Enabling Supportive Communications in Decentralized Multi-Agent Teams

by

Keren Gu

B.S., Massachusetts Institute of Technology, 2014

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Master of Engineering in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2015

Author . . . .
Department of Electrical Engineering and Computer Science
May 22, 2015

Certified by . . . .
Julie A. Shah
Assistant Professor of Aeronautics and Astronautics
Thesis Supervisor

Accepted by . . . .
Albert R. Meyer
Chair, Masters of Engineering Thesis Committee

Enabling Supportive Communications in Decentralized Multi-Agent Teams

by

Keren Gu

Submitted to the Department of Electrical Engineering and Computer Science
on May 22, 2015, in partial fulfillment of the
requirements for the degree of
Master of Engineering in Electrical Engineering and Computer Science

Abstract

Supportive communication is an effective collaboration behavior identified in human teams in which team members share information proactively to improve overall team performance. Prior work formulated this objective as the Single-Agent in a Team Decision Problem (SAT-DP) where agents decide whether or not to communicate an unexpected observation during execution time. We extend the SAT-DP definition to include sequential observations, highlighting the need for belief updates of attributed mental models of agents. These updates must be performed effectively and efficiently to minimize model divergence and maximize the utility of future communications. In this paper, we present a decision-theoretic solution to the sequential SAT-DP. In our solution, we propose the use of Bayesian plan recognition as one of the methods for reducing divergence in mental models. To achieve computational tractability, we use probabilistic ordered AND/OR trees to compactly represent distributions over possible solutions of hierarchical planning problems. Finally, we evaluate and demonstrate the effectiveness of our proposed approach on decentralized agents collaborating in partially observable environments.

Thesis Supervisor: Julie A. Shah
Title: Assistant Professor of Aeronautics and Astronautics

Acknowledgments

I would like to first offer my deepest gratitude for Professor Julie Shah who has been the most supportive supervisor. I thank her for giving me the opportunity to work in the Interactive Robotics Group (IRG), exposing me to the fascinating field of robotics and artificial intelligence, and guiding me through the labyrinth of first year graduate research with her technical expertise and endless positive energy. She has been a true inspiration.

I have encountered many mentors along the way. I am grateful to Stefanos Nikolaidis for first inviting me into the IRG family two years ago and mentoring me through my first research project, and Been Kim for seeding the idea that made this research possible and for being a role model on Sundays.

My one year of graduate school would not have been the same without my wonderful lab mates. They are some of the funniest and most patient people. I am grateful for all their technical advice, and more importantly, deep tangential discussions about the universe, life, and such. I also thank them for all the lunch invitations. Even though I declined most of them to work on this thesis, I really appreciated the gesture on a daily basis.

I would like to thank my friends from undergraduate for still caring about my research and my happiness despite having settled far far away, all around the world. While in Cambridge, I am grateful for the new friends that I've made. Without them, work-life *balance* would only be an abstract concept. I also want to thank my second family in Senior House for their unconditional love and for keeping me sane.

Finally, I thank mom and dad for shaping me into who I am today, for all the sacrifices that they made to give me the privilege of pursuing my dream, and for tolerating my lack of phone calls as I worked on my thesis.

Contents

List of Figures

Chapter 1

Introduction

Autonomous agents collaborating in nondeterministic worlds need to react to unexpected changes in the environment and communicate new information effectively. Decision theoretic approaches to multi-agent communication, such as incorporating communication as actions, produce effective results but often do not scale to large problem instances [12, 22]. On the other hand, scalable communication models that use explicit rule-based communication mechanisms are often not flexible to changing environments, becoming less effective [17, 21].

Human-inspired approaches toward multi-agent communication and coordination aim to computationally model effective behaviors found in human teams, such as anticipatory behaviors [16], proactive information sharing [9, 8, 26, 2], and maintaining shared mental models [24, 27, 6].

This paper presents a decision-theoretic approach towards communication. Specifically, this paper defines and addresses the Sequential Single Agent in a Team Decision Problem (sequential SAT-DP). Sequential SAT-DP is an extension of the SAT-DP problem proposed in [2], which asks whether an agent, with incomplete world knowledge, should communicate a new piece of unanticipated information during execution time in a collaborative setting. The sequential extension of the problem addresses, in addition to whether or not to communicate an observation, how agents should update their beliefs of each other as a result of communication in order to maximize the effectiveness of future communications. By addressing the sequential SAT-DP problem, we enable effective information sharing.

As demonstrated in [2], uncertainties in understanding of a teammate's possible plans are detrimental to the effectiveness of communication. Therefore, in the problem we wish to address, it is critical to maintain an accurate representation of teammates' possible plans. Our solution performs belief updates to maintain accurate mental models of other agents via Bayesian plan recognition similar to that of [24].

Our solution addresses scalability and computational tractability by assuming hierarchical structure in the planning domain. Hierarchical structure have been utilized to exponentially reduce search space in symbolic planning [23] and in solving dec-POMDP with macro-actions [1]. We present a structure to compactly represent a distribution of possible plans that one agent attributes to another.

Previous solutions to the SAT-D problem proposed by [2] and [16] were evaluated on, or assumed, situations where agents have full observations of their teammates. This assumption is often unrealistic in the real world. Applications of multi-agent tasks such as search and rescue operations in disaster zones or surveying of unfamiliar terrains often involve long-range physical separations where communications are expensive, due to either energy constraints of embedded platforms, limited range of wireless transmitters, or security risks of potential interception of messages in hostile territory. We evaluated our proposed solution using a more realistic simulation environment where agents are fully decentralized.

This paper makes the following contributions. First, it formally defines sequential SAT-D problem. Second, it proposes a solution to the sequential SAT-DP by presenting 1) a structure for representing a distribution over possible solutions to an HTN planning problem via a probabilistic ordered AND/OR tree, and 2) methods of belief updates during communication in order to maximize the effectiveness of future communication. Finally, it demonstrates the effectiveness of the proposed approach.

Chapter 2

Problem Definition

This paper addresses a sequential extension to the Single Agent in a Team Decision Problem (SAT-DP) proposed in [2] for fully decentralized partially observable teams, assuming the SharedPlans specification for collaboration [14].

Consider a multi-agent team collaborating in a nondeterministic partially observable environment with discrete time steps where team planning is centralized at $t = 0$ but fully distributed during execution and communication is expensive. Individual agents must reason about whether to communicate newly observed information about the world to other team members in order to optimize overall team utility.

In our problem formulation below, we make basic assumptions that agents share common domain knowledge and identical initial world beliefs. We also assume that agents know their own plans and have partial knowledge of teammates' plans at $t > 0$.

Formally, let $A = \{a_i\}$ be a finite set of agents and Ω be the set of possible observations, which need not be finite. We define the sequential SAT-DP problem with respect to an arbitrary agent a_i with the tuple $(a_i, A_{-i}, b_i^t, V_i^t, \omega_i^t, \phi, c)$ where

- $A_{-i} = A \backslash \{a_i\}$ is the set of other agents in the team,
- b_i^t is a_i's belief of the SharedPlan of the team at time t,
- C_i is the cost function that a_i attributes the world,

- $\omega_i^t \in \Omega$ is a new observation obtained by a_i at time t, $\omega_i^t = \emptyset$ if no new observation at time t is made,

- ϕ is a function that, given a belief b_i^t and an observation $\omega \in \Omega$, produces an updated belief b_i^{t+1}.

- and $c \in \mathbb{R}$ is the cost of communication,

The sequential SAT-D is the problem of determining whether a_i should communicate ω_i^t to agents in A_{-i} at each time step. The sequential extension to the decision problem is necessary because how agents update their beliefs in response to sending and receiving a communication affects future decision making. In addition, this extension allows for time-dependent reasonings.

Chapter 3

Related Work

This chapter discusses a range of prior approaches to reasoning about communications, beginning with classical axiomatic and decision theoretic approaches for multi-agent planning to human-team-inspired approaches for effective communication, such as proactive information sharing and maintenance of shared mental models.

3.1 Multi-Agent Communication

In multi-agent planning, there have been two main approaches towards stipulating communications for collaboration, axiomatic and decision theoretic.

To axiomatically enable communication in team collaborations, formal semantics have been introduced to represent joint intentions over possible-worlds [4] and intentions involving cooperations [14]. Early works stipulated communications whenever certain states are reached, such as discovering infeasibility of a goal [5]. In the STEAM (a Shell for TEAMwork) multi-agent framework, a domain specific decision tree was used to reason about communicating facts that lead to the termination of joint intentions in the team [25]. Extensions of the formal semantics provided a theoretical framework for group communications [17] and for enabling proactive communication [9].

In decision theoretic planning, a common approach to enable communication is to incorporate communication into the agents' action space, as done in the COMmunicative Multiagent Team Decision Problem (COM-MTDP) model [22], and in [12] by augmenting decentralized partially observable Markov decision processes with explicit language of communication, producing the DEC-POMDP-COM model. Recent work introduced macro-actions that encapsulates domain specific hierarchical structures to address intractability issues in solving Dec-POMDP for larger problems [1].

Our proposed work is distinguished from previous work in that, unlike aforementioned axiomatic approaches, our framework is flexible to uncertainties in the world and takes into account the cost and benefit of communication. Previous decision theoretic approaches are computationally intensive and can only solve problems with small state-spaces. While [1] addresses this problem with macro-actions, incorporating communication of new observations as actions is nontrivial. Specifically, the number of actions to incorporate would be proportional to the observation space, which exponentially increases the size of Dec-POMDP. To leverage the macro-actions framework, one must manually partition the set of possible communication actions into macro-actions.

3.2 Communication as Anticipatory Behavior

Anticipatory behaviors have been identified as a key characteristic in effective human teams. A number of previous works have aimed to model communication as a form of anticipatory behavior [8].

[16] introduced a formalized model for reasoning about the cost and benefit of supportive actions, including communications, during the collaboration. Agents in [16] computed the expected value of communicating versus not communicating using a domain specific probability recipe tree (PRT) and acted to minimize expected cost.

[2] proposed a hybrid Belief-Desire-Intention (BDI) and decision-theoretic approach that integrated PRT with an MDP formulation to decide, not only whether or not to communicate but also the best future time to communicate assuming full observation of the teammate's actions.

Both approaches, however, were evaluated using an omniscient agent with full observability of the world and its teammate actions. In our work, we evaluate our model with fully decentralized teams operating under partial observability of the world and teammate actions. Agents need to not only reason over uncertainties in their teammates' plans, but also over other teammates' states. In addition, our work extends the PRT representation to HTN domains and extends the SAT-DP to the sequential decision problem in order to highlight the importance of maintaining accurate mental models over time.

3.3 Reasoning with Mental Models

Maintaining a shared mental model (SMM) has been shown across disciplines to have positive correlation with team performance and to provide explanations for effective human collaboration behaviors [6]. Mental models are internal representations of a situation [26], and shared mental models are extensions of individual mental models into a team context [26]. Applications of mental models range widely from developing better training methods for human teams [6], to better collaborations in mixed human-robot tasks [18, 27, 24], to formalizing and incorporating SMMs into multi-agent teams [26].

In mixed human-robot teams, building and maintaining mental models of other team members is a core component in modeling communication. Hand-crafted communication models and linguistic devices have been introduced to induce and shape human's mental model of the robot [18, 3] thereby enhancing human situational awareness when working with the robot. Capability modeling is introduced to learn domain information probabilistically in order to improve robots' mental models of humans [27].

Recent work in human-robot interaction (HRI) formally defines mental models that robots attribute to human teammates and applies these models to avoid resource conflict. These robots are able to reason over multiple possible human plans and partial goal specifications by applying Bayesian plan recognition techniques to observations of human actions [24]. In contrast from [24], our proposed approach enables reasoning about proactive communication as a means to influence another agent's plan in addition to reactively processing received communications.

In maintaining SMMs in multi-agent teams, the Collaborative Agents for Simulating Teamwork (CAST) framework has incorporated SMMs into team planning representing the SMM as the set of shared knowledge across all members, which is then used by each agent during decision making [26]. In this paper, we do not aim to maintain a single shared mental model, but fully distributed mental models that one agent attributes to another in order to reduce communication overhead needed to maintain an SMM.

Chapter 4

Approach

In this section, we present an approach for solving the subclass of the sequential SAT-D problem where the planning domain is defined with hierarchical structure. We address this subset of problems because, aside from being widely implemented in industry [11], domain specific hierarchical structures reduce the search space of possible plans, and more importantly, allow tractable representation of possible plans.

In decentralized collaboration, we induce supportive communications by first giving agents the ability to attribute mental models, which represent one agent's belief of another agent's possible goals and plans, to other team members. When an observation is obtained, each agent reasons about the expected cost or utility of communicating the information to its teammates and acts to maximize utility or minimize cost.

Communicating useful information relies on having sufficiently accurate mental models of teammates. Previous work has demonstrated that inaccurate beliefs lead to over- or under-communicating [2]. Partial observability and limited communication inevitably lead to diverging mental models, which is when an agent's attributed mental model of another agent is no longer accurate, resulting in ineffective information sharing. To address this problem, we propose three methods to reduce possible divergence of mental models without triggering any additional coordination, one of which applies Bayesian Plan Recognition on the received communication.

4.1 Hierarchical Task Networks

In symbolic planning, a hierarchical task network (HTN) is a planning representation that utilizes domain specific hierarchical knowledge to reduce the search space of possible plans, thereby exponentially reducing planning complexity [7, 11]. HTN domain definition augments classical planning by distinguishing between primitive tasks (executable actions) and compound tasks. Compound tasks can be decomposed into a set of simpler tasks which themselves may be primitive or compound.

Formally, let Q be a finite site of predicates, $S = 2^Q$ be the set of possible states where each state is a set of ground predicates, and TN be the set of possible task networks. We define a task network $tn \in TN$ as a pair (T, ψ) where T is a finite set of tasks, primitive or compound, and ψ is a set of constraints. An HTN planning problem $\mathcal{P}$ is the tuple (Q, O, M, tn_0, s_0) where Q is a finite set of predicates, O is the set of primitive tasks defined over preconditions and effects, M the set of methods representing compound tasks and their decompositions, tn_0 the initial task network representing agents goals, and s_0 the initial state. A method $m \in M$ is a pair $(c(m), pre(m), tn(m))$ where $c(m) \in T$ is the compound task, $pre(m)$ specifies the preconditions for decomposing m, and $tn(m)$ is the decomposed task network of m.

We consider domains where compound tasks may have different decompositions depending on the state of the world, or multiple decompositions. Thus we define a decomposition function that maps a method and a state to a set of possible decompositions:

$$d : M \times S \rightarrow 2^{TN}$$

We also incorporate cost into HTN planning by assigning a cost function $C : O \times S \rightarrow \mathbb{R}$ in order to evaluate our communication models.

An HTN solver, $HOP(Q, O, M, tn_0, s_0)$, outputs a plan, or a sequence of operators, $P = (o_1, \cdots, o_n)$, that completes tn_0 subject to the constraints, and an optimal HTN solver outputs a plan that satisfies the constraints with minimum cost. Given that our formulation allows multiple possible optimal solutions and decompositions of a compound task, we define a comprehensive HTN solver to output a set of all possible plans with the minimum cost.

4.2 Representing Possible Plans

In this section, we propose a probabilistic ordered AND/OR tree, hereafter **Π**-tree, to compactly represent a distribution of possible solutions to an HTN problem. This representation is an extension of the Probabilistic Recipe Tree (PRT) representation from [16]. PRT uses the AND/OR tree structure to define a distribution of possible recipes for completing the action associated with the root node by assigning edge probabilities to OR nodes and their children. Similar structures motivated by AND/OR trees have been previously used in plan recognition and task-planning [15, 13]. The strength of this representation is that it is exponentially more compact than exhaustively representing all possibly recipes or plans [16].

We extend this representation to describe solutions to HTN planning problems. The main additions are enforcing ordering on the children of AND-nodes to represent constraints in the task-networks and consideration for state dependencies. Method decompositions in HTN vary depending on the input state. As a result, one of the key differences is that the **Π**-tree produced for each HTN planning problem could be different depending on the initial world state s_0.

Analogous to PRT, the **Π**-tree defines a distribution of possible plans for completing the HTN problem associated with the root node. Each node has an associated task (or task-network) and a start state. The root node, for example, is defined with tn_0 and s_0. Each compound task $m \in M$ is represented by an OR node where the children are possible decompositions. Given an OR-node for method m with state s, the children of the node is the set of decompositions given by $d(m, s)$, where agents may nondeterministically choose a decomposition during planning. We specify the edge probabilities to each child as the likelihood that an agent chooses the child. We assume that agents act rationally and assign optimal decompositions with uniform probability. Each task network $tn = (T, \phi) \in TN$ is represented by an AND node where each subtask $t \in T$ is a child node. The children of AND-nodes are ordered to respect the constraint ϕ in the task-network. Finally, the leafs of the tree are primitive tasks.

We express the distribution over all possible plans for an HTN problem $\mathcal{P} = (Q, O, M, tn_0, s_0)$ as $\mathbf{\Pi}(tn_0, s_0)$. Note that each subtree in the AND/OR tree structure also represents a distribution. To sample a particular plan from the distribution, we traverse the tree and eliminate every OR node by sampling one child with respect to the edge probability. The leafs of the remaining tree after removing all OR nodes represent one possible plan that achieves the root-task. The likelihood of a given plan is the product over the likelihood of each chosen decomposition, which is also the product of all selected edge probabilities.

The cost of a sampled plan is the sum over all primitive actions with respect to a given cost function. To compute the expected cost over a distribution of plans, we first evaluate the cost of each leaf-node, and then propagate the cost upwards towards the root. AND-nodes propagate the sum over the expected cost of its children and OR-nodes propagate the weighted sum of its children according to each edge probability. The expected cost of the distribution of plans outputted by a comprehensive HOP planner is the cost of any sampled plan from the distribution.

4.3 Teammate Mental Model

We provide each agent the ability to attribute a set of beliefs, desires and intentions to each of its teammates. We define the mental model that $a_i \in A$ attributes to $a_j \in A_{-i}$ with the tuple $M_{i\to j} = (S_{i\to j}, G_{i\to j}, P_{i\to j})$ where

- $S_{i\to j} \subset S$ represents a distribution of possible beliefs of the world state that a_j attributes to a_i;
- $G_{i\to j} \subset TN$ represents a distribution of possible goals that a_i attributes to a_j;
- $P_{i\to j}$ represents a distribution of possible plans as a result of $S_{i\to j}$, $G_{i\to j}$, and a_i's HTN planner.

At $t = 0$, each agent is given, explicitly, identical beliefs of the world, $S^0_{i\to j} = s_0$ for all a_i and a_j, and individual goals of each teammate, $G^0_{i\to j} = G^0_{j\to j}$. For simplicity, we assume that agents do not change their goals during execution. With an HTN solver that outputs a $\mathbf{\Pi}$-tree, each agent attributes the plan distribution of its teammate as

$$P_{i\to j} = \bigcup_{tn_0 \in G_{i\to j}} \bigcup_{s \in S_{i\to j}} \mathbf{\Pi}(tn_0, s) \tag{4.1}$$

, where the likelihood of each plan is

$$\mathbb{P}_{P_{i\to j}}(p) = \mathbb{P}_{\mathbf{\Pi}(tn_0,s)}[p] \cdot \mathbb{P}_{S_{i\to j}}[s] \cdot \mathbb{P}_{G_{i\to j}}[tn_0]. \tag{4.2}$$

For $t > 0$, an agent may update its attributed mental model of its teammate with observation ω^t_i as follows:

- $G^{t+1}_{i\to j} = G^t_{i\to j}$ as stated in our assumption,
- $S^{t+1}_{i\to j} = \{s_j + \omega^t_i\}$ for all $s_j \in S^t_{i\to j}$,
- and $P^{t+1}_{i\to j}$ can be obtained via (4.1) and (4.2).

Recomputing $P^{t+1}_{i\to j}$ for every observation requires many queries to an HTN solver which may be time consuming. In practice, we can improve performance with an update operator defined over $\mathbf{\Pi}(tn_0, s_0)$ and an input observation ω: the operator first evaluates and updates the cost of each leaf node with respect to a new state, which is computed by incorporating the input observation ω with the previous state s_0. Infinite cost is assigned if the precondition of a primitive action node is violated. Then the new costs are propagated upward to the root node and edge probabilities are normalized according to the new cost.

We augment b^t_i in the sequential SAT-DP with $\{M^t_{i\to j} : a_j \in A\}$ to explicitly represent a_i's knowledge of its own plan and the uncertainty in a_i's beliefs of other teammates' plans. Given an observation, we augment the belief update function ϕ to update $M_{i\to j}$ as described above.

4.4 Supportive Communication

In this subsection, we address SAT-DP at a given time interval t where a_i reasons about communicating a new observation ω_i^t to its teammates. For each teammate, $a_j \in A_{-i}$, a_i runs two separate mental simulations and computes the respective expected cost as follows.

To compute the expected cost of communicating observation ω_i^t to agent a_j, we update $M_{i\to j}^t$ with ω_i^t. The expected cost of communicating is

$$\text{Cost}_{\text{comm}}(\omega_i^t) = c + \mathbb{E}_{P_{i\to j}^{t+1}}[C_i(p)],$$

where c is the cost of communication, C_i is the cost function a_i attributes to the world, and the expected cost of the new plan distribution can be computed efficiently with a $\mathbf{\Pi}$-tree.

To compute the expected cost of not communicating the observation ω_i^t to agent a_j, a_i evaluates

$$\text{Cost}_{\text{no-comm}}(\omega_i^t) = \mathbb{E}_{P_{i\to j}^{t}}[C_i(p)].$$

In the case that the observation violates the precondition of a_j's possible future actions, resulting in infinite expected cost, we compute the induced cost up to the point of failure and re-plan to obtain the remaining expected cost. Alternatively, a domain-specific finite cost can be defined for nodes where pre-conditions are violated to account for the cost of re-planning.

With these mental simulations, each agent acts to minimize the projected cost of task execution.

4.5 Reduce Divergence in Mental Models

Our method of reasoning about communication relies solely on having an accurate representation of other team members' plans and states. An increase in uncertainty of a team member's plan results in lower performance of the above communication model. The decentralized nature of the problem as well as uncertainties in task decompositions lead to agents having growing uncertainty in their beliefs of other team members' plans.

As agents explore different subspaces of the world, a single agent may decide not to communicate an observation according to the above reasoning but update its own plan accordingly, which will affect the correctness of others' mental models of the agent. While it is inevitable, without full share of information, that agents' mental models diverge from reality in a nondeterministic world, we take the following approaches to reduce the divergence without introducing new communications.

The first two methods are simple logical assumptions we make about information exchange: First, for each message sent to teammate a_j, we assume it's arrival and update $M_{i\to j}$ as simulated. Second, for each received communication from teammate a_j, we take one step backwards in time to update $M^{t-1}_{i\to j}$ with the observation in the message because the sender must also have made the observation that triggered communication.

Third, for each received message, we draw indirect information about the sender to refine our mental model about them. Intuitively, a message pertaining to a region of the world provides location information regarding the sender's state, which may align with some of our attributed plans but not others. Specifically, when a_i receives an incoming observation from teammate a_j, a_i has indirectly observed a_j's action of observing and communicating ω. With this observation, we are able to refine our belief distribution over $P_{i\to j}$ and $S_{i\to j}$ with simple Bayesian inference as follows:

$$\mathbb{P}_{P'_{i\to j}}[p] \propto \mathbb{P}_{P_{i\to j}}[p] \cdot \mathbb{P}[\omega|p],$$

$$\mathbb{P}_{S'_{i\to j}}[s] \propto \mathbb{P}_{S_{i\to j}}[s] \cdot \mathbb{P}[\omega|s].$$

When an agent's mental model of another is completely incorrect, it is possible that the agent evaluates $\mathbb{P}[\omega|p] = 0$ for all $p \in P_{i\to j}$ and is unable to normalize. In this case, a_i can be confident that $M_{i\to j}$ is off-sync from reality. In future works, this can serve as a strong indicator for triggering communications for re-synchronizing mental models. In this paper, however, we drop all future communications to a_j to avoid sending unnecessary messages.

Chapter 5

Empirical Analysis

We evaluate our approach for sequential SAT-DP with a hybrid of the Mars Explorer Rover (MER) domain [19] and Colored Trails (CT) game [10]. Specifically, we use an HTN representation of the domain as provided by the Simple Hierarchical Ordered Planner 2 (SHOP2) [20].

In this domain, agents are Mars rovers whose goals are to collect and communicate various data to Earth via the Lander. Agents have varying capabilities such as having a colored camera to collect image data and being equipped to sample specific type of minerals. Compound tasks in the domain include *SampleRockData*, *GetImageData*, and $NavigateToWaypoint$, while primitive tasks include *Pickup*, *Drop*, and $MoveToNeighbor$. Compound tasks such as *GetSoilData* may have multiple possible decompositions that are equally optimal if the input state has multiple soil objects equidistant from the agent. Decompositions of compound tasks are task-networks with constraints resulting in ordered lists of subtasks.

We model the costs and uncertainties in the environment as done in the CT game where accessing each location has an associated *access-cost*. Agents initially assign identical and uniform cost of 1 to all actions and have no prior model of how the cost would change. Agents can only observe the real cost of a location once it has accessed the location and incurred the cost. This is analogous to traps with finite costs in the CT game.

During execution, agents update their beliefs over the costs of accessing different

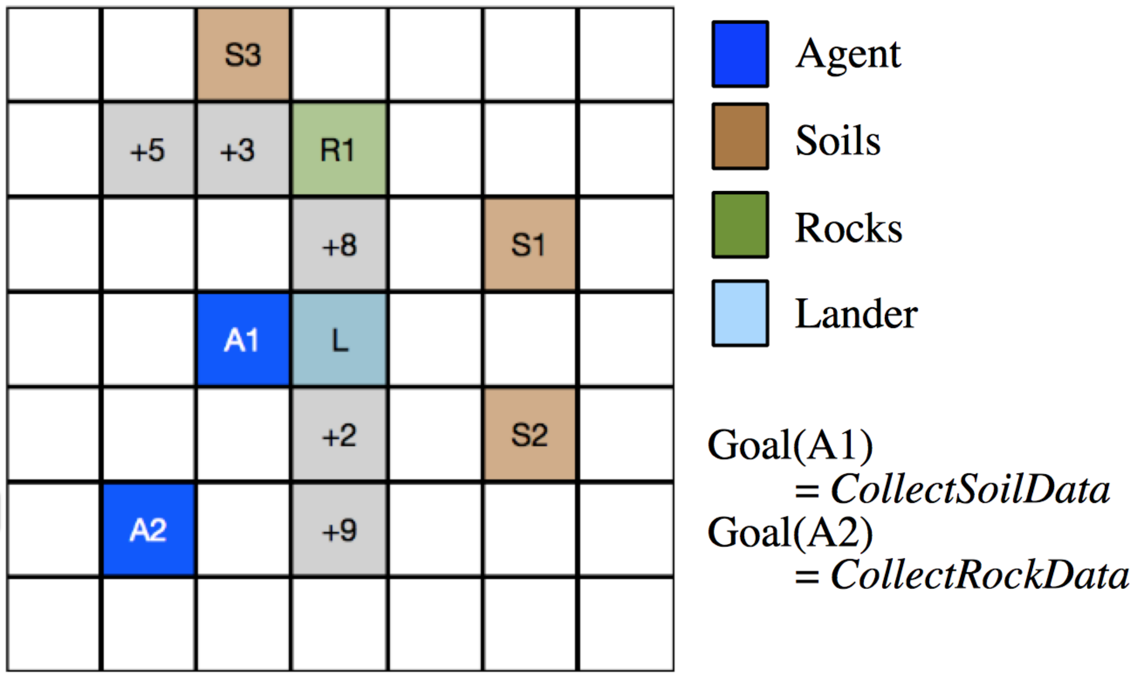

Figure 5-1: An example scenario highlighting the behavior of our approach.

locations. Supportive communication is when an agent discovers and communicates a significant increase in cost to its teammate to prevent the teammate from incurring the same cost at the same location. An example scenario highlighting the behavior of our approach is shown in Figure 5-1.

In Figure 5-1, $A1$ and $A2$ are rovers assigned to collect soil and rock data. Both agents must sample their target, analyze them, and communicate to Earth via the Lander (L). The actual costs of accessing each location is shown in gray $+x$. In this simulation, $A1$ has multiple soil options which means $A2$ is uncertain about the details of $A1$'s plans. When $A1$ observes +8 at location $(2, 3)$, it communicates the observation because $A1$ is certain of $A2$'s plan and that communicating will allow $A2$ to re-plan. When $A2$ receives the communication from $A1$ regarding on +8, $A2$ updates its belief of $A1$'s plan via Bayesian plan recognition becoming confident that $A1$ has chosen to sample $S1$. Furthermore, $A2$ will decide to not communicate observations +5, +3, +2, and +9 given the updated belief over $A1$'s plan.

We test our approach over randomly generated simulation environments. In each simulation, we randomly allocate agents, Landers, and some number of rocks, soils, and other objects around an n-by-n grid world. We also randomly assign mineral sampling capabilities to each agent. When multiple rocks and soils are specified in the environment, each agent may choose at random which sample data to collect. Goals are assigned to each agent such that the team is able to complete the task without communication. However, considering the uncertainties in the world, effective communication would improve team performance.

To generate stochasticity in the environment, each location has some probability of contributing an increase in the access-cost above the default value. The amount increase is uniformly sampled from the interval $[0, 10]$. We model all cost changes in the environment to occur at $t = 0$, but agents can only observe the change once they land in the location. This is to avoid agents communicating observations that later become obsolete.

In a single task execution, we specify an HTN problem and the cost of communication, and a communication model. We provide each agent with an HTN solver and identical domain knowledge. All communications during execution are triggered from the agent's communication model. We evaluate the performance of a single execution as the total incurred cost of all agents to achieve their specified goals.

Our implementation uses a comprehensive HOP adapted from the HTN planner, SHOP2 [20]. We incorporated cost into SHOP2 by implementing a branch and bound solver to compute an optimal solution. The output of our modified planner is the AND-OR tree described in the previous section.

Chapter 6

Results

We compare our proposed approach with four different communication baseline models:

- **NoComm**: Agents never communicate.
- **RandComm**: Agents communicate each observation with probability 0.5.
- **FullComm**: Agents communicate all observations.
- **M1**: Agents treat each observation as a one-shot SAT-DP and reason about communication by evaluating the expected cost of each action.

M2 is our proposed model where agents reason about communication as described in our approach section and use Bayesian plan recognition to reduce divergence in mental models.

Our baseline model **M1** is equivalent to the *inform* protocol from prior work by [16]. However, our simulation environment differs from that of [16] and [2] in that our agents do not have full observation of the board. In [16], agents were evaluated in simulations with low state uncertainty and moderate planning uncertainty. Low state-uncertainty means that the agent responsible for communicating new information has an omniscient view of the board, including the position of the teammate at every time step. Moderate planning uncertainty means that each agent must reason over possible plans of its teammate. In contrast, we conduct simulations with what we consider to be both moderate state uncertainty and moderate planning uncertainty. In our simulations, agents do not observe the actions of its teammates and therefore must reason over both possible plans and possible current states of their teammates.

We evaluate our model with two sets of experiments. In the first experiment, we reduce the planning uncertainty by modifying the HTN solver to output a deterministic plan for a given HTN problem. As a result, the plan distributions that agents attribute to their teammates are deterministic and identical to the agents' actual plans. These simulation scenarios model situations where each agent's mental model of each other is highly accurate. However, there is still uncertainty in communication because agents explore different regions of the world and modify their plans. Agents' attributed state distributions cannot account for changes that were not communicated to the agent.

For this experiment, we generated 100 5-by-5 world configurations with two agents per world and with each location having 0.3 chance of incurring an unexpected cost. For each problem, we ran our models with varying cost of communication c. We report the median total incurred team cost of the five models per c-value over 100 problem configurations, shown in Figure 6-1.

Note that in our experiment, we use the same set of problems for each value of cost of communication c. We expect that the averaged total cost for NoComm is constant regardless of c since no messages were sent during execution and the total cost is the aggregate cost incurred by every agent. On the other hand, FullComm is always linear with slope proportional to the total number of observations made during execution. The intersection between NoComm and FullComm provides insight into the effect of the stochasticity in the world, representing the ratio between the amount of damage caused by the stochasticity and their frequency of unanticipated occurrences in the world. Given a world configuration, the best possible performance is indicated by FullComm with $c = 0$, where teammates share all observations about the world at no extra cost.

The difference in median performance that **M2** demonstrates over **M1** is the result of reducing divergence of mental models via the first two logical approaches where

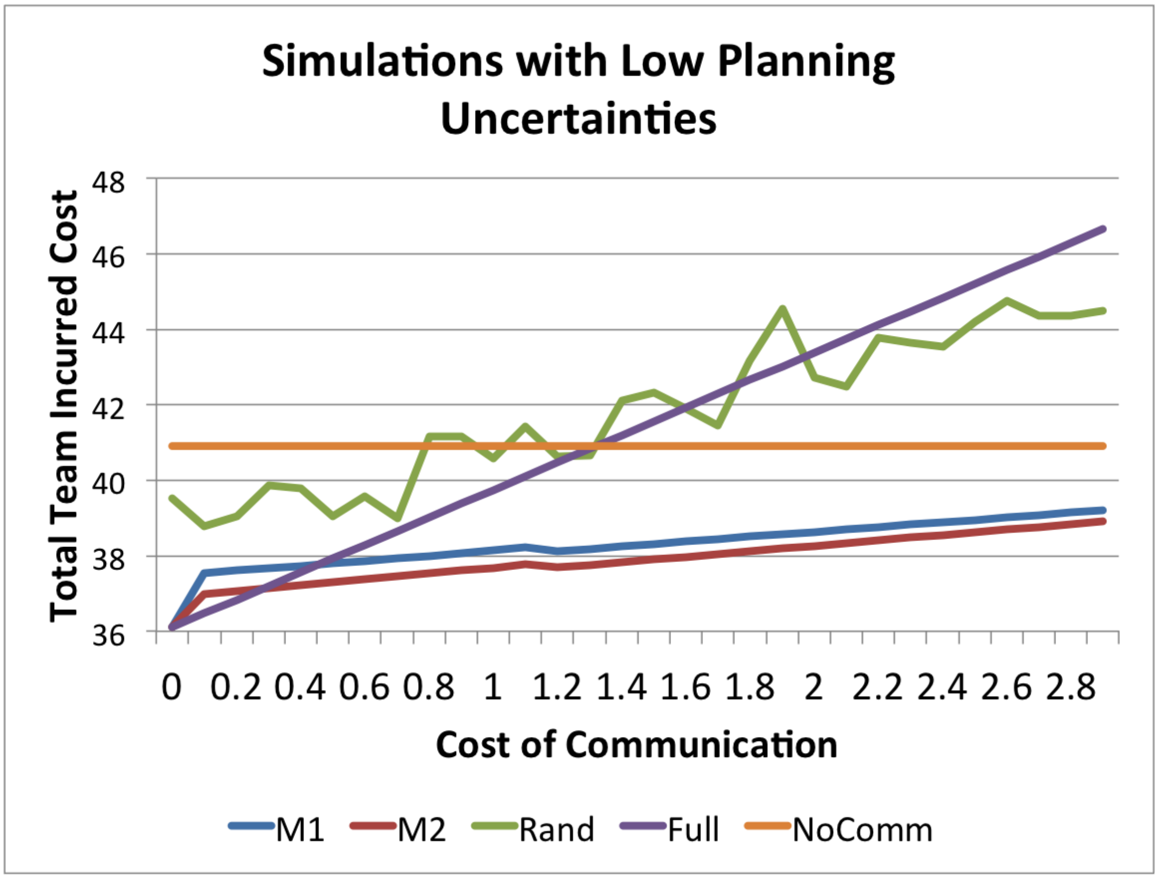

Figure 6-1: Comparing five models in simulation environments with low planning uncertainty where each task had one and only one possible decomposition. The world uncertainties were generated with each locating having 0.3 probability of incurring an increase in cost.

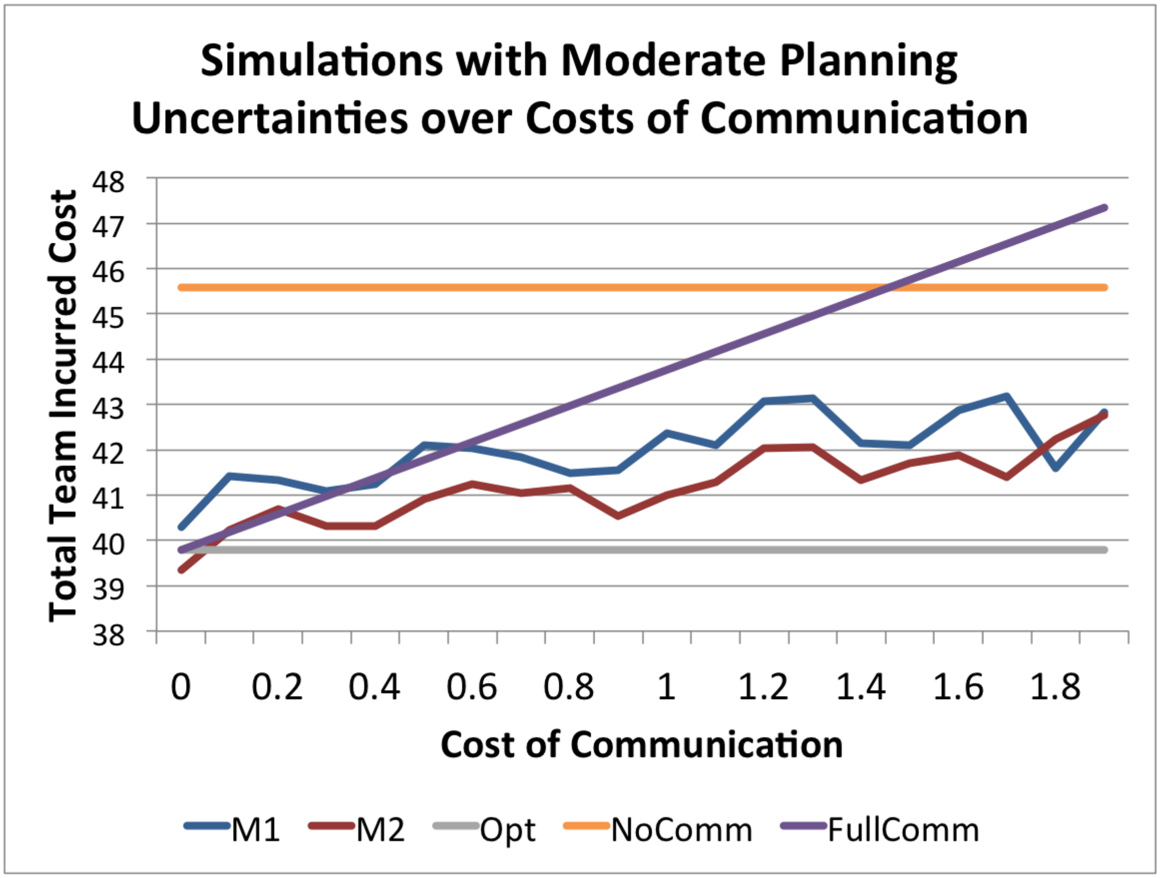

Figure 6-2: Comparing four models in simulation environments with high planning uncertainty and high state uncertainties. *Opt* plots the optimal team performance where agents share all observations assuming 0 cost of communication.

agents assume 1) arrival of outgoing communication to update attributed mental model of the receiver, and 2) sender of an incoming message has previously made the same observation. Because attributed plans are deterministic in this experiment, performing Bayesian plan recognition on incoming messages would not change the distribution.

Our second experiment more realistically evaluates our model and baseline models with both moderate state uncertainty and moderate planning uncertainty. Again, we generate 100 5-by-5 world configurations with the same parameters as the previous experiment. With multiple possible execution plans for a single problem, we repeated each problem 25 times to report the average execution cost for a particular problem with a specific value of c. We plot the median execution cost over 100 problem

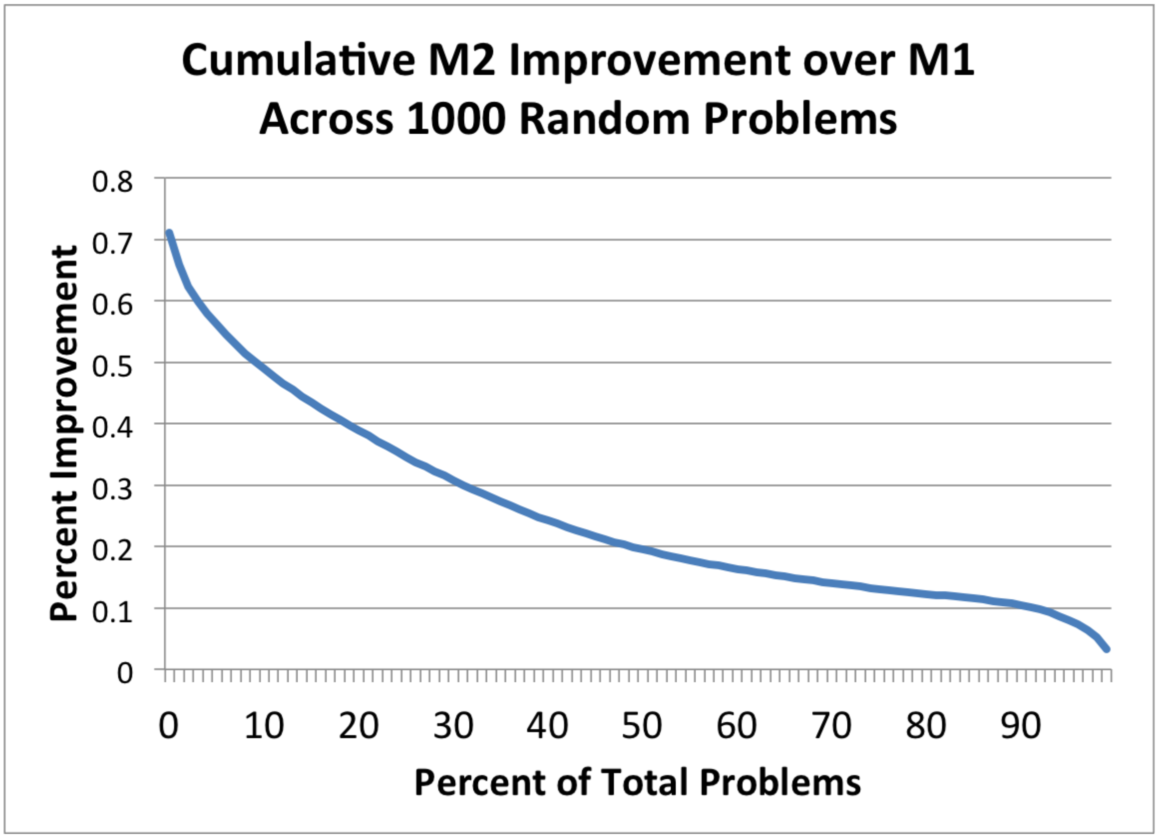

Figure 6-3: An illustration of M2's performance gain over M1 as a function of the top x percent of the total problems, where x is the x-axis.

configurations. Figure 6-2 compares the model performances over varying cost of communication.

The average improvement that **M2** demonstrates over **M1** is the result of reducing divergence of mental models as described in the Approach section, incorporating the two logical methods as well as applying Bayesian plan recognition to incoming messages.

From Figure 6-2, we observe that the median improvement our model demonstrates over the baseline model is around one cost-point, which is about 2%. In most randomly generated problems, **M2** does not demonstrate significant improvement over **M1**. For example, when agents make few observations or decide on exchanging small number of communications, our proposed solution would not be able to demonstrate the benefit of maintaining a synced mental model. However, there are situations, which may occur with low likelihood, that could benefit significantly from effective communication strategies. We analyze the performance of $M2$ and $M1$ over 1000 random problem configurations with cost of communication varying from 0.1 to 2. Figure 6-3 illustrates **M2**'s performance gain over **M1** averaged over the top x percent of the total problems. Specifically, for 1% of the problems, our model demonstrated an average of 71.1% improvement over **M1**, 50.1% improvement when averaged over the top 10% of the problems, 39.8% improvement over the top 20% of the problems, and 20.1% improvement over half of the problems.

Chapter 7

Discussions and Applications

In our approach, we specifically considered HTN planning domains where methods may have multiple decompositions and enabled agents to randomly select a decomposition during planning. As a result, these assumptions introduced planning uncertainties that our approach needed to address. These assumptions could be easily eliminated in multi-agent planning by providing each agent with a deterministic planner or identical random seed to a pseudorandom number generator.

We consider planning uncertainty in our problem in order to build a framework that can seamlessly be extended to mixed human-robot teams. When there are multiple options for executing a task, a human's plan can no longer be deterministic. In addition, the human may have personal preferences or external domain knowledge, which can be modeled using our proposed $\mathbf{\Pi}$-tree by assigning non-uniform edge probabilities between OR-nodes and their children. In future works, these edge probabilities can also be learned from past task executions.

To incorporate a human teammate, we specify the capability of the human in the HTN planning domain and assign high cost of communication to the human agent. This will result in agents communicating only the most important information to the human agent and thereby limiting cognitive load of the human.

Chapter 8

Conclusion and Future Work

In this paper, we formally defined the sequential SAT-DP to incorporate multiple observations and to highlight the need for maintaining accurate mental models of other teammates. We presented a decision-theoretic solution to the sequential SAT-DP where we 1) used a probabilistic ordered AND/OR tree to represent a distribution over possible solutions to a hierarchical planning problem, and 2) performed belief updates by making logical assumptions about communication and by performing Bayesian plan recognition on incoming messages. We demonstrated the effectiveness of our solution by comparing against a baseline model that naively applies SAT-DP solution over time.

The sequential extension to the SAT-DP problem provided in this work not only is more realistic, but also opens the door to many time-based approaches. For example, agents may incorporate confidence in their mental model of other teammates as a function of the time passed since last communication. Other potential avenues for future work includes further exploiting the hierarchical structures in the planning domain to make faster evaluations and belief updates.

Bibliography

[1] Christopher Amato, George D Konidaris, and Leslie P Kaelbling. Planning with macro-actions in decentralized pomdps. In *Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems*, pages 1273–1280. International Foundation for Autonomous Agents and Multiagent Systems, 2014.

[2] Ofra Amir, Barbara J Grosz, and Roni Stern. To share or not to share? the single agent in a team decision problem. *Models and Paradigms for Planning under Uncertainty: a Broad Perspective*, page 19, 2014.

[3] Gordon Briggs and Matthias Scheutz. Facilitating mental modeling in collaborative human-robot interaction through adverbial cues. In *Proceedings of the SIGDIAL 2011 Conference*, pages 239–247. Association for Computational Linguistics, 2011.

[4] Philip R Cohen and Hector J Levesque. Intention is choice with commitment. *Artificial intelligence*, 42(2):213–261, 1990.

[5] Philip R Cohen and Hector J Levesque. Teamwork. *Nous*, pages 487–512, 1991.

[6] Leslie A DeChurch and Jessica R Mesmer-Magnus. Measuring shared team mental models: A meta-analysis. *Group Dynamics: Theory, Research, and Practice*, 14(1):1, 2010.

[7] Kutluhan Erol, James A Hendler, and Dana S Nau. Umcp: A sound and complete procedure for hierarchical task-network planning. In *AIPS*, volume 94, pages 249–254, 1994.

[8] Xiaocong Fan, Rui Wang, Shuang Sun, John Yen, and Richard A Volz. Context-centric needs anticipation using information needs graphs. *Applied Intelligence*, 24(1):75–89, 2006.

[9] Xiaocong Fan, John Yen, and Richard A Volz. A theoretical framework on proactive information exchange in agent teamwork. *Artificial Intelligence*, 169(1):23–97, 2005.

[10] Yaakov Gal, Barbara J Grosz, Sarit Kraus, Avi Pfeffer, and Stuart Shieber. Colored trails: a formalism for investigating decision-making in strategic environments. In *Proceedings of the 2005 IJCAI workshop on reasoning, representation, and learning in computer games*, pages 25–30, 2005.

[11] Ilche Georgievski and Marco Aiello. Htn planning: Overview, comparison, and beyond. *Artificial Intelligence*, 222:124–156, 2015.

[12] Claudia V Goldman and Shlomo Zilberstein. Decentralized control of cooperative systems: Categorization and complexity analysis. *J. Artif. Intell. Res.(JAIR)*, 22:143–174, 2004.

[13] Robert P Goldman, Christopher W Geib, and Christopher A Miller. A new model of plan recognition. In *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*, pages 245–254. Morgan Kaufmann Publishers Inc., 1999.

[14] Barbara J Grosz and Sarit Kraus. Collaborative plans for complex group action. *Artificial Intelligence*, 86(2):269–357, 1996.

[15] Luiz S Homem de Mello and Arthur C Sanderson. And/or graph representation of assembly plans. *Robotics and Automation, IEEE Transactions on*, 6(2):188–199, 1990.

[16] Ece Kamar, Ya'akov Gal, and Barbara J Grosz. Incorporating helpful behavior into collaborative planning. In *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems-Volume 2*, pages 875–882. International Foundation for Autonomous Agents and Multiagent Systems, 2009.

[17] Sanjeev Kumar, Marcus J Huber, David R McGee, Philip R Cohen, and Hector J Levesque. Semantics of agent communication languages for group interaction. In *AAAI/IAAI*, pages 42–47, 2000.

[18] Pierre Lison, Carsten Ehrler, and Geert-Jan M Kruijff. Belief modelling for situation awareness in human-robot interaction. In *RO-MAN*, pages 138–143, 2010.

[19] Derek Long and Maria Fox. The 3rd international planning competition: Results and analysis. *J. Artif. Intell. Res.(JAIR)*, 20:1–59, 2003.

[20] Dana S Nau, Tsz-Chiu Au, Okhtay Ilghami, Ugur Kuter, J William Murdock, Dan Wu, and Fusun Yaman. Shop2: An htn planning system. *J. Artif. Intell. Res.(JAIR)*, 20:379–404, 2003.

[21] Alexander Pokahr, Lars Braubach, and Winfried Lamersdorf. Jadex: A bdi reasoning engine. In *Multi-agent programming*, pages 149–174. Springer, 2005.

[22] David V Pynadath and Milind Tambe. Multiagent teamwork: Analyzing the optimality and complexity of key theories and models. In *Proceedings of the first international joint conference on Autonomous agents and multiagent systems: part 2*, pages 873–880. ACM, 2002.

[23] Earl D Sacerdoti. Planning in a hierarchy of abstraction spaces. *Artificial intelligence*, 5(2):115–135, 1974.

[24] Kartik Talamadupula, Gordon Briggs, Tathagata Chakraborti, Matthias Scheutz, and Subbarao Kambhampati. Coordination in human-robot teams using mental modeling and plan recognition. In *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*, pages 2957–2962. IEEE, 2014.

[25] Milind Tambe. Agent architectures for flexible. In *Proc. of the 14th National Conf. on AI, USA: AAAI press*, pages 22–28, 1997.

[26] John Yen, Xiaocong Fan, Shuang Sun, Timothy Hanratty, and John Dumer. Agents with shared mental models for enhancing team decision makings. *Decision Support Systems*, 41(3):634–653, 2006.

[27] Yu Zhang, Sarath Sreedharan, and Subbarao Kambhampati. Capability models and their applications in planning. In *Proceedings of the 2015 international conference on Autonomous agents and multi-agent systems*. International Foundation for Autonomous Agents and Multiagent Systems, 2015.