathbf{R}](i))$;
- (F2) *fulfilled at position i of vertex v* , if we have $\alpha \in \mathcal{O}bs_{\text{UR}}(\mathbf{E}(v)[\mathbf{R}](j))$ for some index $i \leq j \leq |\mathbf{E}(v)|$;
- (F3) *partially fulfilled at position i of vertex v* , if there is an index $1 \leq i_1 \leq |\mathbf{E}(\downarrow_1(v))|$ such that (i) the UR-request α is fulfilled at position i_1 of vertex $\downarrow_1(v)$ and (ii) the matched pair $(\mathbf{E}(v)[\mathbf{L}](i), \mathbf{E}(v)[\mathbf{R}](i))$ of $\mathbf{E}(v)$ corresponds to the matched pair $(\mathbf{E}(\downarrow_1(v))[\mathbf{L}](i_1), \mathbf{E}(\downarrow_1(v))[\mathbf{R}](i_1))$ of $\mathbf{E}(\downarrow_1(v))$ under the decomposition $(\mathbf{E}(\downarrow_1(v)), \mathbf{E}(\downarrow_2(v)))$ of $\mathbf{E}(v)$.

Similar definitions can be given for the LR-requests of a left shading sequence $\mathbf{E}(v)[\mathbf{L}]$ and for the UL-/LL-requests of a right shading sequence $\mathbf{E}(v)[\mathbf{R}]$.

We say that a decomposition tree \mathcal{T} is *globally fulfilled* if, for every vertex v , every index $1 \leq i \leq |\mathbf{E}(v)|$, and every direction $\mathbf{D} \in \{\text{LR}, \text{UR}\}$ (resp., $\mathbf{D} \in \{\text{UL}, \text{LL}\}$), the following conditions hold:

- (G1) if v is the root, then $\mathcal{R}eq_{\mathbf{D}}(\mathbf{E}(v)[\mathbf{R}](i)) = \emptyset$ (resp., $\mathcal{R}eq_{\mathbf{D}}(\mathbf{E}(v)[\mathbf{L}](i)) = \emptyset$);
- (G2) for every subformula $\alpha \in \mathcal{R}eq_{\mathbf{D}}(\mathbf{E}(v)[\mathbf{L}](i))$ (resp., $\mathcal{R}eq_{\mathbf{D}}(\mathbf{E}(v)[\mathbf{R}](i))$) and every infinite path π that starts at v , there is a vertex v' in π (possibly $v' = v$) such that either $\alpha \notin \mathcal{R}eq_{\mathbf{D}}(\mathbf{E}(v')[\mathbf{L}](i'))$ (resp., $\alpha \notin \mathcal{R}eq_{\mathbf{D}}(\mathbf{E}(v')[\mathbf{R}](i'))$) for all positions i' of vertex v' or α is postponed (F1), fulfilled (F2), or partially fulfilled (F3) at some position i' of vertex v' .

We are now ready to establish a tree (pseudo-)model property for satisfiable formulas of Cone Logic. The next theorem states that (i) given a globally fulfilled decomposition tree \mathcal{T} , there is a labeled stripe $\mathcal{P} = (\mathbf{I} \times \mathbb{Q}, \sigma)$ whose shading coincides with the set of all atoms that are featured by the expressions of \mathcal{T} (soundness) and (ii) given a labeled stripe $\mathcal{P} = (\mathbf{I} \times \mathbb{Q}, \sigma)$, there is a globally fulfilled decomposition tree \mathcal{T} whose expressions feature (at least) the types of all points of \mathcal{P} (completeness). The proof is omitted for the lack of space (it will be included in the extended version of the paper).

Theorem 1. Soundness. *For every globally fulfilled decomposition tree $\mathcal{T} = (\mathbf{V}, \mathbf{E}, \downarrow_1, \downarrow_2)$, there is a labeled stripe $\mathcal{P} = (\mathbf{I} \times \mathbf{Q}, \sigma)$ such that*

$$\mathcal{T}ype_{\mathcal{P}}(\mathbf{I} \times \mathbf{Q}) = \bigcup_{\substack{v \in \mathbf{V} \\ 1 \leq i \leq |\mathbf{E}(v)|}} \left(\mathbf{E}(v)[\mathbf{L}](i) \cup \mathbf{E}(v)[\mathbf{R}](i) \right).$$

Completeness. Conversely, for every labeled stripe $\mathcal{P} = (\mathbf{I} \times \mathbf{Q}, \sigma)$, there is a globally fulfilled decomposition tree $\mathcal{T} = (\mathbf{V}, \mathbf{E}, \downarrow_1, \downarrow_2)$ such that

$$\mathcal{T}ype_{\mathcal{P}}(\mathbf{I} \times \mathbf{Q}) \subseteq \bigcup_{\substack{v \in \mathbf{V} \\ 1 \leq i \leq |\mathbf{E}(v)|}} \left(\mathbf{E}(v)[\mathbf{L}](i) \cup \mathbf{E}(v)[\mathbf{R}](i) \right).$$

5 Reducing Cone Logic to a proper fragment of CTL

In this section we briefly describe a decision procedure that solves the satisfiability problem for Cone Logic taking advantage of the tree (pseudo-)model property stated in Section 4. According to such a property, the problem of establishing whether or not a Cone Logic formula φ is satisfiable can be reduced to the problem of checking the existence of a globally fulfilled decomposition tree \mathcal{T} that features a (φ) -atom \mathbf{A} such that $\varphi \in \mathbf{A}$. The effectiveness of such an approach stems from the fact that the properties that characterize a globally fulfilled decomposition tree can be expressed in a proper fragment of CTL. The satisfiability problem for Cone Logic can thus be decided in (at most) exponential time [8]. Given the state of the art of the decision procedures for CTL, deciding the satisfiability problem for Cone Logic turns out to be quite efficient from a practical point of view. In the following, we show that the satisfiability problem for Cone Logic is actually in PSPACE. In the next section, we will prove that the PSPACE complexity bound is strict, namely, that the satisfiability problem for Cone Logic is PSPACE-hard.

Theorem 2. *The satisfiability problem for Cone Logic, interpreted over the rational plane, is in PSPACE.*

Proof (sketch). We first show how to reduce the satisfiability problem for a Cone Logic formula φ to the satisfiability problem for a suitable CTL formula $\vec{\varphi}$, which is a conjunction of CTL formulas of the forms λ , $\mathbf{AG} \lambda$, $\mathbf{EF} \lambda$, $\mathbf{AG EX} \lambda$, $\mathbf{AG} \delta$, and $\mathbf{AG} (\lambda \rightarrow \mathbf{AF} \delta)$, where λ is a propositional formula and δ is a CTL formula that uses the modal operator \mathbf{AX} in a *positive* way only ((and it has no occurrences of other modal operators). Let us call these formulas *basic* CTL formulas.

To start with, we introduce three distinguished propositional variables, say, 0, 1, and 2, to encode the two successor relations \downarrow_1 and \downarrow_2 of a decomposition tree \mathcal{T} in a labeled tree structure \mathbf{T} . For each vertex v of \mathbf{T} , we associate either 0, 1, and 2 with v depending on whether v is the root, $v = \downarrow_1(\mathbf{u})$, or $v = \downarrow_2(\mathbf{u})$ for some parent vertex \mathbf{u} . Such a labeling can be enforced by means of a suitable conjunction of basic CTL formulas over the signature $\{0, 1, 2\}$ (see below). Next, the stripe expressions associated with \mathcal{T} vertices can be encoded as follows. Since the number of shading sequences can be exponential in $|\varphi|$, we need to encode *one* by

one the elements that belong to each atom featured by each shading sequence. To this end, we introduce a new set Σ of propositional variables l_i^{atom} , $l_i^{cluster}$, r_i^{atom} , $r_i^{cluster}$, $l_{i,\alpha,D}^{obs}$, $l_{i,\alpha,D}^{req}$, $r_{i,\alpha,D}^{obs}$, and $r_{i,\alpha,D}^{req}$, for every index $1 \leq i \leq 4 \cdot |\varphi|$, every subformula α of φ , and every direction $D \in \{LL, LR, UL, UR\}$. Intuitively, the propositional variable l_i^{atom} (resp., $l_i^{cluster}$) holds at a given vertex v of T if and only if the position i of the left shading sequence $E(v)[L]$ in \mathcal{T} contains an atom (resp., a cluster). Similarly, the propositional variable $l_{i,\alpha,D}^{obs}$ (resp., $l_{i,\alpha,D}^{req}$) holds at a given vertex v of T if and only if the subformula α belongs to the set of observables $\mathcal{Obs}_D(E(v)[L](i))$ (resp., the set of requests $\mathcal{Req}_D(E(v)[L](i))$) of the atom/cluster at position i of the left shading sequence $E(v)[L]$. Analogous encodings are given for the right shading sequence $E(v)[R]$. Since the number of subformulas α of φ is linear in $|\varphi|$ and the length of a (maximal) stripe expression $E(v)$ is at most $4 \cdot |\varphi|$, the above-defined propositional variables allow one to represent $E(v)$ in polynomial space. The (local) consistency conditions **C1–C4** can be easily expressed by means of a propositional formula λ_{C1-C4} over the signature Σ and hence they can be enforced globally in the labeled tree structure T by requiring that the basic CTL formula $\mathbf{AG} \lambda_{C1-C4}$ holds at the root of T . Similarly, the matching conditions **M1–M3**, which impose further restrictions on the stripe expressions associated with pairs of adjacent vertices, can be expressed by means of a basic CTL formula of the form $\mathbf{AG} \delta_{M1-M3}$ over the signature $\{0, 1, 2\} \cup \Sigma$, where δ_{M1-M3} contains only positive occurrences of the modal operator \mathbf{AX} (and no occurrences of other modal operators). As for the conditions of global fulfillment, we can enforce Condition **G1** by a simple propositional formula λ_{G1} over the signature Σ and Condition **G2** by a conjunction of basic CTL formulas of the form $\mathbf{AG} (\lambda_{i,D,\alpha} \rightarrow \mathbf{AF} \delta_{i,D,\alpha})$, where i ranges over $\{1, \dots, 4 \cdot |\varphi|\}$, D ranges over $\{LL, LR, UL, UR\}$, α is a subformula of φ , $\lambda_{i,D,\alpha}$ is a propositional formula over the signature Σ , and $\delta_{i,D,\alpha}$ is a CTL formula over the signature $\{0, 1, 2\} \cup \Sigma$ that contains only positive occurrences of the modal operator \mathbf{AX} (and no occurrences of other modal operators). Finally, the existence of a (φ -)atom A in \mathcal{T} such that $\varphi \in A$ can be enforced by a basic CTL formula of the form $\mathbf{EF} \lambda_\varphi$. The size of all the above formulas is polynomial in $|\varphi|$. Altogether, we have that for any formula φ of Cone Logic, we can (effectively) build an *equi-satisfiable* conjunction of basic CTL formulas $\vec{\varphi}$ over the signature $\{0, 1, 2\} \cup \Sigma$ in polynomial time.

We conclude the sketch of the proof by outlining a PSPACE procedure that checks whether the resulting formula $\vec{\varphi}$ is satisfiable or not. First, we write $\vec{\varphi}$ as the conjunction of the following three CTL formulas:

$$\begin{aligned} \vec{\varphi}_{tree} &= (0 \wedge \neg 1 \wedge \neg 2) \wedge \mathbf{AG} \mathbf{AX} (\neg 0 \wedge \neg(1 \wedge 2)) \wedge \mathbf{AG} (\mathbf{EX} 1 \wedge \mathbf{EX} 2) \\ \vec{\varphi}_{path} &= \mathbf{AG} \lambda_{C1-C4} \wedge \mathbf{AG} \delta_{M1-M3} \wedge \lambda_{G1} \wedge \bigwedge_{i,D,\alpha} \mathbf{AG} (\lambda_{i,D,\alpha} \rightarrow \mathbf{AF} \delta_{i,D,\alpha}) \\ \vec{\varphi}_{init} &= \mathbf{EF} \lambda_\varphi \end{aligned}$$

The formula $\vec{\varphi}_{tree}$ defines a labeled tree structure T where each vertex has two distinguishable successors, the formula $\vec{\varphi}_{path}$ verifies that T correctly represents a globally fulfilled decomposition tree \mathcal{T} , and the formula $\vec{\varphi}_{init}$ checks that \mathcal{T} features an atom A such that $\varphi \in A$. Since $\vec{\varphi}_{path}$ contains only *positive* oc-

currences of the modal operators \mathbf{AG} , \mathbf{AF} , and \mathbf{AX} , we can turn $\vec{\varphi}_{path}$ into an equivalent LTL formula $\vec{\varphi}_{path}^{LTL}$ by replacing all occurrences of \mathbf{AG} , \mathbf{AF} , and \mathbf{AX} by \mathbf{G} , \mathbf{F} , and \mathbf{X} , respectively. Formally, we have that for any labeled tree structure T , $\vec{\varphi}$ holds at the root of T if and only if (i) $\vec{\varphi}_{tree}$ and $\vec{\varphi}_{init}$ hold at the root of T and (ii) $\vec{\varphi}_{path}^{LTL}$ holds along all infinite paths of T . By taking advantage of the structure of $\vec{\varphi}_{path}^{LTL}$ (no \mathbf{G} operator is nested into an \mathbf{F} operator), it is possible to show that there exists a *deterministic* Büchi automaton \mathcal{A}_{path} , which can be computed in polynomial time⁴, that recognizes the ω -language of all linear models of $\vec{\varphi}_{path}^{LTL}$. Given the automaton \mathcal{A}_{path} over the input alphabet $\{0, 1, 2\} \times \mathcal{P}(\Sigma)$, we build a *non-deterministic* Büchi automaton $\mathcal{A}_{path}^{\exists}$ that recognizes the projection language $\pi_{0,1,2}(\mathcal{L}^{\omega}(\mathcal{A}_{path}))$. We have that $\mathcal{A}_{path}^{\exists}$ recognizes the ω -language $\{0\} \cdot \{1, 2\}^{\omega}$ if and only if there exists a labeled tree structure T that satisfies both $\vec{\varphi}_{tree}$ and $\vec{\varphi}_{path}^{LTL}$. Since the inclusion problem for regular ω -languages is in PSPACE [10], this gives a procedure that decides, in polynomial space, whether both formulas $\vec{\varphi}_{tree}$ and $\vec{\varphi}_{path}$ hold at the root of some labeled tree structure T . Verifying whether $\vec{\varphi}_{init}$ holds at the root of T as well amounts to solve a reachability problem over a slightly modified version of the non-deterministic Büchi automaton $\mathcal{A}_{path}^{\exists}$. \square

6 Cone Logic and interval temporal logics

In this section, we prove that Cone Logic subsumes an interesting interval temporal logic, called *DD \bar{D} Y \bar{Y} -logic*, which comprises four modal operators $\langle \mathbf{D} \rangle$, $\langle \bar{\mathbf{D}} \rangle$, $\langle \mathbf{Y} \rangle$, and $\langle \bar{\mathbf{Y}} \rangle$. Intuitively, these operators quantify over sub-intervals, super-intervals, ‘younger intervals’, and ‘elder intervals’. From now on, we assume that the underlying temporal domain is (homeomorphic to) the linear ordering $(\mathbb{Q}, <)$ of the rational numbers and that intervals are non-singleton closed convex subsets of such an ordering, namely, sets of the form $[x, y] = \{z \in \mathbb{Q} : x \leq z \leq y\}$, where $x, y \in \mathbb{Q}$ and $x < y$. We shortly denote by \mathbb{I} the set of all intervals over $(\mathbb{Q}, <)$.

The four interval relations \mathbf{D} , $\bar{\mathbf{D}}$, \mathbf{Y} , and $\bar{\mathbf{Y}}$ and the semantics of DD \bar{D} Y \bar{Y} -logic are defined as follows. Let $I = [x, y]$ and $I' = [x', y']$ be two intervals. If $x < x' < y' < y$, then we say that I' is a (strict) *sub-interval* of I , or, conversely, that I is a (strict) *super-interval* of I' . Similarly, if $x' > x$ and $y' > y$, we say that I' is *younger* than I , or, conversely, that I is *elder* than I' . It is worth noticing that the younger-interval relation \mathbf{Y} and the elder-interval relation $\bar{\mathbf{Y}}$ can be characterized as unions of standard Allen’s relations [2] (for instance, the relation \mathbf{Y} is the union of the ‘later’ relation \mathbf{L} , the ‘immediately after’, or ‘meet’, relation \mathbf{A} , and the ‘overlap’ relation \mathbf{O}). As for the semantics of DD \bar{D} Y \bar{Y} -logic, let $\mathcal{P} = (\mathbb{I}, \sigma)$ be an interval structure, where σ is a valuation function that

⁴ First, we turn each conjunct of $\vec{\varphi}_{path}^{LTL}$ into an equivalent deterministic Büchi automaton and then we compute the product automaton for the whole formula $\vec{\varphi}_{path}^{LTL}$. The resulting automaton has size polynomial in $|\vec{\varphi}_{path}^{LTL}|$, provided that the transition labels are symbolically represented by means of suitable propositional formulas over the signature $\{0, 1, 2\} \cup \Sigma$.

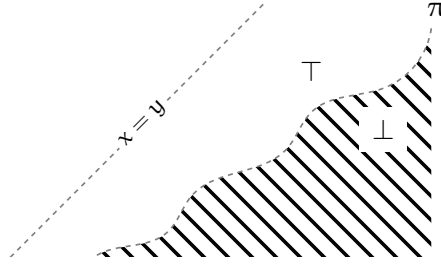


Fig. 4. A π -labeled region delimiting (pseudo)interval-points.

maps intervals in \mathbb{I} to sets of propositional variables. The formulas of $\text{DD}\bar{\text{D}}\text{Y}\bar{\text{Y}}$ -logic are built up from propositional variables using the boolean connectives and the modal operators $\langle \text{D} \rangle$, $\langle \bar{\text{D}} \rangle$, $\langle \text{Y} \rangle$, and $\langle \bar{\text{Y}} \rangle$, with the obvious semantics (for instance, $\mathcal{P}, I \models \langle \text{D} \rangle \varphi$ holds iff there is a sub-interval I' of I such that $\mathcal{P}, I' \models \varphi$).

In [6] Lodaya conjectured the undecidability of the satisfiability problem for the fragment of $\text{DD}\bar{\text{D}}\text{Y}\bar{\text{Y}}$ -logic that features the two modal operators $\langle \text{D} \rangle$ and $\langle \bar{\text{D}} \rangle$ only when interpreted over various classes of linear orderings. Here, we partially disprove such a conjecture by showing that formulas of $\text{DD}\bar{\text{D}}\text{Y}\bar{\text{Y}}$ -logic, interpreted over the rational line, can actually be translated into equi-satisfiable formulas of Cone Logic. Such a translation exploits the fact that there exists a natural bijection between intervals $I = [x, y]$ in \mathbb{I} and points $p = (x, y)$, with $x < y$, of the rational plane (hereafter, we call these points *interval-points*). Moreover, the region of all and only the interval-points of the rational plane can somehow be described by a suitable formula of Cone Logic. More precisely, let \top , \perp , and π be three fresh propositional variables and let ψ_π be the following formula of Cone Logic:

$$\begin{aligned} \psi_\pi = & \blacksquare(\top \vee \perp \vee \pi) \wedge \blacksquare(\neg \top \vee \neg \perp) \wedge \blacksquare(\neg \top \vee \neg \pi) \wedge \blacksquare(\neg \perp \vee \neg \pi) \\ & \wedge \blacksquare(\top \rightarrow \blacklozenge \top \wedge \blacksquare \top) \wedge \blacksquare(\perp \rightarrow \blacklozenge \perp \wedge \blacksquare \perp) \\ & \wedge \blacksquare(\pi \rightarrow \blacksquare^+ \top \wedge \blacksquare^+ \perp) \wedge \blacksquare(\blacklozenge \pi \wedge \blacklozenge \pi \rightarrow \pi \vee \blacklozenge \pi \vee \blacklozenge \pi). \end{aligned}$$

Consider now a labeled rational plane $\mathcal{P} = (\mathbb{P}, \sigma)$ that satisfies ψ_π . We can partition \mathcal{P} in three regions, namely, (i) the region $\top_{\mathcal{P}}$ of all \top -labeled points, (ii) the region $\perp_{\mathcal{P}}$ of all \perp -labeled points, and (iii) the region $\pi_{\mathcal{P}}$ of all π -labeled points (see Figure 4). The region $\pi_{\mathcal{P}}$ has the form of a ‘thin’ oriented trajectory inside the rational plane such that, for every pair of points $p, q \in \pi_{\mathcal{P}}$, there exists another point $r \in \pi_{\mathcal{P}}$ such that either $r \in \text{UR}(p)$ and $q \in \text{UR}(r)$, or $r \in \text{LL}(p)$ and $q \in \text{LL}(r)$. Even though we cannot claim that $\pi_{\mathcal{P}}$ coincides with the diagonal $\{(x, x) : x \in \mathbb{Q}\}$ and $\top_{\mathcal{P}}$ coincides with the set of all interval-points of the rational plane, we can prove the following proposition.

Proposition 1. *For every formula φ of Cone Logic and every labeled rational plane $\mathcal{P} = (\mathbb{P}, \sigma)$ that satisfies $\varphi_\pi = \varphi \wedge \psi_\pi$, there is a labeled rational plane $\mathcal{P}' = (\mathbb{P}, \sigma')$ that satisfies φ_π and such that (i) the π -labeled region $\pi_{\mathcal{P}'}$ coincides with the diagonal $\{(x, x) : x \in \mathbb{Q}\}$ and (ii) the \top -labeled region $\top_{\mathcal{P}'}$ coincides with the set of all interval-points of the rational plane.*

Proof. Let $\mathcal{P} = (\mathbb{P}, \sigma)$ be a model of the formula $\varphi_\pi = \varphi \wedge \psi_\pi$. Without loss of generality, we can assume that for every $x \in \mathbb{Q}$, there is a π -labeled point of the form $p = (x, y)$, with $y \in \mathbb{Q}$ (this follows easily from Theorem 1). We can thus view the region $\pi_{\mathcal{P}}$ as the graph of a strictly increasing function $f_\pi : \mathbb{Q} \rightarrow \mathbb{Q}$ such that, for every point $p = (x, y)$, $\pi \in \sigma(p)$ if and only if $f_\pi(x) = y$. Thus, we can denote by f_π^{-1} the inverse of f_π , which is a strictly increasing function as well, and we can introduce the (monotone) transformation t that maps any point $p = (x, y)$ to the point $t(p) = (x, f_\pi^{-1}(y))$. We then exploit such a transformation to define a new labeling function σ' as follows: for every point p , we let $\sigma'(p) = \sigma(t(p))$. By definition of t , the resulting structure $\mathcal{P}' = (\mathbb{P}, \sigma')$ is homeomorphic to \mathcal{P} and hence it also satisfies the formula φ_π . Moreover, by construction, the region $\pi_{\mathcal{P}'}$ coincides with the diagonal $\{(x, x) : x \in \mathbb{Q}\}$ and, similarly, the region $\top_{\mathcal{P}'}$ coincides with the set of all interval-points of the rational plane. \square

Proposition 1 yields a straightforward translation of any given formula φ of $\text{DD}\bar{Y}\bar{Y}$ -logic into an equi-satisfiable formula φ' of Cone Logic, which is obtained by first replacing in φ every occurrence of the subformula $\langle D \rangle \alpha$ (resp., $\langle \bar{D} \rangle \alpha$, $\langle Y \rangle \alpha$, $\langle \bar{Y} \rangle \alpha$) with the formula $\blacklozenge(\top \wedge \alpha)$ (resp., $\blacklozenge(\top \wedge \alpha)$, $\blacklozenge(\top \wedge \alpha)$, $\blacklozenge(\top \wedge \alpha)$) and then adding the conjunct ψ_π to the resulting formula. Taking advantage of such a translation and of the decision procedure described in Section 5, we immediately obtain that the satisfiability problem for $\text{DD}\bar{Y}\bar{Y}$ -logic is in PSPACE. As a matter of fact, this subsumes previous results from [3]. Moreover, from [11] we know that the satisfiability problem for D-logic, that is, the interval temporal logic that features the subinterval operator $\langle D \rangle$ only, and hence that for $\text{DD}\bar{Y}\bar{Y}$ -logic, is PSPACE-hard. Summing up, we have the following corollary.

Corollary 1. *The satisfiability problem for Cone Logic, interpreted over the rational plane, and that for $\text{DD}\bar{Y}\bar{Y}$ -logic, interpreted over the rational line, are PSPACE-complete.*

7 Conclusions

We would like to conclude by mentioning some natural generalizations of our work. First, we may consider various possible extensions of Cone Logic. For instance, we may think of evaluating formulas of (extended) Cone Logic over *multi-dimensional spaces* (in general, 2^n distinct cone-shaped directions exist in a space with n dimensions) and/or to partition the two-dimensional space into *more than four* cone-shaped cardinal directions (the same for higher-dimensional spaces). In all such cases, we believe it possible to generalize the achieved results in a rather natural way, preserving the tree pseudo-model property of the logic and, possibly, the PSPACE-completeness of its satisfiability problem. Further generalizations envisage the use of region-based spatial logics. As an example, the correspondence between intervals over the rational line and points over the rational plane can be lifted to higher-dimensional objects, proving, for instance, that a suitable spatial logic based on rectangular regions, that is, 2-dimensional intervals, is subsumed by a 4-dimensional point-based modal logic very similar

to Cone Logic. This establishes an interesting bridge between Cone Logic and modal logics of topological relations. Finally, it is worth studying the satisfiability problem for Cone Logic, and, similarly, for $\text{DD}\bar{\text{D}}\text{Y}\bar{\text{Y}}$ -logic, interpreted over different (classes of) structures, e.g., the infinite discrete grid or the Euclidean plane. Even though we expect the satisfiability problem to remain decidable, radically different approaches might be necessary to cope with spaces having discrete or Euclidean topologies.

References

- [1] M. Aiello, I. Pratt-Hartmann, and J. van Benthem. *Handbook of Spatial Logics*. Springer, 2007.
- [2] J.F. Allen. Maintaining knowledge about temporal intervals. *Communications of the Association for Computing Machinery*, 26(11):832–843, 1983.
- [3] D. Bresolin, V. Goranko, A. Montanari, and P. Sala. Tableau-based decision procedures for the logics of subinterval structures over dense orderings. *Journal of Logic and Computation*, doi:10.1093/logcom/exn063, 2008.
- [4] A.U. Frank. Qualitative spatial reasoning about distances and directions in geographic space. *Journal of Visual Languages and Computing*, 3:343–371, 1992.
- [5] D.M. Gabbay, A. Kurucz, F. Wolter, and M. Zakharyashev. *Many-Dimensional Modal Logics: theory and applications*, volume 148 of *Studies in Logic and the Foundations of Mathematics*. Elsevier Science Publishers, 2003.
- [6] K. Lodaya. Sharpening the undecidability of interval temporal logic. In *Proceedings of the 6th Asian Computing Science Conference on Advances in Computing Science*, volume 1961 of *Lecture Notes in Computer Science*, pages 290–298. Springer, 2000.
- [7] M. Marx and M. Reynolds. Undecidability of compass logic. *Journal of Logic and Computation*, 9(6):897–914, 1999.
- [8] A. Meier, M. Mundhenk, M. Thomas, and H. Vollmer. The complexity of satisfiability for fragments of CTL and CTL*. *Electronic Notes in Theoretical Computer Science*, 223:201–213, 2008.
- [9] A. Morales, I. Navarrete, and G. Sciavicco. A new modal logic for reasoning about space: spatial propositional neighborhood logic. *Annals of Mathematics and Artificial Intelligence*, 51(1):1–25, 2007.
- [10] S. Safra. On the complexity of ω -automata. In *Proceedings of the 29th Annual Symposium on Foundations of Computer Science*, pages 319–327. IEEE Computer Society, 1988.
- [11] I. Shapirovsky and V.B. Shehtman. Chronological future modality in Minkowski spacetime. In *Proceedings of the 4th Conference on Advances in Modal Logic*, pages 437–460. King’s College Publications, 2003.
- [12] O. Stock. *Spatial and Temporal Reasoning*. Kluwer Academic, 1997.
- [13] Y. Venema. Expressiveness and completeness of an interval tense logic. *Notre Dame Journal of Formal Logic*, 31(4):529–547, 1990.