A Formalization of Equilibria for Multiagent Planning

Michael Bowling, Rune Jensen, and Manuela Veloso
Computer Science Department
Carnegie Mellon University
Pittsburgh PA, 15213-3891

1 Introduction

Traditionally, planning involves a single agent for which a planner needs to find a sequence of actions that can transform some initial state into some state where a given goal statement is satisfied. But in general, "planning" can be viewed as being concerned with a general action selection problem. The planning framework has been extended from the classical deterministic plan generation problem along many other dimensions, in particular nondeterministic actions. With the introduction of nondeterministic actions, the presence of an environment and other agents can become a consideration. In fact, actions may have nondeterministic effects not only because of the uncertainty of their own execution, but also due to the uncertainty of the actions of other agents. The possible presence of other agents as executors is the challenge of "multiagent planning." The interest in this area has been steadily increasing and many issues remain open.

Despite the existence of planning languages with explicit models of uncontrallable agents [Jensen and Veloso, 2000], we believe that there has not been a formal discussion of the space of multiagent plans or *solutions.* In this work, we do not focus on the problem of plan generation for multiagent planning. Instead, we focus on the interesting question of analyzing and comparing solutions for multiagent planning. Our motivation comes from making an analogy with game theory and the notion of equilibria [Owen, 1995].

Inspired by game theory and extending previous formal definitions of single-agent planning [Cimatti *et al,* 2000], in this paper, we introduce a formal definition of *equilibria* for multiagent planning.

2 Planning Equilibria

In this section, we formalize the concept of a multiagent planning equilibrium. In order to help make these concepts clear we will first describe an example, which is small enough to easily enumerate all of the states.

2.1 A Simple Example — The Narrow Doorway

Consider a two agent robot domain where both agents are in a hallway and want to move into the same room through a single doorway. The agents have an operator to go through the door (G) that only succeeds if the other agent is not also trying to go through the door. They also have the choice of waiting (W). Each agent's goal is simply to be in the room.

2.2 The Formalization

We first begin by formalizing some planning related concepts. The definitions parallel closely with Cimatti and colleagues' single-agent formalization [2000]. We extend their definitions of planning domains, problems, and solutions to encompass multiple agents. We then follow this formal framework with a definition of multiagent planning equilibrium. Notice that the definitions and concepts presented are not bound to any particular planning algorithm or language.

We start by defining a multiagent planning domain.

Definition 1 (Multiagent Planning Domain)
A multiagent planning domain D is a tuple $\langle \mathcal{P}, \mathcal{S}, n, \mathcal{A}_{i=1\ldots n}, \mathcal{R} \rangle$ *where,*

- *$\mathcal{P}$ is the finite set of propositions,*
- *$\mathcal{S} \subseteq 2^{\mathcal{P}}$ is the set of valid states,*
- *n is the number of agents,*
- *$\mathcal{A}_i$ is agent i's finite set of actions, and*
- *$\mathcal{R} \subseteq \mathcal{S} \times \mathcal{A} \times \mathcal{S}$ is a nondeterministic transition relation where $\mathcal{A} = \mathcal{A}_1 \times \ldots \times \mathcal{A}_n$ and must satisfy the following condition. If $\langle s, a, s' \rangle \in \mathcal{R}$ and $\langle s, b, s'' \rangle \in \mathcal{R}$ then, $\forall i$ there exists $s''' \in \mathcal{S}$,*

$$\langle s, \langle a_1, \ldots, a_{i-1}, b_i, a_{i+1}, \ldots, a_n \rangle, s''' \rangle \in \mathcal{R}.$$

I.e., each agent's set of actions that can be executed from a state are independent.

In addition, let $ACTi(s) \subseteq A_i$ be the set of applicable or executable actions in state *s.* Formally,

$$\text{ACT}_i(s) = \{a_i \in \mathcal{A}_i \mid \exists \langle s, \langle \cdots, a_i, \cdots \rangle, \cdot \rangle \in \mathcal{R}\}.$$

The additional condition in the planning domain definition on R requires that each agent be capable of selecting actions independently. Formally this amounts to the following. For all states *s* and executable actions for the agents $a_i \in Act_i(s)$ there exists some transition $\langle s, \langle a_{i=1\ldots n} \rangle, s' \rangle$ that is in 7v.

In the doorway domain, *V* contains two propositions, *A-in-room* and *B-in-room.* The set of states *S* corresponds to all four possible subsets of P*,* since all combinations of propositions are valid in this domain, *n* is two and $\mathcal{A}_{A,B}$ is the set of actions {G, *W}.* The transition relation $\mathcal{R}$ is defined by the rules described above. The complete enumeration of states and transitions is shown in Figure 1. The figure also numbers the states so they can be referred to in an abbreviated

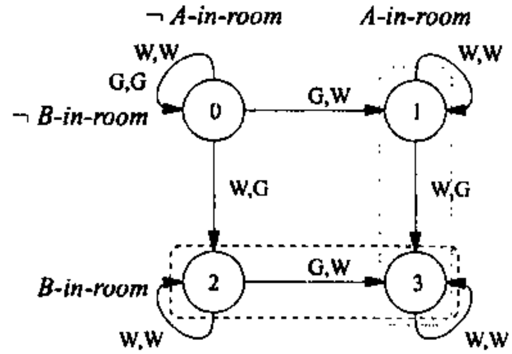

Figure 1: Doorway domain. ... an ... resent As and B\s goal states, $\mathcal{G}_A$, and $\mathcal{G}_B$, respectively.

form. Note that this domain satisfies the independent action condition on R.

Definition 2 (Multiagent Planning Problem)
Let $\mathcal{D} = \langle \mathcal{P}, \mathcal{S}, n, \mathcal{A}_{i=1...n}, \mathcal{R}\rangle$ be a multiagent planning domain. A multiagent planning problem P for $\mathcal{D}$ is a tuple $\langle \mathcal{D}, \mathcal{I}, \mathcal{G}_{i=1...n}\rangle$, where $\mathcal{I} \subseteq \mathcal{S}$ is the set of possible initial states and $\mathcal{G}_i \subseteq \mathcal{S}$ is the set of goal states for agent i.

In the doorway example, the goal states for agent A are {1,3} and for agent B are {2,3}. The initial state set is the singular set {0}. With this definition of domain and problem, we can now formalize a notion of a plan.

Definition 3 (State-Action Table)
A state-action table π_i for agent i in domain V is a set of pairs $\{\langle s, a_i\rangle \mid s \in \mathcal{S}, a_i \in \text{ACT}_i(s)\}$. A joint state-action table π constructed from state-action tables for each agent $\pi_{i=1...n}$ is the set of pairs

$$\{\langle s, \langle a_1, \ldots, a_n\rangle\rangle \mid s \in \mathcal{S}, \langle s, a_i\rangle \in \pi_i\}$$

A (joint) state-action table is complete if and only if for any $s \in S$ there exists some pair $(s, \cdot)$ in the state-action table.

For the doorway domain, a state-action table (or plan) for each agent might be,

$$\begin{aligned} \pi_A &= \{\langle 0, G\rangle, \langle 1, W\rangle, \langle 2, G\rangle, \langle 2, W\rangle\ \langle 3, W\rangle\}, \\ \pi_B &= \{\langle 0, G\rangle, \langle 0, W\rangle, \langle 1, G\rangle, \langle 2, W\rangle, \langle 3, W\rangle\}. \end{aligned}$$

These are also *complete* state-action tables since they specify at least one action for each state. We can combine these tables into a *complctcjoint* state-action table. In general, a joint state-action table together with a multiagent planning domain determines the entire execution of the system. In order to define what it means for a plan to be a solution to a planning problem we need to formalize the notion of reachability and paths of execution. We will do this by first defining the execution structure of the multiagent system.

Definition 4 (Induced Execution Structure)
Let π be a joint state-action table of a multiagent planning domain $\mathcal{D} = \langle \mathcal{P}, \mathcal{S}, n, \mathcal{A}_i, \mathcal{R}\rangle$. The execution structure induced by π from the set of initial states $\mathcal{I} \subseteq \mathcal{S}$ is a tuple $K = \langle Q, T\rangle$ with $Q \subseteq \mathcal{S}$ and $T \subseteq \mathcal{S} \times \mathcal{S}$ inductively defined as follows:

- *if $s \in \mathcal{I}$, then $s \in Q$, and*
- *if $s \in Q$ and there exists a state-action pair $\langle s, a\rangle \in \pi$ and transition $\langle s, a, s'\rangle \in \mathcal{R}$, then $s' \in Q$ and $\langle s, s'\rangle \in T$.*

A state $s \in Q$ is a terminal state of K if and only if there is no $s' \in Q$ such that $\langle s, s'\rangle \in T$.

Intuitively, Q is the set of states that the system could reach during execution of the plan π, and T is the set of transitions that the system could cross during execution. For our doorway domain the execution structure induced by our example joint state-action table is,

$$\begin{aligned} Q &= \{0, 1, 3\}, \\ T &= \{\langle 0, 1\rangle, \langle 0, 0\rangle, \langle 1, 3\rangle, \langle 1, 1\rangle, \langle 3, 3\rangle\}. \end{aligned}$$

We can now formalize an execution path.

Definition 5 (Execution Path)
Let $K = (Q, T)$ be the execution structure induced by a state-action table π from X. An execution path of K from $s_0 \in T$ is a possibly infinite sequence $s_0, s_1, s_2, \ldots$ of states in Q such that, for all states s_i in the sequence:

- *either s_i is the last state of the sequence, in which case S_i is a terminal state of K, or*
- *$\langle s_i, s_{i+1}\rangle \in T$.*

A state s' is reachable from a state s if and only if there is an execution path with $s_0 = s$ and $s_i = s'$.

For our doorway domain and example joint state-action table one execution path from the initial state is,

$$0, 0, 0, 0, 1, 1, \ldots$$

We are now in a position to define what it means for our plan to solve a planning problem. We actually define multiple concepts increasing in strength. These concepts formalize some of the intuitive discussion from the previous section about whether a plan has one or more of the following properties:

- the possibility of reaching the goal,
- a guarantee of reaching the goal, and
- a guarantee of reaching the goal in a finite number of steps.

These concepts and their formalization are inspired and highly related to Cimatti and colleagues' single-agent solution concepts [Cimatti *et al.*, 2000].

Definition 6 (Multiagent Planning Solutions)
Let V be a multiagent planning domain and $P = \langle \mathcal{D}, \mathcal{I}, \mathcal{G}_{i=1...n}\rangle$ be a multiagent planning problem. Let π be a complete *joint state-action table for V. Let $K = (Q,T)$ be the execution structure induced by π from 1. The following is an ordered list of solution concepts increasing in strength.*

1. *π is a weak solution for agent i if and only if for any state in I some state in $\mathcal{G}_i$ is reachable.*
2. *π is a strong cyclic solution for agent i if and only if from any state in Q some state in $\mathcal{G}_i$ is reachable.*
3. *π is a strong solution for agent i if and only if all execution paths, including infinite length paths, from a state in Q contain a state in $\mathcal{G}_i$.*

4. *π is a perfect solution for agent i if and only if for* all *execution paths $s_0, s_1, s_2 \ldots$ from a state in Q there exists some $n \geq 0$ such that $\forall i \geq n,\ s_i \in \mathcal{G}_i$.*

A state-action table's strength $\textsc{Strength}(\mathcal{D}, \mathcal{P}, i, \pi)$ *is the largest number whose condition above applies for agent i. If no conditions apply then* $\textsc{Strength}(\mathcal{D}, \mathcal{P}, i, \pi) = 0$.

For our doorway domain, the joint state-action table is a strong cyclic solution for both agents but not strong (i.e., it has a strength of 2 for both agents). This means that there is a path to the goal from any reachable state. But there are also paths that do not include either agents' goal states, and so it is not a strong solution for either agent.

These solutions define what it means for one agent to be successful given a joint state-action table. The goal of planning from one agent's perspective is to find a plan that has the highest strength given the plans of the other agents. But the other agents' selection of a plan is equally contingent upon the first agent's plan. This recursive dependency leads to our main contribution of the paper: planning equilibria.

Definition 7 (Multiagent Planning Equilibria)
Let V be a multiagent planning domain and V - $\langle \mathcal{D}, \mathcal{I}, \mathcal{G}_{i=1\ldots n} \rangle$ be a multiagent planning problem. Let π be a complete *joint state-action table for V. Let K* — (Q, *T) be the execution structure induced by π from X. π is an equilibrium solution to V if and only if for all agents i and for any complete joint state-action table π' such that $\pi'_{j \neq i} = \pi_j$,*

$$\textsc{Strength}(\mathcal{D}, \mathcal{P}, i, \pi) \geq \textsc{Strength}(\mathcal{D}, \mathcal{P}, i, \pi').$$

I.e., each agent's state-action table attains the strongest solution concept possible given the state-action tables of the other agents.

Note that our example joint state-action table for the doorway domain is *not* an equilibrium. Both agents A and B currently have strength 2, but B can achieve a strength of 4 by choosing a different state-action table. Specifically, B should select the wait (W) action from the initial state and the go (G) action in state 1.

To make the concept of planning equilibria clearer, we illustrate it in the doorway domain. We gave above an example joint state-action table that is not a multiagent planning equilibria for this domain. An equilibria is the following state-action tables:

$$\begin{aligned} \pi_A &= \{\langle 0, G\rangle, \langle 1, W\rangle, \langle 2, G\rangle, \langle 3, W\rangle\}, \\ \pi_B &= \{\langle 0, W\rangle, \langle 1, G\rangle, \langle 2, W\rangle, \langle 3, W\rangle\}. \end{aligned}$$

In this case agent A goes through the door while agent B waits and then follows through the door. This is a perfect plan for both agents and so obviously no agent can achieve a higher strength with a different state-action table. Similarly, the symmetric tables where agent B goes through the door while agent A waits is also an equilibrium. There is an additional equilibrium,

$$\begin{aligned} \pi_A &= \{\langle 0, G\rangle, \langle 0, W\rangle, \langle 1, W\rangle, \langle 2, G\rangle, \langle 3, W\rangle\}, \\ \pi_B &= \{\langle 0, G\rangle, \langle 0, W\rangle, \langle 1, G\rangle, \langle 2, W\rangle, \langle 3, W\rangle\}. \end{aligned}$$

Here both agents nondeterministically decide between going through the door and waiting. This results in a strong cyclic solution for both agents, but given this state-action table for the other agent no strong or perfect plan exists for either agent. So this is also an equilibrium although obviously inferior to the other equilibria where both agents have higher strength plans. In game theory, such a joint strategy is called Pareto dominated.

Collision variation. Consider a variation on this domain where collisions (when both agents choose G) result in the robots becoming damaged and unable to move. In this case, the first two state-action tables above remain equilibria, but the third inferior table no longer is an equilibrium. This joint plan is now only a weak solution for both agents since there is a possibility of never achieving the goal. Each agent can also change to a different plan where it waits for the other agent to get through the door thus achieving a strong cyclic plan and a higher strength.

Door closing variation. Finally, consider that one agent entering the room sometimes causes the door to close behind it. Once the door is closed it cannot be opened and the doorway cannot be used. In this case, the same two joint plans are again an equilibrium but now they have different strengths for the different agents. The first joint state-action table is a strong plan for agent A, but only a weak plan for agent B, though it can do no better. The second is just a symmetry of this.

3 Conclusion

We presented a formalization of multiagent planning and introduced the concept of a multiagent planning equilibrium. This is the first known solution concept that explicitly accounts for the goals of all the agents. This work provides a unifying framework for considering planning in multiagent domains with identical, competing, or overlapping goals. It also opens up many exciting questions related to practical algorithms for finding equilibria, the existence of equilibria, and the coordination of equilibria selection.

Acknowledgments

This research was sponsored by grant No. F30602-00-2-0549. The content of this publication does not necessarily reflect the position of the funding agencies and no official endorsement should be inferred.

References

[Cimatti *et ai*, 2000] Alessandro Cimatti, Marco Pistore, Marco Roveri, , and Paolo Traverso. Weak, strong, and strong cyclic planning via symbolic model checking. IRST Technical report 0104-11, Istituto Trentino di Cultura, April 2000.

[Jensen and Veloso, 2000] Rune Jensen and Manuela Veloso. OBDD-based universal planning for synchronized agents in non-deterministic domains. *Journal of Artificial Intelligence Research,* 13:189-226, 2000.

[Owen, 1995] Guillermo Owen. *Game Theory.* Academic Press, 1995.