Non-Well-Founded Set Based Multi-Agent Action Language*

Francesco Fabiano[1], Idriss Riouak[1], Agostino Dovier[1], and Enrico Pontelli[2]

[1] Dipartimento DMIF, Università di Udine, Udine, Italy
[2] Computer Science Department, New Mexico State University, Las Cruces NM, USA

Abstract As the research in multi-agent domain continues to grow it is becoming more and more important to investigate the agents' relations in such systems: not only to reason about agents' perception of the world but also about agents' knowledge of her and others' knowledge. This type of study is referred as *epistemic reasoning*.
In certain domains, *e.g.*, economy, security, justice and politics, reasoning about others' beliefs could lead to winning strategies or help in changing a group of agents' view of the world.
In this work we formalize the epistemic planning problem where the state description is based on *non-well-founded set* theory. The introduction of such semantics would permit to characterize the planning problem in terms of set operations and not in term of reachability, as the state-of-the-art suggests. Doing so we hope to introduce a more clear semantics and to establish the basis to exploit properties of set based operations inside multi-agent epistemic planning.

Keywords: Epistemic Reasoning · Planning · Multi-agent · Action languages · Non-well-founded sets · Possibilities.

1 Introduction

Multi-agent planning and epistemic knowledge have recently gained attention from several research communities. Efficient autonomous systems that can reason in these domains could lead to winning strategies in various fields such as economy [3], security [4], justice [25], politics [10] and can be exploited by self-driving cars and other autonomous devices that can control several aspects of our daily life.

Epistemic planners are not only interested in the state of the world but also in the knowledge (or beliefs) of the agents. Some problems can be expressed through less expressive languages and need less powerful, and usually faster, planners. For example [18, 24] dealing with problems where dynamic *common knowledge* and unbounded nested knowledge are respectively not needed. On the other hand, to the best of our knowledge, only few systems [21, 22] can reason

* This research is partially supported by the Università di Udine PRID ENCASE project, and by GNCS-INdAM 2017 and 2019 projects.

about epistemic knowledge in multi-agent domains without these limitations, *i.e.*, using the language $\mathcal{L}_{\mathcal{AG}}^{\mathbf{C}}$ presented in Section 2. Such systems, that can reason on the full extent of $\mathcal{L}_{\mathcal{AG}}^{\mathbf{C}}$, base their concept of state on *Kripke structures*. Using a Kripke structure as state has a negative impact on the performances of the planner. First of all, to store all the necessary states, solvers require a high amount of memory. Moreover to perform operations, such as entailment or the application of the transition function, the states have been represented explicitly. That is why, as the research on epistemic reasoning advances [2,5,7], it is interesting to analyze alternative representations for the states that could lead to more efficient operations on the search-space.

In this work we formalized the epistemic planning problem where the states are represented by *possibilities* (introduced in Section 4) which are based on non-well-founded set theory. This representation will allow us to describe the language through set-based operations and also to exploit some of the results from this field, such as the concept of bisimulation, to add important features to the multi-agent epistemic (MEP) community.

The paper is organized as follows: Section 2 will present the concept of epistemic planning. In Section 3 we will introduce what, in our opinion, is the most complete action language for MEP that bases its states on the concept of Kripke structure. The background will be then concluded with Section 4 where we will describe possibilities, an interesting approach that combines non-well-founded set theory with epistemic logic. In Section 5 we will introduce our semantics, based on possibilities, of $m\mathcal{A}^{\rho}$(an action language for MEP) and we will show a comparison with the state-of-the-art action language $m\mathcal{A}^{*}$. We will finally conclude in Section 6 with future works.

2 Epistemic reasoning

In this section we will briefly introduce the background that is necessary to understand multi-agent epistemic reasoning. As defined in [16,27], where the concept of multi-agent propositional epistemic logic is fully explored, the epistemic logic is the logic of *knowledge* and *belief* that different agents have about the world and about the beliefs of each other.

Let $\mathcal{AG}$ be a set of agents and let $\mathcal{F}$ be a set of propositional variables, called *fluents*. We have that each *world* is described by a subset of elements of $\mathcal{F}$ (intuitively, those that are "true" in the world); we also refer to each world as an *interpretation*. For each agent $\mathsf{ag} \in \mathcal{AG}$ we associate a modal operator $\mathbf{B}_{\mathsf{ag}}$ (where $\mathbf{B}$ stands for belief) and we represent the beliefs of an agent as belief formulae in a logic extended with these operators. Moreover, *group operators* are also introduced in epistemic logic, such as $\mathbf{E}_{\alpha}$ and $\mathbf{C}_{\alpha}$, that intuitively represent the belief of a group of agents α and the common knowledge of α respectively. To be more precise, as in [5], we have that

Definition 1 (fluent formulae, atoms and literals). *A* fluent formula *is a propositional formula built using the propositional variables in $\mathcal{F}$ and the tradi-*

tional propositional operators $\wedge, \vee, \Rightarrow, \neg$, *etc. We will use* $\top$ *and* $\bot$ *to indicate* True *and* False*, respectively. A* fluent atom *is a formula composed by just an element* $\mathsf{f} \in \mathcal{F}$*, instead a* fluent literal *is either a fluent atom* $\mathsf{f} \in \mathcal{F}$ *or its negation* $\neg\mathsf{f}$.

Definition 2 (belief formula). *A* belief formula *is defined as follow:*

- *A fluent formula is a belief formula;*
- *let* φ *be belief formula and* $\mathsf{ag} \in \mathcal{AG}$*, then* $\mathbf{B}_{\mathsf{ag}}\varphi$ *is a belief formula;*
- *let* φ_1, φ_2 *and* φ_3 *be belief formulae, then* $\neg\varphi_3$ *and* $\varphi_1 \,\mathsf{op}\, \varphi_2$ *are belief formulae, where* $\mathsf{op} \in \{\wedge, \vee, \Rightarrow\}$*;*
- *all the formulae of the form* $\mathbf{E}_\alpha\varphi$ *or* $\mathbf{C}_\alpha\varphi$ *are belief formulae, where* φ *is itself a belief formula and* $\emptyset \neq \alpha \subseteq \mathcal{AG}$*.*

Let us denote with $\mathcal{L}^{\mathbf{C}}_{\mathcal{AG}}$ the language of the belief formulae over the sets $\mathcal{F}$ and $\mathcal{AG}$. On the other hand we define $\mathcal{L}_{\mathcal{AG}}$ as the language over beliefs formulae that does not allow the use of $\mathbf{C}$. In [14], it is pointed out how these two languages differ in expressiveness and in complexity.

Example 1. Let us consider the formula $\mathbf{B}_{\mathsf{ag}_1}\mathbf{B}_{\mathsf{ag}_2}\varphi$. This formula expresses that the agent ag_1 believes that the agent ag_2 believes that φ is true, instead, $\mathbf{B}_{\mathsf{ag}_1}\neg\varphi$ expresses that the agent ag_1 believes that φ is false.

The classical way of providing a semantics for the language of epistemic logic is in terms of Kripke models [20]:

Definition 3 (Kripke structure). *A* Kripke structure *is a tuple* $\langle S, \pi, \mathcal{B}_1, \ldots, \mathcal{B}_{\mathsf{n}}\rangle$*, such that:*

- S *is a set of worlds;*
- $\pi : S \mapsto 2^{\mathcal{F}}$ *is a function that associates an interpretation of* $\mathcal{F}$ *to each element of* S*;*
- *for* $1 \leq \mathsf{i} \leq \mathsf{n}$*,* $\mathcal{B}_{\mathsf{i}} \subseteq S \times S$ *is a binary relation over* S*.*

A *pointed Kripke structure* is a pair (M, s) where M is a Kripke structure as defined above, and $\mathsf{s} \in S$, where s represents the real world. As in [5], we will refer to a pointed Kripke structure (M, s) as a *state*.

Following the notation of [5], we will indicate with $M[S], M[\pi]$, and $M[\mathsf{i}]$ the components S, π, and $\mathcal{B}_{\mathsf{i}}$ of M, respectively.

Definition 4 (entailment w.r.t. a Kripke structure). *Given the belief formulae* $\varphi, \varphi_1, \varphi_2$*, an agent* ag_{i}*, a group of agents* α*, a Kripke structure* $M = \langle S, \pi, \mathcal{B}_1, ..., \mathcal{B}_{\mathsf{n}}\rangle$*, and the worlds* $\mathsf{s}, \mathsf{t} \in S$*:*

(i) $(M, \mathsf{s}) \models \varphi$ *if* φ *is a fluent formula and* $\pi(\mathsf{s}) \models \varphi$*;*
(ii) $(M, \mathsf{s}) \models \mathbf{B}_{\mathsf{ag}_{\mathsf{i}}}\varphi$ *if for each* t *such that* $(\mathsf{s}, \mathsf{t}) \in \mathcal{B}_{\mathsf{i}}$ *it holds that* $(M, \mathsf{t}) \models \varphi$*;*
(iii) $(M, \mathsf{s}) \models \neg\varphi$ *if* $(M, \mathsf{s}) \not\models \varphi$*;*
(iv) $(M, \mathsf{s}) \models \varphi_1 \vee \varphi_2$ *if* $(M, \mathsf{s}) \models \varphi_1$ *or* $(M, \mathsf{s}) \models \varphi_2$*;*
(v) $(M, \mathsf{s}) \models \varphi_1 \wedge \varphi_2$ *if* $(M, \mathsf{s}) \models \varphi_1$ *and* $(M, \mathsf{s}) \models \varphi_2$*;*

(vi) $(M, \mathsf{s}) \models \mathbf{E}_\alpha \varphi$ *if* $(M, \mathsf{s}) \models \mathbf{B}_{\mathsf{ag_i}} \varphi$ *for all* $\mathsf{ag_i} \in \alpha$;

(vii) $(M, \mathsf{s}) \models \mathbf{C}_\alpha \varphi$ *if* $(M, \mathsf{s}) \models \mathbf{E}^k_\alpha \varphi$ *for every* $k \geq 0$, *where* $\mathbf{E}^0_\alpha \varphi = \varphi$ *and* $\mathbf{E}^{k+1}_\alpha \varphi = \mathbf{E}_\alpha(\mathbf{E}^k_\alpha \varphi)$.

We will no further describe the properties of the Kripke structures since those are not strictly needed to describe the contribution of this paper. The reader who has interest in a more detailed description can refer to [14].

3 The action language $m\mathcal{A}^*$

The $m\mathcal{A}^*$ [5] action language is a generalization of the single-agent action languages, extensively studied in the literature [8, 26], to the case of multi-agent domains for epistemic planning.

The language has a declarative, English-like syntax and an event model based semantics which permits to reason about beliefs.

The semantics of $m\mathcal{A}^*$ is based on the assumption that agents are truthful. The language is built over a signature $(\mathcal{AG}, \mathcal{F}, \mathcal{A})$, where $\mathcal{AG}$ is a finite set of agent identifiers, $\mathcal{F}$ is a set of fluents, and $\mathcal{A}$ is a set of actions.

We will introduce only the basics features of $m\mathcal{A}^*$. The remaining details about the language can be found in [5]. The following will be used as working example through the paper:

Example 2 (Three Agents and the Coin Box). Three agents, A, B, and C, are in a room where in the middle there is a box containing a coin. It is common knowledge that:

- None of the agents know whether the coin lies heads or tails up;
- The box is locked and one needs a key to open it;
- Agent A has the key of the box;
- In order to learn whether the coin lies heads or tails up, an agent can peek into the box, but this require the box to be open;
- If one agent is looking at the box and a second agents peeks into the box, then the first agent will observe this fact and will be able to conclude that the second agent knows the status of the coin. On the other hand, the first agent's knowledge about which face of the coin is up does not change.
- Distracting an agent causes her to not look at the box;
- Signaling an agent to look at the box causes such agent to look at the box;
- Announcing that the coin lies heads or tails up will make this common knowledge among the agents that are listening.

Agent A would like to know whether the coin lies heads or tails up. She would also like to let agent B knowing that she knows this fact. However, she would like to keep this information secret from C.

This can be achieved by: i) *Distracting* C from looking at the box; ii) *Signaling* B to look at the box if B is not looking at it; iii) *Opening* the box; and iv) *Peeking* into the box.

For this domain, we have that $\mathcal{AG} = \{\mathsf{A}, \mathsf{B}, \mathsf{C}\}$, while the set of fluent $\mathcal{F}$ consists of:

– $\texttt{heads}$: the coin lies heads up;
– $\texttt{key}(\texttt{ag})$: agent $\texttt{ag}$ has the key of the box;
– $\texttt{opened}$: the box is open; and
– $\texttt{look}(\texttt{ag})$: agent $\texttt{ag}$ is looking at the box.

Let $\texttt{ag} \in \mathcal{AG}$, the set of actions $\mathcal{A}$ comprises:

– $\texttt{open}$: an agent opens the box;
– $\texttt{peek}$: an agent peeks into the box;
– $\texttt{signal}(\texttt{ag})$: an agent signals to agent $\texttt{ag}$ to look at the box;
– $\texttt{distract}(\texttt{ag})$: an agent distracts agent $\texttt{ag}$;
– $\texttt{shout\_tails}$: an agent announces that the coin lies tails up.

In [5], the authors distinguished between three types of actions in the following way (some examples of action execution can be found in Appendix):

– *World-altering* action (also called *ontic*): used to modify certain properties (*i.e.*, fluents) of the world, *e.g.*, the action $\texttt{open}$ or $\texttt{distract}(\texttt{ag})$ of Example 2.
– *Sensing* action: used by an agent to refine her beliefs about the world, *e.g.*, the action $\texttt{peek}$.
– *Announcement* action: used by an agent to affect the beliefs of other agents. *e.g.*, in Example 2 the action $\texttt{shout\_tails}$.

Given an action instance $\mathsf{a} \in \mathcal{AI}$, where $\mathcal{AI}$ is the set of all the possible action instances $\mathcal{A} \times \mathcal{AG}$, a fluent literal $\mathsf{f} \in \mathcal{F}$, a fluent formula ϕ and a belief formula φ we can quickly introduce the syntax adopted in $m\mathcal{A}^*$.

Executability conditions are captured by statements of the form:

$$\textbf{executable } \mathsf{a} \textbf{ if } \varphi$$

For ontic actions we have:

$$\mathsf{a} \textbf{ causes } \mathsf{f} \textbf{ if } \varphi$$

Sensing actions statements have the form expressed by

$$\mathsf{a} \textbf{ determines } \mathsf{f}$$

Finally announcement actions are expressed as follows:

$$\mathsf{a} \textbf{ announces } \phi.$$

In multi-agent domains the execution of an action might change or not the beliefs of an agent. This because, in such domains, each action instance associates an observability relation to each agent. For example the agent C that becomes oblivious as distracted by the agent A, is not able to see the execution of the action $\texttt{open}\langle\mathsf{A}\rangle$. On the other hand, watching an agent executing a sensing or an announcement action can change the beliefs of who is watching, *e.g.*, the agent B, who is watching the agent A sensing the status of the coin, will know that A

Action type	Full observers	Partial Observers	Oblivious
World-altering	✓		✓
Sensing	✓	✓	✓
Announcement	✓	✓	✓

Table 1: Action type and observability relations.

knows the status of the coin without knowing the status herself. In Table 1 are summarized the possible observability relations for each type of action. Partial observability for World-altering action is not admitted as, whenever an agent is aware of the execution of an ontic action, she must knows its effects on the world as well.

For brevity we address the reader to [5] for the definition of the transition function in $m\mathcal{A}^*$.

4 Non-well-founded sets and Possibilities

This section defines the concept of *possibility* (originally introduced in [15]), based on non-well-founded set theory. This section aims to provide the reader with enough information to understand what possibilities are.

For a more informative introduction the reader is addressed to [1, 6, 16] and to [12] for a logic programming point of view on non-well-founded sets and their equivalence.

4.1 Non-well-founded set theory fundamentals

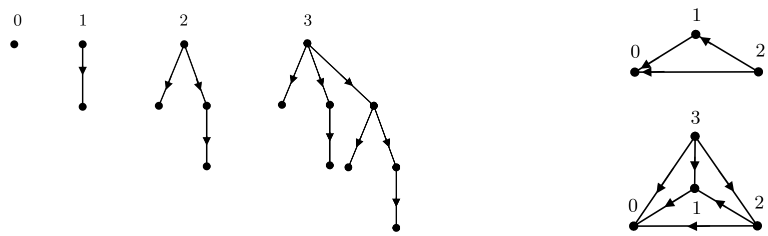
We start by giving some fundamental definitions of non-well-founded set theory. First of all a *well-founded set* is described in [1] as follows:

Definition 5 (well-founded set). *Let E be a set, E' one of its elements, E'' any element of E', and so on. A descent is the sequence of steps from E to E', E' to E'', etc. ... A set is* well-founded *(or* ordinary*) when it only gives rise to finite descents.*

Well-founded set theory states that all the sets in the sense of Definition 5 can be represented in the form of graphs, called *pictures*, (as shown in Figure 1). To formalize this concept of 'picture of a set' however it is necessary to introduce the concept of *decoration*:

Definition 6 (decoration and picture).

- *A* decoration *of a graph $\mathcal{G}=(V, E)$ is a function δ that assigns to each node $\mathsf{n} \in V$ a set δ_n in such a way that the elements of δ_n are exactly the sets assigned to successors of* n, i.e., $\delta_\mathsf{n} = \{\delta_{\mathsf{n}'} \mid (\mathsf{n}, \mathsf{n}') \in E\}$.
- *If δ is a decoration of a pointed graph $(\mathcal{G}, \mathsf{n})$, then $(\mathcal{G}, \mathsf{n})$ is a* picture *of the set δ_n.*



(a) Pictures of von Neumann ordinals where $0 = \emptyset$; $1 = \{\emptyset\}$; $2 = \{\emptyset, \{\emptyset\}\}$; $3 = \{\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}\}$.

(b) Alternative Pictures of von Neumann ordinals 2 and 3.

Figure 1: Well-founded sets represented through graphs [1].

Moreover, in well-founded set theory, it holds the Mostovski's lemma: "each well-founded graph[3] is a picture of exactly one set".

On the other hand in [1] a *non-well-founded*, or *extraordinary set* in the sense of Mirimanoff, is a set that respects Definition 7.

Definition 7 (non-well-founded set). *A set is* non-well-founded *(or* extraordinary*) when among its descents there are some which are infinite.*

In fact, when the **Foundation Axiom**[4] is substituted by the **Anti-Foundation Axiom** (**AFA**), expressed by Aczel in [1] as "*Every graph has a unique decoration*", the following consequences become true:

- Every graph is a picture of exactly one set (**AFA** as is formulated in [16]);
- non-well-founded sets exist given that a non-well-founded pointed graph has to be a picture of a non-well-founded set.

(a) Standard picture Ω.

(b) Unfolding of the picture of Ω.

Figure 2: Representation of the non-well-founded set $\Omega = \{\Omega\}$ [1].

In [1, 16] it is pointed out how non-well-founded sets can also be expressed through systems of equations. This concept will help us to formalize the notion of state in our action language.

[3] A well-founded graph is a graph that doesn't contain an infinite path $\mathsf{n} \to \mathsf{n}' \to \mathsf{n}'' \to \dots$ of successors.

[4] Expressed in [16] as "Only well-founded graphs have decorations".

A quick example of this representation can be derived by the set $\Omega = \{\Omega\}$ (Figure 2). We can, in fact, informally define this set by the (singleton) system of equations $x = \{x\}$. Systems of equations and their solutions are described more formally as follows in [16]:

Definition 8 (system of equations). *For each class of atoms*[5] $\mathcal{X}$ *a* system of equation *in* $\mathcal{X}$ *is a class* τ *of equations* $\mathsf{x} = \mathtt{X}$*, where* $\mathsf{x} \in \mathcal{X}$ *and* $\mathtt{X} \subseteq \mathcal{X}$*, such that* τ *contains exactly one equation* $\mathsf{x} = \mathtt{X}$ *for each* $\mathsf{x} \in \mathcal{X}$*. A solution to a system of equations* τ *is a function* δ *that assigns to each* $\mathsf{x} \in \tau(\mathcal{X})$[6] *a set* δ_x *such that* $\delta_\mathsf{x} = \{\delta_\mathsf{y} \mid \mathsf{y} \in \mathtt{X}\}$*, where* $\mathsf{x} = \mathtt{X}$ *is an equation of* τ*. If* δ *is the solution to a system of equations* τ*, then the set* $\{\delta_\mathsf{x} \mid \mathsf{x} \in \tau(\mathcal{X})\}$ *is called the solution set of that system.*

Since both graphs and systems of equations are representations for non-well-founded sets, it is natural to investigate their relationships. In particular it is interesting to point out how from a graph $\mathcal{G}{=}(V, E)$ it is possible to construct a system of equations τ and vice versa. The nodes in $\mathcal{G}$, in fact, can be the set of atoms $\tau(\mathcal{X})$ and, for each node $\mathsf{v} \in V$, an equation is represented by $\mathsf{v} = \{\mathsf{v}' \mid (\mathsf{v}, \mathsf{v}') \in E\}$. Since each graph has a unique decoration, each system of equations has a unique solution. This is also true when we consider bisimilar systems of equations. In fact we can collapse them into their minimal representation thanks to the concept of *maximum bisimulation* as introduced in [12]. Bisimilar labeled graphs (or Kripke structures) have therefore a unique solution as well since we collapse their representations into the minimal one. This idea will be further expanded in Section 5.2.

4.2 Possibilities

Let us introduce the notion of possibility as in [15]:

Definition 9 (possibilities). *Let* $\mathcal{AG}$ *be a set of agents and* $\mathcal{F}$ *a set of propositional variables:*

- *A* possibility u *is a function that assigns to each propositional variable* $\mathsf{f} \in \mathcal{F}$ *a truth value* $\mathsf{u}(\mathsf{f}) \in \{0, 1\}$ *and to each agent* $\mathsf{ag} \in \mathcal{AG}$ *an information state* $\mathsf{u}(\mathsf{ag}) = \sigma$*.*
- *An* information state σ *is a set of possibilities.*

In Section 5.1 we will use this concept to describe a 'state' of the planning problem. The intuition behind this idea is that a possibility u is a possible interpretation of the world and of the agents' beliefs; in fact $\mathsf{u}(\mathsf{f})$ specifies the truth value of the fluent f in u and $\mathsf{u}(\mathsf{A})$ is the set of all the interpretations the agent A considers possible in u.

Moreover a possibility can be pictured as a decoration of a labeled graph and therefore as a unique solution to a system of equations for possibilities. A

[5] Objects that are not sets and have no further set-theoretic structure.

[6] $\tau(\mathcal{X})$ denotes the class of atoms $\mathcal{X}$ in which τ is described.

possibility represents the solution to the minimal system of equations in which all bisimilar systems of equations are collapsed; that is the possibilities that represent decorations of bisimilar labeled graphs are bisimilar and can be represented by the minimal one. This shows that the class of bisimilar labeled graphs and, therefore, of bisimilar Kripke structures, used by $m\mathcal{A}^*$ as states, can be represented by a single possibility.

Definition 10 (equations for possibilities). *Given a set of agents $\mathcal{AG}$ and a set of propositional variables $\mathcal{F}$, a* system of equations for possibilities *in a class of possibilities $\mathcal{X}$ is a set of equations such that for each $\mathsf{x} \in \mathcal{X}$ there exists exactly one equation of the form $\mathsf{x}(\mathsf{f}) = i$, where $i \in \{0, 1\}$, for each $\mathsf{f} \in \mathcal{F}$, and of the form $\mathsf{x}(\mathsf{ag}) = \mathtt{X}$, where $\mathtt{X} \subseteq \mathcal{X}$, for each $\mathsf{ag} \in \mathcal{AG}$.*
A solution to a system of equations for possibilities is a function δ that assigns to each atom x a possibility δ_{x} in such a way that if $\mathsf{x}(\mathsf{f}) = i$ is an equation then $\delta_{\mathsf{x}(\mathsf{f})} = i$, and if $\mathsf{x}(\mathsf{ag}) = \sigma$ is an equation, then $\delta_{\mathsf{x}(\mathsf{ag})} = \{\delta_{\mathsf{y}} \mid \mathsf{y} \in \sigma\}$.

5 The action language $m\mathcal{A}^\rho$

The research on multi-agent epistemic domain, both in logic and planning, already comprehends several theoretical studies [2,5,7,13,14,16,17,23,27,28] and also a variety of solvers [18,19,21,22,24,29] even if, at the best of our knowledge, only [13,21] can reason without limitations on domains described by $\mathcal{L}_{\mathcal{AG}}^{\mathbf{C}}$.

Anyway multi-agent epistemic solvers still have to reason on domains where the number of fluents and/or agents is limited. This is to reduce the length of the planning problems solution that otherwise would require an excessive quantity of resources (*i.e.*, time and memory). For these reasons the demand of computational resources needed (for example in respect to classical planning) is one of the central problem in MEP.

To reduce this gap several approaches can be used: i) as in [18,19,24,29] the planning domain can be limited to a less expressive class of problems. On the other hand, when generality is required, ii) heuristics can effectively reduce the resolution process, as shown in [21]. Finally another approach to follow could be iii) to consider alternative representations to Kripke structures; this is what $m\mathcal{A}^\rho$ tries to do.

Changing the structure is especially important because different state representation can lead to a better use of the resources, and to exploit properties of the new structure to introduce important functionalities; *e.g.*, $m\mathcal{A}^\rho$ could rely on the concept of *bisimulation* to introduce the notion of *visited states*, being bisimulation an equality criteria for non-well-founded sets.

5.1 State

As main contribution we introduce a modified version of $m\mathcal{A}^*$, called $m\mathcal{A}^\rho$. The difference is in how a state is defined: in $m\mathcal{A}^*$ a state is represented as a Kripke structure while in $m\mathcal{A}^\rho$ a state is a possibility (Section 4.2). For simplicity, we maintain the syntax used in $m\mathcal{A}^*$.

The strict connection of these two structures is highlighted in [16] from the fact that a solution to a system of equations for possibilities (Definition 10) represents a decoration for a labeled graph, which is essentially a Kripke model. In [16], it is also expressed that a "possibility corresponds with a whole class of bisimilar, but structurally different, Kripke models".

Let us define the usage of possibilities as states in a MEP domain where the set of agents is $\mathcal{AG}$ and the set of fluents is $\mathcal{F}$. A possibility is a function that assigns to each propositional variable $\mathsf{f} \in \mathcal{F}$ a truth value $\mathsf{u}(\mathsf{f}) \in \{0, 1\}$ and to each agent $\mathsf{ag} \in \mathcal{AG}$ a set of possibilities $\mathsf{u}(\mathsf{ag})$.

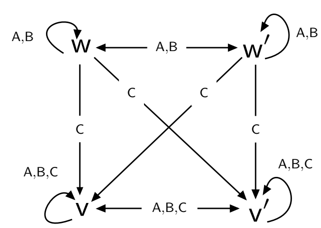
A state in MEP has to encode the truth value of the fluents, as in classical planning, and also the beliefs of the agents about fluents and beliefs themselves. We defined in Section 3 how Kripke structures represent these information. In Figure 3 we use a possibility as a system of equations to encode a state (of the domain in Example 2).

An equation is represented in the form $\mathsf{u} = \{(\mathsf{ag}_1, \sigma), (\mathsf{ag}_2, \sigma'), \dots, \mathsf{f}, \mathsf{f}', \dots\}$ where ag_1, $\mathsf{ag}_2 \in \mathcal{AG}$, σ, σ' are sets of possibilities and $\mathsf{f}, \mathsf{f}' \in \mathcal{F}$. When we write (ag, σ) we mean that, in u, the agent ag believes that the possibilities in σ are plausible. On the other hand only if a fluent f is present in the equation this means that the fluent itself is true in u.

$$\begin{cases} \mathsf{w} & = \{(\mathsf{ag}, \{\mathsf{w}, \mathsf{w}'\}), (\mathsf{C}, \{\mathsf{v}, \mathsf{v}'\}), \mathtt{look}(\mathsf{ag}), \mathtt{key}(\mathsf{A}), \mathtt{opened}\} \\ \mathsf{w}' & = \{(\mathsf{ag}, \{\mathsf{w}, \mathsf{w}'\}), (\mathsf{C}, \{\mathsf{v}, \mathsf{v}'\}), \mathtt{look}(\mathsf{ag}), \mathtt{key}(\mathsf{A}), \mathtt{opened}, \mathtt{heads}\} \\ \mathsf{v} & = \{(\mathsf{A}, \{\mathsf{v}, \mathsf{v}'\}), (\mathsf{B}, \{\mathsf{v}, \mathsf{v}'\}), (\mathsf{C}, \{\mathsf{v}, \mathsf{v}'\}), \mathtt{look}(\mathsf{ag}), \mathtt{key}(\mathsf{A})\} \\ \mathsf{v}' & = \{(\mathsf{A}, \{\mathsf{v}, \mathsf{v}'\}), (\mathsf{B}, \{\mathsf{v}, \mathsf{v}'\}), (\mathsf{C}, \{\mathsf{v}, \mathsf{v}'\}), \mathtt{look}(\mathsf{ag}), \mathtt{key}(\mathsf{A}), \mathtt{heads}\} \end{cases}$$

where $\mathsf{ag} \in \{\mathsf{A}, \mathsf{B}\}$.

(a) System of equations for possibilities.



(b) Decoration of the pointed labeled graph $(\mathcal{G}, \mathsf{w})$.

Figure 3: Representation of the possibility w after the execution of the actions $\mathtt{distract}(\mathsf{C})\langle\mathsf{A}\rangle;\mathtt{open}\langle\mathsf{A}\rangle$ on the initial state of Example 2. The possibility is expanded to its system of equation for clarity.

It is clear that a possibility correspond to a decoration of a pointed labeled graph and therefore to a unique Kripke model up to bisimulation. The representation through possibilities allows, in our opinion, a more clear and concise view of the state. That is because each state is represented by a single possibility; *e.g.*, Figure 3 is represented by $\mathsf{w} = \{(\mathsf{ag}, \{\mathsf{w}, \mathsf{w}'\}), (\mathsf{C}, \{\mathsf{v}, \mathsf{v}'\}), \mathtt{look}(\mathsf{ag}), \mathtt{key}(\mathsf{A}), \mathtt{opened}\}$ where $\mathsf{ag} \in \{\mathsf{A}, \mathsf{B}\}$.

5.2 State equality through Bisimulation

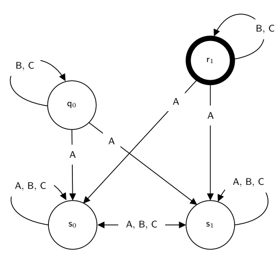
One of the reasons we chose to use possibilities as states is to exploit this new structure to introduce new functionalities to MEP. In fact having the states described as possibilities, which are strongly related to non-well-founded sets, help us to introduce the concept of *visited states*, a core idea in planning.

As said before, a possibility in the sense of decoration, represents all the Kripke structures bisimilar to the decoration itself. This means that with possibilities we can exploit bisimulation to capture the idea of equality between states. In fact, given two bisimilar decorations (or labeled graphs), these, even with structural differences, are represented by the same possibility. On the other hand this is not true when it comes to Kripke structures. This idea is best described through graphical representation; therefore we will use Figure 4 to explain this concept.

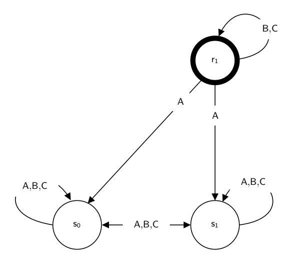
As an example, let us introduce two new actions: `flip` and `tell`(ag). The first one is an ontic action, where an agent inverts the position of the coin; the observability of this action depends on the fluents `looking`. On the other hand, `tell`(ag) means that an agent announces to ag the position in which she thinks the coin lies while all the other agents are oblivious.

Assuming that the coin lies tails up and given a sequence of action instances $\Delta = \texttt{peek}\langle \mathsf{B}\rangle; \texttt{distract}(\mathsf{B})\langle \mathsf{C}\rangle; \texttt{flip}\langle \mathsf{C}\rangle; \texttt{tell}(\mathsf{B})\langle \mathsf{C}\rangle$[7] we show in Figure 4a the result of applying Δ in a slightly modified initial state of Example 2 where $\mathbf{C}_{\{A,B,C\}}(\texttt{opened} \wedge \neg \texttt{look}(\mathsf{A}))$. In Figure 4b, it is represented a Kripke structure that has structural differences in respect to the one in Figure 4a. This means that these two Kripke structures represent two different states in $m\mathcal{A}^*$ even if they are intuitively the same. On the contrary, if we think in term of possibilities, both the Kripke structures of Figure 4 are represented by possibility

$$\mathsf{r}_1 = \{(\mathsf{A}, \{\mathsf{s}_0, \mathsf{s}_1\}), (\mathsf{B}, \{\mathsf{r}_1\}), (\mathsf{C}, \{\mathsf{r}_1\}), \texttt{look}(\mathsf{C}), \texttt{key}(\mathsf{A}), \texttt{opened}, \texttt{heads}\}.$$



(a) The resulting Kripke structure after the execution of Δ.



(b) A Kripke structure bisimilar to the one in Figure 4a.

Figure 4: Bisimilar Kripke Structures.

Thanks to the use of possibilities we can capture bisimilar decoration while the same were considered different in $m\mathcal{A}^*$ and, therefore, check if a state has

[7] We recall that in [5] an action instance is represented as `action`⟨ag⟩ where ag is the agent that executes `action`.

already been visited. Next we define the concept of entailment and finally the transition function in $m\mathcal{A}^{\rho}$.

5.3 Entailment

We now introduce the concept of entailment w.r.t. possibilities.

Definition 11 (entailment w.r.t. possibilities). *Given the belief formulae* $\varphi, \varphi_1, \varphi_2$*, a fluent* f*, an agent* ag*, a group of agents* α*, and a possibility* u*:*

- *(i)* $\mathsf{u} \models \mathsf{f}$ *if* $\mathsf{u}(\mathsf{f}) = 1$*;*
- *(ii)* $\mathsf{u} \models \neg\phi$ *if* $\mathsf{u} \not\models \phi$*;*
- *(iii)* $\mathsf{u} \models \phi_1 \vee \phi_2$ *if* $\mathsf{u} \models \phi_1$ *or* $\mathsf{u} \models \phi_2$*;*
- *(iv)* $\mathsf{u} \models \phi_1 \wedge \phi_2$ *if* $\mathsf{u} \models \phi_1$ *and* $\mathsf{u} \models \phi_2$*.*
- *(v)* $\mathsf{u} \models \mathbf{B}_{\mathsf{ag}}\varphi$ *if for each* $\mathsf{v} \in \mathsf{u}(\mathsf{ag})$*,* $\mathsf{v} \models \varphi$*;*
- *(vi)* $\mathsf{u} \models \neg\varphi$ *if* $\mathsf{u} \not\models \varphi$*;*
- *(vii)* $\mathsf{u} \models \varphi_1 \vee \varphi_2$ *if* $\mathsf{u} \models \varphi_1$ *or* $\mathsf{u} \models \varphi_2$*;*
- *(viii)* $\mathsf{u} \models \varphi_1 \wedge \varphi_2$ *if* $\mathsf{u} \models \varphi_1$ *and* $\mathsf{u} \models \varphi_2$*;*
- *(ix)* $\mathsf{u} \models \mathbf{E}_{\alpha}\varphi$ *if* $\mathsf{u} \models \mathbf{B}_{\mathsf{ag}}\varphi$ *for all* $\mathsf{ag} \in \alpha$*;*
- *(x)* $\mathsf{u} \models \mathbf{C}_{\alpha}\varphi$ *if* $\mathsf{u} \models \mathbf{E}^k_{\alpha}\varphi$ *for every* $k \geq 0$*, where* $\mathbf{E}^0_{\alpha}\varphi = \varphi$ *and* $\mathbf{E}^{k+1}_{\alpha}\varphi = \mathbf{E}_{\alpha}(\mathbf{E}^k_{\alpha}\varphi)$*.*

5.4 Transition function

Finally we introduce the transition function for $m\mathcal{A}^{\rho}$. In defining this transition function we made some assumptions: i) we consider that the given initial state also specifies which world is the pointed one (as said in [5]), this allow us to relax the description of Φ in the case of announcements or sensing actions. Moreover ii) we do not take into consideration the case in which an agent can have *false beliefs*[8] because is still an open question how to deal with them in MEP; we will try to address this problem in future works. The transition function relative to the domain D is $\Phi_D : \mathcal{AI} \times \Sigma \rightarrow \Sigma \cup \{\emptyset\}$ where Σ is the set of all the possibilities.

For the sake of readability, the abbreviations used in the following definition are explained at the end of it.

Definition 12 (transition function for $m\mathcal{A}^{\rho}$). *Let* $\mathsf{a} \in \mathcal{AI}$*, and a possibility* u *be given. If* a *is not executable in* u *then* $\Phi_D(\mathsf{a}, \mathsf{u}) = \emptyset$ *otherwise if* a *is executable in* u *:*

- $\Phi_D(\mathsf{a}, \mathsf{u}) = \mathsf{v}$ *if* a *is an ontic action instance and*

$$\begin{cases} \mathsf{v}(\mathsf{f}) = \mathsf{u}(\mathsf{f}) & \textit{if } \mathsf{f} \neq \mathsf{caused}(\mathsf{a}) \\ \mathsf{v}(\mathsf{f}) = \mathsf{caused}(\mathsf{a}) & \textit{if } \mathsf{f} = \mathsf{caused}(\mathsf{a}) \end{cases}$$

and

$$\begin{cases} \mathsf{v}(\mathsf{ag}) = \mathsf{u}(\mathsf{ag}) & \textit{if } \mathsf{ag} \in O_D \\ \mathsf{v}(\mathsf{ag}) = \bigcup_{\mathsf{w} \in \mathsf{u}(\mathsf{ag})} \Phi_D(\mathsf{a}, \mathsf{w}) & \textit{if } \mathsf{ag} \in F_D \end{cases}$$

[8] An agent ag has a false belief about φ in state s if $\mathsf{s} \models \varphi$ and $\mathsf{s} \models \mathbf{B}_{\mathsf{ag}}\neg\varphi$.

– $\Phi_D(\mathsf{a},\mathsf{u}) = \mathsf{v}$ *if* a *sensed the fluent* f

$$\begin{cases} \emptyset & \textit{if } \mathsf{sensed}(\mathsf{a})[\mathsf{f}] \neq \mathsf{u}(\mathsf{f}) \\ \mathsf{v}(\mathsf{ag}) = \mathsf{u}(\mathsf{ag}) & \textit{if } \mathsf{ag} \in O_D \textit{ and } \mathsf{sensed}(\mathsf{a})[\mathsf{f}] = \mathsf{u}(\mathsf{f}) \\ \mathsf{v}(\mathsf{ag}) = \bigcup_{\mathsf{w} \in \mathsf{u}(\mathsf{ag})} \Phi_D(\mathsf{a},\mathsf{w}) & \textit{if } \mathsf{ag} \in F_D \textit{ and } \mathsf{sensed}(\mathsf{a})[\mathsf{f}] = \mathsf{u}(\mathsf{f}) \\ \mathsf{v}(\mathsf{ag}) = \bigcup_{\mathsf{w} \in \mathsf{u}(\mathsf{ag})} (\Phi_D(\mathsf{a},\mathsf{w}) \cup \Phi_D(\neg\mathsf{a},\mathsf{w})) & \textit{if } \mathsf{ag} \in P_D \textit{ and } \mathsf{sensed}(\mathsf{a})[\mathsf{f}] = \mathsf{u}(\mathsf{f}) \end{cases}$$

– $\Phi_D(\mathsf{a},\mathsf{u}) = \mathsf{v}$ *if* a *announces the fluent formula* ϕ

$$\begin{cases} \emptyset & \textit{if } \mathsf{u} \not\models \phi \\ \mathsf{v}(\mathsf{ag}) = \mathsf{u}(\mathsf{ag}) & \textit{if } \mathsf{ag} \in O_D \textit{ and } \mathsf{u} \models \phi \\ \mathsf{v}(\mathsf{ag}) = \bigcup_{\mathsf{w} \in \mathsf{u}(\mathsf{ag})} \Phi_D(\mathsf{a},\mathsf{w}) & \textit{if } \mathsf{ag} \in F_D \textit{ and } \mathsf{u} \models \phi \\ \mathsf{v}(\mathsf{ag}) = \bigcup_{\mathsf{w} \in \mathsf{u}(\mathsf{ag})} (\Phi_D(\mathsf{a},\mathsf{w}) \cup \Phi_D(\neg\mathsf{a},\mathsf{w})) & \textit{if } \mathsf{ag} \in P_D \textit{ and } \mathsf{u} \models \phi \end{cases}$$

where:

- caused(a) is the fluent modified by the action instance a;
- F_D, P_D, O_D identify the fully observant, partially observant and oblivious agents respectively;
- sensed(a)[f] represents the truth value of the fluent f determined by $\mathsf{a} = \mathsf{u}(\mathsf{f})$;
- $\neg\mathsf{a}$ describes the action that senses the opposite of a; namely sensed(a)[¬f].

Note that sensing and announcement actions generate the empty set when the action effects do not respect the fluents truth values of the possibility where the action is executed. That is because epistemic action (*i.e.*, announcement or sensing) cannot change the state of the world but only the beliefs of the agent.

As example we present the execution of the sequence `distract`(C)⟨A⟩; `open`⟨A⟩ on the initial state of Example 2 encoded by the possibility u in Figure 5a.

$$\begin{cases} \mathsf{u} & = \{(\mathsf{ag},\{\mathsf{u},\mathsf{u}'\}), \mathtt{look}(\mathsf{ag}), \mathtt{key}(\mathsf{A})\} \\ \mathsf{u}' & = \{(\mathsf{ag},\{\mathsf{u},\mathsf{u}'\}), \mathtt{look}(\mathsf{ag}), \mathtt{key}(\mathsf{A}), \mathtt{heads}\} \end{cases}$$

(a) The initial state u.

$$\begin{cases} \mathsf{v} & = \{(\mathsf{ag},\{\mathsf{v},\mathsf{v}'\}), \mathtt{look}(\mathsf{A}), \mathtt{look}(\mathsf{B}), \mathtt{key}(\mathsf{A})\} \\ \mathsf{v}' & = \{(\mathsf{ag},\{\mathsf{v},\mathsf{v}'\}), \mathtt{look}(\mathsf{A}), \mathtt{look}(\mathsf{B}), \mathtt{key}(\mathsf{A}), \mathtt{heads}\} \end{cases}$$

(b) The state v after executing `distract`(C)⟨A⟩.

$$\begin{cases} \mathsf{w} & = \{(\mathsf{A},\{\mathsf{w},\mathsf{w}'\}), (\mathsf{B},\{\mathsf{w},\mathsf{w}'\}), (\mathsf{C},\{\mathsf{v},\mathsf{v}'\}), \mathtt{look}(\mathsf{A}), \mathtt{look}(\mathsf{B}), \mathtt{key}(\mathsf{A}), \mathtt{opened}\} \\ \mathsf{w}' & = \{(\mathsf{A},\{\mathsf{w},\mathsf{w}'\}), (\mathsf{B},\{\mathsf{w},\mathsf{w}'\}), (\mathsf{C},\{\mathsf{v},\mathsf{v}'\}), \mathtt{look}(\mathsf{A}), \mathtt{look}(\mathsf{B}), \mathtt{key}(\mathsf{A}), \mathtt{opened}, \mathtt{heads}\} \end{cases}$$

(c) The state w after executing `distract`(C)⟨A⟩;`open`⟨A⟩.

Figure 5: Execution of `distract`(C)⟨A⟩;`open`⟨A⟩ in Example 2 ($\mathsf{ag} \in \{\mathsf{A},\mathsf{B},\mathsf{C}\}$).

6 Conclusions and Future Works

In this paper we investigated an alternative to Kripke structures as representation for multi-agent epistemic planning states. Doing so we presented $m\mathcal{A}^\rho$, an action language for MEP based on possibilities, introducing MEP in non-well-founded set theory. Moreover we exploited possibilities to define a stronger concept of equality on states collapsing all the bisimilar states into the same possibility. Finally, as with $m\mathcal{A}^\rho$ is more direct to have an implicit state-representation, using possibilities helps in reducing the search-space dimension.

In the near future we intend to implement a planner for $m\mathcal{A}^\rho$ and to study alternative representations. In particular we plan to: i) exploit more set-based operations: especially for the entailment of group operators; ii) formalize the concept of *non-consistent* belief for $m\mathcal{A}^\rho$; iii) investigate more thoroughly the connection between Kripke structures and non-well-founded sets; iv) examine the concept of bisimulation as equality between epistemic states; v) and finally consider other alternatives to Kripke structures, *e.g.*, *OBDD*s [9, 11].

References

1. Aczel, P.: Non-well-founded sets. CSLI Lecture Notes, 14 (1988)
2. Aucher, G., Bolander, T.: Undecidability in epistemic planning. pp. 27–33 (2013), cited By 27
3. Aumann, R., Brandenburger, A., et al.: Epistemic conditions for Nash equilibrium. ECONOMETRICA-EVANSTON ILL- **63**, 1161–1161 (1995)
4. Balliu, M., Dam, M., Le Guernic, G.: Epistemic temporal logic for information flow security. In: Proceedings of the ACM SIGPLAN 6th Workshop on Programming Languages and Analysis for Security. pp. 6:1–6:12. PLAS '11, ACM, New York, NY, USA (2011). https://doi.org/10.1145/2166956.2166962
5. Baral, C., Gelfond, G., Pontelli, E., Son, T.C.: An action language for multi-agent domains: Foundations. CoRR **abs/1511.01960** (2015), http://arxiv.org/abs/1511.01960
6. Barwise, J., Etchemendy, J.: The liar: An essay on truth and circularity. Oxford University Press (1987)
7. Bolander, T., Jensen, M., Schwarzentruber, F.: Complexity results in epistemic planning. vol. 2015-January, pp. 2791–2797 (2015)
8. Bolander, T., Andersen, M.B.: Epistemic planning for single-and multi-agent systems. Journal of Applied Non-Classical Logics **21**(1), 9–34 (2011)
9. Bryant, R.E.: Symbolic boolean manipulation with ordered binary-decision diagrams. ACM Computing Surveys (CSUR) **24**(3), 293–318 (1992)
10. Carbonell Jr, J.G.: Politics: Automated ideological reasoning. Cognitive Science **2**(1), 27–51 (1978)
11. Cimatti, A., Roveri, M.: Conformant planning via symbolic model checking. Journal of Artificial Intelligence Research **13**, 305–338 (2000)
12. Dovier, A.: Logic programming and bisimulation. In: Vos, M.D., Eiter, T., Lierler, Y., Toni, F. (eds.) ICLP. CEUR Workshop Proceedings, vol. 1433. CEUR-WS.org (2015), http://ceur-ws.org/Vol-1433
13. van Eijck, J.: Dynamic epistemic modelling. CWI. Software Engineering [SEN] (2004)
14. Fagin, R., Halpern, J.Y.: Reasoning about knowledge and probability. Journal of the ACM (JACM) **41**(2), 340–367 (1994)
15. Gerbrandy, J., Groeneveld, W.: Reasoning about information change. Journal of Logic, Language and Information **6**(2), 147–169 (1997). https://doi.org/10.1023/A:1008222603071

16. Gerbrandy, J.D.: Bisimulations on planet Kripke. Inst. for Logic, Language and Computation, Univ. van Amsterdam (1999)
17. Van der Hoek, W., Meyer, J.: A complete epistemic logic for multiple agents. In: Epistemic Logic and the Theory of Games and Decisions, pp. 35–68. Springer (1997)
18. Huang, X., Fang, B., Wan, H., Liu, Y.: A general multi-agent epistemic planner based on higher-order belief change. pp. 1093–1101 (2017)
19. Kominis, F., Geffner, H.: Beliefs in multiagent planning: From one agent to many. In: Proc. ICAPS. pp. 147–155 (2015)
20. Kripke, S.A.: Semantical considerations on modal logic. Acta Philosophica Fennica **16**(1963), 83–94 (1963)
21. Le, T., Fabiano, F., Son, T.C., Pontelli, E.: Efp and pg-efp: Epistemic forward search planners in multi-agent domains. In: Twenty-Eighth International Conference on Automated Planning and Scheduling (2018)
22. Liu, Q., Liu, Y.: Multi-agent epistemic planning with common knowledge. In: IJCAI. pp. 1912–1920 (2018)
23. Moore, R.C.: Reasoning about knowledge and action. In: Readings in Artificial Intelligence, pp. 473–477. Elsevier (1981)
24. Muise, C.J., Belle, V., Felli, P., McIlraith, S.A., Miller, T., Pearce, A.R., Sonenberg, L.: Planning over multi-agent epistemic states: A classical planning approach. In: Proc. of AAAI. pp. 3327–3334 (2015)
25. Prakken, H.: Logical tools for modelling legal argument: a study of defeasible reasoning in law, vol. 32. Springer Science & Business Media (2013)
26. Russell, S., Norvig, P.: Artificial Intelligence: A Modern Approach. Prentice Hall Press, Upper Saddle River, NJ, USA, 3rd edn. (2009)
27. Van Ditmarsch, H., van Der Hoek, W., Kooi, B.: Dynamic epistemic logic, vol. 337. Springer Science & Business Media (2007)
28. Van Harmelen, F., Lifschitz, V., Porter, B.: Handbook of knowledge representation, vol. 1. Elsevier (2008)
29. Wan, H., Yang, R., Fang, L., Liu, Y., Xu, H.: A complete epistemic planner without the epistemic closed world assumption. In: IJCAI (2015)

A $m\mathcal{A}^*$ and $m\mathcal{A}^\rho$ comparison

For a more clear comparison we show the execution, on both $m\mathcal{A}^*$ and $m\mathcal{A}^\rho$, of the action instances sequence $\Delta_c = \mathtt{distract}(\mathsf{C})\langle\mathsf{A}\rangle;\mathtt{open}\langle\mathsf{A}\rangle;\mathtt{peek}\langle\mathsf{A}\rangle$, that leads to the desired goal, in the domain expressed in Example 2. With this we want to give a graphical explanation of both the transition functions and state-space defined by the two languages. Each state in $m\mathcal{A}^*$ will be represented by a Kripke structure while in $m\mathcal{A}^\rho$ will be a possibility (expanded to its respective system of equations for clarity).

The observability relations of each action instance in Δ_c is expressed in the following Table:

	$\mathtt{distract}(\mathsf{C})\langle\mathsf{A}\rangle$	$\mathtt{open}\langle\mathsf{A}\rangle$	$\mathtt{peek}\langle\mathsf{A}\rangle$
F_D	A, B, C	A, B	A
P_D	-	-	B
O_D	-	C	C

Table 2: Observability relations of the actions instances in Δ_c.

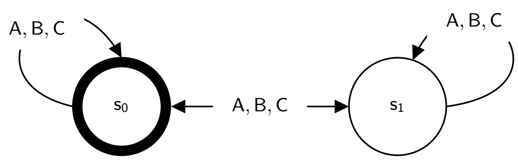
The initial state, based on Example 2, is defined by the conditions:

intially $\mathbf{C}(\mathtt{key}(\mathsf{A}))$
intially $\mathbf{C}(\neg\mathtt{key}(\mathsf{B}))$
intially $\mathbf{C}(\neg\mathtt{key}(\mathsf{C}))$
intially $\mathbf{C}(\neg\mathtt{opened})$
intially $\mathbf{C}(\neg\mathbf{B}_{\mathsf{ag}}\mathtt{heads} \wedge \neg\mathbf{B}_{\mathsf{ag}}\neg\mathtt{heads})$ for $\mathsf{ag} \in \{\mathsf{A}, \mathsf{B}, \mathsf{C}\}$
intially $\mathbf{C}(\mathtt{look}(\mathsf{ag}))$ for $\mathsf{ag} \in \{\mathsf{A}, \mathsf{B}, \mathsf{C}\}$
intially $\neg\mathtt{heads}$

Finally the goal that both the state in Figure 9 entail is expressed with the following formulae:

$$\mathbf{B}_{\mathsf{A}}\neg\mathtt{heads} \wedge \mathbf{B}_{\mathsf{A}}(\mathbf{B}_{\mathsf{B}}(\mathbf{B}_{\mathsf{A}}\mathtt{heads} \vee \mathbf{B}_{\mathsf{A}}\neg\mathtt{heads}))$$
$$\mathbf{B}_{\mathsf{B}}(\mathbf{B}_{\mathsf{A}}\mathtt{heads} \vee \mathbf{B}_{\mathsf{A}}\neg\mathtt{heads}) \wedge (\neg\mathbf{B}_{\mathsf{B}}\mathtt{heads} \wedge \neg\mathbf{B}_{\mathsf{B}}\neg\mathtt{heads})$$
$$\mathbf{B}_{\mathsf{C}}[\bigwedge_{\mathsf{ag}\in\{\mathsf{A},\mathsf{B},\mathsf{C}\}} (\neg\mathbf{B}_{\mathsf{ag}}\mathtt{heads} \wedge \neg\mathbf{B}_{\mathsf{ag}}\neg\mathtt{heads})]$$

The plan for Example 2



$M_0[\pi](\mathsf{s}_0) = \{\mathtt{look}(\mathsf{ag}), \mathtt{key}(\mathsf{A})\}$
$M_0[\pi](\mathsf{s}_1) = \{\mathtt{look}(\mathsf{ag}), \mathtt{key}(\mathsf{A}), \mathtt{heads}\}$
where $\mathsf{ag} \in \{\mathsf{A}, \mathsf{B}, \mathsf{C}\}$.

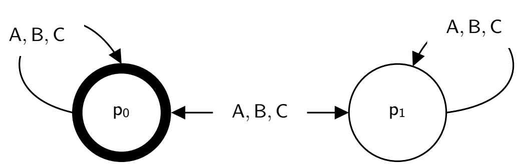
(a) The intial Kripke structure (M_0, s_0).

$$\begin{cases} \mathsf{u} & = \{(\mathsf{ag}, \{\mathsf{u}, \mathsf{u}'\}), \mathtt{look}(\mathsf{ag}), \mathtt{key}(\mathsf{A})\} \\ \mathsf{u}' & = \{(\mathsf{ag}, \{\mathsf{u}, \mathsf{u}'\}), \mathtt{look}(\mathsf{ag}), \mathtt{key}(\mathsf{A}), \mathtt{heads}\} \end{cases}$$

where $\mathsf{ag} \in \{\mathsf{A}, \mathsf{B}, \mathsf{C}\}$.

(b) The initial possibility u.

Figure 6: The initial state.



$M_1[\pi](\mathsf{p}_0) = \{\mathtt{look}(\mathsf{A}), \mathtt{look}(\mathsf{B}), \mathtt{key}(\mathsf{A})\}$
$M_1[\pi](\mathsf{p}_1) = \{\mathtt{look}(\mathsf{A}), \mathtt{look}(\mathsf{B}), \mathtt{key}(\mathsf{A}), \mathtt{heads}\}$

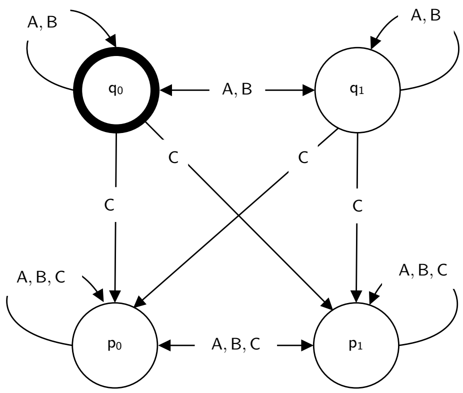
(a) The Kripke stucture (M_1, p_0), obtained after the execution of $\mathtt{distract}(\mathsf{C})\langle\mathsf{A}\rangle$ in (M_0, s_0) (Figure 6a).

$$\begin{cases} \mathsf{v} & = \{(\mathsf{ag}, \{\mathsf{v}, \mathsf{v}'\}), \mathtt{look}(\mathsf{A}), \mathtt{look}(\mathsf{B}), \mathtt{key}(\mathsf{A})\} \\ \mathsf{v}' & = \{(\mathsf{ag}, \{\mathsf{v}, \mathsf{v}'\}), \mathtt{look}(\mathsf{A}), \mathtt{look}(\mathsf{B}), \mathtt{key}(\mathsf{A}), \mathtt{heads}\} \end{cases}$$

where $\mathsf{ag} \in \{\mathsf{A}, \mathsf{B}, \mathsf{C}\}$

(b) Possibility v, obtained after the execution of $\mathtt{distract}(\mathsf{C})\langle\mathsf{A}\rangle$ in u (Figure 6b).

Figure 7: Execution of $\mathtt{distract}(\mathsf{C})\langle\mathsf{A}\rangle$.



$M_2[\pi](\mathsf{q}_0) = \{\mathtt{look}(\mathsf{ag}), \mathtt{key}(\mathsf{A}), \mathtt{opened}\}$ $M_2[\pi](\mathsf{p}_0) = M_1[\pi](\mathsf{p}_0)$
$M_2[\pi](\mathsf{q}_1) = \{\mathtt{look}(\mathsf{ag}), \mathtt{key}(\mathsf{A}), \mathtt{opened}, \mathtt{heads}\}$ $M_2[\pi](\mathsf{p}_1) = M_1[\pi](\mathsf{p}_1)$
where $\mathsf{ag} \in \{\mathsf{A}, \mathsf{B}\}$.

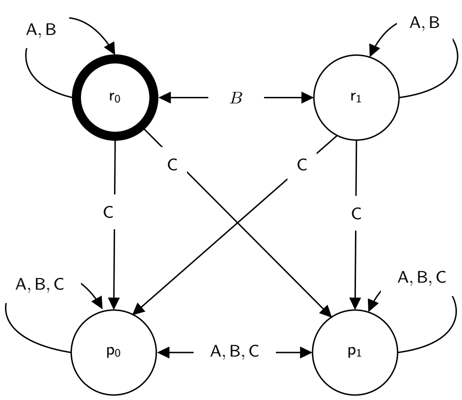
(a) The Kripke stucture (M_2, q_0), obtained after the execution of $\mathtt{open}\langle\mathsf{A}\rangle$ in (M_1, p_0) (Figure 7a).

$$\begin{cases} \mathsf{w} &= \{(\mathsf{ag}, \{\mathsf{w}, \mathsf{w}'\}), (\mathsf{C}, \{\mathsf{v}, \mathsf{v}'\}), \mathtt{look}(\mathsf{ag}), \mathtt{key}(\mathsf{A}), \mathtt{opened}\} \\ \mathsf{w}' &= \{(\mathsf{ag}, \{\mathsf{w}, \mathsf{w}'\}), (\mathsf{C}, \{\mathsf{v}, \mathsf{v}'\}), \mathtt{look}(\mathsf{ag}), \mathtt{key}(\mathsf{A}), \mathtt{opened}, \mathtt{heads}\} \end{cases}$$

where $\mathsf{ag} \in \{\mathsf{A}, \mathsf{B}\}$ and v, v', are the possibilities represented in Figure 7b.

(b) Possibility w, obtained after the execution of $\mathtt{open}\langle\mathsf{A}\rangle$ in v (Figure 7b).

Figure 8: Execution of $\mathtt{open}\langle\mathsf{A}\rangle$.



$M_3[\pi](\mathsf{r}_0) = \{\mathtt{look}(\mathsf{ag}), \mathtt{key}(\mathsf{A}), \mathtt{opened}\}$ $M_3[\pi](\mathsf{p}_0) = M_1[\pi](\mathsf{p}_0)$
$M_3[\pi](\mathsf{r}_1) = \{\mathtt{look}(\mathsf{ag}), \mathtt{key}(\mathsf{A}), \mathtt{opened}, \mathtt{heads}\}$ $M_3[\pi](\mathsf{p}_1) = M_1[\pi](\mathsf{p}_1)$
where $\mathsf{ag} \in \{\mathsf{A}, \mathsf{B}\}$.

(a) The Kripke structure (M_3, q_0), obtained after the execution of $\mathtt{peek}\langle\mathsf{A}\rangle$ in (M_2, q_0).

$$\begin{cases} \mathsf{z} &= \{(\mathsf{A}, \{\mathsf{z}\}), (\mathsf{B}, \{\mathsf{z}, \mathsf{z}'\})(\mathsf{C}, \{\mathsf{v}, \mathsf{v}'\}), \mathtt{look}(\mathsf{ag}), \mathtt{key}(\mathsf{A}), \mathtt{opened}\} \\ \mathsf{z}' &= \{(\mathsf{A}, \{\mathsf{z}'\}), (\mathsf{B}, \{\mathsf{z}, \mathsf{z}'\})(\mathsf{C}, \{\mathsf{v}, \mathsf{v}'\}), \mathtt{look}(\mathsf{ag}), \mathtt{key}(\mathsf{A}), \mathtt{opened}, \mathtt{heads}\} \end{cases}$$

where $\mathsf{ag} \in \{\mathsf{A}, \mathsf{B}\}$ and the possibilities v, v' are represented in Figure 7b.

(b) Possibility z, obtained after the execution of $\mathtt{peek}\langle\mathsf{A}\rangle$ in w (Figure 8b).

Figure 9: Execution of $\mathtt{peek}\langle\mathsf{A}\rangle$.