Cooperative Epistemic Multi-Agent Planning With Implicit Coordination

Thorsten Engesser
Institut für Informatik
Albert-Ludwigs-Universität
Freiburg, Germany
engesset@cs.uni-freiburg.de

Thomas Bolander
DTU Compute
Technical University of Denmark
Copenhagen, Denmark
tobo@dtu.dk

Robert Mattmüller
Institut für Informatik
Albert-Ludwigs-Universität
Freiburg, Germany
mattmuel@cs.uni-freiburg.de

Bernhard Nebel
Institut für Informatik
Albert-Ludwigs-Universität
Freiburg, Germany
nebel@cs.uni-freiburg.de

Abstract

Epistemic Planning has been used to achieve ontic and epistemic control in multi-agent situations. We extend the formalism to include perspective shifts, allowing us to define a class of cooperative problems in which both action planning and execution is done in a purely distributed fashion, meaning coordination is only allowed implicitly by means of the available epistemic actions. While this approach can be fruitfully applied to model reasoning in some simple social situations, we also provide some benchmark applications to show that the concept is useful for multi-agent systems in practice.

1 Introduction

One important task in Multi-Agent Systems is to collaboratively reach a joint goal with multiple autonomous agents. The problem is particularly challenging in situations where the knowledge required to reach the goal is distributed among the agents. Most existing approaches therefore apply some centralized coordinating instance from the outside, strictly separating the stages of communication and negotiation from the agents' internal planning and reasoning processes. In contrast, building upon the epistemic planning framework by Bolander and Andersen (2011), we propose a decentralized planning notion in which each agent has to individually reason about the entire problem and autonomously decide when and how to (inter-)act. For this, both reasoning about the other agents' possible contributions and reasoning about their capabilities of performing the same reasoning is needed. We achieve our notion of implicitly coordinated plans by requiring all desired communicative abilities to be modeled as epistemic actions which then can be planned alongside their ontic counterparts, thus enabling the agents to perform observations and coordinate at runtime. While this imposes certain restrictions on the problems that can be solved, it captures the intuition that communication clearly constitutes an action by itself and, more subtly, that even a purely ontic action can play a communicative role (e.g. indirectly suggesting follow-up actions to another agent). Thus, for many problems our approach appears quite natural. On the practical side, the epistemic planning framework allows a very expressive way of defining both the agents' physical and communicative abilities and thereby seems an ideal choice in our case.

Our work directly builds upon the framework introduced by Bolander and Andersen (2011) and Löwe, Pacuit, and Witzel (2011), who formulated the planing problem in the context of Dynamic Epistemic Logic (DEL) (van Ditmarsch, van der Hoek, and Kooi 2007). Andersen, Bolander, and Jensen (2012) extended the approach to allow strong and weak conditional planning in the single-agent case. Similar to Bolander and Andersen (2011), we use search in the space of epistemic states to find a solution. This is in contrast to compilation approaches inspired by Palacios and Geffner (2009), which are popular and successful in the AI planning community. These approaches map an epistemic (or doxastic) planning problem to a corresponding classical one allowing to solve the problem using a classical planner (Muise et al. 2015; Kominis and Geffner 2015). However, these approaches can only deal with bounded nesting of knowledge (or belief) and can produce only sequential plans. There is an important similarity between the work by Muise et al. (2015) and ours, though: In both approaches, it is possible to shift the perspective from agent to agent along the plan. In particular this possibility of perspective shifts distinguishes these approaches from more traditional multi-agent planning (Brenner and Nebel 2009). Recent work in this area by Nissim and Brafman (2014) proposes a search algorithm for multi-agent planning that allows private actions and a certain degree of decentralization that achieves efficiency at the cost of not supporting reasoning about knowledge of other agents or only implicitly.

2 Theoretical Background

2.1 Epistemic language and epistemic states

We first recapitulate the foundations of DEL, following the conventions of Bolander and Andersen (2011). This means that our version of DEL includes postconditions allowing for ontic/factual change, but postconditions are without loss of expressivity limited to conjunctions of literals (as in classical planning).

Definition 1. The *epistemic language* $\mathcal{L}_{\text{KC}}(P, \mathcal{A})$ with respect to a set of atomic propositions P and a finite set of agents $\mathcal{A}$ is

$$\varphi \ ::= \ p \mid \neg\varphi \mid \varphi \wedge \varphi \mid K_i\varphi \mid C\varphi$$

where $p \in P$ and $i \in \mathcal{A}$.

We read $K_i\varphi$ as "agent i knows φ" and $C\varphi$ as "it is common knowledge (among all agents) that φ". In the following, we will always use P to denote our set of atomic propositions, and $\mathcal{A}$ our set of agents. Formulas of the epistemic language are evaluated in epistemic models.

Definition 2. An *epistemic model* over $(P, \mathcal{A})$ is a triple $\mathcal{M} = \langle W, (R_i)_{i\in\mathcal{A}}, V\rangle$ where

- The *domain* W is a non-empty finite set of *worlds*.
- $R_i \subseteq W \times W$ is an equivalence relation called the *indistinguishability relation* for agent i.
- $V : P \to \mathcal{P}(W)$ assigns a *valuation* to each atomic proposition.

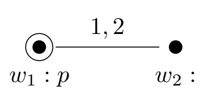
For $W_d \subseteq W$, the pair $(\mathcal{M}, W_d)$ is called an *epistemic state* (or simply *state*), and the worlds of W_d are called the *designated worlds*. An epistemic state is called *global* if W_d is a singleton. The designated world of a global state is called the *actual world*. In general, $(\mathcal{M}, W_d)$ can be thought of as the belief state $\{(\mathcal{M}, \{w\}) \mid w \in W_d\}$ over possible global states. An epistemic state $(\mathcal{M}, W_d)$ is called a *local state* for agent i if W_d is closed under R_i. A local state is *minimal* if W_d is a minimal set closed under R_i. Given an epistemic state $(\mathcal{M}, W_d)$, the *associated local state* of agent i, denoted $(\mathcal{M}, W_d)^i$, is $(\mathcal{M}, \{v \mid wR_iv, w \in W_d\})$.

Definition 3. Let $(\mathcal{M}, W_d)$ be an epistemic state where $\mathcal{M} = \langle W, (R_i)_{i\in\mathcal{A}}, V\rangle$. For $i \in \mathcal{A}$, $p \in P$ and $\varphi, \psi \in \mathcal{L}_{\mathrm{KC}}(P, \mathcal{A})$, we define truth as follows:

$$
\begin{array}{ll}
(\mathcal{M}, W_d) \models \varphi & \text{iff } (\mathcal{M}, w) \models \varphi \text{ f.a. } w \in W_d \\
(\mathcal{M}, w) \models p & \text{iff } w \in V(p) \\
(\mathcal{M}, w) \models \neg\varphi & \text{iff } \mathcal{M}, w \not\models \varphi \\
(\mathcal{M}, w) \models \varphi \wedge \psi & \text{iff } \mathcal{M}, w \models \varphi \text{ and } \mathcal{M}, w \models \psi \\
(\mathcal{M}, w) \models K_i\varphi & \text{iff } \mathcal{M}, w' \models \varphi \text{ f.a. } w' \in W \text{ s.t. } wR_iw' \\
(\mathcal{M}, w) \models C\varphi & \text{iff } \mathcal{M}, w' \models \varphi \text{ f.a. } w' \in W \text{ s.t. } wR^*w'
\end{array}
$$

where R^* is the transitive closure of $\bigcup_{i\in\mathcal{A}} R_i$.

Example 1. A global epistemic state describes an epistemic situation from a global perspective, where the actual world has been pointed out. Consider the following global state $(\mathcal{M}, \{w_1\})$, where the nodes represent worlds and the edges represent the indistinguishability relations (reflexive edges left out):



We use ⦿ to denote the designated worlds, in this case only w_1. The actual world is w_1 where p holds, but for agent 1 (and 2) the actual world w_1 is indistinguishable from the world w_2 where p is false. Since agent 1 is ignorant about whether the actual world is w_1 or w_2, the model that represents his local view on the situation is $(\mathcal{M}, \{w_1, w_2\})$, which is exactly his associated local state of $(\mathcal{M}, \{w_1\})$. The point is that since agent 1 is unable to point out whether the actual world is w_1 or w_2, his internal state must consider both as candidates for being the actual world, and this is exactly what the model $(\mathcal{M}, \{w_1, w_2\}) = (\mathcal{M}, \{w_1\})^1$ does. We have $(\mathcal{M}, \{w_1\})^1 \not\models p$ and $(\mathcal{M}, \{w_1\})^1 \not\models \neg p$, corresponding to the fact that from agent 1's local perspective it can not be verified whether p holds or not.

2.2 Perspective shifts

In general, given an epistemic state $(\mathcal{M}, W_d)$, the associated local state $(\mathcal{M}, W_d)^i$ will represent agent i's internal perspective on that state. Going from $(\mathcal{M}, W_d)$ to $(\mathcal{M}, W_d)^i$ amounts to a *perspective shift*, where the perspective is shifted to the local perspective of agent i. Note that we have the following properties, where the third follows directly from the two first:

Lemma 1. *Let $(\mathcal{M}, W_d)$ be an epistemic state over $(P, \mathcal{A})$, $i \in \mathcal{A}$ and $\varphi \in \mathcal{L}_{\mathrm{KC}}(P, \mathcal{A})$.*

1. *$(\mathcal{M}, W_d)^i \models \varphi$ iff $(\mathcal{M}, W_d) \models K_i\varphi$.*
2. *If $(\mathcal{M}, W_d)$ is local for agent i then $(\mathcal{M}, W_d)^i = (\mathcal{M}, W_d)$.*
3. *If $(\mathcal{M}, W_d)$ is local for agent i then $(\mathcal{M}, W_d) \models \varphi$ iff $(\mathcal{M}, W_d) \models K_i\varphi$.* □

Perspective shifts are of fundamental importance in multi-agent planning to allow an agent to reason about the other agents' possible contributions to a plan.

2.3 Dynamic language and epistemic actions

To model actions, we use the *event models* of DEL.

Definition 4. An *event model* over $(P, \mathcal{A})$ is a tuple $\mathcal{E} = \langle E, (Q_i)_{i\in\mathcal{A}}, \mathrm{pre}, \mathrm{post}\rangle$ where

- The *domain* E is a non-empty finite set of *events*.
- $Q_i \subseteq E \times E$ is an equivalence relation called the *indistinguishability relation* for agent i.
- $\mathrm{pre} : E \to \mathcal{L}_{\mathrm{KC}}(P, \mathcal{A})$ assigns a *precondition* to each event.
- $\mathrm{post} : E \to \mathcal{L}_{\mathrm{KC}}(P, \mathcal{A})$ assigns a *postcondition* to each event. For all $e \in E$, *post*(e) is a conjunction of literals (atomic propositions and their negations).

For $E_d \subseteq E$, the pair $(\mathcal{E}, E_d)$ is called an *epistemic action* (or simply *action*), and the events in E_d are called the *designated events*. Similar to epistemic states, $(\mathcal{E}, E_d)$ is called a *local action* for agent i when E_d is closed under Q_i.

Each event of an epistemic action represents a different possible outcome. By using multiple events $e, e' \in E$ that are indistinguishable (i.e. eQ_ie'), it is possible to obfuscate the outcomes for some agent $i \in \mathcal{A}$, i.e. modeling partially observable actions. Using event models with $|E_d| > 1$, it is also possible to model sensing actions and nondeterministic actions (Bolander and Andersen 2011).

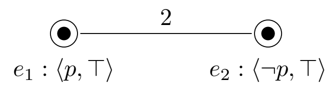
The *product update* is used to specify the successor state resulting from the application of an action in a state.

Definition 5. Let a state $(\mathcal{M}, W_d)$ and an action $(\mathcal{E}, E_d)$ over $(P, \mathcal{A})$ be given with $\mathcal{M} = \langle W, (R_i)_{i\in\mathcal{A}}, V\rangle$ and $\mathcal{E} = \langle E, (Q_i)_{i\in\mathcal{A}}, \mathrm{pre}, \mathrm{post}\rangle$. Then the *product update* is defined as $(\mathcal{M}, W_d) \otimes (\mathcal{E}, E_d) = (\langle W', (R'_i)_{i\in\mathcal{A}}, V'\rangle, W'_d)$ where

- $W' = \{(w, e) \in W \times E \mid \mathcal{M}, w \models \mathrm{pre}(e)\}$
- $R'_i = \{((w, e), (w', e')) \in W' \times W' \mid wR_iw' \text{ and } eQ_ie'\}$
- $V'(p) = \{(w, e) \in W' \mid \mathrm{post}(e) \models p$ or $(\mathcal{M}, w \models p \text{ and } \mathrm{post}(e) \not\models \neg p)\}$
- $W'_d = \{(w, e) \in W' \mid w \in W_d, e \in E_d\}$.

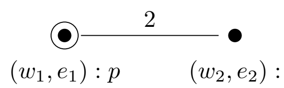
If both $(\mathcal{M}, W_d)$ and $(\mathcal{E}, E_d)$ are local for agent i, then so is $(\mathcal{M}, W_d) \otimes (\mathcal{E}, E_d)$ (Bolander and Andersen 2011).

Example 2. Consider the following epistemic action $(\mathcal{E}, \{e_1, e_2\})$, using the same conventions as for epistemic models, except each event e is labelled by $\langle \mathrm{pre}(e), \mathrm{post}(e)\rangle$:



It is a private sensing action for agent 1, where (only) agent 1 gets to know the truth value of p, since e_1 and e_2 are distinguishable to agent 1. Letting $(\mathcal{M}, \{w_1\})$ denote the state from Example 1, we get:

$(\mathcal{M}, \{w_1\}) \otimes (\mathcal{E}, \{e_1, e_2\}) =$



Hence $(\mathcal{M}, \{w_1\}) \otimes (\mathcal{E}, \{e_1, e_2\})$ is exactly as $(\mathcal{M}, \{w_1\})$ except the indistinguishability edge for agent 1 is removed. So the private sensing action reveals to agent 1 that p is true (without revealing it to agent 2). Before executing the action, agent 1 however does not know whether he will learn p or $\neg p$, which is signified by $(\mathcal{M}, \{w_1\})^1 \otimes (\mathcal{E}, \{e_1, e_2\})$ having both a designated p world and a designated $\neg p$ world (it differs from the model $(\mathcal{M}, \{w_1\}) \otimes (\mathcal{E}, \{e_1, e_2\})$ shown above exactly by (w_2, e_2) also being designated).

We extend the language $\mathcal{L}_{\mathrm{KC}}(P, \mathcal{A})$ into the *dynamic language* $\mathcal{L}_{\mathrm{DEL}}(P, \mathcal{A})$ by adding a modality $[(\mathcal{E}, e)]$ for each event model $\mathcal{E} = (E, (Q_i)_{i\in\mathcal{A}}, \mathrm{pre}, \mathrm{post})$ over $(P, \mathcal{A})$ and $e \in E$. The truth conditions are extended with the following standard clause from DEL:

$$(\mathcal{M}, w) \models [(\mathcal{E}, e)]\varphi \text{ iff}$$
$$(\mathcal{M}, w) \models \mathrm{pre}(e) \text{ implies } (\mathcal{M}, w) \otimes (\mathcal{E}, e) \models \varphi.$$

We define the following abbreviations: $[(\mathcal{E}, E_d)]\varphi := \bigwedge_{e\in E_d}[(\mathcal{E}, e)]\varphi$ and $\langle(\mathcal{E}, E_d)\rangle\varphi := \neg[(\mathcal{E}, E_d)]\neg\varphi$. We say that an action $(\mathcal{E}, E_d)$ is *applicable* in a state $(\mathcal{M}, W_d)$ if for all $w \in W_d$ there is an event $e \in E_d$ s.t. $(\mathcal{M}, w) \models \mathrm{pre}(e)$. Intuitively, an action is applicable in a state if for each possible situation (designated world), at least one possible outcome (designated event) is specified. Let $s = (\mathcal{M}, W_d)$ denote an epistemic state and $a = (\mathcal{E}, E_d)$ an action. Andersen (2015) shows that a is applicable in s iff $s \models \langle a\rangle\top$, and that $s \models [a]\varphi$ iff $s \otimes a \models \varphi$. We now define a further abbreviation: $(\!(a)\!)\varphi := \langle a\rangle\top \wedge [a]\varphi$. Hence:

$$s \models (\!(a)\!)\varphi \ \text{ iff } \ a \text{ is applicable in } s \text{ and } s \otimes a \models \varphi \qquad (1)$$

Thus $(\!(a)\!)\varphi$ means that the application of a is possible and will (necessarily) lead to a state fulfilling φ.

3 Cooperative Planning

We will now consider different types of planning problems and solution concepts in the setting of DEL-based planning. In the simplest case, a planning problem $\langle s_0, A, \varphi_g\rangle$ over $(P, \mathcal{A})$ consists of an initial epistemic state s_0 over $(P, \mathcal{A})$, a set A of epistemic actions over $(P, \mathcal{A})$ and a goal formula φ_g of $\mathcal{L}_{\mathrm{KC}}(P, \mathcal{A})$. Informally, a (sequential) solution to such a planning problem is a sequence of actions $(a_1, \ldots, a_n)$ from A, such that executing the sequence in s_0 leads to a state satisfying φ_g. In the DEL-based setting, the state-transition function mapping a state-action pair (s, a) into the state resulting from executing a in s is given by $(s, a) \mapsto s \otimes a$ (when a is not applicable in s, the state-transition function is taken to be undefined on (s, a)). Hence, more formally, a solution to $\langle s_0, A, \varphi_g\rangle$ is a sequence of actions $(a_1, \ldots, a_n)$ from A such that for all $i = 1, \ldots, n$, the action a_i is applicable in $s_0 \otimes a_1 \otimes \cdots \otimes a_{i-1}$, and $s_0 \otimes a_1 \otimes \cdots \otimes a_n \models \varphi_g$. Note that by (1) above, these conditions are equivalent to simply requiring $s_0 \models (\!(a_1)\!)(\!(a_2)\!)\cdots(\!(a_n)\!)\varphi_g$.

This solution concept is equivalent to the one considered in (Bolander and Andersen 2011). In that paper, the initial state as well as all actions are supposed to be modelled from the perspective of one single planning agent, that is, be local to that agent. Such a setting provides a natural formal framework for a single agent acting alone in a multi-agent environment, but does not provide a systematic solution to the case where multiple agents are (inter)acting towards a joint goal. The latter situation is what we wish to consider in this paper.

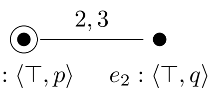
For cooperative multi-agent planning towards a joint goal, we identify the following settings:

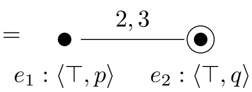
1. *Centralized planning*, meaning one instance (having complete or incomplete knowledge, e.g. as one of the agents) generates a plan in advance, which, if given to and executed by the cooperative agents, will lead to a goal state.
2. *Decentralized planning*. Here each agent does the planning process for himself. Usually this is done as part of a multi-agent architecture where the agents announce their plans, negotiate, solve conflicts, etc.
3. *Decentralized planning with implicit coordination*. In this scenario, all coordination is achieved implicitly through observing the effects of the actions of other agents. The rationale is, if all of the agents act to compatible individual plans (which may include assumptions on other agents' actions and plans), the goal condition can be reached without explicit coordination and commitments.

This paper focuses on the concept of decentralized planning with implicit coordination, which relies closely on the perspective-shifting capabilities of the epistemic planning framework and is, to the best of our knowledge, novel to this paper. Using our approach, it is possible to solve some multi-agent problems by just specifying a common goal directive for all the agents in a system, given some optimistic assumptions, namely common knowledge of the available actions, consistent internal models of the situation and perfect reasoning capacity of all agents. The agents can then take the necessary steps (deducing compatible plans and acting on them) in a generic, autonomous manner.

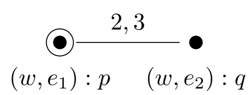
Let A denote a set of actions, and $\mathcal{A}$ a set of agents. In this paper we assume each action to be executable by a single agent, that is, we are not considering joint actions. For technical reasons, we wish each action to be executable by a *unique* agent, which we call the *owner* of the action. More precisely, an *owner function* is a mapping $\omega : A \to \mathcal{A}$, mapping each action to the agent who can execute it (who *owns* it). This approach is closely related to the one by Löwe, Pacuit, and Witzel (2011). Mapping each action to a unique agent can be done without loss of generality, since semantically equivalent duplicates can always be added to the action set.

Example 3. As mentioned above, our framework is based on the assumption that there is common knowledge of the available actions in the action set A (so that agents can always correctly reason about the actions available to themselves and others). This however does not imply that agents are always aware of the actions executed by others. Consider the following two actions owned by agent 1:

$a_p =$ 

$a_q =$ 

The action a_p makes p true and a_q makes q true. If agent 1 executes a_p, agents 2 and 3 will still consider it possible that q was made true (due to the indistinguishability edge in a_p), and conversely for a_q. Let s_0 be a singleton model where no atoms are true: $s_0 =$ ⊙ $w :$. Then a_p is applicable in s_0 and the execution will result in:

$s_0 \otimes a_p =$ 

We see that $s_0 \otimes a_p \models p$ from which we get $s_0 \models (\!(a_p)\!)\, p$, using (1). As s_0 is clearly local to all 3 agents, we can apply item 3 of Lemma 1 to conclude $s_0 \models K_i (\!(a_p)\!)\, p$ for $i = 1, 2, 3$. Note also that $s_0 \otimes a_p \models \neg K_j p$ for $j = 2, 3$, and hence $s_0 \models (\!(a_p)\!)\, \neg K_j p$ using again (1). Hence we get:

$$s_0 \models K_i(\!(a_p)\!)p \qquad \text{for } i = 1, 2, 3 \tag{2}$$

$$s_0 \models (\!(a_p)\!)\neg K_j p \qquad \text{for } j = 2, 3 \tag{3}$$

The intuition is this: All agents know that executing a_p leads to p, since they know the action set. This is captured by (2). However, even after a_p has been executed, agents 2 and 3 do not know p, since for them action a_p is indistinguishable from action a_q (they will not know whether a_p or a_q was chosen by agent 1). This is captured by (3).

The only situations where it makes sense to have both (2) and (3) above are when agents 2 and 3 can mistake the action a_p for another action. To only consider such sensible scenarios we will require our action sets A to satisfy the following closure property: If $(\mathcal{E}, E_d) \in A$ and e is an event in $\mathcal{E}$, then there exists a set E'_d with $e \in E'_d$ and $(\mathcal{E}, E'_d) \in A$. If A satisfies this property, it is called *closed*.

We are now finally ready to formally define our notion of a cooperative epistemic planning problem.

Definition 6. A *cooperative planning problem* (or simply a *planning problem*) $\Pi = \langle s_0, A, \omega, \varphi_g \rangle$ over $(P, \mathcal{A})$ consists of

- an initial epistemic state s_0 over $(P, \mathcal{A})$
- a finite, closed set of epistemic actions A over $(P, \mathcal{A})$
- an owner function $\omega : A \to \mathcal{A}$
- a goal formula $\varphi_g \in \mathcal{L}_{\mathrm{KC}}(P, \mathcal{A})$

such that each $a \in A$ is local for $\omega(a)$. When s_0 is a global state, we call it a *global cooperative planning problem* (or simply a *global planning problem*). When s_0 is local for agent i, we call it a *cooperative planning problem for agent* i (or simply a *planning problem for agent* i). Given a global planning problem $\Pi = \langle s_0, A, \omega, \varphi_g \rangle$, the *associated planning problem for agent* i is $\Pi^i = \langle {s_0}^i, A, \omega, \varphi_g \rangle$.

Given a multi-agent system is facing a global planning problem Π, then each individual agent i is facing the planning problem Π^i (agent i cannot observe the global state s_0 directly, only the associated local state ${s_0}^i$).

In the following, we define sequential and conditional implicitly coordinated solution concepts for cooperative planning problems.

3.1 Sequential Plans

We first want to define our notion of an implicitly coordinated sequential solution to a planning problem. We wish every agent to plan for himself, but come up with cooperative plans involving also the required actions of the other agents. Intuitively, we want the planning agent to be able to both verify the validity of the plan itself (that it reaches the goal), and to verify that each of the involved agents can do the same for their respective subplans given their local information.

Definition 7. Let $\Pi = \langle s_0, A, \omega, \varphi_g \rangle$ be a cooperative planning problem. An *implicitly coordinated plan* (or simply *plan*) for Π is a sequence $(a_1, \ldots, a_n)$ of actions from A such that:

$$s_0 \models K_{\omega(a_1)}(\!(a_1)\!) K_{\omega(a_2)}(\!(a_2)\!) \cdots K_{\omega(a_n)}(\!(a_n)\!)\varphi_g \tag{4}$$

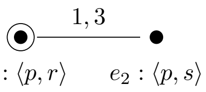
If Π is a planning problem for agent i, we call the plan a *plan for agent* i. A *plan for agent* i to a global planning problem Π is a plan for Π^i.

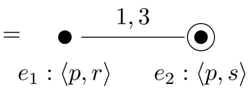
Note that a formula of the form $K_{\omega(a)}(\!(a)\!)\varphi$ expresses "the owner of action a knows that a is applicable and will lead to φ". Equation (4) hence expresses that a solution $(a_1, \ldots, a_n)$ should satisfy the following:

> *The owner of the first action a_1 knows that a_1 is initially applicable and will lead to a situation where the owner of the second action a_2 knows that a_2 is applicable and will lead to a situation where... the owner of the nth action a_n knows that a_n is applicable and will lead to the goal being satisfied.*

Example 4. Consider the cooperative planning problem $\Pi = \langle s_0, \{a_p, a_q\}, \omega, p \rangle$ over $(\{p, q\}, \{1, 2, 3\})$ where s_0, a_p and a_q are as in Example 3 and $\omega(a_p) = \omega(a_q) = 1$. Note that $\Pi = \Pi^i$ for all $i \in \{1, 2, 3\}$: The planning problem "looks the same" to all agents. The single-element sequence (a_p) is an (implicitly coordinated) plan for Π since $\omega(a_p) = 1$ and $s_0 \models K_1(\!(a_p)\!)\, p$ by (2). This is indeed a plan for all agents $i = 1, 2, 3$, since it is a plan for all Π^i. In other words, all agents can individually come up with the implicitly coordinated plan (a_p). They all know that all that has to be done is for agent 1 to execute a_p, and they know that agent 1 knows this himself.

Now consider extending the planning problem as follows. We add two more actions a_r and a_s given by:

$a_r =$ 

$a_s =$ 

Consider the cooperative planning problem $\Pi' = \langle s_0, \{a_p, a_q, a_r, a_s\}, \omega, r\rangle$ where ω is extended by $\omega(a_r) = \omega(a_s) = 2$. A successful plan is clearly (a_p, a_r) since $s_0 \models (\!(a_p)\!)(\!(a_r)\!)r$. However, it does not qualify as an implicitly coordinated plan for any of the agents, since it can be showed that $s_0 \not\models K_1(\!(a_p)\!)K_2(\!(a_r)\!)r$. The problem is that even if agent 1 starts out by executing a_p, agent 2 will not know he did (cf. Example 3), and hence agent 2 will not at runtime know that he can apply a_r to achieve r.

Consider adding a fifth action $p! := \odot\ e : \langle p, \top\rangle$ with $\omega(p!) = 1$. This is a public announcement of p by agent 1. Extending Π' with the action $p!$, all agents can again find an implicitly coordinated plan, namely $(a_p, p!, a_r)$: agent 1 first makes p true, then announces that he did, which makes agent 2 know that he can apply a_r to make r true.

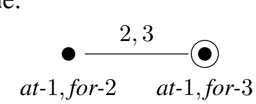
The following proposition gives a more structural characterization of implicitly coordinated plans. It thus becomes clear that such plans can be found by performing a breadth-first search over the set of successively applicable actions, shifting the perspective for each state transition to the owner of the respective action.

Proposition 2. *For a cooperative planning problem* $\Pi = \langle s_0, A, \omega, \varphi\rangle$,

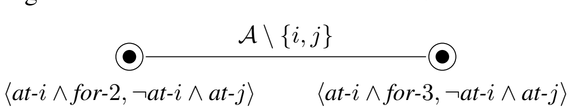
- $()$ *is an implicitly coordinated plan for* Π *iff* $s_0 \models \varphi$
- $(a_1, \ldots, a_n)$ *with* $n \geq 1$ *is an implicitly coordinated plan for* Π *iff* a_1 *is applicable in* ${s_0}^{\omega(a_1)}$ *and* $(a_2, \ldots, a_n)$ *is an implicitly coordinated plan for* $\langle {s_0}^{\omega(a_1)} \otimes a_1, A, \omega, \varphi\rangle$. □

The proof is simple and hence omitted (it relies on (1), Lemma 1 and (4)).

Example 5. As a more practical example, consider a situation with agents $\mathcal{A} = \{1, 2, 3\}$ where a letter is to be passed from agent 1 to one of the other two agents, possibly via the third agent. Mutually exclusive propositions *at*-1, *at*-2, *at*-3 $\in P$ are used to denote the current carrier of the letter, while *for*-1, *for*-2, *for*-3 $\in P$ denote the addressee. In our example, agent 1 has a letter for agent 3, so *at*-1 and *for*-3 are initially true.

$s_0 =$ 

In s_0, all agents know that agent 1 has the letter (*at*-1), but agents 2 and 3 do not know who of them is the addressee (*for*-2 or *for*-3). We assume that agent 1 can only exchange letters with agent 2 and agent 2 can only exchange letters with agent 3. We thus define the four possible actions $a_{12}, a_{21}, a_{23}, a_{32}$, with a_{ij} being the composite action of agent i publicly passing the letter to agent j and privately informing him about the correct addressee. I.e.

$a_{ij} =$ 

Given that the joint goal is to pass a letter to its addressee, the global planning problem then is $\Pi = \langle s_0, A, \omega, \varphi_g\rangle$ with

- $A = \{a_{12}, a_{21}, a_{23}, a_{32}\}$,
- $\omega = \{a_{12} \mapsto 1, a_{21} \mapsto 2, a_{23} \mapsto 2, a_{32} \mapsto 3\}$, and
- $\varphi_g = \bigwedge_{i \in \{1,2,3\}} (\textit{for-}i \rightarrow \textit{at-}i)$.

Consider the action sequence (a_{12}, a_{23}): Agent 1 passes the letter to agent 2, and agent 2 passes it on to agent 3. It can now be verified that

$$
\begin{aligned}
{s_0}^1 &\models K_1(\!(a_{12})\!)K_2(\!(a_{23})\!)\varphi_g \\
{s_0}^i &\not\models K_1(\!(a_{12})\!)K_2(\!(a_{23})\!)\varphi_g \quad \text{for } i = 2, 3
\end{aligned}
$$

Hence (a_{12}, a_{23}) is an implicitly coordinated plan for agent 1, but not for agents 2 and 3.

This is because in the beginning, agents 2 and 3 do not know for who of them the letter is intended and hence cannot verify that (a_{12}, a_{23}) will lead to a goal state. However, after agent 1's execution of a_{12}, agent 2 can distinguish between the possible addressees at runtime, and find his subplan (a_{23}), as contemplated by agent 1.

3.2 Conditional Plans

Sequential plans are often not sufficient to solve a given epistemic planning problem. In particular, as soon as nondeterministic action outcomes or splits on obtained observations come into play, we need *conditional* plans to solve such a problem. Consider for instance a problem where agents 1 and 2 need to cooperate as follows: Agent 1 starts out with an action a_1 that provides necessary information to agent 2 on how to continue, and depending on the new information, agent 2 needs to continue either with action a_2 or a_3 in order to achieve the joint goal. Unlike Andersen, Bolander, and Jensen (2012), who represent conditional plans as action trees with branches depending on knowledge formula conditions, we represent them as policy functions $(\pi_i)_{i \in \mathcal{A}}$, mapping minimal local epistemic states to actions for their respective observer agents.

First we define a few new pieces of notation. For the rest of this section, except in examples, we fix a set P of atomic propositions and a set $\mathcal{A}$ of agents. Hence all considered cooperative planning problems will be over the pair $(P, \mathcal{A})$ without this being mentioned explicitly. Given $i \in \mathcal{A}$, we use $S_i^{\min}$ to denote the set of minimal local states of agent i over $(P, \mathcal{A})$. We use S^{gl} to denote the set of global states over $(P, \mathcal{A})$. For any epistemic state $s = (\mathcal{M}, W_d)$ we let $\textit{Globals}(s) = \{(\mathcal{M}, w) \mid w \in W_d\}$. Hence $\textit{Globals}(s)$ is the set of global states "contained in" s.

We now define two different types of policies, *joint policies* and *global policies*, and later show them to be equivalent.

Definition 8 (Joint policy)**.** Let $\Pi = \langle s_0, A, \omega, \varphi_g\rangle$ be a cooperative planning problem. Then a *joint policy* $(\pi_i)_{i \in \mathcal{A}}$ consists of partial functions $\pi_i : S_i^{\min} \rightarrow A$ where for each $(s, a) \in \pi_i$, $\omega(a) = i$ and a is applicable in s.

Definition 9 (Global policy)**.** Let $\Pi = \langle s_0, A, \omega, \varphi_g\rangle$ be a cooperative planning problem. Then a *global policy* π is a mapping $\pi : S^{\text{gl}} \rightarrow \mathcal{P}(A)$ satisfying the requirements applicability (1), and uniformity (2), (3) below,

(1) For all $s \in S^{\mathrm{gl}}$, $a \in \pi(s)$: a is applicable in $s^{\omega(a)}$.
(2) For all $s \in S^{\mathrm{gl}}$, $a, a' \in \pi(s)$ with $a \neq a'$: $\omega(a) \neq \omega(a')$.
(3) For all $s, t \in S^{\mathrm{gl}}$, $a \in \pi(s)$ s.t. $s^{\omega(a)} = t^{\omega(a)}$: $a \in \pi(t)$.

Proposition 3. *Any joint policy $(\pi_i)_{i \in \mathcal{A}}$ induces a global policy π given by*

$$\pi(s) = \left\{ \pi_i(s^i) \mid i \in \mathcal{A} \text{ and } \pi_i(s^i) \text{ is defined} \right\}.$$

Conversely, any global policy π induces a joint policy $(\pi_i)_{i \in \mathcal{A}}$ given by

$$\pi_i(s^i) = a \quad \text{for all } (s, A') \in \pi,\ a \in A' \text{ with } \omega(a) = i.$$

Proof. First we prove that the induced mapping π as defined above is a global policy. Condition (1): If $a \in \pi(s)$ then $\pi_i(s_i) = a$ for some i, and by definition of joint policy this implies a is applicable in s^i. Condition (2): We prove the contrapositive. Assume $a, a' \in \pi(s)$ with $\omega(a) = \omega(a')$. By definition of π we have $\pi_i(s^i) = a$ and $\pi_j(s^j) = a'$ for some i, j. By definition of joint policy, $\omega(a) = i$ and $\omega(a') = j$. Since $\omega(a) = \omega(a')$ we get $i = j$ and hence $\pi_i(s^i) = \pi_j(s^j)$. This implies $a = a'$. Condition (3): Assume $a \in \pi(s)$ and $s^{\omega(a)} = t^{\omega(a)}$. By definition of π and joint policy, we get $\pi_i(s^i) = a$ for $i = \omega(a)$. Thus $s^i = t^i$, and since $\pi_i(s^i) = a$, we immediately get $\pi_i(t^i) = a$ and hence $a \in \pi(t)$. We now prove that the induced mappings $(\pi_i)_{i \in \mathcal{A}}$ defined above form a joint policy. Constraint (1) ensures the applicability property as required by Definition 8, while the constraints (2) and (3) ensure the right-uniqueness of each partial function π_i. □

By Proposition 3, we can identify joint and global policies, and will in the following move back and forth between the two. Notice that Definitions 8 and 9 allow a policy to distinguish between modally equivalent states. A more sophisticated definition avoiding this is possible, but is beyond the scope of this paper. Usually, a policy π is only considered to be a solution to a planning problem if it is closed in the sense that π is defined for all non-goal states reachable following π. Here, we want to distinguish between two different notions of closedness: one that refers to all states reachable from a centralized perspective, and one that refers to all states considered reachable when tracking perspective shifts. To that end, we distinguish between centralized and individual successor functions.

Definition 10. A successor function is a function $\sigma : S^{\mathrm{gl}} \times A \to \mathcal{P}(S^{\mathrm{gl}})$ mapping pairs of a global states s and applicable actions a to sets $\sigma(s, a)$ of possible successor states.

We can then define the *centralized* successor function as

$$\sigma_{cen}(s, a) = \mathit{Globals}(s \otimes a).$$

It specifies the global states that are possible after the application of a in s. If closedness of a global policy π based on the centralized successor function is required, then no execution of π will ever lead to a non-goal state where π is undefined. Like for sequential planning, we are again interested in the decentralized scenario where each agent has to plan and decide when and how to act by himself under incomplete knowledge. We achieve this by encoding the perspective shifts to the next agent to act in the *individual* successor function

$$\sigma_{ind}(s, a) = \mathit{Globals}(s^{\omega(a)} \otimes a).$$

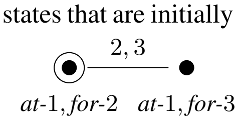
Unlike $\sigma_{cen}(s, a)$, $\sigma_{ind}(s, a)$ considers a global state s' to be a successor of s after application of a if *agent $\omega(a)$ considers s' possible* after the application of a, not only if *s' is actually possible from a global perspective*. Thus, $\sigma_{cen}(s, a)$ is always a (possibly strict) subset of $\sigma_{ind}(s, a)$, and a policy π_{ind} that is closed wrt. $\sigma_{ind}(s, a)$ must be defined for *at least* the states for which a policy π_{cen} that is closed wrt. $\sigma_{cen}(s, a)$ must be defined. This corresponds to the intuition that solution existence for decentralized planning with implicit coordination is a stronger property than solution existence for centralized planning. For both successor functions, we can now formalize what a strong solution is that can be executed collectively by the agents. Our notion satisfies the usual properties of strong plans (Cimatti et al. 2003), namely closedness, propriety and acyclicity.

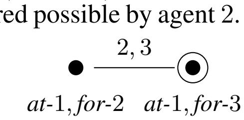
Definition 11 (Strong Policy)**.** Let $\Pi = \langle s_0, A, \omega, \varphi_g \rangle$ be a cooperative planning problem and σ a successor function. A global policy π is called a *strong policy* for Π with respect to σ if

(i) Finiteness: π is finite.
(ii) Foundedness: for all $s \in \mathit{Globals}(s_0)$,
  (1) $s \models \varphi_g$, or
  (2) $\pi(s) \neq \emptyset$.
(iii) Closedness: for all $(s, A') \in \pi$, $a \in A'$, $s' \in \sigma(s, a)$,
  (1) $s' \models \varphi_g$, or
  (2) $\pi(s') \neq \emptyset$.

Note that we do not explicitly require acyclicity, since this is already implied by a literal interpretation of the product update semantic that ensures unique new world names after each update. It then follows from (i) and (iii) that π is proper. We call strong plans with respect to σ_{cen} *centralized policies* and strong plans with respect to σ_{ind} *implicitly coordinated policies*.

Example 6. Consider again the letter passing problem introduced in Example 5. Let $s_{0,2}$ and $s_{0,3}$ denote the global states that are initially considered possible by agent 2.

$s_{0,2} =$ 

$s_{0,3} =$ 

With $s_{1,3} = s_{0,3} \otimes a_{12}$, a policy for agent 2 is given by

$$\pi_1 = \{ s_{0,3} \mapsto a_{12}, s_{0,2} \mapsto a_{12} \}, \pi_2 = \{ s_{1,3} \mapsto a_{23} \}.$$

After the contemplated application of a_{12} by agent 1 (in both cases), agent 2 can distinguish between $s_{1,2} = s_{0,2} \otimes a_{12}$, where the goal is already reached and nothing has to be done, and $s_{1,3}$, where agent 2 can apply a_{23}, leading directly to the goal state $s_{1,3} \otimes a_{23}$. Thus, π is an implicitly coordinated policy for Π^2. While in the sequential case, agent 2 has to wait for the first action a_{12} of agent 1 to be able to find its subplan, it can find the policy $(\pi_i)_{i \in \mathcal{A}}$ in advance by explicitly planning for a run-time distinction.

In general, strong policies can be found by performing an AND-OR search, where AND branching corresponds to branching over different epistemic worlds and OR branching corresponds to branching over different actions. By considering modally equivalent states as duplicates and thereby transforming the procedure into a graph search, space and time requirements can be reduced, although great care has to be taken to deal with cycles correctly.

It is easy to show that implicitly coordinated policies generalize implicitly coordinated plans.

Proposition 4. *Each implicitly coordinated plan* $(a_1, \ldots, a_n)$ *for* $\Pi = \langle s_0, A, \omega, \varphi_g \rangle$ *has a corresponding implicitly coordinated policy* π *for* Π.

Proof sketch. Let $s_i = s_0 \otimes a_1 \otimes \ldots \otimes a_i$. Then we can construct the policy π with $\pi(s'_i) = a_{i+1}$ for all $s'_i \in$ *Globals*(s_i) and all $i = 0, \ldots, n-1$. All three requirements of Definition 9 trivially hold. We further have to show that π is an implicitly coordinated policy for Π. Finiteness and foundedness are trivial. Closedness results from the correspondence to the recursive part of Proposition 2. □

4 Experiments

We implemented a planner that is capable of finding implicitly coordinated plans and policies, and conducted two experiments: one small case study of the Russian card problem (van Ditmarsch 2003) intended to show how this problem can be modeled and solved from an individual perspective, and one experiment investigating the scaling behavior of our approach on private transportation problems in the style of Examples 5 and 6, using instances of increasing size. Our planner is written in C++ and uses breadth-first search with an approximate bisimulation test that is used for state contraction and duplicate detection. All experiments were performed on a computer with a single Intel i7-4510U CPU core.

4.1 Russian Card Problem

In the Russian card problem, seven cards numbered $0, \ldots, 6$ are randomly dealt to three agents. Alice and Bob get three cards each, while Eve gets the single remaining card. Initially, each agent only knows its own cards. The task is now for Alice and Bob to inform each other about their respective cards using only public announcements, without revealing the holder of any single card to Eve. The problem was analyzed and solved from the global perspective by van Ditmarsch et al. (2006), and a given protocol was *verified* from an individual perspective by Ågotnes et al. (2010) before. We want to *solve* the problem from the individual perspective of agent Alice and find an implicitly coordinated policy for her. We only allow ontic announcements about hands, not about knowledge. To keep the problem computationally feasible, we impose some restrictions on the resulting protocol, namely that the first action has to be Alice announcing five possible alternatives for her own hand (one of which has to be her true hand), and that the second action has to be Bob announcing the card Eve is holding. Without loss of generality, we fix one specific initial hand for Alice, namely 012. From a plan for this initial hand, plans for all other initial hands can be obtained by renaming. For simplicity, we only generate *applicable* actions for Alice, i.e. announcements that include her true hand 012. This results in the planning problem having a total of 46376 options for the first action, and 7 for the second action. Still, the initial state s_0 consist of 140 worlds, one for each possible deal of cards. Agents can only distinguish worlds where their own hands differ. Alice's designated worlds in her associated local state of s_0 are those four worlds in which she holds hand 012.

Our planner, run in the conditional planning mode, needs 7282 seconds and approximately 630MB of memory to come up with a solution policy. In the solution, Alice first announces her hand to be one of 012, 034, 156, 236, and 245. It can be seen that each of the five hands other than the true hand 012 contains at least one of Alice's and one of Bob's cards, meaning that Bob will immediately be able to identify the complete deal. Since also every card occurs in exactly two of the five announced hands, of which at least one is considered possible by Eve, she stays unaware of the individual cards of Alice and Bob. Afterwards, Alice can wait for Bob to announce that Eve has either card 3, 4, 5 or 6 (which will not tell Eve anything new, either).

4.2 Mail Instances

Our second experiment concerns the letter passing problem from Examples 5 and 6. We generalized the scenario to allow an arbitrary number of agents with an arbitrary undirected neighborhood graph, indicating which agents are allowed to directly pass letters to each other. As neighborhood graphs, we used randomly generated Watts-Strogatz small-world networks (Watts and Strogatz 1998), exhibiting characteristics that can also be found in social networks. Watts-Strogatz networks have three parameters: The number N of nodes (determining the number of agents in our setting), the average number K of neighbors per node (roughly determining the average branching factor of a search for a plan), and the probability β of an edge being a "random shortcut" instead of a "local connection" (thereby influencing the shortest path lengths between agents). We only generate *connected* networks in order to guarantee plan existence.

We distinguish between the MAILTELL and the MAILCHECK benchmarks. To guarantee plan existence, in both scenarios the actions are modeled such as to ensure that the letter position remains common knowledge among the agents in all reachable states. The mechanics of MAILTELL directly correspond to those given in Example 5. There is only one type of action, publicly passing the letter to a neighboring agent while privately informing him about the final addressee. This allows for sequential implicitly coordinated plans. In the resulting plans, letters are simply moved along a shortest path to the addressee. In contrast, in MAILCHECK, an agent that has the letter can only check if he himself is the addressee or not using a separate action (without learning the actual addressee if it is not him). To ensure plan existence in this scenario, we allow an agent to pass on the letter only if it is destined for someone else. Unlike in MAILTELL, conditional plans are required here. In a solution (policy), the worst-case sequence of successively applied actions contains an action passing the letter to each agent at least once. As soon as the addressee has been reached, execution is stopped. Experiments were conducted for both scenarios with different parameters (see Tables 1, 2 and 3). For each set of parameters, 100 trials were performed. The tables show the average numbers of nodes that were created, expanded and discarded (because a bisimilar state was already considered somewhere else) in the search. For the conditional search in MAILCHECK, compound AND-OR nodes are used. In Table 1, *direct path* denotes the average shortest path length between sender and addressee, while in Tables 2 and 3, *full path* denotes the average length of a shortest path passing through all agents starting from the sender.

While the shortest path length between sender and addressee grows very slowly with the number of agents (due to the shortcut connections in the network), the shortest path passing through all agents roughly corresponds to the number of agents. Since these measures directly correspond to the minimal plan lengths, the observed exponential growth of space and time requirements with respect to them (and to the base K) is unsurprising.

Note also that in both scenarios, the number of agents determines the number of worlds (one for each possible addressee) in the initial state. Since the preconditions of the available actions are mutually exclusive, this constitutes an upper bound on the number of worlds per state throughout the search. Thus we get only a linear overhead in comparison to directly searching the networks for the relevant paths.

5 Conclusion and Future Work

We defined and studied an interesting new cooperative, decentralized planning concept without the necessity of explicit coordination or negotiation. Instead, by modeling all possible communication directly as plannable actions and relying on the ability of the autonomous agents to put themselves into each others shoes (using perspective shifts), some problems can be elegantly solved achieving implicit coordination between the agents. We briefly demonstrated an implementation of both the sequential and conditional solution algorithms and its performance on the Russian card problem and two letter passing problems.

An important starting point for further research concerns concrete problems (e.g. epistemic versions of established multi-agent planning problems) and the question of which kind of communicative actions the agents would need to solve these problems in an implicitly coordinated way. As seen in the MAILTELL benchmark, the dynamic epistemic treatment of a problem does not necessarily lead to more than linear overhead. It will be interesting to identify classes of tractable problems and see how agents cope in a simulated environment. Another issue that is relevant in practice concerns the interplay of the single agents' individual plans. In our setting, the agents have to plan individually and decide autonomously when and how to act. Also, when it comes to action application, there is no predefined notion of agent precedence. This leads to the possibility of *incompatible* plans, and in consequence to the necessity for agents having to *replan* in some cases. While our notion of implicitly coordinated planning explicitly forbids the execution of actions leading to *deadlock situations* (i.e. non-goal states where there is no implicitly coordinated plan for any of the agents), replanning can still lead to *livelocks*. Both the conditions leading to livelocks and individually applicable strategies to avoid them can be investigated. Since we need to be able to deal with replanning anyway, we can follow Andersen, Bolander, and Jensen (2013) and also investigate another successor function σ_{plaus} that maps only into the *most plausible* successor states. We expect this to lead to less branching and thus higher efficiency than the successor functions defined above.

agents	10	20	30	40	50
direct path	1.4	2.3	3.1	3.7	3.6
created	25	116	415	965	1023
expanded	7	36	142	347	359
discarded	2	32	166	455	443
time/s	0.02	0.14	0.65	2.38	5.02

Table 1: MAILTELL, $K = 4$, $\beta = 0.1$

agents	10	15	20	25	30
full path	10.4	16.1	21.7	27.6	33.2
created	402	2073	8065	35691	113481
expanded	361	1968	7771	34890	111582
discarded	552	3229	13126	59827	193555
time/s	0.02	0.18	1.14	8.03	38.76

Table 2: MAILCHECK, $K = 2$, $\beta = 0.1$

agents	7	9	11	13	15
full path	7.0	9.0	11.0	13.0	15.0
created	712	3161	13071	50104	188997
expanded	642	3027	12838	49739	188421
discarded	1859	9167	39528	154756	588582
time/s	0.03	0.21	1.14	5.90	27.30

Table 3: MAILCHECK, $K = 4$, $\beta = 0.1$

Acknowledgments

This work was partly supported by the DFG as part of the SFB/TR 14 AVACS. We thank Christian Becker-Asano for the fruitful discussions of earlier versions of this work.

References

Ågotnes, T.; Balbiani, P.; van Ditmarsch, H. P.; and Seban, P. 2010. Group announcement logic. *Journal of Applied Logic* 8(1):62–81.

Andersen, M. B.; Bolander, T.; and Jensen, M. H. 2012. Conditional epistemic planning. volume 7519 of *Lecture Notes in Computer Science*, 94–106.

Andersen, M. B.; Bolander, T.; and Jensen, M. H. 2013. Don't plan for the unexpected: Planning based on plausibility models. *To appear in Logique et Analyse, 2015*.

Andersen, M. B. 2015. *Towards Theory-of-Mind agents using Automated Planning and Dynamic Epistemic Logic*. Ph.D. Dissertation, Technical University of Denmark.

Bolander, T., and Andersen, M. B. 2011. Epistemic planning for single and multi-agent systems. *Journal of Applied Non-Classical Logics* 21(1):9–34.

Brenner, M., and Nebel, B. 2009. Continual planning and acting in dynamic multiagent environments. *Autonomous Agents and Multi-Agent Systems* 19(3):297–331.

Cimatti, A.; Pistore, M.; Roveri, M.; and Traverso, P. 2003. Weak, strong, and strong cyclic planning via symbolic model checking. *Artificial Intelligence* 147(1–2):35–84.

Kominis, F., and Geffner, H. 2015. Beliefs in multiagent planning: From one agent to many. In *Proceedings of the Twenty-Fifth International Conference on Automated Planning and Scheduling (ICAPS 2015)*. To appear.

Löwe, B.; Pacuit, E.; and Witzel, A. 2011. DEL planning and some tractable cases. In *LORI 2011*, volume 6953 of *Lecture Notes in Artificial Intelligence*, 179–192.

Muise, C.; Belle, V.; Felli, P.; McIlraith, S.; Miller, T.; Pearce, A. R.; and Sonenberg, L. 2015. Planning over multi-agent epistemic states: A classical planning approach. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence (AAAI 2015)*.

Nissim, R., and Brafman, R. I. 2014. Distributed heuristic forward search for multi-agent planning. *Journal of Artificial Intelligence Research (JAIR)* 51:293–332.

Palacios, H., and Geffner, H. 2009. Compiling uncertainty away in conformant planning problems with bounded width. *Journal of Artificial Intelligence Research (JAIR)* 35:623–675.

van Ditmarsch, H. P.; van der Hoek, W.; van der Meyden, R.; and Ruan, J. 2006. Model checking russian cards. *Electronic Notes in Theoretical Computer Science* 149(2):105–123.

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

van Ditmarsch, H. P. 2003. The russian cards problem. *Studia Logica* 75(1):31–62.

Watts, D. J., and Strogatz, S. H. 1998. Collective dynamics of 'small-world' networks. *Nature* 393(6684):440–442.