Probabilistic Strategy Logic with Degrees of Observability

Chunyan Mu[1], Nima Motamed[2], Natasha Alechina[3,2], Brian Logan[1,2]

[1]Department of Computing Science, University of Aberdeen
[2]Department of Information and Computing Sciences, Utrecht University
[3]Department of Computer Science, Open University Netherlands
chunyan.mu@abdn.ac.uk, n.motamed@uu.nl, n.a.alechina@uu.nl, brian.logan@abdn.ac.uk

Abstract

There has been considerable work on reasoning about the strategic ability of agents under imperfect information. However, existing logics such as Probabilistic Strategy Logic are unable to express properties relating to information transparency. *Information transparency* concerns the extent to which agents' actions and behaviours are observable by other agents. Reasoning about information transparency is useful in many domains including security, privacy, and decision-making. In this paper, we present a formal framework for reasoning about information transparency properties in stochastic multi-agent systems. We extend Probabilistic Strategy Logic with new observability operators that capture the degree of observability of temporal properties by agents. We show that the model checking problem for the resulting logic is decidable.

1 Introduction

Multi-Agent Systems (MASs) often involve agents that operate autonomously and interact with each other in dynamic and sometimes adversarial ways. Understanding the transparency and observability of these interactions is crucial for ensuring secure, efficient, and cooperative behaviour. In particular, information transparency and agent observability directly impact the security and privacy of MASs in which agents share information and where there is a risk of unintentional data leakage. Analysing information transparency helps identify potential sources of data leakage and design mechanisms to prevent it. The ability to control what agents can observe and the information they can induce is crucial for safeguarding sensitive data and preventing information leakage. In addition, the decision-making processes of agents are influenced by the information they possess about each other's behaviours and intentions. Quantified analysis of information transparency and agent observability plays a key role in determining the accuracy and effectiveness of decision-making within MASs.

This paper addresses the challenge of specifying, verifying and reasoning about information transparency properties within MASs. In particular, we specify observability properties from a standpoint of information transparency within the *opacity* framework (Bryans et al. 2005). A property Φ is considered to be opaque, if for every behaviour π satisfying Φ there is a behaviour π' violating Φ such that π and π' are observationally equivalent, so that an observer can never be sure if Φ holds or not. Opacity (and its negation, observability) are so called *hyperproperties* that relate multiple execution traces. Verification of hyperproperties is an emerging and challenging topic, see, for example, (Beutner and Finkbeiner 2023).

In order to reason about the observability of behaviours, we introduce *Opacity Probabilistic Strategy Logic* (oPSL), an extension of Probabilistic Strategy Logic (PSL) which allows quantitative analysis of information transparency. oPSL enables us to specify the degree of transparency in system behaviours to an observer under a binding of agents' strategies, taking into account predefined observability of behaviours for the observer. We use a very general approach to observability which allows us to reason both about observability of state properties, and observability of actions. Using the concept of observability allows, for example, the identification of cases where agents observe sensitive information, disclosing potential information leakage or unauthorised access attempts.

We introduce a novel framework for systematically analysing information transparency in stochastic multi-agent systems. The key contributions include:

- A definition of agent observability and information transparency concerning agent behaviours in the context of partially observable stochastic multi-agent systems with concurrent and probabilistic behaviours.
- The introduction of Opacity Probabilistic Strategy Logic (oPSL) incorporating a new observability and degree of observability operators, allowing precise representation of observability properties that are challenging to express in standard logics for MAS.
- Showing that the model-checking problem for oPSL with memoryless strategies is decidable in 3EXPSPACE.

The framework facilitates formal reasoning about agent observability and information transparency analysis in MAS, with applications in security, privacy, game theory, and AI.

The remainder of this paper is organised as follows. In Section 2 we discuss related work. In Section 3 we formalise stochastic multi-agent systems as partially observable stochastic models. Section 4 extends Probabilistic Strategic Logic (PSL), incorporating new operators for quantified observability analysis. Section 5 presented algorithmic approaches for systematic observability assessment within the PSL model checking framework. In Section 6 we conclude and discuss possible directions for future work.

2 Related Work

There has been a significant amount of work on logical specification and formal verification of MAS with imperfect information. Various methods have been developed to support the model checking of such logics, such as those focusing on epistemic logics for knowledge about the state of the system, e.g., (Lomuscio, Qu, and Raimondi 2017; Belardinelli et al. 2017; Kwiatkowska et al. 2019; Belardinelli et al. 2020; Ballot et al. 2024; Malvone, Murano, and Sorrentino 2017). Actions and knowledge in strategic settings has been investigated in (Schobbens 2003; Goranko and Jamroga 2004; Ågotnes 2006). These approaches focus on formulas true in all states or in all histories indistinguishable from the current one. In contrast, we are interested in properties that are either true on all observable traces, or false on all observable traces. In this we follow (Bryans et al. 2005) which introduced the *opacity* framework, which is based on being able to distinguish globally observable or unobservable properties.

Probabilistic Alternating-Time Temporal Logic (PATL*) was proposed in (Chen and Lu 2007). Huang et al. (2012) proposed an incomplete information version of PATL*, providing a framework for reasoning about possible states and actions based on agents' beliefs and strategies. However, it does not directly address information exposure resulting from the observation of agents' actions and behaviours or the security of information flows. Mu and Pang (2023) developed a framework for specifying and verifying opacity in PATL. However, we propose an enhanced framework based on PSL, which offers greater expressiveness and flexibility.

Strategy logic with imperfect information (Berthon et al. 2021) extends traditional modal logic to reason about strategic interactions in situations of imperfect information. It captures agents' decision-making abilities taking into account their limited knowledge. Additionally, work by (Ferrando and Malvone 2023) addressed the verification of logics for strategic reasoning in contexts with imperfect information and perfect recall strategies, employing sub-model generation and Computation Tree Logic (CTL*) model checking. A probabilistic extension of Strategy Logic (PSL) was introduced by (Aminof et al. 2019) for stochastic systems; this work also investigated model-checking problems for agents with perfect and imperfect recall.

Our work is relevant to research on information flow security awareness analysis and verification, and also contributes to the quantified information flow security landscape. While previous studies focused on imperative modelling languages and probabilistic aspects (Alvim et al. 2020) in secure computing systems, there has been little consideration of the dynamic collaboration, interaction, and decision-making patterns found in multi-agent scenarios. In contrast, this paper explores observability issues in MAS from a novel perspective of information transparency, and focuses on quantified information flow security awareness for MAS.

3 Partially Observable MAS

In this section, we introduce *partially observable multi-agent systems* which constitute the formal basis of our approach.

We write $\mathsf{Dist}(X)$ for the set of all discrete probability distributions over a set X, i.e., all functions $\mu : X \to [0, 1]$ s.t. $\sum_{x\in X} \mu(x) = 1$. Given a set X, we denote by 2^X the power set of X, by X^* the set of all finite words over X, and by X^ω the set of all infinite words over X.

Definition 1. A *stochastic transition system* is a tuple $\mathcal{G} = (\mathsf{Ag}, S, \mathsf{Act}, T)$, where

- $\mathsf{Ag} = \{1, \dots, n\}$ is a finite set of *agents*;
- S is a finite set of *states*;
- Act is a finite set of *actions*;
- $T : S \times \mathsf{Act}^{\mathsf{Ag}} \to \mathsf{Dist}(S)$ is a *transition function*.

At each time step, the agents simultaneously choose actions from Act, producing an *action profile* $\alpha = (a^1, \dots, a^n) \in \mathsf{Act}^{\mathsf{Ag}}$. The transition function T specifies for each state s and such an action profile α, a probability distribution $T(s, \alpha) \in \mathsf{Dist}(S)$ over *next* states: $T(s, \alpha)(s')$ is the probability of moving to s' from s when the agents choose α. We write $s \xrightarrow{\alpha} s'$ whenever $T(s, \alpha)(s') > 0$.

Definition 2. Given a stochastic transition system $\mathcal{G}$, a $\mathcal{G}$-path is an infinite sequence $\pi = (s_0, \alpha_0)(s_1\alpha_1)\dots$ of pairs of states s_i and action profiles α_i, such that $s_i \xrightarrow{\alpha_i} s_{i+1}$ for all i. We will usually write $\pi = s_0\alpha_0 s_1\alpha_1 \cdots$ for simplicity of notation. The state s_0 is referred to as the *initial state* of π. We denote by $\pi_{\mathsf{s}}(j)$ the jth state s_j of π, and by $\pi_{\mathsf{a}}(j)$ the jth action profile α_j of π. Furthermore, we denote by $\pi_{\le j}$ the prefix of π up to the jth state, i.e. $\pi_{\le j} = s_0\alpha_0 s_1 \dots s_j$. Given a state s, we write $\mathsf{Paths}_{\mathcal{G}}(s)$ for the set of all $\mathcal{G}$-paths π with initial state s.

We also define *histories* as *finite* sequences $h = s_0\alpha_0 \cdots s_n$, such that $s_i \xrightarrow{\alpha_i} s_{i+1}$ for all $i < n$. We denote the set of all histories h with initial state s by $\mathsf{Hist}_{\mathcal{G}}(s)$. Given a history $h = s_0\alpha_0 \cdots s_n$, we write $last(h)$ for its final state $last(h) = s_n$.

Example 1. Consider the simple message interception scenario illustrated in Figure 1. We have: $\mathsf{Ag} = \{1, 2\}$, $\mathsf{Act}_1 = \{\mathtt{send}, \mathtt{wait}\}$, $\mathsf{Act}_2 = \{\mathtt{copy}, \mathtt{wait}\}$, $S = \{s_0, s_1, s_2, s_3\}$.

Example paths from s_0 include:

$$\pi = s_0 \xrightarrow{(\mathtt{send},\mathtt{copy})} s_1 \xrightarrow{(\mathtt{send},\mathtt{copy})} s_1 \xrightarrow{(\mathtt{send},\mathtt{copy})} s_1 \dots$$

$$\pi' = s_0 \xrightarrow{(\mathtt{send},\mathtt{copy})} s_3 \xrightarrow{(\mathtt{send},\mathtt{copy})} s_3 \xrightarrow{(\mathtt{send},\mathtt{copy})} s_3 \dots$$

Example histories include the following prefixes of π and π':

$$h = s_0 \xrightarrow{(\mathtt{send},\mathtt{copy})} s_1 \xrightarrow{(\mathtt{send},\mathtt{copy})} s_1$$

$$h' = s_0 \xrightarrow{(\mathtt{send},\mathtt{copy})} s_3 \xrightarrow{(\mathtt{send},\mathtt{copy})} s_3$$

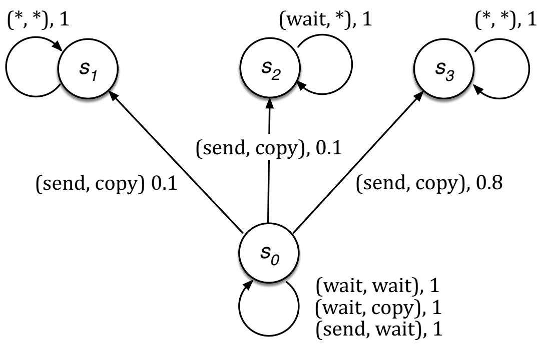

Figure 1: Simple message interception scenario

Observation of behaviours To model the observational capabilities of an agent, we use a set of *observables*, distinct from the states and actions of the transition system. We consider two types of observables: *state* and *action* observables. We denote by Θ_{s} the finite set of state observables, and by Θ_{a} the finite set of action observables. The sets Θ_{s} and Θ_{a} contain distinguished 'invisible' state/action observables $\circ$ and ϵ, respectively. Observables are then defined as pairs of action and state observables, i.e., $\Theta := \Theta_{\mathsf{a}} \times \Theta_{\mathsf{s}}$. An *observation function* is a pair $(\mathsf{obs}^{\mathsf{a}}, \mathsf{obs}^{\mathsf{s}})$ of functions $\mathsf{obs}^{\mathsf{a}} \colon \mathsf{Act}^{\mathsf{Ag}} \to \Theta_{\mathsf{a}}$ and $\mathsf{obs}^{\mathsf{s}} \colon S \to \Theta_{\mathsf{s}}$. For notational convenience, we combine these into a single function $\mathsf{obs} \colon \mathsf{Act}^{\mathsf{Ag}} \times S \to \Theta$ (i.e., with $\mathsf{obs}(\alpha, s) = (\mathsf{obs}^{\mathsf{a}}(\alpha), \mathsf{obs}^{\mathsf{s}}(s))$.

We will often lift observation functions to functions $\mathsf{obs} \colon \mathsf{Paths}_{\mathcal{G}}(s) \to \Theta^{\omega}$ operating on paths by letting $\mathsf{obs}(s_0\alpha_0 s_1 \cdots) = \epsilon\ \mathsf{obs}^{\mathsf{s}}(s_0)\mathsf{obs}^{\mathsf{a}}(\alpha_0)\mathsf{obs}^{\mathsf{s}}(s_1)\cdots$, where ϵ is the distinguished action observable, and lift obs to sets of paths $X \subseteq \mathsf{Paths}_{\mathcal{G}}(s)$ by letting $\mathsf{obs}(X) = \{\mathsf{obs}(\pi) \mid \pi \in X\}$. Finally, given a set $X \subseteq \mathsf{Paths}_{\mathcal{G}}(s)$, we write $\mathsf{obs}^{\sim}(X)$ for the set of all $\pi \in \mathsf{Paths}_{\mathcal{G}}(s)$ for which there exists $\pi' \in X$ with $\mathsf{obs}(\pi) = \mathsf{obs}(\pi')$. In other words, $\mathsf{obs}^{\sim}(X)$ is the set of all paths that are observationally equivalent to some path in X.

Definition 3. A *partially observable multi-agent system* is a tuple $\mathcal{M} = (\mathcal{G}, \mathsf{Ap}, L, \{\mathsf{obs}_i\}_{i \in \mathsf{Ag}})$, where:

- $\mathcal{G} = (\mathsf{Ag}, S, \mathsf{Act}, T)$ is a stochastic transition system;
- Ap is a finite set of *atomic propositions*;
- $L : S \to 2^{\mathsf{Ap}}$ is a *state labelling function*;
- $\mathsf{obs}_i \colon \mathsf{Act}^{\mathsf{Ag}} \times S \to \Theta$ is the observation function of i.

Example 2. The following partially observable multi-agent system models the running example introduced in Example 1.

Let the labelling L be as follows: $L(s_0) = \{init\}$, $L(s_1) = \{stolen\}$, $L(s_2) = \{stolen, warning\}$ and $L(s_3) = \emptyset$.

Let the observation function of agent 1 be as follows:

$$\begin{aligned}
\mathsf{obs}_1^s(s_0) &= init \\
\mathsf{obs}_1^s(s_1) = \mathsf{obs}_1^s(s_3) &= \circ \\
\mathsf{obs}_1^s(s_2) &= warning \\
\mathsf{obs}_1^a(\mathtt{send}, \mathtt{wait}) = \mathsf{obs}_1^a(\mathtt{send}, \mathtt{copy}) &= (\mathtt{send}, \epsilon) \\
\mathsf{obs}_1^a(\mathtt{wait}, \mathtt{wait}) = \mathsf{obs}_1^a(\mathtt{wait}, \mathtt{copy}) &= (\mathtt{wait}, \epsilon)
\end{aligned}$$

Observations of agent 2 are perfect, that is $\mathsf{obs}_2^s(s_i) = s_i$ and $\mathsf{obs}_2^a(\alpha_i) = \alpha_i$ for every state s_i and joint action α_i.

We then have:

$$\mathsf{obs}_1(h) = \mathsf{obs}_1(h') = init \xrightarrow{(\mathtt{send}, \epsilon)} \circ \xrightarrow{(\mathtt{send}, \epsilon)} \circ$$

This implies that agent 1 is not able to distinguish h and h' under obs_1. The same holds for the paths π and π': they are observationally equivalent for agent 1, $\pi' \in \mathsf{obs}^{\sim}(\{\pi\})$.

Agent interactions are governed by strategies informed by their observations. These strategies are often referred to as *uniform strategies*: they do not rely directly on the partially observable multi-agent system, but instead on the observables the model produces.

Definition 4. A *strategy* is a function $\sigma \colon \Theta^* \to \mathsf{Dist}(\mathsf{Act})$. We write Σ for the set of all possible strategies. A *memoryless strategy* is a function $\sigma : \Theta_{\mathsf{s}} \to \mathsf{Dist}(\mathsf{Act})$, i.e., the decision made by the strategy is solely dependent on the observation of the current state, and not on the entire history of observations. A *strategy profile* (resp. *memoryless strategy profile*) is a tuple $\boldsymbol{\sigma} = (\sigma_1, \sigma_2, \ldots, \sigma_n) \in \Sigma^{\mathsf{Ag}}$ of strategies (resp. memoryless strategies) σ_i for each agent i.

Note that memoryless strategy profiles allow us to transform a partially observable multi-agent system and strategy profile into Markov chain, which we consider to consist of a set of states X together with a transition function $t \colon X \to \mathsf{Dist}(X)$. Given a partially observable multiagent system $\mathcal{M}$ (with set of states S and transition function T), and a memoryless strategy profile $\boldsymbol{\sigma}$, we define the Markov chain $M_{\boldsymbol{\sigma}} = (S, t_{\boldsymbol{\sigma}})$ by letting

$$t_{\boldsymbol{\sigma}}(s_1)(s_2) = \sum_{\alpha \in \mathsf{Act}^{\mathsf{Ag}}} T(s_1, \alpha)(s_2) \prod_{i \in \mathsf{Ag}} \sigma_i(\mathsf{obs}_i^{\mathsf{s}}(s_1))(\alpha_i).$$

4 The Logic oPSL

Strategy Logic (Chatterjee, Henzinger, and Piterman 2010) and Probabilistic Strategic Logic (PSL) (Aminof et al. 2019) are formal languages designed for reasoning about multi-agent systems, where autonomous agents make strategic decisions in an environment. These logics focus on capturing the strategic interactions among agents in a multi-agent system, where agents are viewed as rational entities capable of reasoning about their actions and the actions of others. PSL extends Strategy Logic to incorporate uncertainty and probabilistic reasoning, and was designed for scenarios where agents operate in an environment with inherent randomness. PSL formulas are interpreted on multi-agent stochastic transition systems where transitions result from the concurrent actions of agents. An agent's strategy determines the probability that the agent will select a given action in any given situation, typically based on the system's historical evolution.

We extend PSL (Aminof et al. 2019) to allow reasoning about observability operators and strategic abilities of agents. This syntax allows us to specify a wide range of properties and conditions in oPSL, including those related to observability from the perspectives of different agents. It also supports probability-related operations for modelling probabilistic behaviours in MAS. The key addition is the *observability formulae* $\odot_i\Phi$ and the *degree of observability* terms $\mathbf{D}_{\beta,i}(\Phi)$.

Before we introduce the syntax, we first define *bindings* and *valuations*, which are used in the syntax to determine which strategies are used by which agents.

Definition 5. Let Var be a set of *variables*. A *binding* is a function $\beta : \mathsf{Ag} \to \mathsf{Var}$ assigning variables to agents. A *valuation* is a *partial* function $\nu : \mathsf{Var} \rightharpoonup \Sigma$ assigning strategies to some variables. Given a valuation ν, a strategy σ, and a variable x, we denote by $\nu[x \mapsto \sigma]$ the valuation defined as $\nu[x \mapsto \sigma](x) := \sigma$ and $\nu[x \mapsto \sigma](y) := \nu(y)$ for all $y \neq x$.

Intuitively, bindings tell agents which strategies, represented by variables, they should follow, while valuations determine which strategies variables refer to. Note that if the range of β is contained in the domain of ν then composing them produces a strategy profile $\nu \circ \beta \in \Sigma^{\mathsf{Ag}}$.

Definition 6. The *syntax* of oPSL includes three classes of formulae: *history formulae*, *path formulae*, and *arithmetic terms* ranged over by φ, Φ, and τ respectively.

$$\begin{array}{lcl} \varphi & ::= & p \mid \neg\varphi \mid \varphi \vee \varphi \mid \exists x.\varphi \mid \tau \bowtie \tau \mid \odot_i\Phi \\ \Phi & ::= & \varphi \mid \neg\Phi \mid \Phi \vee \Phi \mid \mathbf{X}\Phi \mid \Phi\mathbf{U}\Phi \\ \tau & ::= & c \mid \tau^{-1} \mid \tau \oplus \tau \mid \mathbf{P}_\beta(\Phi) \mid \mathbf{D}_{\beta,i}(\Phi) \end{array}$$

where $p \in \mathsf{Ap}$ is an *atomic proposition*, $x \in \mathsf{Var}$ is a variable, $\bowtie \in \{>, <, =\}$, $i \in \mathsf{Ag}$, $c \in \mathbb{Q}$, $\oplus \in \{+, \times\}$, and $\beta : \mathsf{Ag} \to \mathsf{Var}$ is a binding.

As for PSL, in oPSL the variables $x \in \mathsf{Var}$ represent variations in agent strategies and are quantified over in state formulas like $\exists x.\varphi$. The path formulas contain standard LTL temporal operators: $\mathbf{X}\Phi$ means 'Φ holds in the next step', and $\Phi\mathbf{U}\Psi$ means 'Φ holds until Ψ holds'. The arithmetic term $\mathbf{P}_\beta(\Phi)$ indicates the probability that, given agent i employs strategy $\beta(i)$ for each $i \in \mathsf{Ag}$, a random path in the system satisfies Φ.

The key addition to the logic is the observability operator and the degree of observability. The observability operator $\odot_i\Phi$ states that the behaviours satisfying Φ are observable to agent $i \in \mathsf{Ag}$. Specifically, it requires that for each path π satisfying Φ, there is no observationally equivalent path π' violating Φ from agent i's perspective. This operator allows us to assess the observational capabilities of agent i regarding system behaviours expressed by property Φ. The quantitative observability term $\mathbf{D}_{\beta,i}(\Phi)$ expresses the degree of observability of agent i with respect to Φ-paths under strategy bindings β, which is defined as the probability of having a Φ-path that is not observationally equivalent to a $\neg\Phi$-path, conditional on having a Φ-path.

Before we provide the semantics, we briefly comment on how probabilities of sets of paths are calculated: the following is largely standard in the literature on probabilistic infinite-trace temporal logics, but we restate it hear to clarify the notation. Given a history h with initial state s in $\mathcal{G}$, the *cone* of h, denoted by $\langle h \rangle$, is the set of all paths π with initial state s that extend h, i.e., such that $\pi = h\alpha\pi'$ for some action profile α and path π'. Taking a strategy profile $\boldsymbol{\sigma}$ and history h, we consider a probability measure $\mu_{h,\boldsymbol{\sigma}}$ on the σ-algebra generated by all cones of histories h' with the initial state $last(h)$. This probability measure is the unique measure such that for all histories $h' = s_0\alpha_0 \cdots s_n$, we have

$$\mu_{h,\boldsymbol{\sigma}}(\langle h' \rangle) = \prod_{k<n} T(s_k, \alpha_k)(s_{k+1}) \cdot \prod_{i \in \mathsf{Ag}} \sigma_i(\mathsf{obs}_i(h_{-1}h'_{\leq k}))(\alpha_k^i),$$

where h_{-1} is h with the final state $last(h)$ cut off (since otherwise the state $last(h)$ would appear twice successively in $hh'_{\leq k}$). Intuitively, this is the probability of encountering the history h' after h when everyone acts according to $\boldsymbol{\sigma}$. As with other similar logics, it is a standard exercise to verify that the sets of paths encountered in the semantics are all elements of the σ-algebra we consider.

Definition 7. Given a partially observable multiagent system $\mathcal{M} = (\mathcal{G}, \mathsf{Ap}, L, \{\mathsf{obs}_i\}_{i \in \mathsf{Ag}})$, a valuation $\nu : \mathsf{Var} \rightharpoonup \Sigma$, and a binding $\beta : \mathsf{Ag} \to \mathsf{Var}$, the semantics for oPSL is defined by simultaneous induction over history formulae, arithmetic terms and path formulas as follows.

For a history h, we define:

- $h, \nu \Vdash p$ iff $p \in L(last(h))$.
- $h, \nu \Vdash \neg\varphi$ iff $h, \nu \nVdash \varphi$.
- $h, \nu \Vdash \varphi \vee \psi$ iff $h, \nu \Vdash \varphi$ or $h, \nu \Vdash \psi$.
- $h, \nu \Vdash \exists x.\varphi$ iff there exists $\sigma \in \Sigma$ such that $h, \nu[x \mapsto \sigma] \Vdash \varphi$.
- $h, \nu \Vdash \tau \bowtie \tau'$ iff $\mathcal{V}_{h,\nu}(\tau) \bowtie \mathcal{V}_{h,\nu}(\tau')$ where:
  - $\mathcal{V}_{h,\nu}(c) = c$,
  - $\mathcal{V}_{h,\nu}(\tau^{-1}) = (\mathcal{V}_{h,\nu}(\tau))^{-1}$,
  - $\mathcal{V}_{h,\nu}(\tau \oplus \tau') = \mathcal{V}_{h,\nu}(\tau) \oplus \mathcal{V}_{h,\nu}(\tau')$,
  - $\mathcal{V}_{h,\nu}(\mathbf{P}_\beta(\Phi)) = \mu_{h,\nu\circ\beta}(\{\pi \mid \pi, \nu, 0 \Vdash \Phi\})$,
  - $\mathcal{V}_{h,\nu}(\mathbf{D}_{\beta,i}(\Phi)) = \frac{\mu_{h,\nu\circ\beta}(\{\pi \mid \pi,\nu,0 \Vdash \Phi\} \setminus \mathsf{obs}_i^{\sim}(\{\pi' \mid \pi',\nu,0 \nVdash \Phi\}))}{\mu_{h,\nu\circ\beta}(\{\pi \mid \pi,\nu,0 \Vdash \Phi\})}$.
- $h, \nu \Vdash_{\mathcal{M}} \odot_i\Phi$ iff for all $\pi, \pi' \in \mathsf{Paths}_{\mathcal{G}}(last(h))$ it holds that if $\pi, \nu, 0 \Vdash \Phi$ and $\pi', \nu, 0 \nVdash \Phi$, then $\mathsf{obs}_i(\pi) \neq \mathsf{obs}_i(\pi')$.

For a path π of $\mathcal{G}$ and $k \geqslant 0$, we define:

- $\pi, \nu, k \Vdash \varphi$ iff $\pi_{\leq k}, \nu \Vdash \varphi$.
- $\pi, \nu, k \Vdash \neg\Phi$ iff $\pi, \nu, k \nVdash \Phi$.
- $\pi, \nu, k \Vdash \Phi \vee \Psi$ iff $\pi, \nu, k \Vdash \Phi$ or $\pi, \nu, k \Vdash \Psi$.
- $\pi, \nu, k \Vdash \mathbf{X}\Phi$ iff $\pi, \nu, k+1 \Vdash \Phi$.
- $\pi, \nu, k \Vdash \Phi\mathbf{U}\Psi$ iff there exists $\ell \geq k$ such that $\pi, \nu, \ell \Vdash \Psi$ and for all $m \in [k, \ell)$ we have $\pi, \nu, m \Vdash \Phi$.

As in PSL, we use the convention that for all h, ν, and formulas $\tau \bowtie \tau'$ containing either a subterm ρ^{-1} such that ρ is evaluated to 0, or a subterm $\mathbf{D}_{\beta,i}(\Phi)$ for which the value in the denominator is evaluated to 0, we put $h, \nu \nVdash \tau \bowtie \tau'$ by default, to avoid issues with division by zero.

A variable x is *free* in an oPSL formula if it appears in the domain of a binding β appearing within a subformula $\mathbf{P}_\beta(\Phi)$ or term $\mathbf{D}_{\beta,i}(\Phi)$ that is not within the scope of a quantifier $\exists x$. A history formula is a *sentence* if it contains no free variables.

As usual, the until operator allows to derive the temporal modality $\mathbf{F}$ ("eventually") and $\mathbf{G}$ ("always"): $\mathbf{F}\Phi \stackrel{\text{def}}{=} \texttt{true}\,\mathbf{U}\,\Phi$ and $\mathbf{G}\Phi \stackrel{\text{def}}{=} \neg\mathbf{F}\neg\Phi$.

In the context of the oPSL semantics, if we restrict our attention to *memoryless* strategies, we can consider the semantics to be interpreted over states s instead of histories h.

Example 3. Continuing with the partially observable multi-agent system from Example 2.

Let us consider the property $\mathbf{F}stolen$. It is not observable by agent 1 given a history/state s_0. This is because as we saw before, there are two paths π and π' from s_0, one of which (going through s_1) satisfies $\mathbf{F}stolen$, and the other one (going through s_3) does not. So:

$$h, \nu \not\Vdash_{\mathcal{M}} \odot_1(\mathbf{F}stolen)$$

where $h = s_0$ and ν is any assignment. Clearly,

$$h, \nu \Vdash_{\mathcal{M}} \odot_2(\mathbf{F}stolen)$$

because agent 2 can observe all states and can always distinguish $\mathbf{F}stolen$-paths from $\neg\mathbf{F}stolen$-paths.

Let us consider the degree of observability of $\mathbf{F}stolen$ given the same history and a binding β that assigns strategies (σ_1, σ_2) to the agents. σ_1 is as follows: it requires 1 to perform send with probability 1 in *init*, wait with probability 1 in *warning*, and perform send and wait with probability 0.5 in $\circ$.

σ_2 requires agent 2 to perform copy in s_0, and perform copy and wait with probability 0.5 in s_1, s_2 and s_3.

Recall that $\mathcal{V}_{h,\nu}(\mathbf{D}_{\beta,1}(\mathbf{F}stolen))$ is

$$\frac{\mu_{h,\nu\circ\beta}([\mathbf{F}stolen] \setminus \mathsf{obs}_1^{\sim}([\neg\mathbf{F}stolen]))}{\mu_{h,\nu\circ\beta}([\mathbf{F}stolen])}$$

where we denote by $[\mathbf{F}stolen]$ the set of paths $\{\pi \mid \pi, \nu, 0 \Vdash \mathbf{F}stolen\}$, similarly for $[\neg\mathbf{F}stolen]$.

The set of paths $[\mathbf{F}stolen]$ contains paths going through s_1 and s_2, that is, paths with prefixes $s_0 \xrightarrow{(\mathsf{send},\mathsf{copy})} s_1$ and $s_0 \xrightarrow{(\mathsf{send},\mathsf{copy})} s_2$. $\mu_{h,\nu\circ\beta}$ assigns this set 0.2.

The set of paths $[\neg\mathbf{F}stolen]$ contains paths going through s_3. For agent 1, who cannot distinguish s_1 and s_3, the set of paths observationally equivalent to $[\neg\mathbf{F}stolen]$, $\mathsf{obs}_1^{\sim}([\neg\mathbf{F}stolen])$, contains all paths with prefixes $s_0 \xrightarrow{(\mathsf{send},\mathsf{copy})} s_1$ and $s_0 \xrightarrow{(\mathsf{send},\mathsf{copy})} s_3$.

The set of paths $[\mathbf{F}stolen] \setminus \mathsf{obs}_1^{\sim}([\neg\mathbf{F}stolen])$ therefore contains paths with prefixes $s_0 \xrightarrow{(\mathsf{send},\mathsf{copy})} s_2$. $\mu_{h,\nu\circ\beta}$ assigns this set 0.1.

Hence $\mathcal{V}_{h,\nu}(\mathbf{D}_{\beta,1}(\mathbf{F}stolen)) = \frac{0.1}{0.2} = 0.5$.

Example 4. Lastly, we consider a few more general examples. Let $\mathbf{x} = (x_1, \ldots, x_n)$ be a tuple of n strategy variables.

1) Asking whether, against all possible strategies of the other agents $2, \ldots, n$, agent 1 has a strategy x_1 such that the path formula Ψ is observable to her can be expressed as:
$$\exists x_1.\forall x_2.\forall x_3 \ldots \forall x_n.\, (\odot_i(\Psi))$$
2) Asking whether any other agent $i \geq 2$ has a strategy x_i, such that for all possible strategies of agent 1, the observability degree of the path formula Ψ from 1's view is ≤ 0.1, provided that ψ holds with probability ≥ 0.9, can be expressed as:
$$\forall x_1.\exists x_2 \ldots \exists x_n.\, (\mathbf{P_x}(\Psi) \geq 0.9 \rightarrow \mathbf{D}_{\mathbf{x},1}(\Psi) \leq 0.1)$$

5 Verification of oPSL

Since oPSL extends PSL, its model checking problem is undecidable, even when restricted to partially observable multi-agent systems with only a single agent (Aminof et al. 2019; Brázdil et al. 2006).

However, if we restrict our attention to memoryless strategies, the model checking problem becomes decidable.

Theorem 1. *Model checking* oPSL *sentences when the semantics is interpreted only with memoryless strategies, is decidable.*

We describe the model checking procedure exhibiting decidability in the next section.

5.1 Model Checking Algorithm

In this section, we outline the model checking algorithm for oPSL when considering memoryless strategies. The complete procedure can be found in (Mu et al. 2024).The basis of the procedure lies in that for PSL (Aminof et al. 2019), but the novel operators of oPSL present highly non-trivial challenges, as we will see.

Given a partially observable multi-agent system $\mathcal{M}$, we will define first-order logic formulas $\mathfrak{a}_{\varphi,s}$ by induction over state formulas φ and states s, written in the signature of real arithmetic. The construction will be such that for all oPSL *sentences* φ and states s, we have that $\mathfrak{a}_{\varphi,s}$ holds in the theory of real arithmetic if and only if $s \Vdash \varphi$. Using the decidability of this theory, we then get a model checking algorithm.

Throughout we will often write $\top_{cond}$ for some metalogical condition $cond$, defined as $\top_{cond} := \top$ if $cond$ is true, and $\top_{cond} := \bot$ otherwise. We will denote the equality symbol inside of the real arithmetic formula by $\approx$ to avoid any confusion.

Atoms and Booleans Let $\mathfrak{a}_{p,s} := \top_{p \in L(s)}$. We put $\mathfrak{a}_{\varphi\wedge\psi,s} := \mathfrak{a}_{\varphi,s} \wedge \mathfrak{a}_{\psi,s}$ and $\mathfrak{a}_{\neg\varphi,s} := \neg\mathfrak{a}_{\varphi,s}$.

Existential quantifier Given the formula $\exists x.\varphi$, we introduce variables $r_{x,\theta,a}$, intuitively encoding the probability that the strategy x performs action a upon observing $\theta \in \Theta^{\mathsf{s}}$. Let $\mathfrak{a}_{\exists x.\varphi,s} := \exists (r_{x,\theta,a})_{\theta\in\Theta^{\mathsf{s}}, a\in\mathsf{Act}}[\mathsf{Dist}_x \wedge \mathfrak{a}_{\varphi,s}]$, where

$$\mathsf{Dist}_x := [\bigwedge_{\theta\in\Theta^{\mathsf{s}}, a\in\mathsf{Act}} r_{x,\theta,a} \geqslant 0] \wedge [\bigwedge_{\theta\in\Theta^{\mathsf{s}}} \sum_{a\in\mathsf{Act}} r_{x,\theta,a} \approx 1].$$

The formula Dist_x encodes that the variables $r_{x,\theta,a}$ give probability distributions for each θ.

Full observability formulas Given the formula $\odot_k \Phi$, we write a formula that expresses that it is **not** the case that $s \nVdash \odot_k \Phi$. The reason for this double negation is that we can relatively easily express $s \nVdash \odot_k \Phi$: this holds iff there exist paths $\pi, \pi' \in \mathsf{Paths}_{\mathcal{M}}(s)$ such that $\pi \Vdash \Phi$ and $\pi' \Vdash \neg\Phi$, but $\mathsf{obs}_k(\pi) = \mathsf{obs}_k(\pi')$. To check this, we will describe a rooted directed graph $\mathcal{G}_{\odot_k \Phi}$ inside of the real arithmetic formula, such that infinite walks through the graph (starting from the root) correspond precisely to pairs $\pi, \pi' \in \mathsf{Paths}_{\mathcal{M}}(s)$. Furthermore, we will identify a set $F_{\odot_k \Phi}$ of vertices in the graph, such that infinite walks (π, π') through the graph reach $F_{\odot_k \Phi}$ *infinitely often* if and only if $\pi \Vdash \Phi$, $\pi' \Vdash \neg\Phi$, and $\mathsf{obs}_k(\pi) = \mathsf{obs}_k(\pi')$. Given this, all we then need to express that $s \nVdash \odot_k \Phi$ is that there exists an infinite walk through the graph that hits $F_{\odot_k \Phi}$ infinitely often.

Note that Φ can be considered to be an LTL formula over atomic propositions $W = 2^{\mathsf{Max}(\Phi)}$, where $\mathsf{Max}(\Phi)$ is the set of maximal state subformulas of Φ. So we can construct nondeterministic Büchi automata $A_\Phi = (Q_\Phi, D_\Phi, q^*_\Phi, F_\Phi)$ and $A_{\neg\Phi} = (Q_{\neg\Phi}, D_{\neg\Phi}, q^*_{\neg\Phi}, F_{\neg\Phi})$ over alphabet W, recognizing those $\mathbf{w} \in W^\omega$ such that $\mathbf{w} \Vdash_{\mathsf{LTL}} \Phi$ and $\mathbf{w} \Vdash_{\mathsf{LTL}} \neg\Phi$, respectively. We finally define $\mathfrak{a}_{\odot_k \Phi, s}$ by stating that we cannot reach vertices in $F_{\odot_k \Phi}$ from the root that go back to themselves with non-empty paths, which is an efficient way of stating no infinite walk through the graph hits $F_{\odot_k \Phi}$ infinitely often.

The detailed definitions of $\mathfrak{a}_{\odot_k \Phi, s}$ are omitted here due to lack of space and can be found in (Mu et al. 2024).

Inequalities Given the formula $\tau_1 \leqslant \tau_2$, we introduce variables r_τ for all arithmetic subterms τ appearing in τ_1 and τ_2 (we denote the set of all such subterms by $Sub(\tau_1, \tau_2)$), which will intuitively hold the values the terms τ will take at the state s under consideration, and given the strategies assigned to the strategic variables x. We define $\mathfrak{a}_{\tau_1 \leqslant \tau_2, s}$ as

$$\exists (r_\tau)_{\tau \in Sub(\tau_1, \tau_2)} [Eqn_{\tau_1, s} \wedge Eqn_{\tau_2, s} \wedge r_{\tau_1} \leqslant r_{\tau_2}],$$

where the formula $Eqn_{\tau_i, s}$ encodes that the variable r_{τ_i} indeed holds the value of τ_i in s under the current valuation. We will now define these formulas.

Arithmetic terms We let:

$$Eqn_{c,s} \coloneqq [r_c \approx c],$$

$$Eqn_{\tau^{-1},s} \coloneqq Eqn_{\tau,s} \wedge [r_\tau \times r_{\tau^{-1}} \approx 1],$$

$$Eqn_{\tau_1 + \tau_2, s} \coloneqq Eqn_{\tau_1, s} \wedge Eqn_{\tau_2, s} \wedge [r_{\tau_1 + \tau_2} \approx r_{\tau_1} + r_{\tau_2}],$$

$$Eqn_{\tau_1 \times \tau_2, s} \coloneqq Eqn_{\tau_1, s} \wedge Eqn_{\tau_2, s} \wedge [r_{\tau_1 \times \tau_2} \approx r_{\tau_1} \times r_{\tau_2}],$$

Probabilistic terms Given the formula $\mathbf{P}_\beta(\Phi)$, we 'internalize' the description of the model checking procedure of PCTL* inside of our real arithmetic formula.

Considering Φ to be an LTL formula over $W = 2^{\mathsf{Max}(\Phi)}$, as in the construction for the full observability formula, we construct a deterministic Rabin automaton $A_\Phi = (Q, q^*, \delta, \mathsf{Acc})$ over alphabet W accepting those $\mathbf{w} \in W^\omega$ such that $\mathbf{w} \Vdash_{\mathsf{LTL}} \Phi$. Recall that a (memoryless) strategy profile $\boldsymbol{\sigma}$ (as e.g. given by a binding β and valuation ν) turns our model $\mathcal{M}$ into a Markov chain $M_{\boldsymbol{\sigma}} = (S, t_{\boldsymbol{\sigma}})$, with $t_{\boldsymbol{\sigma}} \colon S \to \mathsf{Dist}(S)$ defined by putting

$$t_{\boldsymbol{\sigma}}(s_1)(s_2) = \sum_{\alpha \in \mathsf{Act}^{\mathsf{Ag}}} T(s_1, \alpha)(s_2) \prod_{i \in \mathsf{Ag}} \sigma_i(\mathsf{obs}_i(s_1))(\alpha_i).$$

Inside of the real arithmetic formula, we wish to compute the probability that a random walk through this Markov chain starting at s satisfies Φ. To do so, we will internalize the product Markov chain construction as is used in PCTL* model checking. The product Markov chain $M_{\boldsymbol{\sigma}} \otimes A$ has the same dynamics as $M_{\boldsymbol{\sigma}}$, while also providing as input to the automaton A the formulas in W which are true at each state. It follows from this construction and the defining property of the DRA A, that the probability we are after is precisely the probability that a random walk through $M_{\boldsymbol{\sigma}} \otimes A$ from (s, q^*) is 'accepted' by A, in the sense that the walk's sequence of automaton states $q_0 q_1 \cdots \in Q^\omega$ satisfies the Rabin acceptance condition Acc. Following the method in PCTL* model checking, the problem can be reduced to computing the *reachability* probability of certain *accepting* terminal strongly connected components, which can be done by solving a system of linear equations. This process can be (efficiently) encoded in real arithmetic, details can be found in (Mu et al. 2024). We only sketch the key formulas here.

We write a formula $Goal_v$ that efficiently expresses that a state v is part of an accepting terminal strongly connected component, as well as a formula Sol, which expresses the solution to the system of linear equations. We can then write out $Eqn_{\mathbf{P}_\beta(\Phi), s}$:

$$\begin{aligned} Eqn_{\mathbf{P}_\beta(\Phi), s} \quad \coloneqq \quad & \exists (r_v^w, r_v^{\mathrm{sol}})_{v, w \in S \times Q}. \\ & \big[Prod_\beta \wedge Sol \wedge r_{\mathbf{P}_\beta(\Phi)} \approx r^{\mathrm{sol}}_{s, q^*}\big], \end{aligned}$$

with $Prod_\beta$ expressing the dynamics of the product Markov chain.

Degree of observability terms Formulas of the form $\mathbf{D}_{\beta,i}(\Phi)$ are significantly more complicated than the previous terms. We will end up defining $Eqn_{\mathbf{D}_{\beta,i}(\Phi), s}$ as:

$$\begin{aligned} Eqn_{\mathbf{D}_{\beta,i}(\Phi), s} \coloneqq\ & \exists r_{\mathbf{P}_\beta(\Phi)} \exists r_{(\mathbf{P}_\beta(\Phi))^{-1}} \exists r_{\mathsf{obs}_i \Phi}. \\ & \big[Eqn_{\mathbf{P}_\beta(\Phi), s} \wedge Eqn_{\mathsf{obs}_i \Phi, s} \\ & \wedge r_{\mathbf{P}_\beta(\Phi)} \not\approx 0 \to [r_{(\mathbf{P}_\beta(\Phi))^{-1}} \times r_{\mathbf{P}_\beta(\Phi)} \approx 1 \\ & \wedge r_{\mathbf{D}_{\beta,i}(\Phi)} \approx r_{\mathsf{obs}_i \Phi} \times r_{(\mathbf{P}_\beta(\Phi))^{-1}}] \\ & \wedge r_{\mathbf{P}_\beta(\Phi)} \approx 0 \to r_{\mathbf{D}_{\beta,i}(\Phi)} \approx 1\big]. \end{aligned}$$

The idea is that we will compute two values: (i) the probability of a random path satisfying Φ (represented by the variable $r_{\mathbf{P}_\beta(\Phi)}$), and (ii) the probability of a random path satisfying Φ whilst *not* being observationally equivalent (for agent i) to a $\neg\Phi$-path (represented by the variable $r_{\mathsf{obs}_i \Phi}$). The degree of observability is then obtained by dividing (ii) by (i) (with some care to deal with the situation in which the denominator is 0). As we already know how to compute (i) from the inductive step for probabilistic terms shown before, we will only focus on computing (ii). Again, due to space constraints, the detailed construction of $Eqn_{\mathsf{obs}_i \Phi, s}$ can be

found in (Mu et al. 2024). The idea is that we will, often internally in the real arithmetic formula, construct two series of automata, which we will then combine afterwards.
First:

(A.1) Compute a deterministic Streett automaton[1] recognizing Φ outside the real arithmetic formula.

(A.2) From that DSA, construct a DSA A_Φ over the alphabet $\mathsf{Act}^{\mathsf{Ag}} \times S$ inside the real arithmetic formula, such that A_Φ accepts precisely those $\mathbf{w}$ such that $s\mathbf{w} \in \mathsf{Paths}_{\mathcal{G}}(s)$ and $s\mathbf{w} \Vdash \Phi$. In other words, the DSA accepts the paths from s that satisfy Φ.

Second:

(B.1) Compute an NBA recognising $\neg\Phi$ outside the real arithmetic formula.

(B.2) From that NBA, build an NBA $A^{\mathsf{NBA}}_{\mathsf{obs}_i \neg\Phi}$ over alphabet $\mathsf{Act}^{\mathsf{Ag}} \times S$ inside the real arithmetic formula, such that $A^{\mathsf{NBA}}_{\mathsf{obs}_i \neg\Phi}$ accepts precisely those $\mathbf{w}$ such that $s\mathbf{w} \in \mathsf{Paths}_{\mathcal{M}}(s)$ and for which there exists $\pi \in \mathsf{Paths}_{\mathcal{M}}(s)$ with $\pi \Vdash \neg\Phi$ and $\mathsf{obs}_i(s\mathbf{w}) = \mathsf{obs}_i(\pi)$. In other words, the NBA accepts the paths from s that are (for agent i) observationally equivalent to paths satisfying $\neg\Phi$.

(B.3) Determinise $A^{\mathsf{NBA}}_{\mathsf{obs}_i \neg\Phi}$ into an equivalent DRA $A^{\mathsf{DRA}}_{\mathsf{obs}_i \neg\Phi}$ using Safra's construction encoded into the real arithmetic formula.

(B.4) Negate $A^{\mathsf{NBA}}_{\mathsf{obs}_i \neg\Phi}$ inside of the real arithmetic formula, obtaining a DSA $A^{\mathsf{DSA}}_{\mathsf{obs}_i \Phi}$ recognizing precisely those sequences $\mathbf{w} \in (\mathsf{Act}^{\mathsf{Ag}} \times S)^\omega$ such that if $s\mathbf{w} \in \mathsf{Paths}_{\mathcal{M}}(s)$, then $s\mathbf{w}$ is not observationally equivalent (for i) to any path satisfying $\neg\Phi$.

We will then, finally, take the following steps:

(C.1) Build the product DSA $A = A_\Phi \otimes A^{\mathsf{DSA}}_{\mathsf{obs}_i \Phi}$ inside of the real arithmetic formula, which recognizes precisely those $\mathbf{w}$ such that $s\mathbf{w} \in \mathsf{Paths}_{\mathcal{M}}(s)$, $s\mathbf{w} \Vdash \Phi$, and $s\mathbf{w}$ is not observationally equivalent to any path satisfying $\neg\Phi$.

(C.2) Construct the product Markov chain $M_{\boldsymbol{\sigma}} \otimes A$ again inside the real arithmetic formula, and express the process of computing the probability of a random walk in it being accepting.

The idea is that we wish to end up with some deterministic automaton recognizing the set of Φ-paths that are not observationally equivalent to a $\neg\Phi$-path. This needs to be deterministic to be able to proceed with the product Markov chain construction. The A-construction builds a deterministic automaton that accepts precisely Φ-paths, while the B-construction builds a deterministic automaton that accepts sequences that are not equivalent to $\neg\Phi$-paths. By taking the product of both constructions, we end up with an automaton that recognizes the language we are after.

[1]A DSA over alphabet A is a tuple $(Q, \delta, q^*, \mathsf{Acc})$, defined identically to a DRA. The difference now is that the automaton accepts $\mathbf{w} \in A^\omega$ iff for *all* $(E, F) \in \mathsf{Acc}$, the automaton's run following $\mathbf{w}$ reaches all states in E finitely often, **or** reaches some state in F infinitely often. In other words, the acceptance condition is dual to a Rabin one.

Note that many parts of the construction happen internally in the formula. This happens anytime a step relies on evaluating oPSL formulas, as such evaluation is only possible relative to a valuation, which we are quantifying over in real arithmetic, and therefore do not have access to.

5.2 Complexity

Theorem 2. *Model checking* oPSL *can be done in space triple exponential with respect to the* oPSL *sentence, and double exponential with respect to the partially observable multi-agent system.*

Proof sketch. The size of the real arithmetic formula is double exponential and the number of quantifiers is easily verified to be single exponential w.r.t. the sentence size. While the dependence of our real arithmetic formula's size on the oPSL sentence is the same as that of PSL, we do get another exponential blowup with respect to the size of the system. This extra blowup is caused by the larger number of quantifiers in our real arithmetic formula: we require at least one quantifier per Safra tree appearing in the B-construction in order to express the dynamics of the product Markov chain.

Since the validity of real arithmetic is decidable in space exponential w.r.t. the number of quantifiers and logarithmic w.r.t. the size of the quantifier-free part of the formula (Renegar 1992; Basu 2014), we therefore arrive at an overall space complexity that is triple exponential w.r.t. the oPSL sentence, and double exponential w.r.t. the system. □

6 Conclusions and Future Work

This paper provides a framework for expressing and reasoning about observability within MAS, along with the capability to quantify the degree of observability under specified strategies. The framework contributes to formal analysis and verification in multi-agent systems, especially for properties relating to information security, privacy, and trustworthiness. In particular, oPSL enables a rigorous analysis of agent observability and information transparency, allowing the assessment of how much information about system behaviours is available to different agents. This is crucial for identifying potential vulnerabilities and understanding the security implications of information exposure.

In considering future directions, there are several areas that would be interesting to explore. First, the interconnections and synergies between oPSL and other logics, such as epistemic logics, would augment the framework's expressive capabilities. Another possible line of work involves extending oPSL to include additional aspects of multi-agent systems, such as hierarchical structures or more complex forms of actions. Adapting oPSL to navigate dynamic and evolving environments, where agents' strategies may undergo temporal transformations, presents another area for investigation, as does investigating the application of the framework in the domains of AI safety and responsibility. Finally, incorporating game-theoretic approaches may allow balancing between utility and security.

References

Ågotnes, T. 2006. Action and Knowledge in Alternating-Time Temporal Logic. *Synth.*, 149(2): 375–407.

Alvim, M. S.; Chatzikokolakis, K.; McIver, A.; Morgan, C.; Palamidessi, C.; and Smith, G. 2020. *The Science of Quantitative Information Flow*. Information Security and Cryptography. Springer.

Aminof, B.; Kwiatkowska, M.; Maubert, B.; Murano, A.; and Rubin, S. 2019. Probabilistic Strategy Logic. In *IJCAI*, 32–38.

Ballot, G.; Malvone, V.; Leneutre, J.; and Laarouchi, Y. 2024. Strategic Reasoning under Capacity-constrained Agents. In Dastani, M.; Sichman, J. S.; Alechina, N.; and Dignum, V., eds., *Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2024*, 123–131. International Foundation for Autonomous Agents and Multiagent Systems / ACM.

Basu, S. 2014. Algorithms in Real Algebraic Geometry: A Survey. *CoRR*, abs/1409.1534.

Belardinelli, F.; Lomuscio, A.; Murano, A.; and Rubin, S. 2017. Verification of Multi-agent Systems with Imperfect Information and Public Actions. In *AAMAS*, 1268–1276. ACM Press.

Belardinelli, F.; Lomuscio, A.; Murano, A.; and Rubin, S. 2020. Verification of multi-agent systems with public actions against strategy logic. *Artificial Intelligence*, 285: 103302.

Berthon, R.; Maubert, B.; Murano, A.; Rubin, S.; and Vardi, M. Y. 2021. Strategy Logic with Imperfect Information. *ACM Trans. Comput. Log.*, 22(1): 5:1–5:51.

Beutner, R.; and Finkbeiner, B. 2023. AutoHyper: Explicit-State Model Checking for HyperLTL. In Sankaranarayanan, S.; and Sharygina, N., eds., *Tools and Algorithms for the Construction and Analysis of Systems - 29th International Conference, TACAS 2023, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2022, Paris, France, April 22-27, 2023, Proceedings, Part I*, volume 13993 of *Lecture Notes in Computer Science*, 145–163. Springer.

Brázdil, T.; Brozek, V.; Forejt, V.; and Kucera, A. 2006. Stochastic Games with Branching-Time Winning Objectives. In *LICS*, 349–358. IEEE Computer Society.

Bryans, J. W.; Koutny, M.; Mazaré, L.; and Ryan, P. Y. A. 2005. Opacity Generalised to Transition Systems. In Dimitrakos, T.; Martinelli, F.; Ryan, P. Y. A.; and Schneider, S. A., eds., *Formal Aspects in Security and Trust, Third International Workshop, FAST 2005, Newcastle upon Tyne, UK, July 18-19, 2005, Revised Selected Papers*, volume 3866 of *Lecture Notes in Computer Science*, 81–95. Springer.

Chatterjee, K.; Henzinger, T. A.; and Piterman, N. 2010. Strategy logic. *Inf. Comput.*, 208(6): 677–693.

Chen, T.; and Lu, J. 2007. Probabilistic Alternating-time Temporal Logic and Model Checking Algorithm. In Lei, J., ed., *Fourth International Conference on Fuzzy Systems and Knowledge Discovery, FSKD 2007, 24-27 August 2007, Haikou, Hainan, China, Proceedings, Volume 2*, 35–39. IEEE Computer Society.

Ferrando, A.; and Malvone, V. 2023. Towards the Verification of Strategic Properties in Multi-Agent Systems with Imperfect Information. In *AAMAS*, 793–801. ACM.

Goranko, V.; and Jamroga, W. 2004. Comparing Semantics of Logics for Multi-Agent Systems. *Synth.*, 139(2): 241–280.

Huang, X.; Su, K.; and Zhang, C. 2012. Probabilistic Alternating-Time Temporal Logic of Incomplete Information and Synchronous Perfect Recall. In *AAAI*. AAAI Press.

Kwiatkowska, M.; Norman, G.; Parker, D.; and Santos, G. 2019. Equilibria-Based Probabilistic Model Checking for Concurrent Stochastic Games. In *FM*, volume 11800 of *Lecture Notes in Computer Science*, 298–315. Springer-Verlag.

Lomuscio, A.; Qu, H.; and Raimondi, F. 2017. MCMAS: an open-source model checker for the verification of multi-agent systems. *International Journal on Software Tools for Technology Transfer*, 19(1): 9–30.

Malvone, V.; Murano, A.; and Sorrentino, L. 2017. Hiding Actions in Multi-Player Games. In Larson, K.; Winikoff, M.; Das, S.; and Durfee, E. H., eds., *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems, AAMAS 2017*, 1205–1213. ACM.

Mu, C.; Motamed, N.; Alechina, N.; and Logan, B. 2024. Probabilistic Strategy Logic with Degrees of Observability. arXiv:2412.15135.

Mu, C.; and Pang, J. 2023. On Observability Analysis in Multiagent Systems. In *ECAI*, volume 372 of *Frontiers in Artificial Intelligence and Applications*, 1755–1762.

Renegar, J. 1992. On the Computational Complexity and Geometry of the First-Order Theory of the Reals, Part I: Introduction. Preliminaries. The Geometry of Semi-Algebraic Sets. The Decision Problem for the Existential Theory of the Reals. *J. Symb. Comput.*, 13(3): 255–300.

Schobbens, P. 2003. Alternating-time logic with imperfect recall. In van der Hoek, W.; Lomuscio, A.; de Vink, E. P.; and Wooldridge, M. J., eds., *1st International Workshop on Logic and Communication in Multi-Agent Systems, LCMAS 2003*, volume 85 of *Electronic Notes in Theoretical Computer Science*, 82–93. Elsevier.