Integrating Multiagent Dialogues, Planning and Plan Execution

Yuqing Tang
Dept. of Computer Science,
Graduate Center, City University of New York
365 Fifth Avenue, New York, NY 10016, USA
ytang@cs.gc.cuny.edu
Advised by: Simon Parsons

Abstract

Coming up with a plan for a team that operates in a non-deterministic environment is a complex process, and the problem is further complicated by the need for team members to communicate during the process of planning and the execution of plans. Dialogues can be used to coordinate both the planning process and the plan execution. In this work I develop a model for constructing joint plans for a team of agents that takes into account their communication needs while accommodating multiple agents’ inconsistent specifications of transitions, initial states, and goals. The model builds on recent developments in symbolic non-deterministic planning, ideas that have not previously been applied to this problem.

Introduction

One of the fundamental problems in multiagent systems is how to get a team of agents to coordinate their behavior. While there are situations in which agents can do this without needing to communicate (Genesereth, Ginsberg, and Rosenschein 1988), in general coordination requires communication. Another important part of coordination is having the agents decide what to do. Since (Bratman, Israel, and Pollack 1988), the process of deciding what to do is considered to break down into two parts — deciding what goals to achieve, what (Bratman, Israel, and Pollack 1988) calls deliberation, and then deciding how those goals might best be achieved, which is usually described as planning. In this work I am interested in the planning part of the process while the problem of establishing joint intentions (Cohen and Levesque 1991) is tackled through non-monotonic reasoning on goals.

I am also greatly concerned with communication. Much recent work on agent communication uses argumentation-based dialogue (Parsons and McBurney 2003), and the long-term goal of our work is to extend existing work on multiagent planning by developing models by which a team of agents can, in the course of an argumentation-based dialogue, influence the external world as a policy, and we consider it to simply be a set of state-action transitions one should take in a given state. We define a non-deterministic state transition domain (NSTD) to be a tuple \( M = (P, S, A, R) \) where:

- \( P = P_S \cup P_A \) is a finite set of propositions;
- \( S \subseteq 2^P_S \) is the set of all possible states;
- \( A \subseteq 2^P_A \) is the set of finite set of actions; and
- \( R \subseteq S \times A \times S \) is the state-transition relation.

We model how the agents can influence the external world as a policy, and we consider it to simply be a set of state-action transitions that takes into account the need to communicate during the process of planning and the plan execution. In this work I develop a model for constructing joint plans for a team of agents that takes into account their communication needs while accommodating multiple agents’ inconsistent specifications of transitions, initial states, and goals. The model builds on recent developments in symbolic non-deterministic planning, ideas that have not previously been applied to this problem.
pairs,
\[ \pi = \{(s_i, a_i)\} \]
where \( s_i \in S \) and \( a_i \in A(s) \) with \( A(s) = \{a \mid \exists s, a, s' \in R\} \) that is the set of actions that are applicable in \( s \). It is the state-action table used of (Cimatti et al. 2003). It is also related to what the literature on MDPs calls a policy (Boutilier, Dean, and Hanks 1999).

Over the proposition variables, a propositional language \( L \) with quantification extension can be defined by allowing standard connectives \( \land, \lor, \to, \neg \) and quantifiers \( \exists, \forall \). The resulting language is a logic of quantified boolean formulae (QBF) (Bryant 1992). A symbol renaming operation, which we use below, can be defined on \( L \). For a formula \( \xi \in L \), if \( \vec{x} \) and \( \vec{x}' \) are two vectors of propositional variables, then a variable renaming operation can be defined by \( \xi[\vec{x} / \vec{x}'] \) which means that all the appearances of variables \( \vec{x} \) are substituted by \( \vec{x}' \).

**Multiagent Systems**

A multiagent system is composed of \( N \) agents, \( AGS = \{T_1, \ldots, T_N\} \). We model each agent \( T_i \) with a NSTD \( M_i = \langle P_i, S_i, A_i, R_i, \pi_i, \rangle \) where \( P_i = P_{i,S} \cup P_{i,A} \) and associated with a local policy \( \pi_i \). The multiagent system as a whole is modeled as a joint NSTD \( M = \langle P, S, A, R \rangle \) along with a joint policy \( \pi \). The set of proposition variables in the joint model is the union of those of individual agents: \( \mathcal{P}_S = \bigcup_{i=1}^{N} P_{i,S} \) and \( \mathcal{P}_A = \bigcup_{i=1}^{N} P_{i,A} \). The joint states, actions, and state transitions are the result of interpreting \( \mathcal{P}_S \) and \( \mathcal{P}_A \) as in the basic NSTD. The individual transition relation and the policy can be projected from the joint system: \( R_i = \exists P_i R \) and \( \pi_i = \exists P_i \pi \) where \( P_i \) means the set of variables in the formula being quantified but not in \( P_i \).

To distinguish from the dialogue model below, we call the individual models \( M_i \)s and the joint model \( M \), and the associated polices \( \pi_i \)s and \( \pi \) the external NSTDs and polices.

**Dialogue Systems**

In addition to the external transition system \( M_i \), each agent \( T_i \) is also associated with dialogue transition system \( M_{i,D} = \langle P_{i,D}, S_{i,D}, A_{i,D}, R_{i,D}, \pi_{i,D} \rangle \) and a dialogue policy \( \pi_{i,D} \). The joint dialogue system is then defined as, \( M_D = \langle P_D, S_D, A_D, R_D, \pi_D \rangle \), along with a joint dialogue policy \( \pi_D \). The set of proposition variables in the joint model is the union of those of individual agents: \( \mathcal{P}_{S,D} = \bigcup_{i=1}^{N} P_{i,S,D} \) and \( \mathcal{P}_{A,D} = \bigcup_{i=1}^{N} P_{i,A,D} \). The joint dialogue states, actions, and state transitions are the result of interpreting \( \mathcal{P}_{S,D} \) and \( \mathcal{P}_{A,D} \) as in the basic NSTD. The individual transition relation and the policy can be projected from the joint dialogue system: \( R_{i,D} = \exists P_i R D \) and \( \pi_{i,D} = \exists P_i \pi D \). Different from the external policy \( \pi \), the requirement for a dialogue policy \( \pi_D \) is that the individual agent can execute the projected \( \pi_{i,D} \) without knowing the other agents’ dialogue states so that the dialogue can help the multiagent system coordinate the joint external policy execution.

**Handling Inconsistent Specifications**

In a multiagent system, different agents, acting on behalf of different users, may have inconsistent information about the transitions, the initial and goal states. We need a mechanism to accommodate these inconsistent information, while the whole system can still reach an agreement on the joint model. Using the STRIPS-style specifications, for example STRIPS (Fikes and Nilsson 1971), PDDL (Ghallab et al. 1998), ADL (Pedaault 1994) as the front-end specifications to our multiagent specifications, each agent \( T_i \) maintains a tuple of information
\[ \langle TR_i, FR_i, MR_i, I_i, G_i \rangle \]
where
- \( TR_i = \{TR_{i,k}\} \) is a list of local state transition specifications,
- \( FR_i = \{FR_{i,k}\} \) is a list of frame specifications,
- \( MR_i = \{MR_{i,k}\} \) is a list of multiagent interaction constraints,
- \( I_i = \{I_{i,k}\} \) is a list of initial state specifications, and
- \( G_i = \{G_{i,k}\} \) is a list of goal state specifications.

The entries in \( TR_i \) and \( FR_i \) depends only on the variables in \( P_i \), and entries in \( MR_i \) can have any variables in \( P \) in general. Here, we take \( FR_{i,k} = p_i,k \leftrightarrow p_i,k' \) for each state variable in \( P_{i,S} \) to specify that \( p_i,k \) doesn’t change by default.

Each entry in \( TR_i, FR_i, MR_i, I_i, G_i \) is associated with a label variable. The association is in the form of a one-to-one mapping
\[ \langle LABEL_{i,k}, INPUT_{i,k} \rangle \]
stands for \( \{lt_{i,k}, TR_{i,k}\}, \{lf_{i,k}, FR_{i,k}\}, \{lm_{i,k}, MR_{i,k}\}, \{li_{i,k}, I_{i,k}\}, \text{and} \{lg_{i,k}, G_{i,k}\} \). Correspondingly, these variables are grouped into \( P_{i,L} \), where * stands any of \( TR, FR, MR, I, G \). With these labeling variables, we can encode the selection of a set of specifications using QBFs. Let \( \sigma \subseteq \bigcup_{i=1}^{N} (TR_i \cup FR_i \cup MR_i) \), we define
\[ SEL(\sigma) = \bigwedge_{TR_{i,k} \not\in \sigma} \neg (lt_{i,k}) \land \bigwedge_{FR_{i,k} \not\in \sigma} \neg (lf_{i,k}) \land \bigwedge_{MR_{i,k} \not\in \sigma} \neg (lm_{i,k}) \]
\[ \land \bigwedge_{TR_{i,k} \in \sigma} (lt_{i,k}) \land \bigwedge_{FR_{i,k} \in \sigma} (lf_{i,k}) \land \bigwedge_{MR_{i,k} \in \sigma} (lm_{i,k}) \]
\[ SEP(\sigma) = SEL(\sigma) \land \bigwedge_{TR_{i,k} \in \sigma} TR_{i,k} \land \bigwedge_{FR_{i,k} \in \sigma} FR_{i,k} \land \bigwedge_{MR_{i,k} \in \sigma} MR_{i,k} \]

Let
\[ C\{INPUT\}_i = \bigwedge_{k} \{LABEL\}_i,k \to \{INPUT\}_i,k \]
where \( \{INPUT\} \) stands any of \( TR, FR, MR, I, G \), and \( \{LABEL\} \) stands for the corresponding label variables to encode the local state transition specification combinations (CTR), frame specifications (CFR), inter-agent specifications respectively (CMR), initial states (CI) and goal states (CG). Now we can encode all the consistent combinations with linear number of BDD operations:
Algorithm 1 Weak Planning

1: function weakPlan(R, I, G) 
   (1) R: The transition relation; 
   (2) I: The set of initial states; (3) G: The set of goals states 
2: FT ← G, SA ← ∅, CoveredStates ← FT 
3: repeat 
4:   newSA ← computePreImage(R, FT') 
5:   FT ← ℓnewSA, FT' 
6:   ∆SA ← prune(SA, newSA) 
7:   SA ← SA ∪ ∆SA 
8:   CoveredStates ← CoveredStates ∨ FT 
9: until ∆SA = ∅ OR (I ⊆ CoveredStates) 
10: return SA 
11: end function

Algorithm 2 Agent T_i’s Pre-Image Computation

1: function computePreImage(CTR_i, FT') 
   (1) CTR_i: Agent T_i’s local combination on transition specifications; (2) FT': The frontier states 
2: Compute CTR_i ∧ FT' and ¬CTR_i ∧ FT' 
3: for j = 1 . . . N, and j ≠ i do 
4:   Send CTR_j ∧ FT' to agent T_j 
5: end for 
6: Compute ¬CTR_i ∧ FT' to agent T_i 
7: Wait for CTR_i ∧ FT' and ¬CTR_i ∧ FT' from all other agent T_k (k = 1, . . . , N and k ≠ i) 
8: return S_A 
9: return 
10: end function

Decentralized Planning

By decomposing the expression of R (Equation 1), we have

\[ R \land FT' = SHR_{TR} \land \bigwedge_{i=1}^{N} (CTR_i \land FT') \land \forall_{P_{LTR}} \left( \bigcup_{i=1}^{N} (\neg CTR_i \land FT') \lor \neg \xi_{LTR} \lor \tilde{\xi}_{LTR} \lor FT' \right) \]

where SHR_{TR} is an expression for the information shared by all the agents, \( \xi_{LTR} \) and \( \tilde{\xi}_{LTR} \) are QBFs to encode the subset (⊆) and superset (⊇) relations on the selection of \( T_{R_i} \)'s. Assuming that each agent T_i knows all the labeling symbols of \( P_{LTR} \), then

- \( CTR_i \land FT' \) can be computed by agent T_i locally
- \( \neg CTR_i \land FT' \) can be computed by agent T_k locally

With this decomposition, we can then have a distributed version of computePreImage as in Algorithm 2.

Proposition 2. If every agent follows the algorithm 2 and algorithm 1, the every agent computes a same joint policy.

Proof. Every agent obtains the same joint state frontier FT' at each search step. The agents following algorithm 2 computes the same pre-image. Therefore, when the algorithm 1 end, all the agents will compute a same joint policy.

Note that the concern of the decentralized planning is not to save computations but to have all the agents reach consensus on a joint policy.

Integrating Dialogues

In (Tang, Norman, and Parsons 2009a) I integrate the planning of external policies with the planning of dialogue policies. The approach incorporates the information of external states, state transitions and policy into the dialogue states. Dialogue actions are then used to communicate the truth values of the variables carrying this information between agents. Dialogue state transitions are automatically generated regarding the ontology of the external states and actions, and the external policy. A dialogue policy is planned by setting the goal dialogue states in which the external joint states and actions are available.
• fully communicated with each other, or
• partially communicated to a level in which every agent can compute a unique local action agreeing with their shared joint policy

**Implementations**

I have been working on a C++ implementation of the system (Tang, Norman, and Parsons 2009b). The focus is on providing utilities to specify the complicated ontology over the propositional variables needed to implement multiagent planning and dialogues. In the implementation, I first specify the ontology for planning and dialogues, and then the system will automatically generate the BDD variables and BDDs for the QBFs. Currently the underlying BDD computation is implemented using my own naive implementation for the convenience of tuning the ontology manipulations. After the model becomes stable, I plan to bridge our implementation to a proven BDD package for efficiency and stability.

**Conclusions**

This paper has presented a model to integrate multiagent dialogues, planning, and plan execution. The model is symbolic, and capable of handling non-deterministic actions. In addition to the model, we have provided procedures for creating joint plans in a centralized or decentralized manner. In combination with our multiagent dialogue models, the plans can include the communication necessary for plan execution — that is the detection and communication of information relevant to the execution of the plan. I believe this is the first time that this kind of planning model, drawn from the literature of non-deterministic planning, has been combined with a communication model and then applied to multiagent teams.

I am working to extend the non-deterministic transition model to Markov Decision Processes (MDPs) (Tang and Parsons 2009) to utilize the statistical information. Another area of future work is incorporating prior work on cognitive models of non-monotonic reasoning and dialogues, the argumentation-based approaches, on planning (Tang and Parsons 2005), which assumes a simple, deterministic model of actions, with the work I have described here. Another area of future work, which addresses the main area in which this model falls short of a model of teamwork, is to consider the formation of joint intentions. Here there is a rich vein of work to draw on, for instance (Cohen and Levesque 1990; Grosz and Kraus 1999), and I will seek to incorporate this into the model.

**References**


Tang, Y.; Norman, T. J.; and Parsons, S. 2009b. Towards the implementation of a system for planning team activities. In *Proceedings of the Second Annual Conference of the ITA*.