Backward Planning in the Logics of Communication and Change[⋆]

Pere Pardo and Mehrnoosh Sadrzadeh

[1] Institut d'Investigació en Intel·ligència Artificial (IIIA - CSIC)
[2] Dept. Computer Science, Oxford University

Abstract. In this contribution we study how to adapt Backward Plan search to the Logics of Communication and Change (LCC). These are dynamic epistemic logics with common knowledge modeling the way in which announcements, sensing and world-changing actions modify the beliefs of agents or the world itself. The proposed LCC planning system greatly expands the social complexity of scenarios involving cognitive agents that can be solved. For example, goals or plans may consist of a certain distribution of beliefs and ignorance among agents. Our results include: soundness and completeness of backward plan search (under BFS), both for deterministic planning and strong non-deterministic planning.

1 Introduction

Practical rationality or decision-making is a key component of autonomous agents, like humans, and correspondingly has been studied at large. Research on this topic has been conducted from several fields: game theory, planning, decision theory, etc. each focusing on a different aspect: strategic decision-making, propositional means-ends analysis, and uncertainty, respectively. Moreover, some of these models of rationality also provide a foundation for the notion of agreement (game-theoretic approaches to negotiation, multi-agent planning, etc). Indeed, the process of reaching an agreement seems to be a particular case of (multi-agent) practical decision-making.

While these models of practical rationality are well-understood, they were (understandably) designed with a considerably low level of expressivity at the object language. For instance, game-theory abstracts from propositional representations of states, actions and goals. Planning algorithms take them into account, but not fully in the object language, which makes it difficult to extend these algorithms to reason about others' beliefs and actions. All this contrasts with the area of logic, where logics with increasing expressivity have been characterized; though abilities or motivational attitudes have only more recently been dealt with, in BDI, logics of strategic ability, etc. In the present contribution we mainly focus on combining dynamic epistemic logic and plan search [9], [5].

Specially relevant to the topic of cognitive agents are the notions of belief, action, goal, norm, etc. The first two elements are the target of dynamic epistemic logics DEL, a recent family of logics for reasoning about agents' communications and observations. We focus on the so-called Logics of Communication and Change LCC [10],

[⋆] AT2012, 15-16 October 2012, Dubrovnik, Croatia. Copyright held by the author(s).

which generalize many previously known DEL logics with physical actions and common knowledge. In particular LCC logics include a rich variety of epistemic actions from the literature on DEL, including several types of communicative actions (truthful or lying, public or private announcements). This is done with the help of action models, that assign each action an accessibility relation describing how the action appears to other agents, and hence what they learn after its execution.

These logics may offer a semantic understanding of the notion of agreement between agents, since reaching an agreement requires these agents to have common knowledge about it. The same can be said about the epistemic aspects involved in the (intentional) process of reaching such an agreement. To this end, we address planning problems expressible in LCC. This is done by introducing goals, which can be an arbitrary formula in the language of LCC. Planning domains consist as usual of an initial state formula, available actions and a goal formula, and the task consists in finding some structure of actions that make the goal formula true when executed in the initial state. Instead of a semantic [3],[11],[7] or a proof-theoretic approach [1], we suggest to reduce plan search into search in the space of proofs. With more detail, search methods operate in the space of LCC formulas that are about plans (their success, and executability). As in those works, an LCC plan is built while having in mind what agents can learn during its execution.

In the present contribution, we extend previous results on Backward Planning by describing simpler algorithms for deterministic and strong non-deterministic planning. These are shown to be sound and complete: if the algorithm terminates, it outputs a solution plan; and if a successful plan exists, then the algorithm will find such a solution.

Motivating example. Our aim, then, is to endow LCC reasoning agents with planning capacities for these logics, so they can achieve their goals in scenarios where other agents have similar cognitive and acting abilities. In particular, LCC planning seems necessary for an agent whose goals consist in (or depend on) a certain distribution of knowledge, ignorance and false beliefs among agents. To illustrate the kind of rational behavior an LCC planner can exhibit, consider the following example.

Example 1. Agent $\mathbf{a}$ just bet some prize agent $\mathbf{b}$ that the next coin toss will land heads ($\mathbf{h}$); $\mathbf{a}$ knows she can sense and even flip the coin without $\mathbf{b}$ ever suspecting it. Given a sensing action that tells $\mathbf{a}$ whether h holds or not, a successful plan seems to be: toss the coin; if sense that $\mathbf{h}$, then announce it to $\mathbf{b}$; otherwise flip the coin and announce h.

Structure of the paper. The paper is structured as follows: after a review of the related work in Section 2, we briefly present the dynamic epistemic logics in LCC in Section 3. Then in Section 4, we present an algorithm for deterministic planning in LCC based on Breadth First Search (BFS) and show it is sound and complete. The rest of the paper is devoted to non-deterministic planning: Section 5 we present an extension of LCC with action composition and choice. Finally, in Section 6 we propose an algorithm for (strong) non-deterministic and show its soundness and completeness. The paper concludes with a summary and some topics for future work.

2 Related Work

Several logics address the related problem of plan existence (rather than plan search) for similarly expressive languages. For example, arbitrary public announcement logics APAL [4], and logics for strategic ability. These logics contain modal operators $\langle C \rangle$ for agents or coalitions, with formulas $\langle C \rangle \varphi$ expressing: there is a plan or strategy allowing agents in C to enforce φ.

More in line with the present aims, among logics for action guidance we find the family of BDI and related logics for intention (or some other motivational attitudes). While these logics usually allow for considerable expressivity w.r.t. intention (and their interaction with beliefs), they are not completely understood, and in fact planning methods have been suggested for BDI agent architectures. In particular, Bolander et al. [3] suggest the use of LCC planning to this end for (a limited fragment of) BDI. In this work [3], the authors study incremental forward planning based on the semantics of update models. By *incremental* we mean that the plan is built in a step-wise fashion: plans are being refined (with further actions) until a solution is found. The framework of [3] is shown to be sound and complete under Breadth First Search, and hence the proposed planning system for LCC becomes semi-decidable. Here we study the opposite direction of search, i.e. backward planning (from the goals to the initial state). The reason is that in forward planning, relevant actions to be considered are those that are *currently executable* actions to be the notion of relevant action, rather than *current goal enforcing* actions, as in backward planning. This makes a difference in LCC since many actions may exist (in the action model) which are everywhere executable, so forward planning would typically face the state explosion problem. Along this line, Aucher [1] (and related papers) introduce regression methods for the fragment of LCC without common knowledge. The present work differs from the latter two by making plan search incremental (as in [3], [11]), but allowing for backward search. The proposed algorithms easily adapt to search for optimal plans in backward planning.

3 Preliminaries: The Logics of Communication and Change

We briefly recall the basics of LCC logics [10]. Since these are built by adding an action model U on top of (an epistemic reading of) propositional dynamic logic PDL, we recall the latter first. The syntax of PDL, denoted $\mathcal{L}_{\mathrm{PDL}}$, is as follows:

$$\begin{aligned} \varphi &::= p \mid \neg\varphi \mid \varphi_1 \wedge \varphi_2 \mid [\pi]\varphi \\ \pi &::= a \mid ?\varphi \mid \pi_1 ; \pi_2 \mid \pi_1 \cup \pi_2 \mid \pi^* \end{aligned}$$

As usual, the symbols $\bot, \vee, \leftrightarrow$ and $\langle \pi \rangle$ are defined from the above as abbreviations.

In the epistemic reading of PDL, denoted as E·PDL, the set of atomic programs a is just the set of agents Ag, and we read the program $[a]$ as *agent a believes (or knows) that*.[3] With the PDL program constructors (composition ";", choice "$\cup$" and the Kleene star "$(\cdot)^*$", we can model:

[3] E·PDL, and LCC, are rather abstract about the particular properties of the accessibility R_a relation of agent a, that distinguish belief from knowledge.

nested beliefs	$[a;b]$	*agent a believes agent b believes that*
group belief	$[B]$, or $[a \cup b]$	*agents in $B (= \{a,b\})$ believe that*
common knowledge	$[B^*]$, or $[(a \cup b)^*]$	*agents in B have common knowledge that*

An E·PDL model $M = (W, \langle R_a \rangle_{a \in \mathsf{Ag}}, V)$ does, as usual, contain a set of worlds W', a relation R_a for each agent a, and an evaluation $V : \mathsf{Var} \to \mathcal{P}(W)$).

An LCC logic will add to an E·PDL language a set of modalities $[\mathsf{U}, \mathsf{e}]$ for each pointed action model U, e with distinguished (actual) action e. These new operators $[\mathsf{U}, \mathsf{e}]$ read *after each execution of action* e *it is the case that*. An action model is a tuple $\mathsf{U} = (\mathsf{E}, \mathsf{R}, \mathsf{pre}, \mathsf{post})$ containing

- $\mathsf{E} = \{\mathsf{e}_0, \ldots, \mathsf{e}_{n-1}\}$, a set of actions
- $\mathsf{R} : \mathsf{Ag} \to (\mathsf{E} \times \mathsf{E})$, a map assigning a relation R_a to each agent $a \in \mathsf{Ag}$
- $\mathsf{pre} : \mathsf{E} \to \mathcal{L}_{\mathrm{PDL}}$, a map assigning a precondition $\mathsf{pre}(\mathsf{e})$ to each action e
- $\mathsf{post} : \mathsf{E} \times \mathsf{Var} \to \mathcal{L}_{\mathrm{PDL}}$, a map assigning a post-condition $\mathsf{post}(\mathsf{e})(p)$, or $p^{\mathsf{post}(\mathsf{e})}$, to each $\mathsf{e} \in \mathsf{E}$ and $p \in \mathsf{Var}$

The semantics of LCC consists in computing $M, w \models [\mathsf{U}, \mathsf{e}]p$ in terms of the product update of M, w and U, e. Their product update is (again) an E·PDL pointed model $M \circ \mathsf{U}, (w, \mathsf{e})$, where $M \circ \mathsf{U} = (W', \langle R'_a \rangle_{a \in \mathsf{Ag}}, V')$ is as follows. The set W' consists of those worlds (w, e) such that $M, w \models \mathsf{pre}(\mathsf{e})$, so executing e will lead to the corresponding state (w, e). The relation $(w, \mathsf{e}) R'_a (v, \mathsf{f})$ holds iff both $w R_a v$ and $\mathsf{e} \mathsf{R}_a \mathsf{f}$ hold. Finally, $V'(p) = \{(w, \mathsf{e}) \in W' \mid M, w \models \mathsf{post}(\mathsf{e})(p)\}$, i.e. the truth-value of p after executing e depends on that of $\mathsf{post}(\mathsf{e})(p)$ before the execution.

Here, though, we will follow [14] and restrict postconditions to $\mathsf{post}(\mathsf{e})(p) \in \{p, \top, \bot\}$. This makes the truth-value of p after e to be either: that before e, or simply true or false irregardless of its truth-value before e. (This framework is shown in [14] to be as expressive as the general case above, in case choice of actions is considered. See Section 6.) The PDL semantics $[\![\cdot]\!]$ for E·PDL-formulas can be extended to a semantics for LCC as follows:

$$[\![[\mathsf{U}, \mathsf{e}]\varphi]\!]^M = \{ w \in W \mid \text{ if } M, w \models \mathsf{pre}(\mathsf{e}) \text{ then } (w, \mathsf{e}) \in [\![\varphi]\!]^{M \circ \mathsf{U}} \}.$$

In [10], the authors define program transformers $T^{\mathsf{U}}_{ij}(\pi)$ provide a mapping between E·PDL programs. Given any combination of ontic or epistemic actions (e.g. public and private announcements) the transformers provide a complete set of reduction axioms, reducing LCC to E·PDL. These axioms push the U, e-modalities inside the formula, and then the basic case $[\mathsf{U}, \mathsf{e}]p$ is turned into an E·PDL formula. See [10] for the details on the $T^{\mathsf{U}}(\pi)$ mapping.

Definition 1. *A calculus for* LCC *is given by the axioms and rules for* PDL *plus* $[\mathsf{U}, \mathsf{e}]\top \leftrightarrow \top$ *and the following*

$$\begin{array}{ll} [\mathsf{U}, \mathsf{e}]p \leftrightarrow (\mathsf{pre}(\mathsf{e}) \to \mathsf{post}(e)(p)) & [\mathsf{U}, \mathsf{e}]\neg\varphi \leftrightarrow (\mathsf{pre}(\mathsf{e}) \to \neg[\mathsf{U}, \mathsf{e}]\varphi) \\ [\mathsf{U}, \mathsf{e}](\varphi_1 \wedge \varphi_2) \leftrightarrow ([\mathsf{U}, \mathsf{e}]\varphi_1 \wedge [\mathsf{U}, \mathsf{e}]\varphi_2) & [\mathsf{U}, \mathsf{e}_i][\pi]\varphi \leftrightarrow \bigwedge_{j=0}^{n-1} [T^{\mathsf{U}}_{ij}(\pi)][\mathsf{U}, \mathsf{e}_j]\varphi \end{array}$$

and necessitation rules for each action model modality: if $\vdash \varphi$ *then* $\vdash [\mathsf{U}, \mathsf{e}]\varphi$*.*

The completeness for this calculus is shown by reducing LCC to E·PDL. The translation simultaneously defined for formulas and programs, resp. t and r, is

$$t([\mathsf{U},\mathsf{e}]p) = t(\mathsf{pre}(\mathsf{e})) \to p^{\mathsf{post}(\mathsf{e})} \qquad t([\mathsf{U},\mathsf{e}_i][\pi]\varphi) = \bigwedge_{j=0}^{j=n-1}[T^{\mathsf{U}}_{ij}(r(\pi))]t([\mathsf{U},\mathsf{e}_j]\varphi)$$
$$t([\mathsf{U},\mathsf{e}]\neg\varphi) = t(\mathsf{pre}(\mathsf{e})) \to \neg t(\varphi) \qquad t([\mathsf{U},\mathsf{e}][\mathsf{U},\mathsf{f}]\varphi) = t([\mathsf{U},\mathsf{e}]t([\mathsf{U},\mathsf{f}]\varphi))$$

r is trivial for all programs, and t is also trivial for the remaining types of formula. The translation assigns to each $\mathcal{L}_{\mathrm{LCC}}$ formula an equivalent formula in $\mathcal{L}_{\mathrm{E\cdot PDL}}$.

4 Backward deterministic LCC planning.

In [8], we studied LCC planning with the help of generalized frame axioms for LCC (actually, theorems in this logic) used to study the regression of an open goal under an action. Here we propose a simpler approach, based on the above reduction of LCC into E·PDL; and moreover they allow for better plan search methods. We will simply use the expressions of the form

$$t([\mathsf{U},\mathsf{e}]\varphi \wedge \langle \mathsf{U},\mathsf{e}\rangle\top) \to [\mathsf{U},\mathsf{e}]\varphi \wedge \langle \mathsf{U},\mathsf{e}\rangle\top$$

These give rise to a goal-regression function: a current goal φ is mapped into $t([\mathsf{U},\mathsf{e}]\varphi \wedge \langle \mathsf{U},\mathsf{e}\rangle\top)$; the latter is the new goal we need to address for an execution of e to end up in a φ-world.

Definition 2. *Given some* LCC *logic for an action model* U*, a* planning domain *is a triple* $\mathbb{M} = (\varphi_T, A, \varphi_G)$*, where* φ_T, φ_G *are consistent* E·PDL *formulas describing, resp., the initial and goal states; and* $A \subseteq \mathsf{E}$ *is the subset of a actions available to the agent.*

A solution *to* $\mathbb{M}$ *is a sequence* $\pi = (\mathsf{f}_1, \ldots, \mathsf{f}_m)$ *of actions in A, such that*

$$\models \varphi_T \to [\mathsf{U},\mathsf{f}_1]\ldots[\mathsf{U},\mathsf{f}_m]\varphi_G \quad \textit{and} \quad \models \varphi_T \to \langle \mathsf{U},\mathsf{f}_1\rangle\ldots\langle \mathsf{U},\mathsf{f}_m\rangle\top$$

Definition 3. *Given some planning domain* $\mathbb{M} = (\varphi_T, A, \varphi_G)$*, the (initial) empty plan is the pair* $\pi_\emptyset = (\emptyset, \varphi_G)$ *and if* $\pi = (\pi, \varphi_{\mathsf{goals}(\pi)})$ *is a plan, then* $\pi(\mathsf{e}) = (\pi^\frown\langle \mathsf{e}\rangle, \varphi_{\mathsf{goals}}(\pi(\mathsf{e})))$*, defined by the goal* $\varphi_{\mathsf{goals}(\pi(\mathsf{e}))} = t([\mathsf{U},\mathsf{e}]\varphi_{\mathsf{goals}(\pi)} \wedge \langle \mathsf{U},\mathsf{e}\rangle\top)$*, is also a plan.A plan* π *is a* leaf *iff* $\varphi_{\mathsf{goals}(\pi(\mathsf{e}))}$ *is inconsistent, or* $\models \varphi_{\mathsf{goals}(\pi(\mathsf{e}))} \to \varphi_{\mathsf{goals}(\pi)}$.

Leafs are plans not worth considering, either because unexecutable or non-contributing to the ultimate goals φ_G. Also note that actions e -as defined above- are deterministic, in the sense that $\models [\mathsf{U},\mathsf{e}]\varphi \vee \psi \leftrightarrow [\mathsf{U},\mathsf{e}]\varphi \vee [\mathsf{U},\mathsf{e}]\psi$. Thus, *deterministic planning* simply consists in "restricting" actions $\mathsf{e} \in \mathsf{E}$ to our current action models, or simply LCC planning. Later we extend LCC with composition $\otimes$ and choice $\cup$ to study the non-deterministic case. A BFS plan search algorithm for deterministic planning is the following.

Theorem 1. *BFS is sound and complete for* LCC *backward planning: the output* π *of the algorithm in Fig. 1 is a solution for* $(\varphi_T, A, \varphi_G)$*, and is such a solution exists, then the algorithm terminates with such an output.*

```
1. LET Plans = $\langle \pi_\emptyset \rangle$ and $\varphi = \varphi_G$.
2. SELECT the first element $\pi$ of Plans. DELETE $\pi$ from Plans.
3. If $\pi$ is a leaf REPEAT step 2. If $\models T \to \mathsf{goals}(\pi)$, then OUTPUT$\pi$.
     Otherwise SET Plans $\leftarrow$ Plans$^\frown\langle \pi(\mathsf{e}) \mid \mathsf{e} \in A \rangle$. REPEAT step 2.
```

Fig. 1. BFS algorithm for deterministic planning with input $(\varphi_T, A, \varphi_G)$.

4.1 Optimal planning.

We may consider as well a cost function $\mathsf{cost} : \mathsf{E} \to \mathbb{R}^+$ assigning to e the cost of an execution of e.[4] This gives a measure of the total cost of a plan $\mathsf{cost}(\pi) = \Sigma_{\mathsf{e} \in \pi} \mathsf{cost}(\mathsf{e})$.

Recall search for optimal plans (e.g. A*) can be defined from BFS in Fig. 1 by adding an extra step in step 3: after updating Plans, this set is re-ordered into a sequence of plans with increasing cost. That is, π is re-ordered before π' in Plans if and only if $\mathsf{cost}(\pi) \leq \mathsf{cost}(\pi')$. Since by definition $\mathsf{cost}(\mathsf{e}) > 0$, and the set E is finite, there can be no infinite sequence of actions with decreasing cost. This implies (using [9] Thm 1) that A* is complete and admissible for deterministic backward planning in LCC using the (trivial) heuristic estimation $h(\pi(\mathsf{e})) = \mathsf{cost}(\pi) + \mathsf{cost}(\mathsf{e})$.

Corollary 1. *A* is sound, and complete for* LCC *backward deterministic planning. It is also admissible: the output of A* is an optimal solution π for $(\varphi_T, A, \varphi_G)$, if some solution exists.*

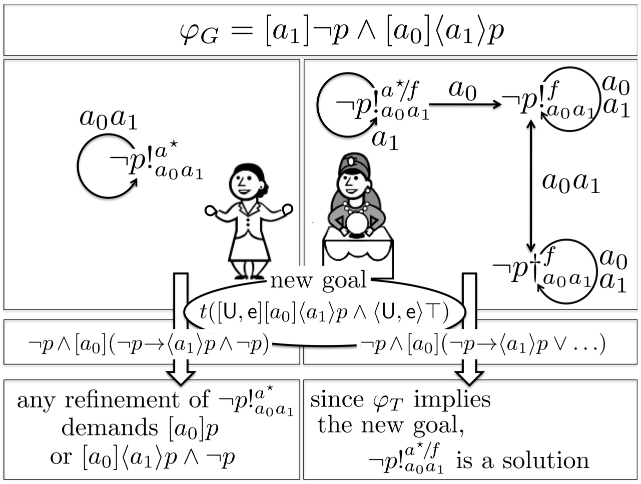
$\varphi_G = [a_1]\neg p \wedge [a_0]\langle a_1 \rangle p$
$a_0 a_1$
$\neg p!^{a^\star}_{a_0 a_1}$
$\neg p!^{a^\star/f}_{a_0 a_1}$
a_0
$\neg p!^{f}_{a_0 a_1}$
a_0
a_1
a_1
$a_0 a_1$
$\neg p\dagger^{f}_{a_0 a_1}$
a_0
a_1
new goal
$t([\mathsf{U},\mathsf{e}][a_0]\langle a_1\rangle p \wedge \langle \mathsf{U},\mathsf{e}\rangle\top)$
$\neg p \wedge [a_0](\neg p \to \langle a_1 \rangle p \wedge \neg p)$
$\neg p \wedge [a_0](\neg p \to \langle a_1 \rangle p \vee \ldots)$
any refinement of $\neg p!^{a^\star}_{a_0 a_1}$
demands $[a_0]p$
or $[a_0]\langle a_1 \rangle p \wedge \neg p$
since φ_T implies
the new goal,
$\neg p!^{a^\star/f}_{a_0 a_1}$ is a solution

Fig. 2. (Left) Announcing $\neg p$ as insider in Example 2. (Right) Announcing $\neg p$ disguised as a fortune-teller.

[4] For example, the energy cost of voicing an announcement that φ into a crowd, $\varphi!^a_B$, can be defined in terms of $|B|$ and the length of φ.

Example 2. Suppose you are a high-ranked employee $a^\star$ in *Phil. Inc* and want to sell some sensible information, $\neg p$ = *Phil. Inc is **not** doing well*, to a close acquaintance a_1, an important investor in *Phil. Inc*. It is commonly ignored that $\neg p$, except for you and a_0, also an important investor in *Phil. Inc*. Knowing $\neg p$, a_0 wants to sell stocks before a_1 does, but she prefers to wait for the price to increase a bit more. As long as $[a_0]\langle a_1\rangle p$ holds, you can sell the info, so your actions e should preserve this formula while causing $[a_1]\neg p$. To make things more difficult, say agents a_0, a_1 are sitting next to each other.

Figure 2 (Left) models your truthful announcement of $\neg p$, presenting yourself *as insider* $a^\star$; that is, $\neg p!^{a^\star}_{a_1}$. Intuitively, this would not preserve $[a_0]\langle a_1\rangle p$; and indeed, available actions would either destroy the old goal or would not be successful about the new goal.

Figure 2 (Right) models $\neg p!^{a^\star/f}_{a_1}$: your truthful announcement that $\neg p$ when disguised as a fortune-teller. You expect a_1 will identify you as $a^\star$, while a_0 will think it common belief that you are a fortune-teller (and that no one can tell a fortune-teller is right or not). The new goal raised by this refinement is already true in any φ_T state. So this action already constitutes a solution to the problem.

5 LCC with composition and choice

In this section we introduce non-deterministic actions into planning models, and study the corresponding logic LCC$_{\otimes\cup}$. This will be used in Section 6 to find an algorithm for non-deterministic planning in this logic. First, we expand any LCC logic with action composition $\otimes$, and later we add choice $\cup$. These operations model the following executions:

- composition $\mathsf{e} \otimes \mathsf{f}$ models an execution of e followed by an execution of f, and
- choice $\mathsf{e} \cup \mathsf{f}$, models non-deterministic actions: each execution of $\mathsf{e} \cup \mathsf{f}$ either instantiates as an execution of e or as an execution of f.

We introduce first composition $\otimes$ in the action models U and define the logic LCC$_{\otimes}$, which can be reduced to LCC. Then, we will introduce choice $\cup$ and its semantics, the latter be based on the semantics of LCC$_{\otimes}$.

5.1 LCC$_{\otimes}$: extending LCC with composition of actions.

On the one hand, the introduction of composition $\otimes$ proceeds as usual. We will simply consider that the action model U is now closed under finite composition $\otimes$. To do so, we must define first the composition of an action model by itself $\mathsf{U} \otimes \mathsf{U}$, also denoted $\mathsf{U}^2 = (\mathsf{E}^2, \mathsf{pre}^2, \mathsf{post}^2, \mathsf{R}^2)$.

Definition 4. *Let* $\mathsf{U} = (\mathsf{E}, \mathsf{R}, \mathsf{pre}, \mathsf{post})$ *be an action model. We define* $\mathsf{U} \otimes \mathsf{U} = (\mathsf{E} \times \mathsf{E}, \mathsf{R} \circ \mathsf{R}, \mathsf{pre}, \mathsf{post})$ *as follows:*

$$\mathsf{pre}(\mathsf{e} \otimes \mathsf{e}') = \mathsf{pre}(\mathsf{e}) \wedge [\mathsf{U}, \mathsf{e}]\mathsf{pre}(\mathsf{e}')$$

$$\mathsf{post}(\mathsf{e} \otimes \mathsf{e}')(p) = \begin{cases} \mathsf{post}(\mathsf{e})(p) & \textit{if}\ \mathsf{post}'(\mathsf{e}')(p) = p \\ \mathsf{post}'(\mathsf{e}')(p) & \textit{if}\ \mathsf{post}'(\mathsf{e}')(p) \in \{\top, \bot\} \end{cases}$$

Lemma 1. *We have the following isomorphism*

$$M \circ (\mathsf{U} \otimes \mathsf{U}) \cong (M \circ \mathsf{U}) \circ \mathsf{U}.$$

Definition 5. *The axioms and rules corresponding to* $\mathrm{LCC}_{\otimes}$ *are those of* LCC *plus: axiom* K *and Necessitation for* $\mathsf{e} \otimes \mathsf{f}$ *and the reduction axiom:*

$$[\mathsf{U}, \mathsf{e} \otimes \mathsf{f}]\varphi \leftrightarrow [\mathsf{U}, \mathsf{e}][\mathsf{U}, \mathsf{f}]\varphi$$

The LCC calculus extends to $\mathrm{LCC}_{\otimes}$ using this reduction axiom. Moreover, the previous Lemma 1 extends to the general update $\mathsf{U}^{\otimes}$ with compositions $\mathsf{e} \otimes \cdots \otimes \mathsf{f}$ of at most n actions in E, for any fixed finite number n. (This is done by taking disjoint unions of update models with n' compositions, for each $n' \leq n$.) The extensions of LCC for these compositional action models (defined by the corresponding axioms) will also be denoted $\mathrm{LCC}_{\otimes}$. Having such a bound n on composition means in practice that we will not know a priori which $\mathrm{LCC}_{\otimes}$ is actually needed to solve the planning problem. Only after the plan search algorithm terminates, we can identify this logic. A similar remark applies to the logical extensions $\mathrm{LCC}_{\otimes\cup}$ studied next.

5.2 $\mathrm{LCC}_{\otimes\cup}$: extending LCC with composition and choice.

Now we will introduce the choice for actions $\mathsf{e} \cup \mathsf{f}$, denoting an indeterminate choice between e and f. Note if the action $\mathsf{e} \cup \mathsf{f}$ is available (in A), then in general e, f will be in E but not in A. As suggested in [10], we adopt the (multi-pointed) semantics and axioms to model choice in an extension of LCC. (Note in the previous semantics we cannot extend $\mathsf{post}(\cdot)(p)$ to the choice of ontic actions $\mathsf{e} \cup \mathsf{f}$ if these disagree about $\mathsf{post}(\cdot)(p)$.)

Definition 6. *Given an epistemic model* M *and an action model* U*, let* $W_d \subseteq W$ *and* $\mathsf{E}_d = \{\mathsf{f}_1, \ldots, \mathsf{f}_k\} \subseteq \mathsf{E}$*. Then* M, W_d *and* U, E_d *are multi-pointed models. We define*

$$\begin{array}{lll} M, W_d \models \varphi & \textit{iff} & M, w \models \varphi \quad \textit{for each } w \in W_d \\ M, w \models [\mathsf{U}, \mathsf{E}_d]\varphi & \textit{iff} & \bigwedge_{\mathsf{f} \in \mathsf{E}_d} M, w \models [\mathsf{U}, \mathsf{f}]\varphi \\ & \textit{iff} & M, w \models \mathsf{pre}(\mathsf{f}) \textit{ implies } M \circ \mathsf{U}, (w, \mathsf{f}) \models \varphi\textit{, for each } \mathsf{f} \in \mathsf{E}_d \end{array}$$

Choice will be indistinctly represented as follows E_d*,* $\{\mathsf{e}, \ldots, \mathsf{f}\}$ *or* $\mathsf{e} \cup \ldots \cup \mathsf{f}$*.*

In summary, the semantics of update with a multi-pointed action model $\mathsf{U}, \mathsf{e} \cup \mathsf{f}$ can be defined by the multi-pointed epistemic model $M \circ \mathsf{U}, \{(w, \mathsf{e}), (w, \mathsf{f})\}$. The semantics of $[\mathsf{U}, \mathsf{e} \cup \mathsf{f}]$ from Definition 6 then becomes $M, w \models [\mathsf{U}, \{\mathsf{e}, \mathsf{f}\}]\varphi$ iff

$$M, w \models \mathsf{pre}(\mathsf{e}) \Rightarrow M \circ \mathsf{U}, (w, \mathsf{e}) \models \varphi \text{ and } M, w \models \mathsf{pre}(\mathsf{f}) \Rightarrow M \circ \mathsf{U}, (w, \mathsf{f}) \models \varphi$$

So in case $M, w \models \mathsf{pre}(\mathsf{e}) \wedge \mathsf{pre}(\mathsf{f})$, the two RHS conditions reduce to satisfaction in a multi-pointed epistemic model: $M \circ \mathsf{U}, \{(w, \mathsf{e}), (w, \mathsf{e}')\} \models \varphi$. In fact, for simplicity here we only consider the case $\mathsf{pre}(\mathsf{e}) = \mathsf{pre}(\mathsf{f})$, for any (available) non-deterministic action $\mathsf{e} \cup \mathsf{f}$.

Also note that $\mathsf{e} \cup \mathsf{f}$ is not an action in the set E of the action model U or $\mathsf{U}^{\otimes}$. The syntax of $\mathrm{LCC}_{\otimes\cup}$ extends that of $\mathrm{LCC}_{\otimes}$ with the further type of formulas $[\mathsf{U}, \mathsf{E}_d]\varphi$, for each $\mathsf{E}_d \subseteq \mathsf{E}$ (whose elements have identical preconditions). In [10], the next reduction axiom was suggested for $\cup$:

$$[\mathsf{U},\mathsf{E}_d]\varphi \leftrightarrow \bigwedge_{\mathsf{e}\in\mathsf{E}_d}[\mathsf{U},\mathsf{e}]\varphi$$

Definition 7. *The axioms and rules corresponding to* LCC$_{\otimes\cup}$ *are those of* LCC$_{\otimes}$ *plus: axiom* K *and Necessitation for* $[\mathsf{U},\mathsf{e}\cup\ldots\cup\mathsf{f}]$*, and the previous reduction axiom, i.e.:*

$$[\mathsf{U},\mathsf{e}\cup\ldots\cup\mathsf{f}]\varphi \leftrightarrow [\mathsf{U},\mathsf{e}]\varphi\wedge\ldots\wedge[\mathsf{U},\mathsf{f}]\varphi$$

Fact 1 *The usual* LCC *reduction axioms for* $[\mathsf{U},\mathsf{e}]$ *are derivable for* $[\mathsf{U},\mathsf{e}\cup\mathsf{e}']$*, except for* $[\mathsf{U},\mathsf{e}\cup\mathsf{f}]\neg\varphi \leftrightarrow (\mathsf{pre}(\mathsf{e}\cup\mathsf{f})\to\neg[\mathsf{U},\mathsf{e}\cup\mathsf{f}]\varphi)$.

Proposition 1. *The axioms and rules from Def. 7 are sound and complete for* LCC$_{\otimes\cup}$.

6 Non-Deterministic Planning

Now we turn into strong planning for domains with non-deterministic actions, i.e. planning involving actions with (truly) disjunctive effects

$$\models [\mathsf{U},\mathsf{f}_0\cup\mathsf{f}_1]p\vee q, \quad \text{but with} \quad \not\models [\mathsf{U},\mathsf{f}_0\cup\mathsf{f}_1]p \quad \text{and} \quad \not\models [\mathsf{U},\mathsf{f}_0\cup\mathsf{f}_1]q$$

(For this, let $\mathsf{post}(\mathsf{f}_0)(p) = \mathsf{post}(\mathsf{f}_1)(q) = \top$, and $\mathsf{post}(\mathsf{f}_0)(q) = q$ and $\mathsf{post}(\mathsf{f}_1)(p) = p$). A strong solution for a given planning domain is a plan such that all of its executions in the initial state lead to a goal state. Thus, ignoring preconditions, the above action $\mathsf{f}_0\cup\mathsf{f}_1$ is a strong solution to $(\varphi_T,\{\mathsf{f}_0\cup\mathsf{f}_1\},\varphi_G)$, if $\varphi_G = p\vee q$; and it is a weak solution if $\varphi_G = p$.) Consider the action of *tossing a coin*, whose effect is *heads or tails*, denoted $h\vee\neg h$. We represent it as a choice between two actions: toss_h, $\mathsf{toss}_{\neg h}$, with postconditions

$$\mathsf{post}(\mathsf{toss}_h)(h) = \top, \quad \text{and } \mathsf{post}(\mathsf{toss}_{\neg h})(h) = \bot$$

Note both toss_h and $\mathsf{toss}_{\neg h}$ are deterministic actions in E, though neither of them will be in A. The reason is that they are not individually available to the agent: only $\mathsf{toss}(h)\cup\mathsf{toss}(\neg h)$ models this agent's abilities and hence only the latter can be used during plan search. On the other hand, $\mathsf{e}\cup\mathsf{f}$ is an element of A but not of E. For simplicity, we only define the choice between two actions $\mathsf{f}_0\cup\mathsf{f}_1$. The following definitions easily generalize to the choice of finitely many actions $\mathsf{e}\cup\cdots\cup\mathsf{f}$.

First we must redefine solution for the non-deterministic case; the reasons for this are the generalization of available actions A to the non-deterministic case, and the the limited reduction axioms that characterize $\mathsf{U},\mathsf{e}\cup\mathsf{f}$ modalities.

Definition 8. *Given an action model* U *in* LCC*, let* A *be a set of actions* e *in* E *and actions of the form* $\mathsf{f}_0\cup\mathsf{f}_1$*, where* $\mathsf{f}_0,\mathsf{f}_1\in\mathsf{E}$*. We define a set* $A^{\otimes\cup}$ *of* A-modalities *in the extended language of* LCC$_{\otimes\cup}$ *as follows:*

$$\begin{array}{rll}
(A) & (\mathsf{E}\cap A)^{<\omega}\subseteq A^{\otimes\cup} & \\
(\cup) & \textit{if } \mathsf{e},\mathsf{e}'\in A^{\otimes\cup}, & \textit{then } \mathsf{e}\cup\mathsf{e}'\in A^{\otimes\cup} \\
(\otimes L) & \textit{if } \mathsf{e}\in A^{\otimes\cup} \textit{ and } \mathsf{e}'\in(\mathsf{E}\cap A)^{<\omega}, & \textit{then } \mathsf{e}'\otimes\mathsf{e}\in A^{\otimes\cup} \\
(\otimes R) & \textit{if } \mathsf{e}\in A^{\otimes\cup} \textit{ and } \mathsf{e}'\in(\mathsf{E}\cap A)^{<\omega}, & \textit{then } \mathsf{e}\otimes\mathsf{e}'\in A^{\otimes\cup}
\end{array}$$

If e *does not make use of* $(\otimes R)$ *we say it is in* search form. *If* e *does not make use of* $(\otimes L)$ *or* $(\otimes R)$ *we say it is in* branching form.

See Figure 3 for an illustration of these and other modalities. Note a modality U, e in branching form essentially consists in a conjunction of $\mathrm{LCC}_{\otimes}$ modalities. The output of a plan search algorithm for $\mathrm{LCC}_{\otimes\cup}$ will be an A-modality U, e in tree form. But let us define first when an A-modality is a solution for a planning domain $\mathbb{M}$.

Definition 9. *Let the* $\mathrm{LCC}_{\otimes\cup}$ *logic of some action model* $\mathsf{U}(=\mathsf{U}^{\otimes})$ *be given, plus a set of available actions A. Let* $\mathbb{M} = (\varphi_T, A, \varphi_G)$ *be some planning domain definable in this logic. We say an action* $\mathsf{e} \in A^{\otimes\cup}$ *is a* solution *for* $\mathbb{M}$ *iff*

$$(1) \models \varphi_T \to [\mathsf{U}, \mathsf{e}]\varphi_G \quad \textit{and} \quad (2) \models \bigwedge_{\overrightarrow{\mathsf{e}} \text{ a path in } \Pi} \varphi_T \to \langle \mathsf{U}, \overrightarrow{\mathsf{e}} \rangle \top$$

The only difference with Definition 2 lies in executability: we check that every possible instantiation $\overrightarrow{\mathsf{e}}$ of e does indeed lead to some state (and so, by (1), the actual instantiation will lead to a goal state).

For (strong) non-deterministic planning, the idea is to reduce the search for a non-deterministic output Π to a tree of deterministic solutions $\Pi = (\{\pi, \ldots\}, \prec)$. With more detail, the planning domain $\mathbb{M} = (\varphi_T, A, \varphi_G)$ will split into a tree of planning domains $\mathbb{M}^{\xi}$, each to with its own deterministic solution π^{ξ}. For this, we need to enrich the notation $\pi^{\xi} = (\pi^{\xi}, \varphi_{\mathsf{goals}(\pi^{\xi})})$ into a 4-tuple as follows

$$\begin{array}{llllll} \textit{plan} = (& \textit{action seq.}, & \textit{open goals}, & \textit{initial state}, & \textit{original goals} &), \\ \pi^{\xi} = (& \pi^{\xi}, & \varphi_{\mathsf{goals}(\pi^{\xi})}, & \varphi_{T^{\xi}}, & \varphi_{G^{\xi}} &) \end{array}$$

The tree structure of these deterministic plans is given by the successor relation $\prec$. A plan π^{ξ} and any of its successors $\pi^{\xi'}$ are related by some formula they share: $\pi^{\xi} \prec \pi^{\xi'}$ means that the initial state $\varphi_{T^{\xi'}}$ in $\mathbb{M}^{\xi'}$ is part of the goals $\varphi_{G^{\xi}}$ for $\mathbb{M}^{\xi}$. Besides the tree structure for these components π^{ξ}, these and their planning domains $\mathbb{M}^{\xi}$ will be totally ordered by some lexicographic ordering $<_{\mathrm{lex}}$ on the labels ξ. This ordering will capture the order in which planning domains π^{ξ} are to be solved.

See Figure 3 for an illustration of the lexicographic ordering of labels ξ, namely $0 <_{\mathsf{lex}} 1 <_{\mathsf{lex}} 2 <_{\mathsf{lex}} 21 <_{\mathsf{lex}} 22 <_{\mathsf{lex}} 221 <_{\mathsf{lex}} 222$.

Another modification w.r.t. LCC deterministic planning concerns the Terminating Condition for plans introduced after refining with some non-deterministic action $\mathsf{e} \cup \mathsf{f}$. For a plan of the form $\pi^{\xi 1}$ addressing the case where $\mathsf{e} \cup \mathsf{f}$ instantiates as f, we replace the condition

$$\models \varphi_{T^{\xi 1}} \to \varphi_{\mathsf{goals}(\pi^{\xi 1})} \quad \text{by} \quad \models [\mathsf{U}, \mathsf{f}]\varphi_{\mathsf{goals}(\pi^{\xi 1})}$$

The reason is that we need to capture the effects of f, for which a single formula might not suffice. (Consider, for example, all the consequences of common knowledge after a public announcement). We will denote this kind of initial states as "$[\mathsf{U}, \mathsf{f}_1](\cdot)$". If this kind of Terminating Condition is satisfied by $\pi^{\xi 1}$, the resulting plan solution for $\mathbb{M}^{\xi 1}$ will be defined as $\pi^{\xi 1}(\mathsf{f}_1)$.

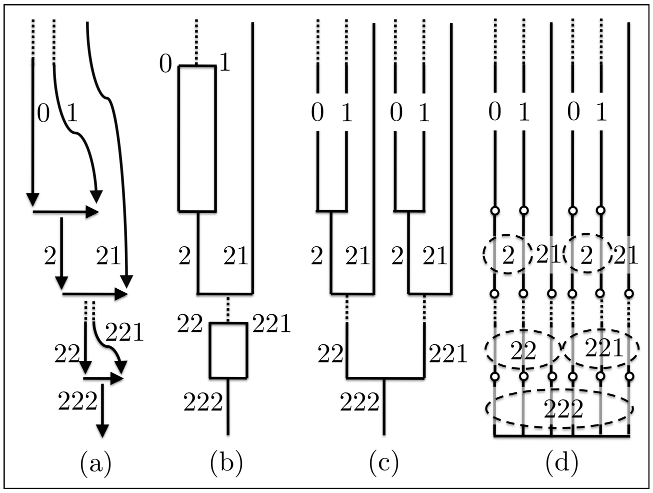

Fig. 3. (a) An output of non-deterministic planning BFS$_{nd}$; (b)-(d) three equivalent representations of (a) as modalities U, e: in (b) compact, (c) search and (d) branching form. Dotted, parallel lines (e.g. within π^{22}, π^{221}) represent two instances of the same actions (independently found by each corresponding BFS execution).

Definition 10. *Given some action model* U*, let* $\mathbb{M} = (\varphi_T, A, \varphi_G)$*. We define a* plan set

$$\Pi = (\{\pi^{\xi}, \pi^{\xi'}, \ldots\}, \prec)$$

where $\prec$ *is a successor relation inducing a (strict) partial order on a set of deterministic plans. The* empty plan set *is defined as* $\Pi_{\emptyset} = (\{\pi^0\}, \emptyset)$*, where* $\pi^0 = \pi_{\emptyset} = (\emptyset, \varphi_G, \varphi_T, \varphi_G)$ *is the empty plan for* $\mathbb{M}$.

If $\Pi = (\{\ldots, \pi^{\xi}, \ldots\}, \prec)$ *is a plan set, let* $\pi^{\xi} = (\pi^{\xi}, \varphi_{\mathsf{goals}(\pi^{\xi})}, \varphi_{T^{\xi}}, \varphi_{G^{\xi}})$ *be the plan in* Π *not solving* $\mathbb{M}^{\xi}$*, and whose label* ξ *is* $<_{\text{lex}}$*-minimal among the plans with this property. Let* $\mathsf{e} \in A \cap \mathsf{E}$ *and* $\mathsf{f} = \mathsf{f}_0 \cup \mathsf{f}_1 \in A$*. The refinements of the plan set* Π *are:*

$$\begin{array}{rl} \Pi(\mathsf{e}) = (& \{\ldots, \quad \pi^{\xi}(\mathsf{e}) \quad, \ldots\}, \prec \quad) \\ \Pi(\mathsf{f}) = (& \{\ldots, \ \pi^{\xi}, \pi^{\xi 1}, \pi^{\xi 2} \ , \ldots\}, \prec^{+}). \end{array}$$

under the convention that the refinement $\pi^{\xi}(\mathsf{e})$ *will be again denoted as* π^{ξ}*. While* $\pi^{\xi}(\mathsf{e})$ *is as in Def. 3 (but now a 4-tuple), the triple of new plans* $\pi^{\xi}, \pi^{\xi 1}, \pi^{\xi 2}$ *and the new partial order* $\prec^{+}$ *are defined as follows.*

plan	*actions*	*open goals*	*initial state*	*original goals*
π^{ξ}	$\pi^{\xi}(\mathsf{f}_0)$	$t([\mathsf{U}, \mathsf{f}_0]\varphi_{\mathsf{goals}(\pi^{\xi})} \wedge \langle \mathsf{U}, \mathsf{f}_0 \rangle \top)$	$[\mathsf{U}, \mathsf{f}_0]\varphi_{\mathsf{goals}(\pi^{\xi})} \wedge \langle \mathsf{U}, \mathsf{f}_0 \rangle \top$	$\varphi_{G^{\xi}}$
$\pi^{\xi 1}$	$\emptyset$	$\varphi_{G^{\xi}}$	"$[\mathsf{U}, \mathsf{f}_1](\cdot)$"	$\varphi_{G^{\xi}}$
$\pi^{\xi 2}$	$\emptyset$	$\varphi_{\mathsf{goals}(\pi^{\xi}(\mathsf{f}_0))} \wedge \varphi_{\mathsf{goals}(\pi^{\xi 1}(\mathsf{f}_1))}$	$\varphi_{T^{\xi}}$	*as open goals*

$$\prec^{+} = \prec \cup \{\langle \pi^{\xi 2}, \pi^{\xi} \rangle, \langle \pi^{\xi 2}, \pi^{\xi 1} \rangle\}.$$

Example 3. Let the output Π for some $(\varphi_T, A, \varphi_G)$ and A be as in Figure 3. The leftmost figure represents the output of the BFS$_{nd}$ algorithm, built incrementally according to the lexicographic ordering: $\Pi = \langle \pi^{\emptyset}, \pi^1, \pi^2, \pi^{21}, \pi^{22}, \pi^{221}, \pi^{222} \rangle$. Arrows also indicate the direction of backward search, with horizontal arrows representing the introduction of a non-deterministic action.

See Figure 4 for a description of the BFS$_{nd}$ algorithm for non-deterministic planning. If a plan $\varPi$ is the output of BFS$_{nd}$, then we also call its $\prec$-minimal element, say $\pi^{\xi 2}$, as π^{root}.

```
1. LET $\mathbf{Plans} = \langle \varPi_\emptyset \rangle$.
2. SELECT the first element $\varPi$ of $\mathbf{Plans}$. DELETE $\varPi$ from $\mathbf{Plans}$.
3. SELECT the $<_{\mathrm{lex}}$-minimal element $\pi^\xi$ of $\varPi$ not solving $\mathbb{M}^\xi$.
     If $\pi^\xi$ is a leaf REPEAT step 2.
     If Terminating Condition holds for $\varphi_{\mathsf{goals}(\pi^\xi)}$,
        and $\xi$ is $<_{\mathrm{lex}}$-maximal in $\varPi$, then OUTPUT $\varPi$.
     Otherwise SET $\mathbf{Plans} \leftarrow \mathbf{Plans}^\frown \langle \varPi(\mathsf{e}) \mid \mathsf{e} \in A \rangle$. REPEAT step 2.
```

Fig. 4. BFS$_{nd}$ algorithm for non-deterministic planning with input $(\varphi_T, A, \varphi_G)$.

First we show how to translate a BFS$_{nd}$ output (plan set) for $\mathbb{M}$ into a modality $\mathsf{U}, \mathsf{e}^\star$ with $\mathsf{e}^\star \in A^{\otimes \cup}$, and such that $\mathsf{U}, \mathsf{e}^\star$ is in search form. Then we show soundness of BFS$_{nd}$, i.e. that $\mathsf{U}, \mathsf{e}^\star$ to be a solution for $\mathbb{M}$.

Definition 11. *Given an output $\varPi$ for a planning domain $(\varphi_T, A, \varphi_G)$, we define the modality $\mathsf{e}^\star$ induced from $\varPi$ as follows. First, for each $\pi^\xi = (\mathsf{f}_m, \dots, \mathsf{f}_1) \in \varPi$, define $\mathsf{f}^\xi = \mathsf{f}_1 \otimes \dots \otimes \mathsf{f}_m$ and $\mathsf{f}^\xi = \mathsf{skip}$ if $\pi^\xi = \pi_\emptyset$. Then, define*

$$\mathsf{e}^\xi = \begin{cases} \mathsf{f}^\xi & \textit{if } \xi \textit{ is } \prec\textit{-maximal} \\ \mathsf{f}^\xi \otimes (\mathsf{e}^{\xi'} \cup \mathsf{e}^{\xi''}) & \textit{if } \xi \prec \xi', \xi'' \end{cases}$$

Finally, we define $\mathsf{e}^\star = \mathsf{e}^{\mathrm{root}}$.

Theorem 2. *For any output for $\varPi$ for a given $\mathbb{M} = (\varphi_T, A, \varphi_G)$, the induced $\mathsf{U}, \mathsf{e}^\star$ modality defines a solution for $\mathbb{M}$:*

$$(1) \models \varphi_T \to [\mathsf{U}, \mathsf{e}^\star]\varphi_G \quad \textit{and} \quad (2) \models \bigwedge_{\overrightarrow{\mathsf{e}} \text{ a path in } \mathsf{e}^\star} \varphi_T \to \langle \mathsf{U}, \overrightarrow{\mathsf{e}} \rangle \top$$

Now we proceed to show that BFS$_{nd}$ is also complete for non-deterministic planning in LCC$_{\otimes \cup}$.

Lemma 2. *For a given action model U and a set of available actions A,*

each modality U, e in $A^{\otimes \cup}$ is logically equivalent to a modality $\mathsf{U}, \mathsf{e}^\star$ in $A^{\otimes \cup}$ search form.

Using this Lemma, we may simply assume next that the solution, to be found by BFS$_{nd}$, is in search form.

Theorem 3. *Given an* LCC *logic of some action model* U*, a set* A*, and a planning domain* $\mathbb{M}$ *expressible in the extended logic* $\mathrm{LCC}_{\otimes\cup}$*. Let* U,e *be a modality, with* $\mathsf{e}\in A$ *in search form. If* U,e *is a solution for* $\mathbb{M}$*, then the algorithm* BFS_{nd} *for* $\mathbb{M}$ *terminates with an output* Π*.*

Let us note that from Theorems 2 and 3 we can conclude that backward planning for $\mathrm{LCC}_{\otimes\cup}$ logics is at least semi-decidable. For the case of (incremental) forward planning, this result is shown in [3]. To illustrate non-determinsitic planning, let us finally solve Example 1.

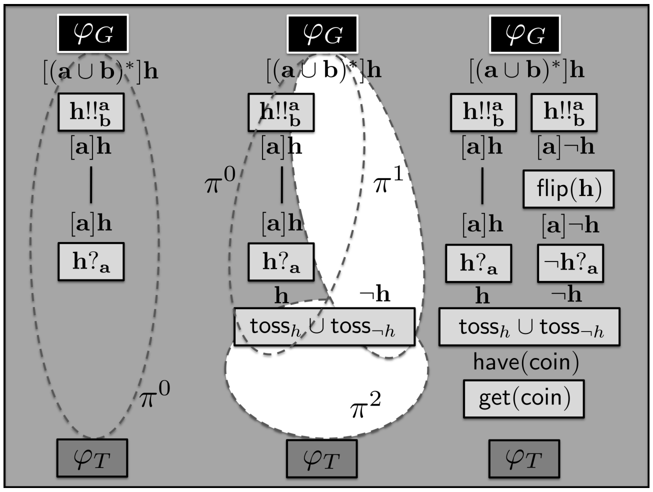
φ_G
$[(\mathbf{a}\cup\mathbf{b})^*]\mathbf{h}$
$\mathbf{h}!!^{\mathbf{a}}_{\mathbf{b}}$
$[\mathbf{a}]\mathbf{h}$
$[\mathbf{a}]\mathbf{h}$
$\mathbf{h}?_{\mathbf{a}}$
π^0
φ_T
φ_G
$[(\mathbf{a}\cup\mathbf{b})^*]\mathbf{h}$
$\mathbf{h}!!^{\mathbf{a}}_{\mathbf{b}}$
$[\mathbf{a}]\mathbf{h}$
π^0
π^1
$[\mathbf{a}]\mathbf{h}$
$\mathbf{h}?_{\mathbf{a}}$
$\mathbf{h}$
$\neg\mathbf{h}$
$\mathsf{toss}_h\cup\mathsf{toss}_{\neg h}$
π^2
φ_T
φ_G
$[(\mathbf{a}\cup\mathbf{b})^*]\mathbf{h}$
$\mathbf{h}!!^{\mathbf{a}}_{\mathbf{b}}$
$\mathbf{h}!!^{\mathbf{a}}_{\mathbf{b}}$
$[\mathbf{a}]\mathbf{h}$
$[\mathbf{a}]\neg\mathbf{h}$
$\mathsf{flip}(\mathbf{h})$
$[\mathbf{a}]\mathbf{h}$
$[\mathbf{a}]\neg\mathbf{h}$
$\mathbf{h}?_{\mathbf{a}}$
$\neg\mathbf{h}?_{\mathbf{a}}$
$\mathbf{h}$
$\neg\mathbf{h}$
$\mathsf{toss}_h\cup\mathsf{toss}_{\neg h}$
$\mathsf{have(coin)}$
$\mathsf{get(coin)}$
φ_T

Fig. 5. Plan search in Example 1. A single plan π^0 in the plan set (Left), is refined by a non-deterministic action (Center), splitting π^0 into π^0,π^1,π^2. These plans are latter solved, as shown in (Right).

Example 4. Recall Example 1, where the planner agent $\mathbf{a}$ must show heads, denoted $\mathbf{h}$, to win the prize. Among the actions A of agent $\mathbf{a}$, we find: getting the coin; tossing the coin; flipping the coin to heads (with precondition that $\mathbf{a}$ knows that $\neg\mathbf{h}$); a set of announcements, including *justified truthful announcements* $\varphi!!^{\mathbf{a}}_{\mathbf{b}}$ (the latter require that the announcing agent knows the truth of φ); and secret observations, for heads $\mathbf{h}?_{\mathbf{a}}$ and tails $\neg\mathbf{h}?_{\mathbf{a}}$. (Their secrecy consists in $\mathbf{b}$ mistaking either action by $\mathsf{skip}_{\mathbf{a}}$.) In 5(Left), the BFS_{nd} algorithm searches first for a single plan π^0. This particular π^0 demands $\varphi_{\mathsf{goals}(\pi^0)}=\mathbf{h}$, and the only action possibly enforcing $\mathbf{h}$ is to toss the coin, which also can lead to $\neg\mathbf{h}$. Refining with this action is shown in (Center). Finally, (Right) shows the output of BFS_{nd} that solves this problem.

Before concluding, let us remark that the previous A* search for optimal planning easily extends to the non-deterministic case. Among the several possible options to define the total cost of a non-deterministic plan, consider e.g. the pessimistic and optimistic estimations based on

$$\mathsf{cost}(\mathsf{e} \cup \mathsf{e}') = \max\{\mathsf{cost}(\mathsf{e}), \mathsf{cost}(\mathsf{e}')\} \text{ and } \mathsf{cost}(\mathsf{e} \cup \mathsf{e}') = \min\{\mathsf{cost}(\mathsf{e}), \mathsf{cost}(\mathsf{e}')\}$$

Thus, the cost of a plan is the $\max$ or $\min$ cost of its paths. For each of these cost aggregation functions $\max, \min$, one can redefine the BFS$_{nd}$ algorithm so that instead of stopping when the first output Π is found (with cost k), it continues search until all plans $\Pi' \in \mathbf{Plans}$ have at least some path whose cost is $\geq k$. Hence, optimal planning can be extended to strong non-deterministic planning in $\mathrm{LCC}_{\otimes\cup}$.

Conclusions and Future Work

We presented backward planning algorithms for a planner-reasoner agent enabling this agent to find deterministic or (non-deterministic) strong plans in multi-agent scenarios. The logics considered here are dynamic epistemic logics with ontic actions, thus allowing the plans to contain communicative actions, sensing or the usual fact-changing actions. We hope the proposed work might serve for practical reasoning in communicative agents, and in particular provide a logical foundation for the modeling (and computing) of agreements among motivated agents. As for future work, several directions seem interesting: like the introduction of dynamic operators for belief revision, and at a more applied level, the study of heuristic criteria for optimal plans.

Acknowledgements: This work has been funded by projects AT (CSD 2007-022); LoMoReVI (FFI2008-03126-E/FILO FP006); ARINF (TIN2009-14704-C03-03); and the GenCat 2009-SGR-1434 and EPSRC grant EP/J002607/1.

References

1. G. Aucher. DEL-sequents for progression, *Journal of Applied Non-Classical Logics* 21(3-4): 289–321 (2011)
2. A. Baltag, L. Moss and S. Solecki. The logic of public announcements, common knowledge and private suspicions, *Proc. of 7th Theoretical Aspects of Representation Knowledge TARK'98* pp. 43–56 (1998)
3. T. Bolander and M. Birkegaard Andersen. Epistemic planning for single- and multi-agent systems *Journal of Applied Non-Classical Logics*, in press (2011)
4. T. French and H. van Ditmarsch. Undecidability for arbitrary public announcement logic, *Advances in Modal Logic AiML 2008*, F. Wolter, H. Wansing, M. de Rijke and Zakharyaschev (eds.), Vol. 7, pp. 23–42, CSLI Stanford (2008)
5. M. Ghallab, D. Nau and P. Traverso. *Automated Planning: Theory and Practice*, Morgan Kaufmann, San Francisco, USA (2004)
6. B. Kooi. Expressivity and completeness for public update logics via reduction axioms, *Journal of Applied Non-Classical Logics*, 17(2): 231–253 (2007)
7. B. Löwe, E. Pacuit and A. Witzel. Planning based on dynamic epistemic logic. *Tech. Report* (2010)
8. P. Pardo and M. Sadrzadeh. Planning in the Logics of Communication and Change, *Proc. of Autonomous Agents and Multiagent Systems AAMAS 2012*, V. Conitzer, M. Winikoff, L. Padgham and W. van der Hoek (eds.) Valencia, Spain (2012)

9. J. Pearl *Heuristics: Intelligent Search Strategies for Computer Problem Solving* Addison-Wesley, 1984.
10. J. van Benthem, J. van Eijck and B. Kooi. Logics of Communication and Change, *Information and Computation* 204:1620–1662 (2006)
11. W. van der Hoek and M. Wooldridge. Tractable Multiagent Planning for Epistemic Goals *Proc. of Autonomous Agents and Multiagent Systems AAMAS 2002*, pp. 1167–1174 (2002)
12. H. van Ditmarsch, W. van der Hoek and B. Kooi. *Dynamic Epistemic Logic*, Springer (2008)
13. H. van Ditmarsch, J. van Eijck, F Sietsma and Y. Wang. On the logic of Lying, *Games, Actions and Social Software* J. van Eijck and R. Verbrugge (eds.) LNAI-FoLLI, Springer (in press)
14. H. van Ditmarsch, B. Kooi. Semantic results for ontic and epistemic change *Proceedings of Logic and the Foundations of Game and Decision Theory LOFT 7* G. Bonanno, W. van der Hoek and M. Wooldridge (eds.) pp. 87–117 (2008)