Exploring the bidimensional space: a dynamic logic point of view

Paper ID 179

ABSTRACT

We present a family of logics for reasoning about agents' positions and movements in the plane which have several potential applications in the area of multi-agent systems (MAS), such as multi-agent planning and robotics. The most general logic includes (i) atomic formulas for representing the truth of a given fact or the presence of a given agent at a certain position of the plane, (ii) atomic programs corresponding to the four basic orientations in the plane (up, down, left, right) as well as the four program constructs of propositional dynamic logic PDL (sequential composition, nondeterministic composition, iteration and test). As this logic is not computably enumerable, we study some interesting decidable and axiomatizable fragments of it. We also present a decidable extension of the iteration-free fragment of the logic by special programs representing movements of agents in the plane.

1. INTRODUCTION

Most of existing logics for multi-agent systems (MAS) including multi-agent epistemic logic [8], multi-agent variants of propositional dynamic logic [16] and logics of action and strategic reasoning such as ATL [1], Coalition Logic CL [13] and STIT [5] are "ungrounded" in the sense that their formal semantics are based on abstract primitive notions such as the concept of Kripke model or the concept of possible world (or state). As a result, there is no direct connection between these *abstract* concepts and the *concrete* environment in which the agents' interact. This kind of grounding problem of logics for MAS becomes particularly relevant for robotic applications. Since robots are situated in spatial environments, in order to make logics for MAS useful for robotics, their semantics have to be grounded on space. Specifically, a formal semantics is required that provides an explicit representation of the space in which the robots' actions and perceptions are situated. Some initial steps into the direction of grounding formal semantics of logics for MAS on space have done in the recent years. Among them, we should mention logics of multi-agent knowledge in both one-dimensional space and two-dimensional space [10, 3], spatio-temporal logics such as constraint LTL applied to model 2D grid environments [2], multi-robot task logic based on monadic second-order logic [15] and logics of robot localization [4]. The present paper shares with these approaches the idea that in order to make existing logics of MAS useful for MAS applications such as multi-agent planning and robotics, their semantics should provide an explicit representation of the agents' environment.

The main motivation of the present work is to provide a logical framework whose language and semantics are, at the same time, simple and sufficiently general to describe (i) the properties of the spatial environment in which several agents can move, and (ii) the consequences of the agents' movements on such a spatial environment. To meet this objective, we have decided to exploit the language of propositional dynamic logic PDL as a general formalism for representing actions of agents and their effects, and to interpret this language on a simple formal semantics of the two-dimensional (2D) space. The reason why we decided to start from the 2D space is that its representation already presents some interesting conceptual aspects as well as some difficulties with respect to the computational properties of the resulting logic. We believe that, before studying action in the 3D space and, more generally, action in n-dimensional spaces (with $n > 2$), a comprehensive logical theory of action in the 2D space is required.

More concretely, this paper presents a family of logics for reasoning about agents' positions and movements in the plane. The most general logic, called Dynamic Logic of Space $\mathsf{DL\text{-}S}^*$, is presented in Section 2. $\mathsf{DL\text{-}S}^*$ includes (i) atomic formulas for representing the truth of a given fact (*atomic facts*) or the presence of a given agent at a certain position of the plane (*positional atoms*), (ii) atomic programs corresponding to the four basic orientations in the plane (up, down, left, right) as well as the four program constructs of PDL (sequential composition, nondeterministic composition, iteration and test). The logic is proved to be non-computably enumerable (non-c.e.) and its satisfiability problem undecidable (Section 3), while its model-checking problem is proved to be decidable in deterministic polynomial time (Section 4). Given the negative properties of $\mathsf{DL\text{-}S}^*$, we decided to study some interesting decidable and axiomatizable fragments of it. This includes the iteration-free fragment of $\mathsf{DL\text{-}S}^*$ (Section 5) as well as a fragment that only allows iteration of the same atomic program (e.g., the action of moving an indefinite number of times to the right) and has no atomic formulas aside from positional atoms (Section 6). As the logic $\mathsf{DL\text{-}S}^*$ only provides a static representation of the 2D space, in Section 7 we present a

Appears in: *Proceedings of the 16th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2017), S. Das, E. Durfee, K. Larson, M. Winikoff (eds.), May 8–12, 2017, São Paulo, Brazil.*

decidable extension of its iteration-free fragment by special programs representing movements of agents in the plane. Conclusion of the paper (Section 8) presents perspectives of future research including integration of an epistemic component in the logic as well as of the concept of coalitional capability in the sense of [13].

2. SPACE

$\mathsf{DL\text{-}S}^*$ (Dynamic Logic of Space) is a dynamic logic which consists of: (i) formulas representing the agents' positions and the truth of facts in the different positions of the bidimensional space, and (ii) programs allowing to move from one position to another position of the bidimensional space.

2.1 Syntax

Assume a countable set of atomic propositions $Atm = \{p, q, \dots\}$ and a finite set of agents $Agt = \{1, \dots, n\}$.

The language of $\mathsf{DL\text{-}S}^*$, denoted by $\mathcal{L}_{\mathsf{DL\text{-}S}^*}(Atm, Agt)$, is defined by the following grammar in Backus-Naur Form:

$$\begin{array}{lll} \alpha & ::= & \Uparrow \mid \Downarrow \mid \Rightarrow \mid \Leftarrow \mid \alpha;\alpha' \mid \alpha \cup \alpha' \mid \alpha^* \mid ?\varphi \\ \varphi & ::= & p \mid \mathsf{h}_i \mid \neg\varphi \mid \varphi \wedge \psi \mid [\alpha]\varphi \end{array}$$

where p ranges over Atm and i ranges over Agt. Other Boolean constructions $\top$, $\bot$, $\vee$, $\rightarrow$ and $\leftrightarrow$ are defined from p, $\neg$ and $\wedge$ in the standard way. Instances of α are called *spatial programs.* When there is no risk of confusion we will omit parameters and simply write $\mathcal{L}_{\mathsf{DL\text{-}S}^*}$. The *modal degree* of a formula $\varphi \in \mathcal{L}_{\mathsf{DL\text{-}S}^*}$ (in symbols $\deg(\varphi)$) is defined in the standard way as the nesting depth of modal operators in φ. Let $\parallel \varphi \parallel$ denote the size of φ. For all (negative or positive) integers x, let $[\Uparrow]^x$ be the modality consisting of x consecutive $[\Downarrow]$ when $x \leq 0$, otherwise let $[\Uparrow]^x$ be the modality consisting of x consecutive $[\Uparrow]$. Similarly for $[\Rightarrow]^x$.

The formula h_i is read "the agent i is here", whereas $[\alpha]\varphi$ has to be read "φ is true in the position that is reachable from the current position through the program α".

We will also be interested in sublanguages of $\mathcal{L}_{\mathsf{DL\text{-}S}^*}$. Given a set P of atomic propositions, a set I of agents and a set A of spatial programs, we denote the restriction of $\mathcal{L}_{\mathsf{DL\text{-}S}^*}(P, I)$ which only allows programs from A by $\mathcal{L}_{\mathsf{DL\text{-}S}^*}(P, I, A)$.

2.2 Semantics

The main notion in semantics is given by the following concept of spatial model.

Definition 1 (Spatial model (SM)). *A spatial model is a tuple $M = (\mathcal{P}, \mathcal{V})$ where:*

- $\mathcal{P} : Agt \longrightarrow \mathbb{Z} \times \mathbb{Z}$ *and*
- $\mathcal{V} : \mathbb{Z} \times \mathbb{Z} \longrightarrow 2^{Atm}$.

The set of all spatial models is denoted by $\mathbf{M}$.

For every $(x, y) \in \mathbb{Z} \times \mathbb{Z}$, $\mathcal{P}(i) = (x, y)$ means that the agent i is in the position (x, y), whereas $p \in \mathcal{V}(x, y)$ means that p is true at the position (x, y). For every $x \in \mathbb{Z}$, $succ(x)$ denotes the direct successor of x (*i.e.*, $x + 1$), while $prec(x)$ denotes the direct predecessor of x (*i.e.*, $x - 1$).

Formulas are evaluated with respect to a spatial model M and a spatial position (x, y). Below, if R, S are binary relations, R^* denotes the transitive, reflexive closure of R and $R \circ S$ the composition of R and S.

Definition 2 (R_α and truth conditions). *Let $M = (\mathcal{P}, \mathcal{V})$ be a spatial model. For all spatial programs α and for all formulas φ, the binary relation R_α on $\mathbb{Z} \times \mathbb{Z}$ and the truth conditions of φ in M are defined by parallel induction as follows:*

$$\begin{array}{lll} R_{\Uparrow} & = & \{((x,y),(x',y')) : x' = x \text{ and } y' = succ(y)\} \\ R_{\Downarrow} & = & \{((x,y),(x',y')) : x' = x \text{ and } y' = prec(y)\} \\ R_{\Rightarrow} & = & \{((x,y),(x',y')) : x' = succ(x) \text{ and } y' = y\} \\ R_{\Leftarrow} & = & \{((x,y),(x',y')) : x' = prec(x) \text{ and } y' = y\} \\ R_{\alpha_1;\alpha_2} & = & R_{\alpha_1} \circ R_{\alpha_2} \\ R_{\alpha_1 \cup \alpha_2} & = & R_{\alpha_1} \cup R_{\alpha_2} \\ R_{\alpha^*} & = & (R_\alpha)^* \\ R_{?\varphi} & = & \{((x,y),(x,y)) : M,(x,y) \models \varphi\} \end{array}$$

$$\begin{array}{rcl} M,(x,y) \models p & \Longleftrightarrow & p \in \mathcal{V}(x,y) \\ M,(x,y) \models \mathsf{h}_i & \Longleftrightarrow & \mathcal{P}(i) = (x,y) \\ M,(x,y) \models \neg\varphi & \Longleftrightarrow & M,(x,y) \not\models \varphi \\ M,(x,y) \models \varphi \wedge \psi & \Longleftrightarrow & M,(x,y) \models \varphi \text{ and } M,(x,y) \models \psi \\ M,(x,y) \models [\alpha]\varphi & \Longleftrightarrow & \forall (x',y') \in \mathbb{Z} \times \mathbb{Z} : \text{if } (x,y) R_\alpha (x',y') \\ & & \text{then } M,(x',y') \models \varphi \end{array}$$

When $(x, y)R_\alpha(x', y')$, we will say that position (x', y') is accessible from position (x, y) by program α.[1]

Remark that formulas like h_i behave like nominals in hybrid logics [6], i.e. their truth sets are singletons.

We say that $\varphi \in \mathcal{L}_{\mathsf{DL\text{-}S}^*}$ is valid, denoted by $\models \varphi$, if and only if, for every spatial model M and position (x, y), we have $M, (x, y) \models \varphi$. We say that formula $\varphi \in \mathcal{L}_{\mathsf{DL\text{-}S}^*}$ is satisfiable if and only if $\neg\varphi$ is not valid.

Before going into more technical details let us present a short example illustrating the expressive power of the logic $\mathsf{DL\text{-}S}^*$.

Example 1. *Two robots, called Ann and Bob, are located respectively at position $(0, 0)$ and $(2, 0)$. Suppose there is a danger between the two robots at position $(1, 0)$. This means Ann is located at $(0, 0)$ and has a danger in his right-hand side, while Bob is located at $(2, 0)$ and has a danger in his left-hand side, that is, $M, (0, 0) \models \mathsf{h}_{Ann} \wedge [\Rightarrow]danger$ and $M, (2, 0) \models \mathsf{h}_{Bob} \wedge [\Leftarrow]danger$.*

2.3 Bisimulation

The essential tool we will use to establish our decidability results is the notion of *bounded bisimulation.*

Definition 3. *Fix a set P of atomic propositions, a set I of agents and a set A of spatial programs. Given spatial models $M_1 = (\mathcal{P}_1, \mathcal{V}_1)$ and $M_2 = (\mathcal{P}_2, \mathcal{V}_2)$, $n < \omega$, we define a binary relation $(M_1, \cdot) \leftrightarroweq_n (M_2, \cdot) \subseteq \mathbb{Z}^2 \times \mathbb{Z}^2$ by induction on n as follows.*

We set $(M_1, \vec{x}) \leftrightarroweq_n (M_2, \vec{y})$ if

1. *for every $i \in I$, $\vec{x} = \mathcal{P}_1(i)$ if and only if $\vec{y} = \mathcal{P}_2(i)$,*
2. *for every $p \in P$, $\vec{x} \in \mathcal{V}_1(p)$ if and only if $\vec{y} \in \mathcal{V}_2(p)$, and*

[1] To be more precise, we should define one relation R^M_α per spatial model M. However, we omit the superscript M since it is clear from the context.

3. if $n > 0$, then for every $\alpha \in A$,

Forth_α *Whenever $xR_\alpha\vec{x}'$, there is $\vec{y}'$ such that $\vec{y}R_\alpha\vec{y}'$ and $(M_1,\vec{x}') \leftrightharpoons_{n-1} (M_2,\vec{y}')$, and*

Back_α *Whenever $\vec{y}R_\alpha\vec{y}'$, there is $\vec{x}'$ such that $\vec{x}R_\alpha\vec{y}'$ and $(M_1,\vec{x}') \leftrightharpoons_{n-1} (M_2,\vec{y}')$.*

We may just write $\vec{x} \leftrightharpoons_n \vec{y}$ instead of $(M,\vec{x}) \leftrightharpoons_n (M,\vec{y})$. The following is then standard:

LEMMA 1. *Fix a set P of atomic propositions, a set I of agents and a set of programs A. If $M_1 = (\mathcal{P}_1, \mathcal{V}_1)$ and $M_2 = (\mathcal{P}_2, \mathcal{V}_2)$ are spatial models and $\varphi \in \mathcal{L}_{\mathsf{DL\text{-}S}^*}(P, I, A)$ has modal degree at most n, then whenever $(M_1,\vec{x}) \leftrightharpoons_n (M_2,\vec{y})$, it follows that $M_1,\vec{x} \models \varphi$ if and only if $M_2,\vec{y} \models \varphi$.*

3. UNDECIDABLITY

This section presents results about undecidability for the satisfiability problem of $\mathcal{L}_{\mathsf{DL\text{-}S}^*}(Atm, Agt)$-formulas. Products of linear logics are logics with two (or more) modalities, interpreted over structures very similar to spatial models. Their formulas are equivalent to $\mathcal{L}_{\mathsf{DL\text{-}S}^*}(Atm, Agt)$-formulas over the class of all spatial models and are often undecidable [9, 11, 14]. This suggests that the satisfiability problem of formulas in $\mathcal{L}_{\mathsf{DL\text{-}S}^*}(Atm, Agt)$, as well as some proper fragments, is undecidable as well. The idea is to allow actions only along the horizontal and vertical axes, which following [12] we call the 'compass directions'. To be precise, we define the *language of compass logic of space* by $\mathcal{L}_{\mathsf{CL\text{-}S}^*}(Atm, Agt) = \mathcal{L}_{\mathsf{DL\text{-}S}^*}(Atm, Agt, C)$, where

$$C = \{\Uparrow, \Downarrow, \Rightarrow, \Leftarrow, \Uparrow^*, \Downarrow^*, \Rightarrow^*, \Leftarrow^*\}.$$

As before, we may omit the parameters Atm, Agt when this does not lead to confusion. By $\mathcal{L}_{\mathsf{CL\text{-}F}^*}$ (the language of compass logic of facts) we denote the special case where $Agt = \varnothing$, and similarly $\mathcal{L}_{\mathsf{CL\text{-}P}^*}$ (the language of compass logic of positions) denotes the case where $Atm = \varnothing$.

We start with the following undecidability result for the satisfiability problem of the latter.

THEOREM 1. *The set of valid formulas of $\mathcal{L}_{\mathsf{CL\text{-}F}^*}$ is not computably enumerable.*

PROOF. This follows from Theorem 5.38 in [9], which states (in their notation) that $\mathsf{PTL}_{\Box\circ} \times \mathsf{PTL}_{\Box\circ}$ is not c.e. But this is a notational variant of a fragment of $\mathsf{CL\text{-}F}^*$, where $\circ_1 \approx [\Rightarrow]$, $\Box_1 \approx [\Rightarrow^*]$, $\circ_2 \approx [\Uparrow]$, and $\Box_2 \approx [\Uparrow^*]$. $\Box$

We remark that we only need two of the four compass directions for this proof, provided they are perpendicular. As a corollary, we obtain undecidability of the larger logic.

COROLLARY 1. *The set of valid formulas of $\mathcal{L}_{\mathsf{DL\text{-}S}^*}$ is not computably enumerable.*

In order to study model-checking, we need a finite representation of spatial models. To this aim, we introduce the following definition of *bounded* spatial model of size n. For $(x, y) \in \mathbb{Z}^2$, write $|(x, y)| \le n$ iff $|x| \le n$ and $|y| \le n$.

DEFINITION 4 (BOUNDED SPATIAL MODEL (BSM)). *Let n be a nonnegative integer. A spatial model $M = (\mathcal{P}, \mathcal{V})$ is said to be n-bounded iff for all $i \in Agt$, $|\mathcal{P}(i)| \le n$ and for all $(x, y) \in \mathbb{Z}^2 \times \mathbb{Z}^2$, if $|(x, y)| \not\le n$ then $\mathcal{V}(x, y) = \emptyset$.*

Observe that while the interpretations of variables are bounded, the frame itself is not; we still interpret formulas over $\mathbb{Z} \times \mathbb{Z}$.

As for the class of all models, the restriction to bounded models gives rise to an undecidable set of valid formulas of $\mathcal{L}_{\mathsf{DL\text{-}S}^*}$-formulas:

THEOREM 2. *The set of formulas of $\mathcal{L}_{\mathsf{CL\text{-}F}^*}$ valid over the class of bounded spatial models is not computably enumerable.*

SKETCH OF PROOF. This essentially follows from Corollary 7.18 in [9], which in their notation states that $\mathsf{Log}\{\langle\mathbb{N}, \ge\rangle \times \mathbb{N}, \ge\rangle\}$ is not c.e. As above, this is a notational variant of a fragment of $\mathsf{CL\text{-}F}^*$, where $\Box_1 \approx [\Rightarrow^*]$ and $\Box_2 \approx [\Uparrow^*]$, although interpreted over frames of the form $\{0, \ldots n\} \times \{0, \ldots, m\}$. That it is not c.e. is obtained by reducing the halting problem for Turing machines to $\mathsf{Log}\{\langle\mathbb{N}, \ge\rangle \times \mathbb{N}, \ge\rangle\}$, representing finite computations as finite models. Minor adjustments of this construction can be used, instead, to represent finite computations as *bounded* models. ■

As before, the undecidability of the set of formulas of $\mathcal{L}_{\mathsf{DL\text{-}S}^*}$ valid over the class of bounded spatial models follows. There are different ways to get out of the undecidability of the satisfiability problem of $\mathcal{L}_{\mathsf{DL\text{-}S}^*}$-formulas as highlighted by Corollary 1. One possibility is to consider the star-free fragment of $\mathcal{L}_{\mathsf{DL\text{-}S}^*}$. Another possibility is to study fragments of $\mathcal{L}_{\mathsf{DL\text{-}S}^*}$ that omit atomic propositions and allow only nominals. These two possibilities are explored, respectively, in Sections 5 and 6.

4. MODEL-CHECKING

The model-checking problem for $\mathcal{L}_{\mathsf{CL\text{-}S}^*}(Atm, Agt)$ is the following: let $\varphi \in \mathcal{L}_{\mathsf{CL\text{-}S}^*}(Atm, Agt)$, let n be a nonnegative integer, let M be an n-bounded spatial model and let $(x, y) \in \mathbb{Z} \times \mathbb{Z}$, is it the case that $M, (x, y) \models \varphi$?

In this section we will show that the model-checking problem for $\mathcal{L}_{\mathsf{CL\text{-}S}^*}(Atm, Agt)$ is in PTIME. We use techniques similar to those used for proving that, e.g., model-checking for ordinary modal logic logic or for CTL is also in PTIME [7], but there are some subtleties in dealing with the state-space being infinite (even if the valuations are bounded).

LEMMA 2. *Let n, d be nonnegative integers. Suppose that $x, x', y, y' \in \mathbb{Z}$ are such that one of the following conditions holds:*

- $x > n + d + 1$, $x' = x - 1$ *and* $y' = y$,
- $x < -n - d - 1$, $x' = x + 1$ *and* $y' = y$,
- $y > n + d + 1$, $x' = x$ *and* $y' = y - 1$,
- $y < -n - d + 1$, $x' = x$ *and* $y' = y + 1$.

Then, for any *n-bounded model M, we have that*

$$(M, (x, y)) \leftrightharpoons_d (M, (x', y')).$$

PROOF. Left to the reader. $\Box$

Hence,

LEMMA 3. *Let φ be a $\mathcal{L}_{\mathsf{CL\text{-}S}^*}(Atm, Agt)$-formula, n be a nonnegative integer and M be an n-bounded model. For all integers x, y, we have:*

- *if* $x > n + \deg(\varphi) + 1$ *then* $M, (x, y) \models \varphi$ *iff* $M, (x - 1, y) \models \varphi$,
- *if* $x < -n - \deg(\varphi) - 1$ *then* $M, (x, y) \models \varphi$ *iff* $M, (x + 1, y) \models \varphi$,
- *if* $y > n + \deg(\varphi) + 1$ *then* $M, (x, y) \models \varphi$ *iff* $M, (x, y - 1) \models \varphi$,
- *if* $y < -n - \deg(\varphi) - 1$ *then* $M, (x, y) \models \varphi$ *iff* $M, (x, y + 1) \models \varphi$.

PROOF. By Lemmas 1 and 2. $\square$

Now, for all $\mathcal{L}_{\textsf{CL-S}^*}(Atm, Agt)$-formulas φ and for all integers z, let z_φ be the integer defined by cases as follows:

Case $|z| \leq n + \deg(\varphi) + 1$**:** In that case, let $z_\varphi = z$.

Case $z < -n - \deg(\varphi) - 1$**:** In that case, let $z_\varphi = -n - \deg(\psi) - 1$.

Case $z > n + \deg(\varphi) + 1$**:** In that case, let $z_\varphi = n + \deg(\psi) + 1$.

The reader may easily verify that for all $\mathcal{L}_{\textsf{CL-S}^*}(Atm, Agt)$-formulas φ and for all integers z, $|z_\varphi| \leq n + \deg(\varphi) + 1$. Now, given a $\mathcal{L}_{\textsf{CL-S}^*}(Atm, Agt)$-formula φ, let $(\varphi_1, \ldots, \varphi_N)$ be an enumeration of the set of all φ's subformulas. Let us assume that for all $a, b \in \{1, \ldots, N\}$, if φ_a is a strict subformula of φ_b then $a < b$. For all $a \in \{1, \ldots, N\}$ and for all $(x, y) \in \mathbb{Z}^2 \times \mathbb{Z}^2$, if $|(x, y)| \leq n + \deg(\varphi_a) + 1$ then we will associate a truth value $tv(a, x, y)$ by case as follows:

Case $\varphi_a = p$**:** In that case, let $tv(a, x, y) =$"$(x, y) \in V(p)$".

Case $\varphi_a = h_i$**:** In that case, $tv(a, x, y) =$"$(x, y) = P(i)$".

Case $\varphi_a = \bot$**:** In that case, let $tv(a, x, y) = \bot$.

Case $\varphi_a = \neg\psi$**:** Let $b \in \{1, \ldots, a\}$ be such that $\psi = \varphi_b$. Remind that $b < a$. In that case, if $tv(b, x, y) = \bot$ then let $tv(a, x, y) = \top$ else let $tv(a, x, y) = \top$.

Case $\varphi_a = \psi \vee \chi$**:** Let $b, c \in \{1, \ldots, a\}$ be such that $\psi = \varphi_b$ and $\chi = \varphi_c$. Remind that $b, c < a$. In that case, if $tv(b, x_\psi, y_\psi) = \bot$ and $tv(c, x_\chi, y_\chi) = \bot$ let $tv(a, x, y) = \bot$ else let $tv(a, x, y) = \top$.

Case $\varphi_a = [\Rightarrow]\psi$**:** Let $b \in \{1, \ldots, a\}$ be such that $\psi = \varphi_b$. Remind that $b < a$. In that case, let $tv(a, x, y) = tv(b, (x + 1)_\psi, y_\psi)$.

Cases $\varphi_a = [\Uparrow]\psi$**,** $\varphi_a = [\Leftarrow]\psi$ **and** $\varphi_a = [\Downarrow]\psi$**:** Similar to the previous case.

Case $\varphi_a = [\Rightarrow^*]\psi$**:** Let $b \in \{1, \ldots, a\}$ be such that $\psi = \varphi_b$. Remind that $b < a$. In that case, if $tv(b, z_\psi, y_\psi) = \bot$ for some integer $z \geq x$ then let $tv(a, x, y) = \bot$ else let $tv(a, x, y) = \top$.

Cases $\varphi_a = [\Uparrow^*]\psi$**,** $\varphi_a = [\Leftarrow^*]\psi$ **and** $\varphi_a = [\Downarrow^*]\psi$**:** Similar to the previous case.

Obviously, within a polynomial time with respect to $\parallel \varphi \parallel$, one can deterministically compute the truth values $tv(a, x, y)$ for $a \in \{1, \ldots, N\}$ and for $(x, y) \in \mathbb{Z}^2 \times \mathbb{Z}^2$ such that $|(x, y)| \leq n + \deg(\varphi_a) + 1$. Consequently,

THEOREM 3. *The model-checking problem for $\mathcal{L}_{\textsf{CL-S}^*}(Atm, Agt)$ is decidable in deterministic polynomial time.*

5. STAR-FREE FRAGMENTS

In this section and the next, we identify two decidable fragments. The first is obtained by restricting the language to $\mathcal{L}_{\textsf{DL-S}}(Atm, Agt)$, as given by the following grammar:

$$\begin{array}{lcl} \alpha & ::= & \Uparrow \mid \Downarrow \mid \Rightarrow \mid \Leftarrow \mid \alpha;\alpha' \mid \alpha \cup \alpha' \mid ?\varphi \\ \varphi & ::= & p \mid \mathsf{h}_i \mid \neg\varphi \mid \varphi \wedge \psi \mid [\alpha]\varphi \end{array}$$

We denote the set of valid formulas of $\mathcal{L}_{\textsf{DL-S}}(Atm, Agt)$ by $\textsf{DL-S}$. The second is the fragment $\mathcal{L}^0_{\textsf{DL-S}}(Atm, Agt)$ given by:

$$\begin{array}{lcl} \alpha & ::= & \Uparrow \mid \Downarrow \mid \Rightarrow \mid \Leftarrow \\ \varphi & ::= & p \mid \mathsf{h}_i \mid \neg\varphi \mid \varphi \wedge \psi \mid [\alpha]\varphi \end{array}$$

The corresponding set of valid formulas will be denoted $\textsf{DL-S}^0$. Note that $\mathcal{L}_{\textsf{DL-S}}(Atm, Agt)$ can be reduced to $\mathcal{L}^0_{\textsf{DL-S}}(Atm, Agt)$:

LEMMA 4. *Every formula $\varphi \in \mathcal{L}_{\textsf{DL-S}}(Atm, Agt)$ is equivalent to some $\varphi^0 \in \mathcal{L}^0_{\textsf{DL-S}}(Atm, Agt)$.*

PROOF. It suffices to observe that the following are valid:

$$[\alpha;\alpha']\psi \leftrightarrow [\alpha][\alpha']\psi$$
$$[\alpha \cup \alpha']\psi \leftrightarrow [\alpha]\psi \wedge [\alpha']\psi$$
$$[?\theta]\psi \leftrightarrow (\theta \rightarrow \psi).$$

With these validities, any formula of $\mathcal{L}_{\textsf{DL-S}}(Atm, Agt)$ can be recursively reduced to an equivalent formula in the language $\mathcal{L}^0_{\textsf{DL-S}}(Atm, Agt)$. $\square$

Our decidability proof will be based on a small model property, obtained by truncating a larger model. Fix a natural number n. Given a model $M = (\mathcal{P}, \mathcal{V})$, we define $M \upharpoonright n = (\mathcal{P} \upharpoonright n, \mathcal{V} \upharpoonright n)$.

- $(\mathcal{P} \upharpoonright n)(i) = \begin{cases} \mathcal{P}(i) & \text{if } |\mathcal{P}(i)| \leq n; \\ (n+1, 0) & \text{otherwise.} \end{cases}$
- $(\mathcal{V} \upharpoonright n)(p) = \mathcal{V}(p) \cap \big([-n, n] \times [-n, n]\big)$.

Observe that $M \upharpoonright n$ is $(n + 1)$-bounded. As a result, when one restricts the discussion to the set of all programs of $\mathcal{L}^0_{\textsf{DL-S}}(Atm, Agt)$,

LEMMA 5. *For all $\vec{x} \in \mathbb{Z}^2 \times \mathbb{Z}^2$, if $|\vec{x}| \leq m \leq n$, then $(M, \vec{x}) \leftrightarrows_{n-m} (M \upharpoonright n, \vec{x})$.*

PROOF. The proof proceeds by a standard induction on m. The atoms and position clauses are trivial since $x \leq n$ and the values of atomic propositions is not changed. For the inductive case, consider (for example) $\alpha = \Rightarrow$. Then, if $\vec{x} = (x_0, x_1)$, $\vec{x} R_{\Rightarrow} \vec{y}$ if and only if $\vec{y} = (x_0 + 1, x_1)$. Clearly $|\vec{y}| \leq m + 1$, so that by the induction hypothesis, $(M, \vec{x}) \leftrightarrows_{n-m-1} (M \upharpoonright n, \vec{y})$, as needed. $\square$

With this we obtain our first decidability result.

THEOREM 4. *The logics $\textsf{DL-S}^0, \textsf{DL-S}$ are decidable. In particular, $\textsf{DL-S}^0$ is in* NP.

PROOF. Since $\textsf{DL-S}$ can be reduced to $\textsf{DL-S}^0$, it suffices to show that the latter is decidable. Suppose that φ is satisfied on some model M; without loss of generality, we can assume that φ is satisfied on the origin. Let n be the modal degree of φ. By Lemma 5, $(M, \vec{0}) \leftrightarrows_n (M \upharpoonright n, \vec{0})$, so by Lemma 1, φ

is also satisfied on $(M \restriction n, \vec{0})$. It follows that φ is satisfiable if and only if it is satisfiable on the class of models such that $\mathcal{P}$ and $\mathcal{V}$ are both $(n+1)$-bounded, so it remains to enumerate all such models and check whether any of them satisfy φ. Note that the size of any $(n+1)$-bounded model is $o(n^2)$, so the complexity bound for $\mathsf{DL\text{-}S}^0$ follows. $\square$

Observe that it does not follow from our techniques that DL-S is in NP, since the reduction procedure is not polynomial.

Now, our aim in this section will be to completely axiomatize $\mathsf{DL\text{-}S}^0$. In this respect, we need the following axioms and inference rules:

- All axioms and inference rules saying that $[\Uparrow]$, $[\Downarrow]$, $[\Rightarrow]$ and $[\Leftarrow]$ are normal modalities,
- $[\alpha]\varphi \leftrightarrow \langle\alpha\rangle\varphi$ for each $\alpha \in \{\Uparrow, \Downarrow, \Rightarrow, \Leftarrow\}$,
- $\varphi \to [\Uparrow]\langle\Downarrow\rangle\varphi$ and $\varphi \to [\Downarrow]\langle\Uparrow\rangle\varphi$,
- $\varphi \to [\Rightarrow]\langle\Leftarrow\rangle\varphi$ and $\varphi \to [\Leftarrow]\langle\Rightarrow\rangle\varphi$,
- $[\alpha_1][\alpha_2]\varphi \leftrightarrow [\alpha_2][\alpha_1]\varphi$ for each $\alpha_1, \alpha_2 \in \{\Uparrow, \Downarrow, \Rightarrow, \Leftarrow\}$,
- $h_i \to [\Uparrow]^x[\Rightarrow]^y\neg h_i$ for each integers x, y such that $x \neq 0$ or $y \neq 0$.

We will say that a formula $\varphi \in \mathcal{L}^0_{\mathsf{DL\text{-}S}}(Atm, Agt)$ is *derivable* if it belongs to the least set of $\mathcal{L}^0_{\mathsf{DL\text{-}S}}(Atm, Agt)$-formulas containing the above axioms and closed under the above inference rules.

Theorem 5. *let φ be an $\mathcal{L}^0_{\mathsf{DL\text{-}S}}(Atm, Agt)$-formula. The following conditions are equivalent: (i) φ is derivable; (ii) φ is valid.*

Proof. (i)$\Rightarrow$(ii): It suffices to check that all axioms are valid and that all inference rules preserve valitity.
(ii)$\Rightarrow$(i): Suppose φ is not derivable. Let d denote the modal degree of φ. By Lindenbaum's Lemma, let Γ be a maximal consistent set of formulas such that $\varphi \notin \Gamma$. Remark that for all $i \in Agt$, there exists at most one pair (x, y) of (negative or positive) integers such that $[\Uparrow]^x[\Rightarrow]^y h_i \in \Gamma$. Let $Agt(\Gamma)$ be the set of all $i \in Agt$ such that $[\Uparrow]^x[\Rightarrow]^y h_i \in \Gamma$ for some pair (x, y) of integers such that $|(x, y)| \leq d$. Let $M = (\mathcal{P}, \mathcal{V})$ be the spatial model defined as follows:

- For all $i \in Agt$, if $i \in Agt(\Gamma)$ then let $\mathcal{P}(i)$ be the unique pair (x, y) of integers such that $[\Uparrow]^x[\Rightarrow]^y h_i \in \Gamma$, else let $\mathcal{P}(i)$ be $(d+1, 0)$,
- for all pairs (x, y) of integers, if $|(x, y)| \leq d$ then let $\mathcal{V}(x, y) = \{p \in Atm : [\Rightarrow]^x[\Uparrow]^y p \in \Gamma\}$, else let $\mathcal{V}(x, y) = \emptyset$.

The reader may easily prove by induction on ψ that if ψ is a subformula of φ then for all pairs (x, y) such that $|(x, y)| \leq deg(\varphi) - deg(\psi)$, $M, (x, y) \models \psi$ iff $[\Rightarrow]^x[\Uparrow]^y\psi \in \Gamma$. Since $\varphi \notin \Gamma$, therefore $M, (0, 0) \not\models \varphi$. Thus, φ is not valid. $\square$

6. COMPASS LOGIC OF POSITIONS

Next we consider the fragment $\mathcal{L}_{\mathsf{CL\text{-}P^*}}$, defined in Section 3. Since there are no atomic propositions, models are somewhat simpler.

Definition 5. *A* position model *is a function $\mathcal{P} \colon Agt \to \mathbb{Z} \times \mathbb{Z}$.*

That is, a position model is just a spatial model without a valuation for atomic propositions. As we will show, position models do not need to have big 'gaps' if we only care about satisfiability of $\mathcal{L}_{\mathsf{CL\text{-}P^*}}$-formulas. This will give us a small model property.

Definition 6. *Let $\mathcal{P}$ be a position model. A* vertical gap *is a set $G = [a, b] \times \mathbb{Z}$ such that for all $i \in Agt$, $\mathcal{P}(i) \notin G$. If $(x, y) \in G$, we say that the* depth *of (x, y) in G is $\min(x - a, b - x)$, and G_m denotes the set of elements of depth at least m; observe that $G_0 = G$, and G_m is also a gap when non-empty. The* removal *of G is the function ρ given by $\rho(x, y) = (x', y)$ where $x' = x$ if $x \leq a$, $x' = \min(a, x - (b - a))$ otherwise.*

A horizontal gap *is defined analogously, but is of the form $\mathbb{Z} \times [a, b]$. The depth and the removal are defined analogously as well.*

In this section we use $\leftrightarrows_n$ for n-bisimilarity with respect to all basic relations of $\mathcal{L}_{\mathsf{CL\text{-}P^*}}$.

Lemma 6. *Let $G = [a, b] \times \mathbb{Z}$ be a vertical gap and $\mathcal{P}$ be a position model. Then, if $(x, y), (x', y) \in G_m$, it follows that $\mathcal{P}, (x, y) \leftrightarrows_m \mathcal{P}, (x', y)$.*

The analogous claim holds for horizontal gaps.

Proof. We proceed by induction on m. The atomic clauses are straightforward since, if $(x, y), (x', y) \in G_0 = G$, then they satisfy no atoms.

For the other clauses, assume the claim inductively for m, and suppose that $(x, y), (x', y) \in G_{m+1}$. Any 'vertical' program $(\Uparrow, \Downarrow, \Uparrow^*, \Downarrow^*)$ stays within $G_{m+1} \subseteq G_m$ so we may immediately apply the induction hypothesis. For example, if $(u, v)R_{\Uparrow^*}(x, y)$, then $u = x$ and $v \geq y$; hence, $(x', y)R_{\Uparrow^*}(x', v)$ and by the induction hypothesis, $(u, v) = (x, v) \leftrightarrows_m (x', v)$. The 'back' clauses and the rest of the vertical programs are entirely symmetrical.

Next consider a 'horizontal' program: $\Leftarrow, \Rightarrow, \Leftarrow^*, \Rightarrow^*$. By symmetry, we will only consider the 'forth' clauses of the 'right' programs. We have that $R_\Rightarrow$ is a function; specifically, $R_\Rightarrow(x, y) = (x+1, y)$. Observe that $(x+1, y) \in G_m$, and similarly $(x'+1, y) \in G_m$. But, by the induction hypothesis, $(x+1, y) \leftrightarrows_m (x'+1, y) = R_\Rightarrow(x', y)$, as needed.

Now suppose that $(x, y)R_{\Rightarrow^*}(u, v)$, so that $u \geq x$ and $v = y$. We consider two cases. If also $u \geq x'$, then we also have that $(x', y)R_{\Rightarrow^*}(u, v)$, and we may use the same witness. Otherwise, $x \leq u < x'$, which means that $(u, v) \in G_{m+1} \subseteq G_m$, so by the induction hypothesis $(x, v) \leftrightarrows_m (x', y)$. But also, $(x', y)R_{\Rightarrow^*}(x', y)$, and we can use it as our witness.

As mentioned, the other clauses are entirely symmetrical and left to the reader. By induction on m, the claim follows. The analogous claim for horizontal gaps is also entirely analogous. $\square$

Lemma 7. *Let $G = [a - m, b + m] \times \mathbb{Z}$ be a vertical gap and ρ the removal of G_m. Then, $\mathcal{P}, \vec{x} \leftrightarrows_m \rho\mathcal{P}, \rho(\vec{x})$.*

Proof. We prove, by induction on $k \leq m$, that $\mathcal{P}, \vec{x} \leftrightarrows_k \rho\mathcal{P}, \rho(\vec{x})$. For $k = 0$ this is clear, since if $(x, y) \in G_m$, no nominal occurs on (x, y) or on $\rho(x, y) = (a, y)$. Otherwise, $(x, y) = \mathcal{P}(i)$ if and only if $\rho(x, y) = \rho\mathcal{P}(i)$.

Now, assume the claim for k, and let $\rho(x, y) = (x', y)$. The 'forth' clauses for $\alpha \in \{R_\Uparrow, R_\Downarrow, R^*_\Uparrow, R^*_\Downarrow\}$ follow by observing that if $(x, y)R_\alpha(u, v)$, then $\rho(x, y)R_\alpha\rho(u, v)$; for example, if

$\alpha = \Downarrow$, then we must have $u = x$ and $v = y - 1$, and since ρ fixes the y coordinate we have that if $\rho(x, y) = (x', y)$, then $\rho(u, v) = (x', y - 1)$, as needed. Similarly, for the 'back' clause, if $\rho(x, y) = (x', y)$ and $(x', y)R_\alpha(u, v)$, we must have $u = x'$ and can readily observe that $(x, y)R_\alpha(x, v)$ and $\rho(x, v) = (x', v)$, so that by the induction hypothesis, $\mathcal{P}, (x, v) \leftrightarrows_k \rho\mathcal{P}, \rho(x', v)$.

Next we look at $\alpha \in \{\Leftarrow, \Rightarrow, \Leftarrow^*, \Rightarrow^*\}$. First, we check the 'forth' clauses. If $(x, y)R_\Leftarrow(u, v)$, then $u = x - 1$ and $v = y$. If $x \notin (a, b]$, then it readily follows that $\rho(x, y)R_\Leftarrow\rho(u, v)$, and we may use the induction hypothesis. If instead $x \in (a, b]$, then $\rho(u, v) = \rho(x, y) = (a, y)$. However, $R_\Leftarrow(a, y) = (a - 1, y) \in G_{m-1}$, so by the induction hypothesis and Lemma 6,

$$\rho\mathcal{P}, (a - 1, y) \leftrightarrows_k \mathcal{P}, (a - 1, y) \leftrightarrows_k \mathcal{P}, (x - 1, y),$$

as needed. For $\alpha = \Leftarrow^*$, suppose $(x, y)R_{\Leftarrow^*}(u, v)$. Then, $y = v$, and since ρ is non-decreasing on the first component, we also have $\rho(x, y)R_{\Leftarrow^*}\rho(u, v)$. The cases for the 'right' programs are similar.

Finally, we check the 'back' clauses for the horizontal programs. Observe that $R_\Leftarrow, R_\Rightarrow$ are functional, so the 'forth' and 'back' clauses are equivalent. Hence we consider only $R_{\Leftarrow^*}, R_{\Rightarrow^*}$. If $(x', y)R_{\Leftarrow^*}(u, y)$, then consider two cases. If $u \le a$, then $\rho(u, y) = (u, y)$ and $u \le x' \le x$, so we have that $(x, y)R_{\Leftarrow^*}(u, y)$ and we may use the induction hypothesis on (u, y). If $u > a$, then $\rho(u + b - a, y) = (u, y)$, and we may use the induction hypothesis on $(u + b - a, y)$. But note that, in this case, we must have that $x = x' + b - a$, so $(x, y)R_{\Leftarrow^*}(u + b - a, y)$.

Finally, if $(x', y)R_{\Rightarrow^*}(u, y)$, again consider two cases. If $u < a$, then $\rho(u, y) = (u, y)$ and $u \ge x' = x$, so we have that $(x, y)R_{\Rightarrow^*}(u, y)$ and we may use the induction hypothesis on (u, y). If $u \ge a$, then $\rho(u + b - a, y) = (u, y)$, and we may use the induction hypothesis on $(u + b - a, y)$. Note that, in this case, $x \le x' + b - a \le u + b - a$, so $(x, y)R_{\Rightarrow^*}(u, y)$, as needed.

The case for a horizontal gap is similar. $\square$

THEOREM 6. *If $\varphi \in \mathcal{L}_{\textsf{CL-P}^*}$ is satisfiable, it is satisfiable on a position model where all coordinates of positions are bounded by* $2(|\varphi| + 1)^2$.

PROOF. Assume that φ is satisfied on some position model $\mathcal{P}$. Suppose that $x_1 \le \ldots \le x_n$ are the x-coordinates of all positions of agents such that h_i appears in φ, together with the evaluation point, $(0, 0)$ (note that $n \le |\varphi| + 1$). If for some $i < n$ we have that $x_{i+1} - x_i > 2(|\varphi| + 1)$, then $G = (x_i, x_{i+1}) \times \mathbb{Z}$ is a horizontal strip with $G_{|\varphi|}$ having width at least two, so that its removal is not the identity.

Now, if the x_i's are not bounded by $2(|\varphi| + 1)^2$, note that such a gap must exist so we can remove it. After enough iterations, we can bound all x_i's. Then we proceed to bound the vertical components analogously. $\square$

THEOREM 7. *Satisfiability of $\mathcal{L}_{\textsf{CL-P}^*}$-formulas is decidable in* NP.

PROOF. We can decide the satisfiability of φ by guessing a position model $\mathcal{P}$ with all coordinates bounded by $2(|\varphi|+1)^2$ and model-checking whether φ holds at $(0, 0)$. $\square$

In the next section, we will extend the static view of the two-dimensional space by a dynamic component allowing agents to move.

7. SPACE AND MOVEMENT

DL-S studied in the previous sections is a logic for representing static properties of the bidimensional space. Specifically, in DL-S, positions of agents in the space do not change. The aim of this section is to extend $\mathcal{L}_{\textsf{DL-S}}(Atm, Agt)$ by programs describing the agents' movements in the bidimensional space. We assume that agents act in a synchronous way (i.e., they act in parallel). We call the resulting language $\mathcal{L}_{\textsf{DL-SM}}(Atm, Agt)$ and the resulting logic DL-SM (Dynamic Logic of Space and Moving).

7.1 Syntax

In $\mathcal{L}_{\textsf{DL-SM}}(Atm, Agt)$, agent i is associated with her corresponding repertoire of actions $Act_i = \{\Uparrow_i, \Downarrow_i, \Leftarrow_i, \Rightarrow_i, nil_i\}$. $\Uparrow_i$ is agent i' action of moving up, $\Downarrow_i$ is agent i' action of moving down, $\Leftarrow_i$ is agent i' action of moving left, $\Rightarrow_i$ is agent i' action of moving right and nil_i is agent i's action of doing nothing.

The set of joint of actions is defined to be $\Delta = \prod_{i \in Agt} Act_i$. Elements of Δ are denoted by $\delta, \delta', \ldots$ For every $\delta \in \Delta$, δ_i denotes the element in δ corresponding to agent i.

Since the logic DL-S* is undecidable, we start from its decidable star-free fragment as the basis of our dynamic extension by programs describing the agents' movements.

The language, denoted by $\mathcal{L}_{\textsf{DL-SM}}(Atm, Agt)$, is defined by the following grammar in Backus-Naur Form:

$$\begin{array}{lcl} \alpha & ::= & \Uparrow \mid \Downarrow \mid \Rightarrow \mid \Leftarrow \mid \alpha; \alpha' \mid \alpha \cup \alpha' \mid ?\varphi \\ \beta & ::= & \delta \mid \beta; \beta' \mid \beta \cup \beta' \mid ?\varphi \\ \varphi & ::= & p \mid \mathsf{h}_i \mid \neg\varphi \mid \varphi \wedge \psi \mid [\alpha]\varphi \mid [\beta]\varphi \end{array}$$

where p ranges over Atm and i ranges over Agt. Instances of β are called *movement programs.*

7.2 Semantics

The semantics is a model update semantics as in the style of dynamic epistemic logic (DEL) [17].

DEFINITION 7 ($R^{(x,y)}_\beta$ AND TRUTH CONDITIONS). *Let $M \in \mathbf{M}$ be a spatial program. For all movement programs β, for all formula φ and for all positions (x, y), the binary relation $R^{(x,y)}_\beta$ on $\mathbf{M} \times \mathbf{M}$ and the truth conditions of φ in M are defined by parallel induction as follows. (We only give the truth condition for $[\beta]\varphi$ as the truth conditions for the boolean constructs and for $[\alpha]\varphi$ are as in DL-S*):*

$$\begin{array}{lcl} R^{(x,y)}_\delta & = & \{(M, M') : \mathcal{V}' = \mathcal{V} \textit{ and } \forall i \in Agt, \mathcal{P}'(i) = \mathcal{P}^{\delta_i}(i)\} \\ R^{(x,y)}_{\beta_1;\beta_2} & = & R^{(x,y)}_{\beta_1} \circ R^{(x,y)}_{\beta_2} \\ R^{(x,y)}_{\beta_1\cup\beta_2} & = & R^{(x,y)}_{\beta_1} \cup R^{(x,y)}_{\beta_2} \\ R^{(x,y)}_{?\varphi} & = & \{(M, M) : M, (x, y) \models \varphi\} \end{array}$$

where:

$$\begin{array}{lll} \mathcal{P}^{\delta_i}(i) = (x, succ(y)) & \textit{if} & \delta_i = \Uparrow_i \textit{ and } \mathcal{P}(i) = (x, y) \\ \mathcal{P}^{\delta_i}(i) = (x, prec(y)) & \textit{if} & \delta_i = \Downarrow_i \textit{ and } \mathcal{P}(i) = (x, y) \\ \mathcal{P}^{\delta_i}(i) = (succ(x), y) & \textit{if} & \delta_i = \Rightarrow_i \textit{ and } \mathcal{P}(i) = (x, y) \\ \mathcal{P}^{\delta_i}(i) = (prec(x), y) & \textit{if} & \delta_i = \Leftarrow_i \textit{ and } \mathcal{P}(i) = (x, y) \\ \mathcal{P}^{\delta_i}(i) = (x, y) & \textit{if} & \delta_i = nil_i \textit{ and } \mathcal{P}(i) = (x, y) \end{array}$$

$$\begin{array}{lcl} M, (x, y) \models [\beta]\varphi & \Longleftrightarrow & \forall (M, M') \in \mathbf{M} \times \mathbf{M} : \textit{if } MR_\beta M' \\ & & \textit{then } M', (x, y) \models \varphi \end{array}$$

Definitions of validity and satisfiability for DL-SM generalize those for DL-S* in a straighforward manner.

Before going into more technical details, let us illustrate the expressive power of DL-SM with the aid of the short example of Section 2.

EXAMPLE 2. *(Cont.) Ann and Bob want to meet at the same spatial position in a safe (non-dangerous) place. It turns out that the sequence of joint actions*
$$\beta_0 = (\Uparrow_{Ann}, \Uparrow_{Bob}); (\Rightarrow_{Ann}, \Leftarrow_{Bob})$$
leads the two robots to achieve their objective by meeting at the safe location $(1,1)$. *Indeed, we have* $M,(1,1) \models [(\Uparrow_{Ann}, \Uparrow_{Bob}); (\Rightarrow_{Ann}, \Leftarrow_{Bob})](\mathsf{h}_{Ann} \wedge \mathsf{h}_{Bob} \wedge \neg danger)$.

7.3 Decidability and axiomatization

The aim of this section is to show how the satisfiability problem of DL-SM can be reduced to the satisfiability problem of DL-S. Given the decidability result and the complete axiomatization for the latter of Section 5, this reduction will provide a decidability result as well as an axiomatization for the former.

PROPOSITION 1. *The following* $\mathcal{L}_{\textsf{DL-SM}}(Atm, Agt)$*-formulas are valid:*

$$[\alpha;\alpha']\varphi \leftrightarrow [\alpha][\alpha']\varphi \quad (1)$$
$$[\alpha\cup\alpha']\varphi \leftrightarrow ([\alpha]\varphi \wedge [\alpha']\varphi) \quad (2)$$
$$[\beta;\beta']\varphi \leftrightarrow [\beta][\beta']\varphi \quad (3)$$
$$[\beta\cup\beta']\varphi \leftrightarrow ([\beta]\varphi \wedge [\beta']\varphi) \quad (4)$$
$$[?\varphi]\psi \leftrightarrow (\varphi \rightarrow \psi) \quad (5)$$
$$[\delta]p \leftrightarrow p \quad (6)$$
$$[\delta]\mathsf{h}_i \leftrightarrow [F_i(\delta)]\mathsf{h}_i \quad (7)$$
$$[\delta]\neg\varphi \leftrightarrow \neg[\delta]\varphi \quad (8)$$
$$[\delta](\varphi\wedge\psi) \leftrightarrow ([\delta]\varphi \wedge [\delta]\psi) \quad (9)$$
$$[\delta][\alpha]\varphi \leftrightarrow [\alpha][\delta]\varphi \quad (10)$$

with $\alpha \in \{\Uparrow, \Downarrow, \Rightarrow, \Leftarrow\}$ *and where the function* F_i *is defined as follows:*

$$F_i(\delta) = \Uparrow \text{ if } \delta_i = \Downarrow_i$$
$$F_i(\delta) = \Downarrow \text{ if } \delta_i = \Uparrow_i$$
$$F_i(\delta) = \Rightarrow \text{ if } \delta_i = \Leftarrow_i$$
$$F_i(\delta) = \Leftarrow \text{ if } \delta_i = \Rightarrow_i$$
$$F_i(\delta) = ?\top \text{ if } \delta_i = nil_i$$

As the following rule of replacement of equivalents preserves validity:

$$\frac{\psi_1 \leftrightarrow \psi_2}{\varphi \leftrightarrow \varphi[\psi_1/\psi_2]} \quad (11)$$

the equivalences of Proposition 1 together with this allow to find for every $\mathcal{L}_{\textsf{DL-SM}}(Atm, Agt)$-formula an equivalent formula of $\mathcal{L}_{\textsf{DL-S}}(Atm, Agt)$ studied in Section 5. Call *red* the mapping which iteratively applies the equivalences of Proposition 1 from the left to the right, starting from one of the innermost modal operators. *red* pushes the dynamic operators $[\beta]$ inside the formula, and finally eliminates them when facing an atomic formula The mapping *red* is inductively defined by:

1. $red(p) = p$
2. $red(\mathsf{h}_i) = \mathsf{h}_i$
3. $red(\neg\varphi) = \neg red(\varphi)$
4. $red(\varphi\wedge\psi) = red(\varphi) \wedge red(\psi)$
5. $red([\alpha]\varphi) = [\alpha]red(\varphi)$ with $\alpha \in \{\Uparrow, \Downarrow, \Rightarrow, \Leftarrow\}$
6. $red([\alpha;\alpha']\varphi) = [\alpha][\alpha']red(\varphi)$
7. $red([\alpha\cup\alpha']\varphi) = ([\alpha]red(\varphi) \wedge [\alpha']red(\varphi))$
8. $red([?\varphi]\psi) = red(\neg(\varphi \wedge \neg\psi))$
9. $red([\delta]p) = p$
10. $red([\delta]\mathsf{h}_i) = [F_i(\delta)]\mathsf{h}_i$
11. $red([\delta]\neg\varphi) = red(\neg[\delta]\varphi)$
12. $red([\delta](\varphi\wedge\psi)) = red([\delta]\varphi \wedge [\delta]\psi)$
13. $red([\delta][\alpha]\varphi) = red([\alpha][\delta]\varphi)$ with $\alpha \in \{\Uparrow, \Downarrow, \Rightarrow, \Leftarrow\}$
14. $red([\beta;\beta']\varphi) = [\beta][\beta']red(\varphi)$
15. $red([\beta\cup\beta']\varphi) = ([\beta]red(\varphi) \wedge [\beta']red(\varphi))$

We can state the following proposition.

PROPOSITION 2. *Let* $\varphi \in \mathcal{L}_{\textsf{DL-SM}}(Atm, Agt)$. *Then,* $\varphi \leftrightarrow red(\varphi)$ *is valid.*

Decidability of the satisfiability problem of DL-SM follows straightforwardly from the decidability of the star-free fragment $\mathcal{L}_{\textsf{DL-S}}(Atm, Agt)$ of DL-S* (Theorem 4). Indeed, *red* provides an effective procedure for reducing a formula φ in $\mathcal{L}_{\textsf{DL-SM}}(Atm, Agt)$ into an equivalent formula $red(\varphi)$ in $\mathcal{L}_{\textsf{DL-S}}(Atm, Agt)$.

THEOREM 8. *The logic DL-SM is decidable.*

Thanks to the completeness result for the star-free fragment of DL-SM and the reduction axioms of Proposition 1, we can state the following theorem.

THEOREM 9. *The logic DL-SM is completely axiomatized by the axioms and rules of inference of the star-free fragment of DL-SM given in Section 5, the valid formulas of Proposition 1 and the rule of replacement of equivalents.*

8. PERSPECTIVES

Before concluding the paper, we discuss two perspectives for the extension of the logic DL-S* and DL-SM by concepts of perceptual knowledge and coalitional capability.

Perceptual knowledge.

DL-S* and DL-SM support reasoning about properties of the 2D space as well as about positions and movements of agents in the 2D space. However, an agent in the space does not only move but also sees where other agents are, how the space around her is, what other agents do, etc. More generally, agents in the space have perceptual knowledge (i.e., knowledge based on what they see). We want to propose here a simple extension of DL-S* and DL-SM by modal operators of perceptual knowledge. Specifically, we consider epistemic-like operators of type S_i^k describing what an agent could see from her current position, if she had a range of vision of size $k \in \mathbb{N}$. An agent's range of vision of size k

corresponds to the square centered at the agent's position with side length equal to $2 \times k$. We call the latter agent i's *neighborhood* of size k.

In order to provide an interpretation of the operator S_i^k, the following concept of indistinguishibility is required. Let $i \in Agt$ and let $M = (\mathcal{P}, \mathcal{V})$ and $M' = (\mathcal{P}', \mathcal{V}')$ be two spatial models. We say that M and M' are *indistinguishable* for agent i given her current position and her range of vision of size k, denoted by $M \sim_i^k M'$, if and only if:

$$\begin{array}{rcl} \mathcal{V}'(x,y) & = & \mathcal{V}(x,y) \\ \mathcal{P}'(j) & = & \mathcal{P}(j) \end{array}$$

for all $(x,y) \in \mathbb{Z} \times \mathbb{Z}$ and for all $j \in Agt$ such that $(x,y) \in \mathcal{D}(i,k)$ and $\mathcal{P}(j) \in \mathcal{D}(i,k)$ with

$$\begin{aligned} \mathcal{D}(i,k) = & \{(x',y') : \mathcal{P}_x(i) - k \leq x' \leq \mathcal{P}_x(i) + k \text{ and} \\ & \mathcal{P}_y(i) - k \leq y' \leq \mathcal{P}_y(i) + k\} \end{aligned}$$

where $\mathcal{P}_x(i)$ and $\mathcal{P}_y(i)$ are, respectively, the x-coordinate and the y-coordinate in $\mathcal{P}(i)$.

This notion of indistinguishibility is essential to provide a truth condition of the formula $\mathsf{S}_i^k\varphi$ that has to be read "if agent i had a range of vision of size k, then i could see that φ is true from her current position". Let M be a spatial model and let $(x,y) \in \mathbb{Z} \times \mathbb{Z}$. Then:

$$\begin{aligned} M,(x,y) \models \mathsf{S}_i^k\varphi \iff & \forall M' \in \mathbf{M} : \text{if } M \sim_i^k M' \\ & \text{then } M', \mathcal{P}(i) \models \varphi \end{aligned}$$

It is easy to check that $\sim_i^k$ is an equivalence relation. However, this does not imply that the operator S_i^k is an S5 modality, as its interpretation requires to change the reference point from position (x,y) to position $\mathcal{P}(i)$. Nonetheless, S_i^k satisfies the principles of the system KD45, namely:

$$(\mathsf{S}_i^k\varphi \wedge \mathsf{S}_i^k(\varphi \to \psi)) \to \mathsf{S}_i^k\psi \tag{12}$$

$$\neg(\mathsf{S}_i^k\varphi \wedge \mathsf{S}_i^k\neg\varphi) \tag{13}$$

$$\mathsf{S}_i^k\varphi \to \mathsf{S}_i^k\mathsf{S}_i^k\varphi \tag{14}$$

$$\neg\mathsf{S}_i^k\varphi \to \mathsf{S}_i^k\neg\mathsf{S}_i^k\varphi \tag{15}$$

$$\frac{\varphi}{\mathsf{S}_i^k\varphi} \tag{16}$$

as well as the following form of "local reflexivity" property:

$$(\mathsf{h}_i \wedge \mathsf{S}_i^k\varphi) \to \varphi \tag{17}$$

S_i^k satisfies additional principles that are proper to its spatial interpretation. For instance, let

$$\begin{aligned} \mathcal{P}rg(k) = & \{\Leftarrow^h\Uparrow^h : 0 \leq h \leq k\} \cup \{\Rightarrow^h\Uparrow^h : 0 \leq h \leq k\} \cup \\ & \{\Leftarrow^h\Downarrow^h : 0 \leq h \leq k\} \cup \{\Rightarrow^h\Downarrow^h : 0 \leq h \leq k\} \end{aligned}$$

be the set of spatial programs that allow to reach *all and only* those points in an agent's neighborhood of size k. Then, under the previous interpretation of the operator S_i^k, the following formulas become valid for every $\alpha \in \mathcal{P}rg(k)$:

$$\mathsf{h}_i \to ([\alpha]p \leftrightarrow \mathsf{S}_i^k[\alpha]p) \tag{18}$$

$$\mathsf{h}_i \to ([\alpha]\mathsf{h}_j \leftrightarrow \mathsf{S}_i^k[\alpha]\mathsf{h}_j) \tag{19}$$

This means that if an agent i has a range of vision of size k, then she can perceive all facts that are true and all agents that are positioned in her neighborhood of size k. The following formula is an example of instance of the previous validity:

$$\mathsf{h}_i \to ([\Uparrow]\mathsf{h}_j \leftrightarrow \mathsf{S}_i^1[\Uparrow]\mathsf{h}_j) \tag{20}$$

The latter means that if agent i has a range of vision of size 1 then, agent j is above her iff agent i perceives this.

We postpone to future work a study of the complexities of model-checking and of decidability for the extensions by epistemic operators S_i^k of the different logics presented in the paper.

Coalitional capability.

DL-SM provides an interesting basis for the development of a logic of coalitional capabilities in the two-dimensional space. We take the concept of 'coalitional capability' in the sense of Coalition Logic CL [13]. Specifically, we say that coalition $C \subseteq Agt$ has the capability of ensuring φ, denoted by $\langle\!\langle C \rangle\!\rangle\varphi$, if and only if "there exists a joint action δ_C of coalition C such that, by performing it, outcome φ will be ensured, no matter what the agents outside C decide to do". The extension of DL-SM by coalitional capability operators $\langle\!\langle C \rangle\!\rangle$ is rather simple, as the agents' action repertoires only includes the four basic movements in the plane ($\Uparrow_i, \Downarrow_i, \Leftarrow_i$ and $\Rightarrow_i$) and the action of doing nothing (nil_i).

Following Section 7, for every coalition $C \subseteq Agt$ we define its set of joint of actions $\Delta_C = \prod_{i \in C} Act_i$ and denote elements of Δ_C by $\delta_C, \delta'_C, \ldots$ Then, the truth condition of the operator $\langle\!\langle C \rangle\!\rangle$ goes as follows: $M,(x,y) \models \langle\!\langle C \rangle\!\rangle\varphi$ if and only if $\exists\delta_C \in \Delta_C$ such that

$$\forall\delta'_{Agt\setminus C} \in \Delta_{Agt\setminus C} : M,(x,y) \models [\delta_C, \delta'_{Agt\setminus C}]\varphi.$$

Since δ_C and $\delta'_{Agt\setminus C}$ are finite, $\langle\!\langle C \rangle\!\rangle\varphi$ is expressible in DL-SM but at the price of an exponential blowup in the size of the formula φ.

It is easy to check that the operator $\langle\!\langle C \rangle\!\rangle$ satisfies the following basic principles of the coalitional capability operator by [13]:

$$\neg\langle\!\langle C \rangle\!\rangle\bot \tag{21}$$

$$\langle\!\langle C \rangle\!\rangle\top \tag{22}$$

$$\neg\langle\!\langle \emptyset \rangle\!\rangle\neg\varphi \to \langle\!\langle Agt \rangle\!\rangle\varphi \tag{23}$$

$$\langle\!\langle C \rangle\!\rangle(\varphi \wedge \psi) \to \langle\!\langle C \rangle\!\rangle\varphi \tag{24}$$

$$\begin{aligned} (\langle\!\langle C_1 \rangle\!\rangle\varphi \wedge \langle\!\langle C_2 \rangle\!\rangle\psi) \to & \langle\!\langle C_1 \cup C_2 \rangle\!\rangle(\varphi \wedge \psi) \\ & \text{if } C_1 \cap C_2 = \emptyset \end{aligned} \tag{25}$$

$$\frac{\varphi \leftrightarrow \psi}{\langle\!\langle C \rangle\!\rangle\varphi \leftrightarrow \langle\!\langle C \rangle\!\rangle\psi} \tag{26}$$

$\langle\!\langle C \rangle\!\rangle$ satisfies additional principles that are proper to its spatial interpretation. For instance, it is easy to check that, under the previous interpretation, the following two formulas become valid:

$$\neg\langle\!\langle C \rangle\!\rangle\mathsf{h}_i \text{ if } i \notin C \tag{27}$$

$$([\Uparrow]\mathsf{h}_i \vee [\Rightarrow]\mathsf{h}_i \vee [\Downarrow]\mathsf{h}_i \vee [\Leftarrow]\mathsf{h}_i) \to \langle\!\langle \{i\} \rangle\!\rangle\mathsf{h}_i \tag{28}$$

The two validities captures the basic idea that an agent has exclusive control of her position in the sense that: (i) if coalition C does not include agent i then C cannot force i to be "here", and (ii) agent i has the capability to move "here" if she is "around".

We postpone to future work a more systematic analysis of the basic principles of the operator $\langle\!\langle C \rangle\!\rangle$ as well as a study of a strategic capability operator in the sense of ATL [1] based on the semantics of the logic DL-SM.

REFERENCES

[1] R. Alur, T. A. Henzinger, and O. Kupferman. Alternating-time temporal logic. *Journal of the ACM*, 49(5):672–713, 2002.

[2] B. Aminof, A. Murano, S. Rubin, and F. Zuleger. Automatic verification of multi-agent systems in parameterised grid-environments. In *Proceedings of the International Conference on Autonomous Agents & Multiagent Systems (AAMAS 2016)*, pages 1190–1199. ACM, 2016.

[3] P. Balbiani, O. Gasquet, and F. Schwarzentruber. Agents that look at one another. *Logic Journal of the IGPL, :Combined Special Issue: Best papers of FAMAS 2007 and FAMAS 2009*, 21(3):438–467, juin 2013.

[4] V. Belle and H. J. Levesque. A logical theory of localization. *Studia Logica*, 104(4):741–772, 2016.

[5] N. Belnap, M. Perloff, and M. Xu. *Facing the future: agents and choices in our indeterminist world*. Oxford University Press, New York, 2001.

[6] P. Blackburn. Representation, reasoning, and relational structures: a hybrid logic manifesto. *Logic Journal of the IGPL*, 8:339–365, 2000.

[7] E. Clarke, E. Emerson, and A. Sistla. Automatic verification of finite-state concurrent systems using temporal logic specifications. *ACM Transactions on Programming Languages and Systems*, 8:244, 1986.

[8] R. Fagin, J. Y. Halpern, Y. Moses, and M. Vardi. *Reasoning about knowledge*. MIT Press, Cambridge, 1995.

[9] D. Gabbay, A. Kurucz, F. Wolter, and M. Zakharyaschev. *Many-Dimensional Modal Logics: Theory and Applications*. Elsevier, 2003.

[10] O. Gasquet, V. Goranko, and F. Schwarzentruber. Big brother logic: logical modeling and reasoning about agents equipped with surveillance cameras in the plane. In *Proceedings of the International conference on Autonomous Agents and Multi-Agent Systems (AAMAS 2014)*, pages 325–332. ACM Press, 2014.

[11] M. Marx and M. Reynolds. Undecidability of compass logic. *Journal of Logic and Computation*, 9:897–941, 1997.

[12] M. Marx and Y. Venema. *Multi-Dimensional Modal Logic*, volume 4 of *Applied Logic Series*. Springer Netherlands, first edition, 1997.

[13] M. Pauly. A modal logic for coalitional power in games. *Journal of Logic and Computation*, 12(1):149–166, 2002.

[14] Z. M. Reynolds, M. On the products of linear modal logics. *Journal of Logic and Compution*, 11:909–931, 2001.

[15] S. Rubin, F. Zuleger, A. Murano, and B. Aminof. Verification of asynchronous mobile-robots in partially-known environments. In *Proceedings of the 18th International Conference on Principles and Practice of Multi-Agent Systems (PRIMA 2015)*, volume 9387 of *LNCS*, pages 185–200. Springer, 2015.

[16] R. A. Schmidt, D. Tishkovsky, and U. Hustadt. Interactions between knowledge, action and commitment within agent dynamic logic. *Studia Logica*, 78(3):381–415, 2004.

[17] H. van Ditmarsch, W. van der Hoek, and B. Kooi. *Dynamic Epistemic Logic*. Springer, 2008.