A simple logic of the hide and seek game

Dazhu Li[1,2], Sujata Ghosh[3], Fenrong Liu[4], and Yaxin Tu[5]

[1] Institute of Philosophy, Chinese Academy of Sciences, Beijing
[2] Department of Philosophy, University of Chinese Academy of Sciences
[3] Indian Statistical Institute, Chennai
[4] The Tsinghua–UvA JRC for Logic, Department of Philosophy, Tsinghua University, Beijing
[5] Institute for Interdisciplinary Information Sciences, Tsinghua University, Beijing

Abstract. We discuss a simple logic to describe one of our favourite games from childhood, hide and seek, and show how a simple addition of an equality constant to describe the winning condition of the seeker makes our logic undecidable. There are certain decidable fragments of first-order logic which behave in a similar fashion with respect to such a language extension, and we add a new modal variant to that class. We discuss the relative expressive power of the proposed logic in comparison to the standard modal counterparts. We prove that the model checking problem for the resulting logic is P-complete. In addition, by exploring the connection with related product logics, we gain more insight towards having a better understanding of the subtleties of the proposed framework.

Keywords: Hide and seek games · Cops and robber games · Modal logic · Identity constant · Computational complexity · Product logic

1 From games to logic

Everyone remembers the pleasure of playing hide and seek in her or his childhood. After calling out "I am ready, you can come to find me", the fun part is to stay at your secret spot, not making any noise, and to expect that the other player would not discover you. Once you are found, the other wins. Let us consider a setting with two players, *hider* and *seeker*. Following the research program of [10], the game of hide and seek is naturally seen as a graph game, where seeker and hider are located at two different nodes, and are allowed to move around. The goal of seeker is to meet hider, while the goal of hider is to avoid seeker. For the game that many of us played in childhood, the hider basically stays at one place, whereas seeker moves from one node to another. We can describe such graph games using the basic modal logic. However, if we consider a simple modification by allowing moves for both players (akin to the game of cops and robber [34]), the setting becomes quite diverse. On the one hand, these graph games are natural candidates for modelling computational search problems, on the other hand, the nuanced interaction between the players playing hide and seek is a showcase of interactive players having their goals entangled, which is a popular phenomenon in social networks. In other words, the graph game of hide and seek provides us with an ideal arena where we can study reasoning about social interaction and challenges therein arising from such intertwined objectives of players. In the following we will make these games more precise and provide a language to express strategic reasoning and winning conditions of players.

However, before going into the logic details, let us first get a feel about the hide and seek game regarding the information available to the players. That will also lead us to understand the kind of reasoning that we plan to explore for such games. Essentially, it is an imperfect information game where the seeker is not aware of the position of hider, whereas the hider may or may not know the exact position of the seeker. Both players know the game graph where they move about and are aware of their own positions and moves. Now, the modification that we talk about makes the setting even more interesting information-wise, as then we can consider different levels of information available to both the players. However, to keep things simple we start off from a high-level modeller's perspective, that is, we reason *about* such games. Thus, we reason about players' observations and moves with the assumption that the whole graph and the players' positions at each stage of the game are available to us. We leave the players' perspectives for future work.

Coming back to the game proper, we have the two players located at two different nodes. To model their moves we consider a pair of states as an evaluation point rather than a single state in a Kripke model, and consider distinct modalities to express the moves of the players. The evaluation of these two different modalities, one for each player, can then be assessed coordinate-wise with respect to the pair of states.

In addition, a winning condition for the hide and seek game corresponding to the seeker finding the hider can be modelled by considering a pair of states whose first and second elements are the same. This basically gives us the identity relation which can be expressed by introducing a special identity proposition. We first note that one can show the decidability of the satisfiability problem of the two-dimensional modal logic mentioned above, without the special proposition. Interestingly enough, such a simple addition, viz. incorporating the identity proposition, transforms a decidable modal logic into an undecidable one. In fact, there are various elegant examples of logics that suggest that taking this identity relation into account may change previously decidable logics (without equality) into undecidable ones, e.g., the Gödel class of first-order formulas with identity (cf. [24]). A more recent example is the logic of functional dependence with function symbols (see [6, 35]). We add one more logic to this class. This result also refutes a conjecture mentioned in [10] which stated that the extended logic with the identity proposition will remain decidable. The related notion of expressive power of the proposed logic is also discussed here.

We note that this modified version of the hide and seek game played on graphs is a special case of the cops and robber game [34], a classic pursuit-evasion game played on graphs, where several cops attempt to catch a robber. The hide and seek game corresponds to the game having a single cop chasing a robber. Thus, this study opens up the possibility of a logical analysis of these cops and robber games with all their generality (cf. [34]) which have been well-studied from algorithmic and combinatorial perspectives.

Before going any further we would like to mention that this work is an extension of our conference proceedings version [31]. In comparison to the proceedings version, the article has been strengthened in the following ways: with respect to the original results (e.g., relation between modal equivalence and bisimulation, undecidability of the proposed logic), we have added proof details as well as explanations about the ideas involved, so as to provide a better understanding of these technical results that were presented in [31]. Furthermore, we have also provided new results and discussion with respect to our logic in terms of (i) a number of interesting validities, (ii) computational complexity of the model checking problem of the logic, and (iii) several representation results to provide a new research direction that links logics of our style with the well-developed field of product logics. Finally, several open problems are proposed for interested researchers.

In Section 2 we introduce a logic (LHS) to reason about plays and winning conditions in the hide and seek game. Section 3 deals with the relative expressive power of the language and relevant notions of bisimulation are introduced to facilitate the discussion. Section 4 explores the computational behavior of the resulting logic showing that the satisfiability problem for LHS is undecidable, whereas the (finite) model checking problem for LHS is P-complete. Section 5 opens up a new direction by embedding LHS into the framework of product logics. Finally, Section 6 provides a discussion on related work, and Section 7 gives pointers to further research.

2 Logic of hide and seek (LHS)

Let us first introduce a logic to describe the game of hide and seek, LHS, followed by some typical validities and an informal discussion about the expressivity of the proposed logic.

Definition 1 (Language $\mathcal{L}$). *Let $\mathsf{P_H}$ denote a countable set of propositional variables for player Hider, and $\mathsf{P_S}$ for player Seeker. The language $\mathcal{L}$ of LHS is given as follows:*

$$\varphi ::= p_H \mid p_S \mid I \mid \neg\varphi \mid (\varphi \wedge \varphi) \mid \langle \mathsf{H} \rangle \varphi \mid \langle \mathsf{S} \rangle \varphi$$

where $p_H \in \mathsf{P_H}$, $p_S \in \mathsf{P_S}$, and I is a propositional constant. Other Boolean connectives are defined in the usual way, and so are the corresponding box modalities $[\mathsf{H}]$ and $[\mathsf{S}]$.

Without loss of generality, the modal operator representing hider's moves is given by $\langle \mathsf{H} \rangle$ and that representing seeker's moves is given by $\langle \mathsf{S} \rangle$. Formulas are evaluated in *standard relational models* $\mathbf{M} = (W, R, \mathsf{V})$, where W is a non-empty set of vertices, $R \subseteq W \times W$ is a set of edges, and $\mathsf{V} : \mathsf{P_H} \cup \mathsf{P_S} \to 2^W$ is a valuation function. Moreover, for any $s, t \in W$, we call $(\mathbf{M}, s, t)$ a *pointed graph model for two players* (for simplicity, *graph model*): intuitively, s and t represent respectively the positions of players Hider and Seeker. To simplify notations, we also employ $\mathbf{M}, s, t$ for $(\mathbf{M}, s, t)$. As usual, we call (W, R) a *frame*. Semantics for LHS is given by the following:

Definition 2 (Semantics). *Let $\mathbf{M} = (W, R, \mathsf{V})$ be a model and $s, t \in W$. Truth of formulas $\varphi \in \mathcal{L}$ at the graph model $(\mathbf{M}, s, t)$, written as $\mathbf{M}, s, t \vDash \varphi$, is defined recursively as follows:*

$$
\begin{array}{rcl}
\mathbf{M}, s, t \vDash p_H & \Leftrightarrow & s \in \mathsf{V}(p_H) \\
\mathbf{M}, s, t \vDash p_S & \Leftrightarrow & t \in \mathsf{V}(p_S) \\
\mathbf{M}, s, t \vDash I & \Leftrightarrow & s = t \\
\mathbf{M}, s, t \vDash \neg\varphi & \Leftrightarrow & \mathbf{M}, s, t \nvDash \varphi \\
\mathbf{M}, s, t \vDash \varphi \wedge \psi & \Leftrightarrow & \mathbf{M}, s, t \vDash \varphi \textit{ and } \mathbf{M}, s, t \vDash \psi \\
\mathbf{M}, s, t \vDash \langle \mathsf{H} \rangle \varphi & \Leftrightarrow & \exists s' \in W \textit{ s.t. } Rss' \textit{ and } \mathbf{M}, s', t \vDash \varphi \\
\mathbf{M}, s, t \vDash \langle \mathsf{S} \rangle \varphi & \Leftrightarrow & \exists t' \in W \textit{ s.t. } Rtt' \textit{ and } \mathbf{M}, s, t' \vDash \varphi
\end{array}
$$

As mentioned earlier, the above language has two modalities, one for each player. Accordingly, all the formulas are evaluated in a graph model. The constant I denotes the identity relation in a game graph to describe the meeting of two players, signifying the fact that the seeker has found the hider. Let us denote LHS_{-I} to be the fragment of LHS without the constant I.

Given a model $\mathbf{M}$ and a set $U \subseteq W$ of states, define $R(U) := \{t \in W \mid$ *there is* $s \in U$ *with* $Rst\}$, denoting the set of successors of the points in U. For simplicity, we write $R(s)$ for $R(\{s\})$ when U is a singleton $\{s\}$. Also, given a model $\mathbf{M}$, $[\![\varphi]\!]^{\mathbf{M}} := \{(s,t) \mid \mathbf{M}, s, t \vDash \varphi\}$ is *the truth set* of formula φ in $\mathbf{M}$. We can introduce the logical notions such as *satisfiability*, *validity* and *modal equivalence* in the usual way, and we omit the details.

Here are some principles that are useful to see the basic features of logic. In what follows, the notation $\clubsuit$ refers to either H or S. First, we have

$$[\clubsuit](\varphi \to \psi) \to ([\clubsuit]\varphi \to [\clubsuit]\psi) \tag{1}$$

It says that both $[\mathsf{H}]$ and $[\mathsf{S}]$ can be distributed over an implication, which may be expected. What is more interesting is the interaction between different Boolean connectives with respect to the constant I, e.g.,

$$\langle \clubsuit \rangle (I \wedge \varphi) \to [\clubsuit](I \to \varphi) \tag{2}$$

capturing the uniqueness of the state that is identical to a given one. Moreover, involving constant I, we have

$$I \to (\langle \mathsf{H} \rangle \top \leftrightarrow \langle \mathsf{S} \rangle \top) \tag{3}$$

$$I \to ([\mathsf{S}]\langle \mathsf{H} \rangle I \wedge [\mathsf{H}]\langle \mathsf{S} \rangle I) \tag{4}$$

where principle (3) states the 'symmetry' of the structure w.r.t. I-pairs, while principle (4) expresses a kind of 'closure' property. Finally, validities of LHS are *not* closed under substitution, say, both validities

$$p_S \to [\mathsf{H}]p_S \qquad p_H \to [\mathsf{S}]p_H \tag{5}$$

may fail after replacing p_S/p_H with some p_H/p_S.

Going back to the hide and seek game itself, one can consider different variants played on the game graph model, e.g., the players can move simultaneously or sequentially. In a sequential play, one can also consider different orders of play. In this paper, we assume that the players move sequentially, and that Hider starts the game. Local one-step winning positions (pairs of states describing the current positions of the players) for each player can be expressed in our language as follows:

- Hider: $\langle \mathsf{H}\rangle[\mathsf{S}]\neg I$
- Seeker: $[\mathsf{H}]\langle \mathsf{S}\rangle I$

More generally, winning positions for Hider and Seeker can be described as:

- Hider: $\forall n(\langle \mathsf{H}\rangle[\mathsf{S}])^n\neg I$
- Seeker: $\exists n([\mathsf{H}]\langle \mathsf{S}\rangle)^n I$

Note that the above conditions involve countable conjunction/disjunction of finite iterations of interactions between two players. The interactions $\langle \mathsf{H}\rangle[\mathsf{S}]$/ $[\mathsf{H}]\langle \mathsf{S}\rangle$ are expressed with two separate modalities, but they are considered as a single unit. These are not expressible in our language. As mentioned in the introduction, we are currently exploring an extension of this language with modal substitution operators which would also provide a finitary way to express such countable Boolean operations.

Remark 1. There are other ways to give suitable logics capturing the hide and seek game. For instance, one can replace identity constant I with C, denoting '*catching*': $\mathbf{M}, s, t \models C$ iff $R(s) \subseteq R(t)$. From the perspective of the game, constant C describes that all states accessible to the hider are accessible to the seeker as well. In contrast to I which states that the seeker has already won, C indicates that she can win in the next round. They amount to the same condition for games of perfect information: if the seeker has the ability to meet the hider she will actually do that, if she is rational. However, from a logical perspective, their interpretations are entirely different, leading to distinct expressive features. For an illustration, let us note that C can be defined as $[\mathsf{H}]\langle \mathsf{S}\rangle I$ in LHS, but I is not definable in the logic extending LHS_{-I} with C.[6] The constant proposition C with the given interpretation may also useful in describing cop-win graphs in the cops and robber game involving a single cop [34], we leave it for future work.

In the next two sections we will explore some logical properties of LHS regarding its expressiveness, the satisfiability, and model checking problems.

3 Bisimulation and expressive power

The notion of bisimulation is an important tool for studying the expressive power of modal logics. We are now going to explore a suitable notion tailored to our logic. We usually need to be careful when introducing the conditions: on the one hand, the definition should ensure that the logic cannot distinguish bisimilar models (i.e., the desired notion is strong enough), but on the other hand, it should also hold between two models whenever they cannot be distinguished by the logical language (thus, it is weak enough). In what follows, we take the standard bisimulation (see e.g., [14]) as the benchmark and investigate the relations between expressiveness of basic modal logic M, LHS_{-I} and LHS. Let us start by comparing that for LHS_{-I} and M.

[6] To prove this, we need a notion of bisimulation for the language with constant C, which can be defined as usual. Then, with the help of this notion, one can find two models that cannot be distinguished by the language with C but can be distinguished by the language of LHS. We leave the details to the readers.

The standard bisimulation, denoted by $\underline{\leftrightarrow}^s$, provides us a semantic characterization of the expressiveness of the basic modal language $\mathcal{L}_\Box$. And at a first glance, the semantic design of logic LHS_{-I} is similar to that of the basic modal logic, except that we now need to consider two states simultaneously when evaluating formulas. So, is logic LHS_{-I} invariant under the standard notion? First, we provide a positive answer in the following sense:

Proposition 1. *If* $(\mathbf{M}, w) \underline{\leftrightarrow}^s (\mathbf{M}', w')$ *and* $(\mathbf{M}, v) \underline{\leftrightarrow}^s (\mathbf{M}', v')$*, then* $(\mathbf{M}, w, v)$ *and* $(\mathbf{M}', w', v')$ *satisfy the same formulas of* LHS_{-I}*.*

Proof. The proof is straightforward by applying induction on formulas φ of LHS_{-I}. We omit the cases for Boolean connectives $\neg$ and $\wedge$, and only consider p_H and $\langle\mathsf{S}\rangle\psi$, since p_S and $\langle\mathsf{H}\rangle\varphi$ can be proved similarly.

(1). First, formula φ is p_H. The following sequence of equivalences holds:

$$\mathbf{M}, w, v \vDash p_H \text{ iff } w \in \mathsf{V}(p_H) \text{ iff } w' \in \mathsf{V}'(p_H) \text{ iff } \mathbf{M}', w', v' \vDash p_H$$

The first equivalence holds directly by the semantics of LHS_{-I}. Next, since $(\mathbf{M}, w) \underline{\leftrightarrow}^s (\mathbf{M}', w')$ and the fact that truth of basic modal formula is invariant under standard bisimulation [14], we have the second equivalence. Finally, the last one follows again from the semantics of LHS_{-I}.

(2). Formula φ is $\langle\mathsf{S}\rangle\psi$. It suffices to show just one direction from $\mathbf{M}, w, v \vDash \varphi$ to $\mathbf{M}', w', v' \vDash \varphi$. By $\mathbf{M}, w, v \vDash \varphi$, there is some $s \in W$ with Rvs and $\mathbf{M}, w, s \vDash \psi$. From $(\mathbf{M}, v) \underline{\leftrightarrow}^s (\mathbf{M}', v')$, it follows that there exists $s' \in W'$ such that $R'v's'$ and $(\mathbf{M}, s) \underline{\leftrightarrow}^s (\mathbf{M}', s')$. By the inductive hypothesis, it holds that $\mathbf{M}', w', s' \vDash \psi$. Consequently, $\mathbf{M}', w', v' \vDash \langle\mathsf{S}\rangle\psi$. The proof is completed. ◻

Therefore, the standard bisimulation is strong enough to measure the expressive power of LHS_{-I}. But, is it also weak enough? Unfortunately, we have the following negative result:

Proposition 2. *There are* $(\mathbf{M}, w, v)$ *and* $(\mathbf{M}', w', v')$ *s.t. they satisfy the same* LHS_{-I}*-formulas but at least one of* $(\mathbf{M}, w) \underline{\leftrightarrow}^s (\mathbf{M}', w')$*,* $(\mathbf{M}, v) \underline{\leftrightarrow}^s (\mathbf{M}', v')$ *may not hold.*[7]

Proof. We show this by giving a counterexample. Consider the models $\mathbf{M}$ and $\mathbf{M}'$ depicted in Figure 1. It holds that $(\mathbf{M}, w_1, w_2)$ and $(\mathbf{M}', v_1, v_2)$ satisfy the same LHS-formulas, and hence, the same LHS_{-I}-formulas, but we do not have $(\mathbf{M}, w_1) \underline{\leftrightarrow}^s (\mathbf{M}', v_1)$. ◻

Intuitively, the failure originates from the 'evaluation-gap' between the two worlds in our graph models $(\mathbf{M}, s, t)$: when considering atomic properties of s, both LHS_{-I} and LHS can only describe those in $\mathsf{P_H}$, but not the ones in $\mathsf{P_S}$.[8]

[7] Strictly speaking, a negative result holds even for the basic modal logic (see [14]). However, it is still ideal if the notion of bisimulation can behave well in a large class of models (e.g., *image-finite models*). But, as illustrated by the counterexample used to show the result, the standard notion even excludes situations that are very simple but cannot be distinguished by LHS_{-I}.

[8] From the perspective of games, the evaluation-gap suggests a way to handle situations where the two players have different observations even when they are at the same position. For example, the gap might allow us to consider further enrichments so that the states in the playing arena can encode different properties for the players: a crowded street reducing the possible moves of the escaping robber is helpful for a chasing cop, meanwhile, it is definitely a disaster to the robber.

Fig. 1. Two graph models $(\mathbf{M}, w_1, w_2)$ and $(\mathbf{M}', v_1, v_2)$ satisfying same LHS-formulas.

Now, it is time to introduce the notion of bisimulation for LHS, from which we can easily obtain that for LHS_{-I}. Here is the definition:

Definition 3 (Bisimulation for LHS models). *Let* $\mathbf{M} = (W, R, \mathsf{V})$, $\mathbf{M}' = (W', R', \mathsf{V}')$ *be two models and let* $s, t \in W$ *and* $s', t' \in W'$. *We say,* $(\mathbf{M}, s, t)$ *is bisimilar to* $(\mathbf{M}', s', t')$ *(denoted by* $(\mathbf{M}, s, t) \underline{\leftrightarrow} (\mathbf{M}', s', t')$*) if*

Atom: $(\mathbf{M}, s, t)$ *and* $(\mathbf{M}', s', t')$ *satisfy the same propositional letters.*
Meet: $s = t$ *iff* $s' = t'$.
Zig$_\mathsf{H}$: *if there exists* $u \in W$ *such that* Rsu, *then there exists* $u' \in W'$ *such that* $R's'u'$ *and* $(\mathbf{M}, u, t) \underline{\leftrightarrow} (\mathbf{M}', u', t')$.
Zig$_\mathsf{S}$: *if there exists* $v \in W$ *such that* Rtv, *then there exists* $v' \in W'$ *such that* $R't'v'$ *and* $(\mathbf{M}, s, v) \underline{\leftrightarrow} (\mathbf{M}', s', v')$.
Zag$_\mathsf{H}$, Zag$_\mathsf{S}$: *those analogous clauses in the converse direction of* **Zig$_\mathsf{H}$** *and* **Zig$_\mathsf{S}$**, *respectively.*

With this definition, it is now easy to check that $(\mathbf{M}, w_1, w_2)$ and $(\mathbf{M}', v_1, v_2)$ in Figure 1 are bisimilar. Although the clauses above look rather routine, it is instructive to notice some subtle aspects of the definition that are in line with our previous observation: the condition **Atom** in effect just requires that $\mathsf{V}(s) \cap \mathsf{P_H} = \mathsf{V}'(s') \cap \mathsf{P_H}$ and $\mathsf{V}(t) \cap \mathsf{P_S} = \mathsf{V}'(t') \cap \mathsf{P_S}$, but s and s' may satisfy different properties p_S and p'_S, say, from $\mathsf{P_S}$, and t and t' may satisfy different properties p_H and p'_H, say, from $\mathsf{P_H}$. Moreover, the clause **Meet** aims to deal with the constant I, and the others are analogous to the zigzag conditions in standard situations.

By dropping the clause **Meet** above, we get the notion for LHS_{-I}, and by $(\mathbf{M}, s, t) \underline{\leftrightarrow}^- (\mathbf{M}', s', t')$ we denote the case that $(\mathbf{M}, s, t)$ and $(\mathbf{M}', s', t')$ are LHS_{-I}-bisimilar. With Definition 3, it holds that:

Proposition 3. *If* $(\mathbf{M}, s, t) \underline{\leftrightarrow} (\mathbf{M}', s', t')$, *then* $(\mathbf{M}, s, t)$ *and* $(\mathbf{M}', s', t')$ *satisfy the same* LHS*-formulas. Also, if* $(\mathbf{M}, s, t) \underline{\leftrightarrow}^- (\mathbf{M}', s', t')$, *then they satisfy the same* LHS_{-I}*-formulas.*

It can be proved by induction on the structure of LHS-formulas and LHS_{-I}-formulas, respectively. Therefore, the language $\mathcal{L}$ cannot distinguish between bisimilar models. However, our previous discussion indicates that having a very strong notion is never the final goal: it is equally important to ask whether the notion is also weak enough. This time we are going to present a positive result w.r.t. a class of models that are LHS-*saturated*:

Definition 4 (LHS-saturation). *A model* $\mathbf{M} = (W, R, V)$ *is said to be* LHS*-saturated, if for any set* Φ *of formulas and states* $w, v \in W$, *it holds that:*

- *If* Φ *is finitely satisfiable in* $R(w) \times \{v\}$, *then the whole set* Φ *is satisfiable in* $R(w) \times \{v\}$, *and*

- *If Φ is finitely satisfiable in $\{w\} \times R(v)$, then the whole set Φ is satisfiable in $\{w\} \times R(v)$.*

The notion is essentially obtained by adapting the so-called *m-saturation* [14] to fit into our logics. As usual, any finite model is LHS-saturated. Furthermore, in terms of infinite $\mathbf{M}$, it intuitively requires that $\mathbf{M}$ contains 'enough' states: for instance, if every finite subset of Φ can be satisfied by some pairs in $R(w)\times\{v\}$, then there must also be a pair satisfying Φ itself. By restricting Φ to the fragment without I, we have a notion for LHS_{-I}, called LHS_{-I}*-saturation*. Now we have enough background to show that:

Proposition 4. *For all $\mathbf{M}$ and $\mathbf{M}'$ that are LHS-saturated, if $(\mathbf{M}, s, t)$ and $(\mathbf{M}', s', t')$ satisfy the same formulas of LHS, then $(\mathbf{M}, s, t) \underline{\leftrightarrow} (\mathbf{M}', s', t')$. Moreover, when $\mathbf{M}$ and $\mathbf{M}'$ are LHS_{-I}-saturated, if $(\mathbf{M}, s, t)$ and $(\mathbf{M}', s', t')$ satisfy the same formulas of LHS_{-I}, then $(\mathbf{M}, s, t) \underline{\leftrightarrow}^- (\mathbf{M}', s', t')$.*

Proof. Assume that $\mathbf{M}$ and $\mathbf{M}'$ are LHS-saturated, and $(\mathbf{M}, s, t)$ and $(\mathbf{M}', s', t')$ satisfy the same formulas of LHS. We show that the modal equivalence relation itself is a bisimulation relation for LHS. Clause **Atom** holds immediately by the fact that $(\mathbf{M}, s, t)$ and $(\mathbf{M}', s', t')$ are modally equivalent w.r.t. LHS. In what follows, we just show **Meet** and $\mathbf{Zig}_{\mathsf{H}}$ are satisfied.

(1). **Meet.** We have the following equivalences: $s = t$ iff $\mathbf{M}, s, t \vDash I$ iff $\mathbf{M}', s', t' \vDash I$ iff $s' = t'$. So, the condition is satisfied.

(2). $\mathbf{Zig}_{\mathsf{H}}$. Assume that $u \in W$ with Rsu. Let Φ denote the set of LHS-formulas that are true at $\mathbf{M}, u, t$. For each finite $\Gamma \subseteq \Phi$, $\mathbf{M}, s, t \vDash \langle\mathsf{H}\rangle \bigwedge \Gamma$. Thus, $\mathbf{M}, s', t' \vDash \langle\mathsf{H}\rangle \bigwedge \Gamma$. So, Φ is finitely satisfiable w.r.t. $R'(s') \times \{t'\}$. Since $\mathbf{M}'$ is LHS-saturated, there exists $u' \in W'$ such that $R's'u'$ and each formula of Φ is true at $\mathbf{M}', u', t'$. Thus, $(\mathbf{M}, u, t)$ and $(\mathbf{M}', u', t')$ satisfy the same LHS-formulas.

So, it concludes that $(\mathbf{M}, s, t) \underline{\leftrightarrow} (\mathbf{M}', s', t')$. Moreover, by dropping the consideration on I, we can prove the case involving logic LHS_{-I}. □

Therefore, w.r.t. $\mathsf{LHS}/\mathsf{LHS}_{-I}$-saturated models, our notion of bisimulation coincides with the corresponding notion of modal equivalence.

Having shown that our novel notions behave well, we end this section with the following result concerning the relations among aforementioned varieties of bisimulations:

Proposition 5. *With respect to the three varieties of bisimulations $\underline{\leftrightarrow}^s$, $\underline{\leftrightarrow}$ and $\underline{\leftrightarrow}^-$, we have the following:*

(1). Both $\underline{\leftrightarrow}^s$ and $\underline{\leftrightarrow}$ are strictly stronger than $\underline{\leftrightarrow}^-$:
- *(1.1). If $(\mathbf{M}, w) \underline{\leftrightarrow}^s (\mathbf{M}', w')$ and $(\mathbf{M}, v) \underline{\leftrightarrow}^s (\mathbf{M}', v')$, then it holds that $(\mathbf{M}, w, v) \underline{\leftrightarrow}^- (\mathbf{M}', w', v')$. But the converse does not hold.*
- *(1.2). If $(\mathbf{M}, w, v) \underline{\leftrightarrow} (\mathbf{M}', w', v')$, then it holds that $(\mathbf{M}, w, v) \underline{\leftrightarrow}^- (\mathbf{M}', w', v')$. But the converse direction does not hold.*

(2). $\underline{\leftrightarrow}^s$ and $\underline{\leftrightarrow}$ are incomparable:
- *(2.1). When $(\mathbf{M}, w) \underline{\leftrightarrow}^s (\mathbf{M}', w')$ and $(\mathbf{M}, v) \underline{\leftrightarrow}^s (\mathbf{M}', v')$, it does not necessarily hold that $(\mathbf{M}, w, v) \underline{\leftrightarrow} (\mathbf{M}', w', v')$.*
- *(2.2). When it holds that $(\mathbf{M}, w, v) \underline{\leftrightarrow} (\mathbf{M}', w', v')$, each of $(\mathbf{M}, w) \underline{\leftrightarrow}^s (\mathbf{M}', w')$ and $(\mathbf{M}, v) \underline{\leftrightarrow}^s (\mathbf{M}', v')$ can fail.*

Proof. We show the two claims one by one.

(1). The relation between $\underline{\leftrightarrow}^s$ and $\underline{\leftrightarrow}^-$ follows from Propositions 1, 2 and 4. Also, it is obvious that $\underline{\leftrightarrow}$ is stronger than $\underline{\leftrightarrow}^-$. For an example, consider the two models given in Figure 2: it holds $(\mathbf{M}, w_1, w_1) \,\underline{\leftrightarrow}^-\, (\mathbf{M}', v_1, v_1)$, but $\mathbf{M}, w_1, w_1 \models \langle \mathsf{H} \rangle \langle \mathsf{S} \rangle \neg I$ and $\mathbf{M}', v_1, v_1 \not\models \langle \mathsf{H} \rangle \langle \mathsf{S} \rangle \neg I$. Now, by Proposition 3, we do not have $(\mathbf{M}, w_1, w_1) \,\underline{\leftrightarrow}\, (\mathbf{M}', v_1, v_1)$.

(2). Consider the models in Figure 2. It is not hard to see that the states w_1 and v_1 cannot be distinguished by the basic modal language, but this would not be the case when we consider LHS. Thus, standard bisimulations need not be bisimulations of LHS. On the other hand, using the models in Figure 1, it is not hard to see that bisimulations of LHS may also be excluded by the notion of standard bisimulation. This completes the proof. ◻

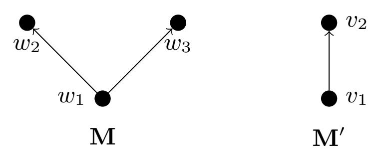

Fig. 2. $(\mathbf{M}, w_1, w_1) \,\underline{\leftrightarrow}^-\, (\mathbf{M}', v_1, v_1)$, but not $(\mathbf{M}, w_1, w_1) \,\underline{\leftrightarrow}\, (\mathbf{M}', v_1, v_1)$.

Properties of LHS- and LHS_{-I}- bisimulation explored here are very basic, and several further questions are worth studying. For instance,

Open problem. What is the computational complexity of checking for bisimulation of LHS or LHS_{-I}? Are they as complex as each other?

4 Computational behavior of LHS

Essentially, LHS introduces a propositional constant to deal with equality in a modal logic framework. This universally accepted relation of indiscernibility is simple in nature. However, as we mentioned in Section 1, there are various elegant examples of logics that suggest that taking this relation into account may change previously decidable logics (without equality) into undecidable ones. In this section, we are going to contribute one more instance to this class: in what follows, we first show that LHS does not have the tree model property or the finite model property, and then prove that the satisfiability problem for LHS is undecidable. Then, by considering the relations between LHS and other relevant logics, we identify the complexity of the model checking problem for our logic, and it turns out that there is a huge gap between the complexity of these two problems: as we shall see, the model checking problem for LHS is P-complete, which would also give us an upper bound for determining the winner in a given game of hide and seek (with a finite graph).[9]

[9] Given a hide and seek game over a finite graph (W, R), where hider and seeker are at s, t, respectively, and the cardinality of W is denoted by $|W|$, one can check that seeker can win iff she can win in $|W|^2$ rounds. So, determining the winner in the game equates to checking whether the formula $\bigvee_{n \leq |W|^2} ([\mathsf{H}]\langle \mathsf{S} \rangle)^n I$ is true at (s, t) in the model (W, R, V), where V is an arbitrary valuation function.

Usually, the tree model property and the finite model property are positive signals for the computational behavior of a logic (cf. e.g., [14]). However, in what follows, we will show that LHS lacks both the properties. Let us begin with a simple result concerning the tree model property:

Proposition 6. LHS *does not have the tree model property.*

Proof. Consider the following formula:

$$\varphi_r := I \wedge \langle \mathsf{H} \rangle \top \wedge [\mathsf{H}] I$$

It is easy to see that it is satisfiable. Also, let $\mathbf{M} = (W, R, \mathsf{V})$ and $u, v \in W$ such that $\mathbf{M}, u, v \vDash \varphi_r$. From I it follows that $u = v$. Also, the conjunct $\langle \mathsf{H} \rangle \top$ indicates that the state u has successors, i.e., $R(u) \neq \emptyset$. Moreover, for all $s \in R(u)$, we have $s = v$. Therefore, $R(u) = \{u\}$. Consequently, the model $\mathbf{M}$ cannot be a tree. The proof is completed. □

Moreover, by constructing a '*spy-point*' [15], i.e., all states that are reachable from u in n-steps can also be reached in one step, we can also prove:

Theorem 1. LHS *lacks the finite model property.*

Proof. Let φ_∞ be the conjunction of the following formulas:

$$\begin{array}{ll} (F1) & I \wedge [\mathsf{H}] \neg I \\ (F2) & \langle \mathsf{H} \rangle [\mathsf{H}] \bot \\ (F3) & [\mathsf{H}] \langle \mathsf{S} \rangle (\neg I \wedge \langle \mathsf{S} \rangle \top \wedge [\mathsf{S}] I) \end{array}$$

Let us briefly comment on the intuition with these formulas. First, $(F1)$ shows that the two states in the current graph model are the same and the point is irreflexive. Also, formula $(F2)$ states that the point can reach a state that is a dead end having no successors. Additionally, $(F3)$, motivated by [28], indicates that the point has more than one successor and for all its successors i, there is also another successor j such that j has i as its only successor.

After presenting the basic ideas of those formulas, we show that the formula φ_∞ is satisfiable. Consider the model $\mathbf{M}_\infty = (W, R, \mathsf{V})$ that is defined as follows:

- $W := \{s\} \cup \mathbb{N}$
- $R := \{\langle s, i \rangle \mid i \in \mathbb{N}\} \cup \{\langle i+1, i \rangle \mid i \in \mathbb{N}\}$
- For all $p \in \mathsf{P_S} \cup \mathsf{P_H}$, $\mathsf{V}(p) := \emptyset$.

See Figure 3 for an illustration. By construction, it can be easily checked that the formula holds at (s, s), i.e., $\mathbf{M}_\infty, s, s \vDash \varphi_\infty$.

Next, let $\mathbf{M} = (W, R, \mathsf{V})$ be an arbitrary model such that $u \in W$ and $\mathbf{M}, u, u \vDash \varphi_\infty$. We are going to show that W is infinite. To do so, we claim that the model contains the following sequence of states of $\mathbf{M}$:

$$w_0, w_1, w_2, w_3, w_4, \ldots$$

such that for all $i \in \mathbb{N}$, the following conditions hold:

P1. $\mathbf{M}, w_i, w_{i+1} \vDash \neg I \wedge \langle \mathsf{S} \rangle \top \wedge [\mathsf{S}] I$

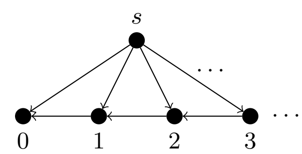

Fig. 3. The model $\mathbf{M}_\infty$.

P2. $\langle u, w_i\rangle \in R$

P3. $R(w_0) = \emptyset$, and for $1 \leq i$, $R(w_i) = \{w_{i-1}\}$

By making an induction on $i \in \mathbb{N}$, we show that there is always such a sequence of those w_i's.

First, let us consider the basic case that $i = 0$. As $\mathbf{M}, u, u \vDash (F2)$, we know that there is $w_0 \in W$ such that Ruw_0 and $\mathbf{M}, w_0, u \vDash [\mathsf{H}]\bot$. Therefore, $R(w_0) = \emptyset$, i.e., we have already obtained the dead end. Moreover, by formula $(F3)$, it holds $\mathbf{M}, w_0, u \vDash \langle\mathsf{S}\rangle(\neg I \wedge \langle\mathsf{S}\rangle\top \wedge [\mathsf{S}]I)$. Therefore, there exists $w_1 \in W$ such that Ruw_1, $w_0 \neq w_1$ and $R(w_1) = \{w_0\}$. Now, it is not hard to see that the clauses P1-P3 hold for both w_0 and w_1.

Now, suppose that we have already had all those states $w_{i\leq n}$, and we proceed to show that there exists w_{n+1} satisfying the conditions P1-P3. By the induction hypothesis, we have Ruw_n. Now, from the formula $(F3)$, it follows that $\mathbf{M}, w_n, u \vDash \langle\mathsf{S}\rangle(\neg I \wedge \langle\mathsf{S}\rangle\top \wedge [\mathsf{S}]I)$. So, there is a state $w_{n+1} \in W$ such that Ruw_{n+1} and $\mathbf{M}, w_n, w_{n+1} \vDash \neg I \wedge \langle\mathsf{S}\rangle\top \wedge [\mathsf{S}]I$. This indicates that w_{n+1} satisfies the requirements P1 and P2. Also, as $\neg I$, $w_n \neq w_{n+1}$. Furthermore, from $\mathbf{M}, w_n, w_{n+1} \vDash \langle\mathsf{S}\rangle\top \wedge [\mathsf{S}]I$, we know that $R(w_{n+1}) = \{w_n\}$, which indicates that the node satisfies P3 as well.

Moreover, from properties P1 and P3, we can infer that whenever $i \neq j$, $w_i \neq w_j$. To be more specific, we have $\mathbf{M}, w_i, w_i \vDash \langle\mathsf{H}\rangle^i\top \wedge [\mathsf{H}]^{i+1}\bot$ for each i. Thus, we have infinitely many states w_i. So, the model $\mathbf{M}$ is infinite. $\square$

4.1 Undecidability of the satisfiability problem for LHS

Now, by encoding the $\mathbb{N} \times \mathbb{N}$ *tiling problem* with the satisfiability problem for our logic, we show that LHS is undecidable. A *tile* t is a 1×1 square, of fixed orientation, with colored edges *right(t), left(t), up(t)* and *down(t)*. The $\mathbb{N} \times \mathbb{N}$ tiling problem is: given a finite set $T = \{t^1, \ldots, t^n\}$ of tile types, is there a function $f : \mathbb{N} \times \mathbb{N} \to T$ such that *right(f(n,m)) = left(f(n+1,m))* and *up(f(n,m)) = down(f(n,m+1))*? The problem is known to be undecidable [13].

Theorem 2. *The satisfiability problem of* LHS *is undecidable.*

Proof. Let $T = \{T^1, \ldots, T^n\}$ be a finite set of tile types. For each $T^i \in T$ we use $u(T^i), d(T^i), l(T^i), r(T^i)$ to represent the colors of its up, down, left and right edges, respectively. We are going to define a formula φ_{tile} such that:

$$\varphi_{\mathsf{tile}} \text{ is satisfiable iff } T \text{ tiles } \mathbb{N} \times \mathbb{N}.$$

To do so, we will use three relations in models (W, R^s, R^r, R^u) in the proof to follow. In line with this, syntactically we have six operators $[\mathsf{H}]^\star$ and $[\mathsf{S}]^\star$

for $\star \in \{s, r, u\}$. Intuitively, all the relations describe the transitions of the left evaluation point and the right evaluation point of a graph model: in what follows, we are going to construct a spy point over relation R^s, and the relations R^u and R^r representing moving up and to the right, respectively, from the corresponding tile to the other.

These three relations are useful to present the underlying intuitions of the formulas that will be constructed. Also, they are helpful in making these formulas short and readable, facilitating a better understanding of the frame. Crucially, this does not change the computational behavior of the original LHS: the three relations can be reduced to one relation as that of our standard models.[10] However, due to page-limit constraints, we forego those details here. Now, we proceed to present the details of φ_{tile}, whose components will be divided into four groups. Let us begin with the first one.

Group 1: Infinite many states induced by R^s and their 'scope'

$$
\begin{array}{ll}
(U1) & I \wedge [\mathsf{H}]^s \neg I \\
(U2) & \langle \mathsf{H} \rangle^s [\mathsf{H}]^s \bot \\
(U3) & [\mathsf{H}]^s \langle \mathsf{S} \rangle^s (\neg I \wedge \langle \mathsf{S} \rangle^s \top \wedge [\mathsf{S}]^s I) \\
(U4) & [\mathsf{H}]^s [\mathsf{S}]^s ([\mathsf{H}]^s \bot \wedge [\mathsf{S}]^s \bot \to I) \\
(U5) & [\mathsf{H}]^s [\mathsf{S}]^s (\langle \mathsf{H} \rangle^s \langle \mathsf{S} \rangle^s I \to I)
\end{array}
$$

Notice that formulas $(U1)$-$(U3)$ are just the R^s-version of the formulas used in the proof for Theorem 1, which were proposed to create infinite models. Immediately, we know that there exists an infinite sequence of states, say,

$$w_0, w_1, w_2, \ldots$$

such that $R^s(w_{i+1}) = \{w_i\}$ and $R^s(w_0) = \emptyset$. Also, for the current evaluation pair (e.g., (s, s)), we have $\{w_i \mid i \in \mathbb{N}\} \subseteq R^s(s)$.

Now let us spell out what formulas $(U4)$ and $(U5)$ express. Essentially, both the formulas establish a 'border' for the scope of nodes that are (directly or indirectly) reachable from s via the relation R^s. Specifically, the formula $(U4)$ shows that $R^s(s)$ contains only a dead end that is exactly the state w_0 listed above, and moreover, the formula $(U5)$ indicates that for any $w_i, w_j \in R^s(s)$, if they can reach the same state in one R^s-step, then they must be the same point, i.e., $w_i = w_j$. See Figure 4 for two counterexamples without the properties of $(U4)$ or $(U5)$. From the two formulas, we know that $R^s(s) = \{w_i \mid i \in \mathbb{N}\}$, i.e., the R^s-successors of s are exactly those w_i.

Intuitively, we will use these w_i to represent tiles. To make this precise, beyond the simple linear order of R^s among those states, we still need to structure them with the two relations R^r and R^u in a subtler way. Our next group of formulas concerns some basic features of the two relations:

Group 2: Basic features of R^u and R^r

$$
\begin{array}{lll}
(U6) & [\mathsf{H}]^s [\mathsf{H}]^\dagger \langle \mathsf{S} \rangle^s I & \dagger \in \{u, r\} \\
(U7) & [\mathsf{H}]^s (\langle \mathsf{H} \rangle^u \top \wedge \langle \mathsf{H} \rangle^r \top) &
\end{array}
$$

[10] For examples of encoding the $\mathbb{N} \times \mathbb{N}$ tiling problem with a single relation, we refer to, e.g., [3].

Fig. 4. Two impossible cases of the R^s-structure of $R^s(s)$: the case on the left cannot satisfy $(U4)$, while the other case cannot satisfy $(U5)$.

$$(U8) \qquad [\mathsf{H}]^s[\mathsf{S}]^s(I \to [\mathsf{H}]^u\neg I \wedge [\mathsf{H}]^r\neg I)$$

$$(U9) \qquad [\mathsf{H}]^s[\mathsf{S}]^s(I \to [\mathsf{H}]^\dagger\neg\langle\mathsf{H}\rangle^\dagger I) \qquad \dagger \in \{u, r\}$$

Before listing more formulas, let us briefly comment on these properties.

For all $i \in \mathbb{N}$, the formulas of $(U6)$ essentially give $R^r(w_i)$ and $R^u(w_i)$ a 'scope'. Specifically, they guarantee that $R^r(w_i), R^u(w_i) \subseteq \{w_0, w_1, \ldots\}$. Therefore, when considering the two relations, we only need to consider those w_i, and there do not exist other states that are involved.

The formula $(U7)$ states that every w_i has successors via R^u and R^r, i.e., $R^u(w_i) \neq \emptyset$ and $R^r(w_i) \neq \emptyset$. Intuitively, this expresses that every tile has at least one tile above it and at least one tile to its right.

Also, the formula $(U8)$ indicates that for all $i \in \mathbb{N}$, we do not have $R^r w_i w_i$ or $R^u w_i w_i$. Thus, with respect to tiles $\{w_0, w_1, \ldots\}$, both the relations R^u and R^r are irreflexive. That is, a tile cannot be above or to the right of itself. Moreover, formulas in $(U9)$ show that both the relations R^r and R^u are asymmetric. So, for instance, in terms of the relation R^r, a tile cannot be to the right as well as to the left of another tile.

Except those basic features captured by formulas of Group 2, what might be more important is our next group of formulas, which structure the states in a *grid* with R^r and R^u:

Group 3: Grid formed by R^u and R^r

$$(U10) \qquad [\mathsf{H}]^s[\mathsf{S}]^s(\langle\mathsf{H}\rangle^\dagger I \to [\mathsf{H}]^\dagger I) \qquad \dagger \in \{u, r\}$$

$$(U11) \qquad [\mathsf{H}]^s[\mathsf{S}]^s(I \to [\mathsf{H}]^u[\mathsf{S}]^r\neg I)$$

$$(U12) \qquad [\mathsf{H}]^s[\mathsf{S}]^s(I \to [\mathsf{H}]^u[\mathsf{H}]^r\neg I \wedge [\mathsf{H}]^r[\mathsf{H}]^u\neg I)$$

$$(U13) \qquad [\mathsf{H}]^s[\mathsf{S}]^s(I \to [\mathsf{H}]^u[\mathsf{H}]^r[\mathsf{H}]^u\neg I)$$

$$(U14) \qquad [\mathsf{H}]^s[\mathsf{S}]^s(I \to [\mathsf{H}]^u[\mathsf{S}]^r\langle\mathsf{H}\rangle^r\langle\mathsf{S}\rangle^u I)$$

Whereas $(U7)$ tells us that all those w_i have R^r- and R^u-successors, formulas in $(U10)$ state that every w_i has at most one R^r-successor and at most one R^u-successor. Thus, both $(U7)$ and $(U10)$ ensure that the transitions between those w_i via R^u and R^r are essentially *functions*: precisely, for all $i \in \mathbb{N}$ and $\dagger \in \{u, r\}$, $R^\dagger(w_i)$ is a singleton. Therefore, every tile has exactly one tile above, and has exactly one tile to its right.

Moreover, formula $(U11)$ suggests that given a tile, its R^r-successor and R^u-successor are always different: for all $i \in \mathbb{N}$, $R^r(w_i) \cap R^u(w_i) = \emptyset$. That is, a tile cannot be above as well as to the right of another tile.

Additionally, $(U12)$ shows that no tile can be both above/below and to the right/left of another tile, and $(U13)$ disallows cycles following successive steps of the R^u, R^r and R^u relations, in this order. Formula ($U14$) states the property of '*confluence*': for all tiles w_i, w_j, w_k, if $R^u w_i w_j$ and $R^r w_i w_k$, then there exists another tile w_n with $R^r w_j w_n$ and $R^u w_k w_n$. Now, tiles are arranged in a grid.

Now, it remains to set a genuine tiling, which can be achieved by our fourth group of formulas. In usual cases this work is often routine when we have an infinite grid-like model (cf. e.g., [14]), but in our case we still should be careful.

Group 4: Tiling the model

$$(U15) \qquad [\mathsf{H}]^s(\bigvee_{1\leq i\leq n} t^i_H \wedge \bigwedge_{1\leq i<j\leq n} \neg(t^i_H \wedge t^j_H))$$

$$(U16) \qquad [\mathsf{S}]^s(\bigvee_{1\leq i\leq n} t^i_S \wedge \bigwedge_{1\leq i<j\leq n} \neg(t^i_S \wedge t^j_S))$$

$$(U17) \qquad [\mathsf{H}]^s[\mathsf{S}]^s(I \to \bigvee_{1\leq i\leq n} (t^i_H \wedge t^i_S))$$

$$(U18) \qquad [\mathsf{H}]^s(\bigwedge_{1\leq i\leq n} (t^i_H \to \langle \mathsf{H}\rangle^u \bigvee_{1\leq j\leq n, u(T_i)=d(T_j)} t^j_H))$$

$$(U19) \qquad [\mathsf{H}]^s(\bigwedge_{1\leq i\leq n} (t^i_H \to \langle \mathsf{H}\rangle^r \bigvee_{1\leq j\leq n, r(T_i)=l(T_j)} t^j_H))$$

Formulas ($U15$)-($U16$) indicate that a node can be occupied by 'two' tiles t^i_H and t^j_S. As one node can only be occupied by exactly one tile, the statement here may look a bit strange. However, we would like to argue that essentially there exists no problem, see our discussion on formula ($U17$) below.

By formula ($U17$), for every fixed $i \in \mathbb{N}$, when both t^i_H and t^j_S hold at a node, then we have $i = j$, i.e., they are of the same tile type T^i. In this sense, we can say that the subscripts 'H' and 'S' are just 'position-labels' to refer to the evaluation nodes in the current graph model, and a node in the model is essentially occupied by exactly one tile. Moreover, for the same reason, although for each T^i, we have different propositional atoms t^i_S and t^i_H, all types of tiles we use are exactly those given by the original T, but not any extra ones.

Based on the analyze above, when tiling the model, taking one of the parts for Hider and Seeker into account is enough, which explains why only those t^i_H are involved in formulas ($U18$) and ($U19$): the former one states that colors of tiles match going up, while the latter expresses that they match going right.

Now, let φ_{tile} be the conjunctions of all formulas listed in the four groups. Based on our analyses above, any model satisfying φ_{tile} is a tiling of $\mathbb{N} \times \mathbb{N}$.

On the other hand, it remains to show the other direction. Now suppose that a function $f : \mathbb{N} \times \mathbb{N} \to T$ is a tiling of $\mathbb{N} \times \mathbb{N}$. Then, we can define a model $\mathbf{M}_t = (W, R^s, R^u, R^r, \mathsf{V})$ in the following:

- $W := \{s\} \cup (\mathbb{N} \times \mathbb{N})$
- R^s consists of the following:
  - For all $x \in \mathbb{N} \times \mathbb{N}$, $\langle s, x\rangle \in R^s$
  - For all $\langle n, 0\rangle \in \mathbb{N} \times \mathbb{N}$ with $1 \leq n$, $\langle\langle n + 1, 0\rangle, \langle 0, n\rangle\rangle \in R^s$
  - For all other $\langle n, m + 1\rangle \in \mathbb{N} \times \mathbb{N}$, $\langle\langle n, m + 1\rangle, \langle n + 1, m\rangle\rangle \in R^s$
- $R^u := \{\langle\langle n, m\rangle, \langle n, m + 1\rangle\rangle \mid n, m \in \mathbb{N}\}$

- $R^r := \{\langle\langle n, m\rangle, \langle n+1, m\rangle\rangle \mid n, m \in \mathbb{N}\}$
- $\mathsf{V}(t^i_H) = \mathsf{V}(t^i_S) = \{\langle n, m\rangle \in \mathbb{N} \times \mathbb{N} \mid f(\langle n, m\rangle) = T^i\}$, for all $i \in \{1, \ldots, n\}$
- $\mathsf{V}(p_H) = \mathsf{V}(q_S) = \emptyset$, for all other $p_H, q_S \in \mathsf{P_H} \cup \mathsf{P_S}$.

Figure 5 presents a crucial fragment of the structure. By construction, one can check that $\mathbf{M}_t, s, s \vDash \varphi_{\mathsf{tile}}$. This completes the proof. □

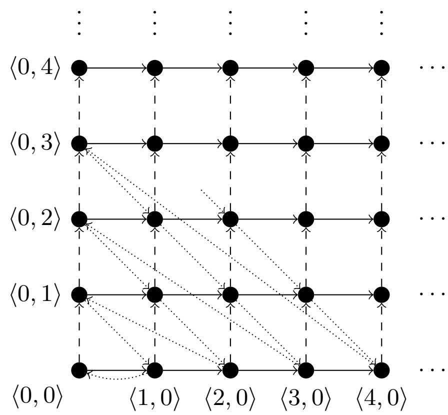

Fig. 5. The restriction of the structure of $\mathbf{M}_t$ to $\mathbb{N} \times \mathbb{N}$, where the resulting R^s, R^u, R^r are represented by dotted-, dashed- and solid-arrows respectively. To obtain the whole structure, we just need to add a new state s and draw a dotted-arrow from s to every member of $\mathbb{N} \times \mathbb{N}$.

Remark 2. It is worth noting that the proof for Theorem 2 essentially indicates that in the presence of the constant I, the logical device for the hide and seek games with *two* players is already undecidable. Based on the undecidability proof, one can infer the undecidability of the class of logics generalizing our framework to capture the games with $3 \leq n \in \mathbb{N}$ players.

4.2 Complexity of the model checking problem for LHS

To identify the complexity of the model checking problem for LHS, in this section we are going to establish the bounds for the same. We focus on the lower bound result for now. In Section 3, we have already seen that the notions of bisimulation for logics M, LHS_{-I} and even LHS are closely related. Now, we continue to introduce a translation $\mathsf{t} : \mathcal{L}_\Box \to \mathcal{L}$ from the basic modal language into the language of LHS. Here are the details:

- $\mathsf{t}(p) := p_H$, for each propositional letter p.
- The function preserves Boolean connectives $\neg$ and $\wedge$.
- $\mathsf{t}(\Diamond\varphi) := \langle\mathsf{H}\rangle\mathsf{t}(\varphi)$

In line with the translation, a model $\mathbf{M} = (W, R, \mathsf{V})$ for the basic modal logic gives rise to a model $\mathbf{M}^+ = (W, R, \mathsf{V}^+)$ for LHS, where

- $\mathsf{V}^+(p_H) = \mathsf{V}(p)$ for each propositional letter of the basic modal language $\mathsf{V}^+(p_S) = \emptyset$.

Now we claim the following:

Proposition 7. *For any basic modal formula* φ, $\mathbf{M} = (W, R, \mathsf{V})$ *and* $s, t \in W$, $\mathbf{M}, s \vDash \varphi$ *iff* $\mathbf{M}^+, s, t \vDash \mathsf{t}(\varphi)$.

Proof. By induction on the formulas φ of the basic modal language. □

The translation t in effect turns M into the part for hider in LHS, i.e., a fragment of LHS without $p_S \in \mathsf{P_S}$, $\langle \mathsf{S} \rangle$ or I, but definitely, one can also treat M as the fragment for seeker containing no $p_H \in \mathsf{P_H}$, $\langle \mathsf{H} \rangle$ or I. Having established a translation from M to LHS, our next step is to embed LHS into FOL to come up with an upper bound result. Unlike the basic modal logic, our logic is not contained in the two-variable fragment of FOL, as illustrated by the undecidability result. In what follows, we proceed to show that LHS can be reduced into the fragment of FOL with 3 variables. Let $\mathcal{L}_1$ be the first-order language with countable many unary predicate symbols P_H^i, P_S^i, a binary relation R and equality $\equiv$.

Definition 5 (First-order translation with 3 variables). *Let* x_1, x_2, x_3 *be three variables, and* $i, j \in \{1, 2, 3\}$ *be distinct from each other. The translation* $\mathcal{T}_{i,j}$, *carried out with respect to variables* x_i, x_j, *is recursively defined as follows:*

$$\begin{aligned}
\mathcal{T}_{i,j}(p_H) &:= P_H x_i \\
\mathcal{T}_{i,j}(p_S) &:= P_S x_j \\
\mathcal{T}_{i,j}(I) &:= x_i \equiv x_j \\
\mathcal{T}_{i,j}(\neg\varphi) &:= \neg\mathcal{T}_{i,j}(\varphi) \\
\mathcal{T}_{i,j}(\varphi_1 \land \varphi_2) &:= \mathcal{T}_{i,j}(\varphi_1) \land \mathcal{T}_{i,j}(\varphi_2) \\
\mathcal{T}_{i,j}(\langle \mathsf{H} \rangle\varphi) &:= \exists x_{\{1,2,3\}\setminus\{i,j\}}(Rx_i x_{\{1,2,3\}\setminus\{i,j\}} \land \mathcal{T}_{\{1,2,3\}\setminus\{i,j\},j}(\varphi)) \\
\mathcal{T}_{i,j}(\langle \mathsf{S} \rangle\varphi) &:= \exists x_{\{1,2,3\}\setminus\{i,j\}}(Rx_j x_{\{1,2,3\}\setminus\{i,j\}} \land \mathcal{T}_{i,\{1,2,3\}\setminus\{i,j\}}(\varphi))
\end{aligned}$$

Some comments about these clauses are in order. First, the order of the subscript i, j in $\mathcal{T}_{i,j}$ matters, and in general $\mathcal{T}_{i,j}(\varphi)$ is *not* the same as $\mathcal{T}_{j,i}(\varphi)$. Next, Definition 5 essentially gives us six functions: one for each ordered pair $\langle x_i, x_j \rangle$, with $\{x_i, x_j\} \subseteq \{x_1, x_2, x_3\}$ and $i \neq j$. Finally, except for $\{x_1, x_2, x_3\}$, the formula $\mathcal{T}_{i,j}(\varphi)$ does not contain any other variables, and for the free variables, we have $Free(\mathcal{T}_{i,j}(\varphi)) \subseteq \{x_i, x_j\}$. Here is an example to illustrate how the translation works.

Example 1. Consider the formula $\langle \mathsf{H} \rangle(p_H \land \langle \mathsf{S} \rangle(\neg p_S \land I))$. Its translation w.r.t. x_1 and x_2 is as follows:

$$\begin{aligned}
&\mathcal{T}_{1,2}(\langle \mathsf{H} \rangle(p_H \land \langle \mathsf{S} \rangle(\neg p_S \land I))) \\
&= \exists x_3(Rx_1x_3 \land \mathcal{T}_{3,2}(p_H) \land \mathcal{T}_{3,2}(\langle \mathsf{S} \rangle(\neg p_S \land I))) \\
&= \exists x_3(Rx_1x_3 \land P_H x_3 \land \exists x_1(Rx_2x_1 \land \mathcal{T}_{3,1}(\neg p_S \land I))) \\
&= \exists x_3(Rx_1x_3 \land P_H x_3 \land \exists x_1(Rx_2x_1 \land \neg P_S x_1 \land x_3 \equiv x_1))
\end{aligned}$$

The formula intuitively states that the left state in question has a p_H-successor that is also accessible from the right evaluation state, and the successor is not p_S. Starting from $\mathcal{T}_{1,2}$, the example illustrates how we move to the translation carried out w.r.t. other pairs of variables. Now, let us show the correctness of the translation:

Proposition 8. *Let* $\mathbf{M} = (W, R, \mathsf{V})$ *be a model,* $s, t \in W$*, and* φ *a formula of* LHS*. Then, for all distinct* $i, j \in \{1, 2, 3\}$*,* $\mathbf{M}, s, t \vDash \varphi$ *iff* $\mathbf{M} \vDash \mathcal{T}_{i,j}(\varphi)[s, t]$*.*[11]

Proof. By induction on formulas φ of LHS. □

As a corollary, we have the following:

Corollary 1. LHS *can be reduced into the 3 variable fragment of* FOL*, with a function having a polynomial size increase.*

Now, we proceed to show the complexity of the model checking problem:

Theorem 3. *Model checking for* LHS *is* P*-complete.*

Proof. A lower bound is provided by Proposition 7: the model checking for LHS is P-hard, since model checking for M is P-complete (see, e.g., [8]). On the other hand, as proved by [40], the model checking for every finite variable fragments of FOL, with a fixed number of variables, is in P. So, Corollary 1 establishes an upper bound for us: the model checking for LHS is in P. Thus, it can be concluded that the model checking problem for LHS is P-complete. □

Remark 3. Similar to the case of Theorem 2, the complexity result of the model checking problem applies to the generalizations with more players. Given such a logic L with $n \geq 3$ evaluation states, here are two observations. On the one hand, one can embed the basic modal logic into a fixed part of L. On the other hand, we can also generalize Corollary 1: by adapting Definition 5, one can translate L into a $n + 1$ variable fragment of FOL.

In line with the complexity result of the model checking problem, Algorithm 1 presents a method to obtain the truth set of a formula φ in a given model $\mathbf{M}$ in $O(|\varphi| \times |W|^3)$ time, where $|\varphi|$ is the length of φ defined as follows:

- $|p_H| = |p_S| = |I| = 1$
- $|\neg\varphi| = |\langle \mathsf{H} \rangle \varphi| = |\langle \mathsf{S} \rangle \varphi| = |\varphi| + 1$
- $|\varphi \wedge \psi| = |\varphi| + |\psi| + 1$

It might be useful to point out that $|\varphi|$ is no less than the cardinality of the set of all sub-formulas of φ.

Digression: expressivity once more. So far, we have already established a notion of bisimulation and a first-order translation for our logic. With these results in hand, a natural next step is as follows:

Open problem. Show a van Benthem style characterization theorem for LHS.[12]

5 Zoom out: LHS as a product logic

Those who are familiar with product logics may wonder its precise connection with our work. We will now put LHS in the landscape of product logics (see, e.g., [33, 20–22, 29]). More concretely, we will first identify the counterpart of LHS in product logics based on a sub-class of the so called 'extended product

[11] By $\mathbf{M} \vDash \mathcal{T}_{i,j}(\varphi)[s, t]$, we mean that when values of x_i and x_j in φ are s, t respectively, $\mathcal{T}_{i,j}(\varphi)$ is satisfied by $\mathbf{M}$.

[12] [18] is an ongoing project to solve this problem and several related issues.

Algorithm 1: LHS model checking, where a syntactic-growing sequence is a sequence of formulas $(\varphi_1, \cdots, \varphi_n)$ such that the size of φ_i is no larger than that of φ_j whenever $i < j$.

Input: φ: a formula, $\mathbf{M} = (W, R, \mathsf{V})$: a model
Output: The truth set of φ in $\mathbf{M}$

function $\mathsf{Truth}(\varphi, \mathbf{M})$
  Record all sub-formulas of φ with a syntactic-growing sequence $S(\varphi)$
  Record all truth sets of formulas in $S(\varphi)$, one by one, in a table *truth* as follows:
  forall $\psi \in S(\varphi)$ **do**
    $truth(\psi) \leftarrow \emptyset$
    if $\psi \in \mathsf{P_H}$ **then**
      $truth(\psi) \leftarrow \mathsf{V}(\psi) \times W$
    if $\psi \in \mathsf{P_S}$ **then**
      $truth(\psi) \leftarrow W \times \mathsf{V}(\psi)$
    if $\psi = I$ **then**
      $truth(\psi) \leftarrow \{(s, s) \mid s \in W\}$
    if $\psi = \neg\chi$ **then**
      $truth(\psi) \leftarrow W \times W \setminus truth(\chi)$
    if $\psi = \psi_1 \wedge \psi_2$ **then**
      $truth(\psi) \leftarrow truth(\psi_1) \cap truth(\psi_2)$
    if $\psi = \langle \mathsf{H} \rangle \chi$ **then**
      forall $(s', t) \in truth(\chi)$ *and* $s \in W$ **do**
        if Rss' **then**
          $truth(\psi) \leftarrow truth(\psi) \cup \{(s, t)\}$
    if $\psi = \langle \mathsf{S} \rangle \chi$ **then**
      forall $(s, t') \in truth(\chi)$ *and* $t \in W$ **do**
        if Rtt' **then**
          $truth(\psi) \leftarrow truth(\psi) \cup \{(s, t)\}$
  return $truth(\varphi)$

models'. The formulation is natural, but restrictions are imposed on the subclass that cannot be captured by the logic. Some restrictions are relaxed to build the logic on a larger class of models. We will relate LHS with product logics by showing representation results for our models in terms of certain product models. Although this is just a first step, these results may help to obtain other results for LHS, e.g., a complete Hilbert style proof system.

5.1 General setting: product models

First of all, to interpret the language $\mathcal{L}$ properly, let us build a class of models on product models. Given two frames $\mathcal{F}_1 = (W_1, R_1)$ and $\mathcal{F}_2 = (W_2, R_2)$, their product frame is $(W_1 \times W_2, R^l, R^r)$ with the following:

$$\begin{array}{lll} R^l \langle u, v \rangle \langle s, t \rangle & \text{iff} & R_1 us \text{ and } v = t \\ R^r \langle u, v \rangle \langle s, t \rangle & \text{iff} & R_2 vt \text{ and } u = s \end{array}$$

With this, we define the following enrichment:

Definition 6 (Extended product models). *An extended product model* $\mathbb{M} = (S \times U, R^l, R^r, \mathbb{I}, \mathbb{V})$ *for* LHS *is a tuple such that*

(1). $(S \times U, R^l, R^r)$ *is a product frame,*
(2). $\mathbb{I} = \{\langle s, s\rangle \mid s \in S \cap U\}$*, and*
(3). $\mathbb{V} : \mathsf{P_H} \cup \mathsf{P_S} \to 2^{S\times U}$ *is a valuation function satisfying the following:*

$$\begin{aligned} \langle s,t\rangle \in \mathbb{V}(p_S) &\Leftrightarrow S \times \{t\} \subseteq \mathbb{V}(p_S) \\ \langle s,t\rangle \in \mathbb{V}(p_H) &\Leftrightarrow \{s\} \times U \subseteq \mathbb{V}(p_H) \end{aligned}$$

As observed, the extended product models with a new component $\mathbb{I}$, aiming to tackle the propositional constant, and their valuation functions are restricted with further clauses that are crucial to ensure an appropriate transition between these valuation functions and those of standard relational models.[13] We call tuples $(S \times U, R^l, R^r, \mathbb{I})$ without valuation functions *extended product frames.* Truth conditions w.r.t. extended product models are straightforward:

$$\begin{aligned} \mathbb{M}, \langle s,t\rangle \vDash p &\Leftrightarrow \langle s,t\rangle \in \mathbb{V}(p), \quad \text{for each } p \in \mathsf{P_H} \cup \mathsf{P_S} \\ \mathbb{M}, \langle s,t\rangle \vDash I &\Leftrightarrow \langle s,t\rangle \in \mathbb{I} \\ \mathbb{M}, \langle s,t\rangle \vDash \langle \mathsf{H}\rangle\varphi &\Leftrightarrow \exists \langle s',t'\rangle \in S \times U \text{ s.t. } R^l\langle s,t\rangle\langle s',t'\rangle \text{ and } \mathbb{M}, \langle s',t'\rangle \vDash \varphi \\ \mathbb{M}, \langle s,t\rangle \vDash \langle \mathsf{S}\rangle\varphi &\Leftrightarrow \exists \langle s',t'\rangle \in S \times U \text{ s.t. } R^r\langle s,t\rangle\langle s',t'\rangle \text{ and } \mathbb{M}, \langle s',t'\rangle \vDash \varphi \end{aligned}$$

In the remainder of this section, we are mainly interested in those extended product models with certain properties. Having LHS in mind, a natural and direct class of such models, denoted by $\mathbb{G}$, is as follows:

- $S = U$, and
- $R^l\langle s_1, t\rangle\langle s_2, t\rangle$ iff $R^r\langle t, s_1\rangle\langle t, s_2\rangle$.

Thus, an extended product model $(S \times U, R^l, R^r, \mathbb{I}, \mathbb{V})$ belongs to $\mathbb{G}$ if, and only if, the part $(S \times U, R^l, R^r)$ is the product frame of some (S, R) and itself. Essentially, there is a precise match between our original models and $\mathbb{G}$.

From standard models to $\mathbb{G}$. A standard model $\mathbf{M}$ can give rise to an extended product model, denoted by $\mathbb{M}^{\mathbf{M}}$, in the following way:

Definition 7 ($\mathbb{M}^{\mathbf{M}}$). *Let* $\mathbf{M} = (W, R, \mathsf{V})$ *be a standard model. The corresponding extended product model* $\mathbb{M}^{\mathbf{M}}$ *is a tuple* $(W \times W, R^l, R^r, \mathbb{I}, \mathbb{V})$ *where*

- $(W \times W, R^l, R^r)$ *is the resulting product frame of* (W, R) *and itself,*
- $\mathbb{I} = \{\langle w, w\rangle \mid w \in W\}$,
- $\langle s,t\rangle \in \mathbb{V}(p_H)$ *iff* $s \in \mathsf{V}(p_H)$,
  $\langle s,t\rangle \in \mathbb{V}(p_S)$ *iff* $t \in \mathsf{V}(p_S)$.

It is easy to check that $\{\mathbb{M}^{\mathbf{M}} \mid \mathbf{M}$ *is a standard model*$\} \subseteq \mathbb{G}$. Now, by induction on formulas, we are able to show that:

Proposition 9. *Let* $\mathbf{M} = (W, R, \mathsf{V})$ *be a standard model and* $s, t \in W$*. Then, for all formulas* $\varphi \in \mathcal{L}$*,* $\mathbf{M}, s, t \vDash \varphi$ *iff* $\mathbb{M}^{\mathbf{M}}, \langle s,t\rangle \vDash \varphi$.

We leave this as an exercise, and now proceed to show the other direction:

From $\mathbb{G}$ to standard models. Every extended product model of $\mathbb{G}$ is associated with a standard one in the following:

[13] It would be instructive to see the connection between the clauses and formulas given by (5) in Section 2.

Definition 8 ($\mathbf{M}^{\mathbb{M}}$). *Let* $\mathbb{M} = (W \times W, R^l, R^r, \mathbb{I}, \mathbb{V})$ *be a* $\mathbb{G}$*-model. Then, we can obtain a standard model* $\mathbf{M}^{\mathbb{M}} = (W, R, \mathsf{V})$ *such that:*

- Rs_1s_2 *iff* $R^l\langle s_1, t\rangle\langle s_2, t\rangle$ *and* $R^r\langle t, s_1\rangle\langle t, s_2\rangle$ *for all* $t \in W$*,*
- $s \in \mathsf{V}(p_H)$ *iff* $\langle s, t\rangle \in \mathbb{V}(p_H)$ *for all* $t \in W$*, and* $s \in \mathsf{V}(p_S)$ *iff* $\langle t, s\rangle \in \mathbb{V}(p_S)$ *for all* $t \in W$*.*

It is simple to see that the resulting $\mathbf{M}^{\mathbb{M}}$ is well-defined. With the above definition, we have:

Proposition 10. *Let* $\mathbb{M} = (W \times W, R^l, R^r, \mathbb{I}, \mathbb{V}) \in \mathbb{G}$*. For all* $\varphi \in \mathcal{L}$*, it holds that* $\mathbb{M}, \langle s, t\rangle \vDash \varphi$ *iff* $\mathbf{M}^{\mathbb{M}}, s, t \vDash \varphi$*.*

Proof. We prove by induction on formulas. We merely show the case for $\langle \mathsf{H}\rangle\varphi$.

Assume that $\mathbb{M}, \langle s, t\rangle \vDash \langle \mathsf{H}\rangle\varphi$. Then, there exists $\langle s', t\rangle \in W \times W$ with $R^l\langle s, t\rangle\langle s', t\rangle$ and $\mathbb{M}, \langle s', t\rangle \vDash \varphi$. Moreover, for all $t_1 \in W$, $R^l\langle s, t_1\rangle\langle s', t_1\rangle$ and $R^r\langle t_1, s\rangle\langle t_1, s'\rangle$. With Definition 8, it holds Rss'. By the inductive hypothesis, $\mathbb{M}, \langle s', t\rangle \vDash \varphi$ is followed by $\mathbf{M}^{\mathbb{M}}, s', t \vDash \varphi$. Therefore, $\mathbf{M}^{\mathbb{M}}, s, t \vDash \langle \mathsf{H}\rangle\varphi$.

For the other direction, suppose that $\mathbf{M}^{\mathbb{M}}, s, t \vDash \langle \mathsf{H}\rangle\varphi$. By the semantics (w.r.t. standard models), there exists $s' \in W$ such that Rss' and $\mathbf{M}^{\mathbb{M}}, s', t \vDash \varphi$. By the inductive hypothesis, the latter is followed by $\mathbb{M}, \langle s', t\rangle \vDash \varphi$. Moreover, Rss' implies that $R^l\langle s, t\rangle\langle s', t\rangle$ (recall Definition 8). So, $\mathbb{M}, \langle s, t\rangle \vDash \langle \mathsf{H}\rangle\varphi$. □

From Propositions 9 and 10, it follows directly that

Theorem 4. *The logic given by* $\mathbb{G}$ *is exactly* LHS*.*

In the sense above, the class of standard models is equivalent to $\mathbb{G}$. Thus, the product framework provides us a new angle to view our logic.

5.2 'Asymmetrizing' domains: rectangle frames

However, it is important to notice that w.r.t. LHS, the class $\mathbb{G}$ has many features that cannot be defined by the logic, e.g., the system cannot define the property that the domain is a product of a set with itself. Thus, we propose the following notion of 'rectangle frames':[14]

Definition 9 (Rectangle frames). *Let frames* $\mathcal{F}_1 = (S, R_1)$ *and* $\mathcal{F}_2 = (U, R_2)$ *be two frames and* $(S \times U, R^r, R^l)$ *be their product frame. A rectangle frame* $\mathsf{F} = (S \times U, R^l, R^r, \mathbb{I})$ *is an extended product frame satisfying the following restriction:*

$$\text{For all } w \in S \cap U,\ R_1(w) = R_2(w) \subseteq S \cap U.$$

We denote by $\mathbb{R}$ *the class of rectangle frames.*

Notice that a point-generated frame of an $\mathbb{R}$-frame is in $\mathbb{R}$ as well, and we in effect can identify $\mathbb{R}$ with their point-generated frames. So, in what follows, we just work with the latter class.

Theorem 5. *The logic captured by* $\mathbb{G}$*-frames is the same as that of* $\mathbb{R}$*.*

[14] To simplify the discussion, in this section we focus on *frames*.

Proof. It is easy to see that each $\mathbb{G}$-frame is an $\mathbb{R}$-frame. The crucial part is to show that *each $\mathbb{R}$-frame is a point-generated frame of some $\mathbb{G}$-frame.*

To prove this, we need to present a method illustrating how to construct a $\mathbb{G}$-frame $\mathcal{G} = (W \times W, R^l, R^r, \mathbb{I})$ from a given $\mathbb{R}$-frame $\mathcal{R} = (S \times U, R_1^l, R_2^r, \mathbb{I}')$. Details are as follows:

- $W := S \cup U$
- For all $\langle s_1, t_1\rangle, \langle s_2, t_2\rangle \in W \times W$,
  $R^l\langle s_1, t_1\rangle\langle s_2, t_2\rangle$ iff $t_1 = t_2$ and ($R_1 s_1 s_2$ or $R_2 s_1 s_2$).
  $R^r\langle s_1, t_1\rangle\langle s_2, t_2\rangle$ iff $s_1 = s_2$ and ($R_1 t_1 t_2$ or $R_2 t_1 t_2$).
- $\langle s_1, t_1\rangle \in \mathbb{I}$ iff $s_1 = t_1$.

One can check that the resulting frame is a $\mathbb{G}$-frame. Now we denote by $\langle m, n\rangle$ the root of $\mathcal{R}$, and let us proceed to show that $\mathcal{R}$ is a point-generated frame of $\mathcal{G}$. To achieve our goal, we just need to show that a state $\langle s, t\rangle \in \mathcal{G}$ that can be reached from $\langle m, n\rangle$ in one step is a state of $\mathcal{R}$ as well: repeating the reasoning, we can show that all states of $\mathcal{G}$ that are reached from $\langle m, n\rangle$ in i steps are also states of $\mathcal{R}$. We just show the case for $R^l\langle m, n\rangle\langle s, t\rangle$, and that for R^r is similar.

Immediately, $n = t$. Also, we have $R_1 ms$ or $R_2 ms$. If $R_2 ms$ is the case, then from $\langle m, n\rangle \in S \times U$ it follows $m \in S \cap U$. Thus, by the definition of $\mathbb{R}$, $R_2 ms$ implies that $R_1 ms$. So, it suffices to consider $R_1 ms$ only. Now, by the definition of R_1^l, it holds $R_1^l\langle m, n\rangle\langle s, n\rangle$. This completes the proof. ◻

We end this part by noting that many earlier notions and results for LHS can be transferred into this new setting easily, including those involving bisimulation and first-order translation. Conversely, we note that there are a large number of general results and techniques for product logics, and it is interesting to investigate which ones can transfer to our logic. We believe the connections established in this section may shed light on further study of LHS, e.g., its axiomatization, that is left as future work:

Open problem. Is LHS finitely axiomatizable?

A possible way to answer this might be to analyze the counterpart of LHS in product logic, in which the notion of rectangle frames and its possible further generalizations might be a starting point [18].

6 Related works

Graph games and modal logics. Motivated by a simple graph game of hide and seek, this work belongs to a broader program [10] that promotes a study of graph game design in tandem with matching new modal logics. As stated earlier, this paper is an extension of [31]. In recent years, several interesting graph games have been studied. For instance, in sabotage games [9], a player moves along a link available to her on a graph to reach some fixed goal region, while her opponent removes an arbitrary link in each round to prevent her from reaching her goal. The games are captured by the sabotage modal logic (SML), which was presented first in [11], since then its logical properties have been studied. [2, 4] provided a first-order translation for SML, which together with [1] proposed a notion of bisimulation for the logic. [1, 2, 32] showed that

SML has a PSPACE-complete model checking problem and an undecidable satisfiability problem. Also, [12] provided a Hilbert-style calculus for the logic extending SML with formulas of *hybrid logic* [15]. Sabotage games and its logic also have many variants that were studied in-depth (see, e.g., [36]), and we refer the readers to [5] for extensive references to modal logics for graph changes.

Games in which links are removed *locally* according to certain conditions which were expressed explicitly in the language have been studied in [30]. Moreover, several variants of sabotage games were applied to the learning/teaching scenarios [23], and their computational behaviors were analyzed. Following this direction, a new game setting allowing both link deletion and link addition was developed in [7] to capture some interesting features of the learning process. Closely related to [7], a class of *relation-changing logics*, containing operators to swap, delete or add links, was explored in [3, 1]. Instead of modifying links, in poison games [19], a player can poison a node to make it unavailable to the opponent. These games have been studied with diverse modal approaches in [16, 25]. Additionally, by updating valuation functions of models, a dynamic logic of local fact change was studied in [38], which captures a class of graph games in which properties of states might get affected by those of others.

Product logics with diagonal constant. As illustrated in the previous sections, technically our framework is close to many-dimensional modal logics [33, 20]. In particular, a class of product logics was studied in [27, 28, 26] with the *diagonal constant* δ.[15] We focus on $\mathbf{K} \times^{\delta} \mathbf{K}$, which has been used to denote different logics in the literature: [28, 26] use it for a product logic augmenting $\mathbf{K} \times \mathbf{K}$ with δ, while [27] uses it for a product logic whose frame $(W \times W, R^r, R^l)$ is the product of a frame (W, R) with itself. A crucial difference between our logic and $\mathbf{K} \times^{\delta} \mathbf{K}$ in [27] (and, in [28, 26]) is that propositional variables in LHS are two-sorted. This seemingly innocent feature may have interesting consequences, which we leave for future work. In what follows, we will compare our logic with $\mathbf{K} \times^{\delta} \mathbf{K}$ as presented in [28, 26]: it was shown that the logic $\mathbf{K} \times^{\delta} \mathbf{K}$ lacks the finite model property and is undecidable, which seem very similar to our results at a first glance. However, our logic differs from this one both conceptually and technically.

First, our formulas are evaluated at *pairs* of states, where each of the states can occur by itself (and, not just as a constituent of an ordered pair), which makes it possible for us to study the relationship between two states directly. In $\mathbf{K} \times^{\delta} \mathbf{K}$, even though formulas are evaluated at pairs of states, these pairs themselves form nodes in the domain. As a result, product logic cannot express the more fine-grained relation (i.e., identity) between the two components forming a pair. In [28, 26], δ is interpreted as a special subset of the domain, not necessarily consisting of pairs formed by the same components from those dimensions. Therefore, we can say that constant I works at a meta level while δ in [28, 26] is a notion of the object level. Notice that this is essentially a difference between our original proposal and those explored in Section 5. But definitely, this does not exclude a possible 'mixture' of the two frameworks. As suggested in Section 5, technically LHS can be reduced to product logics with δ. On the other hand, product models themselves can also be viewed as special models (with two relations) for LHS (and then I denotes the identity of *two* pairs).

[15] In two dimensional models δ holds at a state (s, t) just in the case that $s = t$.

Next, techniques adopted to establish the undecidability of LHS are very different. Similar to all other product logics, various relations representing transitions of states in different dimensions are considered in [28, 26]. Moreover, the product nature endows the relations with possible interactions: say, *commutativity* and *confluence*. With such interactions, product logics obtain grid-like structures automatically. However, as illustrated in our proofs, a crucial step in proving undecidability of LHS was exactly to build such a shape. In other words, these extra efforts make our proof technically non-trivial.

7 Conclusion and future work

Summary. Motivated by the hide and seek game, this paper studies a modal logic LHS that allows us to talk about moves for each player, as well as the situation of meeting. Specifically, formulas in this logic are evaluated at two states of the domain, representing positions of different players. A constant I expressing the meeting of two players is explored in depth, which adds a natural and novel treatment of equality in modal logics. We establish a series of results concerning its expressive power and computational behavior. A new notion of bisimulation for LHS is proposed, and is compared systematically with those of related logics. The model checking problem for LHS is proved to be P-complete. We have also shown that the logic does not enjoy the tree model property or the finite model property, and its satisfiability problem is undecidable, which refutes a conjecture made by van Benthem and Liu in their recent paper [10]. Finally, we looked into the connections between our logic and the framework of product logics, which shed light on further study of LHS.

Further directions. We mention a few directions that are worth pursuing further. Several open problems have been formulated along the way, including the axiomatization of LHS, and issues regarding its expressive power. Regarding the language, the constant I seems rather simple and innocent, but surprisingly, our logic turned out to be undecidable. It makes sense to understand this phenomenon better, and possibly by investigating some alternative logics (e.g., the logic mentioned in Remark 1). In Sections 5 and 6, we have seen certain similarities/differences between our work and product logics, but many more issues remain to be explored along this direction. We are aware that product logics have various extensions with promising applications, e.g., hybrid product logics [37], and it would be interesting to consider some natural extensions of LHS, too. As stated earlier, we have taken a high-level modeller's perspective to study the hide and seek game. We reason about players' observations and moves with the assumption that the whole graph and the players' positions at each stage of the game are available to us. Pursuing strategic reasoning from the players' perspectives *in* the game would be a natural next step.

Finally, as mentioned in various places, our work has a natural connection with the game of cops and robber in the vast literature of graph games (see, e.g., [17, 34]). We are exploring richer versions of these games, focusing on different characterization results of *cop-win graphs*, mostly from the players' perspectives. We have extended these logical frameworks of games on graphs with modal substitution operators [39] which enable us to express winning positions of players in the general sense that we discussed in Section 2. We will continue this line of research in the future.

Acknowledgements. We thank Johan van Benthem for his inspiring suggestions. We wish to thank the anonymous reviewers for their helpful comments for improvements. Sujata Ghosh acknowledges Department of Science and Technology, Government of India for financial support vide Reference No DST/CSRI/2018/202 under Cognitive Science Research Initiative (CSRI) to carry out this work. Dazhu Li is supported by the National Social Science Foundation of China [22CZX063], Fenrong Liu and Yaxin Tu are supported by Tsinghua University Initiative Scientific Research Program.

References

1. Areces, C., Fervari, R., Hoffmann, G.: Moving arrows and four model checking results. In: Ong, L., de Queiroz, R. (eds.) Proceedings of WoLLIC '2012. LNCS, vol. 7456, pp. 142–153 (2012)
2. Areces, C., Fervari, R., Hoffmann, G.: Relation-changing modal operators. Logic Journal of the IGPL **23**, 601–627 (2015)
3. Areces, C., Fervari, R., Hoffmann, G., Martel, M.: Satisfiability for relation-changing logics. Journal of Logic and Computation **28**, 1443–1470 (2018)
4. Aucher, G., van Benthem, J., Grossi, D.: Sabotage modal logic: Some model and proof theoretic aspects. In: van der Hoek, W., Holliday, W., Wang, W. (eds.) Proceedings of LORI '2015. LNCS, vol. 9394, pp. 1–13 (2015)
5. Aucher, G., van Benthem, J., Grossi, D.: Modal logics of sabotage revisited. Journal of Logic and Computation **28**(2), 269–303 (2018)
6. Baltag, A., van Benthem, J.: A simple logic of functional dependence. Journal of Philosophical Logic **50**, 939–1005 (2021)
7. Baltag, A., Li, D., Pedersen, M.Y.: On the right path: A modal logic for supervised learning. In: Blackburn, P., Lorini, E., Guo, M. (eds.) Proceedings of LORI '2019, LNCS, vol. 11813, pp. 1–14. Springer (2019)
8. van Benthem, J.: Modal Logic for Open Minds. CSLI Publications (2010)
9. van Benthem, J.: Logic in Games. The MIT Press (2014)
10. van Benthem, J., Liu, F.: Graph games and logic design. In: Liu, F., Ono, H., Yu, J. (eds.) Knowledge, Proof and Dynamics, p. 125–146. Springer (2020)
11. van Benthem, J.: An essay on sabotage and obstruction. In: Hutter, D., Stephan, W. (eds.) Mechanizing Mathematical Reasoning. LNCS, vol. 2605, pp. 268–276 (2005)
12. van Benthem, J., Li, L., Shi, C., Yin, H.: Hybrid sabotage modal logic. Journal of Logic and Computation (2022)
13. Berger, R.: The Undecidability of the Domino Problem. AMS (1966)
14. Blackburn, P., de Rijke, M., Venema, Y.: Modal Logic. Cambridge University Press (2001)
15. Blackburn, P., Seligman, J.: Hybrid languages. Journal of Logic, Language and Information **4**, 251–272 (1995)
16. Blando, F.Z., Mierzewski, K., Areces, C.: The modal logics of the poison game. In: Liu, F., Ono, H., Yu, J. (eds.) Knowledge, Proof and Dynamics. pp. 3–23. Logic in Asia: Studia Logica Library, Springer (2020)
17. Bonato, A., Nowakowski, R.J.: The Game of Cops and Robbers on Graphs. AMS (2011)
18. Chen, Q., Li, D.: Logics of the hide and seek game: expressiveness, decidability and axiomatization. Manuscript (2023)

19. Duchet, P., Meyniel, H.: Kernels in directed graphs: a poison game. Discrete Mathematics **115**, 273–276 (1993)
20. Gabbay, D., Kurucz, A., Wolter, F., Zakharyaschev, M.: Many-Dimensional Modal Logics: Theory and Applications. Elsevier (2003)
21. Gabbay, D., Shehtman, V.: Products of modal logics. Part I. Journal of the IGPL **6**, 73–146 (1998)
22. Gabbay, D., Shehtman, V.: Products of modal logics. Part II. Journal of the IGPL **2**, 165–210 (2000)
23. Gierasimczuk, N., Kurzen, L., Velázquez-Quesada, F.R.: Learning and teaching as a game: A sabotage approach. In: He, X., Horty, J., Pacuit, E. (eds.) Proceedings of LORI '2009, LNCS, vol. 5834, pp. 119–132. Springer (2009)
24. Goldfarb, W.: The unsolvability of the Gödel class with identity. Journal of Symbolic Logic **49**, 1237–1252 (1984)
25. Grossi, D., Rey, S.: Credulous acceptability, poison games and modal logic. In: Agmon, N., Taylor, M.E., Elkind, E., Veloso, M. (eds.) Proceedings of AAMAS 2019. pp. 1994–1996 (2019)
26. Hampson, C., Kikot', S., Kurucz, A.: The decision problem of modal product logics with a diagonal, and faulty counter machines. Studia Logica **104**, 455–486 (2016)
27. Kikot', S.P.: Axiomatization of modal logic squares with distinguished diagonal. Mathematical Notes **88**, 238–250 (2010)
28. Kurucz, A.: Products of modal logics with diagonal constant lacking the finite model property. In: Ghilardi, S., Sebastiani, R. (eds.) Frontiers of Combining Systems. LNCS, vol. 5749, pp. 279–286 (2009)
29. Kurucz, A.: Combining modal logics. In: Blackburn, P., van Benthem, J., Wolter, F. (eds.) Handbook of Modal Logic, Studies in Logic and Practical Reasoning, vol. 3, pp. 869–924. Elsevier (2007)
30. Li, D.: Losing connection: the modal logic of definable link deletion. Journal of Logic and Computation **30**, 715–743 (2020)
31. Li, D., Ghosh, S., Liu, F., Tu, Y.: On the subtle nature of a simple logic of the hide and seek game. In: Silva, A., Wassermann, R., de Queiroz, R. (eds.) Proceedings of WoLLIC '2021. LNCS, vol. 13038, pp. 201–218 (2021)
32. Löding, C., Rohde, P.: Model checking and satisfiability for sabotage modal logic. In: Pandya, P.K., Radhakrishnan, J. (eds.) FSTTCS 2003. LNCS, vol. 2914, pp. 302–313 (2003)
33. Marx, M., Venema, Y.: Multi-Dimensional Modal Logic. Kluwer Academic Publishers (1997)
34. Nowakowski, R., Winkler, R.P.: Vertex-to-vertex pursuit in a graph. Discrete Math **13**, 235–239 (1983)
35. Pützstück, P.: Decidability and Bisimulation for Logics of Functional Dependence. Bachelor's thesis, Mathematical Foundations of Computer Science, RWTH-Aachen University (2020)
36. Rohde, P.: On games and logics over dynamically changing structure. Ph.D. thesis, RWTH Aachen University (2005)
37. Sano, K.: Axiomatizing hybrid products: How can we reason many-dimensionally in hybrid logic? Journal of Applied Logic **8**, 459–474 (2010)
38. Thompson, D.: Local fact change logic. In: Liu, F., Ono, H., Yu, J. (eds.) Knowledge, Proof and Dynamics. pp. 73–96. Logic in Asia: Studia Logica Library, Springer (2020)
39. Tu, Y., Ghosh, S., Li, D., Liu, F.: Modal substitution logic. Manuscript (2023)
40. Vardi, M.: On the complexity of bounded–variable queries. In: Proceedings of the ACM 14th Symposium on Principles of Database Systems. pp. 266–276 (1995)