On Acting Together: Without Communication

Marcus J. Huber **Edmund H. Durfee**

Artificial Intelligence Laboratory
The University of Michigan
1101 Beal Avenue
Ann Arbor, Michigan 48109-2110
{marcush, durfee}@engin.umich.edu

Abstract

It is important, in situations where teams of agents can commit to joint, long-term objectives, that the agents be able to identify when the team objective is no longer important, or is futile. Prior work has typically assumed that agents should believe that joint commitments hold unless explicitly told otherwise by other agents. In this paper, we argue that this assumption expects too much from the communication channels and from agents that initiate the abandonment of the commitment. A more general, robust method for monitoring a joint commitment is to make each agent responsible for acquiring evidence that supports or refutes the contention of the sustained commitment. While this evidence can include explicit messages from others, it also can include observations made by the agent and interpretations of those observations. We thus build from formal notions of commitment to collective goals, and conventions for explicit maintenance of beliefs about commitments, and extend this work to allow observations and the consideration of imperfect communication channels. Our approach has been implemented and tested in the application domain of cooperative robotic reconnaissance in a sometimes hostile environment.

Keywords: Distributed Artificial Intelligence, Cooperation, Coordination, and Conflict

Introduction

Some tasks require team commitment to long-term objectives, followed by ongoing local elaboration of plans to achieve the objectives. For example, unmanned ground vehicles performing reconnaissance tasks can converge on a shared mission plan before crossing into hostile territory and spreading out. As it individually executes its portion of the shared plan, each agent continually selects specific actions to take based on the shared objectives and its ongoing perception of its environment. In the course of execution, however, one or more individuals might recognize that continued pursuit of the objectives is unimportant or futile, while other agents might be unable to recognize this fact. These other agents, who continue to pursue objectives expecting to be supported by a team which is actually no longer is committed to the objectives, could be wasting their time (at best) or endangering themselves and others (at worst). Thus, it is important for an agent that is part of a team to represent its commitment to team objectives and reason about when to abandon that commitment.

Levesque *et al* [Levesque *et al.*, 1990] have developed an explicit representation of joint goals and commitments, along with an implicit (hardwired) response to commitment abandonment. Jennings [Jennings, 1993] has followed up that work with an explicit representation to joint plans and responsibilities, along with explicit (rule-based) responses to commitment abandonment. In both cases, the basic idea is that responsibility for updating team members that a commitment has been abandoned resides with the agent that first drops the commitment. Typically, an agent that is dropping the commitment warns the others that it is doing so by explicitly sending them messages. But there are some problems with this. First, it could be that an agent that must drop a commitment is in no position to send messages, either because it drops the commitment due to being incapacitated (which means that it is likely unable to alert others) or due to being in a "stressed" situation (such as finding itself surrounded by enemy agents) where survival might be its predominant concern and it either cannot or should not contact others. Second, communication in practical systems is uncertain to successfully convey unambiguous information [Halpern and Moses, 1984], so achieving common knowledge of the status of team commitments might be problematic.

Third, it could be obvious to the other agents that the commitment has been dropped solely from observations. For example, consider the team reconnaissance task previously mentioned, where agents perform a "bounding overwatch" by alternatively "bounding" between concealed locations and laying in concealment, "overwatching" agents whose turn it is to move. If, during this maneuver, an overwatching agent sees a bounding agent turn around and head for home, the overwatcher should rightfully question the commitment of the bounder. In this case, explicit message passing might not be necessary, and actually could be detrimental: if the bounder has discovered an enemy position, activating its radio might only serve to draw attention to itself and the team!

This leads us to the view that it is not unreasonable to assign to each agent the job of actively monitoring for itself the commitment of others to the joint goals. In other words, agents are perceiving much about their environments already; with sufficient focus based on the mission plan, making observations about others' commitments can be interwoven with other observations. In practical terms, this means that an agent should be monitoring others to the extent it can, continuously validating that they appear to be continuing their commitment to the joint goal, and noticing when this commitment wavers.

Thus, our work can be seen as a counterpoint to the previous work. We begin with the same notion of commitment to joint persistent goals among members of a team, but rather than considering conventions (or social laws) that dictate the actions that an agent that abandons the commitment must take to be a good team player, we instead consider how an agent that is part of a team should take responsibility for monitoring the commitment into its own hands. Thus, rather than a single, global, convention that all agents must follow for all commitments, the alternative methods for monitoring and reacting to abandoned commitments can be personalized to each agent based upon the commitments that matter to it. Each agent can then reason about whom to observe in order to determine the other agents' commitment, rather than being required to monitor every agent. Our work therefore also permits a certain freedom in that, if communication *is* possible but realistically imperfect, an agent can augment explicit communication-based conventions with observations to increase its confidence in the commitment of others, possibly focusing on what it considers the most critical subset of the team.

What this amounts to, then, is that agents monitoring the commitment need to attempt to interpret what they know of the activities of others in terms of whether those activities support the belief that the joint commitment is still in force. And these activitiescan still include explicit communication actions. This is a variant of the plan recognition problem. Plan recognition in general is hard because the activities of others must be matched against a potentially vast number of plans. However, for this application, the number of plans is tremendously reduced, since the problem is really just to validate that the agents are pursuing plans that are consistent with the joint commitment (mission plan). This makes the problem tractable, and leads to a real implementation.

To realize an implementation of these notions, however, requires several things. First, the basic definitions of joint persistent goals and related concepts must be mapped into an executable representation that supports the ongoing monitoring of the execution context to detect whether a commitment should continue to hold. While this could be done in a number of plan execution systems, in this paper we detail more precisely how it can be supported in UM-PRS [Huber *et al.*, 1994c], our variant of Georgeff *et al*'s PRS [Ingrand *et al.*, 1992]. The representation explicitly captures the context under which commitments should be believed to hold, and thus agents following conventions that cause them to update each other upon commitment abandonment can use these mechanisms. But, as a more general solution, where monitoring commitment to team goals is seen as just one more aspect of its environment that an agent should attend to, we describe how the plan execution representation can be automatically mapped into a representation for plan recognition. With these capabilities, we can then show how an agent team that might not be able to explicitly communicate during (mission) plan execution can still reason about commitments to their mutual benefit.

Joint Commitment

The idea of individual members of a team forming a joint commitment to a team goal is not new. In [Levesque *et al.*, 1990], Levesque, Cohen, and Nunes developed a formal definition of such a joint commitment. In their work, agents who commit to achieving a joint goal commit to either working toward the goal until it is achieved or, if the agent recognizes that the goal has already been achieved or can never be achieved, it will take it upon itself to make sure that the other agents in the team come to know that fact also, so that the entire team can drop the joint commitment.

Some basic definitions of Levesque *et al* are: (BEL x p) indicates that agent x believes proposition p is true; (GOAL x p) indicates agent x has the goal of achieving proposition p; (MB x y p) states that agent x and agent y mutually belief that proposition p is true; (UNTIL p q) means that until proposition p is true, proposition q will remain true, $\Diamond$p indicates that proposition p is true at some point in the future; and $\Box$p indicates that proposition p is true from now on. For complete definitions, see [Levesque *et al.*, 1990].

Below, we present the most important definitions, those of Weak Mutual Goal (WMG), Weak Goal (WG), and Joint Persistent Goal (JPG), respectively.

Definition WMG (Weak Mutual Goal):
(WMG x y p) $\stackrel{\text{def}}{=}$
(MB x y (WG x y p) $\wedge$ (WG x y p))

Definition WG (Weak Goal): (WG x y p) $\stackrel{\text{def}}{=}$
[$\neg$(BEL x p) $\wedge$ (GOAL x $\Diamond$p)] $\vee$
[(BEL x p) $\wedge$ (GOAL x $\Diamond$(MB x y p))] $\vee$
[(BEL x $\Box\neg$p) $\wedge$ (GOAL x $\Diamond$(MB x y $\Box\neg$p))]

Definition JPG (Joint Persistent Goal):
(JPG x y p q) $\stackrel{\text{def}}{=}$
(MB x y $\neg$p) $\wedge$ (MG x y p) $\wedge$
(UNTIL [(MB x y p) $\vee$ (MB x y p $\Box\neg$p) $\vee$ (MB x y $\neg$q)])
(WMG x p))

Basically, what these definitions say is that agents that have a joint persistent goal believe that all of the agents involved think: the goal has not yet been achieved; that they all have the (mutual) goal of achieving the goal; and that they will continue to have that mutual goal until either they all believe that the goal has been accomplished, they all believe that the goal is not accomplishable, or they all believe that some other condition (the q term) is no longer true (for example, in bounding overwatch, pursuing the goal of bounding to the next concealment point only makes sense if the goal of collectively moving forward still makes sense). We will omit the q term from here on, as it is a simple conditional and has no real impact upon later discussion. Furthermore, if any of the agents comes to *individually* believe that the goal has been achieved, cannot be achieved, or the q condition is no longer true, it must make sure that the other agents are made aware of this belief, so that they can all drop the joint persistent goal. This mutual awareness is implicitly acomplished via communication between the involved agents. Later, we show how to extend this to also handle situations where communication is error-prone or difficult.

Plan Embodiment

The first step toward implementation of our approach is to embody the definitions of joint persistent goals in real multi-agent plans. It should be noted that little procedural knowledge is implicit within Levesque *et al*'s definitions, most of it being logical statements concerning the agents' beliefs with respect to the state of progress toward the joint goal. For the sake of comparison, we will look at the case of joint goals for two agents. And, as was mentioned earlier, we will illustrate the implementation in the syntax of UM-PRS [Huber *et al.*, 1994c], an implementation of the Procedural Reasoning System (PRS) [Ingrand *et al.*, 1992]. In this section, we first give a general overview of UM-PRS and then show how some of the JPG definitions can be implemented within UM-PRS.

UM-PRS

UM-PRS supports all of the standard planning constructs such as conditional branching, context, iteration, subgoaling, etc. A World Model holds the facts that represent the current state of the world. A Knowledge Area (KA) defines a procedural method for accomplishing a goal. Its applicability is limited to a particular PURPOSE (or goal) and may be further constrained to a certain CONTEXT. The procedure to follow in order to accomplish the goal is given in the BODY. Variables may be used throughout a KA, and are primarily used in actions in the KA BODY, in the KA CONTEXT, and in a KA's PURPOSE expression. Variables are represented by a text identifier preceded by a $ symbol.

The body of a KA describes the sequence of actions to be taken in order to accomplish the PURPOSE (goal) of the KA. Each action in a KA can specify a goal to ACHIEVE. In addition, a KA action can be a primitive function to EXECUTE directly, an ASSERTion of a fact to the world model, a RETRACTion of a fact from the world model, an UPDATE of a factin the world model, a FACT or a RETRIEVE statement that retrieves relation values from the world model, or an ASSIGN statement that assigns variables the results of run–time computations. Furthermore, iteration and branching are accomplished through WHILE, DO, OR, and AND actions. Comments are preceded by "//" symbols. Refer to [Huber *et al.*, 1994c] for a complete description of the runtime semantics of UM-PRS. We will discuss the semantics that are relevant, as needed.

Joint Persistent Knowledge Areas

We show the KAs that embodies the definition of a Weak Mutual Goal from section *Joint Commitment* in Figure 1. In the contexts of the KAs are relations expressing the belief of the agent's about the joint goal p. They are in the form `FACT <goal> <state> <agent> <value>`, where <goal> is any of the goal propositions involved, <state> is one of "ACTIVE", "achieved", or "achievable" (indicating whether the goal is being pursued, has been accomplished, or is accomplishable, respectively), <agent> is the agent for whom the belief is about, and <value> is either "True" or "False".

To implement weak mutual goals, we define two KAs shown in Figure 1. The first KA is valid when the agent is actively pursuing the goal p and believes that the other agent is either doing the same thing, or is in the midst of informing it of the reason for dropping the joint goal.[1] The simple KA body says that the agent is actively trying to achieve the goal, and then subgoals so that it might start executing the plan for accomplishing the goal.

The second KA models the situation where the agent believes that the goal p has already been achieved or that satisfying the goal is not possible, with the same beliefs for the other agent as in the first KA. The KA body is different in that the agent is no longer actively pursuing goal p, so this is reflected in the world model. The WMG is still valid because of the lack of mutual belief in the invalidating information, so the agent subgoals to ACHIEVE `WM_p` in order to bring about the mutual beliefs.

We show the KAs that embodies the definition of a Weak Mutual Goal in Figure 1. These two KAs handle the three situations of the underlying Weak Goal definition. The first KA is valid when the agent is actively pursuing the goal p and believes that the other agent is

[1] UM-PRS views the context of a KA as a set of conditions that must remain true during the course of execution of the KA. If the context becomes invalid at any point, the KA fails.

```
KA { NAME:  "Weak Mutual Goal to achieve goal p"
PURPOSE: ACHIEVE WMG_p $THIS_AGENT $TEAM_AGENT;
CONTEXT:
    (FACT p "ACHIEVED" $THIS_AGENT "False")                     // (BEL x p)
    (OR (AND (FACT p "ACHIEVED" $TEAM_AGENT "False")           // (NOT (BEL y p))
          (FACT p "ACTIVE" $TEAM_AGENT "True"))                // (GOAL y p)
       (AND (FACT p "ACHIEVED" $TEAM_AGENT "True")             // (BEL y p)
          (FACT MB_achieved_p "ACTIVE" $TEAM_AGENT "True")) // (GOAL y (MB x y p))
       (AND (FACT p "ACHIEVABLE" $TEAM_AGENT "False")          // (BEL y never(p))
          (FACT MB_never_p "ACTIVE" $TEAM_AGENT "True"))));  // (GOAL y (MB x y never(p)))
BODY:
    UPDATE (p "ACTIVE" $THIS_AGENT "True");
    ACHIEVE WG_p $THIS_AGENT $TEAM_AGENT;
}

KA { NAME:  "Weak Mutual Goal p that holds while (NOT (MB x y never(p)))"
PURPOSE: ACHIEVE WMG_p $THIS_AGENT $TEAM_AGENT;
CONTEXT:
    (OR (FACT p "ACHIEVED" $THIS_AGENT "True")                 // (BEL x p)
       (FACT p "ACHIEVABLE" $THIS_AGENT "False"))              // (BEL x never(p))
    (OR (AND (FACT p "ACHIEVED" $TEAM_AGENT "False")           // (NOT (BEL y p))
          (FACT p "ACTIVE" $TEAM_AGENT "True"))                // (GOAL y p)
       (AND (FACT p "ACHIEVED" $TEAM_AGENT "True")             // (BEL y p)
          (FACT MB_achieved_p "ACTIVE" $TEAM_AGENT "True")) // (GOAL y (MB x y p))
       (AND (FACT p "ACHIEVABLE" $TEAM_AGENT "False")          // (BEL y never(p))
          (FACT MB_never_p "ACTIVE" $TEAM_AGENT "True"))));  // (GOAL y (MB x y never(p)))
BODY:
    UPDATE (p "ACTIVE" $THIS_AGENT "False");
    ACHIEVE WG_p $THIS_AGENT $TEAM_AGENT;
}
```

Figure 1: Knowledge Areas for implementing Weak Mutual Goals.

either doing the same thing or else is in the midst of informing it of the reason for dropping the joint goal.[2] The simple KA body says that the agent is actively trying to achieve the goal, and then subgoals so that it might start executing the plan for accomplishing the goal.

The second (bottom) KA in Figure 1 models the situation where the agent believes that the goal p has already been achieved or that satisfying the goal is not possible, with the same beliefs for the other agent as in the first KA. The KA body is different in that the agent is no longer actively pursuing goal p, so this is reflected to the world model. The WMG is still valid, however, because of the lack of mutual belief in the invalidating information, so the agent subgoals in order to bring about the mutual beliefs.

If any of the context terms fails because of a local change in belief, in either of the above KAs, that KA will no longer be valid and will fail. UM-PRS will then transfer execution to the (now valid) alternate KA. If at any point the context fails due to some change in the other agent's beliefs (now known by itself, satisfying mutual belief), the entire joint venture falls apart and the joint goal is no longer held by the agents.

Continuing, to implement the three means of satisfying the Weak Goal definition, we define the three Knowledge Areas shown in Figure 2. The WG_p KAs above represent the commitment that an agent makes to the other agent involved in the joint goal. In the first KA, the agent believes that the goal p has yet to be achieved (and is still achievable) and so subgoals so that it can accomplish p.

The second KA is applicable when the agent believes that the goal p has already been accomplished and so must fulfill its commitment to establish mutual belief in this fact (indicated by the goal to achieve MB_achieved_p, which we leave unspecified).

The third KA is valid in the context where it is no longer possible to accomplish the goal p. When the agent believes this, it fulfills its commitment to inform the other agent of this fact (indicated by the goal to achieve MB_achieved_p, which we also leave unspecified).

It is at this level that Jennings' idea of convention [Jennings, 1993] can come into play. If another convention needs to be specified, one or more additional Knowledge Areas may be written such that, depending upon the failure condition, differing conventions (e.g. Jennings' joint responsibility) will be followed.

Finally, to implement joint persistent goals, we define the Knowledge Area shown in Figure 3. This KA models the JPG definition from section *Joint Commitment* directly, with some simplification of duplicate terms ((MB x y NOT(p)) would appear twice). In the Knowledge Area's context can be seen the terms representing the mutual belief that the goal p has not yet been accomplished and the mutual goal of accomplishing p. The `UNTIL` clause of the JPG definition is modeled in the remainder of the context and in the KA body. Given the semantics of UM-PRS, the first three terms of the `UNTIL` clause must be inverted so that *while* they are all true, the agent will continue to pursue the goal.

If any of the context terms fails because of a local change in belief, that particular KA will no longer be valid and will fail. UM-PRS will then transfer execu-

[2] UM-PRS views the context of a KA as a set of conditions that must remain true during the course of execution of the KA. If the context becomes invalid at any point, the KA fails.

```
KA { NAME:  "Achieve p"
PURPOSE: ACHIEVE WG_p $THIS_AGENT $TEAM_AGENT;
CONTEXT:
     FACT p "ACHIEVED" $THIS_AGENT "False";  // (NOT (BEL x p))
BODY:
     UPDATE (p "ACTIVE" $THIS_AGENT "True");
     ACHIEVE JPG_p $THIS_AGENT $TEAM_AGENT;
}

KA { NAME:  "Achieved p, so inform others for (MB x y p)"
PURPOSE: ACHIEVE WG_p $THIS_AGENT $TEAM_AGENT;
CONTEXT:
     FACT p "ACHIEVED" $THIS_AGENT "True";  // (BEL x p)
BODY:
     UPDATE (MB_achieved_p "ACTIVE" $THIS_AGENT "True");
     ACHIEVE MB_achieved_p $TEAM_AGENT;
}

KA { NAME:  "p unachievable, so inform others for (MB x y never(p))"
PURPOSE: ACHIEVE WG_p $THIS_AGENT $TEAM_AGENT;
CONTEXT:
     FACT p "ACHIEVABLE" $THIS_AGENT "False"; // (BEL x never(p)
BODY:
     UPDATE (MB_never_p "ACTIVE" $THIS_AGENT "True");
     ACHIEVE MB_never_p $TEAM_AGENT;
}
```

Figure 2: Knowledge Areas for implementing Weak Goals.

```
KA { NAME:  "Joint Persistent Goal p"
PURPOSE: ACHIEVE JPG_p $THIS_AGENT $TEAM_AGENT;
CONTEXT:
     FACT p "ACTIVE" $THIS_AGENT "True";        // (MG x y p)
     FACT p "ACTIVE" $TEAM_AGENT "True";

     FACT p "ACHIEVED" $THIS_AGENT "False";     // (MB x y (NOT p))
     FACT p "ACHIEVED" $TEAM_AGENT "False";

     FACT p "ACHIEVABLE" $THIS_AGENT "True";  // (MB x y eventually(p))
     FACT p "ACHIEVABLE" $TEAM_AGENT "True";
BODY:
     ACHIEVE p $THIS_AGENT $TEAM_AGENT;
}
```

Figure 3: Knowledge Areas for implementing Joint Persistent Goals.

tion to the (now valid) alternate KA. If at any point the context fails due to some change in the other agent's beliefs (now known by itself, satisfying mutual belief), the entire joint venture falls apart and the joint goal is no longer held by the agents.

Joint Persistent Goals Through Perception

As mentioned earlier, the Levesque *et al* definitions require communication mechanisms in their procedural bodies so that they can update the other agents in the team of information of global import. The communication actions, not shown in the above KAs, would be found in the KAs responsible for achieving the goals of **p**, **MB_achieved_p**, and **MB_never_p** (the original joint goal, the goal to establish mutual belief that **p** has already been achieved, and the goal to establish mutual belief that **p** can never be achieved, respectively). Our interests lie in coordination when communication is very costly, difficult due to noise, or perhaps not possible at all, when an agent then has to rely upon its own perceptions to perform coordinated activities. In this vein, then, some means is necessary to ascertain the same information normally passed via messages between the agents. Our approach is similar to plan recognition, where the agent makes observations of the other agent's behavior and updates it's model of the other agent's beliefs and motivations based upon these observations.

The plan representation shown above in section *Plan Embodiment* is not conducive to performing plan recognition, however. The second step toward implementation of our approach, then, is to convert our joint goal plan models into a representation conducive to plan recognition—a probabilistic reasoning framework called a belief network.

As a brief overview, a belief network is a directed acyclic graph representing the dependencies among a set of random variables. Each random variable ranges over a domain of outcomes, with a conditional probability distribution specifying the probabilities for each state for given all combinations of outcome values for the predecessors of the random variable in the network. For a more thorough account of belief networks, see, for example, [Pearl, 1988] or [Neapolitan, 1990].

Converting plan models into representations conducive to plan recognition is normally a difficult and time consuming operation. In [Huber *et al.*, 1994a; Huber *et al.*, 1994b] we describe a methodology that we have developed by which plans such as those shown above can automatically be converted into belief networks. These procedures handle a broad class of plans, including those with sequential actions, conditional branching, subgoaling, and iteration. In general, the procedures take plan constructs and build portions of belief networks that model those constructs. See [Huber *et al.*, 1994a; Huber *et al.*, 1994b] for a full description of the transformation methods.

Experiments

To this point, we have discussed the general issues involved with extending the communication-based Joint Persistent Goal definition of Levesque *et al* to be less reliant on communication. We will now demonstrate how an agent can monitor the commitments of others purely through observation, and then we discuss how our work can applies to noisy or error-prone explicit communication. The domain of our example is that of military reconnaisance, where two (or more) cooperative agents are engaged in performing a bounding overwatch. In Figure 4 are four KAs relevant to the bounding portion of the bounding overwatch task.

These four KAs are added to those of the definition of Joint Persistent Goal and, using the transformation methods of [Huber *et al.*, 1994a; Huber *et al.*, 1994b], our system maps this collection of KAs into a belief network, starting with the competing top-level goals of the joint goal ACHIEVE **WG_performed_bound** (i.e. in the JPG KAs in section *Plan Embodiment*, replace **p** with **performed_bound**), and the individual goal of ACHIEVE **enemy_dealt_with**.

Integrated with the Joint Persistent Goal model, the final belief network for performing the "bounding" and "dealing with the enemy" tasks is shown in Figures 5-8. In these figures, the highlighted nodes with the scissor icon above them indicates that the nodes are fully detailed in a belief network in a later figure. We have shown in previous work [Huber *et al.*, 1994a] that the belief network for the four KAs above permits useful queries to be made concerning the beliefs of the observed agent (e.g. "Is the other agent performing a bound?", "Is there an enemy in the vicinity?", etc.) In general, the behavior of the belief network in Figures 5-8 should be such that, if the observed agent is doing as it is supposed (in the team sense) and is performing the bounding portion of the overall task, observations will support the belief in the upper elements in the belief network, namely **JPG_performed_bound**, **WMG_performed_bound**, and finally, to WG_performed_bound. If however, the observing agent notices the other agent moving into a growth of foliage (hiding is one of the means of dealing with an enemy agent appearing), confidence in the other agent maintaining its commitment to the bounding overwatch goal should drop.

The observing agent can use these changes during its reasoning and can make decisions based on whether it thinks the other agents are still committed to the joint persistent goal. If it becomes probable that the joint goal has been abandoned by an agent, a convention of some sort may then dictate what the observing agent must do in order to fulfill its obligations to other team agents before it, too, abandons the joint commitment.

Network Node	Probability distribution
ACHIEVE **WMG_performed_bound**	0.20 0.80 0.20
ACHIEVE **WG_performed_bound**	0.37 0.21 0.42
ACHIEVE **JPG_performed_bound**	0.37 0.21 0.42
ACHIEVE **performed_bound**	0.36 0.23 0.41
ACHIEVE **enemy_dealt_with**	0.67 0.17 0.16

Table 1: Prior probabilities for final, integrated belief network.

Network Node	Probability distribution
ACHIEVE **WMG_performed_bound**	0.41 0.43 0.10
ACHIEVE **enemy_dealt_with**	0.16 0.20 0.64

Table 2: Posterior probabilities given that the observed agent is hiding.

We performed several tests on the belief network to verify that it exhibits the desired behavior. In one test, we checked to see if an observing agent could determine when a commitment was broken. For comparison, the prior probability distributions for pertinent random variables in the belief network are shown in Table 1. For this test, we provided evidence that two actions, EXECUTE **find_concealing_foliage** and EXECUTE **move_into_foliage**, had been performed. These actions are part of the KA for dealing with an enemy agent in the vicinity (the agent tries to hide). Given these two observations, the posterior distributions for several pertinent nodes are shown in Table 2.

Comparison between the two tables show a significant shift in belief that the observed agent has abandoned its joint commitment and is, instead, pursuing an alternative goal, that of hiding from an enemy agent. The observing agent can also use the belief network to determine the probability that an enemy agent is in the vicinity, even if it has not seen an enemy agent itself, by querying the random variable **enemy_in_vicinity**, a "True"/"False" variable. The posteriors for this node are 0.05 for "False", and 0.95 for "True". With the knowledge that the observed agent has abandoned its commitment to performing the joint task, and the knowledge that it is likely that there is an enemy agent nearby, the observing agent is much better informed than if it had not been using our scheme.

A second test that we performed verifies that the network behaves correctly when the observed agent is continuing to work toward the joint goal. In this test, we provided evidence that the agent was observed "bounding" to its next observation position (i.e. the actions of EXECUTE **determine_next_viapt**, EXECUTE **navigate_to_next_viapt**, EXECUTE **find_cover**, EXECUTE **navigate_to_cover**, and EXECUTE **move_into_cover** were performed). With

```
KA { NAME:  "perform bound"
PURPOSE: ACHIEVE bound_performed $THIS $TEAM;
CONTEXT:
BODY:
     ACHIEVE moved_to_next_viapt;
     EXECUTE find_cover $X $Y;
     EXECUTE navigate_to_cover $X $Y;
     EXECUTE move_into_cover;
}

KA { NAME:  "move to the next via point"
PURPOSE: ACHIEVE moved_to_next_viapt;
CONTEXT:
BODY:
     EXECUTE determine_next_viapt $X $Y;
     EXECUTE move_to_next_viapt $X $Y;
}

KA { NAME:  "hide"
PURPOSE: ACHIEVE enemy_dealt_with;
CONTEXT:
FACT enemy_in_vicinity "True";
BODY:
    OR
     {     EXECUTE find_concealing_foliage $X $Y;
           EXECUTE move_into_foliage $X $Y; }
     {     EXECUTE find_concealing_object $X $Y;
           EXECUTE move_behind_object $X $Y; }
}

KA { NAME:  "attack"
PURPOSE: ACHIEVE enemy_deal_with;
CONTEXT:
FACT enemy_in_vicinity "True";
BODY:
    EXECUTE move_into_range;
    EXECUTE aim;
    EXECUTE fire;
}
```

Figure 4: Knowledge Areas for implementing Bounding.

Network Node	Probability distribution
ACHIEVE WMG_performed_bound	0.06 0.46 0.48
ACHIEVE WG_performed_bound	0.03 0.18 0.79
ACHIEVE JPG_performed_bound	0.02 0.11 0.87
ACHIEVE performed_bound	0.00 0.10 0.90
ACHIEVE enemy_dealt_with	0.80 0.10 0.00

Table 3: Posterior probabilities given that the observed agent is bounding.

this evidence, the belief network node distributions are as shown in Table 3.

Again, comparing the posteriors to the priors, we see a significant change in the beliefs in the mutual goal of performing the bounding task. This time, however, they are in support of the observed agent being committed to the joint goal. The agent's belief that the other agent has dropped the joint goal has decreased to less than one-third of the prior and there is still strong belief that it is still actively pursuing the goal. Meanwhile, the belief that the observed agent is actively pursuing its individual goal, ACHIEVE **enemy_dealt_with**, has dropped to a fraction of the prior. The observing agent, then, would continue to have confidence that the observed agent is still working toward fulfilling its role in the team commitment.

An important additional test was to verify that our approach still handles the situation when explicit communication is used. Explicit messages conveying information modeled in the belief network is easily added as evidence. For example, if another agent informs the observing agent that the goal of performing the bound is impossible to accomplish, the random variables modeling this contextual information (MB x y $\Box\neg$p) can be set and this evidence propagated through the network. The posterior distributions for the pertinent nodes are shown in Table 4.

The belief that the observed agent has dropped its commitment to the joint goal is overwhelming. And, correctly, the system has shifted belief that the Weak

Network Node	Probability distribution
ACHIEVE WMG_performed_bound	0.53 0.34 0.13
ACHIEVE WG_performed_bound	0.79 0.10 0.11
ACHIEVE JPG_performed_bound	0.99 0.01 0.00
ACHIEVE enemy_dealt_with	0.42 0.29 0.29

Table 4: Posterior probabilities after an explicit message from the observed agent.

Mutual Goal is no longer being pursued as mutual belief has been established and the convention of Levesque *et al* has been satisfied. Of course, however, if other (enemy) agents in the environment might attempt to disrupt the team by injecting such messages into the system, then the received message might not, by itself, provide overwhelming evidence for abandonment. The belief network allows a probabilistic combination of what an agent sees as well as what it hears to lead it to the most likely conclusion.

Conclusion and Summary

We have generalized the notion of "conventions" for managing changes in commitment to permit individual agents to take responsibility for monitoring the commitments of others on whom they depend. This monitoring can involve observing the others, communicating with the others, or a combination of the two. As a result, our methods allow agents to maintain reasonable models of collective commitment in realistic situations where perfect communication is seldom possible.

In our discussion, we have shown how joint persistent goals can be implemented in concrete, multi-agent plans. We also discussed the extensions necessary for agents to be able to use these concepts when unable to communicate, and introduced the idea of using plan recognition to accomplish the information gathering portion of this paradigm. We also showed how the concrete plans can be manipulated into a representation amenable to the plan recognition process. Finally, we demonstrated that a multi-agent system built using the mechanisms described in this paper would work as intended. When

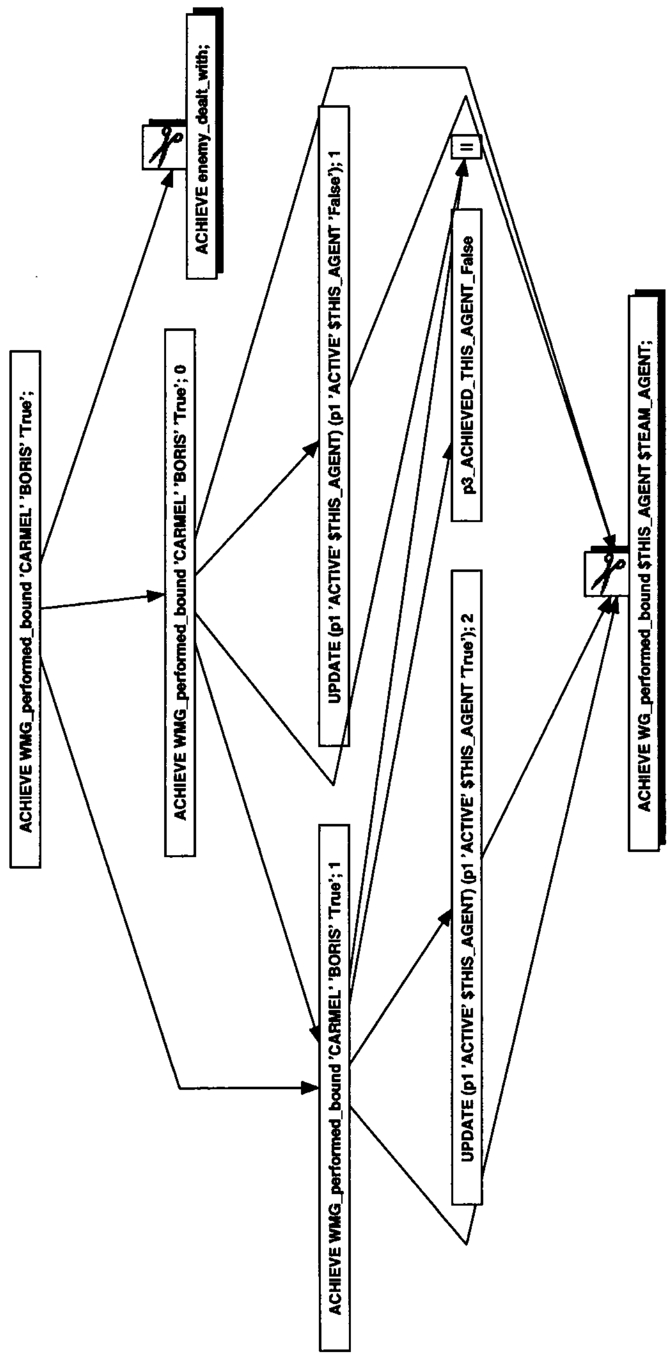

Figure 5: Top level of the belief network for the joint persistent goal and private goal of dealing with an enemy in the vicinity.

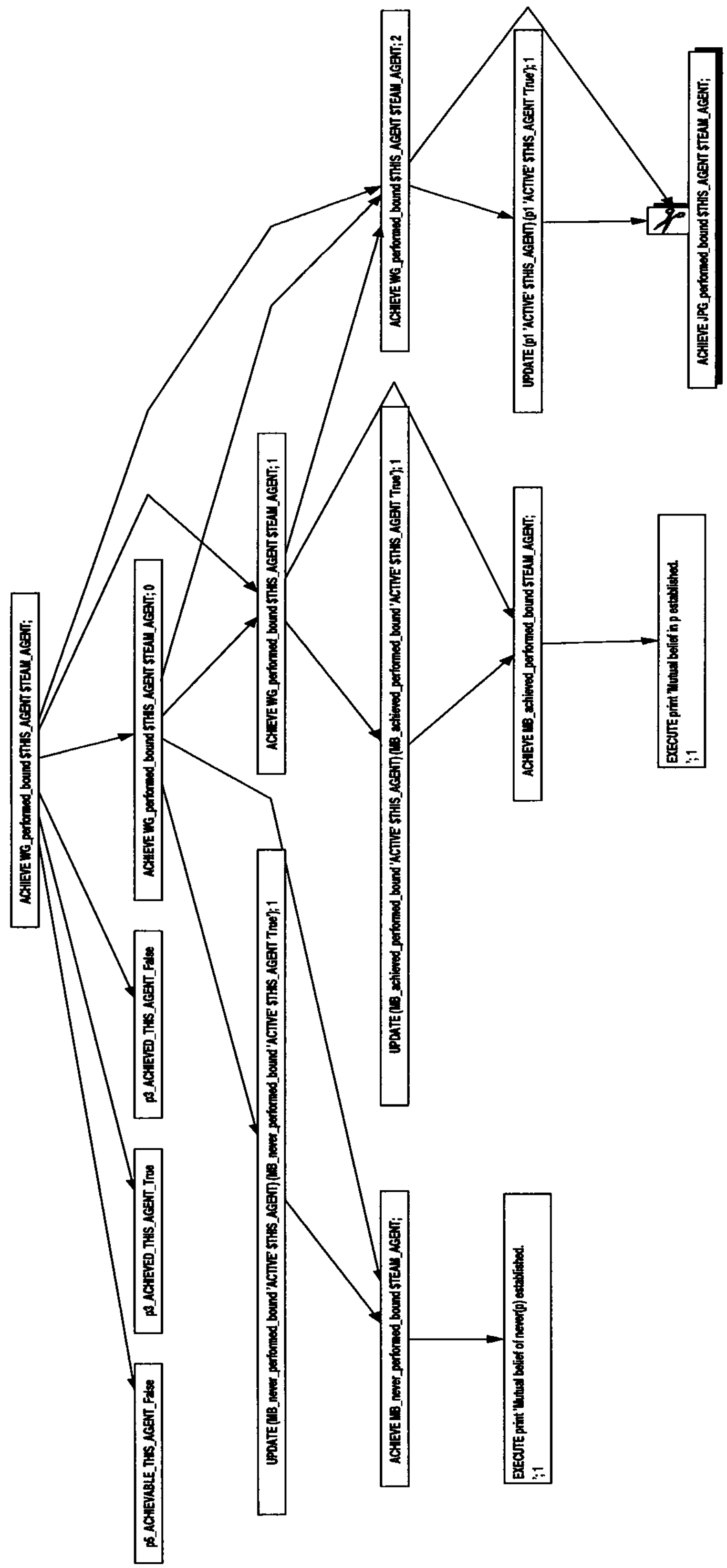

Figure 6: Belief network for WG_performed_bound.

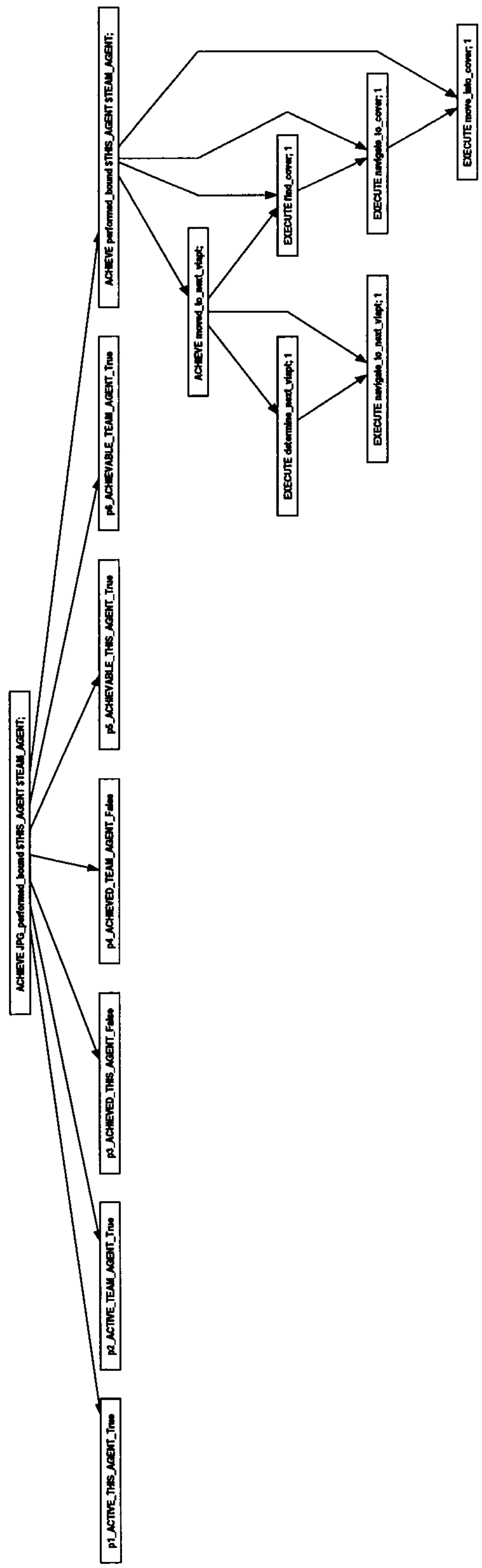

Figure 7: Belief network for JPG_performed_bound.

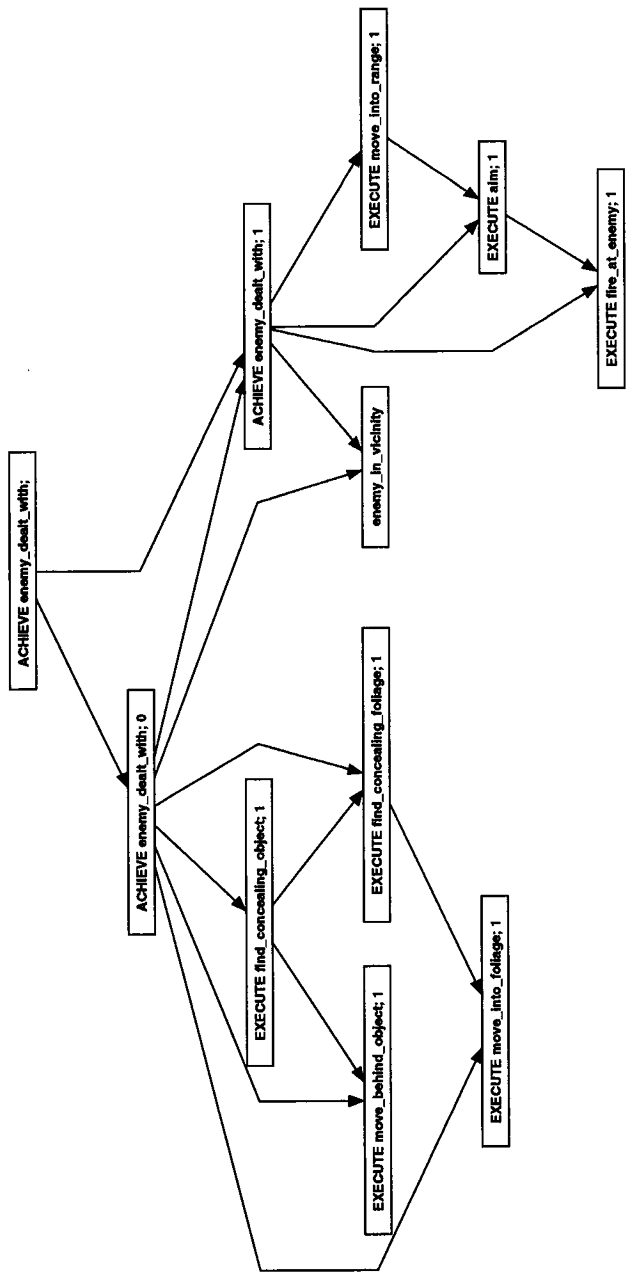

Figure 8: Belief network for enemy_dealt_with.

all agents are working to fulfill their commitments to the joint goal, the agents recognize the continued commitment; when the joint goal is abandoned by any of the agents, the other agents can detect this without any communication and can decide how best to proceed.

The research in this paper illustrates that teams of agents can be coordinated while utilizing the theory of joint persistent goals in domains where communication is impossible or unreliable. And we have shown that perception and inferencing is a viable means of acquiring the information that would normally be acquired through communication. We feel that these are important steps toward development of a more robust theory of multi-agent coordination, where groups of agents must be able to effectively accomplish a wide variety of tasks in a wide range of environmental conditions.

References

[Halpern and Moses, 1984] Halpern,Joseph Y. and Moses,Yoram . Knowledge and common knowledge in a distributed environment. In *Third ACM Conference on Principles of Distributed Computing*, 1984.

[Huber *et al.*, 1994a] Huber,Marcus J. , Durfee,Edmund H. , and Wellman,Michael P. . The automated mapping of plans for plan recognition. In *Workshop on Distributed Artificial Intelligence*, pages 137–152, Lake Quinalt, WA, July 1994.

[Huber *et al.*, 1994b] Huber,Marcus J. , Durfee,Edmund H. , and Wellman,Michael P. . The automated mapping of plans for plan recognition. In *Proceedings of the Tenth Conference on Uncertainty in Artificial Intelligence*, pages 344–351, Seattle, Washington, July 1994.

[Huber *et al.*, 1994c] Huber,Marcus J. , Lee,Jaeho , Kenny,Patrick , and Durfee,Edmund H. . *UM-PRS V2.7 Programmer and User Guide.* The University of Michigan, 1101 Beal Avenue, Ann Arbor MI 48109, Oct 1994.

[Ingrand *et al.*, 1992] Ingrand,Francois , Georgeff,Michael , and Rao,Anand . An architecture for real-time reasoning and system control. *IEEE Expert*, 7(6):34–44, December 1992.

[Jennings, 1993] Jennings,Nick R. . Commitments and conventions: The foundation of coordination in multi-agent systems. *Knowledge Engineering Review*, 8(3):223–250, 1993.

[Levesque *et al.*, 1990] Levesque,Hector J. , Cohen,Philip R. , and Nunes,Jose H. T. . On acting together. In *Proceedings of the National Conference on Artificial Intelligence*, pages 94–99, July 1990.

[Neapolitan, 1990] Neapolitan,Richard E. . *Probabilistic Reasoning in Expert Systems.* John Wiley and Sons, 1990.

[Pearl, 1988] Pearl,Judea . *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference.* Morgan Kaufmann, San Mateo, CA, 1988.