ON MODELING AND SIMULATING AGENT TEAMWORK IN CAST

John Yen, Xiaocong Fan, Shuang Sun, Ray Wang,
Cong Chen, Kaivan Kamali, Mike Miller[2] and Richard A. Volz[2]

[1] *Research Center for Team-based Agents*
School of Information Sciences and Technology
The Pennsylvania State University
University Park, PA16802
E-mail: {jyen, zfan}@ist.psu.edu
[2] *Department of Computer Science*
Texas A&M University
College Station, TX 77843

Effective human teams use overlapping shared mental models for anticipating information needs of teammates and for offering relevant information proactively. The long-term goal of our research is to empower agents with such "shared mental models" so that they can be used to better simulate, train, or support human teams for their information fusion, interpretation, and decisions. Toward this goal, we have developed a team agent architecture called CAST that enables agents to infer information needs of teammates, which further enables agents to assist teammates by proactively delivering needed information to them. In this paper, we focus on two key issues related to proactive information delivery behavior. First, we model the semantics of proactive information delivery as an attempt (called ProAssert), which extends the performative Assert in Joint Intention Theory. Second, we introduce a decision-theoretic approach for reasoning about whether to act on a potential proactive assert. Experimental results suggested that the decision-theoretic communication strategy enhances the team performance. The formal semantics and the decision-theoretic communication strategies together provide a sound and practical framework that enables further studies regarding proactive information delivery for supporting the decision making of a team involving human and agents.

1 Introduction

Effective human teams use overlapping shared mental models for anticipating information needs of teammates and for offering relevant information proactively. Agents empowered with such "shared mental models" can be used to better simulate, train, or support human teams for their information fusion, interpretation, and decisions. This is a highly challenging objective, since the scope of shared mental model is very broad. For instance, shared mental model obviously includes shared ontologies about the problem domain, shared team goals or commitments, shared knowledge about the team structure and the process of the team, etc.

Toward this long-term goal, in this paper, we focus on three issues related to shared mental model (SMM) for a specific assist behavior among team members–proactive delivery of information to needed teammates before they request. Obviously, this assist behavior requires an agent to know what information is needed by its teammates. Information needs, hence, become a critical component of shared mental model that enables the proactive information delivery behavior. However, it is desirable for an agent to reason about, rather than just remember, information needs so that it can modify or refine its anticipation about information needs of other teammates based on other more basic constructs in the shared mental model. We address these issue in Section 3 by formally defining information needs within the framework of the SharedPlan theory. The definition extends our earlier work[19, 20] by distinguishing two kinds of information needs: information that is needed for a teammate to perform its task, and information that is needed by a teammate to protect (i.e., avoid conflict with) its goals. This definition of information needs lays the foundation for developing algorithms (e.g., the DIARG algorithm in CAST) for agents to dynamically reason about information needs of their teammates.

Even if an agent can reason about information needs of its teammates, such capability is not useful unless it can translate it into communicative acts (i.e., proactive information delivery) that can benefit the team. Obviously, the semantics of such proactive communicative acts are much more complicated than the typical performatives like Assert[1, 2], since they are motivated by anticipations to the future. We argue in Section 3 that the semantics of proactive inform action is different from that of a conventional inform action in that it also include the speakers intent to communicate about his belief regarding the recipients information needs. A benefit of this formal semantics for proactive inform is that it enables agents to establish a communication protocol with two different kinds of reject reply: (1) reject due to knowing conflicting information, and (2) reject due to not needing the information. These two different replies, in turn, can be used by agents to update their SMM about the teammates they tried to help.

Finally, agents need to choose proactive communication strategies in a way that can benefit the team. Even a seemingly simple decision such as "whether to proactively inform a teammate about information I" can be complicated by several factors. First, the "value" of the information I may be difficult to assess. Second, the agent may be uncertain about the outcomes of communicating I vs. not communicating I, due to its incomplete knowledge about the world. In Section 4, we propose a decision-theoretic approach that addresses these issues by using a shared mental model that handles uncertainty. We have implemented this communication strategy in the CAST agent architecture and evaluated it empirically by using a scenario that involves two opposing agent teams. The evaluation result is reported in Section 5.

2 CAST Overview

CAST (Collaborative Agents for Simulating Teamwork) is designed to enhance collaboration in a team that consists of both human agents and software agents. Studies about human team have identified proactively offering information needed by teammates as one of the key behaviors of effective teamwork. Such a behavior is based on a mental model shared among members of the team. The main novelty of the CAST architecture, hence, is that it enables agents not only to develop and update their shared mental model but also to use such models for proactive information exchanges and accomplishing other effective team behaviors.

The design of CAST architecture is guided by three objectives: scalability, efficiency, and adaptability. Scalability is achieved using a high level language (MALLET) for describing team task knowledge. Efficiency is realized by algorithms that utilize the team task knowledge effectively. Adaptability is accomplished by dynamic responsibility assignment built into the architecture. MALLET (Multi-Agent Logic Language for Encoding Teamwork) is a high-level team knowledge representation language, which provides descriptors for encoding knowledge about individual/team actions and plans, as well as specifications of team structures (e.g., roles and responsibilities). CAST is a domain-independent team-based agent architecture. The domain knowledge of CAST agents came from MALLET. Team knowledge in MALLET is compiled into Predicate-nets, which are used by CAST kernel for generating effective teamwork behaviors.

CAST kernel is composed of a set of algorithms that CAST agents use to decide what actions (including communication actions) they will take at each time step. All of the kernel algorithms rely on a computational shared mental model of the team. Two novel algorithms of CAST are DRS and DIARG. DRS dynamically selects agents for responsibilities in a team plan based on constraints specified in the plan. DIARG identifies opportunities for proactive delivery of information needed by teammates. The two algorithms together achieve efficiency (by sending information only to those who need them) and adaptability (through constraint-based dynamic responsibility assignments) for a team involving CAST.

The shared mental model of a CAST agent includes three components: (1) the shared teamwork knowledge described in MALLET, (2) the shared belief about responsibility assignment of the teammates captured in the predicate net, and (3) the shared belief about the world, which is stored as

Horn clauses in a Prolog-like knowledge base. The CAST kernel updates its shared mental model through its sensing, communication, and coordination with other agents. A more detailed description of CAST can be found in [19].

3 Foundation of Proactive Information Delivery

In the following, let TA be a fixed set of agents in the team under concern. We base our analysis on actions. Actions have various associated properties, such as the collection of potential doers, the condition under which the action can be performed, the consequences of performing it, etc. As in [7], we use $\alpha, \beta, \gamma \cdots$ to refer to actions, and assume a set of functions can be used to obtain the various properties associated with an action. Specifically, we use $Act_\alpha, pre(\alpha)$ and $post(\alpha)$ to return information regarding the potential doers, the preconditions and effects of α, respectively. More specifically, Act_α returns a set of tuples in the form $\langle Ag_i, level_i, T_i, Cost_i \rangle$[3], where Ag_i is a set of agents capable of performing α, and includes only those agents which really have contribution to the performance of α. T_i and $Cost_i$ are the time duration and cost for Ag_i to perform α, respectively. The value of $level_i$ is either *basic* or *complex* [7], specifying whether α is basic or complex wrt Ag_i. If $level_i$ is *basic*, the action is performable at will with no further decomposition or planning, and $Ag_i = \{G\}$ must be a singleton, i.e., agent $G \in TA$ can do α individually. When Ag_i is not a singleton, then α is a multiple agent action for the agents in Ag_i. A complex action (action expression), being fully instantiated, is a sequence of basic-level (primitive) actions satisfying certain properties, which characterize what will happen in each possible world. A *recipe* for action α is a specification of a group of subsidiary actions at different levels of abstraction, the doing of which under certain constraints constitutes the performance of α. Thus, a recipe is in per se composed of an action expression and a set of constraints on the action expression. By meta predicate $Action(G, \alpha)$ we mean G is a candidate doer for (complex) action α.

As in the SharedPlan theory, we use modal operator $Do(G, \alpha, t, \Theta)$ to denote that G(a group or a single) does(did) (complex) action α at t under constraints Θ. Bel and MB are standard modal operators for belief and mutual belief, respectively. $unknown(A, p, t) \triangleq \neg Bel(A, p, t) \wedge \neg Bel(A, \neg p, t)$, which means that agent A does not know (hold any belief about) the state of p at time t. There exist four kinds of intentional attitudes. $Int.To(G, \alpha, t, t_\alpha, C_\alpha)$ means G at t intends to do α at t_α in the context C_α, where C_α accounts for the reason of doing α. $Int.Th(G, p, t, t_\alpha, C_p)$ means G at t intends that p hold at t' with C_p as its intentional context.

---

[3] An action may be taken as a basic action for one agent, but may be a complex action for another.

$Pot.Int.To$ ($Pot.Int.Th$) is similar to $Int.To$ ($Int.Th$) except that it could not be evolved into $Int.To$ ($Int.Th$) before being reconciled with the already adopted intentions-to (intentions-that). Should there be any conflicts, $Pot.Int.To$ ($Pot.Int.Th$) will be dropped. $Int.To$ is re-defined in [18] to embed pre-information checking.

3.1 Anticipate Information Needs of Teammates

The most challenging issue in enabling agents to proactively deliver information to teammates is for them to anticipate information needs of teammates. This is challenging because anticipating the information needs of teammates requires a shared mental model about their responsibilities, their goals, and their belief, all of which can be difficult to predict due to the dynamic nature of the environment.

There are at least two types of information needs. The first type of information needs enables an agent to perform certain (complex) actions, which contributes to an agent's individual commitments to the whole team. We call this type of information need *action-performing information need*. The second type of information need allows an agent to protect a goal from potential threats that may result in a conflict with the goal. Knowing such information will help teammates to handle threat to the team goals. Thus, we call this type of information need *goal-protection information need*. For instance, suppose fighters are responsible for protecting bombers which have a goal of destroying the enemy base. The dynamic locations of enemies are action-performing information for fighters, since prior to performing *firing*, the fighters have to know where the targets are. The same kind of information is goal-protection information for bombers, since if the bombers are unaware of the approaching enemies, they might be destroyed from the flank and the mission might become impossible.

Next we give the definition of information needs in terms of the mental states and capability of an agent. Meta-predicate $Need(A, I, t)$ is used to denote the fact that agent A needs information I at future time t. Instead of defining $Need$ directly, we'd rather define it in the contexts of beliefs.

Definition 1. $\forall A, B \in TA, I, t, t' \geq t \cdot$

$$
\begin{aligned}
&Bel(A, Need(B, I, t'), t) \triangleq Bel(A, \neg Bel(B, I, t'), t) \wedge \\
&((\exists \alpha, t_0 \leq t \cdot Action(B, \alpha) \wedge Bel(A, Int.To(B, \alpha, t_0, t', \Theta_\alpha), t) \wedge \\
&\quad (Bel(A, (I = pre(\alpha)), t) \vee \\
&\qquad (\exists \beta \cdot Action(B, \beta) \wedge Bel(A, (I = pre(\beta)), t) \wedge \\
&\qquad\ (\exists t_\beta < t' \cdot\ Bel(A, (Do(B, \beta, t_\beta, \Theta_\beta) \Rightarrow pre(\alpha)), t) \wedge \\
&\qquad\ Bel(A, \exists R_\beta \cdot CBA(B, \beta, R_\beta, t_\beta, \Theta_\beta)), t))))) \\
&\vee (\exists GR \subseteq TA \cdot (A \in GR) \wedge (B \in GR) \wedge \\
&\qquad Bel(A, \exists t_1, C_1 \cdot Int.Th(GR, \phi, t, t_1, C_1), t) \wedge \\
&\qquad Bel(A, \neg Bel(B, I, t') \Rightarrow \neg \phi, t))).
\end{aligned}
$$

That is, A at t believes B will need I by t' means, at t agent A believes either one set of the following facts. (1). B will not believe I at t', B intends at t_0 to do some actions α at t', and I is exactly the pre-condition of α; (2). B will not believe I at t', B intends at t_0 to do some actions α at t', and I is the pre-condition of some other action β, the performance of which could lead to $pre(\alpha)$, and B can bring about β by following some recipes [4]; (3). B will not believe I at t', both A and B belong to the same (sub-)team, the goal of which is ϕ, but if B could not get I at t', the goal becomes impossible.

The following axiom plays a key role in connecting information needs with proactive communication actions. It says that, when an agent has realized (been told) that another agent might need a piece of information, it will generate an intention-that to try to provide help. We use predicate $needs(B, I)$ to denote the context for such intention, which is generated for the reason that B will need I.

Axiom 1 $\forall A, B \in TA, I, t, t' > t \cdot\ Bel(A, Need(B, I, t'), t)$
$\Rightarrow Int.Th(A, Bel(B, I, t'), t, t', needs(B, I))$.

3.2 Attempts as Certain Mental States

Following Cohen and Levesque's work[1, 4], we attempt to bridge the joint-intention theory and the SharedPlan theory by the semantic analysis of "communicative acts", and show that the treatment of performative-as-attempt can be successfully carried out in the framework of the SharedPlan theory. However, in our analysis, an *Attempt* will no longer be treated as a complex action expression, but be treated as a view, or a slice of the current mental state of the performers. Based on this, we formally define the semantics of a new kind of performatives– proactive assert(tell), which is prevalent in teamwork domains involving both human and software agents.

In [1] attempt was defined as a complex action expression. Agent x attempts to achieve P via achieving Q by doing e means just prior to e, x chooses that P should eventually become true, and intends that e should produce Q relative to that choice. That is, before doing e, x's mental state has to satisfy certain pre-conditions.

Rather than defining an attempt as a complex action expression, we treat it as a certain slice of mental state which could legally lead to the commitment of doing the associated event.

Definition 2. $Attempt(x, e, P, Q, \theta, t, t') \triangleq$
$\neg Bel(x, P, t) \wedge\ Int.Th(x, P, t, t', \neg(Bel(x, P, t) \vee \theta)) \wedge$

---

[4] Note that β might be a complex single action or a multiple action. Hence B might not be able to get $pre(\alpha)$ until all the constituent actions of β have been performed.

$$
\begin{aligned}
&Bel(x, post(e) \Rightarrow Q, t) \wedge \\
&(\exists t_1 \cdot (t \leq t_1 < t') \wedge Int.To(x, Exec(x, e, t_1), t, t_1, \\
&\qquad Int.Th(x, P, t, t', \neg(Bel(x, P, t) \vee \theta)))).
\end{aligned}
$$

That is, at time point t, agent x attempts to make P hold at t' via achieving Q by doing e relative to the escape condition θ means, agent x does not believe P hold currently, it personally knows Q follows from the effects of e, it intends that P hold at t' relative to θ and the fact that P doesn't hold currently, and it intends to execute e before t' relative to its intention that P hold at t'.

Since intentions are persistent by default, if an agent makes an attempt, it will keep the attempt until either t' comes, or it comes to believe P hold by some effort, or the escape condition becomes true. In any case, all the constituent intentions of the attempt will be dropped because the corresponding constrains do not hold any longer.

The semantics of performatives are given by choosing appropriate formulas (involving mutual beliefs) to substitute for P and Q in the definition of *Attempt*.

3.3 Proactive Assert

There are two restrictions in the semantics of *Assert*[17]. First, since the escape condition of $Assert(s, a, p, t, t')$ is given as $\exists t_1 \cdot (t \leq t_1 \leq t') \wedge Bel(s, Bel(a, \neg p, t_1), t)$, s will not assert p to a, if s currently believes $Bel(a, \neg p, t_1)$ at some time t_1 between t and t'. However, in some cases an agent wants to change the other's mind when this agent believes the other agent is holding false beliefs. Second, when an agent refuses an assertion about p, it is assumed that the refusal is only due to the fact that the receiver does not want to change its current belief about $\neg p$. However, an agent might anticipate the future information needs of the other teammates, and proactively assert some relevant information to the needer. In such cases, the receiver might refuse an assertion if it believes it will never need the information. Such kind of refusal should convey some meta-level information (above p) to the asserting agent, which could accordingly revise or refine its meta-level knowledge to improve its ability of anticipation in the future. Thus, as a complementation to *Assert*, we need to define a new elementary performative *ProAssert*, which will be based on the information needs among teammates.

Definition 3. $ProAssert(s, a, p, t, t') \triangleq Bel(s, p, t) \wedge$ $Bel(s, Need(a, p, t'), t) \wedge$ $Attempt(s, e, \exists t_0 \cdot (t \leq t_0 \leq t') \wedge Bel(a, p, t_0), \exists t'' \cdot (t \leq t'' \leq t') \wedge MB(\{s, a\}, P, t''), \theta, t, t')$, *where* $P = \exists t_b \cdot (t'' \leq t_b \leq t') \wedge$ $Int.Th(s, Bel(a, Bel(s, p \wedge Need(a, p, t'), t), t_b), t, t_b, \neg Bel(s, \exists t_a \cdot Bel(a, p, t_a), t))$, $\theta = \exists t_1 \cdot (t \leq t_1 \leq t') \wedge Bel(s, \neg Need(a, p, t_1), t)$.

That is, agent s at t pro-asserts p to a by t' is an attempt with a's belief about p by t' as the ultimate goal, and the honest goal is to bring about mutual beliefs that s intends that a believes "s believes p and s believes a will need p at t'". $ProAssert(s, a, p, t, t')$ will not hold when s comes to believe $\neg p$, or when s comes to believe that a will not need p by t'.

Similar to *Assert*, upon receiving a pro-assertion about p, the receiver will deliberate on whether to accept it or not. If it decides to accept p, it will come to believe p, and the acceptance should free the speaker from maintaining the already achieved intention. If it decides to refuse p, it is actually trying to inform the asserting agent that it does not need p.

From the definition of $Need$, it can be inferred that a's information needs was actually generated by a's intention to do some action at t'. s got to know a's information needs by reasoning about a's mental state according to the current team plan shared with a. For instance, suppose all the fighters and scouts share the same team plan to *destroy_enemy_base*, where the scouts keep searching for the location of the enemy base, and the fighters will move towards the enemy base as soon as they get to know such information. In such scenario, the scouts could get to know fighters' information needs by checking their shared team recipe and the current positions of the fighters in the recipe, and then commit to $ProAssert$. [5]

The following theorem can be proved by using Axiom(1) and the help axiom in the SharedPlan theory. It says that if agent A knows B will need I at t', A will try to provide help by pro-asserting I to B.

Theorem 1. $\forall A, B \in TA, I, t, t' > t \cdot$
$Bel(A, Need(B, I, t'), t) \wedge Bel(A, I, t) \wedge$
$(\nexists t_0 \cdot (t_0 \leq t) \wedge Bel(A, ProAsserted(A, B, I, t_0, t'), t)) \Rightarrow$
$(\exists t_t \cdot Pot.Int.To(A,$
$ProAssert(A, B, I, t_t, t'), t, t_t, needs(B, I)))$.

4 A Decision-theoretic Communication Strategy for ProAssert

Communications often carry certain cost. Therefore, an agent needs to evaluate the tradeoff between the cost and the utility of proactive communications before actually doing it. Furthermore, an agent is also required to be able to deal with uncertainties, since it may only have incomplete information about the world, the potential cost and the potential utility of proactive information delivery. Accordingly, it may hold wrong models of teammates. Therefore, we propose a decision-theoretic approach for an agent to reason about whether to reconcile its potential-intentions regarding $ProAssert$ to

---

[5] Here, we are assuming all the agents know the preconditions and effects of all the domain actions. If domain information is distributed, this assumption can be weakened later by allowing agents to communicate the preconditions and effects of the immediate next level actions as we did in [17].

intentions to do it, once it receives a piece of information that matches the need of its teammates.

As shown in theorem 1, if agent A believes B will need I at t', A will try to provide help by adopting an potential intention-to regarding ProAssert. Whether such potential intentions-to can be reconciled to intentions-to depends on agents decisions based on the state of the environment and the mental state of the decision-maker themselves. One of the pre-conditions for agent A to hold a potential intention to $ProAssert$ I to B is that $Bel(A, Need(B, I, t'), t)$ must hold. However, most of such information needs come from A's anticipation based on B's lacking of expected behavior or lacking of sufficient observability[6], and such anticipations about teammates might be wrong. Hence, actually each information need is associated with a probability, which measures to what degree A's belief of $Bel(A, Need(B, I, t'), t)$ conforms to the real state of $Need(B, I, t')$.

Even if B really need I at t', i.e., $Need(B, I, t')$ holds, B might have already known I before t by whatever means that is out of A's anticipation.That is, it might be the case that A spends efforts and time in vain to $ProAssert$ I to B, who has already got information I from other sources. Hence, when A decides to reconcile its potential intentions to intentions with respect to pro-asserting information I to B, it should also consider the probability of $Bel(B, I, t)$. Since our agents assume beliefs persist on default when making decisions, and they always try to maintain those beliefs that will be useful in the future, agent A will not perform $ProAssert$ if it believes B currently also believes I (so that B will satisfy its future needs by its own).

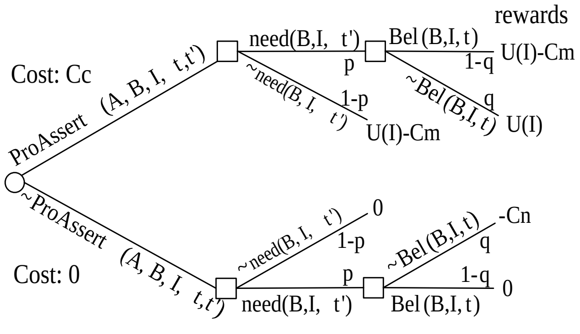

Fig. 1. Decisions on ProAssert under Uncertainty

---

[6] There are cases that A get to know B's information needs through being informed by B.

4.1 Decision Analysis for ProAssert

As explained above, from $Pot.Int.To(A, ProAssert(A, B, I, t, t), t, t', needs (B, I))$ to $Int.To(A, ProAssert(A, B, I, t, t'), t, t', needs(B, I))$, agent A needs to make decisions based on the communication cost C_c for performing $ProAssert$, the benefits U_I of information I, the probability p of $needs(B, I, t')$, and the probability q of $Bel(B, I, t)$.

Figure 1 shows the decision tree for the agent to decide whether to ProAssert information I to a teammate. Rewards and costs are measured to the whole team. There are two choices P($ProAssert$ with communication cost C_c) and NP ($\neg ProAssert$ with communication cost 0). If agent A chooses P, there are three possible outcomes. The first possible outcome is that $Need(B, I, t')$ holds with probability p and $\neg Bel(B, I, t)$ holds with probability q. In such case the reward to the team is $U(I)$, the utility if information I. The second is that $Need(B, I, t')$ holds with probability p and $Bel(B, I, t)$ holds with probability $(1-q)$. In such case the reward to the team is $U(I) - C_m$, where C_m is a penalty for sending information that B already known. And the third is that $\neg Need(B, I, t')$ holds with probability $(1 - p)$, while the reward to the team also reduces to $U(I) - C_m$. If agent A chooses NP, there are also three possible outcomes. The first possible outcome is that $Need(B, I, t')$ holds with probability p and $\neg Bel(B, I, t)$ holds with probability q. The reward to the team in such case is 0. The second is that $Need(B, I, t')$ holds with probability p and $Bel(B, I, t)$ holds with probability $(1 - q)$, the reward to the team reduces to $-C_n$ since B needs I but A does not pro-assert I. And the third is that $\neg Need(B, I, t')$ holds with probability $(1 - p)$, where the reward to the team is 0 since B does not need I.

The final choice of $ProAssert(A, B, I, t, t')$ or $\neg ProAssert\ (A, B, I, t, t')$ is based on their expected utility. The expected utility EU(P) of P is $U(I) + C_m \times p \times q - C_m - C_c$, while the expected utility EU(NP) of P is $-C_n \times p \times q$. The agent will choose the one with the higher expected utility. Hence, agent A commits to $ProAssert$ iff $EU(P) > EU(NP)$.

We have shown formally in Section 3.2 that there are two types of information needs: *action-performing information needs* and *goal-protection information needs*. The calculation of the expected utilities of ProAssert motivated by these two types of information needs are different. The former assesses the utility $U(I)$ of information I from the utility of action enabled by the communication, whereas the latter uses the negative utility of failing the related goal as C_n for accessing the expected utility of NOT communicating.

4.2 Factors for Decisions about Proactive Communications

The communication cost C_c for performing $ProAssert$, the benefits $U(I)$ of information I, the probability p of $Need\ (B, I, t')$, and the probability q

of $Bel(B, I, t)$ can be computed approximately based on the shared domain knowledge, the structure of team plans, and the progress of teammates on the shared team plans.

Communication Cost We take communication cost as being composed of resource cost (network, energy, etc.) and the potential risks of being overheard by opponents. The resource cost is a constant, while communication risk is dynamically changed, and dependent on domains. Take a battle-field domain as an example, there are situations when a scout communicates with his teammates, his location might be detected by enemies. In these situations the scout will have a higher communication risk, which can be evaluated based on the utility U_S of the scout himself, the probability p_d for him to be detected (may be inverse to the distance between him and the detecting source), and the probability p_k for him to be killed (depends on the distance between him and the closest enemy, and the fighting power of the enemy). Hence, the communication risk will be $C_0 + U_S \cdot p_d \cdot p_k$, where C_0 is the constant resource cost.

Probability of $\boldsymbol{Need(B, I, t')}$ For the action performing-information, a decision-maker could figure out the potential information needers by checking the roles played by its teammates in the current active team plan [19]. Information requirement has been defined for each action type. In a team plan, the doer (or doers for joint action) of an action is either specified explicitly or assigned by means of dynamic agent binding (select actual doers from candidates who are capable to do the action). In the first case, The probability of $Need(B, I, t')$ is either 0 or 1, since it is easy to check if B is one of the needers of information I. In the second case, suppose the number of candidates is Num, the probability of $Need(B, I, t)$ is either $1/Num$ or 0, depending on whether B is a candidate. The set of candidates for an action is contractible as the plan evolves[19], so the probability of $Need(B, I, t)$ is also changed from time to time.

Its a bit more complicated to determine the probability of $Need(B, I, t')$ when I is a goal-protection information, since it's dependent on the domains and contexts. For instance, suppose a scout observed that an enemy fighter is chasing after its team member F_1, and another teammate F_2 with weak fighting power is moving toward the enemy fighter. In this scenario, the scout will believe the enemy fighter has more threat on F_2 than F_1, since F_2 is more likely to be killed. Since the location information of the enemy fighter is critical for F_2 to survive, the probability of $Need(F_2, I, t')$ should be higher than that of $Need(F_1, I, t')$.

Probability of $\boldsymbol{Bel(B, I, t)}$ The probability of $Bel(B, I, t)$ is evaluated based on the observability of the decision-maker, and whether I is static information or not. (1). If decision-maker A has observed B is doing another action that also requires I, A could conclude that B knows I with a higher probability. (2). If A knows B's observability as prior knowledge, and from the current context A believes that B could observe I, then A will assume B knows I with a higher probability. (3). If I is a static information, and A knows it has told I to B, or A has ever received I from B before, the probability of $Bel(B, I, t)$ is 1. (4). In other cases, A could assume the probability of $Bel(B, I, t)$ is very low.

Utility of Information The utility of a piece of information refers to the sum of the gain of knowing it and the lost of unknowing it, both of which are evaluated from the team's perspective. For instance, suppose a scout observes an enemy fighter approaching one of its teammates, the utility of enemy fighter is U_W, and the utility of his teammates is U_F. When the scout tells the location of the enemy fighter to the teammate, let the probability of the teammate's being destroyed by the enemy fighter be p_f, the probability of the enemy fighter's being destroyed by his teammate be p_w. When the scout does not tell the location information to the teammate, let the probability of the teammate's being destroyed be p_e, and the probability of the enemy's being destroyed be p_n. Then, the utility of the location information of can be computed by $(U_W \cdot p_f - U_F \cdot p_w + U_W \cdot p_e - U_F \cdot p_n)$.

To some degree, the utility of information depends on the specific domain problems, and is usually evaluated case by case. The Information that is changing from time to time has short term value, i.e., the utility of such information has to be evaluated whenever the information is considered. The information that is unchangeable or not changed frequently has stable utility, which, in most cases, could be pre-determined offline and only re-evaluated whenever necessary.

5 Experiments

In [20] we reported the experiment results that show teams using proactive information delivery perform much better than teams not using it. To further improve the performance of teams using proactive information delivery, we have extended the CAST architecture with decision-theoretic proactive communication strategies. We have also implemented a test-bed to evaluate different communication strategies for pro-assert.

The test-bed is composed of two opposing agent teams, the blue team and the red team. In a 21×21 grid world, the goal of the blue team is to destroy the home base of the red team, while the red team tries to protect their base by attacking any approaching agents of the blue team. Agents in the blue team could play one of three roles: the scout, who can sense but can not shoot, the fighter, who can shoot but can not sense, and the bomber, who can only bomb the enemy base. The roles of the blue team are designed to maximize their needs for proactive communications. To complete the mission, at least 4 bombers have to surround the enemy base and perform a joint-action called co-fire to the enemy base.

The behavior of the blue team is governed by team plans and individual plans specified in MALLET, along with other related teamwork knowledge. Being informed about the location of the enemy base, the bombers will move toward the enemy base and try to synchronize co-fire action to complete the mission, while the unassigned fighters will also move toward the enemy base to protect bombers whenever needed. When informed about the location of a moving (red team) enemy, a dynamically assigned fighter (based on the team's SMM about the constraint of the assignment) will move toward the enemy's location and shoot at it, while the bombers will try to move away from the threat.

Each agent in the red team can sense as well as shoot. To introduce risks for communications, the enemy base has a communication detection range. If a blue team agent speaks inside the communication detection range, it can be detected. The likelihood of this detection reduces linearly as the distance to the enemy base increases.

We devised two sets of communication strategies for the blue team, which are listed in Table 1. Both strategies adopt a decision-theoretic approach for the scout in the Blue Team to decide whether to proactively inform fighters when the scout detects an enemy agent. The two strategies differ, however, on how they handle the decisions for whether to proactively inform bombers. The strategy $S1$ always informs the closest bomber about the enemy detected so that the bomber can escape. The strategy $S2$ adopts a decision-theoretic approach for deciding on whether to inform the closest bomber about the enemy detected. It should be noted that informing fighters proactively is driven by action-performing information needs, whereas informing bombers proactively is motivated by goal-protection information needs. Hence, this experiment covers both types of proactive communications discussed in previous sections.

Table 1. Comm Strategies used for Team A

Strategy	Inform Fighters	Inform Bombers
S1	Decision-theoretic	always
S2	Decision-theoretic	Decision-theoretic

The following results are based on the following parameter settings of the test-beds. The number of enemy agents is 5, the number of scouts, fighters and bombers in the blue team is 3, 4, and 6 respectively. The sensing range of scout is 6. The shooting range of fighters are 8, with an effect of radius 1.

The sensing range of the enemy agents is 1, while their shooting range is 6 with an effect of radius 1. We randomly generated 50 initial configurations for the locations of agents in both teams. The enemy base is always in the center of the grid.

Table 2. Experimental results for Team A

Communication range	Strategy S1	Strategy S2
4	4	6
5	2	7
6	6	13
7	5	5
8	5	11

Table 2 summaries the number of successfully completed missions (out of a total of 50 missions) for the blue team using the two communication strategies. The independent variable of the experiment is communication detection range of the enemy base. As shown in the table, strategy $S2$ outperforms strategy $S1$. This experiment result suggests that decision-theoretic communication strategies can be effective for team-based agents to decide on whether to proactively deliver needed information to teammates.

6 Comparisons and Conclusions

Communication plays an essential role in the forming, evolving, and terminating of both joint intentions and shared plans. Hence the semantics of communicative actions (performatives) is critical not only for agent designers to understand the requisite mental state prior to the performance of a communicative act by the speaker, but also for the listener to assume certain mental state of the speaker must hold, which facilitates to achieve certain mutual beliefs among them. The semantics of communicative acts are initially defined in terms of beliefs and intentions from the perspective of each individual agent[1, 2, 4], and later from a team's point of view[5, 3, 11]. Compared with Cohen and Levesque's work, in our analysis, an *Attempt* is no longer treated as a complex action expression, but is treated as a view of the current mental state of the performer. Consequently, the semantics of a performative is not only defined in terms of the mental state of the performer, it actually is no more than that.

In the original SharedPlan theory[8, 6, 7], communicative acts are treated implicitly as normal actions, and are raised as help behaviors in establishing requisite mutual beliefs and ensuring the satisfaction of intentions-that. However, the semantics of performatives are missing there. The work in this paper actually shows that the treatment of performative-as-attempt can be successfully carried out in the SharedPlan framework. This is useful on its own, since the joint-intention theory and the SharedPlan theory are proved to be equivalent in specifying the semantics of communication primitives.

Tambe adopted a hybrid approach in implementing the communication mechanisms for STEAM[16]. In STEAM, Communication is mainly raised (implicitly) from the prescriptions of joint intentions, while additional communication is generated by checking the explicit declaration of information-dependency relationships among domain actions. Our work in this paper, however, is focusing on giving semantics for the underpinning communicative acts in the framework of SharedPlan theory, by which we hope all the communicative actions among a team could be raised explicitly and uniformly as a kind of help behaviors.

In [13], a communication paradigm was proposed for periodic team synchronization (PTS) domains with only a single, unreliable, low-bandwidth communication channel for agents that might belong to adversary teams. However, they are more concerned about dynamic team formation in a class of PTS domains, while this paper is focusing on the semantics of communication acts, and how they are adopted as help behaviors in supporting team activities. No doubtly, the treatment of communication acts as help behaviors is useful in dynamic team formations.

In addition, we introduced a decision-theoretic approach for reasoning about whether to act on a potential proactive assert. Experimental results suggested that the decision-theoretic communication strategy enhances the team performance. The formal semantics and the decision-theoretic communication strategies together provide a sound and practical framework that enables further studies regarding proactive information delivery for supporting the decision making of a team involving human and agents.

7 Acknowledgments

We wish to thank Dr. Tom Ioerger for earlier discussions on topics related to this paper. We are also in debt to Ryan Rozich for developing a grid world simulator for our experiments. This research has been supported by AFOSR MURI grant No. F49620-00-1-0326.

References

1. P. R. Cohen and H. J. Levesque. Performatives in a rationally based speech act theory. In *Proceedings of the 28th Annual Meeting of the Association for Computational Linguistics*, pages 79–88, 1990.
2. P. R. Cohen and H. J. Levesque. Rational interaction as a basis for communication. *Intentions in Communication*, pages 221–255, 1990.
3. P. R. Cohen and H. J. Levesque. Teamwork. *Nous*, 25(4):487–512, 1991.

4. P. R. Cohen and H. J. Levesque. Communicative actions for artificial agents. In *Proceedings of the International Conference on Multi-Agent Systems*. AAAI Press, June 1995.
5. P. R. Cohen, H. J. Levesque, and S. I. A. On team formation. *Contemporary Action Theory*, 1997.
6. B. Grosz and S. Kraus. Collaborative plans for complex group actions. *Artificial Intelligence*, 86:269–358, 1996.
7. B. Grosz and S. Kraus. The evolution of sharedplans. *Foundations and Theories of Rational Agencies*, pages 227–262, 1999.
8. B. Grosz and C. Sidner. Plans for discourse. *Intentions in communication*, 1990.
9. E. J. Horvitz, J. S. Breese, and M. Henrion. Decison theory in expert systems and artificial intelligence. *International Journal of Approximate Reasoning*, 2.
10. R. A. Howard. Information value theory. *IEEE Transaction on Systems Science and Cybernetics*, SSC(2):22–26, 1966.
11. S. Kumar, M. J. Huber, D. R. McGee, P. R. Cohen, and L. H. J. Semantics of agent communication languages for group interaction. In *Proceedings of 17th National Conference on Artificial Intelligence (AAAI 2000)*, pages 42–47. AAAI Press/The MIT Press, July 2000.
12. D. Pynadath and M. Tambe. Multiagent teamwork: Analyzing the optimality and complexity of key theories and models. In *Proceedings of the First Autonomous Agents and Multiagent Systems Conference (AAMAS)*. ACM Press, July 2002.
13. P. Stone and M. Veloso. Task decomposition, dynamic role assignment, and low-bandwidth communication for real-time strategic teamwork. *Artificial Intelligence*, 110:241–273, 1999.
14. K. Sycara and M. Lewis. Forming shared mental models. In *Proc. of the 13th Annual Meeting of the Cognitive Science Society*, pages 400–405, 1991.
15. K. Sycara and M. Lewis. Modeling teams of specialists. In *Proceedings of the 27th Hawaii International Conference on System Sciences*. IEEE Society Press, January 1994.
16. M. Tambe. Towards flexible teamwork. *Journal of Artificial Intelligence Research*, 7:83–124, 1997.
17. J. Yen and X. Fan. The semantics of proactive communication acts among team-based agents. In *Proc. of the 14th IEEE International Conference on Tools with Artificial Intelligence (to appear)*, November 2002.
18. J. Yen, X. Fan, and R. A. Volz. On proactive delivery of needed information to teammates. In *Proc. of the Workshop on Teamwork and Coalition Formation at AAMAS'02*, pages 53–61, July 2002.
19. J. Yen, J. Yin, T. Ioerger, M. Miller, D. Xu, and R. Volz. Cast:colaborative agents for simulating teamworks. In *IJCAI'2002*, pages 1135–1142, 2001.
20. J. Yin, M. S. Miller, T. R. Ioerger, J. Yen, and R. A. Volz. A knowledge-based approach for designing intelligent team training systems. In *Proceedings of the Fourth International Conference on Autonomous Agents*, pages 427–434, 2000.