On Simple Expectations and Observations of Intelligent Agents: A Complexity Study

Sourav Chakraborty[1] , **Avijeet Ghosh**[1] , **Sujata Ghosh**[2] , **François Schwarzentruber**[3]
[1]Indian Statistical Institute, Kolkata
[2]Indian Statistical Institute, Chennai
[3]Univ Rennes, IRISA, France
{sourav, avijeet_r}@isical.ac.in, sujata@isichennai.res.in, francois.schwarzentruber@ens-rennes.fr

Abstract

Public observation logic (POL) reasons about agent expectations and agent observations in various real world situations. The expectations of agents take shape based on certain protocols about the world around and they remove those possible scenarios where their expectations and observations do not match. This in turn influences the epistemic reasoning of these agents. In this work, we study the computational complexity of the satisfaction problems of various fragments of POL. In the process, we also highlight the inevitable link that these fragments have with the well-studied Public announcement logic.

1 Introduction

Reasoning about knowledge among multiple agents plays an important role in studying real-world problems in a distributed setting, e.g., in communicating processes, protocols, strategies and games. *Multi-agent epistemic logic (*EL*)* (Fagin et al. 1995) and its dynamic extensions, popularly known as *dynamic epistemic logics (*DEL*)* (van Ditmarsch, van der Hoek, and Kooi 2008) are well-known logical systems to specify and reason about such dynamic interactions of knowledge. Traditionally, agents' knowledge is about facts and EL/DEL mostly deals with this phenomenon of 'knowing that'. More recently, the notions of 'knowing whether', 'knowing why' and 'knowing how' have also been investigated from a formal viewpoint (Wang 2018).

These agents also have expectations about the world around them, and they reason based on what they observe around them, and such observations may or may not match the expectations they have about their surroundings. Following (Wang 2011), such perspectives on agent reasoning were taken up by (van Ditmarsch et al. 2014) and studied formally in the form of *Public observation logic (*POL*)*. We present below a situation that POL is adept at modelling. The example is in the lines of the one considered in (Chakraborty et al. 2022):

Example 1. *Let us consider a robotic vacuum cleaner ($vbot$) that is moving on a floor represented as a 7×7 grid (see Figure 1). On the top right of the floor, there is a debris-disposal area, and on the bottom left, there is a power source to recharge. Two children Alice and Bob are awed by this new robotic cleaner. They are watching it move and trying to guess which direction it is moving. The system is adaptive, thus the global behaviour is not hard-coded but learned. We suppose that $vbot$ moves on a grid and the children may observe one of the four directions: right (►), left (◄), up(▲) or down(▼), and of course, combinations of them. Note that, for example, observing ◄ means that the bot moves one step left. Let Alice be aware of a glitch in the bot. Then her expectations regarding the $vbot$'s movements include the following possibilities:*

1. *The bot may go up or right for debris-disposal, but may make an erroneous move, that is, a down or a left move.*
2. *The bot may go towards power source without error.*

The only difference between Bob's expectation and that of Alice is that Bob does not consider the bot to make an error while moving towards debris-disposal since he is unaware of the glitch.

Suppose the $vbot$ is indeed moving towards power from the center of the grid. Hence if the bot makes one left move, ◄, Bob would know that the bot is moving towards power whereas Alice would still consider moving towards debris-disposal a possibility.

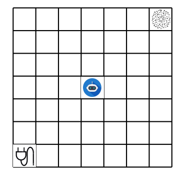

Figure 1: A robotic vacuum cleaner on the floor (in the middle of the grid). The power source is at bottom left, whereas the debris-disposal area is at top right.

The example concerns certain rules that we follow in our daily life, they deal with situations where agents expect certain observations at certain states based on some pre-defined *protocols*, viz. the bot mechanism in the example given above. They get to know about the actual situation by observing certain actions which agree with their expectations corresponding to that situation. POL does not deal with the protocols themselves, but the effect those protocols have in our understanding of the world around us in terms of our expectations and observations. In (Chakraborty et al. 2022) we have investigated the computational complexity of the model-checking problem of different fragments of POL, and

	Single-agent	Multi-agent
Word POL^-	NP-Complete	PSPACE-Complete
POL^-	PSPACE-Hard	NEXPTIME-Complete

Figure 2: Complexity results of satisfiability of various fragments of POL^-.

in this paper, we will deal with the computational complexity of the satisfaction problem of various proper fragments of POL (cf. Figure 2). We will show how certain simple fragments of POL give rise to high complexity with respect to their computational behaviour.

To prove the complexity results of some fragment(s) of POL we use a translation to Public announcement logic (PAL) (Plaza 2007), whereas, for other fragment(s), a tableau method is utilized where the tableau rules provide a mix of modal logic reasoning and computations of language theory residuals.

Outline. In Section 2, we recall the relevant definitions of POL. In Section 3, we describe an application of the satisfiability problem of POL^-. In Section 4 we present a NEXPTIME algorithm for POL^- using the tableau method. In Section 5, we prove that POL^- is in NEXPTIME-Hard. In section 6, we present the complexity results for various fragments of POL^-. Section 7 discusses related work, and Section 8 concludes the paper.

2 Background

In this section, we provide a brief overview of a fragment of public observation logic (POL) (van Ditmarsch et al. 2014), which we term as POL^-.

2.1 A Fragment of POL(POL^-)

Let Agt be a finite set of agents, $\mathcal{P}$ be a countable set of propositions describing the facts about the state and $\mathbf{\Sigma}$ be a finite set of actions.

An *observation* is a finite string of actions. In the vacuum bot example, an observation may be $\blacktriangleleft\ \blacktriangledown\ \blacktriangleright\ \blacktriangle$ and similar others. An agent may expect different potential observations to happen at a given state, but to model human/agent expectations, such expectations are described in a finitary way by introducing the *observation expressions* (as star-free regular expressions over $\mathbf{\Sigma}$):

Definition 1 (Observation expressions)**.** *Given a finite set of action symbols $\mathbf{\Sigma}$, the language $\mathcal{L}_{obs}$ of* observation expressions *is defined by the following BNF:*

$$\pi \quad ::= \quad \emptyset \mid \varepsilon \mid a \mid \pi \cdot \pi \mid \pi + \pi$$

where $\emptyset$ denotes the empty set of observations, the constant ε represents the empty string, and $a \in \mathbf{\Sigma}$.

In the bot example, the observation expression $(\blacktriangleleft \cdot \blacktriangledown + \blacktriangleright \cdot \blacktriangle)$ models the expectation of the bot's movement in either way, towards the power source or the debris-disposal area, whereas $(\blacktriangleleft)^3 \cdot (\blacktriangledown)^3$ models the expectation of moving towards the power source.

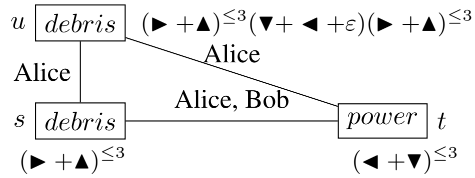

Figure 3: Model describing the initial knowledge of the two agents Alice and Bob about the expectation of the $vbot$.

The size of an observation expression π is denoted by $|\pi|$. The semantics for the observation expressions are given by *sets of observations* (strings over $\mathbf{\Sigma}$), similar to those for regular expressions. Given an observation expression π, its *set of observations* is denoted by $\mathcal{L}(\pi)$. For example, $\mathcal{L}(\blacktriangleright) = \{\blacktriangleright\}$, and $\mathcal{L}(\blacktriangleleft \cdot \blacktriangledown + \blacktriangleright \cdot \blacktriangle) = \{\blacktriangleleft\blacktriangledown, \blacktriangleright\blacktriangle\}$. The (star-free) regular language $\pi\backslash w$ is the set of words given by $\{v \in \mathbf{\Sigma}^* \mid wv \in \mathcal{L}(\pi)\}$. The language $Pre(\pi)$ is the set of prefixes of words in $\mathcal{L}(\pi)$, that is, $w \in Pre(\pi)$ iff $\exists v \in \mathbf{\Sigma}^*$ such that $wv \in \mathcal{L}(\pi)$ (namely, $\mathcal{L}(\pi\backslash w) \neq \emptyset$).

Example 2. $(\blacktriangleleft \cdot \blacktriangledown)\backslash \blacktriangleleft = (\blacktriangleleft \cdot \blacktriangledown + \blacktriangleright \cdot \blacktriangle)\backslash \blacktriangleleft = \blacktriangledown$, *and* $Pre(\blacktriangleleft \cdot \blacktriangledown + \blacktriangleright \cdot \blacktriangle) = \{\varepsilon, \blacktriangleleft, \blacktriangleleft\blacktriangledown, \blacktriangleright, \blacktriangleright\blacktriangle\}$.

We now present a modified version of *epistemic expectation models* from (van Ditmarsch et al. 2014) that capture the expected observations of agents. They can be seen as epistemic models together with, for each state, a set of potential or expected observations. Recall that an epistemic model is a tuple $\langle S, \sim, V\rangle$ where S is a non-empty set of states, $\sim$ assigns to each agent in Agt an equivalence relation $\sim_i \subseteq S \times S$, and $V : S \to 2^{\mathcal{P}}$ is a valuation function.

Definition 2 (Epistemic expectation model with finite observations)**.** *An* epistemic expectation model with finite observations *$\mathcal{M}$ is a quadruple $\langle S, \sim, V, Exp\rangle$, where $\langle S, \sim, V\rangle$ is an epistemic model (the* epistemic skeleton *of $\mathcal{M}$) and $Exp : S \to \mathcal{L}_{obs}$ is an expected observation function assigning to each state an observation expression π such that $\mathcal{L}(\pi) \neq \emptyset$ (finite non-empty set of finite sequences of observations). A pointed epistemic expectation model with finite observations is a pair $(\mathcal{M}, s)$ where $\mathcal{M} = \langle S, \sim, V, Exp\rangle$ is an epistemic expectation model with finite observations and $s \in S$. In what follows we will use the 'epistemic expectation model' to denote the 'epistemic expectation model with finite observations'.*

Intuitively, Exp assigns to each state a set of potential or expected observations. We now provide the model definition of the example mentioned in the introduction (cf. Figure 3) For the sake of brevity, we do not draw the reflexive arrows. If the $vbot$ moves one step left, $\blacktriangleleft$, then while Alice still considers moving to the debris-disposal area a possibility, Bob does not consider that possibility at all, as described by Example 1, and depicted by the edge in Figure 3 between the states u and t, annotated by Alice and not Bob.

The logic POL was introduced to reason about agent knowledge via the matching of observations and expectations, and as we mentioned earlier, the difference between POL and POL^- is just a technical one. The main idea expressed in these logics is the following: While observing an action, people would tend to delete some impossible scenarios where they would not expect that observation to happen. For this purpose, the update of epistemic expectation models with respect to some observation $w \in \mathbf{\Sigma}^*$ is provided below.

Definition 3 (Update by observation)**.** *Let w be an observation over $\mathbf{\Sigma}$ and let $\mathcal{M} = \langle S, \sim, V, Exp\rangle$ be an epistemic expectation model. The updated model $\mathcal{M}|_w = \langle S', \sim', V', Exp'\rangle$ is defined by: $S' = \{s \mid \mathcal{L}(Exp(s)\backslash w) \neq \emptyset\}$, $\sim'_i = \sim_i|_{S'\times S'}$, $V' = V|_{S'}$, and $Exp'(s) = Exp(s)\backslash w$.*

The main idea of the updated model is to delete the states where the observation w could not have happened. To reason about agent expectations and observations, the language for POL^- is provided below.

Definition 4 (POL^- syntax)**.** *Given a countable set of* propositional variables $\mathcal{P}$*, a finite sets of* actions $\mathbf{\Sigma}$*, and a finite set of* agents Agt*, the* formulas φ *of* POL^- *are given by:*

$$\varphi \quad ::= \quad \top \mid p \mid \neg\varphi \mid \varphi \wedge \varphi \mid K_i\varphi \mid [\pi]\varphi$$

where $p \in \mathcal{P}$, $i \in Agt$, and $\pi \in \mathcal{L}_{obs}$.

Intuitively, $K_i\varphi$ says that 'agent i knows φ and $[\pi]\varphi$ says that 'after any observation in π, φ holds'. The other propositional connectives are defined in the usual manner. We also define $\langle\pi\rangle\varphi$ as $\neg[\pi]\neg\varphi$ and $\hat{K}_i\varphi$ as $\neg K_i\neg\varphi$. Typically, $\langle\pi\rangle\varphi$ says that 'there exists an observation in π such that φ holds'. Formula $\hat{K}_i\varphi$ says that 'agent i imagines a state in which φ holds'.

The logic POL^- is the Star-Free **fragment of** POL, that is, it is the set of formulas in which the π's do not contain any Kleene star $*$. A more restricted version is the Word **fragment of** POL^-, where π's are words, that is, observation expressions without $+$ operators. We consider both the **single-agent word fragment of** POL^-, and **multi-agent word fragment of** POL^-. Furthermore, we consider **single-agent** POL^-, and **multi-agent** POL^- (full POL^-).

Definition 5 (Truth definition for POL^-)**.** *Given an epistemic expectation model $\mathcal{M} = (S, \sim, V, Exp)$, a state $s \in S$, and a POL^--formula φ, the truth of φ at s, denoted by $\mathcal{M}, s \vDash \varphi$, is defined by induction on φ as follows:*

$$\begin{array}{rcl}
\mathcal{M}, s \vDash p & \Leftrightarrow & p \in V(s) \\
\mathcal{M}, s \vDash \neg\varphi & \Leftrightarrow & \mathcal{M}, s \nvDash \varphi \\
\mathcal{M}, s \vDash \varphi \wedge \psi & \Leftrightarrow & \mathcal{M}, s \vDash \varphi \textit{ and } \mathcal{M}, s \vDash \psi \\
\mathcal{M}, s \vDash K_i\varphi & \Leftrightarrow & \textit{for all } t : (s \sim_i t \textit{ implies } \mathcal{M}, t \vDash \varphi) \\
\mathcal{M}, s \vDash [\pi]\varphi & \Leftrightarrow & \textit{for all observations } w \textit{ over } \Sigma, \\
& & w \in \mathcal{L}(\pi) \cap Pre(Exp(s)) \\
& & \textit{implies } \mathcal{M}|_w, s \vDash \varphi
\end{array}$$

where $Pre(\pi)$ is the set of prefixes of words in $\mathcal{L}(\pi)$, that is, $w \in Pre(\pi)$ iff $\exists v \in \mathbf{\Sigma}^*$ such that $wv \in \mathcal{L}(\pi)$ (namely $\mathcal{L}(\pi\backslash w) \neq \emptyset$).

The truth of $K_i\varphi$ at s follows the standard possible world semantics of epistemic logic. The formula $[\pi]\varphi$ holds at s if for every observation w in the set $\mathcal{L}(\pi)$ that matches with the beginning of (i.e., is a prefix of) some expected observation in s, φ holds at s in the updated model $\mathcal{M}|_w$. Note that s is a state in $\mathcal{M}|_w$ because $w \in Pre(Exp(s))$. Similarly, the truth definition of $\langle\pi\rangle\varphi$ can be given as follows: $\mathcal{M}, s \vDash \langle\pi\rangle\varphi$ iff there exists $w \in \mathcal{L}(\pi) \cap Pre(Exp(s))$ such that $\mathcal{M}|_w, s \vDash \varphi$. Intuitively, the formula $\langle\pi\rangle\varphi$ holds at s if there is an observation w in $\mathcal{L}(\pi)$ that matches with the beginning of some expected observation in s, and φ holds at s in the updated model $\mathcal{M}|_w$. For the example described earlier, we have:

- $\mathcal{M}, t \models [\blacktriangleleft](K_{Bob}\neg debris \wedge \hat{K}_{Alice}debris)$, if the $vbot$ moves one step left, $\blacktriangleleft$, then while Alice still considers moving to the debris-disposal area a possibility, Bob does not consider that possibility at all.

Satisfiability Problem for POL^-: Given a formula φ, does there exist a pointed epistemic expectation model $\mathcal{M}, s$ such that $\mathcal{M}, s \models \varphi$? We investigate the complexity of this problem. The fragments of POL^- that we consider are (i) single-agent word fragment, (ii) multi-agent word fragment, (iii) single-agent POL^-, and, (iv) full POL^-.

3 An Application

Let us now consider a scenario which can be aptly described using the satisfiability problem of POL^-. We go back to the cleaning bot example introduced earlier. Let Alice be agent a and Bob be agent b. Suppose the $vbot$ is moving towards the power source without making any error. Evidently, the possibilities considered by the agents, based on the information available to them are given as follows:

- Possibilities considered by Alice who has the information about the glitch in the bot:

$$\hat{K}_a debris \wedge \hat{K}_a\langle \blacktriangleleft + \blacktriangledown\rangle debris \wedge \hat{K}_a power$$

- Possibilities considered by Bob who is not aware of the glitch in the bot:

$$\hat{K}_b debris \wedge \hat{K}_b power$$

Now, we model the *expectations* as follows: Consider the expression, $\pi^p_n = (\blacktriangledown + \blacktriangleleft)^n$ that represents a sequence of moves of length n the bot can make to get to to the power source without any error. We use a formula P_n to express the following: As long as the bot is observed to make n many moves towards the power source, reaching it is still a possibility.

$$\begin{aligned}
P_n =& (\langle\blacktriangleleft\rangle\top \wedge \langle\blacktriangledown\rangle\top) \\
& \wedge [\pi^p_1](\langle\blacktriangleleft\rangle\top \wedge \langle\blacktriangledown\rangle\top) \\
& \wedge [\pi^p_2](\langle\blacktriangleleft\rangle\top \wedge \langle\blacktriangledown\rangle\top) \ldots \\
& \wedge [\pi^p_n](\langle\blacktriangleleft\rangle\top \wedge \langle\blacktriangledown\rangle\top)
\end{aligned}$$

The first conjunct of P_n translates to move towards the power source, a move towards down or left can be observed. The second conjunct translates to the following: after the observation of a single left or down movement, another left or down movement can be observed. The other conjuncts can be described similarly.

For the scenario described in the introduction, we can consider P_n to create a formula where n is at most 3, without an error. Let us denote such a formula by ψ_p. Similarly, a formula can express the movement towards debris-disposal with at most one error and with no error as ψ_{de} and ψ_d, respectively. A situation where the bot is moving towards the power source without any error, but a considers the possibility of moving towards debris-disposal with an error can be expressed as $\hat{K}_a\psi_{de}\wedge\psi_p$. Similarly, a formula can be considered for modelling the expected observation when both the agents consider the possibility of the bot moving towards debris-disposal area without an error: $\hat{K}_a\psi_d \wedge \hat{K}_b\psi_d$. We call the (finite) set of all such formulas, Γ_p. Similarly, we can construct a set Γ_{de} of formulas, when the bot can make an error while going towards debris-disposal area or Γ_d when it is moving towards the debris-disposal without any error.

Suppose we want to conclude the following in the current scenario: After one wrong move, b knows that the bot is not moving towards debris-disposal, but a still considers the possibility. The formula, $INFO_{ab}$, say, turns out to be

$$\langle \blacktriangledown + \blacktriangleleft \rangle (K_b power \wedge \hat{K}_a debris)$$

The actual scenario is that the bot is indeed moving towards $power$. Hence, to check whether $INFO_{ab}$ can be concluded in this scenario, a satisfiability solver for POL^- can check the (un)satisfiability of the formula

$$\neg\Big(\big(\bigwedge_{\psi\in\Gamma_p} \psi\big) \rightarrow INFO_{ab}\Big)$$

4 Algorithm for the Satisfiability Problem of POL^-

In this section, we design a proof system using the tableau method to prove satisfiability of POL^-.

A term in a tableau proof is of the form $(\sigma \;\; w \;\; \psi) \mid (\sigma \;\; w \;\; \checkmark) \mid (\sigma,\sigma')_i$, where $i \in Agt$. The σ is called a state label that represents a state in the model, $w \in \Sigma^*$ is a word over a finite alphabet and ψ is a formula in POL^-.

The term $(\sigma \;\; w \;\; \psi)$ represents the fact that the state labelled by σ survives after the model is projected on the word w, and after projecting on w, ψ holds true in the state corresponding to σ.

The term $(\sigma \;\; w \;\; \checkmark)$ represents the fact that the state labelled by σ survives after the model is projected on word w.

The term $(\sigma_1,\sigma_2)_i$ represents in the model, the states represented by σ_1 and σ_2 should be indistinguishable for the agent $i \in Agt$, where Agt is a finite set of agents.

For space reasons, the term $(\sigma_1,\sigma_2)_{i\in Agt}$ stands for the set of terms $\{(\sigma_1,\sigma_2)_i \mid i \in Agt\}$.

Without loss of generality, the formula φ is assumed to be in Negative Normal form, the syntax of which is as follows:

$$\varphi := \top \mid p \mid \neg p \mid \psi\vee\chi \mid \psi\wedge\chi \mid$$
$$\hat{K}_i\psi \mid K_i\psi \mid \langle\pi\rangle\psi \mid [\pi]\psi$$

Given a formula we denote by φ, $FL(\varphi)$ the Fischer-Ladner Closure of φ, (see (Harel, Tiuryn, and Kozen 2000)).

Propositional Rules

Clash rule	$\dfrac{(\sigma \;\; w \;\; p),\ \ (\sigma \;\; w \;\; \neg p)}{\bot}$
AND rule	$\dfrac{(\sigma \;\; w \;\; \psi\wedge\chi)}{(\sigma\ w\ \psi),(\sigma\ w\ \chi)}$
OR rule	$\dfrac{(\sigma \;\; w \;\; \psi\vee\chi)}{(\sigma\ w\ \psi) \mid (\sigma\ w\ \chi)}$

Knowledge Rules

Knowledge	$\dfrac{(\sigma \;\; w \;\; K_i\psi),\ \ (\sigma' \;\; w \;\; \checkmark),\ \ (\sigma,\sigma')_i}{(\sigma'\ w\ \psi)}$
Possibility	$\dfrac{(\sigma \;\; w \;\; \hat{K}_i\psi)}{(\sigma,\sigma_n)_i, (\sigma_n\ w\ \checkmark), (\sigma_n\ w\ \psi)}$
Reflexivity	$\dfrac{--}{(\sigma,\sigma)_i, \text{ for all } i\in Agt}$
Transitivity	$\dfrac{(\sigma,\sigma'')_i,\ \ (\sigma'',\sigma')_i}{(\sigma,\sigma')_i}$
Symmetry	$\dfrac{(\sigma',\sigma)_i}{(\sigma,\sigma')_i, i\in Agt}$

Diamond and Box Rules

Diamond Decompose	$\dfrac{(\sigma \;\; w \;\; \langle\pi\pi'\rangle\psi)}{(\sigma\ w\ \langle\pi\rangle\langle\pi'\rangle\psi)}$
Diamond ND Decompose	$\dfrac{(\sigma \;\; w \;\; \langle\pi_1+\pi_2\rangle\psi)}{(\sigma\ w\ \langle\pi_1\rangle\psi) \mid (\sigma\ w\ \langle\pi_2\rangle\psi)}$
Diamond Project	$\dfrac{(\sigma \;\; w \;\; \langle a\rangle\psi)}{(\sigma\ wa\ \checkmark),(\sigma\ wa\ \psi)}$
Box Project	$\dfrac{(\sigma \;\; w \;\; [\pi]\psi),\ \ (\sigma\ wa\ \checkmark)}{(\sigma\ wa\ [\pi\backslash a]\psi)}$
Empty Box	$\dfrac{(\sigma \;\; w \;\; [\epsilon]\psi)}{(\sigma\ w\ \psi)}$

Survival Rules

Constant Valuation Up	$\dfrac{(\sigma \;\; w \;\; p)}{(\sigma\ \epsilon\ p)} \quad \dfrac{(\sigma \;\; w \;\; \neg p)}{(\sigma\ \epsilon\ \neg p)}$
Survival Chain	$\dfrac{(\sigma \;\; wa \;\; \checkmark)}{(\sigma\ w\ \checkmark)}$

Figure 4: Tableau rules. σ is any state symbol, w is any word, p is any propositional variable, i is any agent, π is any regular expression, a is any letter. Note that Reflexivity rule has no antecedent.

4.1 The Tableau Rules

The tableau rules for this fragment have been shown in Figure 4. Here an inference rule looks like this: $\dfrac{A}{C_1|C_2|\ldots|C_n}$.

Here each C_i and A is a set of tableau terms. The C_is are called consequences, A is the antecedent. Intuitively the rule is interpreted as "If all the terms in A are true, then all the terms in at least one of C_i's are true".

In Figure 4, the left column is the rule name and the right column is the rule. For example, the Box Project Rule states that "The state labelled by σ survives after projection on word w and it satisfies $[\pi]\psi$ ($(\sigma \;\; w \;\; [\pi]\psi)$) and σ still survives a further projection on letter a($(\sigma \;\; wa \;\; \checkmark)$) then after further projection on a, $[\pi\backslash a]\psi$ should hold true in the state labelled by σ ($(\sigma \;\; wa \;\; [\pi\backslash a]\psi)$).". Recall $\pi\backslash a$ denotes the residual of π by a (see Section 2).

Similarly, the Diamond Project rule says that if a certain state σ, under some word projection w has to satisfy $\langle a\rangle\psi$, then that state σ has to survive projection on wa and also satisfy ψ under the same projection.

A tableau proof can be assumed a tree. Each node of the tree is a set of tableau terms Γ. An inference rule can be applied in the following way:

If $A \subseteq \Gamma$ and C_i's are not in Γ, the children of Γ are $\Gamma \cup C_i$ for each $i \in [n]$.

When no rules can be applied on a Γ, we say Γ is saturated (leaf node in the proof tree).

If $\bot \in \Gamma$, we say that branch is **closed**. If all the branches of the proof tree is **closed**, we say the **tableau is closed**, else is open.

Given a POL^- formula φ, we start with $\Gamma = \{(\sigma\ \epsilon\ \varphi), (\sigma\ \epsilon\ \checkmark)\} \cup \{(\sigma, \sigma)_i, i \in Agt\}$.

Example 3. *Suppose we aim at deciding whether*

$$\varphi := \hat{K}_i\langle a\rangle p \wedge \langle a\rangle K_i \neg p$$

is satisfiable or not. For simplicity we suppose there is a single agent i. Here are the terms added to the set of terms:

1. $(\sigma\ \epsilon\ \varphi), (\sigma\ \epsilon\ \checkmark)$ *(initialization)*
2. $(\sigma, \sigma)_i$ *by Reflexivity rule*
3. $(\sigma\ \epsilon\ \hat{K}_i\langle a\rangle p)$, $(\sigma\ \epsilon\ \langle a\rangle K_i\neg p)$ *by AND rule*
4. $(\sigma'\ \epsilon\ \langle a\rangle p), (\sigma'\ \epsilon\ \checkmark), (\sigma, \sigma')_i$ *by Possibility rule*
5. $(\sigma', \sigma')_i$ *by Reflexivity rule*
6. $(\sigma', \sigma)_i$ *by Symmetry rule*
7. $(\sigma'\ a\ p), (\sigma'\ a\ \checkmark)$ *by Diamond Project on 2*
8. $(\sigma\ a\ \checkmark), (\sigma\ a\ K_i\neg p)$ *by Diamond Project on 2*
9. $(\sigma'\ a\ \neg p)$ *by Knowledge rule on 3, 5, 6*
10. $\bot$ *by Clash rule on 5,7*

As we obtain $\bot$, the formula φ is not satisfiable (by the upcoming Theorem 6).

4.2 Soundness and Completeness of the Tableau Rules

In this section, we provide the soundness and completeness proof of the Tableau method for the satisfiability of POL^-

Theorem 6. *Given a formula φ, if φ is satisfiable, then the tableau for $\Gamma = \{(\sigma\ \epsilon\ \varphi), (\sigma\ \epsilon\ \checkmark)\}$ is open.*

Theorem 7. *Given a formula φ, if the tableau for $\Gamma = \{(\sigma\ \epsilon\ \varphi), (\sigma\ \epsilon\ \checkmark)\}$ is open, then φ is satisfiable.*

The proof of Theorem 6 is done by induction. For shortage of space, we give the proof of Theorem 6 in the extended version (Chakraborty et al. 2023).

Proof of Theorem 7. Since by assumption, the tableau for $\Gamma = \{(\sigma\ \epsilon\ \varphi), (\sigma\ \epsilon\ \checkmark)\}$ is open, there exists a branch in the tableau tree where in the leaf node there is a set of terms Γ_l such that it is saturated and $\bot \notin \Gamma_l$.

For the purpose of this proof, let us define a relation over the words $\bar{w}$ that appears in Γ_l. For any two word $\bar{w}_1$ and $\bar{w}_2$ that appears in Γ_l, $\bar{w}_1 \leq_{pre} \bar{w}_2$ if and only if $\bar{w}_1 \in Pre(\bar{w}_2)$. Now, this relation is reflexive ($\bar{w}_1 \in Pre(\bar{w}_1)$), asymmetric (if $\bar{w}_1 \in Pre(\bar{w}_2)$ and $\bar{w}_2 \in Pre(\bar{w}_1)$ then $\bar{w}_1 = \bar{w}_2$) and transitive (if $\bar{w}_1 \in Pre(\bar{w}_2)$ and $\bar{w}_2 \in Pre(\bar{w}_3)$ then $\bar{w}_1 \in Pre(\bar{w}_3)$). Hence this relation creates a partial order among all the words occurring in Γ_l. We also denote $\bar{w}_1 <_{pre} \bar{w}_2$ to interpret the fact that $\bar{w}_1 \leq_{pre} \bar{w}_2$ and $\bar{w}_1 \neq \bar{w}_2$.

Now we create a model $\mathcal{M} = \langle W, \{R_i\}_{i \in Agt}, V, Exp\rangle$ out of Γ_l and prove that φ is satisfied by some state in the model.

- $W = \{s_\sigma \mid \sigma$ is a distinct label in the terms in $\Gamma_l\}$
- $R_i = \{\{s_{\sigma_1}, s_{\sigma_2}\} \mid (\sigma_1, \sigma_2)_i \in \Gamma_l\}$
- $V(s_\sigma) = \{p \mid (\sigma\ \epsilon\ p) \in \Gamma_l\}$
- $Exp(s_\sigma) = \sum_{w \in \Lambda_\sigma} w$, where $\Lambda_\sigma = \{w \mid (\sigma\ w\ \checkmark) \in \Gamma_l$ and $\nexists w' : ((\sigma\ w'\ \checkmark) \in \Gamma_l$ and $w <_{pre} w')\}$

Note that, the new state label σ_n is only created in the possibility rule, with a reflexive relation on itself. Now consider the set $R' = \{(\sigma, \sigma') \mid \{(\sigma\ w\ \checkmark), (\sigma'\ w'\ \checkmark)\} \subseteq \Gamma_l\}$. Hence this can be considered a binary relation over the set of all distinct σ that occurs in Γ_l. When a σ' is created by the possibility rule, by the relation rules, the reflexive, symmetric and transitive conditions are satisfied with respect to every other label that has previously been there. Hence R' is an equivalence relation, hence making R_i in the model an equivalence relation.

Now, Theorem 7 follows from the following two claims, the proofs of which we present later.

Claim 8. *If $(\sigma\ w\ \checkmark) \in \Gamma_l$ then s_σ survives in $\mathcal{M}|_w$.*

Claim 9. *For any word w that occurs in Γ_l, any label σ and any formula ψ, If $(\sigma\ w\ \psi) \in \Gamma_l$ and $(\sigma\ w\ \checkmark) \in \Gamma_l$ then s_σ survives in $\mathcal{M}|_w$ and $\mathcal{M}|_w, s_\sigma \models \psi$.*

Proof of Claim 8. We induct on the size of $|w|$.

Base Case. Let $|w| = 1$. Hence $w \in \{\epsilon\} \cup \Sigma$. Since $\Gamma \subseteq \Gamma_l$ and $(\sigma\ \epsilon\ \checkmark)$, and s_σ is in $\mathcal{M}|_\epsilon = \mathcal{M}$.

For the case $w = a$ for any $a \in \Sigma$. Hence there exists a word w' that occurs in a term in Γ_l labelled by σ such that $w \in Pre(w')$) and there is no other word bigger than w' such that w' is in its prefix, since the proof is on finite words and formula, the proof terminates. Hence by definition of $w' \in \mathcal{L}(Exp(s_\sigma))$ which guarantees survival of s_σ in $\mathcal{M}|_a$.

Induction Hypothesis. Assume the statement to be true for $|w| = n$.

Inductive Step. Consider the case where $|w| = n + 1$.

By assumption, $(\sigma\ w\ \checkmark) \in \Gamma_l$. Hence by the fact that Γ_l is saturation and by the rule "Survival Chain", there is $(\sigma\ w'\ \checkmark) \in \Gamma_l$, where $w = w'a$ for some $a \in \Sigma$. Hence by IH, the result follows that s_σ survives in $\mathcal{M}|_{w'}$.

Now, by termination, there are finite many unique words occurring in Γ_l. Clearly, $w' \leq_{pre} w$. Since there are finite many words, there is a w_*, which is of maximum size such that $w \leq_{pre} w_*$ and $(\sigma\ w_*\ \checkmark) \in \Gamma_l$. Hence $w_* \in \Lambda_\sigma$ in the definition of Exp of the model. Therefore $w_* \in \mathcal{L}(Exp(s_\sigma))$ and since $w' \leq_{pre} w \leq_{pre} w_*$, s_σ survives in $\mathcal{M}|_{w'}$, hence s_σ shall survive in $\mathcal{M}|_w$. ☐

Proof of Claim 9. Naturally, we shall induct upon the size of ψ.

Base Case. Let ψ is of the form p or $\neg p$. By the definition of the function V for the model and the previous proof, the statement stands true.

Induction Hypothesis. Let us consider the statement is true for any ψ such that $|\psi| < n'$ for some n'.

Inductive Step. We prove for $|\psi| = n'$. Again, we go case by case on the syntax of ψ.

- $\psi = \hat{K}_i\chi$. Since Γ_l is saturated, by the rule of possibility, $\{(\sigma' \ \ w \ \ \chi), (\sigma, \sigma')_i, (\sigma' \ \ w \ \ \checkmark)\} \subseteq \Gamma_l$. By IH on the subformula χ, the definition of the model, the proof of the previous statement, and the rule "survival chain", $s_{\sigma'}$ survives in $\mathcal{M}|_w$ and $\mathcal{M}|_w, s_{\sigma'} \vDash \chi$. Also by definition, $\{s_\sigma, s_{\sigma'}\} \in R_i$, hence proving $\mathcal{M}|_w, s_\sigma \vDash \hat{K}_i\chi$.
- $\psi = K_i\chi$. Since Γ_l is saturated, and by previous statement $s_{\sigma'}$ is surviving for every $(\sigma' \ \ w \ \ \checkmark)$, by the rule of knowledge $(\sigma' \ \ w \ \ \chi) \in \Gamma_l$ for every $(\sigma, \sigma')_i$. Hence by IH on subformula, $\mathcal{M}|_w, s_{\sigma'} \vDash \chi$ for every σ' such that $\{\sigma, \sigma'\} \in R_i$.
- $\psi = \langle \pi + \pi' \rangle\chi$. Since Γ_l is saturated, hence by the ND Decomposition, either the term $(\sigma \ \ w \ \ \langle\pi\rangle\chi) \in \Gamma_l$ or $(\sigma \ \ w \ \ \langle\pi'\rangle\chi) \in \Gamma_l$. By IH, $\mathcal{M}|_w, s_\sigma \vDash \langle\pi\rangle\chi$ or $\mathcal{M}|_w, s_\sigma \vDash \langle\pi'\rangle\chi$ and hence $\mathcal{M}|_w, s_\sigma \vDash \langle\pi + \pi'\rangle\chi$.
- $\psi = \langle\pi\pi'\rangle\chi$. Since Γ_l is saturated, hence $(\sigma \ \ w \ \ \langle\pi\rangle\langle\pi'\rangle\chi) \in \Gamma_l$. By IH, since $\langle\pi\rangle\langle\pi'\rangle\chi \in FL(\psi)$, hence $\mathcal{M}|_w, s_\sigma \vDash \psi$.
- $\psi = \langle a\rangle\chi$. Note that we don't consider a general word w' in the diamond as given $w' = aw''$, a formula $\langle w'\rangle\chi$ is satisfiable if and only if $\langle a\rangle\langle w''\rangle\chi$ is satisfiable.
- $\psi = [\pi]\chi$. Let us consider $(\sigma \ \ wa \ \ \checkmark) \in \Gamma_l$ for some $a \in \Sigma$. Hence by the proof of the first statement, $s_\sigma \in \mathcal{M}|_{wa}$. Also $|\mathcal{L}(\pi)| < |\mathcal{L}(\pi \backslash a)|$. Hence by induction on the size of formula $\mathcal{M}|_{wa}, s\sigma \vDash [\pi\backslash a]\chi$ which implies $\mathcal{M}|_w, s_\sigma \vDash [\pi]\chi$. □

This completes the proof of Theorem 7 □

4.3 A NEXPTIME Upper Bound

Now we design an algorithm based on tableau and prove existence of an algorithm that takes non-deterministically exponential steps with respect to the size of φ. Now given a φ, we now create a tree of nodes, where each node T_σ contains terms of the tableau of the form $(\sigma \ \ w \ \ \psi)$ and $(\sigma \ \ w \ \ \checkmark)$, where $w \in \Sigma^*$ is a word that is occuring in tableau, and ψ is a formula in $FL(\varphi)$. Each node T_σ refers to a state label σ in tableau, a term of the $(\sigma \ \ w \ \ \psi) \in T_\sigma$ intuitively translates to in the state corresponding to σ, after projecting model on w, the state survives and there ψ is satisfied. Similarly, $(\sigma \ \ w \ \ \checkmark) \in T_\sigma$ means state corresponding to σ survives after projection on w. The tableau tree created, we call it $\mathcal{T}_\mathsf{P}$

We saturate the rules carefully such that each node in the tree corresponds to a single state in the model. This technique is well studied in (Halpern and Moses 1992).

Theorem 10. *The satisfiability of* POL^- *is in* NEXPTIME.

Proof. Given the tree $\mathcal{T}_\mathsf{P}$ we create in the procedure, a node T_σ is marked satisfiable iff it does not have bot, $\{(\sigma \ \ w \ \ K_i\psi), (\sigma \ \ w \ \ \neg\psi)\} \nsubseteq T_\sigma$ and all its successors are marked satisfiable. We prove three statements:

- **Statement 1:** Each node is of at most exponential size, that is, has at most exponential many terms.
- **Statement 2:** Maximum children a node can have is polynomial.
- **Statement 3:** The height of the tree is polynomial.

Proof of Statement 1. Since a term in a node T_σ is of the form $(\sigma \ \ w \ \ \psi)$, where w is a word over some finite alphabet Σ and ψ is a formula of POL^-.

According to the shape of the rules, a formula that can be derived is always in $FL(\varphi)$. Since $|FL(\varphi)| \leq O(|\varphi|)$(Harel, Tiuryn, and Kozen 2000), hence there can be at most $O(|\varphi|)$ many formulas.

Also, since a regular expression π occuring in a modality is star-free (that is does not contain the Kleene star), hence a word $w \in \mathcal{L}(\pi)$ is of length at most $|\pi|$ which is again of length at most φ. Also there are at most $|FL(\varphi)|$ many regular expressions. Hence there are at most $|\Sigma|^{O(p(|\varphi|))}$, where $p(X)$ is some polynomial on X, many unique words possible. Hence therefore, there can be at most exponential many terms in a single node.

Proof of Statement 2. From a node T_σ, a child is created for every unique triplet of $(\sigma \ \ w \ \ \hat{K}\psi)$ in T_σ. Number of such triplets possible is, as proved is at most polynomial with respect to $|\varphi|$.

Proof of Statement 3. For proving this, we use $md(\Gamma)$, given a set of formulas Γ, is the maximum modal depth over all formulas in Γ. Finally we define $F(T_\sigma)$ as the set of formulas occuring in the node T_σ.

Consider T_σ, the node $T^i_{\sigma'}$ is i- successor of T_σ and $T^j_{\sigma''}$ be the j successor of $T^i_{\sigma'}$ $(i \neq j)$. Note that all the formulas in $F(T^j_{\sigma''})$ are from FL closure of all the K_j and $\hat{K}_j$ formulas from $F(T^i_{\sigma'})$.

Also all the formulas in $F(T^i_{\sigma'})$ are in the FL closure of the K_i and $\hat{K}_i$ formulas occusring in T_σ. Hence $md(T^j_{\sigma''}) \leq md(F(T^i_{\sigma'}))$. Therefore, there can be at most $O(|\varphi|^c)$ such agent alterations in one path of $\mathcal{T}_P$ (not linear because there can be polynomial many words paired with each formula).

Now let us consider how many consecutive i succesors can happen in a path. Suppose a T_σ has a new i-successor node $T_{\sigma'}$ for the term $(\sigma \ \ w \ \ \hat{K}_i\psi)$. Due to the fact that the indistinguishability relation is equivalence for each agent due to the Transitivity, Symmetry rule and the reflexivity that infers in the possibility rule, hence all the possibility and the knowledge formula terms of the form $(\sigma \ \ w' \ \ \hat{K}_i\xi)$ or $(\sigma \ \ w' \ \ K_i\xi)$ of agent i are in the successor node $T_{\sigma'}$ in the form $(\sigma' \ \ w' \ \ \hat{K}_i\xi)$ or $(\sigma' \ \ w' \ \ \hat{K}_i\xi)$ respectively, along with the term $(\sigma' \ \ w \ \ \psi)$. Hence the number of such unique combination of terms will be at most polynomial to the size of $|FL(\varphi)|$.

Therefore, the height of $\mathcal{T}_\mathsf{P}$ is polynomial with respect to the $|\varphi|$. □

5 Hardness of Satisfiability in POL^-

In this section, we give a lower bound to the Satisfiability problem of POL^-. We reduce the well-known NEXPTIME-

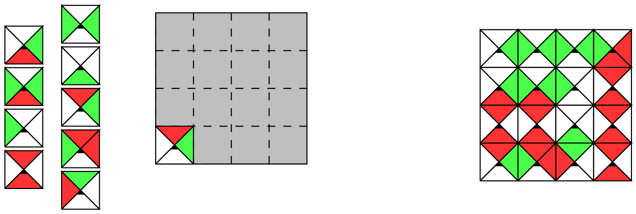

Figure 5: A set of tile types and an empty square, and a solution.

Complete Tiling problem to come up with a formula in the POL^- fragment that only has 2 agents.

Theorem 11. *POL^- satisfiability problem is $\mathsf{NEXPTIME}$-Hard.*

Proof. We reduce the $\mathsf{NEXPTIME}$-Complete tiling problem of a square whose size is 2^n where n is encoded in unary (van Emde Boas 2019) (see Figure 5). The instance of the tiling problem is (T, t_0, n) where T is a set of tile types (e.g), t_0 is a specific tile that should be at position $(0, 0)$, and n is an integer given in unary. Note that the size of the square is exponential in n. We require the colours of the tiles to match horizontally and vertically.

The idea of the reduction works as follows. We consider two tilings A and B. We will construct a formula $tr(T, t_0, n)$ expressing that the two tilings are equal, contains t_0 at (0, 0), and respect the horizontal and vertical constraints.

With the help of two epistemic modalities K_i and K_j we can simulate a standard K modal logic $\Box$. For the rest of the proof, we consider such a $\Box$ modality and its dual $\Diamond$. We encode a binary tree whose leaves are pairs of positions (one position in tiling A and one in tiling B). Such a tree is of depth $4n$: n bits to encode the x-coordinate in tiling A, n bits to encode the x-coordinate in tiling B, n bits to encode the y-coordinate in tiling A, n bits to encode the y-coordinate in tiling B. A pair of positions is encoded with the $4n$ propositional variables: $p_0, \dots, p_{4n-1}$. The first $p_0, \dots, p_{2n-1}$ encodes the position in tiling A while the later $p_{2n}, \dots, p_{4n-1}$ encodes the position in tiling B. At each leaf, we also use propositional variables q_t^A (resp. q_t^B) to say there is tile t at the corresponding position in tiling A (resp. tiling B). The following formula enforces the existence of that binary tree $\mathcal{T}$ by branching over the truth value of proposition p_ℓ at depth ℓ:

$$\bigwedge_{\ell<4n} \Box^\ell \left(\Diamond p_\ell \wedge \Diamond \neg p_\ell \wedge \bigwedge_{i<\ell} (p_i \rightarrow \Box p_i) \wedge (\neg p_i \rightarrow \Box \neg p_i) \right) \quad (1)$$

Now, by using of specific Boolean formulas over $p_0, \dots, p_{4n-1}$, it is easy to express equality, presence of t_0 at $(0, 0)$ and horizontal and vertical constraints:

$$\Box^{4n} \left(\bigvee_t q_t^A \;\wedge\; \bigwedge_{t \neq t'} (\neg q_t^A \vee \neg q_{t'}^A) \right) \quad (2)$$

$$\Box^{4n} \left(\bigvee_t q_t^B \;\wedge\; \bigwedge_{t \neq t'} (\neg q_t^B \vee \neg q_{t'}^B) \right) \quad (3)$$

$$\Box^{4n} (\text{position in tiling } A = 0) \rightarrow q_{t_0}^A \quad (4)$$

$$\Box^{4n} \left(\begin{array}{l} x\text{-coordinate of position in } A \\ = 1 + x\text{-coordinate of position in } B \end{array} \right) \quad (5)$$

$$\rightarrow \bigvee_{t, t' \mid t \text{ matches } t' \text{ horizontally}} (q_t^A \wedge q_{t'}^B) \quad (6)$$

$$\Box^{4n} \left(\begin{array}{l} y\text{-coordinate of position in } A \\ = 1 + y\text{-coordinate of position in } B \end{array} \right) \quad (7)$$

$$\rightarrow \bigvee_{t, t' \mid t \text{ matches } t' \text{ vertically}} (q_t^A \wedge q_{t'}^B) \quad (8)$$

The main difficulty is to be sure that all pairs of positions with the same position for - let's say - tiling A indicates the same tile for the tiling A (i.e. the same variable q_t^A is true). To this aim, we will write a formula of the following form

$$[\pi_{\text{ any position in A }}] \bigvee_t \Box^{4n} q_t^A \;\wedge\; [\pi_{\text{any position in B}}] \bigvee_t \Box^{4n} q_t^B.$$

To be able to perform observations to select any position in tiling A (resp. B) whatever the position in tiling B (resp. A) is, we introduce the alphabet $\Sigma = \{A, \bar{A}, B, \bar{B}\}$. We write these two formulas that make a correspondence between valuations on the leaves and observations:

$$\Box^{4n} \bigwedge_{i=0..2n-1} [A + \bar{A}]^i \left(\begin{array}{l} (p_i \rightarrow \langle A \rangle \top \wedge [\bar{A}] \bot) \wedge \\ (\neg p_i \rightarrow \langle \bar{A} \rangle \top \wedge [A] \bot) \end{array} \right) \quad (9)$$

$$\Box^{4n} \bigwedge_{i=2n..4n-1} [B + \bar{B}]^{i-2n} \left(\begin{array}{l} (p_i \rightarrow \langle B \rangle \top \wedge [\bar{B}] \bot) \wedge \\ (\neg p_i \rightarrow \langle \bar{B} \rangle \top \wedge [B] \bot) \end{array} \right) \quad (10)$$

The idea is that a $2n$-length word on alphabet $\{A, \bar{A}\}$ corresponds to a valuation over $p_1, \dots, p_{2n-1}$, and thus a position in tiling A and only that $2n$-length word on alphabet $\{A, \bar{A}\}$ is observable. In the same way, a word on alphabet $\{B, \bar{B}\}$ corresponds to a valuation over $p_{2n}, \dots, p_{4n-1}$, thus a position in tiling B.

We also say that the inner node (non-leaf) of the binary tree is never pruned by observations (all $2n$-length words over $\{A, \bar{A}, B, \bar{B}\}$ are observable):

$$\Box^{<4n} \bigwedge_{i=0..2n-1} [\Sigma]^i (\langle A \rangle \top \wedge \langle \bar{A} \rangle \top \wedge \langle B \rangle \top \wedge \langle \bar{B} \rangle \top) \quad (11)$$

The formula for ensuring the uniqueness of q_t^A whatever the position in tiling B, and the other way around are then:

$$[(A + \bar{A})^{2n}] \bigvee_t \Box^{4n} q_t^A \wedge [(B + \bar{B})^{2n}] \bigvee_t \Box^{4n} q_t^B \quad (12)$$

The intuition works as follows. When evaluating $[(A + \bar{A})^{2n}] \Box^{4n} q_t^A$, we consider all words w in $\mathcal{L}((A + \bar{A})^{2n})$

and we consider any pruning $\mathcal{M}|_w$ of the model $\mathcal{M}$ which contains the binary tree $\mathcal{T}$. In $\mathcal{M}|_w$, only the leaves where the valuation on $p_0, \dots, p_{2n-1}$ that corresponds to w stays. With $\bigvee_t$, we choose a tile type t in T. The modality $\Box^{4n}$ then reaches all the leaves and imposes that q_t^A holds.

The reduction consists of computing from an instance (T, t_0, n) of the tiling problem the POL^- formula $tr(T, t_0, n)$ which is the conjunction of (1-12), which is computable in poly-time in the size of (T, t_0, n) (recall n is in unary). Furthermore, one can check that (T, t_0, n) is a positive instance of the tiling problem iff $tr(T, t_0, n)$ is satisfiable. $\Box$

6 Complexity Results of Fragments of POL^-

In this section, we consider a few fragments of POL^- and we give complexity results for them. First, we consider the single agent fragment of POL^-, and then we prove complexity results for the word fragment of POL^- (both single and multi-agent) using reductions to PAL.

6.1 Single Agent Fragment of POL^-

While we have shown (in Theorem 10) that the satisfiability problem of the POL^- is NEXPTIME-Hard, the hardness proof holds only for the case when the number of agents is at least 2. However, we prove that satisfiability problem in the single Agent fragment of POL^- is PSPACE-Hard, although single-agent epistemic logic $S5$ is NP-Complete.

We prove it by reducing TQBF into our problem. The TQBF problem is: given a formula φ of the form $Q_1 x_1 Q_2 x_2 \dots Q_n x_n \xi(x_1, x_2, \dots, x_n)$ where $Q_i \in \{\forall, \exists\}$ and $\xi(x_1, x_2, \dots, x_n)$ is a Boolean formula in CNF over variables $x_1, \dots, x_n$, decide whether the formula φ is true.

Theorem 12. *The satisfiability problem for single agent fragment of* POL^- *is* PSPACE*-Hard.*

The proof follows in the same lines as the proof of PSPACE-Hardness of the model-checking problem of the POL^- (Chakraborty et al. 2022). The complete proof can be found in the extended version (Chakraborty et al. 2023).

6.2 Word Fragment of POL^-

To investigate the complexity of the satisfaction problem of the word fragment of POL^-, we use a translation of POL^- to PAL. Before going forward, let us give a very brief overview of the syntax and semantics of PAL.

Public announcement logic (PAL) To reason about announcements of agents and their effects on agent knowledge, PAL (Plaza 2007) was proposed. The underlying model that is dealt with in PAL is epistemic, $\langle S, \sim, V\rangle$ where S is a non-empty set of states, $\sim$ assigns to each agent in Agt an equivalence relation $\sim_i \subseteq S \times S$, and $V : S \to 2^{\mathcal{P}}$ is a valuation function. The language is given as follows:

Definition 13 (PAL syntax)**.** *Given a countable set of* propositional variables $\mathcal{P}$*, and a finite set of* agents Agt*, a formula* φ *in Public Announcement Logic (*PAL*) can be defined recursively as:*

$$\varphi := \top \mid p \mid \neg\varphi \mid \varphi \wedge \varphi \mid K_i\varphi \mid [\varphi!]\varphi$$

where $p \in \mathcal{P}$*, and* $i \in Agt$*.*

Typically, $[\varphi!]\psi$ says that 'if φ is true, then ψ holds after having publicly announced φ'. Similarly, as in POL^- syntax, the respective dual formulas are defined as,

$$\hat{K}_i\psi = \neg K_i \neg\psi$$
$$\langle\varphi!\rangle\psi = \neg[\varphi!]\neg\psi$$

Formula $\langle\varphi!\rangle\psi$ says that φ is true, and ψ holds after announcing φ. Before going into the truth definitions of the formulas in PAL, let us first define the notion of model update.

Definition 14 (Model Update by Announcement)**.** *Given an epistemic model,* $\mathcal{M} = \langle S, \sim, V\rangle$*,* $s \in S$*, and a* PAL *formula* φ*, the model* $\mathcal{M}|_\varphi = \langle S', \sim', V'\rangle$ *is defined as:*

- $S' = \{s \in S \mid \mathcal{M}, s \vDash \varphi\}$
- $\sim'_i = \sim_i|_{S' \times S'}$,
- $V'(s) = V(s)$ *for any* $s \in S'$.

Now we are all set to give the truth definitions of the formulas in PAL with respect to pointed epistemic models:

Definition 15 (Truth of a PAL formula)**.** *Given an epistemic model* $\mathcal{M} = \langle S, \sim, V\rangle$ *and an* $s \in S$*, a* PAL *formula* φ *is said to hold at* s *if the following holds:*

- $\mathcal{M}, s \vDash p$ *iff* $p \in V(s)$*, where* $p \in \mathcal{P}$.
- $\mathcal{M}, s \vDash \neg\varphi$ *iff* $\mathcal{M}, s \nvDash \varphi$.
- $\mathcal{M}, s \vDash \varphi \wedge \psi$ *iff* $\mathcal{M}, s \vDash \varphi$ *and* $\mathcal{M}, s \vDash \psi$.
- $\mathcal{M}, s \vDash K_i\varphi$ *iff for all* $t \in S$ *with* $s \sim_i t$, $\mathcal{M}, t \vDash \varphi$.
- $\mathcal{M}, s \vDash [\psi!]\varphi$ *iff* $\mathcal{M}, s \vDash \psi$ *implies*$\mathcal{M}|_\psi, s \vDash \varphi$.

On complexity To study the satisfiability problem for the word fragment of POL^-, we transfer the following result from PAL to POL^-:

Theorem 16. *(Lutz 2006) The satisfiability problem of* PAL *is* NP*-Complete for the single-agent case and* PSPACE*-Complete for the multi-agent case.*

PAL is the extension of epistemic logic with dynamic modal constructions of the form $[\varphi!]\psi$ that expresses 'if φ holds, then ψ holds after having announced φ publicly'. The dynamic operator $\langle\pi\rangle$ in the word fragment of POL^- consists in announcing publicly a sequence of observations. W.l.o.g. as π is a word $a_1 \dots a_k$, $\langle\pi\rangle$ can be rewritten as $\langle a_1\rangle \dots \langle a_k\rangle$. In other words, we suppose that the POL^- dynamic operators only contain a single letter. The mechanism of POL^- is close to Public announcement logic (PAL). Observing a consists in announcing publicly that wa occurred where w is the observations already seen so far.

We introduce fresh atomic propositions p_{wa} to say that letter a is compatible with the current state given that the sequence w was already observed.

For all words $w \in \Sigma^*$, we then define tr_w that translates a POL^- formula into a PAL formula given that w is the already seen observations seen so far:

$$\begin{aligned} tr_w(p) =& p \\ tr_w(\neg\varphi) =& \neg tr_w(\varphi) \\ tr_w(\varphi \wedge \psi) =& tr_w(\varphi) \wedge tr_w(\psi) \\ tr_w(K_i\varphi) =& K_i tr_w(\varphi) \\ tr_w(\langle a\rangle\varphi) =& \langle p_{wa}!\rangle tr_{wa}(\varphi) \end{aligned}$$

We finally transform any POL^- formula φ into $tr(\varphi) := tr_\epsilon(\varphi)$.

Example 4. *Consider the* POL^- *formula* $\varphi := [a]\bot \wedge \langle a\rangle\langle a\rangle\top$. $tr(\varphi)$ *is* $[p_a!]\bot \wedge \langle p_a!\rangle\langle p_{aa}!\rangle\top$. *Note that if* p_a *is false, the truth value of* p_{aa} *is irrelevant.*

Proposition 17. φ *is satisfiable in the word fragment of* POL^- *iff* $tr(\varphi)$ *is satisfiable in* PAL.

Proof. (sketch) $\boxed{\Rightarrow}$ Suppose there is a pointed POL^- model $\mathcal{M}, s_0$ such that $\mathcal{M}, s_0 \models \varphi$. We define $\mathcal{M}'$ to be like $\mathcal{M}$ except that for all states s in $\mathcal{M}$, for all $w \in \Sigma^*$, we say that p_w is true at $\mathcal{M}', s$ iff $Exp(s)\backslash w \neq \emptyset$. It remains to prove that $\mathcal{M}', s_0 \models tr(\varphi)$. We prove by induction on φ that for all $w \in words(\varphi)$, if $Exp(s)\backslash w \neq \emptyset$ then $\mathcal{M}|_w, s \models \varphi$ iff $\mathcal{M}', s \models tr_w(\varphi)$.

We only show the interesting case of $\varphi = \langle a\rangle\psi$. Here the $\mathsf{tr}_w(\langle a\rangle\psi) = \langle p_{wa}!\rangle\mathsf{tr}_{wa}(\psi)$. By assumption, $\mathcal{M}|_w, s \models \langle a\rangle\psi$. Hence $\mathcal{M}|_{wa}, s \models \psi$. Therefore $Exp(s)\backslash wa \neq \emptyset$. By definition of $\mathcal{M}'$, p_{wa} is true in s. Therefore by IH $\mathcal{M}', s \models \mathsf{tr}_{wa}(\psi)$. And since p_{wa} is true, hence $\mathcal{M}', s \models \langle p_{wa}!\rangle\mathsf{tr}_{wa}\psi$. Conversely, assuming $\mathcal{M}', s \models \langle p_{wa}!\rangle tr_{wa}(\psi)$. Hence p_{wa} is true in s. By definition, p_{wa} is true iff $Exp(s)\backslash wa \neq \emptyset$. Also by IH, $\mathcal{M}|_{wa}, s \models \psi$. Hence $\mathcal{M}|_w, s \models \langle a\rangle\psi$.

$\boxed{\Leftarrow}$ Suppose there is a pointed epistemic model $\mathcal{M}', s_0$ such that $\mathcal{M}', s_0 \models tr(\varphi)$. We define a POL^- model $\mathcal{M}$ like $\mathcal{M}'$ except that for all states s, $Exp(s) = \{w \in \Sigma^* \mid \mathcal{M}, s \models p_w\}$. It remains to prove that $\mathcal{M}, s_0 \models \varphi$. For the rest of the proof, we prove by induction on φ that for all $w \in \Sigma^*$, if $Exp(s)\backslash w \neq \emptyset$ then $\mathcal{M}|_w, s \models \varphi$ iff $\mathcal{M}', s \models tr_w(\varphi)$. The proof goes similarly as earlier. $\square$

Note that the single-agent and multi-agent word fragment of POL^- is a syntactic extension of propositional logic and the multi-agent epistemic logic respectively, which are NP-Hard and PSPACE-Hard respectively. From the fact that the satisfiability problem of single agent and the multi-agent fragments of PAL is in NP and PSPACE respectively, we have the following corollaries of Proposition 17.

Corollary 18. *The satisfiability problem of the single-agent word fragment of* POL^- *is* NP*-Complete.*

Corollary 19. *The satisfiability problem of the multi-agent Word fragment of* POL^- *is* PSPACE*-Complete.*

7 Related Work

The complexity of Dynamic Epistemic Logic with action models and non-deterministic choice of actions is NEXPTIME-Complete too (Aucher and Schwarzentruber 2013) and their proof is similar to the one of Theorem 11.

The tableau method described for POL^- uses a general technique where terms contain the observations/announcements/actions played so far. This technique was already used for PAL (Balbiani et al. 2010), DEL (Aucher and Schwarzentruber 2013), and for a non-normal variant of PAL (Ma et al. 2015).

Decidability of (single-agent) epistemic propositional dynamic logic (EPDL) with Perfect Recall (PR) and No Miracles (NM) is addressed in (Li 2018). Although PR and NM are validities in POL^-, there are differences to consider even in single agent. Firstly, in an EPDL model, a possible state can execute a program a and can non-deterministically transition to a state among multiple states, whereas in POL^-, if a state survives after observation a, it gives rise to the same state except the Exp function gets residued. Also, in EPDL, after execution of a program, the state changes hence the propositional valuation in the state changes, whereas in POL^-, the state *survives* after a certain observation and hence the propositional valuation remains the same.

Whereas in POL^-, observations update the model, there are other lines of work in which specifying what agents observe define the epistemic relations in the underlying Kripke model (Charrier et al. 2016) (typically, two states are equivalent for some agent i if agent i observes the same facts in the two states).

8 Perspectives

This work paves the way to an interesting technical open question in modal logic: the connection between POL^- and product modal logics. Single-agent POL^- is close to the product modal logic $S5 \times K$, the logic where models are Cartesian products of an S5-model and a K-model. Indeed, the first component corresponds to the epistemic modality $\hat{K}_i$ while the second component corresponds to observation modalities $\langle\pi\rangle$. There are however two important differences. First, in POL^-, valuations do not change when observations are made. Second, the modality $\langle\pi\rangle$ is of branching at most exponential in π while modalities in K-models do not have branching limitations. We conjecture that the two limitations can be circumvented but it requires some care when applying the finite model property of product modal logic $S5 \times K$. If this connection works, it would be a way to prove NEXPTIME-Completeness of star-free single-agent POL^-.

Recall that POL^- is close to PAL with propositional announcements only (see Proposition 17). We conjecture some connections between POL^- and arbitrary PAL (French and van Ditmarsch 2008), and more precisely with Boolean arbitrary public announcement logic (van Ditmarsch and French 2022). Indeed, the non-deterministic choice $+$ enables to check the existence of some observation to make (for instance, $\langle(a+b)^{10}\rangle\varphi$ checks for the existence of a 10-length word to observe), which is similar to checking the existence of some Boolean announcement.

The next perspective is also to tackle POL with Kleene-star in the language. This study may rely on techniques used in epistemic temporal logics. PAL with Kleene-star is undecidable (Miller and Moss 2005). Again, the undecidability proof relies on modal announcements. Since POL is close to Boolean announcements, this is a hope for POL to be decidable. The idea would be to exploit the link between dynamic epistemic logics and temporal logics (van Ditmarsch, van der Hoek, and Ruan 2013), and rely on techniques developed for tackling the satisfiability problem in epistemic temporal logics (Halpern and Vardi 1989).

References

Aucher, G., and Schwarzentruber, F. 2013. On the complexity of dynamic epistemic logic. In Schipper, B. C., ed., *Proceedings of the 14th Conference on Theoretical Aspects of Rationality and Knowledge (TARK 2013), Chennai, India, January 7-9, 2013*.

Balbiani, P.; van Ditmarsch, H.; Herzig, A.; and de Lima, T. 2010. Tableaux for public announcement logic. *J. Log. Comput.* 20(1):55–76.

Chakraborty, S.; Ghosh, A.; Ghosh, S.; and Schwarzentruber, F. 2022. On verifying expectations and observations of intelligent agents. In Raedt, L. D., ed., *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI 2022, Vienna, Austria, 23-29 July 2022*, 2568–2574. ijcai.org.

Chakraborty, S.; Ghosh, A.; Ghosh, S.; and Schwarzentruber, F. 2023. On simple expectations and observations of intelligent agents: A complexity study. *CoRR* abs/2306.02769.

Charrier, T.; Herzig, A.; Lorini, E.; Maffre, F.; and Schwarzentruber, F. 2016. Building epistemic logic from observations and public announcements. In Baral, C.; Delgrande, J. P.; and Wolter, F., eds., *Principles of Knowledge Representation and Reasoning: Proceedings of the Fifteenth International Conference, KR 2016, Cape Town, South Africa, April 25-29, 2016*, 268–277. AAAI Press.

Fagin, R.; Halpern, J. Y.; Moses, Y.; and Vardi, M. Y. 1995. *Reasoning about Knowledge*. Cambridge, MA, USA: MIT Press.

French, T., and van Ditmarsch, H. P. 2008. Undecidability for arbitrary public announcement logic. In *Advances in Modal Logic*, 23–42.

Halpern, J. Y., and Moses, Y. 1992. A guide to completeness and complexity for modal logics of knowledge and belief. *Artif. Intell.* 54(2):319–379.

Halpern, J. Y., and Vardi, M. Y. 1989. The complexity of reasoning about knowledge and time. i. lower bounds. *J. Comput. Syst. Sci.* 38(1):195–237.

Harel, D.; Tiuryn, J.; and Kozen, D. 2000. *Dynamic Logic*. Cambridge, MA, USA: MIT Press.

Li, Y. 2018. Tableaux for a combination of propositional dynamic logic and epistemic logic with interactions. *J. Log. Comput.* 28(2):451–473.

Lutz, C. 2006. Complexity and succinctness of public announcement logic. In Nakashima, H.; Wellman, M. P.; Weiss, G.; and Stone, P., eds., *5th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2006), Hakodate, Japan, May 8-12, 2006*, 137–143. ACM.

Ma, M.; Sano, K.; Schwarzentruber, F.; and Velázquez-Quesada, F. R. 2015. Tableaux for non-normal public announcement logic. In Banerjee, M., and Krishna, S. N., eds., *Logic and Its Applications - 6th Indian Conference, ICLA 2015, Mumbai, India, January 8-10, 2015. Proceedings*, volume 8923 of *Lecture Notes in Computer Science*, 132–145. Springer.

Miller, J. S., and Moss, L. S. 2005. The undecidability of iterated modal relativization. *Stud Logica* 79(3):373–407.

Plaza, J. 2007. Logics of public communications. *Synth.* 158(2):165–179.

van Ditmarsch, H., and French, T. 2022. Quantifying over boolean announcements. *Log. Methods Comput. Sci.* 18(1).

van Ditmarsch, H.; Ghosh, S.; Verbrugge, R.; and Wang, Y. 2014. Hidden protocols: Modifying our expectations in an evolving world. *Artificial Intelligence* 208:18–40.

van Ditmarsch, H.; van der Hoek, W.; and Kooi, B. 2008. *Dynamic Epistemic Logic*, volume 337 of *Synthese Library*. Netherlands: Springer.

van Ditmarsch, H.; van der Hoek, W.; and Ruan, J. 2013. Connecting dynamic epistemic and temporal epistemic logics. *Log. J. IGPL* 21(3):380–403.

van Emde Boas, P. 2019. The convenience of tilings. In *Complexity, Logic, and Recursion Theory*. CRC Press. 331–363.

Wang, Y. 2011. Reasoning about protocol change and knowledge. In Banerjee, M., and Seth, A., eds., *Logic and Its Applications - 4th Indian Conference, ICLA 2011, Delhi, India, January 5-11, 2011. Proceedings*, volume 6521 of *Lecture Notes in Computer Science*, 189–203. Springer.

Wang, Y. 2018. Beyond knowing that: A new generation of epistemic logics. In *Jaakko Hintikka on Knowledge and Game-Theoretical Semantics*, volume 12 of *Outstanding Contributions to Logic*. Springer. 499–533.