Proceedings of the Thirty-Third International Conference on Automated Planning and Scheduling (ICAPS 2023)

Planning in Multi-Agent Domains with Untruthful Announcements

Loc Pham, Tran Cao Son, Enrico Pontelli

New Mexico State University, Las Cruces, NM 88003, USA
{lpham,tson,epontell}@cs.nmsu.edu

Abstract

Earlier epistemic planning systems for multi-agent domains generate plans that contain various types of actions such as ontic, sensing, or announcement actions. However, none of these systems consider *untruthful announcements*, i.e., none can generate plans that contain a lying or a misleading announcement. In this paper, we present a novel epistemic planner, called EFP3.0, for multi-agent domains with untruthful announcements. The planner is similar to the systems EFP or EFP2.0 in that it is a forward-search planner and can deal with unlimited nested beliefs and common knowledge by employing a Kripke based state representation and implementing an update model based transition function. Different from EFP, EFP3.0 employs a specification language that uses *edge-conditioned update models* for reasoning about effects of actions in multi-agent domains. We describe the basics of EFP3.0 and conduct experimental evaluations of the system against state-of-the-art epistemic planners. We discuss potential improvements that could be useful for scalability and efficiency of the system.

Introduction

Announcements are a special type of actions that exists only in multi-agent domains. An announcement is the act of communicating a piece of information, often described by a formula, from an agent (or a group of agents) to another agent (or a group of agents), so that the latter believes in the announced formula. When the announcers of the formula believe in it, the announcement is said to be *truthful.* We typically divide *untruthful* announcements in two groups. The announcement is a *lying* announcement if the announcers believe in the negation of the announced formula, i.e., they attempt to convey a false information. The announcement is a *misleading* announcement if the announcers do not know the truth value of the announced formula. Contrary to a truthful announcement, that helps reduce uncertainties for the listeners, a successful untruthful announcement will either induce a false belief or increase uncertainties in the listeners' beliefs. Modeling and reasoning about untruthful announcements has been an intense research topic for philosophers and logician (Hintikka 1962; Baltag and Moss 2004; Fallis 2020; van Ditmarsch, van der Hoek, and Kooi 2007; van Ditmarsch et al. 2012; van Ditmarsch, Hendriks, and Verbrugge 2020; van Ditmarsch 2014). The discussion in the special issue on *"Lying in Logic, Language, and Cognition"* by (van Ditmarsch, Hendriks, and Verbrugge 2020) or a recent paper (Fallis 2020) highlights the fact that untruthful announcements are an important part of human behavior and communication. It has also been pointed out that, in certain situations, making an untruthful announcement is the only way for an agent to achieve their goal (Zlotkin and Rosenschein 1991). In other words, executing a plan with some untruthful announcements is sometimes necessary. It is therefore important for an epistemic planner in multi-agent domains to deal with untruthful announcements and generate plans with such actions.

Planning with untruthful announcements belongs to the area of epistemic planning in multi-agent domains, which has attracted a lot of attention by the automated planning community in recent years. Theoretical properties (e.g., un/decidability of the planning task) have been investigated in (Aucher and Bolander 2013; Bolander, Jensen, and Schwarzentruber 2015; Bolander et al. 2020; Charrier, Maubert, and Schwarzentruber 2016). Studies of epistemic planning and several epistemic planners with different capabilities can be found in (Bolander and Andersen 2011; Crosby, Jonsson, and Rovatsos 2014; Engesser et al. 2017; van der Hoek and Wooldridge 2002; Wan, Fang, and Liu 2021; Löwe, Pacuit, and Witzel 2011; Muise et al. 2021; Kominis and Geffner 2015, 2017; Wan et al. 2015; Le et al. 2018; Fabiano et al. 2020). With the exception of the two epistemic planners EFP (Le et al. 2018) and EFP2.0 (Fabiano et al. 2020), earlier search-based epistemic planners do not work with unlimited nested beliefs. None of them, however, deal with untruthful announcements. To the best of our knowledge, none of the existing epistemic planners is able to solve the following variation of the coin in the box domain from (Baral et al. 2022):

Example 1 (A Variation of The Coin in the Box Domain) *Three agents A, B, and C are in a room. In the middle of the room there is a box containing a coin. It is common knowledge that:*

- *Nobody knows which face of the coin is showing;*
- *The box is locked and one needs a key to open it; agent A is the only one with such key;*

- *To determine the face of the coin, one can peek into the box if the box is opened;*
- *An agent, observing another agent peeking into the box, will be able to conclude that the agent who peeked knows which face of the coin is showing—but without knowing themselves the state of the coin;*
- *Distracting an agent causes them to not look at the box;*
- *Signaling an agent causes such agent to look at the box;*
- *Announcing the state of the coin will cause anyone who is looking (and does not have conflicting beliefs with that announcement) to believe in the announcement;*
- *A is looking, while B and C are not looking at the box.*

We assume that the coin lies tails up. Agent A wishes to know which face of the coin is showing and A would like both B and C to believe in the opposite. It is easy to see that A could achieve such goal by: **(a)** *signaling B to look at the box;* **(b)** *signaling C to look at the box;* **(c)** *opening the box;* **(d)** *peeking into the box; and* **(e)** *lying about the value of the coin (announcing heads in this case).*

None of the existing epistemic planning systems can solve the above problem, since lying (A tells B and C a false information) has not been considered in the underlying specification language of these systems. Building on recent research in action languages that allow untruthful announcements (Pham, Son, and Pontelli 2022) and the search based planners EFP, we develop a novel epistemic planner, called EFP3.0, for multi-agent domains with untruthful announcements. We note that due to the difference between the transition functions of the specification languages used by EFP and EFP3.0, the implementation of EFP3.0 is significantly different from the one used by EFP. More on the differences of their implementations can be found in the discussion section.

The main contributions of this paper are: **(i)** a novel epistemic planning system for multi-agent domains, EFP3.0, that can generate plans containing untruthful announcements and **(ii)** an experimental evaluation showing that EFP3.0 is competitive with state-of-the-art epistemic planners even though it employs $m\mathcal{A}^*$, a richer specification language, which uses a transition function based on edge-conditioned update models. This requires the implementation of **(iii)** a new parser for $m\mathcal{A}^*$ domains and **(iv)** a module for computing the transition function of $m\mathcal{A}^*$.

The paper is organized as follows. The next section reviews the background of epistemic planning and the specification language that considers lying/misleading announcements. Next, we provide a description of the architecture and components of EFP3.0. The paper then presents an experimental evaluation of EFP3.0, a comparison with some other planning systems and a discussion of their strengths and weaknesses. The source code of EFP3.0 and input files used for the experimental studies are available at https://github.com/phhuuloc/EFP3.0.

Background

Epistemic Planning

A *multi-agent* domain $\langle \mathcal{AG}, \mathcal{F} \rangle$ includes a finite and non-empty set of agents $\mathcal{AG}$ and a set of propositional variables (*fluents*) $\mathcal{F}$ encoding properties of the world. *Belief formulae* over $\langle \mathcal{AG}, \mathcal{F} \rangle$ are defined by the BNF:
$\varphi ::= p \mid \neg\varphi \mid (\varphi \wedge \varphi) \mid (\varphi \vee \varphi) \mid \mathbf{B}_i\varphi \mid \mathbf{E}_\alpha\varphi \mid \mathbf{C}_\alpha\varphi$
where $p \in \mathcal{F}$ is a fluent, $i \in \mathcal{AG}$, and $\emptyset \neq \alpha \subseteq \mathcal{AG}$. A *fluent formula* is a belief formula which does not contain any occurrence of $\mathbf{B}_i$, $\mathbf{E}_\alpha$, and $\mathbf{C}_\alpha$. Formulae of the form $\mathbf{E}_\alpha\varphi$ and $\mathbf{C}_\alpha\varphi$ are referred to as *group formulae*. Moreover, when $\alpha = \mathcal{AG}$, we simply write $\mathbf{E}\varphi$ and $\mathbf{C}\varphi$ to denote $\mathbf{E}_\alpha\varphi$ and $\mathbf{C}_\alpha\varphi$, respectively. $\mathcal{L}(\mathcal{AG}, \mathcal{F})$ denotes the set of belief formulae over $\langle \mathcal{AG}, \mathcal{F} \rangle$.

The semantic for languages in epistemic logic is usually based on *pointed Kripke models*. In this paper, we follow the presentation by (Fagin et al. 1995).

Definition 1 (Kripke model) *Given a set of agents $\mathcal{AG}$, a* Kripke model *M is a tuple $\langle S, \pi, \{\mathcal{B}_i\}_{i \in \mathcal{AG}} \rangle$, where:*

- *S is a set of worlds (denoted by $M[S]$);*
- *$\pi : S \mapsto 2^{\mathcal{F}}$ is a function that associates an interpretation of $\mathcal{F}$ to each element of S (denoted by $M[\pi]$);*
- *For $i \in \mathcal{AG}$, $\mathcal{B}_i \subseteq S \times S$ is a binary relation over S (denoted by $M[i]$).*

Definition 2 (Pointed Kripke model) *A* pointed Kripke model *is a pair (M, s) where M is a Kripke model as defined above, and $s \in M[S]$, where s points at the real world.*

The satisfaction relation $\models$ between belief formulae and a state (M, s) is defined as follows:

1. $(M, s) \models p$ if p is a fluent and $M[\pi](s)(p)$ is true;
2. $(M, s) \models \neg\varphi$ if $(M, s) \not\models \varphi$;
3. $(M, s) \models \varphi_1 \wedge \varphi_2$ if $(M, s) \models \varphi_1$ and $(M, s) \models \varphi_2$;
4. $(M, s) \models \varphi_1 \vee \varphi_2$ if $(M, s) \models \varphi_1$ or $(M, s) \models \varphi_2$;
5. $(M, s) \models \mathbf{B}_i\varphi$ if $\forall t.[(s, t) \in \mathcal{B}_i \Rightarrow (M, t) \models \varphi]$;
6. $(M, s) \models \mathbf{E}_\alpha\varphi$ if $(M, s) \models \mathbf{B}_i\varphi$ for every $i \in \alpha$;
7. $(M, s) \models \mathbf{C}_\alpha\varphi$ if $(M, s) \models \mathbf{E}^k_\alpha\varphi$ for every $k \geq 0$, where
   - $\mathbf{E}^0_\alpha\varphi = \varphi$ and
   - $\mathbf{E}^{k+1}_\alpha = \mathbf{E}_\alpha(\mathbf{E}^k_\alpha\varphi)$.

An *epistemic state* (or *e-state*) is a pair (M, W_d) where M is an Kripke model and $W_d \subseteq M[S]$. A truth value of a formula φ with respect to an e-state (M, W_d) is defined by

$$(M, W_d) \models \varphi \qquad \text{iff} \qquad \forall s \in W_d.[(M, s) \models \varphi]$$

Another important notion in epistemic planning is that of *update model* (also called *event models* or *action model*) (Baltag and Moss 2004; Baltag, Moss, and Solecki 1998; Baltag and Moss 2004; van Benthem, van Eijck, and Kooi 2006), that models changes when an action has been executed in an e-state. As we are utilizing a specification language that employs edge-conditioned update models (Bolander 2018), we review this notion next. Let us start with some preliminary notations. An $\mathcal{L}(\mathcal{AG}, \mathcal{F})$-substitution is a set $\{p_1 \rightarrow \varphi_1, \ldots, p_k \rightarrow \varphi_k\}$, where each p_i is a distinct proposition in $\mathcal{F}$ and each $\varphi_i \in \mathcal{L}(\mathcal{AG}, \mathcal{F})$. We will implicitly assume that for each $p \in \mathcal{F} \setminus \{p_1, \ldots, p_k\}$, the substitution contains $p \rightarrow p$. $SUB_{\mathcal{AG},\mathcal{F}}$ denotes the set of all $\mathcal{L}(\mathcal{AG}, \mathcal{F})$-substitutions.

Definition 3 (Update Model) *Given a set of agents $\mathcal{AG}$, an* update model *$\mathbf{\Sigma}$ is a tuple $\langle E, \{R_i\}_{i \in \mathcal{AG}}, pre, sub \rangle$, where:*

- *E is a set, whose elements are called* events*;*

- $R_i \subseteq E \times \mathcal{L}(\mathcal{AG}, \mathcal{F}) \times E$ *is the accessibility relation of agent* i *between events;*
- $pre : E \rightarrow \mathcal{L}(\mathcal{AG}, \mathcal{F})$*is a function mapping each event* $e \in E$ *to a formula in* $\mathcal{L}(\mathcal{AG}, \mathcal{F})$*; and*
- $sub : E \rightarrow SUB_{\mathcal{AG},\mathcal{F}}$ *is a function mapping each event* $e \in E$ *to a substitution in* $SUB_{\mathcal{AG},\mathcal{F}}$*.*

A pair $(\mathcal{E}, E_d)$*, consisting of an update model* $\mathcal{E} = (E, \{R_i\}_{i\in\mathcal{AG}}, pre, sub)$ *and a non-empty set of designated events* $E_d \subseteq E$*, is called an* epistemic action.

Given an epistemic action $(\mathcal{E}, E_d)$ and an epistemic state (M, W_d), we say that $(\mathcal{E}, E_d)$ is executable in (M, W_d) if, for each $s \in W_d$, there exists at least one $e \in E_d$ such that $(M, s) \models pre(e)$. The execution of $(\mathcal{E}, E_d)$ in (M, W_d) results in an epistemic state $(M, W_d) \otimes (\mathcal{E}, E_d) = (M', W'_d)$:

- $M'[S] = \{(s, e) \in M[S] \times E \mid (M, s) \models pre(e)\}$;
- $((s, e), (s', e')) \in M'[i]$ iff $(s, e), (s', e') \in M'[S]$, $(s, s') \in M[i]$, $(e, \gamma, e') \in R_i$ and $(M, s) \models \gamma$;
- For each $(s, e) \in W'$ and $p \in \mathcal{F}$, $\pi'((s, e))(p)$ is true iff $p \rightarrow \varphi \in sub(e)$ and $(M, s) \models \varphi$;
- $W'_d = \{(s, e) \in M'[S] \mid s \in W_d \, and \, e \in E_d\}$.

An *epistemic planning domain* is defined as a state-transition system $\Sigma = (S, A, \gamma)$, where S is a set of epistemic states of $\mathcal{L}(\mathcal{F}, \mathcal{AG})$, A is a finite set of epistemic actions of $\mathcal{L}(\mathcal{F}, \mathcal{AG})$, and $\gamma(s, a) = s \otimes a$ if $s \otimes a$ is defined and $\otimes$ is the above operation. An *epistemic planning problem* is a triple (Σ, s_0, ϕ_g) where $\Sigma = (S, A, \gamma)$ is an epistemic planning domain on $(\mathcal{F}, \mathcal{AG})$, $s_0 \in S$ is the initial state, and ϕ_g is a formula in $\mathcal{L}(\mathcal{F}, \mathcal{AG})$. An action sequence $a_1, \ldots, a_n$ where $s_0 \otimes a_1 \otimes \ldots \otimes a_n \models \phi_g$ is a *solution* of the problem.

Action Language $m\mathcal{A}^*$ with Untruthful Announcements

EFP employs the action language $m\mathcal{A}$, an earlier version of $m\mathcal{A}^*$ (Baral et al. 2022), as its specification language which does not consider untruthful announcements. In addition, EFP uses a finitary theory (Son et al. 2014) to specify the initial state of an epistemic planning problem.

In this paper, we employ the same method used in EFP for specifying the initial state and an extension of $m\mathcal{A}^*$ that can deal with lying and misleading announcements (Pham, Son, and Pontelli 2022) for planning problem specification. For simplicity of the presentation, we will still refer to this language as $m\mathcal{A}^*$. Technically, the key difference between this extension and the version in (Baral et al. 2022) lies in the use of edge-conditioned update model in the definition of the transition function. Its advantages have been discussed in (Pham et al. 2022). In $m\mathcal{A}^*$, actions are categorized into three different classes:

1. World-altering actions (ontic actions) are actions whose execution changes the real state of the world;
2. Sensing actions are actions whose execution refines agents' belief about the world; and
3. Announcement actions are actions whose execution affects the beliefs of other agents through communication.

An *action instance* a is of the form $a\langle\alpha\rangle$, where a is an action name and α is a set of agents. Intuitively, $a\langle\alpha\rangle$ says that the set of agents α executes a. An action domain over a multi-agent domain $(\mathcal{AG}, \mathcal{F})$ and a set of action instances $\mathcal{AI}$ contains statements of the following forms:

a **executable** ψ (1)
a **causes** ℓ **if** φ (2)
a **determines** φ (3)
a **announces** φ (4)
z **observes** a **if** δ_z (5)
z **aware_of** a **if** θ_z (6)
initially ψ (7)

where ℓ is a fluent literal (a fluent $f \in \mathcal{F}$ or its negation $\neg f$), ψ is a belief formula, φ, δ_z and θ_z are fluent formulae, $\mathsf{a} \in \mathcal{AI}$, and $z \in \mathcal{AG}$. (1) encodes the executability condition of a and ψ is referred as the *precondition of* a. A statement of the form (2) describes the effect of the ontic action a, i.e., if ψ is true then ℓ will be true after the execution of a. (3) says that the agents who execute a to learn the value of the formula φ. (4) encodes an *announcement* action, whose owner announces that φ is true. Statements of the forms (5)–(6) encode the observability of agents given an occurrence of a. (5) indicates that agent z is a full observer of a if δ_z holds. (6) states that agent z is a partial observer of a if θ_z holds. z, a, and δ_z (resp. θ_z) are referred to as the observing agent, the action instance, and the condition of (5) (resp. (6)). We assume that the sets of ontic actions, sensing actions, and announcement actions are pairwise disjoint. Furthermore, for every pair of a and z, if z and a occur in a statement of the form (5), then they will not occur in any statement of the form (6) and vice versa. Statements of the form (7) indicate that ψ is true in the initial state.

Another important concept in multi-agent domains is *action observability*. The execution of an action may or may not change the belief of an agent depending on whether she is aware of the action's occurrence or not. In $m\mathcal{A}^*$, the notion of *frame of reference* determines the observability of an action occurrence by a set of agents. Formally, the *frame of reference* for the execution of a in (M, s) is a tuple $(F_D(\mathsf{a}, M, s), P_D(\mathsf{a}, M, s), O_D(\mathsf{a}, M, s))$ where:

- $F_D(\mathsf{a}, M, s) = \{x \in \mathcal{AG} \mid [x$ **observes** a **if** $\delta_x] \in D$ such that $(M, s) \models \delta_x\}$
- $P_D(\mathsf{a}, M, s) = \{x \in \mathcal{AG} \mid [x$ **aware_of** a **if** $\theta_x] \in D$ such that $(M, s) \models \theta_x\}$
- $O_D(\mathsf{a}, M, s) = \mathcal{AG} \setminus (F_D(\mathsf{a}, M, s) \cup P_D(\mathsf{a}, M, s))$

Intuitively, $F_D(\mathsf{a}, M, s)$ (resp. $P_D(\mathsf{a}, M, s)$ and $O_D(\mathsf{a}, M, s)$) are the agents that are fully observant (resp. partially observant and oblivious/other) of the execution of a in the state (M, s). $m\mathcal{A}^*$ assumes that the sets $F_D(\mathsf{a}, M, s)$, $P_D(\mathsf{a}, M, s)$, and $O_D(\mathsf{a}, M, s)$ are pairwise disjoint. We will often write F, P, and O to denote these sets, respectively.

We next review the new semantics of $m\mathcal{A}^*$ that employs edge-conditioned update models by describing the update models for five types of actions: ontic, sensing, truthful announcements, lying announcements and misleading announcements.

Definition 4 (Ontic Actions) *Let* a *be an ontic action with the precondition* ψ*. The update model for* a*, denoted by*

$\omega(\mathsf{a})$, *is defined by* $\langle E, \{R_i\}_{i\in\mathcal{AG}}, pre, sub\rangle$ *where:*

- $E = \{\sigma, \epsilon\}$*;*
- R_i=$\{(\sigma,\delta_i,\sigma),(\sigma,\neg\delta_i,\epsilon),(\epsilon,\top,\epsilon)\}$ *where "i* **observes** a **if** δ_i*" belongs to D;*
- $pre(\sigma) = \psi$ *and* $pre(\epsilon) = \top$*; and*
- $sub(\epsilon) = \emptyset$ *and* $sub(\sigma) = \{p \rightarrow \Psi^+(p,\mathsf{a}) \vee (p \wedge \neg\Psi^-(p,\mathsf{a})) \mid p \in \mathcal{F}\}$, *where:*

  $\Psi^+(p,\mathsf{a}) = \bigvee\{\varphi \mid [\mathsf{a}$ **causes** p **if** $\varphi] \in D\}$ *and*

  $\Psi^-(p,\mathsf{a}) = \bigvee\{\varphi \mid [\mathsf{a}$ **causes** $\neg p$ **if** $\varphi] \in D\}$.

Figure 1 illustrates the edge-conditioned update model for an ontic action. In the figure, square nodes represent events and a link between two nodes x and v with a label $i \in S : \theta$ indicates that for $i \in S$, (x, θ, v) belongs to R_i.

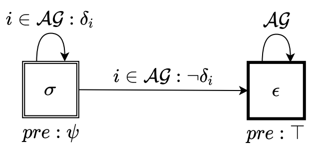

Figure 1: Ontic Action Edge-Conditioned Update Model

Definition 5 (Sensing and Truthful Announcement Actions) *Let* a *be a sensing action that senses φ or a truthful announcement action that announces φ with the precondition ψ. The update model for* a*, denoted by $\omega(\mathsf{a})$, is defined by* $\langle E, \{R_i\}_{i\in\mathcal{AG}}, pre, sub\rangle$ *where:*

- $E = \{\sigma, \tau, \epsilon\}$*;*
- $R_i = \{(\sigma, \delta_i \vee \theta_i, \sigma), (\tau, \delta_i \vee \theta_i, \tau), (\sigma, \neg\delta_i \wedge \theta_i, \tau), (\tau, \neg\delta_i \wedge \theta_i, \sigma), (\sigma, \neg\delta_i \wedge \neg\theta_i, \epsilon), (\tau, \neg\delta_i \wedge \neg\theta_i, \epsilon), (\epsilon, \top, \epsilon)\}$ *where "i* **observes** a **if** δ_i*" and "i* **aware_of** a **if** θ_i*" belong to D;*
- $pre(\sigma) = \psi \wedge \varphi$*,* $pre(\tau) = \psi \wedge \neg\varphi$ *and* $pre(\epsilon) = \top$*;*
- $sub(x) = \emptyset$ *for each* $x \in E$.

Figure 2 and Figure 3 show the edge-conditioned update models for sensing actions and truthful announcement actions, respectively.

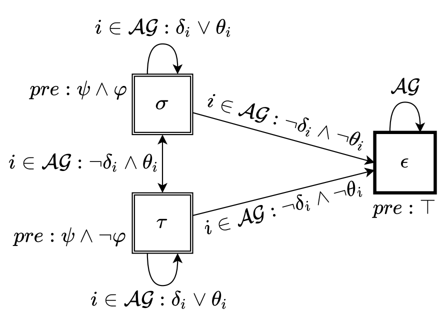

Figure 2: Sensing Action Edge-Conditioned Update Model

Observe that with edge-conditioned update models, the new semantic can take into consideration the observability of agents at the local level (i.e, at each possible world). A

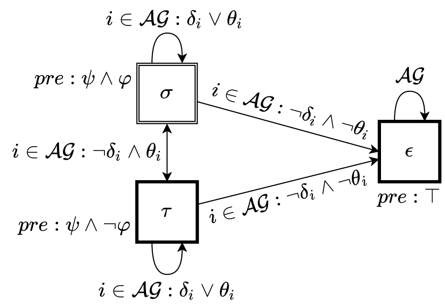

Figure 3: Edge-Conditioned Update Model for Truthful Announcement Action

consequence of this treatment is that the belief of an agent i, who is a full observer, about the belief of another agent j with respect to the action occurrence will change in accordance to the belief of i about j in the true state of the world. This property does not hold in previous planners such as EFP1.0.

Definition 6 (Lying Announcement Actions) *Let* $\mathsf{a} = a\langle\alpha\rangle$ *be an announcement of formula φ and (M, s) be a state such that $(M, s) \models \mathbf{B}_\alpha\neg\varphi$. The update model for* a *in (M, s), $\omega(\mathsf{a}, (M, s))$, is defined by* $\langle E, \{R_i\}_{i\in\mathcal{AG}}, pre, sub\rangle$ *where:*

- $E = \{\sigma, \zeta, \tau, \chi, \mu, \epsilon\}$*;*
- $R_i = \{\ (\sigma, i \in F(\mathsf{a}, M, s) : i \in \alpha \vee \mathbf{B}_i\mathbf{B}_\alpha\neg\varphi, \sigma)$, $(\sigma, i \in F(\mathsf{a}, M, s) : \mathbf{B}_i\neg\varphi \vee \mathbf{B}_i\mathbf{B}_\alpha\neg\varphi, \zeta)$, $(\sigma, i \in F(\mathsf{a}, M, s) : \neg\mathbf{B}_i\neg\varphi, \tau)$, $(\sigma, i \in P(\mathsf{a}, M, s), \chi)$, $(\sigma, i \in P(\mathsf{a}, M, s), \mu)$, $(\sigma, i \in O(\mathsf{a}, M, s), \epsilon)$, $(\zeta, i \in F(\mathsf{a}, M, s) : \mathbf{B}_i\neg\varphi \vee \mathbf{B}_i\mathbf{B}_\alpha\neg\varphi, \zeta)$, $(\zeta, i \in F(\mathsf{a}, M, s) : \neg\mathbf{B}_i\neg\varphi, \tau)$, $(\zeta, i \in P(\mathsf{a}, M, s), \chi)$, $(\zeta, i \in P(\mathsf{a}, M, s), \mu)$, $(\zeta, i \in O(\mathsf{a}, M, s), \epsilon)$, $(\tau, i \in F(\mathsf{a}, M, s) : \mathbf{B}_i\neg\varphi \vee \mathbf{B}_i\mathbf{B}_\alpha\neg\varphi, \zeta)$, $(\tau, i \in F(\mathsf{a}, M, s), \tau)$, $(\tau, i \in P(\mathsf{a}, M, s), \chi)$, $(\tau, i \in P(\mathsf{a}, M, s), \mu)$, $(\tau, i \in O(\mathsf{a}, M, s), \epsilon)$, $(\chi, i{\in}F(\mathsf{a}, M, s) \cup P(\mathsf{a}, M, s), \chi)$, $(\chi, i{\in}P(\mathsf{a}, M, s) \vee i \in F(\mathsf{a}, M, s) : \mathbf{B}_i\varphi, \mu)$, $(\chi, i{\in}O(\mathsf{a}, M, s), \epsilon)$, $(\mu, i{\in}F(\mathsf{a}, M, s) \cup P(\mathsf{a}, M, s), \mu)$, $(\mu, i{\in}P(\mathsf{a}, M, s) \vee i \in F(\mathsf{a}, M, s) : \mathbf{B}_i\neg\varphi, \chi)$, $(\mu, i{\in}O(\mathsf{a}, M, s), \epsilon)$, $(\epsilon, i \in \mathcal{AG}, \epsilon)\}$
- *The preconditions pre are:*
  - $pre(\sigma) = \mathbf{B}_\alpha\neg\varphi$*;*
  - $pre(\chi) = \neg\varphi$*;*
  - $pre(\zeta) = \neg\varphi$*;*
  - $pre(\mu) = \varphi$*;*
  - $pre(\tau) = \varphi$*;*
  - $pre(\epsilon) = \top$*;*
- $sub(x) = \emptyset$ *for each* $x \in E$.

Definition 7 (Misleading Announcement Actions) *Let* $\mathsf{a} = a\langle\alpha\rangle$ *be an announcement with the announced formula φ and (M, s) be a state such that $(M, s) \models \neg(\mathbf{B}_i\varphi \vee \mathbf{B}_i\neg\varphi)$. The update model for* a *in (M, s), $\omega(\mathsf{a}, (M, s))$, is defined by* $\langle E, \{R_i\}_{i\in\mathcal{AG}}, pre, sub\rangle$ *as follow:*

- $E = \{\sigma, \zeta, \tau, \chi, \mu, \epsilon\}$*;*
- $R_i = \{\ (\sigma, i \in F(\mathsf{a}, M, s) \cup P(\mathsf{a}, M, s) : i \in \alpha \vee \mathbf{B}_i\neg(\mathbf{B}_\alpha\neg\varphi \vee \mathbf{B}_\alpha\varphi), \sigma)$, $(\sigma, i \in$

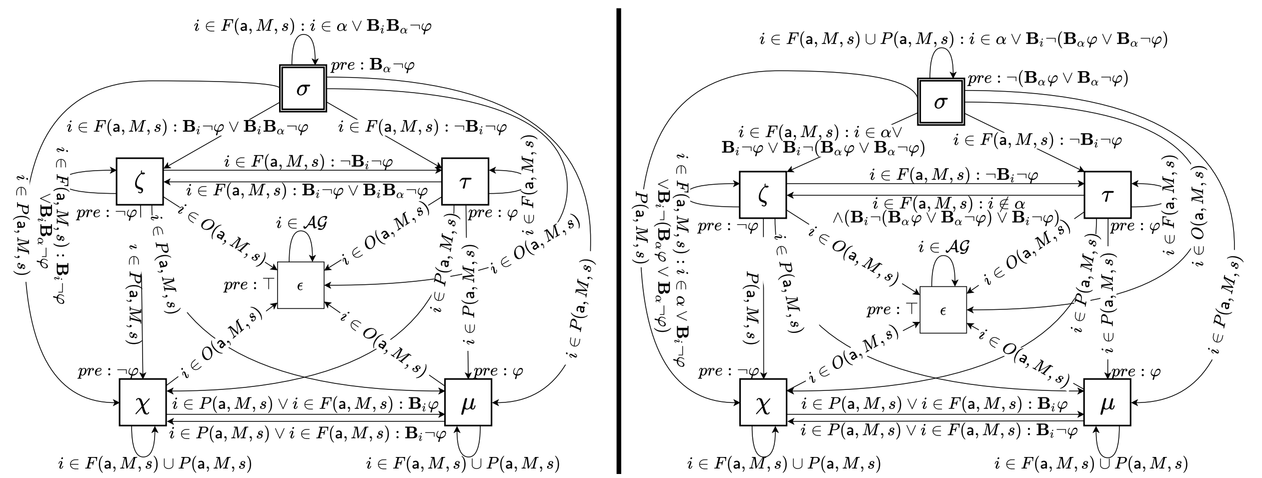

Figure 4: Update model for lying announcement (left) and misleading announcement (right) of φ

$F(\mathsf{a}, M, s) : \mathbf{B}_i\neg\varphi \vee \mathbf{B}_i\neg(\mathbf{B}_\alpha\neg\varphi \vee \mathbf{B}_\alpha\varphi), \zeta)$, $(\sigma, i \in F(\mathsf{a}, M, s) : \neg\mathbf{B}_i\neg\varphi, \tau)$, $(\sigma, i \in P(\mathsf{a}, M, s), \chi)$, $(\sigma, i \in P(\mathsf{a}, M, s), \mu)$, $(\sigma, i \in O(\mathsf{a}, M, s), \epsilon)$, $(\zeta, i \in F(\mathsf{a}, M, s) : \mathbf{B}_i\neg\varphi \vee \mathbf{B}_i\neg(\mathbf{B}_\alpha\neg\varphi \vee \mathbf{B}_\alpha\varphi), \zeta)$, $(\zeta, i \in F(\mathsf{a}, M, s) : \neg\mathbf{B}_i\neg\varphi, \tau)$, $(\zeta, i \in P(\mathsf{a}, M, s), \chi)$, $(\zeta, i \in P(\mathsf{a}, M, s), \mu)$, $(\zeta, i \in O(\mathsf{a}, M, s), \epsilon)$, $(\tau, i \in F(\mathsf{a}, M, s) : i \notin \alpha \wedge (\mathbf{B}_i\neg\varphi \vee \mathbf{B}_i\neg(\mathbf{B}_\alpha\neg\varphi \vee \mathbf{B}_\alpha\varphi)), \zeta)$, $(\tau, i \in F(\mathsf{a}, M, s), \tau)$, $(\tau, i \in P(\mathsf{a}, M, s), \chi)$, $(\tau, i \in P(\mathsf{a}, M, s), \mu)$, $(\tau, i \in O(\mathsf{a}, M, s), \epsilon)$, $(\chi, i {\in} F(\mathsf{a}, M, s) \cup P(\mathsf{a}, M, s), \chi)$, $(\chi, i {\in} P(\mathsf{a}, M, s) \vee i \in F(\mathsf{a}, M, s) : \mathbf{B}_i\varphi, \mu)$, $(\chi, i {\in} O(\mathsf{a}, M, s), \epsilon)$, $(\mu, i {\in} F(\mathsf{a}, M, s) \cup P(\mathsf{a}, M, s), \mu)$, $(\mu, i {\in} P(\mathsf{a}, M, s) \vee i \in F(\mathsf{a}, M, s) : \mathbf{B}_i\neg\varphi, \chi)$, $(\mu, i {\in} O(\mathsf{a}, M, s), \epsilon)$, $(\epsilon, i \in \mathcal{AG}, \epsilon)\}$

- *The preconditions pre are:*
  - $pre(\sigma) = \neg(\mathbf{B}_\alpha\neg\varphi \vee \mathbf{B}_\alpha\varphi)$;
  - $pre(\zeta) = \neg\varphi$;
  - $pre(\tau) = \varphi$;
  - $pre(\chi) = \neg\varphi$;
  - $pre(\mu) = \varphi$;
  - $pre(\epsilon) = \top$;
- $sub(x) = \emptyset$ *for each* $x \in E$.

Figure 4 shows the edge-conditioned update models for lying announcement actions (left) and misleading announcement actions (right).

Observe that the update model of a misleading announcement has the same set of events as that of a lying announcement. The key differences between them are: the precondition of the designated event σ; the additional condition for agents in P in σ; the requirement for an agent i who knows about α's belief is $\mathbf{B}_i\neg(\mathbf{B}_\alpha\neg\varphi \vee \mathbf{B}_\alpha\varphi)$ instead of $\mathbf{B}_i\mathbf{B}_\alpha\neg\varphi$ as in a lying announcement; and the additional $i \notin \alpha$ for the relation from τ to ζ, which ensure that anyone who believes in the misleading announcement also believes that α knows φ too.

The edge-conditioned update models described in Definitions 4-7 are then used to define a transition function Φ_D that represents the transitions between states of an action domain D in the usual way (see, e.g., (Baral et al. 2022)). We omit the definition of Φ_D for brevity. We would like to note that the edge-conditioned update models in Definitions 4–7 satisfy several fundamental properties for reasoning about effects of actions in multi-agent domains, advocated in (Baral et al. 2022), such as:

- if an agent is fully aware of the execution of an action instance then her beliefs will be updated with the effects of such action execution;
- an agent who is only partially aware of the action occurrence will believe that the agents who are fully aware of the action occurrence are certain about the action effects;
- an agent who is oblivious of the action occurrence will be ignorant about its effects.

In addition to the above properties, as discussed in (Pham, Son, and Pontelli 2022), under the assumption that agents will believe in an announcement only when they do not know the truth value of the announced formula and cannot realize that the announcers are lying or misleading then

- if an agent is certain about the truth of a formula, even if it is incorrect in the actual world, or if she realizes that the announcement is untruthful (i.e. she knows that the announcers make a false statement) then she will not change her belief about the formula, regardless of what the announcers say;
- if an agent is uncertain about the truth of a formula and cannot reason that the announcers are untruthful then she will believe what the announcers say.

Due to space limitation, the formal statements of the properties and their proofs are deferred to the Supplementary Document on GitHub.

EFP3.0: An Epistemic Forward Planner for Domains with Untruthful Announcements

In this section, we describe the implementation of EFP3.0 and an experimental evaluation of the planner. The planner is implemented using the code of EFP1.0 (Le et al. 2018) downloaded from https://github.com/tiep/EpistemicPlanning. It therefore inherits the overall structure of EFP1.0 and different modules such as:

- the parser: this is because of the syntax of $m\mathcal{A}$ is similar to $m\mathcal{A}^*$ and requires only some modifications for use in EFP3.0;
- the module for computing the set of initial Kripke structures representing the initial e-state of the problem; and
- the module for checking whether a formula is true in a Krippke structure. This module is used for checking actions' executability and validating goal's satisfaction.

The significant portion of the implementation of EFP3.0 is for the task of computing (*i*) the edge-conditioned update models corresponding to the execution of an actions in a given pointed Kripke structure; and (*ii*) the transition function Φ_D. In addition, we also implement a heuristic in guiding the search. For completeness of the paper, we start with the description of the overall structure of EFP3.0 that is similar to that of EFP1.0.

Overall Architecture

The overall architecture of EFP3.0 is given in Algorithm 1. The key modules of EFP3.0 are: *(i)* a pre-processor; *(ii)* initial e-state computation; and *(iii)* a search engine. A planning problem in $m\mathcal{A}^*$ is specified by a tuple $\langle \mathcal{AG}, \mathcal{F}, D, \phi_g \rangle$ where $\langle \mathcal{AG}, \mathcal{F} \rangle$ is a multi-agent domain, D is an action domain over $\langle \mathcal{AG}, \mathcal{F} \rangle$, and ϕ_g is a formula representing the goal. Given an action domain D, $s_0(D)$ denotes the set of of statements of the form (7) in D. The components are described below.

- *Pre-processor*: this module is responsible for parsing the planning problem description, and setting up the planning domain, which includes the list of agents, the list of actions, the rules for computing frames of reference, and the list of fluents. This module is also responsible for the initialization of necessary data structures (e.g., e-state).
- *Initial e-state computation*: Compute the initial state is not a trivial task in multi-agent epistemic planning when the system relies on a transition function employing Kripke structures. In particular, given a belief formula φ_i, it is possible to generate infinite e-states that respect φ_i. Similar to the previous versions, EFP3.0 implements the algorithm given in (Son et al. 2014) for computing the initial e-state.
- *Search engine*: this module is responsible for computing a solution. We implemented two versions of EFP3.0: a best-first search and a breadth-first search version. For the best-first search version, the number of satisfied subgoals is used. This heuristic is denoted by $h^{maxSubGoal}$.

Experimental Evaluation

We compare our new multi-agent epistemic planner that can deal with lying announcements and incorrect second-order beliefs: EFP3.0, with the two older versions EFP1.0 (presented in (Le et al. 2018)) and EFP2.0 (presented in (Fabiano et al. 2020)). Since EFP1.0 has been extensively compared with other systems in the literature, we did not include a comparison between EFP3.0 with other systems for brevity. All the experiments are performed on a 2.9 GHz Quad-Core Intel Core i5 machine with 16GB of memory.

Algorithm 1: EFP3.0($\langle \mathcal{F}, \mathcal{AG}, D, \phi_g \rangle$)

Input: A planning problem $P = \langle \mathcal{F}, \mathcal{AG}, A, \phi_g \rangle$
Output: A solution for P if exists; failed otherwise
Compute the initial e-state given $s_0(D)$: $(\mathcal{M}_i, W_i)$
Initialize a priority queue $q = [(\{(\mathcal{M}_i, W_i)\}, [])]$
while q is not empty **do**
    $(\Omega, plan) = dequeue(q)$
    If $(\mathcal{M}, W_d) \models \phi_g$ for every $(\mathcal{M}, W_d) \in \Omega$ then return $plan$
    for action a executable in every $(\mathcal{M}, W_d)$ in Ω **do**
        Compute $\Omega' = \bigcup_{(\mathcal{M}, W_d) \in \Omega} \Phi(a, (\mathcal{M}, W_d))$
        Compute heuristics and insert $(\Omega', plan \circ a)$ into q
    end for
end while
return failed

Benchmarks. We evaluate EFP3.0 on benchmarks collected from the literature that are summarized in (Kominis and Geffner 2015; Muise et al. 2021) and have been used in evaluating state-of-the-art epistemic planning systems detailed in (Fabiano et al. 2020; Le et al. 2018). To evaluate the new features of EFP3.0, we create new benchmarks from current ones that consider goals requiring some lying or misleading announcements in the plan (e.g., as in Example 1). In particular, these domains are:

1. *Coin in the Box* (**CB**): $n \geq 3$ agents are in the room where in the middle there is a box containing a coin. None of the agents know whether the coin lies heads or tails up and the box is locked. One agent has the key to open the box. The goals usually consist of some agents knowing whether the coin lies heads or tails up while other agents know that these agents know the status of the coin or are ignorant about this. For the modified versions (**CB-FB**)/(**CB-LA**), the goals usually consist of some agents having false belief about other agents' beliefs; or some agents wanting to deceive others by making a lying announcement (Results in Table 1).
2. *Grapevine* (**Gr**(n,k)): $n \geq 2$ agents are located in $k \geq 2$ rooms. An agent can move freely from one room to its adjacent, and she can share a "secret" with agents in the same room with her. The goals usually require that some agents know certain "secrets" while others do not. We also modify this domain with lying announcements (**Gr-LA**(n,k)), the goals would require that some agents have false belief about certain "secrets" (Results in Table 1).
3. *Selective Communication* (**SC**(n,k)): $n \geq 2$ agents start in one of the $k \geq 2$ rooms in a corridor. An agent can tell (truthfully) some information and all agents in her or neighboring rooms can hear what was told. Every agent is free to move from one room to its adjacent. The goals usually require that some agents know certain information while other agents ignore them. Similarly, we have modified this domain with lying announcements (**SC-LA**(n,k)). The goals require that some agents believe in some false information (Results in Table 2).
4. *Collaboration and Communication* (**CC**(n,m,k)): $n \geq 2$

agents move along a corridor with $k \geq 2$ rooms in which $m \geq 1$ boxes can be located. Whenever an agent enters a room, she can determine if a certain box is in that room. Moreover, agents can communicate (truthfully) information about the boxes' position to other *attentive* agents. The goals consider agents' positions and their beliefs about the boxes. Again, we also have modified this domain with lying announcements (**CC-LA**(n,m,k)). The goals would require agents believe in some false information (Results in Table 3).

L	EFP1	EFP2-k	EFP2-p	EFP3	EFP3-h
CB with $\|\mathcal{AG}\| = 3$, $\|\mathcal{F}\| = 8$, $\|\mathcal{AI}\| = 21$					
2	.03	.03	.003	.03	.03
3	.19	.11	.02	.16	.17
5	*WP*	1.56	.25	2.54	2.81
6	*WP*	24.16	1.58	44.30	53.26
7	162.74	106.77	3.68	144.55	2.23
CB-FB with $\|\mathcal{AG}\| = 3$, $\|\mathcal{F}\| = 8$, $\|\mathcal{AI}\| = 24$					
3	*OM*	*TO*	*TO*	.71	*TO*
4	*OM*	*TO*	*TO*	7.03	*TO*
CB-LA with $\|\mathcal{AG}\| = 3$, $\|\mathcal{F}\| = 8$, $\|\mathcal{AI}\| = 24$					
4	*OM*	*TO*	*TO*	.56	.15
5	*OM*	*TO*	*TO*	4.52	1.00
6	*OM*	*TO*	*TO*	35.73	1.34
Gr(3,2) with $\|\mathcal{AG}\| = 3$, $\|\mathcal{F}\| = 9$, $\|\mathcal{AI}\| = 24$					
2	.46	.39	.12	1.61	.39
3	3.98	3.01	.51	12.53	.54
4	25.55	23.73	3.83	136.37	.49
5	178.60	240.51	23.64	1288.05	.80
Gr(4,2) with $\|\mathcal{AG}\| = 2$, $\|\mathcal{F}\| = 12$, $\|\mathcal{AI}\| = 40$					
2	2.95	6.75	.84	16.26	3.78
3	28.54	86.91	5.73	172.78	5.10
4	276.02	1307.99	42.17	1820.73	4.54
Gr(5,2) with $\|\mathcal{AG}\| = 5$, $\|\mathcal{F}\| = 15$, $\|\mathcal{AI}\| = 60$					
2	31.54	30.57	5.83	852.96	154.23
Gr-LA(3,2) with $\|\mathcal{AG}\| = 3$, $\|\mathcal{F}\| = 9$, $\|\mathcal{AI}\| = 42$					
2	*TO*	*TO*	*TO*	4.15	1.28
3	*TO*	*TO*	*TO*	77.11	1.76
4	*TO*	*TO*	*TO*	2012.80	3.41
Gr-LA(4,2) with $\|\mathcal{AG}\| = 2$, $\|\mathcal{F}\| = 12$, $\|\mathcal{AI}\| = 72$					
2	*TO*	*TO*	*TO*	177.57	41.61

Table 1: Run-time for *Coin in the Box* and *Grapevine*

In detailing the results of the experiments, we use the following notations:

- L to indicate the optimal length of the plan;
- $|\mathcal{AG}|$, $|\mathcal{F}|$, and $|\mathcal{AI}|$ to denote the size of the sets of agents, fluents, and action instances, respectively;
- *WP* to indicate that the solving process returned a Wrong Plan;
- *TO*(*OM*) to indicate that the solving process did not return any solution before the timeout (1 hour) (or run *out of memory*);
- EFP1 to denote the Breadth-First search planner presented in (Le et al. 2018);
- EFP2-k to denote the Breadth-First search planner while using Kripke structures as e-state representation and the transition function of (Baral et al. 2020);
- EFP2-p to denote the Breadth-First search planner while using possibilities as e-state representation and the transition function of (Fabiano et al. 2020);
- EFP3 to denote our Breath-First search planner that makes use of edge-conditioned update model;
- EFP3-h to identify our solver while using the heuristic $h^{maxSubGoal}$.

L	EFP1	EFP2-k	EFP2-p	EFP3	EFP3-h
SC(7,4) with $\|\mathcal{AG}\| = 7$, $\|\mathcal{F}\| = 5$, $\|\mathcal{AI}\| = 7$					
5	.27	.84	.04	.36	.31
7	2.50	19.16	.17	2.35	*TO*
8	6.25	85.21	.38	4.55	.37
SC(9,11) with $\|\mathcal{AG}\| = 9$, $\|\mathcal{F}\| = 12$, $\|\mathcal{AI}\| = 14$					
4	.38	.98	.02	.43	.31
5	.66	1.06	.04	.89	.50(6)
6	1.47	1.71	.11	1.81	1.77
7	4.35	3.63	.29	4.85	.75(8)
8	13.89	10.30	.83	13.92	1.25(10)
9	47.91	36.63	2.05	42.01	1.94(12)
10	160.90	137.47	5.47	121.17	2.20(13)
11	526.64	524.00	15.85	345.07	3.03(14)
SC-LA(7,4) with $\|\mathcal{AG}\| = 7$, $\|\mathcal{F}\| = 5$, $\|\mathcal{AI}\| = 11$					
6	*TO*	*TO*	*TO*	2.94	1.84
7	*TO*	*TO*	*TO*	9.54	*TO*
8	*TO*	*TO*	*TO*	46.60	*TO*
SC-LA(9,11) with $\|\mathcal{AG}\| = 9$, $\|\mathcal{F}\| = 12$, $\|\mathcal{AI}\| = 25$					
4	*TO*	*TO*	*TO*	.56	.46
5	*TO*	*TO*	*TO*	1.29	.77(6)
6	*TO*	*TO*	*TO*	3.81	2.80
7	*TO*	*TO*	*TO*	14.72	1.52(8)
8	*TO*	*TO*	*TO*	57.44	3.85
9	*TO*	*TO*	*TO*	238.16	3.12(12)
10	*TO*	*TO*	*TO*	902.84	3.54(13)

Table 2: Run-time for *Selective Communication* domain

Results. Tables 1–3 detail the performance of the five systems that we experimented with. Time is reported in seconds. For all domains, we run the systems until EFP3.0 timed out. For instance, EFP3.0 times out with the problem **Gr-LA**(4,2) for $L = 3$. Therefore, Table 1 does not include the line with $L = 3$ for this domain. For the set of benchmarks without untruthful announcements, we make the following observations:

- EFP3.0 and EFP3.0-h perform reasonably well in all domains comparing to others. The overhead of computing the edge-conditioned update models reflects in the run-time of EFP3 .0 in problems with large number of actions (e.g., **Gr**(3,2)). As the heuristic $h^{maxSubGoal}$ is inadmissible, EFP3.0-h did not find a solution in some problems.

L	EFP1	EFP2-k	EFP2-p	EFP3	EFP3-h	L	EFP1	EFP2-k	EFP2-p	EFP3	EFP3-h
CC(2,2,3) with $\|\mathcal{AG}\| = 2$, $\|\mathcal{F}\| = 10$, $\|\mathcal{AI}\| = 16$						**CC**(2,2,4) with $\|\mathcal{AG}\| = 2$, $\|\mathcal{F}\| = 14$, $\|\mathcal{AI}\| = 40$					
3	.32	.19	.04	.29	.31	3	5.00	3.81	.21	4.09	4.11
5	5.95	2.15	.40	4.53	.29	5	93.43	43.61	5.76	74.49	4.29
7	154.17	62.89	6.07	98.57	.58	7	*TO*	2761.65	141.10	3269.66	8.57
CC(3,2,3) with $\|\mathcal{AG}\| = 3$, $\|\mathcal{F}\| = 13$, $\|\mathcal{AI}\| = 24$						**CC**(3,3,3) with $\|\mathcal{AG}\| = 3$, $\|\mathcal{F}\| = 15$, $\|\mathcal{AI}\| = 60$					
3	1.64	1.77	.08	1.51	1.48	3	10.83	9.60	.41	9.37	9.97
4	4.22	2.66	.28	3.88	1.18	4	28.44	21.07	2.28	28.63	9.39
5	39.45	16.78	2.04	33.71	1.67	5	559.02	495.82	41.09	528.55	11.11
6	148.39	62.89	6.51	130.33	1.90(7)	6	2420.31	1381.36	128.91	2222.05	9.45
CC-LA(2,2,3) with $\|\mathcal{AG}\| = 2$, $\|\mathcal{F}\| = 10$, $\|\mathcal{AI}\| = 24$						**CC-LA**(2,2,4) with $\|\mathcal{AG}\| = 2$, $\|\mathcal{F}\| = 14$, $\|\mathcal{AI}\| = 28$					
4	*TO*	*TO*	*TO*	1.53	*TO*	3	*TO*	*TO*	*TO*	4.84	3.84
5	*TO*	*TO*	*TO*	14.04	.47	4	*TO*	*TO*	*TO*	19.79	*TO*
6	*TO*	*TO*	*TO*	120.95	*TO*	5	*TO*	*TO*	*TO*	175.01	20.39
7	*TO*	*TO*	*TO*	1349.77	*TO*	6	*TO*	*TO*	*TO*	1533.21	*TO*
CC-LA(3,2,3) with $\|\mathcal{AG}\| = 3$, $\|\mathcal{F}\| = 13$, $\|\mathcal{AI}\| = 36$						**CC-LA**(3,3,3) with $\|\mathcal{AG}\| = 3$, $\|\mathcal{F}\| = 15$, $\|\mathcal{AI}\| = 42$					
4	*TO*	*TO*	*TO*	6.82	*TO*	3	*TO*	*TO*	*TO*	10.43	8.51
5	*TO*	*TO*	*TO*	42.59	9.45	4	*TO*	*TO*	*TO*	92.44	*TO*
6	*TO*	*TO*	*TO*	486.83	21.45	5	*TO*	*TO*	*TO*	711.90	82.07

Table 3: Run-time for *Collaboration and Communication* domain

- EFP2.0-p and EFP3.0-h are fastest among the experimented planners; EFP3.0-h is often faster whenever it can return the answers and the plan is long. However, EFP3.0-h does not guarantee to return the possible optimal plan. For example, EFP3.0-h returns a plan with 14 actions in 3.03 seconds for the problem **SC**(9,11) for $L = 11$ (the optimal plan has 11 actions).

For benchmarks with untruthful announcements, we observe that EFP3.0 does scale up reasonably well. Even with a simple heuristic, EFP3.0-h can scale up much better. However, the inadmissibility of the heuristic does affect EFP3.0-h performance. It is understandable that other systems cannot solve problems in this category because the semantics of earlier specification languages does not consider untruthful announcements. We note that the use of edge-conditioned models of $m\mathcal{A}^*$ allows EFP3.0 to overcome the problem of returning wrong plan by EFP1.0 that was documented in (Fabiano et al. 2020).

Discussion. We would like to close this section with a short discussion on state-of-the art epistemic planning systems in the literature. Apart from EFP1.0 and EFP2.0, our planner is closely related to RP-MEP (Muise et al. 2021) or the system described in (Kominis and Geffner 2015). Similar to our approach, in (Engesser et al. 2017), the author also uses possible worlds semantic and event models to construct a cooperative planning for multi-agent domains. However, their system only works with **S5** theories, and hence, can not reason with false beliefs. Another approach for epistemic planning in multi-agent domains was proposed in the planner MEPK (Wan, Fang, and Liu 2021) and was extended[1] in (Liu and Liu 2018). However, the planner described in (Liu and Liu 2018) does not consider common knowledge of groups of agents. (Liu and Liu 2018) also employs a specification language that requires all changes to the beliefs of agents, including common knowledge to all agents, after an action occurrence be specified explicitly. We note that none of these systems deal with untruthful announcements.

[1]The paper (Wan, Fang, and Liu 2021) is the journal version of the original MEPK that is extended in (Liu and Liu 2018).

Conclusions

In this paper, we present a novel epistemic planner, called EFP3.0, that can generate plans in domains with untruthful announcements and deal with goals of having some agent to have false beliefs. To the best of our knowledge, EFP3.0 is the first epistemic planner with this feature. The planner employs $m\mathcal{A}^*$ as a specification language whose semantics relies on the notion of edge-conditioned update model and overcomes known-issues of an earlier update model based semantics. We describe the implementation of EFP3.0 and the results of an experimental evaluation of two versions of the system, one is a breadth-first-search planner and another one is a best-first-search planner with the heuristic $h^{maxSubGoal}$. The experimental evaluation shows that EFP3.0 and EFP3.0-h indeed can solve planning problems in domains with untruthful announcements and perform reasonably well comparing to other systems in domains without untruthful announcements. The experiment results also show that the development of epistemic planners is still in its infancy and a lot of research will need to be done to create a scalable and efficient epistemic planner comparable to state-of-the-art planning systems in single agent domains. This will be our focus in the immediate future.

Acknowledgments

The authors have been partially supported by NSF grants 2151254, 1914635, 1812628, and 1757207.

References

Aucher, G.; and Bolander, T. 2013. Undecidability in Epistemic Planning. In *Proceedings of the 23rd International Joint Conference on Artificial Intelligence, IJCAI 2013*.

Baltag, A.; and Moss, L. 2004. Logics for epistemic programs. *Synthese*, 139: 165–224.

Baltag, A.; Moss, L.; and Solecki, S. 1998. The logic of public announcements, common knowledge, and private suspicions. In *7th TARK*, 43–56.

Baral, C.; Gelfond, G.; Pontelli, E.; and Son, T. C. 2020. An Action Language for Multi-Agent Domains: Foundations. *arXiv.org*, https://arxiv.org/abs/1511.01960v3.

Baral, C.; Gelfond, G.; Pontelli, E.; and Son, T. C. 2022. An action language for multi-agent domains. *Artif. Intell.*

Bolander, T. 2018. *Seeing Is Believing: Formalising False-Belief Tasks in Dynamic Epistemic Logic*. Cham: Springer International Publishing.

Bolander, T.; and Andersen, M. 2011. Epistemic Planning for Single and Multi-Agent Systems. *Journal of Applied Non-Classical Logics*, 21(1).

Bolander, T.; Charrier, T.; Pinchinat, S.; and Schwarzentruber, F. 2020. DEL-based epistemic planning: Decidability and complexity. *Artif. Intell.*, 287: 103304.

Bolander, T.; Jensen, M. H.; and Schwarzentruber, F. 2015. Complexity Results in Epistemic Planning. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015*.

Charrier, T.; Maubert, B.; and Schwarzentruber, F. 2016. On the Impact of Modal Depth in Epistemic Planning. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, New York, NY, USA, July, 2016*.

Crosby, M.; Jonsson, A.; and Rovatsos, M. 2014. A Single-Agent Approach to Multiagent Planning. In *21st European Conf. on Artificial Intelligence (ECAI'14)*.

Engesser, T.; Bolander, T.; Mattmüller, R.; and Nebel, B. 2017. Cooperative Epistemic Multi-Agent Planning for Implicit Coordination. In *Proceedings of the Ninth Workshop on Methods for Modalities, M4M@ICLA 2017*.

Fabiano, F.; Burigana, A.; Dovier, A.; and Pontelli, E. 2020. EFP 2.0: A Multi-Agent Epistemic Solver with Multiple E-State Representations. *Proceedings of the International Conference on Automated Planning and Scheduling*.

Fagin, R.; Halpern, J.; Moses, Y.; and Vardi, M. 1995. *Reasoning about Knowledge*. MIT press.

Fallis, D. 2020. Shedding Light on Keeping People in the Dark. *Topics in Cognitive Science*, 12(2): 535–554.

Hintikka, J. 1962. *Knowledge and belief*. Cornell University Press.

Kominis, F.; and Geffner, H. 2015. Beliefs In Multiagent Planning: From One Agent to Many. In *Proceedings of the 25th International Conference on Automated Planning and Scheduling, ICAPS 2015, Jerusalem, Israel, June, 2015*.

Kominis, F.; and Geffner, H. 2017. Multiagent Online Planning with Nested Beliefs and Dialogue. In *Proceedings of the Twenty-Seventh International Conference on Automated Planning and Scheduling, ICAPS 2017*.

Le, T.; Fabiano, F.; Son, T. C.; and Pontelli, E. 2018. EFP and PG-EFP: Epistemic Forward Search Planners in Multi-Agent Domains. In *International Conference on Automated Planning and Scheduling (ICAPS)*. AAAI Press.

Liu, Q.; and Liu, Y. 2018. Multi-agent Epistemic Planning with Common Knowledge. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018*.

Löwe, B.; Pacuit, E.; and Witzel, A. 2011. DEL Planning and Some Tractable Cases. In van Ditmarsch, H.; Lang, J.; and Ju, S., eds., *Logic, Rationality, and Interaction*.

Muise, C.; Belle, V.; Felli, P.; McIlraith, S.; Miller, T.; Pearce, A. R.; and Sonenberg, L. 2021. Efficient Multi-agent Epistemic Planning: Teaching Planners About Nested Belief. *arXiv.org*, https://arxiv.org/abs/2110.02480.

Pham, L.; Izmirlioglu, Y.; Son, T. C.; and Pontelli, E. 2022. A New Semantics for Action Language mA*. In *Proceedings of the 24th International Conference on Principles and Practice of Multi-Agent Systems*, PRIMA 22.

Pham, L.; Son, T. C.; and Pontelli, E. 2022. Update Models for Lying and Misleading Announcements. In *Proceedings of the 37th ACM/SIGAPP Symposium on Applied Computing*, SAC '22. New York, NY, USA.

Son, T. C.; Pontelli, E.; Baral, C.; and Gelfond, G. 2014. Finitary S5-Theories. In *Logics in Artificial Intelligence - 14th European Conference, JELIA 2014*.

van Benthem, J.; van Eijck, J.; and Kooi, B. P. 2006. Logics of communication and change. *Inf. Comput.*

van der Hoek, W.; and Wooldridge, M. 2002. Tractable multiagent planning for epistemic goals. In *Proceedings of The First International Joint Conference on Autonomous Agents & Multiagent Systems, AAMAS 2002*.

van Ditmarsch, H. 2014. Dynamics of lying. *Synth.*, 191(5): 745–777.

van Ditmarsch, H.; Hendriks, P.; and Verbrugge, R. 2020. Editors' Review and Introduction: Lying in Logic, Language, and Cognition. *Top. Cogn. Sci.*, 12(2): 466–484.

van Ditmarsch, H.; van der Hoek, W.; and Kooi, B. 2007. *Dynamic Epistemic Logic*. Springer.

van Ditmarsch, H.; van Eijck, J.; Sietsma, F.; and Wang, Y. 2012. On the Logic of Lying. In *Games, Actions and Social Software - Multidisciplinary Aspects*.

Wan, H.; Fang, B.; and Liu, Y. 2021. A general multi-agent epistemic planner based on higher-order belief change. *Artificial Intelligence*, 301: 103562.

Wan, H.; Yang, R.; Fang, L.; Liu, Y.; and Xu, H. 2015. A Complete Epistemic Planner without the Epistemic Closed World Assumption. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015*.

Zlotkin, G.; and Rosenschein, J. S. 1991. Incomplete information and deception in multi-agent negotiation. In *IJCAI'91: Proceedings of the 12th international joint conference on Artificial intelligence*. San Francisco, CA, USA.