University Library



A gateway to Melbourne's research publications

Minerva Access is the Institutional Repository of The University of Melbourne

Author/s:
Miller, T;Pearce, AR;Sonenberg, L

Title:
Social Planning for Trusted Autonomy

Date:
2018

Citation:
Miller, T., Pearce, A. R. & Sonenberg, L. (2018). Social Planning for Trusted Autonomy. Abbass, HA (Ed.). Scholz, J (Ed.). Reid, DJ (Ed.). Foundations of Trusted Autonomy, Foundations of Trusted Autonomy, (1), 117, pp.67-86. Springer International Publishing.

Persistent Link:
https://hdl.handle.net/11343/297376

License:
CC BY

Chapter 4
Social Planning for Trusted Autonomy

Tim Miller, Adrian R. Pearce and Liz Sonenberg

4.1 Introduction

Early work on Trusted Autonomy (See Sect. 4.5) introduced the term *social autonomy* (See Chap. 1) to capture the idea that to be coordinated with other agents or keep its commitments, an agent must relinquish some of its autonomy, but that an agent that is sociable and responsible can still be autonomous: it would attempt to coordinate with others where appropriate and keep its commitments as much as possible, but it would exercise its autonomy in entering into those commitments in the first place [1].

It has been argued that human-machine trust can enhance performance in complex situations [2], and while we acknowledge there are many unanswered questions about the relationship between human-human trust, and human-machine trust, especially in the context of technology advances impacting machine capability for autonomy [3], we adopt the hopefully uncontroversial perspective that successful human-agent interaction demands that the agent behaves in an intuitive and explainable way from the perspective of the human.

So the work described here, on computational mechanisms for constructing and representing explainable plans in human-agent interactions, addresses one aspect of what it will take to meet the requirements of a trusted autonomous system. In turn, such properties are essential to enable the deployment of autonomous systems from the laboratory into production, such as in manufacturing assembly environments, assistive robotics, disaster management, defence applications, and self-driving cars.

Consider a simple example of a self-driving car that receives information that a road on its planned route is blocked. Re-planning the route to take a different road is straightforward, but the autonomy in the car should inform the passengers of this so

T. Miller (✉) · A. R. Pearce · L. Sonenberg
Department of Computing and Information Systems,
University of Melbourne, Melbourne, VIC 3010, Australia
e-mail: tmiller@unimelb.edu.au

A. R. Pearce
e-mail: adrianrp@unimelb.edu.au

L. Sonenberg
e-mail: l.sonenberg@unimelb.edu.au

© The Author(s) 2018

H. A. Abbass et al. (eds.), *Foundations of Trusted Autonomy*, Studies in Systems,
Decision and Control 117, https://doi.org/10.1007/978-3-319-64816-3_4

that they understand why an unusual route is taken. However, the autonomy should not inform the passengers if they are aware of this road closure already; for example, on the return trip.

We assert that scenarios such as this require *social planning*. Social planning is automated planning in which the planning agent maintains and reasons with an explicit model of the humans with which it interacts, including the human's goals, intentions(See Sect. 6.6), and belief, as well as their potential behaviours. Indeed, humans themselves use these concepts to make decisions that are intuitive, explainable, and acceptable to other people. This phenomenon is known as *Theory of Mind*(See Sect. 15.8), a term introduced by Premack and Woodruff in the context of the study of animal behaviour [4] and widely used since in philosophy, psychology amd cognitive science, e.g. [5, 6].

The state-of-the-art in artificial intelligence offers limited foundations on such constructs. Indeed, as articulated recently, challenges for artificial intelligence in the delivery of systems that can operate autonomously under some conditions, but cannot always complete an entire task on their own (so-called *semi-autononmous systems*), include the development of realtime activity and intent recognition techniques, the design of representations for human actions that are usable in the context of automated planning, integrated with interfaces that facilitate communication and transfer of control between the human and the machine, and supported by novel execution architectures [7]. The work described in this chapter addresses (in part) the second and fourth of these issues.

Specifically, we seek to build artificial agents that are able to fluidly operate in complex dynamic environments with humans, interacting in a 'human-intuitive' manner. We are developing building blocks towards the design of non-human agents whose actions can be trusted and understood by humans, and towards approaches that take these factors into account when designing the collaboration with humans.

The structure of this Chapter is as follows. So far we have offered an overview of the challenge of planning in human-agent teams, with a specific focus on social planning as one way to increase transparency and explainability, and hence a critical enabler of trust. Section 4.1 provides some high level background on (classical) planning and the motivation for social planning. Section 4.3 includes an introduction to a recent body technical work by the authors and collaborators in social planning - specifically in multi-agent epistemic planning [8–15]. In Sect. 4.4, we present two scenarios that illustrate the benefits of planning in the presence of nested belief reasoning and first-person multi-agent planning, hence indicating how social planning could be used as a means for planning human-agent interaction explicitly as part of the 'deliberation' cycle. Section 4.5 offers some brief summary remarks.

4.2 Motivation and Background

In this section, we outline some background material required to understand the chapter, as well as some motivation for our work.

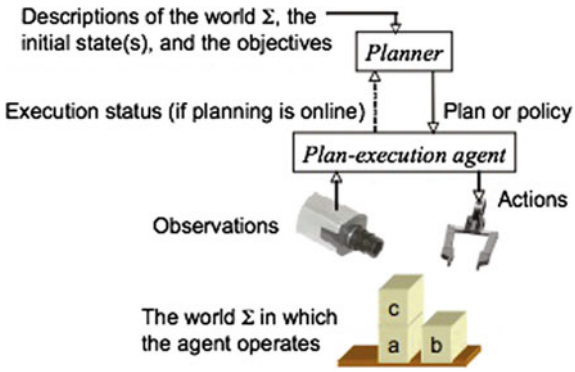


Fig. 4.1 Conceptual model of AI planning (from [16])

4.2.1 Automated Planning

Planning research in classical planning has yielded highly efficient mechanisms for plan synthesis suiting single-agent scenarios. Figure 4.1 outlines a conceptual model of AI planning. A planning problem is formulated as a tuple $\langle F, \mathcal{I}, \mathcal{G}, \mathcal{A} \rangle$, with the following meanings:

1. F is a set of Boolean fluents describing the objects within the world of interest;
2. $\mathcal{I} \subseteq F$ is the initial state represented as the set of Boolean fluents that are true in the world before the plan-execution agent performs any actions;
3. $\mathcal{G} \subseteq F$ is a set of fluents describing the desired objectives, such as achieving a goal or performing a specific task; and
3. $\mathcal{A}$ is a set of actions, described as a pair containing a precondition specifying the fluents that must be true for that action to be executed, and the effects that action will have on the world, described as fluents that will become true or false.

The output consists of either a plan (a sequence of actions for the agent to perform) or a policy (an action to perform for each reachable state).

A simple and commonly supported extension to classical planning is *conditional effects*. A conditional effect of an action is of the form $(\mathcal{C} \rightarrow l)$, in which $\mathcal{C}$ is a set of fluents representing a condition, and l is a single fluent. The informal semantics of such an effect is that *if* $\mathcal{C}$ held before the action was executed, then l holds after the action is executed. A single action can have multiple such conditional effects.

Much research over the last three decades has focused on the problem of *offline classical planning*, proposing compact state and transition encodings and effective domain-independent heuristics. This has led to massive improvements in classical planning tools, which can solve problems with hundreds of actions and large state spaces ($\approx 2^{1000}$ states) from several milliseconds to just a few hours.

However, classical planning is the simplest of the domain-independent planning problems, as it assumes the following:

- Deterministic events: The effects of all actions are deterministic — there is only one possible set of effects, and those effects happen each time the action is applied in the real world.
- Worlds change only as the result of an action: the only manner in which a world changes is when the planning agent executes an action — the world is otherwise static.
- Fully observable (omniscience): the state of the world is always fully observable — as such, when an action is applied, the agent can see the effects fully.
- Single actor (omnipotence): There are no other agents in the world, either cooperative, adversarial, and ambivalent.

Clearly, none of these assumption hold in the setting of real-world autonomy. However, more recently, research in the area of automated planning has focused on relaxing the problem description to enable a wider range of problems to be specified. In particular, planning in non-deterministic [17] and partially-observable [18] domains has matured to the point in which many problems in these domains can be solved efficiently offline, producing robust policies for execution. A key part of almost all solutions in this area is that a classical planning tool is used to solve part of the richer, underlying problem.

However, most planning research to date is still lacking in one key area: the consideration of other agents (human or otherwise) in the domain.

4.2.2 From Autistic Planning to Social Planning

To move into planning into multi-agent environments, agents must move out of the so-called *autistic* realm and into the *social* realm [19]. This means that a single agent reasoning in a multi-agent environment must have a Theory of Mind, considering the possible behaviours and mental states of others in the environment.

Building on recent analysis by Bolander and Herzig [20], we note that extending classical planning to the multi-agent case presents many new challenges:

1. Planners must track beliefs (or knowledge) of other agents, which are typically incomplete and only partially correct.
2. These beliefs include higher-order beliefs; that is, beliefs about other agents' beliefs about other agents' beliefs, etc. (as in Fig. 4.2).
3. Other agents have their own goals and intentions, which may be cooperative or competitive with our own, and these goals and intentions direct their actions, which influence our ability to achieve our own goals.
4. Chosen actions should be plausible or acceptable from the perspective of other agents; for example, in an adversarial setting in which an agent is attempting to conceal their real identity, their actions must conform to the identity attributed to them by their adversaries.

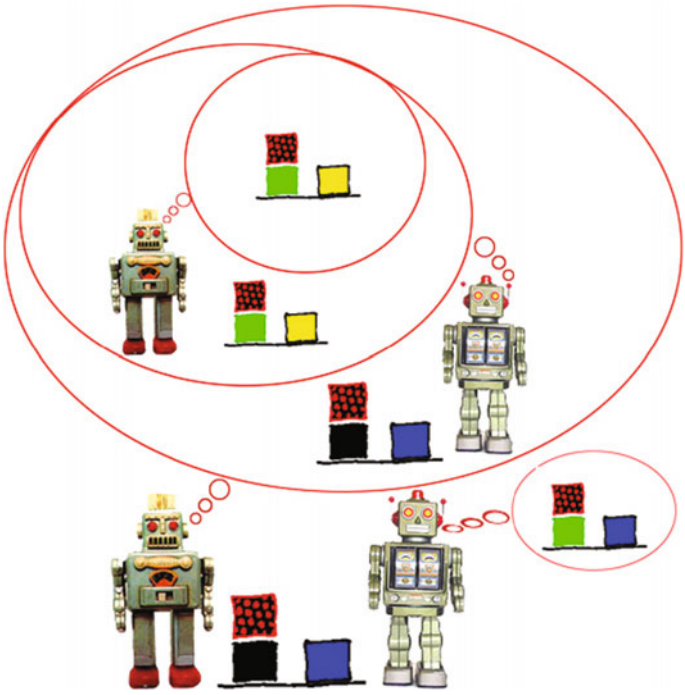


Fig. 4.2 Tracking others' beliefs about beliefs (taken from http://www2.compute.dtu.dk/~tobo/epistemic-planning-cph/index.php)

These present significant computational challenges: the actions of the other agents can induce a combinatorial explosion in the number of contingencies to be considered, making both the search space and the solution size exponentially larger, hence demanding novel methods [12, 21, 22].

The ability to hold a Theory of Mind to oneself and others, and to understand that others are doing the same, is important in many domains. Consider two fighter pilots seeking to disable an enemy radar defended by missiles. To do so, the pilots need to fool the enemy missile operators into believing that the two aircraft are attacking from the opposite direction to what they are truly attacking, in order to get close enough to the radar. Further, they need to attack simultaneously — one will destroy the radar while the other provides cover. However, they may be required to approach without communication, to reduce the chance of revealing their location. Thus, their agreed plan is to attack simultaneously only when they believe the enemy is deceived, *and* they believe that their team member believes that the enemy is deceived. To do this, they need to independently observe the same events as each other in the environment, and from these, update their theory of the others' mental state, as well as that of the enemy. Provided that both pilots are able to observe key events and understand that the observations of these events are common (known as *co-presence*), then they can coordinate their actions without communication.

In a first-order theory of mind, the reasoner considers that other people have beliefs, desires, etc. that influence their behaviour; e.g. *they believe we are attacking from the opposite direction*. In a second-order theory of mind, the reasoner allows that others are doing the same about us and other people; e.g. *my co-pilot believes that the enemy believe that we are attacking from the opposite direction*. In higher-order theories of mind, this nesting continues; *I believe that my co-pilot believes the enemy believe that we are attacking from the opposite direction, and I believe my co-pilot believes I believe this*.

Such reasoning has received much attention in empirical studies of children's and adults' reasoning, e.g. [6, 23–25] and there is considerable evidence that many adults have ToM abilities of levels 3 and 4, with some subjects succeeding in tasks requiring level 5 reasoning, yet even level 2 reasoning is beyond the reach of almost all state-of-the-art planning tools.

Multi-agent systems research has contributed a deep understanding of concepts such as group knowledge, group belief, and collective intention, often informed by philosophical and psychological perspectives, e.g. [26–30]. Studies have also examined computational models of ToM, e.g. [8, 31, 32], and also the impact of different levels of awareness that an agent has about the others acting in a team task context, e.g. [33]. Although the tools used in such investigations are highly expressive – typically description logics and rich multi-modal logics, and some bespoke algebraic belief update mechanisms – they are not accompanied by efficient reasoning engines, so fall short of providing practical means for systematically operationalising complex analyses.

Existing multi-agent planning tools that do take into account the beliefs, goals, intentions and capabilities of others, e.g. [34], consider a *third-person view*, in which a plan is constructed for a team, and each member is given their part to execute. When planning must be *distributed* amongst a team (including, when humans are to be in the loop), a semi-autonomous system must plan for its own actions while considering others explicitly - i.e. such reasoning demands a *first-person view*.

4.3 Social Planning

The authors, in conjunction with several collaborators, have made recent advances in this area; notably in the area of multi-agent epistemic planning. In this section, we overview two of the key advances made and provide a high-level technical overview of these. The two areas are:

1. *Efficient epistemic planning* — Bolander and Anderson [21] define the concept of *epistemic planning domains*, a generalisation of classical planning domains in which action models can have preconditions and effects on the (possibly nested) belief of others. They also show epistemic planning to be decidable in the single-agent case, but only semi-decidable in the multi-agent case.

In recent work, the authors, along with other collaborators, showed how restricted forms of epistemic knowledge bases can be used for efficient querying [10, 11, 15], and proposed a method that used these knowledge bases to take extend planning domains with higher-order belief operators, in a similar spirit to Bolander and Anderson's epistemic planning, and encode these as propositional planning problem [13]. The resulting encoding allows a large class of epistemic planning problems to be solved efficiently.

2. *First-person perspective multi-agent planning* — The authors and their collaborators propose a computational model for reasoning about and with others in multi-agent environments using heterogeneous *agent models* [8, 9], and subsequently instantiate this model as a non-deterministic planning problem [14]. The result is a planning tool that can produce policies for acting in a multi-agent environment, in which the policy has been compiled such that the agent considers the actions of others as it deliberates.

The latter item allows an agent to act in a multi-agent world considering the other agents' actions, while the former extends this with a Theory of Mind about the other agents' beliefs. Integrating these two pieces provides a tool for social planning: the ability to consider the possible behaviours and mental states (in this case, beliefs and goals) of others during the deliberation process.

4.3.1 A Formal Model for Multi-agent Epistemic Planning

In this section, we present a formal model for our multi-agent epistemic planning problem. This problem extends standard planning problems with the addition of epistemic fluents and multi-agent actions.

4.3.1.1 Epistemic Fluents

The notion of epistemic planning refers to the ability to reason about knowledge (or belief), rather than just about facts of the world. In the example of the two fighter pilots outlined in Sect. 4.2.2, these pilots are reasoning about the knowledge/beliefs of their partners as well as that of their adversary. Such reasoning is imperative for Theorem of Mind reasoning: to put oneself in the shoes of another, one must adopt their perspective of the world, including their understanding of the environment and others within it.

Epistemic logics extend standard propositional logics with *modal operators*, in which the mode of the formula represents the perspective of individual agents and groups of agents. First, we present some background material on epistemic and doxastic[1] logics that is required for this chapter. Throughout the remainder of this

[1] We use the term "epistemic" to refer to both knowledge and belief throughout the paper.

chapter, we will assume that the epistemic logic use is modal logic KD (see Fagin et al. [35] for a definition of this), and as such, is truly a belief operator, rather than a knowledge operator.

Due to the high computational complexity of epistemic logic, we adopt a simplified version of epistemic/doxastic logic by restricting modal formulae to *restricted modal literal* (RML) [36], proposed by Lakemeyer and Lespérance. An RML is defined using the following grammar:

$$\phi ::= p \mid [i]\phi \mid \neg\phi$$

where p is a propositional literal and i is an agent identified. Note that an RML cannot contain disjunctions, and is always in *negation normal form* (NNF). A set of RMLs, which is equivalent to their conjunction, is called a *proper epistemic knowledge base* (PEKB).

These RMLs are the fluents used in our epistemic planning problems: they offer an increase in expressiveness over propositional fluents, but as we will show later, they do not greatly increase the difficultly of solving the problem.

4.3.1.2 First-Person Multi-agent Planning

Similar to our earlier work [14], we define a first-person multi-agent planning problem as a tuple

$$\langle Ag, F, \mathcal{I}, \mathcal{G}_{i=0\cdots|Ag|-1}, \mathcal{A}_{i=0\cdots|Ag|-1}\rangle$$

where:

- Ag is the set of agents in the world, including the planning agent specially designated as 0;
- F is a set of epistemic fluents, in which each fluent is an RML;
- $\mathcal{I} \subseteq F$ is the initial state of the world;
- $\mathcal{G}_i \subseteq F$ is the goal for agent $i \in Ag$; and
- $\mathcal{A}_i$ is the finite set of actions agent i can execute.

Note the difference between this and the definition of classical planning outlined in Sect. 4.2: there is a set of agents associated with the problem definition, fluents can be epistemic, each agent has a goal, and actions are associated with particular agents.

Each action $a \in \mathcal{A}_i$ is a tuple of the form $\langle \text{Pre}_a, \text{Eff}_a\rangle$ where $\text{Pre}_a \subseteq F$ is the precondition that must hold for the action to be executed, and Eff_a is a *set* of one or more possible conditional effects, in which exactly one of the effects will hold after the execution of the action, but we do not know which until the action has been executed; that is, actions can be non-deterministic. We assume here that the non-deterministic effects are fully-observable; that is, the agents do not which outcome will occur, but they can observe the outcome immediately after the action is executed.

```
(:action share
     :derive-condition    (at $agent$ ?l)
     :parameters          (?a ?as - agent ?l - loc)
     :precondition        (and (at ?a ?l) [?a](secret ?as))
     :effect              (and
                              (forall ?a2 - agent
                                  (when     (at ?a2 ?l)
                                       [?a2](secret ?as)))
                          )
 )
```

Fig. 4.3 An epistemic PDDL description of sharing a secret

The set of all joint actions between agents is the cross product of all individual actions: $\mathcal{A} = \mathcal{A}_0 \times \cdots \times \mathcal{A}_{|Ag|-1}$. To model that it is possible for some agents to perform an action while others do not, (at least some) agents must be equipped with a "noop" (no operation) action, which has no effects.

Example 1 Consider the *Grapevine problem*, based on the well-known *gossip problem*, in which agents can move between rooms, share a secret piece of information in their room, but only those agents in the room will learn the secret when it is shared. The epistemic Planning Domain Description Language (PDDL) [37] extension of this action can be modelled as in Fig. 4.3.

In this example, `?l` is a room, `?a` is the agent sharing the secret, and `?as` are the other agents in the room. The fluent `[?a2](secret ?as)` means that agent `?a2` believes fluent `(secret ?as)`. Note that any agent can execute this action. Action preconditions can be used to restrict actions to only a subset of the agents in the domain.

The *derive condition* at the top of the action definition models the conditions of *mutual awareness*. Essentially, this says that for any agent in the room `?l`, they will derive the effects of this action if the action is executed. In essence, they will be *aware* that the action has been executed and will see its effect. They will therefore know the secret, but also know that all other agents in the room know the secret.

The types of goals one could consider in this example are: to share one's secret with only a subset of the agents; to deceive a particular set of agents; or to have every agent share their secret with everyone else.

A solution to a first-person multi-agent planning problem is a *policy* $P : 2^F \to \mathcal{A}_0$, thus mapping a partial state (a set of fluents) to an action specifying which action the 0 agent should take in a state that satisfies the partial state.

4.3.2 Solving Multi-agent Epistemic Planning Problems

While multi-agent epistemic planning problems are significantly more expressive than standard classical or contingent planning problems, they often can be solved with

some compilations to and modifications of existing — albeit advanced — planning technology.

As noted earlier, we solve this problem in two ways. First, we compile away the epistemic fluents in the planning problem into standard propositional fluents, such that any action defined using epistemic fluents can be compiled into an equivalent action and solved using an existing planner, such as a classical planner or non-deterministic planner. Second, by modifying an existing non-deterministic planning tool to consider multiple agents (without epistemic fluents), and then treating the effects of other agents actions as non-determinism in the environment. Thus, compiling a multi-agent epistemic planning problem into a multi-agent propositional planning problem and using this multi-agent planner, we can solve this rich class of problems.

4.3.2.1 Compiling Away Epistemic Fluents

There are several parts to the compilation – in this section we describe just the two most important: encoding consistent belief update; and encoding the perspective of other agents when the planning agent is unsure whether they witnessed an event. These both extend a *base encoding*, which strips away epistemic fluents are replaces them with propositional fluents suitable for our (non-epistemic) multi-agent planner. Technical details about this encoding can be found in Muise et al. [13]. In this section, we simply provide the intuition behind these via some examples.

Base Encoding. The base encoding describes a simple multi-agent planning problem that is not equivalent to the original problem. This encoding is then extended to deal with belief update and uncertain firing of events.

Put simply, the encoded problem takes the original problem and compiles it to an alternative problem such that each epistemic fluent in the action models, initial state, and goal is encoded into a proposition; that is, fluents of the form `[?a]p` are compiled to `a_p`. Thus, `a_p` represents the agent a believing p as a proposition. This replacement is nested for nested beliefs; for example, `[?a][?b][?c]p` is encoded as `a_b_c_p`. Negations of the form `not([?a]p)` are encoded as `not_a_p`.

Belief Update. In classical planning, belief update is straightforward: when a proposition becomes true, it is no longer false, and vice versa. However, in epistemic planning, the problem is not so simple. Consider the *Grapevine* example described in Example 1, in which agent 1 learns secret s, modelled as the epistemic fluent `[?a]s`. The propositional fluent `a_s` models this, however, we must also consider that if `[?a]s` is true, then so is `not([?a] not(s))` — if agent a believes s, then is should not believe the negation of s. Thus, for every compiled action in which `a_s` becomes true, so too must `not_a_not_s`. This counters for epistemic actions in which `not([?a] not(s))` is a precondition for example. If we add only `a_s` to the state, then `not_a_not_s` will not be true when that precondition is evaluated for another action. As such, the encoded model would not be equivalent without this modified belief update.

The reverse problem occurs if we want to no longer believe `not([?a] not(s))` – we must also remove `[?a]s`.

To solve these problems, one could modify our multi-agent underlying planner to know that whenever `a_s` is true, then `not_a_not_s` must also be true. Instead, we extend the base encoding by adding additional effects to actions that explicitly consider these situations, resulting in an encoding that faithfully encodes the dynamics of the original problem.

Compiling this down not only allows us to keep the epistemic and multi-agent parts of our solution loosely coupled – it means that our epistemic compilation tool can be used for other problems and other planners that support PDDL,[2] such as other classical planners, temporal planners, non-deterministic planners, etc.

Uncertain Firing. Consider again the *Grapevine* scenario, and an example in which we model the trustworthiness of agents. We may have a model of the `share` action that only believes a secret an agent shares if we believe that agent is trustworthy. For this, we would use a *conditional effect* on the action of the form `[0]trustworthy(?a) --> [0]secret(?a)`, meaning that we only add `[0]secret(?a)` to our state if `[0]trustworthy(?a)` was in our state before the action was executed (recall that the planning agent is agent 0). This models what is intended, but what if agent 0 is unsure whether agent *a* is trustworthy? That is, neither `[0]trustworthy(?a)` nor `[0]not(trustworthy(?a))` are in the state. Should `[0]secret(?a)` be added to the state?

Our intuition is that the solution should be to *remove* `[0]not (secret(?a))` (if it is in the state) if `not([0]trustworthy(?a))` holds before the action executes. Note here that `not([0]trustworthy(?a))` is not the same as `[0]not (trustworthy(?a))`. In the latter, we model that agent 0 believes that *a* is not trustworthy, while in the former we model that it is not the case that agent 0 believes *a* is trustworthy – agent 0 may be unsure of agent *a*'s trustworthiness.

Thus, we model that if 0 is unsure whether *a* is trustworthy, then it should *not* believe the secret, but it should at least no longer believe that the secret is false either: it should be uncertain whether the secret is true or not.

4.3.2.2 Multi-agent Problems as Non-deterministic Problems

The difference between single-agent and multi-agent planning problems is clear: in multi-agent planning problems, the agents must consider not only their own actions, but actions of other agents as well. For example, consider the simple two-player game *Tic-Tac-Toe*. When playing a move, we should not only consider whether we can get three pieces in a row, which is trivial in a single-player version, but also whether our opponent can block us or whether they can also get their own three pieces in a row.

One way to model other agents is to treat them as a dynamic environment. That is, when we execute an action, and another agent can subsequently change the world,

[2]Note: the underlying planner must support *conditional effects* for our compilation to work.

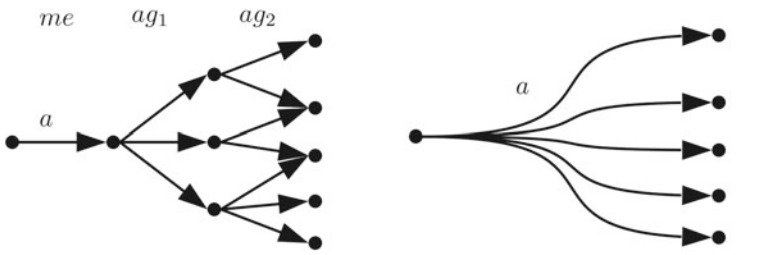


Fig. 4.4 Treating other agents' actions as non-determinism

we treat their action as arbitrary changes in a dynamic environment, such that it is as if the environment itself changed, rather than being explicitly changed by another agent.

Such an approach presents an opportunity to build on recent advances in *non-deterministic* planning [17] to extend planning technology to multi-agent environments. In non-deterministic planning, actions can have multiple possible effects, but the actual effect cannot be known until after the action is executed.

Using techniques in non-deterministic planning, we can cast the problem of planning in multi-agent environments as a non-deterministic planning task. Essentially, we can treat the actions of other agents in the environment as non-deterministic effects of our own actions.

Figure 4.4 outlines the intuition behind this idea. Figure 4.4: Left shows an agent *me* considering the execution of action a. If agents ag_1 and ag_2 will then subsequently perform actions, then from a deliberation perspective, the possible effects of executing action a should be consider as all possible effects of agents ag_1 and ag_2's actions. Figure 4.4: Right illustrates the non-deterministic treatment of this.

Modelling the problem like this results in a faithful encoding of the original problem; however, as noted earlier in Sect. 4.2, the actions of the other agents can induce a combinatorial explosion in the number of contingencies to be considered, making the search for solutions too high for all but the most trivial applications.

One way that we mitigate this problem is to consider the *intent* of the other agents in the scenario. That is, if we know/believe that the other agents have some particular intent, then we are able to reduce the branching factor by focusing the search only on those actions that are *plausible* given the other agents' intent.

For example, consider *Tic-Tac-Toe*. We know our opponent's goal: to win the game. Given this, if we are planning what to do in the state of the game shown in Fig. 4.5, where the opponent is **O** and we are player **X**. If it is our move, one possible move is to place **X** in the bottom right corner, which sets us up to win along the bottom row. We then need to consider player **Y**'s moves. A rational agent would consider that player **O**'s most plausible response is to play in the top-right corner, winning the game. Player **O**'s next most plausible move (if one can really consider

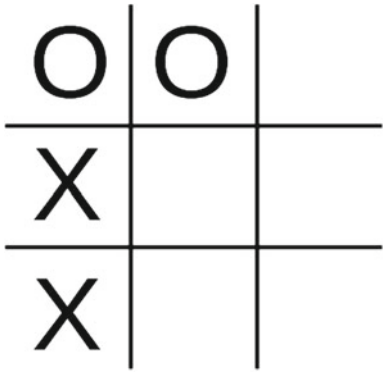


Fig. 4.5 A sample game state of *Tic-Tac-Toe*

any such move as plausible!), would be to block the cell at the middle bottom, thus preventing us from winning.

Our search algorithm considers this by looking at the other agents' goals and using standard planning search heuristics to decide which actions are the best for the other agents when assessing what they may do. It uses these heuristics to rank the opponents move from most plausible to least plausible. Then, it considers the agent's most plausible action first at each stage of a scenario, until the search terminates. Then, it considers the next most plausible action, and so on.

Using the Tic-Tac-Toe example, our algorithm would consider player **Y** first playing top-right, then it would consider middle-bottom. Other moves are implausible and it is little value to explore them. The search continues until either: (a) the entire search terminates, in which case we have a complete solution and the plausibility ranking is meaningless; or more likely (b) a pre-defined time or memory budget is exhausted, at which point it has the best move considering the search space that it has explored.

This search strategy is highly effective in many domains, because it does not assume complete rationality of the other agents; nor that the model we have of the other agents is complete. That is, rather than determine exactly which action other agents will choose, it considers all, but only reasons about the effects of the most plausible ones. Given enough time and memory, this will result in a complete search, but for large problems, it focuses the search on those actions that are the most likely.

Technical details of the problem formulation, solution, and evaluation of this approach can be found in Muise et al. [14].

4.4 Social Planning for Human Robot Interaction

To demonstrate the benefits of social planning, we present two case studies involving semi-autonomous teams, which we have adapted to illustrate the benefits of planning in the presence of nested belief and first-person multi-agent planning.

4.4.1 Search and Rescue

Disaster response and management involves a number of important tasks, such as preparation before disaster, response and restoration. If we consider a scenario of a simplified search and rescue mission following a natural disaster, such as an earthquake, there are a series of tasks that must be undertaken to search for survivors and get them to the appropriate service, such as medical evacuation. As part of this, we can imagine a scenario in which two unmanned ground vehicles (UGVs) and a human operator are working as a three-member semi-autonomous team to locate and assess survivors.

The environment of this scenario consists of a set of buildings, organised according to a known map. However, the buildings may be damaged, leading to unexpected inaccessibility to search regions or locations. Buildings may contain survivors or could be empty, but this is initially unknown.

The human supervisor oversees the entire mission and coordinates the UGVs. They can interact with the agents controlling the vehicles by assigning goals, such as to search a particular building or to return back to particular base location. They can also query the agents on their current goals, intentions, and beliefs; including the nested beliefs of the other agent and the operator themselves.

The two UGVs have the same capabilities, which can be modelled as actions in epistemic PDDL, such as:

1. Moving to specific way-points, identified by coordinates on the map, including inside buildings.
2. Attempt to open doors to buildings/rooms (which may either succeed or fail).
3. Go into buildings and rooms, providing the doors are open.
4. Take a picture and upload it for assessment.
5. Drop a first-aid survival pack, provided that the agent believes there is a survivor in need of this.
6. Drop water to a survivor.
7. Lift a survivor onto one of the vehicles, which requires the assistance of the other vehicle.
8. Communicate with the other agent or the supervisor.

This final action — communication — is enabled by the epistemic actions. Communication can simply be modelled as an action with epistemic effects. For example, one agent can send a message to the others indicating that a particular door is blocked (e.g. by rubble).

Such an action can be modelled as follows:

Parameters The parameters are the agents to which to send the message, and the location of the door that is blocked.

Precondition The precondition is that the sending agent believes that the door is blocked, and believes the recipient does not believe that the door is blocked.

Effects The effects are that the sending agent believes that the receiving agents believe that the door is blocked, and further, the sending agent believes that the receiving agent believes that the sending agent believes that the door is blocked. Similarly, an agent receiving such a message would believe that the door is blocked and that the sender believes that the door is locked.

Although this scenario is a simplification of a real search and rescue mission, it illustrates the implications that explicitly modelling these communication actions has on the scenario. We assert that these go some way towards improving the interactions between the semi-autonomous team, such as:

Efficiency It may be that the other vehicle agent could want to use that door later as part of a plan, and now knows to plan a different route to get inside the building/room that does not used this blocked door.

Lower Communication Overhead By explicitly representing the beliefs of others (and potentially their nested beliefs of the team), the amount of communication overhead can be reduced. For example, the precondition of the action above is that the receiving agents do not already believe this information. Thus, if the planning agent already has information noting that another agent beliefs the door is blocked, it will not send this information on. This is particularly important in semi-autonomous teams, in which human collaborators are more easily overloaded with information than their artificial team members.

Re-planning Having an explicit model of the other agents' Theory of Mind, and being able to update this model, enables agents to identify that their expectation of their team members' behaviour is no longer valid, thus triggering them to re-assess or re-plan the intentions and plans of their team members.

Coordination As outlined in the task of the fighter pilots in Sect. 4.2.2, providing updates of each others mental states allows agents to synchronise on joint actions that require e.g. simultaneous execution, such as lifting a survivor onto one of the vehicles to transport them back to a location with further medical assistance, which may require that each agent believes that the survivor is on the stretcher and believes that the other agent believes this as well.

Transparency It provides some transparency to the human supervisor, informing them why the agent is not entering the room that it had originally planned, without (at least in some cases) the agent having to explicitly update the supervisor on its new plan.

Epistemic goals Being able to model the epistemic effects allow us to pose epistemic goals, such as that agent A believes something is true while agent B believes the opposite — in other words, one of the agents is deceived.

While it is straightforward to model communication actions in other planning languages, the ability to model the *epistemic effects* of these actions, and have these effects represented as a Theory of Mind, enables additional possibilities over using propositional planning, particularly regarding coordination and transparency.

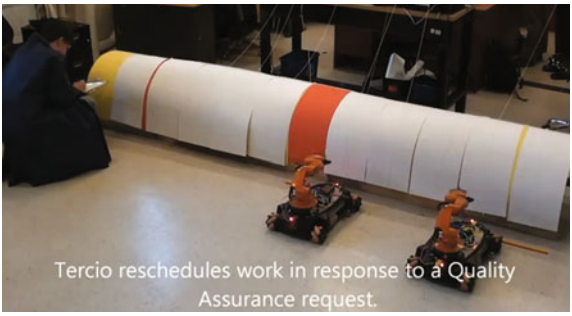


Fig. 4.6 Industrial painting robots with optional human inspection; presented in [39], the video can be found at http://tiny.cc/2aytrw

Further to this though, we assert that epistemic first-person multi-agent problem is a more natural way to model these problems, compared to existing approaches, such as keeping a separate model for each other agent [38].

In particular, the ability to model and reason about epistemic goals requires the ability to model basic multi-agent epistemic effects — they cannot be captured with separate models.

4.4.2 Collaborative Manufacturing

In Fig. 4.6 we consider a variant of a painting and assessment task presented in [39] where robot actions are only *partially observable* to human operators. The task involves the real-time scheduling of painting robots for the fuselage of an aeroplane. Human operators optionally intervene in the painting process to assess the quality of the painted surfaces.

The painting robots must adapt and re-schedule to the optional assessment of panels by human operators, to allow the panels to have sufficient drying time and to achieve the goal of painting the fuselage in the time allocated to the task. The temporal constraints are captured in this task using a form of temporal constraint networks termed simple temporal networks (STNs) [40].[3] The time required for the application of each coat is captured using the STNs, along with the time before the (optional) assessment of each coat; including the assessment time. From the perspective of the painting robots, paths through STNs emerge according to the non-deterministic choice of humans according to which panels they chose to assess. This forms a branching tree, similar to Fig. 4.4: Left, as painting robots and human operators interleave painting and assessment tasks. At any instant, there is a *minimal* STN that achieves all of the tasks within a minimum time, which will be traversed according to the optional assessments that are potentially performed by human operators in the future.

[3] See [39] for the STN encoding details for this task and the video at http://tiny.cc/2aytrw.

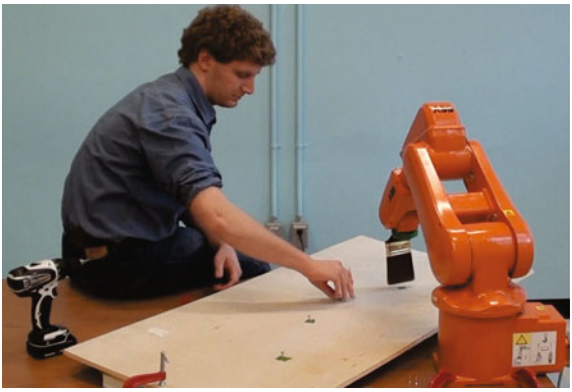


Fig. 4.7 Industrial assembly with human-robot interaction; presented in [41], the video can be found at http://tinyurl.com/7n439eg

If the robots work too far down the fuselage from the human operators, humans cannot distinguish *which* panels the robots are painting. If nested belief is used during planning this allows robots to choose panels to paint which are observable to human assessors. The robots know that actions observable to the human operators allow the humans to infer the minimal STN. The robots therefore know that humans know the robots know the humans know the minimal STN. Humans can therefore understand the choice of panel robots make to paint; and can even take this choice into account in deciding which panel next to assess. Thus, theory of mind facilitates the robots to maintain human knowledge of which panel(s) are ready to assess. Social planning builds trust between robots and human operators—leading to goal achievement within shorter times.

In another industrial task, shown in Fig. 4.7, an assembly task is shown where a human and a robot share in a simple assembly task that involves placing fasteners then applying torque to each fastener. The robot shares the task by applying sealant to each hole ahead of placement of the fastener. The robot must be able to handle different preferences of human operators. For example, operators may choose to place all the fasteners first, then apply torque to each one. Alternatively operators may choose to place each fastener then apply torque to each one immediately following placement. The approach described in [41] shows an approach that can adapt to the preferences of humans using dynamic scheduling, the video can be found at http://tinyurl.com/7n439eg.

We adapt this task to utilise social planning. If we assume the goal is to minimise the overall time to complete the task, theory of mind facilitates robots to adapt to humans changing their preferred assembly behaviour part-way through the achievement of task, as they learn to perform the task within less time. Using theory of mind principles, the robot uses social planning in the knowledge that the human knows the robot knows the human has learned the shortest human-robot interleaving strategy. This enables the robot to perform other preparatory tasks, such as fetching and positioning the correct number of fasteners, further shortening the time to complete the task.

4.5 Discussion

We have presented an an outline of several principal elements of the emerging field of social planning. These include theory of mind, as we move to first-person perspective planning in a multi-agent setting, and we present a formal model for first-person multi-agent epistemic planning. We have covered two emerging solution techniques for solving multi-agent epistemic planning problems, including an approach for compiling away epistemic fluents, where multi-agent problems are posed as non-deterministic problems, for which solutions are quite well understood. Finally, we presented two case studies of semi-autonomous systems by adapting examples from the literature to utilise social planning and theory of mind principles to demonstrate the benefits for realising trusted autonomy. These examples demonstrate how social planning can used to improve the interaction between humans and robots in semi-autonomous teams.

The work forms an important step towards achieving trusted autonomy where the perspective of both humans and robots are explicitly modelled using a first-person theory of mind approach. There is excellent potential for the exploitation of recent developments in efficient epistemic and non-deterministic reasoning techniques. For example, recent techniques in proper epistemic databases such as 'knowing whether' [10] can be used to *establish the knowledge* of human operators during more complex tasks without knowing the knowledge itself, and the observability of *asynchronously occurring* actions can even be modelled [42]. Further work and experimentation is warranted to explore the application of these and other related techniques in social planning.

References

1. M.N. Huhns, D.A. Buell, Trusted autonomy. IEEE Internet Comput. **6**(3), 92–95 (2002)
2. H.A. Abbass, E. Petraki, K. Merrick, J. Harvey, M. Barlow, Trusted autonomy and cognitive cyber symbiosis: open challenges. Cogn. Comput. **8**(3), 385–408 (2016)
3. S. Wheeler, Trusted autonomy: conceptual developments in technology foresight. Technical Report DSTO-TR-3153, Defence Science and Technology Group (DSTG), 2015
4. D. Premack, G. Woodruff, Does the chimpanzee have a theory of mind? Behav. Brain Sci. **1**(4), 515–526 (1978)
5. J. Call, M. Tomasello, Does the chimpanzee have a theory of mind? 30 years later. Trends Cogn. Sci. **12**(5), 187–192 (2008)
6. A.I. Goldman, Theory of mind, in *Oxford Handbook of Philosophy and Cognitive Science*, Chap. 17, ed. by E. Margolis, R. Samuels, S. Stich (Oxford University Press, Oxford, 2012), pp. 201–213
7. S. Zilberstein, Building strong semi-autonomous systems, in *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, Austin, Texas, USA*, 25–30 Jan 2015, pp. 4088–4092
8. P. Felli, T. Miller, C.J. Muise, A.R. Pearce, L. Sonenberg, Artificial social reasoning: computational mechanisms for reasoning about others, in *Social Robotics—6th International Conference, ICSR 2014, Sydney, NSW, Australia, October 27–29, 2014. Proceedings* (2014), pp. 146–155

9. P. Felli, T. Miller, C.J. Muise, A.R. Pearce, L. Sonenberg, Computing social behaviours using agent models, in *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina*, 25–31 July 2015, pp. 2978–2984
10. T. Miller, P. Felli, C.J. Muise, A.R. Pearce, L. Sonenberg, 'Knowing whether' in proper epistemic knowledge bases, in *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, Phoenix, Arizona, USA*, 12–17 Feb 2016, pp. 1044–1050
11. T. Miller, C.J. Muise, Belief update for proper epistemic knowledge bases, in *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA*, 9–15 July 2016, pp. 1209–1215
12. T. Miller, A. Pearce, L. Sonenberg, F. Dignum, P. Felli, C. Muise, Foundations of human-agent collaboration: situation-relevant information sharing, in *2014 AAAI Fall Symposium Series* (2014)
13. C.J. Muise, V. Belle, P. Felli, S.A. McIlraith, T. Miller, A.R. Pearce, L. Sonenberg, Planning over multi-agent epistemic states: a classical planning approach, in *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, Austin, Texas, USA*, 25–30 Jan 2015, pp. 3327–3334
14. C.J. Muise, P. Felli, T. Miller, A.R. Pearce, L. Sonenberg, Planning for a single agent in a multi-agent environment using FOND, in *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA*, 9–15 July 2016, pp. 3206–3212
15. C.J. Muise, T. Miller, P. Felli, A.R. Pearce, L. Sonenberg, Efficient reasoning with consistent proper epistemic knowledge bases, in *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2015, Istanbul, Turkey*, 4–8 May 2015, pp. 1461–1469
16. T.-C. Au, U. Kuter, D. Nau, Planning for interactions among autonomous agents, in *International Workshop on Programming Multi-Agent Systems* (Springer, Berlin, 2008), pp. 1–23
17. C.J. Muise, S.A. McIlraith, J. Christopher Beck, Improved non-deterministic planning by exploiting state relevance. In *Proceedings of the Twenty-Second International Conference on Automated Planning and Scheduling, ICAPS 2012, Atibaia, São Paulo, Brazil*, 25–19 June 2012
18. H. Palacios, H. Geffner, Compiling uncertainty away in conformant planning problems with bounded width. J. Artif. Intell. Res. **35**, 623–675 (2009)
19. F. Dignum, G.J. Hofstede, R. Prada. From autistic to social agents, in *Proceedings of the 12th International AAMAS Conference* (2014), pp. 1161–1164
20. T. Bolander, A. Herzig, Group attitudes and multi-agent planning: overview and perspectives (2014), Accessed at http://www.sintelnet.eu/wiki/reports/Thomas_Bolander_WG3_report.pdf
21. T. Bolander, M.B. Andersen, Epistemic planning for single- and multi-agent systems. J. Appl. Non Class. Logics **21**(1), 9–34 (2011)
22. A. Pearce, L. Sonenberg, P. Nixon, Toward resilient human-robot interaction through situation projection for effective joint action, in *Robot-Human Teamwork in Dynamic Adverse Environment: AAAI Fall Symposium* (2011), pp. 44–48
23. A. Brandenburger, X. Li, Thinking about thinking and its cognitive limits (2015), http://adambrandenburger.com/articles/papers/. Accessed Sept 2016
24. T. Kneeland, Identifying higher-order rationality. Econometrica **83**(5), 2065–2079 (2015)
25. H.D. Schlinger, Theory of mind: an overview and behavioral perspective. Psychol. Rec. **59**(3), 435–448 (2009)
26. M. Gilbert, Modelling collective belief. Synthese **73**(1), 185–204 (1987)
27. R. Hakli, Group beliefs and the distinction between belief and acceptance. Cogn. Syst. Res. **7**(2), 286–297 (2006)
28. L. Lismont, P. Mongin, On the logic of common belief and common knowledge. Theor. Decis. **37**(1), 75–106 (1994)
29. R. Tuomela, W. Balzer, Collective acceptance and collective social notions. Synthese **117**(2), 175–205 (1998)

30. H. van Ditmarsch, J. van Eijck, R. Verbrugge, Common knowledge and common belief, in *Discourses on Social Software*, ed. by J. van Eijck, R. Verbrugge, vol. 5 of *Texts in Logic and Games* (2009), pp. 99–122
31. L. Van Maanen, R. Verbrugge, A computational model of second-order social reasoning, in *Proceedings of the 10th International Conference on Cognitive Modeling* (2010), pp. 259–264
32. H. Weerd, R. Verbrugge, B. Verheij, Negotiating with other minds: the role of recursive theory of mind in negotiation with incomplete information, in *Autonomous Agents and Multi-Agent Systems* (2015), pp. 1–38
33. H. De Weerd, R. Verbrugge, B. Verheij, How much does it help to know what she knows you know? An agent-based simulation study. Artif. Intell. **199**, 67–92 (2013)
34. R.F. Kelly, A.R. Pearce, Asynchronous knowledge with hidden actions in the situation calculus. Artif. Intell. **221**, 1–35 (2015)
35. Y.M.R. Fagin, J.Y. Halpern, M.Y. Vardi, *Reasoning about Knowledge* (MIT Press, Cambridge, MA, 1995)
36. G. Lakemeyer, Y. Lespérance, Efficient reasoning in multiagent epistemic logics, in *European Conference on Artificial Intelligence* (2012), pp. 498–503
37. D. McDermott, M. Ghallab, A. Howe, C. Knoblock, A. Ram, M. Veloso, D. Weld, D. Wilkins, PDDL—The Planning Domain Definition Language (1998)
38. V.V. Unhelkar, J.A. Shah, Contact: Deciding to communicate during time-critical collaborative tasks in unknown, deterministic domains, in *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, Phoenix, Arizona, USA*, 12–17 Feb 2016, pp. 2544–2550
39. M.C Gombolay, R. Wilcox, J.A Shah. Fast scheduling of multi-robot teams with temporospatial constraints, in *Robotics: Science and Systems IX*, Technische Universität Berlin, Germany, June 2013, pp. 49–56
40. R. Dechter, I. Meiri, J. Pearl, Temporal constraint networks. Artif. Intell. **49**(1–3), 61–95 (1991)
41. R. Wilcox, S. Nikolaidis, J. Shah, Optimization of temporal dynamics for adaptive human-robot interaction in assembly manufacturing, in *Robotics Science and Systems VIII* (2012), pp. 441–448
42. R.F. Kelly, *Asynchronous Multi-Agent Reasoning in the Situation Calculus*. Ph.D. University of Melbourne (2008)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (http://creativecommons.org/licenses/by/4.0/), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

