Available online at www.sciencedirect.com
SCIENCE DIRECT
Artificial Intelligence 169 (2005) 23–97
Artificial Intelligence
www.elsevier.com/locate/artint

A theoretical framework on proactive information exchange in agent teamwork

Xiaocong Fan [a,*], John Yen [a], Richard A. Volz [b]

[a] School of Information Sciences and Technology, The Pennsylvania State University, University Park, PA 16802, USA
[b] Department of Computer Science, Texas A&M University, College Station, TX 77843, USA

Received 16 November 2004; received in revised form 23 May 2005; accepted 16 June 2005

Available online 10 August 2005

Abstract

Proactive information delivery is critical to achieving effective teamwork. However, existing theories do not adequately address proactive information delivery. This paper presents a formal framework for proactive information delivery in agent teamwork. First, the concept of information need is introduced. Second, a new modal operator, InfoNeed is used to represent information needs. The properties of the InfoNeed operator and its relationships to other mental modal operators are examined, four types of information needs are formally identified, and axioms for anticipating the information needs of other agents are proposed and justified. Third, the axiom characterizing chains of helpful behavior in large agent teams is given. Fourth, the semantics for two proactive communicative acts (ProInform and 3PTSubscribe) is given using a reformulation of the Cohen–Levesque semantics for communicative acts in terms of the SharedPlans formalism of Grosz and Kraus. The work in this paper not only provides a better understanding of the underlying assumptions required to justify proactive information delivery behavior, but also provides a coherent basis for the specification and design of agent teams with proactive information delivery capabilities.

Keywords: Proactive information delivery; Information needs; Agent teamwork; Shared mental model; Communicative acts

* Corresponding author.
E-mail addresses: zfan@ist.psu.edu (X. Fan), jyen@ist.psu.edu (J. Yen), volz@cs.tamu.edu (R.A. Volz).

0004-3702/$ – see front matter
doi:10.1016/j.artint.2005.06.005

1. Introduction

Proactivity refers to the ability to take initiatives, make conscious decisions, and take positive actions to achieve chosen goals. Proactivity is taken to be one of the key characteristics of software agents [98]. Proactive agents cannot only respond to external stimuli in a timely manner, but they can also deliberate on, choose among, and act upon the possibilities created by the stimuli.

Proactivity is especially critical for teamwork in business management [28], psychology [48] and artificial intelligence. Kirkman and Rosen [53] found that proactive teams are more effective than less proactive teams; Jauch and Kraft [49] and LaPorte and Consolini [60] showed that organizations with proactive agents generally outperform those with only reactive agents, especially in how efficiently they respond. The effects of different types of agent proactivity on organizational decision-making performance also have been studied [63], and it is found that proactive behavior becomes more critical for organizational success as work becomes more dynamic and decentralized [27].

There are various kinds of proactive behaviors in teamwork. One of these, helping behavior, is of particular interest in the literature [74,99]. In the multi-agent system (MAS) field, helping behavior can be illustrated in several existing formal frameworks. For instance, from the viewpoint of the theory of Joint Intentions [23,25], helping behavior occurs whenever a team member helps another with his/her responsibilities in order to achieve the goals to which they are committed. The SharedPlans theory even has axioms for specifying helping behaviors [42,43].

On the other hand, effective teamwork relies on communication. Communication plays an essential role in dynamic team formation [88], in maintaining shared situation awareness [94], in coordinating team activities [91,99], and more theoretically, in the forming, evolving, and terminating of both joint intentions [25] and shared plans [43].

This paper centers on a communication-related helping behavior—proactive information delivery, by which we mean "providing relevant information without being asked". One motivation of our study of proactive information delivery in the context of teamwork is that it is widely recognized that the "ask/reply" approach, although useful and even necessary in many cases, does have limitations; and proactive communication may provide a complementary solution [27,99]. For instance, an information consumer in a team—whether a human or software agent—may not realize that certain information it has is already out of date. If this agent had to verify the validity of every piece of information before using it (e.g., in decision making), the team could be easily overwhelmed by the amount of communication entailed by these verification messages. Proactive information delivery offers an alternative, as it shifts the burden of updating information from the information consumer to the information provider, who typically has direct knowledge about any changes. In addition, an agent, due to its limited knowledge, may not realize that it needs certain information. For instance, a piece of information may be obtained only through a chain of inferences (e.g., being fused according to certain domain-related rules). If the agent does not have all the knowledge needed to make such a chain of inferences, it simply cannot realize that it needs the information, and thus does not know enough to request it. Proactive information delivery allows teammates to assist the agent in such a circumstance.

In fact, to overcome the above-mentioned limitations of ‘ask’, many human teams incorporate proactive information delivery in their planning. For example, in BattleSpace InfoSphere [94], an echelon unit’s plan (e.g., the operational order for an army brigade) often anticipates critical decisions the commanders need to make and specifies information needed to make these decisions in the plan. These decisions are called “decision points” in the operational order, and the information needs of the commander are called “Commander’s Critical Information Requirements” (CCIR). Based on a CCIR, the unit’s intelligence officer and scouts are able to proactively deliver relevant information to the commander. We believe this approach adopted by human teams provides critical evidence that software agents could also benefit from being equipped with proactive information delivery capabilities.

Inter-agent communication has been studied extensively [34]. For instance, many researchers have been studying agent communication languages (ACL) that agents in distributed computing environments use to share information. KQML [59] and FIPA’s ACL [1] are two attempts toward a standardized ACL. The mental-state semantics of ACL is one of the most developed areas [11,57], where most efforts are based on Cohen and Levesque’s work [22]. However, aside from the open issues in ACL verification, ontology integration, and conversational semantics [17], we claim that proactive information delivery behavior cannot be elegantly captured by the existing ACL performatives. In other words, the existing ACLs [1,59] are not expressive enough to represent proactive communications among agents.

In the rest of this section, we trace existing research on proactive information delivery behavior from different disciplines. In particular, we examine the indirect speech acts in discourse theory, psychological studies in human teamwork, helping behavior originated from maintaining SharedPlans, and information-pushing technology. This review enables us to identify the key issues that a theory about proactive information delivery should address.

1.1. Proactive information delivery in discourse theory

Proactive information delivery behavior was first recognized in the 1970s by researchers in the field of human discourse understanding [3,4,64] in their studies of indirect speech acts [80]. Indirect speech acts are those that appear to mean one thing yet are treated as though they mean something else. Based on a plan-recognition model of the language-comprehension process [3], Allen explained why a hearer could generate helpful responses that convey more information to the speaker than was explicitly requested. Here, the proactivity relies on the hearer’s ability to analyze direct and indirect speech acts, to infer the speaker’s plans and then to detect obstacles in those plans. It is claimed that “many instances of helpful behavior [in discourse] arise because the observing agent recognizes an obstacle in the other agent’s plan and acts to remove the obstacle”. For instance, sentence A in the following session is a typical indirect speech act:

A: *When is the train to Hamilton leaving?*
B: *That train was canceled.*

The explicit obstacle to A is "the leaving time of the train to Hamilton". However, knowing well that the train was canceled, B may want to notify A regarding other obstacles, especially ones that A is not aware of. Here, B provides a helpful response (without being directly asked) because B believes that A intends to go to Hamilton today and believes that A does not know that today's train has been canceled. In essence, B provided extra information based on B's inference of A's goal and A's plan for achieving that goal.

1.2. Proactive information delivery in human teamwork

Human teamwork depends on handling and sharing information [90]. Researchers have sought to understand the potential relationships between information and team performance [27,61]. Team members typically tend to proactively seek new information to achieve their joint goals [45]. Some psychological studies about high-performing teams have identified the ability to proactively offer information needed by teammates as one of the key characteristics of effective teamwork [32,68,86].

Proactive information delivery occurs more frequently when human teams need to filter and fuse an overwhelming amount of information and to make critical decisions under time pressure. For instance, applications for dynamic domains such as BattleSpace InfoSphere [94] often require a large number of intelligent agents and human agents to form a team to cooperate effectively in information gathering, information fusion and information delivery for making better group decisions.

It is well recognized that helping behavior in human teams is enabled by some "overlapping shared mental models" that are developed and maintained by members of the team [16,74,76,78,89]. It is also shown that the hypothetical cognitive construct of shared mental models could explain certain coordinated team behaviors [14,15,92].

1.3. Proactive behavior in the SharedPlans theory

Another significant thread of research in human dialogues is to explain certain properties of discourse using the notion of SharedPlans. In this view, the participants in a discourse mutually believe they are working toward establishing the beliefs and intentions that are necessary for one to say that the participants have a shared plan [44]. In this study, proactive behavior is implicitly captured in a 'conversational default rule' (CDR2), which states that an agent in a group will adopt an intention to do an action if the performance of the action would contribute to the achievement of the group's joint goal. Proactive information delivery can thus be taken as one reification of this schematic rule with appropriate communicative actions as the substitutes.

Using SharedPlans to explore the proactivity in human discourse can be traced to Lochbaum's work [65–67], where she showed that the SharedPlans theory provides a more detailed account of an agent's motivations for an utterance or initiation of a discourse. In her model, each segment of a discourse is understood in terms of a shared plan corresponding to the purpose of that segment, and the utterances of a segment are understood as the participating agents' contribution toward the completion of the shared plan. The objective of these shared plans is to let action performers acquire knowledge (e.g., recipes and values for action parameters) necessary to perform their actions. Proactive information delivery is embodied in utterances leading to the completion of these shared plans. In particular, an agent's reflection that there is a lack of knowledge about an action to be performed initiates an information-seeking dialogue. The hearer, knowing the speaker's information needs, tries to help by providing relevant information. While the proactive information delivery revealed by Lochbaum only lies in the information-seeking dialogues regarding knowledge preconditions, her approach of using the SharedPlans theory could be extended to also cover proactivity involving physical preconditions and other constraints. The major limitation lies in the weak notion of proactivity: it does not address the proactive nature of providing information without being asked.

1.4. Information pushing

Information pushing [29], which has been widely adopted by Web-based information services, refers to the behavior of delivering information to a user based on a personalized profile specific to that user. Information pushing is certainly related to proactive information delivery because an effective user profile typically defines what information is needed and when it is needed. Information is delivered to a user if and only if it fits the personalized criteria set by the user. Among others, the criteria could include complicated and dynamic metrics to ensure that users are not "spammed" [2]. The metrics can also be automatically updated or learned from users' behavior patterns.

Flexibility and rationality are critical to applications using information pushing technology. It is noted that most of the systems broke down when users tried to go beyond the predefined information needs [2]. Thus, the assumption that all information needs in dynamic settings can be defined in advance is simply wrong. On the other hand, intelligent pushing is desirable because presenting too much information would lead to cognitive overload. In that case, users are forced to take into account information they already know or consider irrelevant.

1.5. The objective and desiderata

We first characterize the agent teams to which our proposed theory will apply. The teams we are considering share the following: (1) they have distributed expertise, so team members need to exchange information; (2) they are working under pressure and need to deliver information in a timely manner; (3) communication is limited for various reasons, further necessitating selective information exchange; and (4) the team has to filter, fuse, and interpret overwhelming amounts of information. These characteristics are common for many teams in the real world.

The objective of developing a theory about proactive information delivery is three-fold. First, the theory should provide a guide for the specification and design of agent architectures, algorithms, and applications that support proactive information delivery capabilities. Second, the theory will shed light on the mental states of the performers in proactive communication actions, as well as uncover the limitations and necessary assumptions of proactive information exchanges in a multi-agent system. Third, the theory offers opportunities for exploiting novel agent communication protocols that support proactive teamwork behaviors.

However, such a theory cannot be directly derived from any of the existing frameworks described above for the following reasons. Allen's work relies too much on interpreting preceding utterances, psychological studies give little insight on how to build computational models for proactive information delivery, the SharedPlans theory lacks enough support for reasoning about and acting upon teammates' information needs, and information pushing is too much limited and typically unidirectional. We next describe each in more detail.

Allen's observations of how a lack of information in human discourse can elicit helping behavior [3] certainly shed light on the study of proactive information delivery behavior in multi-agent teamwork settings. Based on Allen's studies, the following issues are critical to establishing a theory for proactive information exchange: (1) relevance: proactive behavior should be directed toward an addressee's goal; (2) shared knowledge: the participants need to have certain shared knowledge to recognize each other's plans; (3) intentional semantics: the speaker's mental attitudes, as expressed through speech acts, can affect those of the addressee.

However, proactive information delivery becomes more convoluted in the agent teams described above. First, in human dialogues, indirect speech acts can be understood by considering the *idiomatic* meaning behind the literal meaning, by using inference schema (i.e., to rate the potential choices by heuristics or inference rules), or by using background/context knowledge to infer others' intentions [3,12]. When modeling proactive information delivery in large agent teams or teams mixed with human and software agents, more subtle issues need to be considered, such as the level of abstraction, shared mental states, computational complexity. Second, Allen's work relies heavily on the audience's recognition, based on certain rules and heuristics, of the speaker's intentions and plans [6]. While modeling discourse understanding as plan recognition is reasonable for human discourse, it is not practical for large agent teams because each member needs to recognize teammates' plans which may have numerous alternatives [22].[1] Agents in a large team can easily diverge in anticipating a certain team member's intentions due to the difficulty of matching teammates' inferences. Such a divergence may impact team performance and even inhibit the team's achievement of its joint goals. Third, the distinction between indirect and direct speech acts in human discourse is no longer that important for teams facing overwhelming amounts of information under time pressure. More likely, proactive information delivery is triggered by an agent's anticipation of teammates' needs *without* any preceding conversation, rather than triggered by the agent's understanding of the implicit meaning of preceding speech acts (e.g., as in the ask/reply mode, also known as the *master-slave assumption* [44]).

Even though psychological studies [15,16] have shown that members of high performing human teams often offer relevant information to teammates before they ask, it is difficult from these empirical studies to derive a general formalism for proactive information delivery behavior that also applies to agent teams. However, the studies do suggest that the anticipation of information needs and shared awareness of team activities are critical constructs of theories about proactive information delivery.

[1] It is still computationally hard even if the *intended recognition assumption* [19] is adopted.

The SharedPlans theory seems promising because it does support the shared awareness of team activities. In addition, the SharedPlans theory has axioms for deriving help behaviors [42,43]. Being slightly extended, it can also cover proactive information delivery behavior in agent teams. However, this view is not particularly satisfying. To the extent possible, we would like to establish a framework where information needs can be treated as a first-class notion so that agents can reason about and act upon teammates' information needs. The proactivity in Lochbaum's work [67] relies on discourse understanding. We believe this suffers from the same limitations as Allen's work [3]: it requires the information provider to infer the speaker's information needs from the *preceding* utterances. What is more important in proactive information delivery is that an agent should rationally anticipate teammates' information needs and *push* the relevant information in a timely manner. In a study by Grosz and Sidner [44], one of the two-person discourses implies information can be delivered without being requested. However, the utterances are more like orders and clarifications from a commander.

Although proactive information delivery and personalized information pushing are similar in that they both send information to an information consumer in a proactive way based on the anticipation of his/her information needs, they differ in several aspects. For instance, the former requires a more abstract and broader understanding of the information consumer's needs (e.g., a shared awareness of the team goals, the planned team activities or each other's roles and responsibilities). Also, proactive communications are bi-directional in a team whereas personalized information pushing is only from the computer to the user. On the other hand, proactive information delivery can be viewed as a general extension of personalized information pushing in the context of teamwork. Thus, feasibility and rationality, as suggested by the practice of information pushing technology, should be considered in developing theories for proactive information delivery. In particular, such a theory should support dynamic reasoning about information needs (e.g., activate/deactivate information needs when an agent switches its attention from one activity to another) and allow decision making on whether to provide help.

To summarize, a theory about proactive information delivery in agent teams should address at least three issues. First, the concept of "information needs" should be treated as a first-class notion. The properties of information needs and its possible relationships to agents' mental attitudes should be examined. Also information needs should be relativized to certain contexts in order to support dynamic reasoning. Second, the theory should allow an agent to anticipate teammates' information needs based on logical axioms, assumptions, heuristic rules, or approximate reasoning. Such anticipation may demand the modeling of shared team activities (e.g., team processes) and nested epistemic states (e.g., one's belief about teammates' beliefs). Third, the theory should connect information needs to proactive communications. The connection should be intuitively simple while flexible enough so that agents can make the final decisions on whether and how to communicate. The theory also should define appropriate intentional semantics for proactive communicative actions (henceforth, *proactive performatives*) in terms of information needs and other mental attitudes. In addition, conversation protocols involving proactive performatives should be covered in order to investigate how information needs as well as information are exchanged proactively in agent teams.

The rest of the paper is organized as follows. In Section 2, we elicit the methodology for building our theory and establish the base layer. We give some preparation in Section 3. Specifically, we discuss the assumptions about mental attitudes, the composition of contexts, the properties of collaborative agents, the structure of inference knowledge, and the re-formulation of the Cohen–Levesque semantics of communicative acts.

In Section 4, we elucidate the concept of information need. In particular, we use reference expressions to represent information and information needs; introduce a modal operator InfoNeed to express information needs, examine the properties of InfoNeed and explore its relationships with other mental modal operators; analyze levels of information needs based on the idea of social inference trees; and formally identify four types of information needs prevalent in agent teamwork. In Section 5, we propose and justify axioms for anticipating others' information needs, and in Section 6, we give an axiom which relates information needs with potential intentions. These axioms together allow agents in a team to take appropriate actions to satisfy the anticipated information needs.

In Section 7, we formally define the semantics of ProInform(proactive inform), explore the potential composition of the context of proactive performatives, give a conversation protocol involving ProInform, and prove some properties related to ProInform. Similarly, in Section 8, we formally define the semantics of 3PTSubscribe(proactive third-party subscribe), give a conversation protocol involving 3PTSubscribe, and prove some properties related to 3PTSubscribe.

In Section 9, we discuss the role of agent observability in approximately modeling teammates' belief states and point out some potential implications of the presented theory. We compare our theory with related work in Section 10, and summarize the paper in Section 11.

2. Methodology and the base layer

Among others [52,97], the Joint Intentions theory [23,62] (henceforth, JIT) and the SharedPlans theory [42,44] (henceforth, SPT) are two widely accepted formalisms for modeling teamwork. Each has been successfully applied in guiding the design and implementation of multi-agent systems, such as GRATE* [50], STEAM [91], COLLAGEN [77] and CAST [99].

SPT provides an axiomatic theory of collaborative plans based on four types of intentional attitudes: Int.To, Int.Th, Pot.Int.To, and Pot.Int.Th. A shared plan is characterized in a mental-state view as a particular collection of beliefs and intentions. A group of agents have a shared plan if and only if they hold the specified beliefs and intentions. Thus, collaboration typically involves agents trying to establish and maintain those required mental attitudes, and each believes the other agents are doing likewise. A shared plan is associated with an action decomposition hierarchy. In the process of constructing a shared plan, a group of agents and various subgroups need to make numerous decisions on reconciling potential intentions, on choosing parameters for actions, on selecting recipes (courses of action) and on assigning agents or subgroups to actions at every level of the evolving decomposition hierarchy. Hunsberger proposed using SharedPlans Trees to explicitly represent the choices already made by a group working on some shared plan [47]. SPT, allowing the evolution of shared plans from partial plans in a distributed fashion, offers a general approach to group planning.[2]

JIT generalizes the belief-goal-commitment model of agent mental states [20] by proposing a notion of joint intentions, from which individuals derive their own intentions. Compared with SPT, JIT embodies a stronger dependency on communication: Whenever agents realize a joint goal is satisfied, becomes unachievable or irrelevant, they are required to inform one another of the achievement or impossibility of the joint goal before abandoning the joint commitment. Due to such a strong requirement on communication in establishing and maintaining joint intentions, Cohen and Levesque introduced *speech act theory* into their framework [21,22,24,25]. Their idea of performative-as-attempt has been widely accepted as a standard in ascribing mentalistic semantics to communicative acts [1,46,55,56].

In JIT, individual intentions are represented by INTEND1 and INTEND2,[3] which correspond to Int.To and Int.Th in SPT, respectively. One difference between these two notions of intentions is that they embody different degrees of commitment. INTEND1 and INTEND2 employ a strong notion of commitment: an agent commits to its persistent goal until it believes the goal is satisfied, believes it is unachievable, or believes the relativization condition is false. In SPT, however, intentions (Int.To, Int.Th) entail commitment in a weaker sense: an agent may drop intentions for a variety of reasons [42].

A certain "cohesive force" is needed for establishing, monitoring and disbanding joint activities [25]. The concepts of joint intentions and shared plans serve as such a cohesive force in JIT and SPT, respectively. But whatever the cohesive force is, it should be fostered and maintained by all team members in their pursuit of the team goals. For this reason, agents having a joint intention cannot act knowingly to foil the fulfillment of the joint intention; likewise, agents having a shared plan must reconcile potential intentions prudently so that agents will not ruin the already adopted intentions and thereby inhibit the completion of the shared plan.

While JIT and SPT are able to capture many important behaviors of agent teamwork, both of them exhibit certain limitations. For instance, JIT considers actions and plans only at a high-level without considering the decomposition of complex actions at different levels of abstraction. JIT also does not capture how agents elaborate upon their partial joint and individual plans [25]. Compared with JIT,[4] SPT explores the hierarchical structure of shared plans and addresses partiality in a significant way. However, SPT still lacks an adequate semantics for the "potential intention" operators and the "intentional context" terms [42] (but see Fan and Yen's recent attempt [37] for the semantics of potential intentions). Furthermore, it may be problematic that the definition of Int.Th does not link to the primitive Commit operator [25]. But on the other hand, the respective weakpoints of the two theories may be attributed to their different emphases: JIT focuses on investigating the

[2] Such a mental-state view of plans originated from Bratman [10].

[3] (INTEND1 A α C) represents that agent A intends to do action α relative to condition C. (INTEND2 A p C) represents that agent A intends that p hold relative to condition C [20].

[4] One subtle difference between SPT and JIT is that JIT relies on a notion of irreducible collective intentionality while SPT adopts an "individualistic" approach to collective intentionality (similar to Bratman's) in that it requires no irreducible notion of "we-intention [93]".

need for maintaining a joint commitment while SPT centers on the process of evolving shared plans and the treatment of partiality through agents' means-ends reasoning. In such a sense, they complement each other. In fact, some implemented agent systems, such as STEAM [91] and RETSINA [39], benefit from both.

An important consequence of this brief investigation of SPT and JIT is the opportunity to rethink the issues we identified earlier, regarding the development of a theory of proactive information delivery, and to deliberate on the methodology for doing so. As generic theories for agent teamwork, neither JIT nor SPT can be directly used to characterize proactive information delivery because both lack the concept of information needs, which we believe is a key ingredient in a theory of proactive communication. However, SPT and JIT do offer a basis for developing a theory of proactive information delivery. The specific elements of JIT useful in developing such a theory include (1) the formal treatment of speech acts, which establishes a sound foundation for intention-based semantics of performatives and, as we will show, proactive communicative acts; and (2) the motivation behind communication related to dropping joint intentions, which helps in inferring others' information needs. The specific elements of SPT that can be leveraged include (1) the rich and clean model of shared team plans,[5] which is critical in enabling agents to anticipate teammates' information needs; (2) the decomposition hierarchy of shared recipes, which encourages selective information exchange: only that subgroup of agents selected to work on an activity need know the top-level information of the plan subtree associated with the activity [47].

Thus, instead of starting from scratch, the above observation persuades us to develop a layered formalism built on top of JIT and SPT. There are two options to do so. We can stay within JIT, or alternatively stay within SPT, by introducing a notion of information need and axioms for deriving helping behaviors and borrow the necessary features from the other. If we stay within Cohen and Levesque's formalism (JIT), the semantics of proactive performatives can be defined after the notion of information needs is introduced. However, it is difficult for agents to anticipate others' information needs within JIT. To address this, the formalism can be extended to allow agents to hierarchically recognize teammates' active plans. We choose to extend Grosz and Kraus's SharedPlans theory (SPT) and translate the idea of 'performative-as-attempt' from JIT to SPT (as shown in Fig. 1) for three reasons: (1) the hierarchical expansion of shared plans allows agents to infer others' potential intentions, which in turn allows agents to anticipate others' possible information needs; (2) even though helping behavior follows smoothly from both JIT and SPT, SPT actually provides axioms for deriving helping behavior. If extended, they can also be used to characterize chains of helping behaviors; (3) SPT does not provide a semantics for performatives analogous to that provided by JIT. The lack of a formal grounding for performatives can discourage the in-depth studies of communication among SPT-based agents.

To make the paper self-inclusive, we first summarize the key concepts underlying the base layer before considering proactive information delivery in later sections.

[5] The richness originates from the treatment of partiality and the cleanness from distinguishing between intention types.

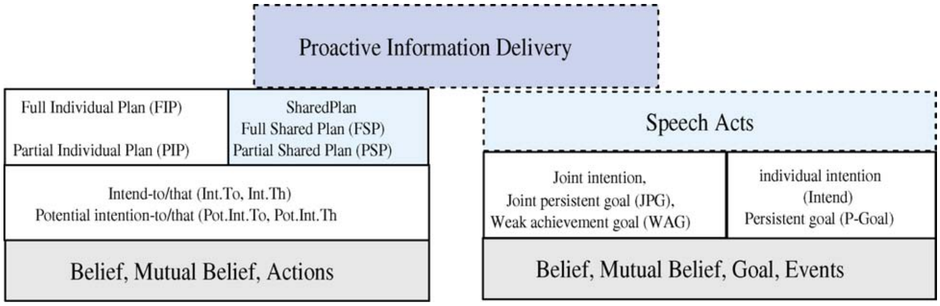

Fig. 1. The two layers of our framework.

2.1. The base layer

2.1.1. Basics of the SharedPlans theory

The SharedPlans formalism of collaborative planning originated from Pollack's mental state model of plans [73] and was further extended to accommodate partial plans and the evolution of shared plans over time [42,43].

We adjust some notations in the SharedPlans theory [42] for our convenience. We use $A, B, \ldots, A', \ldots$ to refer to individual agents; use $\alpha, \beta, \ldots$ (ϵ and its primes are used specifically to refer to utterance events) to denote act-types; use t with superscripts or subscripts to denote time points (by default, t refers to the current time point);[6] use R with an act-type as its subscript to denote a recipe for that type of actions (a recipe is composed of action decomposition and constraints); use C with a subscript, such as C_α,C_p, to refer to an intentional context, and use Θ with an act-type as its subscript to denote constraints for that type of actions. The components of a constraint may be classified into three types: execution preconditions, recipe-constraints (e.g., time, location or other resources considered in the selection of recipes for the action), and constraints considered in reconciling potential conflicts. The composition of a context will be discussed later.

Shared plans are defined in terms of modal operators, meta-predicates (i.e., abbreviations for complex formulas involving other predicates or modal operators) and actions. In addition to Bel (belief) and MB (mutual belief), three modal operators are used to relate agents and actions (Exec, Commit, and Do), and four modal operators are used to specify the attitudes of intention (Int.To and Pot.Int.To apply to actions while Int.Th and Pot.Int.Th apply to propositions). $\mathsf{Exec}(A, \alpha, t, \Theta_\alpha)$ represents that agent A has the ability to perform basic-level action α at time t under the constraints Θ_α; $\mathsf{Commit}(A, \alpha, t_1, t_2, C_\alpha)$ represents the commitment of agent A at t_1 to perform the basic-level action α at t_2 under the context C_α; $\mathsf{Do}(A, \alpha, t, \Theta_\alpha)$ represents that an agent (or a group of agents) A performs action α at time (beginning at, in the case of an interval) t under constraints Θ_α.

$\mathsf{Int.To}(A, \alpha, t, t_\alpha, C_\alpha)$ represents that at time t, agent A intends to do α at time t_α in the context C_α. Int.To stimulates means-end reasoning. When the action that an agent in-

6 Time is treated as an ordered set of discrete points. We assume primitive actions performed at time t will be done by the next time point. The performance of a complex action may span several time points.

tends to do is a basic-level action (and the agent does have the ability of doing that action), Int.To reduces to Commit; otherwise, the agent will try to compose a recipe for the action before doing it. Int.Th(A, p, t, t', C_p) represents that agent A at time t intends that p hold at t' under the intentional context C_p. While intentions-to only apply to individual agent actions, intentions-that can be used to initiate team activities involving a group of cooperators. In fact, Int.Th plays an essential role in meshing subplans, helping teammates, and reconciling resource or intention conflicts. Pot.Int.To$(A, \alpha, t, t_\alpha, C_\alpha)$ represents that agent A has a potential intention to do α. Int.To is used to represent goals to which agents are fully committed, while Pot.Int.Tos refer to possible goals to which agents are not yet fully committed. An agent may convert a Pot.Int.To to an Int.To if the potential intention does not contradict the already adopted intentions. A Pot.Int.To has to be dropped should there be any conflicts. Similarly, Pot.Int.Th(A, p, t, t', C_p) refers to a potential intention-that. A Pot.Int.Th needs to go through a similar deliberation process before it can be adopted as a full-fledged intention.

Several meta-predicates were defined. Among others, CBA$(A, \alpha, R_\alpha, t_\alpha, \Theta_\alpha)$ represents that agent A is able to bring about single-agent action α at t_α under constraints Θ_α by following recipe R_α. CBA represents the knowledge an agent has about its ability to perform an action in a plan. Meta-predicate CONF is used to represent actions/propositions conflicts. In particular, CONF$(\alpha, p, t_\alpha, t_p, C_\alpha, C_p)$ represents the situation in which the performance of action α conflicts with p continuing to hold. Shared mental states among a team of agents are reflected in their partial shared plans (denoted by PSP) or full shared plans (denoted by FSP).

Grosz and Kraus proposed several axioms that can be used to derive helpful behaviors [42,43]. For instance, they gave two axioms (A5 and A6, [42]) which state that an agent will form a potential intention to do *all* the actions it thinks might be helpful.[7] Later, they gave another axiom (Axiom 2, [43]) which states that if an agent intends-that p hold and there exist some alternative actions the agent can take that would lead to p holding, then the agent must be in one of three potential states: (1) the agent holds a potential intention to do some of these actions; (2) the agent holds an intention to do some of these actions; or (3) the agent has reconciled all possible actions it could take and determined they each conflict in some way with other intentions [43]. In essence, the two treatments [42,43] are consistent; they characterize "intending-that" from two perspectives. The former shows how Pot.Int.Tos are triggered from Int.Ths. The latter reflects the process of means-ends reasoning: an agent first adopts a Pot.Int.To, then reconciles it with existing intentions—either adopting it as an actual intention, or dropping it to consider other options. In both treatments, it could be the case that all Pot.Int.Tos serving the same ends have been tried but none can be reconciled into actual intentions.

[7] Probably, this is one subtle point where "potential intentions" differ from what Cohen and Levesque term an agent "goal". It is too strong to require an agent to adopt a goal (i.e., chosen desire [20]) to do *all* the alternative actions that serve the same ends. But this is acceptable for potential intentions because turning a potential intention into a full-fledged intention requires some preliminary means-ends reasoning [42]. An agent can drop all the other potential intentions serving the same ends after the agent successfully reconciles a potential intention into an intention. One consequence is that an agent cannot hold conflicting goals but can hold conflicting potential intentions.

Axiom 1. $\forall A, p, t, \beta, t_\beta, t' > t_\beta, C_p\cdot$
$\mathsf{Int.Th}(A, p, t, t', C_p) \wedge \neg\mathsf{Bel}(A, p, t) \wedge \mathsf{LEAD'}(A, \beta, p, t, t_\beta, \Theta_\beta) \Rightarrow$
$\mathsf{Pot.Int.To}(A, \beta, t, t_\beta, \Theta_\beta \wedge C_p).$

Fig. 2. The axiom characterizing helping behavior.

Grosz and Kraus [43] defined a LEAD predicate, where $\mathsf{LEAD}(A, \beta, t_\beta, p, \Theta_\beta, t)$ represents that agent A doing action β at time t_β under constraints Θ_β leads to proposition p holding. Because we are only interested in communicative acts, which are assumed to be single-agent acts, we now define LEAD'—a slightly scaled-down version of LEAD in which β represents a single-agent, communicative act. $\mathsf{LEAD'}(A, \beta, p, t, t_\beta, \Theta_\beta)$ holds iff (1) agent A believes there exists a recipe it can follow to do action β, and (2) either β directly leads to p holding, or the doing of β 'leads to' another agent's being able to do (CBA) some action α, which directly leads to p holding.

Definition 1. $\mathsf{LEAD'}(A, \beta, p, t, t_\beta, \Theta_\beta) \triangleq \mathsf{Bel}(A, P1, t) \wedge$
$[\mathsf{Bel}(A, P2, t) \vee \mathsf{Bel}(A, P3, t)]$, where
$P1 = \exists R_\beta \cdot \mathsf{CBA}(A, \beta, R_\beta, t_\beta, \Theta_\beta),$
$P2 = (\mathsf{Do}(A, \beta, t_\beta, \Theta_\beta) \Rightarrow p)$, and
$P3 = \mathsf{Do}(A, \beta, t_\beta, \Theta_\beta) \Rightarrow \psi$, where
$\psi = [\exists B, \alpha, R_\alpha, t_\alpha, t''\cdot (t_\alpha > t_\beta) \wedge (t_\alpha > t'') \wedge$
$\quad \mathsf{CBA}(B, \alpha, R_\alpha, t_\alpha, \Theta_\alpha) \wedge$
$\quad\quad \mathsf{Pot.Int.To}(B, \alpha, t'', t_\alpha, \Theta_\alpha) \wedge$
$\quad\quad\quad (\mathsf{Do}(B, \alpha, t_\alpha, \Theta_\alpha) \Rightarrow p)].$

We now generalize the axioms A5 and A6 [42] into Axiom 1 in terms of LEAD'. Axiom 1 in Fig. 2 says that if an agent does not believe that p is true now, but has an intention that p be true at some future time, it will consider doing some action β (Pot.Int.To) if it believes that the performance of β could contribute to making p true directly or indirectly through another action by another agent.

2.1.2. Performatives as attempts

Following Searle's observation on speech acts [79], Cohen and Levesque developed a formal foundation for agent communication based on their theory of rational agency [20]. In their approach [21,22], all illocutionary acts are treated as attempts. Cohen and Levesque have given several slightly different definitions of an `attempt` [21,22,24,55], all of which define "attempt" as a complex action expression involving a chosen goal and an intention.[8] One of their definitions [55], which has a time argument, is given below:

[8] Basic notions [21,55]: (HAPPENS *expr*) and (DONE *expr*) represent that a sequence of events described by action expression *expr* will happen next or has just happened, respectively. Unilateral mutual belief is defined recursively: $(\mathtt{BMB}\ A\ B\ p) \triangleq (\mathtt{BEL}\ A\ p \wedge (\mathtt{BMB}\ B\ A\ p))$. $(\mathtt{BEFORE}\ expr\ p) \triangleq (\mathtt{DONE}\ p?; expr)$, $(\mathtt{AFTER}\ expr\ p) \triangleq (\mathtt{HAPPENS}\ expr; p?)$.

Definition 2. $(\texttt{attempt}\ A\ \epsilon\ P\ Q\ t) \triangleq t?; \phi?; \epsilon$, where
$\phi = (\texttt{BEL}\ A\ \neg P) \wedge \Theta \wedge (\texttt{INTEND1}\ A\ t?; \epsilon; Q?\ \Theta)$, where
$\Theta = (\texttt{GOAL}\ A\ (\texttt{HAPPENS}\ \epsilon; \diamond P?))$.

An attempt at time t to achieve P via Q is a complex action expression in which the agent A is the actor of event ϵ, and just prior to ϵ, the agent believes P is false, *chooses* that P should eventually become true, and *intends* that ϵ should produce Q relative to that choice. Here, P represents some ultimate goal that may or may not be achieved by the attempt while Q represents an honest effort. More specifically, if the attempt does not achieve the goal P, the agent may retry the attempt, try some other strategy or even drop the goal. However, if the attempt does not succeed in achieving the honest effort Q, the agent is committed to retrying until either Q is achieved or Q becomes unachievable or irrelevant [21,84].

The semantics of elementary performatives are given by choosing appropriate formulas (involving mutual beliefs) to substitute for P and Q in the definition of `attempt`. The following is a variant of `inform` defined by Kumar et al. [55].

Definition 3. $(\texttt{inform}\ A\ B\ \epsilon\ I\ t) \triangleq (\texttt{attempt}\ A\ \epsilon\ \phi\ \psi\ t)$, where
$\phi = (\texttt{BMB}\ B\ A\ I)$,
$\psi = (\texttt{BMB}\ B\ A\ P)$, where
$P = (\texttt{BEFORE}\ \epsilon\ [\texttt{GOAL}\ A\ (\texttt{AFTER}\ \epsilon\ [\texttt{BEL}\ B\ (\texttt{BEFORE}\ \epsilon\ (\texttt{BEL}\ A\ I))])])$.

The goal of an `inform` is that the addressee B come to believe that there is a mutual belief between him and the informing agent A that the proposition I is true. The intention of an `inform` is that B come to believe another mutual belief; namely, that before performing the `inform`, the informing agent A had the goal that "after the `inform` is performed, B will believe that A believed I before performing the `inform`".

A request with respect to action α is defined as an attempt of the speaker to make both the speaker and the addressee believe that the speaker intends that the addressee commit to performing the action α. Here, the speaker's commitment is to having his/her wants understood by the addressee. The following is a variant of the `request` defined by Cohen and Levesque [24], where $(\texttt{HELPFUL}\ B\ A)$ represents that agent B is helpful to agent A [22].

Definition 4. $(\texttt{request}\ A\ B\ \epsilon\ \alpha\ t) \triangleq (\texttt{attempt}\ A\ \epsilon\ \phi\ \psi\ t)$, where
$\phi = (\texttt{DONE}\ B\ \alpha)$,
$\psi = (\texttt{BMB}\ B\ A\ P)$, where
$P = (\texttt{BEFORE}\ \epsilon\ (\texttt{GOAL}\ A\ [(\texttt{DONE}\ B\ \alpha) \wedge (\texttt{AFTER}\ \epsilon\ (\texttt{INTEND1}\ B\ \alpha\ \Theta))]))$, where
$\Theta = (\texttt{GOAL}\ A\ (\texttt{DONE}\ B\ \alpha) \wedge (\texttt{HELPFUL}\ B\ A))$.

Other traditional communicative acts (e.g., ASK) can be defined in terms of `inform` and `request` by using compositionality. For instance, the formal semantics of performatives in several agent communication languages, such as Arcol [11], KQML [58], and FIPA's ACL [1], are all framed in this way. Currently, the idea of *performative-as-attempt* has been extended to the area of Proxy-Communicative acts [46], group interactions [56] and conversation protocols [55,84].

3. Preliminaries

In this section, we make some preparation for further explorations. Our assumptions on the belief and intentional attitudes are stated first; then, we consider the notions related to actions and contexts. Agents in teamwork settings are supposed to be sincere and helpful; thus, in Section 3.3 we define team sincerity and propose a richer axiom to specify chains of helping behaviors. To describe stratified information needs, we introduce the notion of social inference trees in Section 3.4. Several research groups, e.g., CAST [99], have been using the Cohen–Levesque semantics in systems implemented using the SharedPlans framework, but there has not been a formal grounding of that semantics in this framework. To this end, in Section 3.5, we re-formulate the Cohen–Levesque semantics of communicative acts using the SharedPlans formalism. In the following, all free variables are implicitly universally quantified.

3.1. Assumptions on mental attitudes

We adopt the typical treatment of the belief attitude and assume Bel conforms to the K, D, 4 and 5 axioms of modal logic [35]. In addition, we assume that the idempotence property holds for Bel, i.e., $\mathsf{Bel}(A, \mathsf{Bel}(A, p, t), t) \Leftrightarrow \mathsf{Bel}(A, p, t)$ (the $\Leftarrow$ part corresponds to axiom '4'). We adopt the K and D axioms of modal logic for the intentional attitudes Int.To and Int.Th, and adopt the K axiom for Pot.Int.To and Pot.Int.Th.[9] Possible worlds semantics is used where each possible world is a temporal structure.

Intentions and beliefs persist by default until the newly acquired information causes conflicts or the original contexts of the intentions no longer hold or the intentions have been achieved. For Int.Th, as well as the associated context, the second time argument also serves as one constraint on holding an intention-that. More specifically, suppose agent A has an intention $\mathsf{Int.Th}(A, p, t, t_3, C_p)$, and C_p is true before t_3. As time goes on from t to some time t_1 ($< t_3$), the intention will become $\mathsf{Int.Th}(A, p, t_1, t_3, C_p)$. Now suppose that at time t_1 agent A comes to believe p. Because p might change between t_1 and t_3, A should continue to hold the intention until t_3. Of course, in some cases, achievement goals can be reduced to maintenance goals. For instance, if p is maintainable for A (e.g., A can control the changing of p), A could replace $\mathsf{Int.Th}(A, p, t_1, t_3, C_p)$ with $\mathsf{Int.To}(A, \mathit{maintain}(p), t_1, t_3, C_p)$, so that the agent is committed to maintaining p until t_3.

[9] Numerous researchers have struggled over how best to represent agent intentions [20,42,93]. Grosz and Kraus did not provide explicit constraints on accessibility relations for these intentional attitudes. K and D are typically adopted for normal intentions [75]. We adopt K for potential intentions because K is the weakest constraint on normal modal operators. D is not applicable to Pot.Int.To or Pot.Int.Th because an agent could hold conflicting potential intentions. K may be insufficient for potential intentions and more constraints can be explored in future studies. However, in this paper, should confusion occur, Grosz and Kraus's informal interpretation of potential intentions [42] applies.

In addition to those axioms (p. 17, [42]) given in the SharedPlans theory, we assume "goals are known" [21,47] for the relationships between Bel and intentions (Int.To and Int.Th).

Assumption 1. (1) $\mathsf{Int.To}(A, \alpha, t, t', C_\alpha) \Rightarrow \mathsf{Bel}(A, \mathsf{Int.To}(A, \alpha, t, t', C_\alpha), t)$,
(2) $\mathsf{Int.Th}(A, p, t, t', C_p) \Rightarrow \mathsf{Bel}(A, \mathsf{Int.Th}(A, p, t, t', C_p), t)$.

We thus have,

$$\mathsf{Int.To}(A, \alpha, t, t', C_\alpha) \Leftrightarrow \mathsf{Bel}(A, \mathsf{Int.To}(A, \alpha, t, t', C_\alpha), t),$$
$$\mathsf{Int.Th}(A, p, t, t', C_p) \Leftrightarrow \mathsf{Bel}(A, \mathsf{Int.Th}(A, p, t, t', C_p), t).$$

We also assume agents have perfect recall of what was believed.

Assumption 2. $\mathsf{Bel}(A, \mathsf{Bel}(B, p, t_0), t) \Rightarrow \forall t' \geqslant t \cdot \mathsf{Bel}(A, \mathsf{Bel}(B, p, t_0), t')$.

To represent objective views that may conflict with the subjective views of resource-bounded agents, we introduce a predicate $\mathsf{Hold}(p, t)$,[10] which means p is objectively true at time t. Note that Hold is external to any rational agent. It presupposes an omniscient perspective from which to evaluate p. In other words, assume there exists an omniscient agent G, then $\mathsf{Hold}(p, t) = \mathsf{Bel}(G, p, t)$. Hold will be used only within belief or intention contexts, say $\mathsf{Bel}(A, \mathsf{Hold}(p, t), t)$, which means agent A believes from the omniscient's perspective p is true. Since omniscience is always trustable, $\mathsf{Bel}(A, \mathsf{Hold}(p, t), t) \Rightarrow \mathsf{Bel}(A, p, t)$, but not vice versa.

We define some abbreviations needed later. Awareness (Bif: believe-if [1]);[11] belief contradiction (CBel) between two agents that is recognized by one, but not necessarily both; and wrong beliefs (WBel) are given as:

Definition 5 (*Abbreviations*).

$$\mathsf{Bif}(A, p, t) \triangleq \mathsf{Bel}(A, p, t) \vee \mathsf{Bel}(A, \neg p, t),$$
$$\mathsf{UBif}(A, p, t) \triangleq \neg\mathsf{Bif}(A, p, t),$$
$$\mathsf{CBel}(A, B, p, t) \triangleq (\mathsf{Bel}(A, p, t) \wedge \mathsf{Bel}(A, \mathsf{Bel}(B, \neg p, t), t)) \vee (\mathsf{Bel}(A, \neg p, t) \wedge \mathsf{Bel}(A, \mathsf{Bel}(B, p, t), t)),$$
$$\mathsf{WBel}(A, p, t) \triangleq (\mathsf{Hold}(p, t) \wedge \mathsf{Bel}(A, \neg p, t)) \vee (\mathsf{Hold}(\neg p, t) \wedge \mathsf{Bel}(A, p, t)).$$

Note that $\mathsf{CBel}(A, B, p, t)$ means from agent A's point of view, there is a contradiction regarding p between B and A itself. It may be the case that there actually is not a contradiction at all (i.e., A was wrong). Also note that the definition given is not reflexive, so $\mathsf{CBel}(A, A, p, t)$ does not hold. Nor is it symmetric. For example, A might believe there

[10] In this paper, propositions may or may not have a time or time interval associated with them. If a proposition p has an associated time argument, it can be different from the external times, say, t in $\mathsf{Hold}(p, t)$.

[11] We assume that belief bases allow three possible truth values for propositions.

is a contradiction while B believes that the beliefs are consistent. From the definitions we also have: $\mathsf{Bel}(A, \mathsf{WBel}(B, p, t), t) \Rightarrow \mathsf{CBel}(A, B, p, t)$.

3.2. Actions and context

An action is either primitive or complex. Complex actions can be built from primitive actions by using the constructs of dynamic logic: $\alpha; \beta$ for sequential composition, $\alpha|\beta$ for nondeterministic choice, $p?$ for testing (where p is a logical formula), and α^* for repetition. Let $post(\alpha)$ return a conjunction of propositions that describe the effects of α.

The SharedPlans theory defines a recipe for a complex act-type γ as a specification of a group of subsidiary actions, the doing of which under certain constraints constitutes the performance of γ. Thus, a recipe *per se* is composed of an action expression and a set of constraints on the action expression. A set of recipes can be specified for an act-type. Let $recipe_A(\alpha)$ be the set of recipes of α specified for agent A. $recipe_A(\alpha)$ and $recipe_B(\alpha)$ may be the same, overlapped, or even disjoint.

The SharedPlans theory assumes that all actions are intended, committed and performed in some specific *context* [42]. In this paper, the notion of context is extended to be used in three different ways. In addition to being arguments of intentions, contexts will also play important roles in the notion of information needs and the semantics of proactive performatives introduced later.

We use C or C with subscripts or superscripts to refer to contexts. Nevertheless, the subscript (or superscript) on a context does not impart any meaning to the context; the meaning of a context only depends on where the context occurs. For instance, when C_1 occurs as an argument of Int.To(intention-to), it refers to the context in which the action (another argument of Int.To) is being done; when C_1 occurs as an argument of Int.Th(intention-that), it refers to the context in which the proposition (another argument of Int.Th) is intended. However, to make notations more consistent, we use C_α (or C_p) to refer to the context in which action α (or proposition p) is concerned.

Grosz and Kraus allowed an intentional context to include terms. For instance, the partial recipe a group of agents have is part of the intentional context for the intention-to do subsidiary actions [43]. To be uniform, we choose to use meta-predicates to represent terms in contexts. For example, $has.recipe(A, \alpha, R, t)$, which was initially used by Lochbaum [67], can be used to represent that "agent A has recipe R to do action α at time t". We thus take a context as being composed of a set of formulae, which are collectively evaluated as one conjunction. Our treatment of context establishes a correspondence between the intentional context in SPT and the term "escape clause" in JIT. However, the notion of intentional contexts is richer than escape clauses. In addition to conditions that allow agents to be free from their commitments, intentional contexts can also include other constraints that guide replanning or recipe selection.

In general, the constituent formulae of a context may play different roles. Some part of a context may serve as constraints. For instance, the deadline of doing α and the dependency of α on other actions might impose constraints on the performance of α. Some part of a context may serve as traces of explanation. For instance, a chain of actions may be invoked for the achievement of a certain goal. Some part of a context may serve as criteria for attention management such as goal reconciliation or task delegation; and some may serve to specify agent social relationships based on the agent's social roles, sincerity, helping behavior, etc. Thus, we assume each context formula is associated with certain meta-level information indicating the roles of the formula, and functions are defined for obtaining those components of a context related to a specific role. For example, $Constr(C_1)$ denotes the constraints component of context C_1.

Let p_1, p_2, and p_3 be formulae, $C_1 = \{p_1, p_2\}$. For notational convenience, we use $C_1 \cup \{p_3\}$, $C_1 \wedge p_3$, and $p_1 \wedge p_2 \wedge p_3$ interchangeably when they are used as context. We also use $p \in C$ to represent that p is a part of the context C.

3.3. Properties of collaborative agents

We fix *TA* and *TB* to be two agent teams, each with a finite number of members. $T, T_1, T_2, \ldots$ are used to refer to subsets of *TA* (or *TB*).

Within a team, all the member agents share certain joint objectives, and we assume they are sincere in communication and exhibit a certain degree of helpfulness in their collaborations for achieving their shared goals. An agent is insincere if it knowingly wants others to come to believe false things [22]. Thus, we say that an agent is sincere if the agent desires to achieve consistent beliefs with other teammates.

Axiom 2 in Fig. 3 says that, (1) whenever an agent A has an intention toward letting another agent B believe p, then A also commits to making p true; (2) whenever an agent A has an intention toward letting another agent B believe that 'A believes p', then A itself really believes p and commits to making p true.

The sincerity axiom in Fig. 3 involves two forms of intentions. By Axiom 1 (cf. Fig. 2), the intention $\mathsf{Int.Th}(A, \mathsf{Bel}(B, p, t_2), t, t_1, C)$ could lead agent A to perform certain (communicative) actions (e.g., inform B directly, or request that another agent inform B). The intention $\mathsf{Int.Th}(A, \mathsf{Hold}(p, t_2), t, t_1, C)$ serves as a constraint on adopting new intentions. For instance, to be sincere, before t_2 (say, t') agent A cannot adopt an intention $\mathsf{Int.Th}(A, \neg p, t, t', C')$, nor can A adopt intentions to perform actions that may inhibit B from getting the truth regarding p. One consequence of this axiom is that a sincere agent only communicates information that it believes to be true.

In essence, this characterization of agent sincerity is equivalent to Cohen and Levesque's [22]. Their definition requires that whenever agent A wants agent B to come to believe p, it actually wants B to come to know p. Since knowledge is true belief, A's chosen desire of "letting B know p" serves the same role as $\mathsf{Int.Th}(A, \mathsf{Hold}(p, t_2), t, t_1, C)$ does. Note that both definitions allow an agent to perform third-party communicative actions to fulfill its sincerity to another agent. This is attainable because there is no constraint in Cohen and Levesque's definition that may preclude an agent from adopting an action e involving a third-party. Later we will show that in our framework third-party actions can be triggered

Axiom 2 (Sincerity: from A to B).
(1) $\mathsf{Int.Th}(A, \mathsf{Bel}(B, p, t_2), t, t_1, C) \Rightarrow \mathsf{Int.Th}(A, \mathsf{Hold}(p, t_2), t, t_1, C)$,
(2) $\mathsf{Int.Th}(A, \mathsf{Bel}(B, \mathsf{Bel}(A, p, t), t_2), t, t_1, C) \Rightarrow$
$[\mathsf{Bel}(A, p, t) \wedge \mathsf{Int.Th}(A, \mathsf{Hold}(p, t_2), t, t_1, C)]$.

Fig. 3. The sincerity axiom.

from intentions: When agent A intends that agent B believe p but A itself is unaware of p, A can request other teammates to help B.

Now we come to the concept of "helpfulness". Axiom 1 (cf. Fig. 2) is a specification for helping behaviors: if an agent intends that p hold, it must be willing to consider undertaking actions that will help achieve p [43]. For instance, suppose an agent A has an intention to make p true (Int.Th). A will adopt a potential intention to do action α if the performance of α will enable another agent B to perform some action β, which would directly make p true. Ask/reply is an instance of using this axiom: after being asked, if an agent has adopted an intention to honor the information request, 'reply' is a potential way to do so. Enabling others' physical actions also falls into this category. For example, a logistics person supplies ammunition to a fighter in a joint mission.

Cohen and Levesque also defined "helpfulness" in their framework [22]. Agent A is helpful to agent B if for any action that B wants A to do, A actually adopts a goal to eventually do that action, whenever such a goal would not conflict with A's own. The notion of helpfulness characterized in Axiom 1 is different from Cohen and Levesque's. In Axiom 1, an agent adopts helping behavior relative to its commitments to team activity or its own individual goals involving other agents (e.g., if agent A has intention $\mathsf{Int.Th}(A, \mathsf{Bel}(B, p, t'), t, t', C)$, it will commit to doing some action if that action can lead to letting B know p at t'); while in Cohen and Levesque's approach, an agent adopts its commitment (helping behavior) relative to the other agent's goal. In a sense, Cohen and Levesque's notion is more like a response of one agent to another's request—consider doing what is appealed for. Helpfulness in Axiom 1, on the other hand, is more self-motivated—consider helping if both parties can benefit from it. Hence, Axiom 1 turns out to be more useful to us for studying proactive behavior.

However, Axiom 1 is still not rich enough to cover helping behaviors involving three or more parties. Such behaviors occur predominantly in large hierarchical teams with subteams. As illustrated in Fig. 4, suppose agent $A1$ needs help (described by predicate p); only agents in subteam 1 know $A1$'s needs, and there is no teammate in subteam 1 who can directly satisfy $A1$. Assume an agent $A2$ knows that some other agent (say, $B1$) in another subteam *may* be able to directly satisfy $A1$ by performing action α. In the real case, in order to be helpful, $A2$ may consider informing $B1$ about $A1$'s needs or requesting that $B1$ help $A1$. However, Axiom 1 cannot give an account for such indirect helping behavior because it requires that $A2$ has to believe that the performance of α by $B1$ ensures that $A1$'s needs will be satisfied, which is not the case here. Moreover, $A2$ may only know that $B1$ could indirectly contribute to $A1$'s needs rather than directly satisfy them. In such cases, $B1$ may

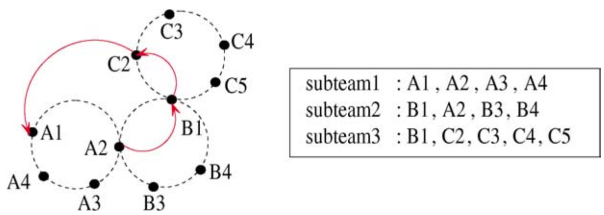

Fig. 4. An illustration of chain of helping.

Axiom 3. $\forall p, t, \beta, t_\beta, t' > t_\beta, C_p\cdot$
$[\mathsf{Int.Th}(A, p, t, t', C_p) \wedge \neg\mathsf{Bel}(A, p, t) \wedge \mathsf{Lead}(A, \beta, p, t, t_\beta, \Theta_\beta)] \Rightarrow$
$\mathsf{Pot.Int.To}(A, \beta, t, t_\beta, \Theta_\beta \wedge C_p)$, *where*
$\mathsf{Lead}(A, \beta, p, t, t_\beta, \Theta_\beta) \triangleq \mathsf{Bel}(A, P1, t) \wedge [\mathsf{Bel}(A, P2, t) \vee \mathsf{Bel}(A, P3, t)]$, *where*
$P1 = \exists R_\beta \cdot \mathsf{CBA}(A, \beta, R_\beta, t_\beta, \Theta_\beta)$,
$P2 = (\mathsf{Do}(A, \beta, t_\beta, \Theta_\beta) \Rightarrow p)$,
$P3 = \mathsf{Do}(A, \beta, t_\beta, \Theta_\beta) \Rightarrow \psi$, *where*
$\psi = [\exists B, \alpha, R_\alpha, t_\alpha, t''\cdot (t_\alpha > t_\beta) \wedge (t_\alpha > t'') \wedge$
$\mathsf{CBA}(B, \alpha, R_\alpha, t_\alpha, \Theta_\alpha) \wedge$
$\mathsf{Pot.Int.To}(B, \alpha, t'', t_\alpha, \Theta_\alpha \wedge \underline{\mathsf{Lead}(B, \alpha, p, t'', t_\alpha, \Theta_\alpha)})]$

Fig. 5. The axiom characterizing chains of helping behavior.

forward $A1$'s needs to agent $C2$ in yet another subteam, knowing that $C2$ may be able to contribute. We need to generalize Axiom 1 to cover such chains of helping behavior.

To generalize Axiom 1, we need to redefine the LEAD' meta-predicate, which requires changing only one clause in the definition of the LEAD'. We choose to define 'lead-to' recursively. In so doing, it would seem that by simply replacing the conjunct "$\mathsf{Do}(B, \alpha, t_\alpha, \Theta_\alpha) \Rightarrow p$" in the definition of the LEAD' by "$\mathsf{LEAD}'\ (B, \alpha, p, t'', t_\alpha, \Theta_\alpha)$", chains of helping behaviors could be explained. But this simply does not work because it would impose too strong requirements on agents' beliefs. More specifically, it would require agent A to believe that the performance of action β can (1) enable agent B to perform action α, (2) motivate B to consider doing it (Pot.Int.To), and (3) necessitate B to adopt beliefs required by LEAD' with respect to α (since LEAD' is defined in terms of beliefs). However, generally an agent cannot guarantee, even within its beliefs, that its action can affect others' beliefs. In this case, even though the performance of β by A could enable B to perform α, B itself may not be able to realize this.

An alternate approach is to drop "$\mathsf{Do}(B, \alpha, t_\alpha, \Theta_\alpha) \Rightarrow p$" and add a recursion clause to the intentional context of the Pot.Int.To. The modified axiom is shown in Fig. 5.

The recursive definition of Lead in Fig. 5 states that A believes it can bring about action β, the performance of which can either result in p, or enable another agent B to do some action α and motivate B to consider doing α under the context that (1) B believes it can bring about α, (2) B believes the performance of α can result in p or enable yet another agent to do some action which can lead to p, and so on. Such a chain of reasoning initiates a chain of helping behavior which may ultimately satisfy the needer (agent A). The definition is succinct and neatly combines direct and indirect helping behaviors. The Leads within the context of Pot.Int.Tos serve as an explanation for the chain of helping behaviors among agents.

Some would argue for the avoidance of complex nestings of existential quantifiers and modal belief operators. For instance, in the original SharedPlans theory [42], even decisions that have already been made by a group are modeled implicitly using existential quantification, which makes it difficult to reason about certain properties. Hunsberger [47] used SharedPlan Trees as arguments of meta-predicates to reformulate SharedPlans. The reformulation eliminates certain existential quantifications and thereby allows a set of important theorems to be proved. However, this is not the case here. The two uses of existential quantifiers within Bel in the definition of Lead are necessary. For instance, $\exists R_\beta \cdot \mathsf{Bel}(A, \mathsf{CBA}(A, \beta, R_\beta, t_\beta, \Theta_\beta), t)$ is much stronger than $\mathsf{Bel}(A, \exists R_\beta \cdot \mathsf{CBA}(A, \beta, R_\beta, t_\beta, \Theta_\beta), t)$. If the former were used, it would require A to hold a same recipe in all the belief accessible worlds. For the same reason, the second existential quantifier cannot be moved outside the scope of Bel.

The new definition of Lead does not necessarily cause problems of infinite recursion either. First, there is a base clause. The recursion terminates whenever an agent can directly bring about p. Second, the nested reasoning is not typically performed by a single agent (say, agent $A2$) but distributed among all the agents involved in the help chain. In system implementation, it may be sufficient for an agent to consider at most one level of recursion. For instance, in Fig. 4, agent $A2$ can help $A1$ with the information need p if $A2$ knows $B1$ can perform some action that can make the state of affairs closer to p without necessarily knowing that $B1$ may further ask $C2$ for help.

Compared with Axiom 1, Axiom 3 can be used to characterize chains of helping in large agent teams. In particular, it establishes a basis for choosing a third-party communicative act defined in a later section. Even though third-party communicative acts have been studied in the Joint Intentions theory [46], in addition to other differences from our approach (e.g., driven by explicit needs), the Joint Intentions theory does not have such an axiom from which helping behaviors can be derived (it is the joint intentions that serve as the motivation to any helping behaviors). Consequently, the Joint Intentions theory lacks a generic characterization of helping behavior involving three or more parties.

We assume that Axiom 3 holds in belief contexts. In the rest of this article, we also assume that all agents in a team are helpful to others, and use $\mathsf{Helpful}(A)$ as a meta-predicate to refer to Axiom 3.

3.4. Preconditions and social inference trees

Prior to performing a plan or action, an agent typically needs to check whether the plan or action is both physically and epistemically feasible [30]. In other words, obstacles to plans or actions come in one of two varieties: physical and informational. Accordingly, we distinguish physical preconditions from informational preconditions.

For instance, suppose that in a battlefield domain there is a complex action called $\mathit{RemoveThreat}(?e, ?loc, ?dir, ?num)$. Upon knowing a threat from an enemy unit, the performers of this action may either choose to attack the enemy from the flank, or wait for the enemy to become exhausted. This *RemoveThreat* can be represented as:

$$(\mathit{MoveToFlank}(?e, ?loc, ?dir); \mathit{Fire}(?e, ?num)) \mid$$
$$((\mathit{FarAway}(?e, \mathit{Self})?; \mathit{Wait}(\mathit{Self}))^*; \mathit{Fire}(?e, ?num)).$$

Assume that the preconditions of *RemoveThreat* involve three pieces: (1) $\mathit{CanFight}(\mathit{Self})$: the agent can fight—this may require the agent to have enough fighting power, to move, etc.; (2) $\mathit{Threat}(?e, ?loc, ?dir, ?num)$: the agent knows the threat to be removed; and (3) $\mathit{Outmatch}(?e, ?num)$: the agent knows its own team outnumbers the enemy unit. Here, for the complex action *RemoveThreat*, $\mathit{CanFight}(\mathit{Self})$ is an example of physical preconditions while $\mathit{Outmatch}(?e, ?num)$ and $\mathit{Threat}(?e, ?loc, ?dir, ?num)$ are kinds of informational preconditions.

As far as helping behavior is concerned, the other agents can help the performers of *RemoveThreat* overcome the physical obstacles. For instance, if a performer cannot fight, the other agents could enable the performer by delivering supplies or removing potential barriers. The other agents can also help the performers of *RemoveThreat* with the informational obstacles. For instance, if a performer does not know of an approaching threat, the other agents could provide threat information to the performer proactively. Since proactive information delivery is our concern, we will focus on informational preconditions only.

Informational preconditions may also have different varieties. A complex action or plan may have associated *constraints* (or preconditions) which have to be satisfied prior to the action being performed. For instance, suppose prior to removing a threat an agent has to know whether its own team outnumbers the enemies. In addition, a complex action or plan may have *knowledge preconditions* [30,69]: the agent has to know enough to carry out a plan. Lochbaum [67] recast the observations on knowledge preconditions made by Morgenstern [69] into the terminology of the SharedPlans framework. She used the predicates *has.recipe* and *id.params* respectively to represent that (1) agents need to know recipes (*know-how* information) for the acts to be performed, and (2) agents must be able to identify the parameters of the acts to be performed.

Without loss of generality, we assume that agents in working teams already have applicable recipes for single-agent actions and could exchange meta-level information to collaboratively construct shared partial recipes for complex actions. To further simplify the issue of parameter identification, we also assume that the parameters of an action either have constant values or their values are propagated from a higher-level action (plan) or can be determined if the preconditions are satisfied. Consequently, the task of parameter identification for an action (or plan) is reduced to satisfying the preconditions of the action (or plan). For example, the parameters of *RemoveThreat* are determined as soon as the predicate *Threat* can be unified successfully with the agent's belief base.

Now we formally characterize action preconditions. As we mentioned in Section 3.2, several recipes may be specified for a complex action. Thus, different from Grosz and Kraus' treatment [42], we assume that action preconditions depend on recipes. For instance, another recipe for *RemoveThreat* can be specified as: recruit an echelon unit to induce the approaching enemy to move away from the crucial area. To carry out this recipe, the agents recruited to remove the threat also need to know the location of the crucial area, as well as the pre-requisite information about the approaching enemy. Let R_α be a recipe for action α, and $pre(R_\alpha)$ be the preconditions specified for R_α. Then we use $pre_A(\alpha)$ to denote the set of preconditions associated with any recipe for doing α that A knows about, i.e., $pre_A(\alpha) = \bigcup_{R_\alpha \in recipe_A(\alpha)} pre(R_\alpha)$.[12] Thus, $I \in pre_A(\alpha)$ simply states that I is some precondition for some recipe for doing α that A knows about. This will be used in Section 5.1 to allow an agent to anticipate teammates' information needs regarding action α based on those recipes of α known to the agent. In approximately anticipating others' information needs, an agent A needs to consider all the known recipes for α because (1) A may not know the sets of recipes for α its teammates are considering, and (2) even if A

[12] In case that α is a primitive action, $pre_A(\alpha)$ refers to the collection of preconditions associated with α.

and its teammates have the same set of recipes for α, A may not know which recipes its teammates are going to choose to perform α.[13]

In order to facilitate the reasoning of information needs at different levels of abstraction and to represent the context of inference, we introduce the concept of *social inference trees*, which organize predicates in a tree-like structure capturing the hierarchical reasoning as reflected in agents' inference knowledge. Generally, a collection of predicates (e.g., action preconditions, constraints, intentional contexts or preference conditions) can be structured into a social inference tree.

Continuing with the example from the beginning of this section, suppose that agents' inference knowledge is represented by Horn clauses, as follows:

(1) *Threat* is the head predicate of a Horn clause:

$$\begin{aligned}
&\mathit{Threat}(?e, ?loc, ?dir, ?num) \leftarrow \\
&\qquad [\mathit{IsEnemy}(?e), \\
&\qquad\quad \mathit{At}(?e, ?loc, \mathit{NOW}), \\
&\qquad\qquad \mathit{Dir}(?e, ?dir), \\
&\qquad\qquad\quad \mathit{Number}(?e, ?num)].
\end{aligned}$$

That is, the agent could deduce the existence of a threat if it had belief about the identified enemy unit (*IsEnemy*), the location of the enemy unit (*At*), the direction in which the enemy unit is moving (*Dir*), and the number of enemies in the unit (*Number*).

(2) *Dir* is the head predicate of a Horn clause:

$$\begin{aligned}
&\mathit{Dir}(?e, ?dir) \leftarrow \\
&\qquad [\mathit{At}(?e, ?l1, \mathit{NOW} - 1), \\
&\qquad\quad \mathit{At}(?e, ?l2, \mathit{NOW}), \\
&\qquad\qquad \mathit{Compass}(?l1, ?l2, ?dir)].
\end{aligned}$$

That is, to deduce the moving direction, the agent needs to know the change of location, from which to infer the direction. And

(3) *CanFight* is the head predicate of a Horn clause:

$$\mathit{CanFight}(\mathit{Self}) \leftarrow [\mathit{HaveFP}(\mathit{Self}), \mathit{CanMove}(\mathit{Self})].$$

That is, to be able to fight, the agent needs to have enough fighting power, and also be able to move to the targets. Fig. 6 shows the inference tree constructed for the preconditions of *RemoveThreat*, where the dotted subtree from node 'Threat' will be discussed later.

[13] This may cause computational explosion. Certain strategies can be employed to simplify the reasoning in implementing practical agent systems. For instance, to reduce the scope of reasoning, agents can record and learn the most frequently used recipes by its teammates. The social inference trees discussed later in this paper can also alleviate this problem. However, general approaches to dealing with computational complexity are beyond the scope of this paper.

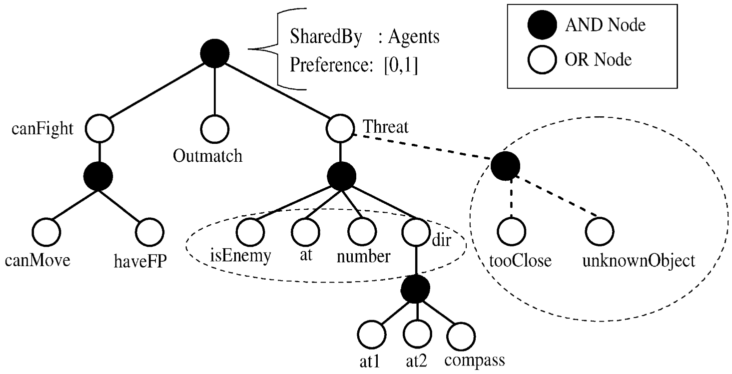

Fig. 6. The inference tree for the preconditions of *RemoveThreat*.

Similar to AND/OR trees [87], a social inference tree is composed of "AND" nodes and "OR" nodes, where each "OR" node is labeled with a predicate, and every "AND" node represents one piece of inference knowledge. The truth value of the predicate labeling its parent OR node can be inferred from the truth values of the predicates labeling its child nodes. Social inference trees can be generated at compile time and refined at run time to reflect the dynamics of agents' inference knowledge.

The notion of social inference trees differs from AND/OR trees in several aspects. First, agents in a team may have different inference trees due to their differences in inference capability. Second, each AND node in a social inference tree is associated with a list of agents who share the corresponding inference knowledge. Such lists play a very important role in proactive communications. For instance, knowing a teammate has the same inference knowledge, an agent may not consider the teammate as a needer of the inferred information unless being explicitly requested. Moreover, the agent list actually provides points of contact between information needers and information providers. When an agent matches the information needs of a teammate with the predicate labeling the parent of an AND node, the agent can consider initiating a third-party communication action toward some potential provider in the agent list. Third, each leaf node is associated with a list of agents who have the ability to observe the information relevant to the predicate labeling the node. Knowing the observability of teammates helps an agent find an appropriate information provider. Fourth, each AND node is associated with a dynamically adjustable preference value. In cases where a predicate can be inferred multiple ways, the corresponding OR node will have multiple AND nodes as its children. For example, the dotted subtree in Fig. 6 shows another way of inferring threat from lower-level information. In such cases, the preference information can be leveraged to guide an agent in its information gathering and fusing activities. For instance, to minimize inter-agent dependence, an agent may prefer to use the subtree that involves the least number of teammates; to improve robustness, an agent may prefer to use the subtree where most of the OR nodes have multiple branches. Consequently, such preference information is useful in circumscribing the scope of reasoning in anticipating others' information needs.

In an inference tree, the nodes at the same level collectively form a context for each individual. For instance, in Fig. 6, as far as threat identification is concerned, $Dir(?e, ?dir)$

is useful only when it is evaluated together with $\mathit{IsEnemy}(?e)$, $\mathit{At}(?e, ?\mathit{loc}, \mathit{NOW})$, and $\mathit{Number}(?e, ?\mathit{num})$. Thus,

$$\{\mathit{Dir}(?e, ?\mathit{dir}), \mathit{IsEnemy}(?e), \mathit{At}(?e, ?\mathit{loc}, \mathit{NOW}), \mathit{Number}(?e, ?\mathit{num})\} \qquad \text{(pc1)}$$

collectively establishes a context for each of the individual predicates.

Such contexts at inference level, together with inference trees, can be used in collaborative constraint satisfaction. Suppose that agents A_1, A_2 and A_3 share the inference tree shown in Fig. 6, and that A_3 is the agent of the *RemoveThreat* action. Assume both A_1 and A_2 have identified an enemy unit ($e1$) approaching A_3, who is unaware of the threat from $e1$. Also assume that A_1 can only observe the location and moving direction of e1, as represented by the predicates $\mathit{At}(e1, \mathit{area}_4, \mathit{NOW})$ and $\mathit{Dir}(e1, \mathit{northeast})$; A_2 can only observe the enemy number, $\mathit{Number}(e1, 100)$, of unit $e1$. Obviously, neither A_1 nor A_2 alone can enable A_3 to do *RemoveThreat*. However, they can collaboratively help A_3 because A_1 knows that $\mathit{At}(e1, \mathit{area}_4, \mathit{NOW})$ and $\mathit{Dir}(e1, \mathit{northeast})$ will be useful for A_3 in the context $pc1$, and A_2 knows that $\mathit{Number}(e1, 100)$ will be useful for A_3 in that context.

Such contexts at inference level can also be used to account for anticipated information needs and the exchange of incomplete information. Both concepts will be discussed in Section 4.

3.5. *Reformulating performative-as-attempt in the SharedPlans framework*

Following the idea of "performative-as-attempt" [21,22], we will model the intentional semantics of proactive performatives to establish certain mutual beliefs between the speaker and the addressee (or addressees). In order to do that, we first need to reformulate the concept of "attempt" within the framework of the SharedPlans theory. Then, the semantics of "inform" and "request" are given in terms of attempts. This serves partially to validate our approach of encoding "performative-as-attempt" in the SharedPlans framework.

Definition 6. $\mathsf{Attempt}(A, \epsilon, P, Q, C_n, t, t_1) \triangleq \phi?; \epsilon$, where

$$\begin{aligned}
\phi = [&\neg\mathsf{Bel}(A, P, t) \wedge \\
&\quad \mathsf{Pot.Int.Th}(A, P, t, t_1, C_n) \wedge \\
&\quad\quad \mathsf{Int.Th}(A, Q, t, t_1, \neg\mathsf{Bel}(A, P, t) \wedge C_n) \wedge \\
&\quad\quad\quad \mathsf{Int.To}(A, \epsilon, t, t, \psi)], \text{ where}
\end{aligned}$$

$$\psi = \mathsf{Bel}(A, \mathit{post}(\epsilon) \Rightarrow Q, t) \wedge \mathsf{Pot.Int.Th}(A, P, t, t_1, C_n).$$

Here, P represents some ultimate goal that may or may not be achieved by the attempt while Q represents what it takes to make an honest effort. The agent has only a limited commitment (potential intention) to the ultimate goal P, while having a full-fledged intention to achieve Q. More specifically, if the attempt does not achieve the goal P, the agent may retry the attempt, try some other strategy or even drop the goal. However, if the attempt does not succeed in achieving the honest effort Q, the agent is committed to retrying (e.g., performing ϵ again) until either Q is achieved, becomes unachievable (time t_1 arrives) or irrelevant (the escape condition C_n no longer holds) [21,84]. Thus, the Attempt involves an intention to achieve Q by performing ϵ while the underlying intention was to achieve P. Of course, P and Q may refer to the same formula.

For example, if P in the above definition is replaced by $\mathsf{Bel}(B, I, t)$, that means that agent A desires that $\mathsf{Bel}(B, I, t)$ hold. While $\mathsf{Bel}(B, I, t)$ may be unachievable for A, $\mathsf{MB}(\{A, B\}, \mathsf{Bel}(B, \mathsf{Bel}(A, I, t), t_1))$ (Q in this case) can be achieved by exchanging appropriate messages with B. In case of communication failure in establishing the mutual belief, A will retry until the mutual belief is achieved, C_n no longer holds or the deadline t_1 comes. Here ϵ may refer to a sequence of *send*, the act of wrapping the message in a wire language and physically sending it. When communication is reliable and sincerity is assumed, one *send* may suffice.

This definition of attempt differs from the definition of attempt (`attempt`) from Section 2.1.2 as follows.

- It introduces the explicit time-point t_1 that represents the deadline to terminate an attempt. When time t_1 arrives, an attempt becomes unachievable, thus the agent is released from retrying it.
- It includes C_n, which represents the context of an attempt.[14] C_n could be a placeholder in specifying escape conditions for Attempt, as well as describing the relationship between P and Q.
- It replaces the overly strong clause (`BEL` $A \neg p$) with the weaker clause $\neg\mathsf{Bel}(A, p, t)$, which we believe to be more reasonable.
- It uses potential intentions instead of `GOAL`s (used in the earlier definition `attempt`) to represent the ultimate goal. This is much flexible because it allows the expression of situations where an agent has an ultimate goal that may conflict with existing intentions.
- An Int.Th and an Int.To together are used to simulate the term `INTEND1` in `attempt`. (`INTEND1` $A\ t?; \epsilon; Q?\ C$) says that A intends to do the action ϵ with the result that Q holds. To highlight A's *intention* that Q hold, we use an Int.Th to represent the chosen honest goal, and use an Int.To to represent the agent's intention to do the event ϵ in the context that it believes the doing of ϵ will make Q true, which is related to the achievement of the ultimate goal P.

According to *speech act* theory [81], every speech act has an utterance event associated with it. For the purposes of this paper, we simply assume that all utterance events are single-agent complex actions, for which each agent has full individual recipes. For instance, when the honest goal of a performative is to establish certain mutual beliefs, the recipe for the corresponding ϵ may involve negotiations, persuasions or failure-handling.

We now formally define the successful doing of an attempt (performative).

Definition 7. $\mathsf{SuccDone}(A, \mathsf{Attempt}(A, \epsilon, P, Q, C_n, t_0, t_1)) \triangleq$
$\exists\Theta, \exists t' \cdot (t_0 \leqslant t' \leqslant t_1) \wedge Do(A, \epsilon, t_0, C_n \wedge \Theta) \wedge \mathsf{Hold}(Q, t')$.

[14] Cohen and Levesque [22] argued for such an extra argument.

That is, the successful doing of a performative means that the associated utterance event was done and the honest goal Q holds at some time before t_1.

Let $\chi(A, B, \ldots, t_0, t_1, \ldots)$ be a communicative act where A is the sender, B is the receiver, t_0 is the time to perform the act, t_1 is the deadline. Axiom 4 says that if a performative was done then both parties mutually believe that it was done.

Axiom 4. $\mathsf{SuccDone}(A, \chi(A, B, \ldots, t_0, t_1, \ldots)) \Rightarrow$
$\exists t' \leqslant t_1 \cdot \mathsf{MB}(\{A, B\}, \mathsf{SuccDone}(A, \chi(A, B, \ldots, t_0, t_1, \ldots)), t')$.

The semantics of elementary performatives is given by choosing appropriate formulas (involving mutual beliefs) to substitute for P and Q in the definition of Attempt. Inform is defined as the speaker's attempt to establish a mutual belief with the addressee about the speaker's goal to let the addressee know what the speaker knows.

Definition 8. $\mathsf{Inform}(A, B, \epsilon, p, t, t_a) \triangleq (t < t_a)?; \mathsf{Attempt}(A, \epsilon, P, Q, C_p, t, t_a)$,
where $P = \mathsf{MB}(\{A, B\}, p, t_a)$,
$Q = \exists t'' \cdot (t \leqslant t'' < t_a) \wedge \mathsf{MB}(\{A, B\}, \psi, t'')$,
$C_p = \mathsf{Bel}(A, p, t) \wedge \mathsf{Bel}(A, \mathsf{UBif}(B, p, t), t)$, where
$\psi = \exists t_b \cdot (t'' \leqslant t_b < t_a) \wedge \mathsf{Int.Th}(A, \mathsf{Bel}(B, \mathsf{Bel}(A, p, t), t_b),\ t, t_b, C_p)$.

This definition re-formulates the definition of `inform` in Section 2.1.2. When communication is reliable and agents trust each other, it is easy to establish the mutual belief about ψ required in the honest goal of Inform: agent B believes ψ upon receiving a message with content ψ from agent A; A knows this, and B knows A knows this, and so on. Here, unlike in re-formulating Attempt, we choose to use Int.Th rather than Pot.Int.Th to imitate the `GOAL` operator used in the definition of `inform` in Section 2.1.2. The reason is that the `GOAL` in `attempt` characterizes the speaker's ultimate goal while the `GOAL` in `inform` reflects the speaker's honest effort (the least to achieve), which can be better simulated by Int.Th. Pot.Int.Th is too weak to represent honest goals because potential intentions are subjected to reconciliation before being adopted as actual intentions. This does not lose flexibility because an agent can make decisions on communication before actually doing an Inform.

Fig. 7 shows the order of time points used in the definition of Inform. Both the time t in Definition 8 and the definition of `inform` in Section 2.1.2 refer to the same point. ψ in Definition 8 states that immediately before the performance of ϵ (i.e., at t) A intends that after ϵ (i.e., at t_b) B believes "A believes p before ϵ (i.e., at t)". Thus, the different characterizations of time are equivalent in the definitions of Inform and `inform`.

For all the theorems about communication to be established later, we assume that an agent will not choose to do anything that could thwart its attempts.

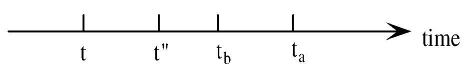

Fig. 7. The order of time points in Inform.

Assumption 3. $\mathsf{Attempt}(A, \epsilon, P, Q, C_p, t, t_1) \Rightarrow$
$$[\forall \alpha, t' \leqslant t_1 \cdot \mathsf{Bel}(A, \mathsf{CONF}(\alpha, Q, t', t_1, \Theta_\alpha, C_p), t') \Rightarrow \neg \mathsf{Do}(A, \alpha, t', \Theta_\alpha)].$$

Assumption 3 can be taken as the 'inertia' counterpart of Axiom 2 in Fig. 3. While Axiom 2 states that a sincere agent will try its best to bring about p and keep p holding, Assumption 3 only requires that an agent not knowingly make p false. More specifically, suppose that after agent A successfully performs a performative, the honest effort Q holds at some time point t' before the deadline t_1. Then, A will not do any action that conflicts with Q continuing to hold. Because it is assumed in Section 3.1 that beliefs/intentions persist by default, we can conclude that Q holds until t_1 from agent A's perspective.

Proposition 1. *Successful performance of the* Inform *act establishes between the sender and the addressee a mutual belief that the sender believes the informed proposition. Formally,*

$$\models \mathsf{SuccDone}(A, \mathsf{Inform}(A, B, \epsilon, p, t, t_a)) \Rightarrow \mathsf{MB}(\{A, B\}, Bel(A, p, t), t_a).$$

Proof. (1) Assume $\mathsf{SuccDone}(A, \mathsf{Inform}(A, B, \epsilon, p, t, t_a))$.

(2) By (1), Definition 7 and Definition 8, there exists a time $t_1 \leqslant t_a$ such that

$$\mathsf{MB}(\{A, B\}, \psi, t_1), \quad \text{where}$$

$$\psi = \exists t_b \cdot (t_1 \leqslant t_b < t_a) \wedge \mathsf{Int.Th}(A, \mathsf{Bel}(B, \mathsf{Bel}(A, p, t), t_b), t, t_b, C_p).$$

(3) A is assumed to be sincere, thus by Axiom 2 we have

$$\mathsf{Int.Th}(A, \mathsf{Bel}(B, \mathsf{Bel}(A, p, t), t_b), t, t_b, C_p) \Rightarrow$$
$$[\mathsf{Bel}(A, p, t) \wedge \mathsf{Int.Th}(A, \mathsf{Hold}(p, t_b), t, t_b, C_p)].$$

(4) From (2) and (3) we have $\mathsf{MB}(\{A, B\}, \mathsf{Bel}(A, p, t), t_1)$.

(5) By Assumption 2, we can conclude that $\mathsf{MB}(\{A, B\}, \mathsf{Bel}(A, p, t), t_a)$. □

A request with respect to action α is defined as the speaker's attempt to make both the speaker and the addressee believe that the speaker intends that the addressee commit to performing the action α. We reformulate the definition of `request` in Section 2.1.2 as:

Definition 9. $\mathsf{Request}(A, B, \epsilon, \alpha, t, t_a, \Theta_\alpha) \triangleq$
$(t < t_a)?; \mathsf{Attempt}(A, \epsilon, P, Q, C_p, t, t_a)$, where
$P = \mathsf{Do}(B, \alpha, t_a, \Theta_\alpha)$,
$Q = \exists t'' \cdot (t \leqslant t'' < t_a) \wedge \mathsf{MB}(\{A, B\}, \psi, t'')$,
$C_p = \mathsf{Bel}(A, \exists R_\alpha \cdot \mathsf{CBA}(B, \alpha, R_\alpha, t_a, \Theta_\alpha), t) \wedge$
$\quad \mathsf{Int.Th}(A, \mathsf{Do}(B, \alpha, t_a, \Theta_\alpha), t, t_a, \Theta_\alpha)$, where
$\psi = \exists t_b < t_a \cdot \mathsf{Int.Th}(A, \mathsf{Int.To}(B, \alpha, t_b, t_a, C_p \wedge \mathsf{Helpful}(B)), t, t_b, C_p)$.

The Request means that agent A at t has an attempt where (1) the ultimate goal is for B to perform α at t_a, and (2) the honest goal is to establish a mutual belief that agent A has an intention that agent B commit to performing α. All must be in appropriate contexts.

Both the time t in Definition 9 and in the definition of `request` in Section 2.1.2 refer to the same point. Formula ψ in Definition 9 states that immediately before the performance of ϵ (i.e., at t), A intends that after ϵ (i.e., at t_b), B intends to do α relative to A's wanting and B's being helpful. Thus, the different characterizations of time are equivalent in the definitions of Request and `request`.

According to the definition, agent A would be under no obligation to inform B that its request is no longer valid when A discovers that C_p no longer holds. Smith and Cohen defined another version of *Request* in terms of a *PWAG* (persistent weak achievement goal) rather than an intention [85]. That means, upon discovering that the goal has been achieved or become impossible to achieve, or that C_p is on longer true, agent A will be left with a persistent goal to reach mutual belief with B about the achievement or impossibility, which will free B from the commitment toward A regarding α. Rather than introducing a counterpart of *PWAG* into our framework, we prefer to encode such team-level obligations using an axiomization approach. Axiom 5 states that any agent intending others to be involved in a team activity should also adopt an intention to release those agents from the obligations whenever the intentional context no longer holds.

Axiom 5. $\models [\mathsf{Int.Th}(A, \mathsf{Int.To}(B, \alpha, t_1, t_\alpha, C_p \wedge C'), t, t_1, C_p) \wedge$
$(t < t_1 < t_\alpha) \wedge \mathsf{Bel}(A, \neg C_p, t)] \Rightarrow$
$\mathsf{Int.To}(A, \mathsf{Inform}(A, B, \epsilon, \neg C_p, t, t_1), t, t, C)$, *where*
$C = \mathsf{Int.Th}(A, \mathsf{Int.To}(B, \alpha, t_1, t_\alpha, C_p \wedge C') \wedge \mathsf{Bel}(A, \neg C_p, t)$.

The semantics associated with the receipt of a Request is a bit involved. In addition to realizing that the sender wishes him/her to commit to the action, the receiver can make certain deductions based upon knowledge of the semantics of Request. In particular, the receiver may deduce that the sender believes that there is a recipe the receiver could be following that would lead the receiver to bring about α. Note that the Request does not indicate which recipe the receiver should follow, only that the sender believes one exists. This is sufficient, though it does not guarantee that the receiver will actually perform α. If the receiver is not directly aware of such a recipe, it could lead the receiver to initiate a search for an appropriate recipe. If the receiver cannot find one as the sender expected, the receiver could free himself from the obligation and let the sender know the reason.

It is worth noting that action contracting is one important case Grosz and Kraus considered in defining SharedPlans [42]. Request is very useful in developing communication protocols that allow agents to contract out actions to others.

4. Information needs

4.1. Information and incomplete information

Information is defined in WordNet Dictionary as a message received and understood that reduces the recipient's uncertainty. We adopt the definition described in the Open Archival Information System (OAIS) [70]: information is "any type of knowledge that can be exchanged, and it is always represented by some type of data".

To represent information, we start with the identifying reference expression (IRE), which is used in FIPA [1] to identify objects in the appropriate domain of discourse. For any n-ary predicate symbol p, it will be written in the form $p(\vec{?x}, \vec{c})$, where $\vec{?x}$ is a set of variables and $\vec{c}$ is a set of constants in appropriate domains. For example,

$$\mathit{authorship}(\{?f, ?h, ?m, ?v\}, \text{‘}\mathit{Reasoning\ about\ knowledge}\text{’})$$

represents that “Who are the four authors of the book *Reasoning about knowledge*”.

IRE is written using one of three referential operators defined in the FIPA specification, as follows:

(1) $(\mathit{iota}\ \vec{?x}\ \ p(\vec{?x}, \vec{c}))$ refers to “the collection of objects, which maps one-to-one to variables in $\vec{?x}$ and there is no other solution, such that p is true of the objects”; the term is undefined if for any variable in $\vec{?x}$ no object or more than one object can satisfy p (together with substitutions for other variables);
(2) $(\mathit{all}\ \vec{?x}\ \ p(\vec{?x}, \vec{c}))$ refers to “the collection of sets of all objects that satisfy p; each set (could be an empty set) corresponds one-to-one to a variable in $\vec{?x}$”; and
(3) $(\mathit{any}\ \vec{?x}\ \ p(\vec{?x}, \vec{c}))$ refers to “any collection of objects, which maps one-to-one to variables in $\vec{?x}$, such that p is true of the objects”; it is undefined if no collection of objects (substituents of variables in $\vec{?x}$) can satisfy p.

For simplicity of notation, we will omit the operator *any* when the context is clear and its absence will cause no confusion. Hence, expressions of the form $(\mathit{any}\ \vec{?x}\ \ p(\vec{?x}, \vec{c}))$ can be simplified to $p(\vec{?x}, \vec{c})$. These three forms of IREs are expressive enough to specify agents' needs for information regarding the values of parameters in a formula.

Throughout this paper, we consider two forms of information: factual information and referential information. A factual information refers to a fact while a referential information may refer to a collection of facts. For instance, “Tom has done his homework” is a factual information, and “Who have done their homework–Tom and Tim” is a referential information. Factual information is represented by a proposition (a predicate with constant arguments). Referential information is represented by clauses of the form, $\mathit{Refer}(\mathit{ire}, \mathit{obj})$, where *ire* is an identifying reference expression, and *obj* is the set of objects bound to the variables in *ire*. In the following, we will use I $(I', I_1, \ldots)$ to represent the information to be communicated. When I refers to a proposition, the sender is informing the receivers that the predicate is true. When I refers to $\mathit{Refer}(\mathit{ire}, \mathit{obj})$, the sender is informing the receivers that *obj* contains the collection (of sets) of objects that satisfy the p to which the *ire* refers according to the sender's belief base.

Information can be classified along several dimensions. For instance, we can distinguish quasi-static information which seldom changes once acquired (e.g., recipes for actions) from dynamic information. Depending on how information is acquired, there are *observable* information, *computable* information (e.g., by inference rules), and *a priori* information (common domain knowledge). We mainly focus on dynamic, observable information.

4.1.1. Incomplete information

Normally, in referential information, all the variables are bound to values, e.g., $\mathit{Refer}(\mathit{threat}(e1, ?\mathit{loc}, ?\mathit{dir}, ?\mathit{num}), (\mathit{area}_1, \mathit{south}, 100))$. In multi-agent systems, information exchange also involves incomplete information (with unbound variables). This is of special significance in teamwork settings. For instance, in generating shared plans [43], parameter identification for team activities depends on the exchange of incomplete information.

Hence, we assume agents are capable of recording and manipulating incomplete information.[15] For example, suppose that in agent B's belief base, agent A is in the northeast and there is a piece of incomplete information: $\mathit{threat}(e, \mathit{area}_4, \mathit{northeast}, ?\mathit{num})$, which means that agent B has observed an enemy unit e with an unknown number of enemies in area 4, moving northeast. There are many reasons for exchanging incomplete information rather than waiting until it becomes complete: agent B may never be able to get the number of e for lack of observability; agent A may already have the number of e from another teammate; A may be able to deal with the threat even when it is incompletely specified, etc.

Given a predicate symbol p, if $p(?v_1, \ldots, ?v_i, c_1, \ldots, c_j)$ belongs to agent A's belief base, it means that agent A believes that there exist some unknown objects $?v_m$ $(1 \leqslant m \leqslant i)$, which together with the already identified objects c_n $(1 \leqslant n \leqslant j)$, have the relation p. Generally, in a piece of incomplete information p, it is not necessary that all the constant arguments of p come after all the variable arguments (not being identified yet). In addition, a variable could occur more than once in p when p denotes a complex relation. For example, suppose $p_1(?x, ?y, ?z)$ denotes such a relation: $?x$ is a logistics person, $?y$ is in the army, and $?z$ is a relative of both $?x$ and $?y$. As the state of affairs changes, agent A may acquire information $p_1(?x, ?x, \mathit{Aaron})$, which means that A believes that *Aaron* has a relative who is a logistics person in the army. Thus, to simplify the following discussion, we assume that any incomplete information can be represented in a normative form $H(\vec{v}, \vec{c})$, where $\vec{v}$ and $\vec{c}$ are vectors of variable identifiers and constant identifiers respectively, such that (1) the original multiple appearances of variables or constants are removed; and (2) the order of elements in $\vec{v}$ and $\vec{c}$ does not matter. For example, $p_1(?x, ?x, \mathit{Aaron})$ can be normalized as $p_1(\{?x\}, \{\mathit{Aaron}\})$. We also assume certain mapping information is preserved in the normative representation so that normative incomplete information can be properly de-normalized.[16]

Social inference trees (cf. Section 3.4) can be leveraged to generate incomplete information. For example, the social inference tree pictured in Fig. 6 can be used to generate $\mathit{threat}(e, \mathit{area}_4, \mathit{northeast}, ?\mathit{num})$ by fusing the information: $\mathit{IsEnemy}(e)$, $\mathit{At}(e, \mathit{area}_4)$, $\mathit{Dir}(e, \mathit{northeast})$. Also, several pieces of incomplete information can be combined if they are complementary. Continuing the above example, suppose agent A also gets another incomplete threat $\mathit{threat}(e, ?\mathit{loc}, ?\mathit{dir}, 100)$ from agent C (i.e., the enemy unit e has 100 enemies, but their location and moving direction are unknown to C). Then $\mathit{threat}(e, \mathit{area}_4, \mathit{northeast}, 100)$ can be derived by A.

[15] In implementation, the inference engine should treat incomplete information and complete information (e.g., facts) separately.

[16] In essence, the normalization can be achieved solely by defining new predicate symbols.

4.2. The concept of information needs

An information need may state that the agent needs to know the truth value of a proposition. For instance, suppose a person sends a query $\mathit{Weather}(\mathit{Cloudy}, \mathit{Today})$ to a weather station. The weather station will realize that the person wants to know, at least literally,[17] whether it is cloudy today. More often than not, an agent may want any information that matches his/her constraints, rather than simply querying whether a specific proposition is true or false. In particular, an agent may want to know the values of some arguments of a predicate that would make the predicate true [87]. For example, a person may send a query $\mathit{Weather}(?x, \mathit{Today})$ to a weather station. This will trigger the weather personnel, if willing to be helpful, to inform the person about a change in the weather conditions whenever necessary.

Thus, in regard to information, an expression of information needs may also be in one of two forms: a factual proposition or a reference expression (which actually specifies a class of information). In what follows, N is used to refer to an information need-expression, and $\mathit{pos}(N)$ $(\mathit{ref}(N))$ is true if N is a proposition (reference expression).

Now we come to the representation of information needs. Obviously, an information need should specify the need-expression as well as the information consumer (needer). Typically, a need becomes meaningless after a certain point when some event happens. For instance, an agent may no longer need to know the location of enemy units e if e has already been defeated. Thus, information needs often have an associated time limit. In addition, a need is only applicable in certain contexts. The contexts of a need may serve as relativizing conditions [56] or describe the reason for adopting the need. For instance, the context of an information need may include the context of the needer's relevant intentions. Those teammates who know the information need of some agent will consider helping the agent as long as the context of the information need remains true. Later, the contexts of information needs will be considered in transforming information needs of teammates to intentions to assist them (refer to Section 6). The contexts also will be used in constructing the contexts for need-driven communicative actions (refer to Section 7,8).

To combine the above factors, a modal operator $\mathsf{InfoNeed}(A, N, t, C_n)$ is introduced to represent information needs. In cases where N is a proposition, the operator means that agent A needs to know the truth value of N by t under the context C_n; in cases where N is a reference expression, the operator means agent A needs to know those objects that satisfy the reference expression N.

Note that for at least two reasons the notion of information needs cannot be defined simply as intentions of beliefs, say $\mathsf{Int.Th}(A, \mathsf{Bel}(A, I, t'), t, t_b, C_n)$. First, since the $\mathsf{Int.Th}$ operator is subject to numerous axioms (e.g., axioms constraining which other intentions an agent might subsequently adopt), using $\mathsf{Int.Th}$ to represent an information need might be too restrictive. For instance, without $\mathsf{InfoNeed}$, it is cumbersome to clearly express the situation where an agent has an information need of which the agent is unaware. Second, an intention-that involves a commitment to means-ends reasoning which may be inapplicable to mere information needs. Moreover, suppose that $Q = \mathsf{Int.Th}(A, \mathsf{Bel}(A, I, t'), t, t_b, C_n)$

[17] Except for indirect speech acts [80].

(IN1) $\models \mathsf{InfoNeed}(A, N, t', C_n) \wedge (t' \geqslant t'') \Rightarrow \mathsf{InfoNeed}(A, N, t'', C_n)$;
(IN2) $\models \mathsf{InfoNeed}(A, N, t', C_n) \equiv \mathsf{InfoNeed}(A, \neg N, t', C_n)$, where N is a proposition;
(IN3) $\models \mathsf{InfoNeed}(A, H(\vec{?x}, \vec{c}), t', C_i) \Rightarrow \forall ?y \in \vec{?x}, \forall k \in dom(?y, H(\vec{?x}, \vec{c})) \cdot \mathsf{InfoNeed}(A, H(\vec{?x} \backslash ?y, \vec{c} \oplus k), t', C_i)$;
(IN4) $\models \mathsf{InfoNeed}(A, N_1 \wedge N_2, t', C_i) \Rightarrow \mathsf{InfoNeed}(A, N_1, t', C_i) \wedge \mathsf{InfoNeed}(A, N_2, t', C_i)$;
(IN5) $\models \mathsf{InfoNeed}(A, N_1 \vee N_2, t', C_i) \Rightarrow \mathsf{InfoNeed}(A, N_1, t', C_i) \wedge \mathsf{InfoNeed}(A, N_2, t', C_i)$;
(IN6) $\models \mathsf{InfoNeed}(A, N_1, t', C_i) \wedge (N_2 \Rightarrow N_1) \Rightarrow \mathsf{InfoNeed}(A, N_2, t', C_i)$, where $N_2 \not\equiv \mathit{False}$;
(IN7) $\models \mathsf{Bel}(A, H(\vec{?x}, \vec{c}), t) \Rightarrow \mathsf{Bel}(A, \exists t', C' \cdot \mathsf{InfoNeed}(A, H(\vec{?x}, \vec{c}), t', C'), t)$.

Fig. 8. The axioms characterizing InfoNeed: IN1–IN7.

represents an information need of agent A, and $\mathsf{Bel}(B, Q, t)$ holds. Then, agent B would not choose to help A, considering that agent A itself could commit to certain means-ends reasoning to acquire I (e.g., ask some other teammate). To enable B to help A in such a case, it is desirable to introduce the InfoNeed operator.

In the rest of this section, instead of providing an explicit semantics for InfoNeed, we choose to give an axiomatization for it. First, InfoNeed is closed temporally into the past. Axiom IN1 in Fig. 8 states that, if an agent has a need regarding N by t', it also needs N any time before t'. However, as far as proactive communication is concerned, only 'future' needs make sense. Thus, when Axiom IN1 is applied, only the needs backward up to the current time t are considered.

Axiom IN2 states that in the case where the need-expression is a proposition, the information need is insensitive to negation.

To explain Axiom IN3, we first define the notations to be used. Given a vector $\vec{o}$ of identifiers and any identifier d, define

$$\vec{o} \setminus d \triangleq \begin{cases} (o_1, \ldots, o_i, o_{i+1}, \ldots, o_k) & \text{if } \vec{o} = (o_1, \ldots, o_i, d, o_{i+1}, \ldots, o_k), \\ \vec{o} & \text{if } d \text{ not occur in } \vec{o}, \end{cases}$$

$$\vec{o} \oplus d \triangleq \begin{cases} (o_1, \ldots, o_k, d) & \text{if } \vec{o} = (o_1, \ldots, o_k), \\ (d) & \text{if } \vec{o} \text{ if empty.} \end{cases}$$

Given $P(\vec{?x}, \vec{c})$ and $?y \in \vec{?x}$, let $dom(?y, P(\vec{?x}, \vec{c}))$ be the value domain of the variable $?y$ with respect to the predicate P. That is, $dom(?y, P(\vec{?x}, \vec{c}))$ is a set of values such that, if any of the values is substituted for $?y$, there will still be values for the rest of the variables in $\vec{?x}$ that will make P true.

Axiom IN3 states that for information needs involving reference need-expressions or normalized incomplete information, abstract needs imply more concrete needs. For example, if agent B knows that agent A needs information about threats of the form $\mathit{Threat}(?e, ?loc, ?dir, ?num)$, B may rationally assume that A is also interested in more concrete information like: $\mathit{Threat}(e1, ?loc, ?dir, 100)$, even though the information is still incomplete.

Axioms IN4 and IN5 state that InfoNeed distributes over conjunction, but not over disjunction because the truth value of $N_1 \vee N_2$ may depend on the truth values of both N_1 and N_2.

Axioms IN6 states that weaker information needs entail stronger ones. For example, believing that agent B needs N_1 and "$N_2 \Rightarrow N_1$" is commonly known, it is rational for agent A to assume that B also needs N_2. Otherwise, B could have derived N_1 by itself. However, Axioms IN6 is actually too strong; it implies that agent A needs all unsatisfiable formulas. But on the other hand, it is acceptable in practical reasoning systems if an agent is prohibited from applying Axiom IN6 to an inference rule when the antecedent is found unsatisfiable.

Oftentimes, an agent may get incompletely specified information from its teammates, who believe the beneficiary agent will need the information even though it is incomplete. If an agent really needs the information which is currently of incomplete form, the agent can generate an information needs from the incompletely specified information so that it can refine the information when the missing part becomes available later. Axiom IN7 states that an agent can derive information needs by reflecting on incomplete information (e.g., reference expressions in which $\vec{?x}$ is not empty).

In general, an agent may not be able to figure out its own information needs for many reasons (e.g., lack of inference knowledge). Thus, we have

$$\mathsf{InfoNeed}(A, N, t', C_i) \not\Rightarrow \mathsf{Bel}(A, \mathsf{InfoNeed}(A, N, t', C_i), t),$$

$$\neg\mathsf{InfoNeed}(A, N, t', C_i) \not\Rightarrow \mathsf{Bel}(A, \neg\mathsf{InfoNeed}(A, N, t', C_i), t).$$

More axioms will be introduced in later sections. Those axioms can be taken as characterizations of the relationships between InfoNeed, Bel, and intentions.

We now define a generated set. For any set of formula C, let $Needs(C)$ be a set of need-expressions generated from C:

1. $p \in Needs(C)$, if $p \in C$ is a proposition;
2. $(any\ \vec{?x}\ \ p(\vec{?x}, \vec{c})) \in Needs(C)$, if $p(\vec{?x}, \vec{c}) \in C$.[18]

For example, given a set

$$C = \{IsEnemy(?e), At(?e, ?loc), HaveSupply(Self)\},$$

then

$$\begin{aligned} Needs(C) = \{&(any\ ?e\ IsEnemy(?e)),\\ &(any\ (?e\ ?loc)\ At(?e, ?loc)),\\ &HaveSupply(Self)\}. \end{aligned}$$

[18] Depending on domains, need-expressions of the form $(iota\ \vec{?x}\ \ p(\vec{?x}, \vec{c}))$ or $(all\ \vec{?x}\ \ p(\vec{?x}, \vec{c}))$ can also be generated. For instance, if α is a joint action where some doer should be exclusively identified, an *iota* expression is preferred. An *all* expression is suitable if all objects that can be substituted for variables in $\vec{?x}$ will be needed in the performance of α.

Need-expressions can be generated from action (or plan) preconditions. For an action α, we write $Needs_A(\alpha)$ to refer to $Needs(pre_A(\alpha))$, where A is an agent. The generated need-expression sets will be used in Section 5 to derive the information needs anticipated for teammates.

4.3. Levels of information needs

The notion of social inference tree (cf. Section 3.4) helps in handling levels of information needs. Because of the axioms IN6 and IN4 (cf. Fig. 8) of InfoNeed, there may exist information needs at different levels but for the same purpose. For instance, suppose agent A recognized that an enemy unit e is approaching agent B, who needs to react to the threat (say, perform *RemoveThreat*) no later than time t'. Now assume that agent A believes

$$\mathsf{InfoNeed}(B, \mathit{Threat}(e, ?loc, ?dir, ?num), t', C),$$

where C records A's explanation for the need. Then, by Axiom IN6, A will believe $\mathsf{InfoNeed}(B, \mathit{IsEnemy}(e) \wedge \mathit{At}(e, ?loc) \wedge \mathit{Dir}(e, ?dir) \wedge \mathit{Number}(e, ?num), t', C)$, and by Axiom IN4, A will also believe

$$\mathsf{InfoNeed}(B, \mathit{IsEnemy}(e), t', C),\ \mathsf{InfoNeed}(B, \mathit{At}(e, ?loc), t', C),$$
$$\mathsf{InfoNeed}(B, \mathit{Dir}(e, ?dir), t', C),\ \mathsf{InfoNeed}(B, \mathit{Number}(e, ?num), t', C).$$

Such proliferation process may continue in a top-down way along the inference tree and result in several levels of information needs. Redundant assistance may occur if A attempts to satisfy all these information needs.

Here, social inference trees can be leveraged to preclude the consideration of redundant information needs. The idea is to consider information needs first from the most abstract level. Only when an agent cannot satisfy the information needs at level i (i.e., there is critical information unknown),[19] will it consider those needs at level $i + 1$.

For the example shown in Fig. 6, suppose that, as the doer of *RemoveThreat*, agent B needs threat information, and agent A has identified an enemy unit: $\mathsf{Bel}(A, \mathit{IsEnemy}(e1), t)$. To help B with its information needs about threat, A can first check whether $\mathit{Threat}(e1, ?loc, ?dir, ?num)$ holds or not. Assume that A's belief base includes facts: $\mathit{At}(e1, area_2, \mathit{NOW})$, $\mathit{Dir}(e1, north)$, and $\mathit{Number}(e1, 80)$ (which may be observed by A itself or informed by others). Then A could successfully identify a threat by fusing the lower-level information together. In such a case, A can simply deliver the identified threat information to B, instead of going further to satisfy B's lower-level information needs along the inference tree.

In complex cases, an agent may choose to deliver information to satisfy a needer's information needs at multiple levels. Continue the above example, assuming that A's belief base only includes information: $\mathit{At}(e1, area_2, \mathit{NOW})$ and $\mathit{Number}(e1, 80)$. In this case, A could only identify an incomplete threat $\mathit{Threat}(e1, area_2, ?dir, 80)$. As well as delivering this incomplete information to B, A may want to figure out what hindered it from inferring information regarding the enemy's moving direction, and to help satisfy B's lower-level

[19] When incomplete information is allowed, an agent can *partially* satisfy an information need.

information needs as far as it can. This reasoning process can be carried along inference trees, and breadth-first algorithms can be designed such that agents will not offer redundant assistance regarding others' information needs. Hierarchically considering others' information needs is of great significance especially when information consumers only have limited cognitive capacity, because it allows information consumers to always consider higher-level information first and ignore the less important or irrelevant information upon being overloaded.

4.4. Types of information needs in agent teamwork

A team is a set of agents having a shared objective and a shared mental state [25]. In the SharedPlans theory, shared objectives are given in terms of intentions-that (a team's wanting to do a certain team action), and shared mental states are reflected by partial shared plans (PSP) and full shared plans (FSP). As well as establishing requisite mutual beliefs and ensuring the satisfaction of shared objectives, communication in effective agent teams also plays a central role in satisfying others' information needs. In agent teamwork, we distinguish four types of information needs usually emerging in the pursuit of team or individual goals.

Action-performing information need. This type of information needs enables an agent to perform simple or complex actions, the performance of which can contribute to the whole team. Typically, an action-performing information need is derived from the preconditions of the action. For instance, in the example given in Section 3.4, *Threat* is a kind of action-performing information need with respect to action *RemoveThreat*.

Decision-making information need. As well as domain actions, those information needs emerging in the mental action *decision-making* are of particular interest. Without loss of generality, we assume complex recipes (e.g., for team activities) may contain 'decision-points', and a decision-point can have several branches specifying alternative courses of action (COA) that agents can follow to achieve a certain goal. In the terminology of the SharedPlans theory, each potential choice (i.e., COA) of a decision-point can be taken as a potential intention, and a decision maker agent ought to select one from the collection of potential intentions and upgrade it to a full-fledged intention. The reasoning about decision-making information needs allows team members to help the decision maker select a better course of action.

Typically, each branch of a decision-point can be associated with some *preference* constraints. For instance, in reactive planning [7], preference criteria can be specified for each of the plans achieving the same goal. The collection of preference constraints involved in a decision-point are important factors that affect the quality of decision making; the more information relevant to the preference constraints is available, the better the decision maker can evaluate the potential options. For instance, in fire-rescue domains, firefighters normally use water to extinguish fires. Suppose a building containing materials that react with water is on fire. It is crucial firefighters know of the contents so that they can choose a better course of action.

Goal-protection information need. This type of information needs allows an agent to protect a goal (intention-that) from becoming unachievable. Information regarding potential threats to the accomplishment of a committed goal belongs to this category; knowing such information will help an agent adjust its behavior to remove or avoid the threat. For instance, suppose that the goal of a logistics unit is to *transport ammunition to the front*, and approaching enemy units pose a threat to the accomplishment of the logistics unit's goal. Then, the information about the enemy units (e.g., moving direction) is needed by the logistics unit to protect its goal. Knowing the approaching threat, the logistics unit could adjust its supply route to keep its goal achievable.

Information about conflicts between potential intentions and full-fledged intentions also belongs in this category; knowing such information will help an agent rationally postpone or drop those potential intentions that may cause conflicts. For instance, suppose that an agent has an intention to achieve p by doing action α, and at the same time it has a potential intention to do action β. Knowing that there exists a resource conflict between α and β will enable the agent to drop the potential intention, which, if adopted as an intention, would impede the achievement of p.

Goal-escape information need. Because a goal ultimately becomes achieved, unachievable or irrelevant [20], this type of information is needed by an agent to drop impossible or irrelevant goals. A goal is achievable and relevant only when its context holds. Thus, typically goal-escape information needs can be derived from the context of the goal under concern. If any part of a goal context no longer holds, an agent who observed this fact needs to inform the other teammates involved in pursuing the same goal, so that they can abandon this impossible or irrelevant goal.

5. Anticipating information needs

To proactively deliver information to teammates, the information providers should be aware of the teammates' information needs. There are at least two ways to achieve this. An information provider can wait for the information consumers' articulation of their information needs. Alternatively, a provider can proactively *anticipate* teammates' potential information needs based on certain shared mental models.

The concept of recipes in the SharedPlans theory offers us the basis for studying agents' capabilities of anticipating teammates' information needs. In this section, we propose some axiom schemas for agents to anticipate the different types of information needs identified in the previous section. The ways of anticipating others' information needs proposed here lay the foundation for developing algorithms for agents to reason dynamically about information needs of their teammates.

5.1. Action-performing information needs

Oftentimes an agent cannot proceed due to obstacles to individual or team actions. Here we focus on informational obstacles, which refer to the prerequisite information for per-

Axiom 6 (Action-performing information need).
$\forall A, B \in T \subseteq TA, \alpha, C_\alpha, \gamma, t, t' \geqslant t, \forall N \in Needs_A(\alpha), \forall R_\gamma \in recipe(\gamma)$,
1. $\mathsf{Bel}(A, \mathsf{Int.To}(B, \alpha, t, t', C_\alpha), t) \Rightarrow \mathsf{Bel}(A, \mathsf{InfoNeed}(B, N, t', C_{n1}), t)$,
2. $[\mathsf{Bel}(A, \mathsf{Pot.Int.To}(B, \alpha, t, t', C_\alpha), t) \wedge$
$\mathsf{Bel}(A, has.recipe(T, \gamma, R_\gamma, t) \wedge \alpha \in R_\gamma, t)] \Rightarrow$
$\mathsf{Bel}(A, \mathsf{InfoNeed}(B, N, t', C_{n2}), t)$, *where*
$C_{n1} = C_\alpha \wedge \mathsf{Int.To}(B, \alpha, t, t', C_\alpha)$,
$C_{n2} = C_\alpha \wedge has.recipe(T, \gamma, R_\gamma, t) \wedge \alpha \in R_\gamma \wedge \mathsf{Pot.Int.To}(B, \alpha, t, t', C_\alpha)$.

Fig. 9. The axiom for deriving action-performing information needs.

forming an action, and we assume they are specified as part of the preconditions of the action.

Intuitively, we say that an agent A can anticipate that another agent B will need to know the pre-requisite information for performing an action if A recognizes that B has a (possibly potential) intention to do that action. Formally, Axiom 6 in Fig. 9 states that agent A believes that agent B will need the information described by N by time t', if A believes that B is (potentially) intending to perform action α at time t' under context C_α. The context of the information need consists of C_α and B's (potential) intention to perform α.

To justify this axiom, some issues deserve further explanation. First, Axiom 6 should not be understood as "any agent that might end up doing an action is considered to have a (potential) intention to do that action". An agent could do actions either reactively or deliberatively. Axiom 6 just offers one way for an agent to anticipate others' action-performing information needs. Knowing others' intentions or potential intentions helps the anticipation. Otherwise, an agent cannot help teammates unless being explicitly requested to do so.

The second question is how an agent gets to know others' intentions. We assume agents in *TA* as a team are either evolving or acting on some shared plans that have been collaboratively generated for some team task. After members of a team have agreed with each other on some specific recipe for doing an action, even though they individually might have different partial views of the recipe, each of them should have some minimum knowledge regarding the evolving recipe. Such knowledge could include the decomposition (at least at the immediate next level [47]) of the actions he/she is committed to or is jointly committed to with teammates. An agent could also know who are the assigned doers of already resolved subactions, as well as the performance sequence of those subactions. Therefore, an agent can infer the actual intentions-to from his/her partial view of the recipe-tree on which all the teammates are working.

An agent could also infer teammates' potential intentions-to from the evolving shared recipe they are working on. A critical point made in the SharedPlans theory is that planning is interleaved with acting. Usually, a group of agents may not have a complete plan until after they have done some of the actions in the partial recipe [42]. This means agents can act on partial recipes although there are some actions that still need to be resolved (e.g., through task allocation) or decomposed further. For those unresolved actions, an anticipating agent cannot surely know who will be the actual performers; the best it can do is to assume that all those agents with the requisite capability would be the potential performers. Thus, the anticipating agent could imagine that all the potential performers of an unresolved subaction are potentially intending to do the subaction. Note that these potential intentions may only exist in the anticipating agent's imagination, which serves to activate the anticipating agent to provide help proactively. Also, not all potential intentions are useful in deriving information needs. Part 2 of Axiom 6 requires that agent A infer agent B's action-performing information needs only if B's potential intention is relevant to a shared recipe of some team activity γ that involves both A and B. In other words, the action α should be part of R_γ.

Based on the above discussion, the following axiom is added to our framework to allow agents to derive teammates' potential intentions.

Axiom 7. $\forall A, B \in T \subseteq TA, \alpha, \gamma, t, t_\alpha \geqslant t, \forall R_\gamma \in \mathit{recipe}(\gamma), \forall R_\alpha \in \mathit{recipe}(\alpha)\cdot$
$[\mathsf{Bel}(A, \mathit{has.recipe}(T, \gamma, R_\gamma, t), t) \wedge$
$\quad \mathsf{Bel}(A, \alpha \in R_\gamma, t) \wedge$
$\quad\quad \mathsf{Bel}(A, \mathsf{CBA}(B, \alpha, R_\alpha, t_\alpha, \Theta_\alpha), t)] \Rightarrow$
$\quad\quad\quad \mathsf{Bel}(A, \mathsf{Pot.Int.To}(B, \alpha, t, t_\alpha, C_\alpha), t)$,[20] *where*
$C_\alpha = \Theta_\alpha \wedge \mathit{has.recipe}(T, \gamma, R_\gamma, t) \wedge \alpha \in R_\gamma$.

Third, different agents may have different recipes for an action. Even though agents do share some recipes, an agent may not know exactly which recipe will be used by another agent to achieve its goal. We relax this in Axiom 6 by letting the anticipating agent only consider those recipes it is aware of (refer to the definition of $\mathit{pre}_A(\alpha)$ and $\mathit{Needs}_A(\alpha)$). This means, an agent is only using information it has about α to determine the information needs of others. One drawback is that the anticipated information needs may not reflect the real information needs. This can be improved by allowing agents to exchange expertise on recipes. On the other hand, as a helping behavior, anticipating others' information needs does not always have to be precise. In cases where the beneficiary agent realizes its needs were incorrectly predicted, it may trigger certain conversation sessions, which allow the anticipating agent to refine its model regarding others' information needs.

Fourth, Axiom 6 indicates that an agent may generate one information need for any need-expression in $\mathit{Needs}_A(\alpha)$. Whenever communication bandwidth permits, the axiom could be leveraged to enhance team-wide situation awareness. However, most multi-agent systems only have restricted communication bandwidth. Moreover, according to the definition of *Needs*, the set of need-expressions generated for an action could be large. Thus, certain assumptions common to all the teammates need to be employed to preclude unnecessary assistance. As far as action-performing is concerned, an agent may not proceed when lacking some prerequisite information for performing an action; it may simply wait until more information becomes available (e.g., being informed by teammates). Thus, if "wait" is taken as a common assumption among team members, it is unnecessary for teammates to inform an action performer of the negation of information related to the action

[20] It is possible that agent A may be informed that agent B had already discarded the COA involving α for some reason (e.g., due to actions/intentions conflicts). A weaker version of Axiom 7 can be given to incorporate such a possibility.

preconditions. For example, suppose that agent A requires p to be true prior to performing α. Agent B need not inform A about $\neg p$ when it believes p is false. Alternatively, an agent may proceed even lacking some prerequisite information. If this is commonly assumed, agent B may want to let A know $\neg p$, hoping that A could then choose a more appropriate recipe.

On the other hand, even though the set of anticipated information needs is large, an agent may not service all of them using proactive communications. In practice, a decision theoretic approach can be employed to achieve selective communication [91]. Such decisions can be influenced by various factors including the possibility that the prospective beneficiary agent already knows the information, the possible side-effects (e.g., overheard by opponents) of sending the information, and the cost of communication bandwidth. However, deciding on whether to help others with the anticipated information needs is very complex in its own right and, in general, beyond what is considered in this paper.

Fifth, the contexts, C_{n1} and C_{n2}, in Axiom 6 are composed of agent B's (potential) intention under A's concern and the context of the (potential) intention. This is easy to justify because the anticipated information need will make no sense if A no longer believes B has the (potential) intention, or from A's viewpoint, the (potential) intention is no longer relevant. One thing worth noting here is that in Axiom 6 the context C_α actually refers to A's estimation, which may be different from the actual intentional context of B. The question is to what extent an agent could approximate its teammates' intentional contexts. In the pursuit of higher-level joint goals, there are various reasons for an agent to hold a (potential) intention. In Section 3.2, we identified four possible uses of contexts: constraints, trace of explanation, attention management, and social specification. Among these four components, the social specification part of C_α is typically taken as common knowledge to the whole team; the trace of explanation part can be better estimated in cases where both A and B are working on the same team activity (i.e., they have shared plans and shared recipes). However, it is harder for an agent to approximate the constraints and the attention-management parts because normally they depend on the intention holder itself. Sharing intentions or learning meta-information regarding teammates' capabilities, capacities or strategies helps in improving the approximation.

The following lemma indicates that under certain contexts, an agent can anticipate others' action-performing information needs from their intentions-that.

Lemma 1. $\forall A, B \in T \subseteq TA$, $\forall \phi, \alpha, \gamma, C_\phi, \Theta_\alpha, t,\ t' \geqslant t,\ t'' \geqslant t'$, $\forall N \in Needs_A(\alpha)$, $\forall R_\gamma \in recipe(\gamma)\cdot$

$$[\mathsf{Bel}(A, \mathsf{Int.Th}(B, \phi, t, t'', C_\phi), t) \wedge$$
$$\mathsf{Bel}(A, \mathit{has.recipe}(T, \gamma, R_\gamma, t) \wedge \alpha \in R_\gamma, t) \wedge$$
$$\mathsf{Bel}(A, \neg\mathsf{Bel}(B, \phi, t), t) \wedge$$
$$\mathsf{Bel}(A, \mathit{Lead}(B, \alpha, \phi, t, t', \Theta_\alpha), t)]$$
$$\Rightarrow \mathsf{Bel}(A, \mathsf{InfoNeed}(B, N, t', C_n), t), \textit{ where}$$
$$C_n = \Theta_\alpha \wedge C_\phi \wedge \mathit{has.recipe}(T, \gamma, R_\gamma, t) \wedge \alpha \in R_\gamma \wedge \mathsf{Pot.Int.To}(B, \alpha, t, t', \Theta_\alpha \wedge C_\phi).$$

Proof. $\mathsf{Bel}(A, \mathsf{Pot.Int.To}(B, \alpha, t, t', \Theta_\alpha \wedge C_\phi), t)$ follows from the antecedents and Axiom 3. $\mathsf{Bel}(A, \mathsf{InfoNeed}(B, N, t', C_n), t)$ follows from Axiom 6(2).

Axiom 8 (Decision-making information need).
$\forall A, B \in TA, \phi, C_\phi, t, t' \geqslant t, t'' > t', N, \Omega\cdot$
$[\mathsf{Bel}(A, N \in \mathit{reckon}_A(B, \Omega, \phi), t) \wedge$
$(\bigwedge_{\alpha_i \in \Omega} [\mathsf{Bel}(A, \mathsf{Pot.Int.To}(B, \alpha_i, t, t', C_{\alpha_i}), t) \wedge$
$\mathsf{Bel}(A, \mathsf{Int.Th}(B, \phi, t, t'', C_\phi) \in C_{\alpha_i}, t)])] \Rightarrow$
$\mathsf{Bel}(A, \mathsf{InfoNeed}(B, N, t', C_n), t)$, *where*
$C_n = C_\phi \wedge \mathsf{Int.Th}(B, \phi, t, t'', C_\phi) \wedge \bigwedge_{\alpha_i \in \Omega} \mathsf{Pot.Int.To}(B, \alpha_i, t, t', C_{\alpha_i})$.

Fig. 10. The axiom for deriving decision-making information needs.

5.2. Decision-making information needs

An agent A may be able to recognize the information needs of another agent B if A knows B is facing a critical point for choosing its next course of action. This happens when A knows the possible choices B is considering, and A knows some information that may help B make a better decision. Being helpful, A will assume B needs the information.

For example, suppose in fire-rescue domains, N is *has_chemical*$(T1, M1)$, which means the building $T1$ contains a chemical material $M1$ that can produce noxious vapor when reacting with water. As an engineer of the building, A knows the fact N. But the firefighters, who use water to extinguish fires by default even though they have different means to choose from, are unaware of this fact. Herein, the firefighters have a goal to put out the fire on $T1$ with minimum loss and have a potential intention to extinguish the fire using water. In such a case, A is obligated to let the fighters know N, so that they can drop the potential intention of extinguishing the fire using water and adopt another means instead.

Let Ω denote the set of possible choices (i.e., complex actions) of a decision point, and each candidate action in Ω be associated with certain cues.[21] The more information regarding these cues that is available, the better an agent can evaluate the utility of choosing this option. In general, we use *reckon*(B, Ω, ϕ) to refer to the set of information (need-expressions generated from cues associated with the actions in Ω) used by B in evaluating the utilities of choices with respect to the goal state ϕ, and we use $\mathit{reckon}_A(B, \Omega, \phi)$ to refer to A's approximation of *reckon*(B, Ω, ϕ) based on what is known to agent A about the decision point. The need-expressions in $\mathit{reckon}_A(B, \Omega, \phi)$ will be evaluated with respect to agent A's knowledge. For example, suppose *backup*$(\alpha, ?num)$ is in $\mathit{reckon}_A(B, \Omega, \phi)$, where $\alpha \in \Omega$. This means that agent A believes knowing how many teammates can backup α is a factor for B in deciding whether to choose to do α. Agent A can take *backup*$(\alpha, ?num)$ as agent B's information need associated with the decision point, and consider helping B with the relevant information acquired through evaluating *backup*$(\alpha, ?num)$ according to agent A's belief base.

Axiom 8 in Fig. 10 states that in cases where agent A believes that agent B is considering several potential actions in its pursuit of some adopted commitment ϕ, A will assume B will need information $N \in \mathit{reckon}_A(B, \Omega, \phi)$. Here, A is assuming that B needs N because that is what A would need to evaluate the actions toward achieving ϕ. The context

[21] Such cues may include the availability of requisite resources, the number of teammates that can provide back-up behavior upon failure, the possible side effects on joint goals, as well as domain-dependent ones.

C_n of the information need consists of B's chosen intention and its context as well as B's potential intentions.

Axiom 8 is useful for an agent to help a teammate evaluate multiple (typically exclusive) potential intentions to see which one works better in fulfilling the teammate's goals. This to some extent generalizes the approach used in GPGP [31], where agents provide information to a local scheduler which can then construct better schedules.

5.3. Goal-protection information needs

We formally characterize two types of information needs that allow an agent to protect a committed goal from becoming unachievable.

The first type of goal-protection information needs is related to internal threats: lacking such information, an agent may act in an irrational way that would prohibit fulfilling its chosen goal. The information that will be needed to reconcile a potential intention with an already adopted intention is of special interest to this paper.

Axiom 9 in Fig. 11 says that agent A will assume agent B needs to know N if A knows (1) B has a chosen goal ϕ and the potential intention to do action α, and (2) if N holds, B's doing α will make ϕ impossible. Knowing N will allow agent B to maintain the achievability of ϕ by dropping the potential intention to do α. This axiom is quite interesting because it states how an agent can help a teammate in reconciling conflicts between potential intentions and adopted intentions, which is a critical issue in evolving shared plans [43].

To give a concrete example. Suppose Eric is committed to giving an invited talk at a conference from 10 a.m. to 12 a.m. on some day next month (i.e., $\mathsf{Int.Th}(\mathit{Eric}, \mathit{talk\_happened}, t, 10, \mathit{last}(1h))$), and he happens to have a routine lab meeting scheduled from 9 a.m. to 11

Axiom 9 (Goal-Protection Information Needs—Type 1).
$\forall A, B \in TA, \phi, C_\phi, t, t' \geqslant t, t'' > t', t_1 < t', N, \alpha, C_\alpha \cdot$
$[\mathsf{Bel}(A, \mathsf{Int.Th}(B, \phi, t, t'', C_\phi) \wedge$
$\mathsf{Pot.Int.To}(B, \alpha, t, t', C_\alpha), t) \wedge$
$\mathsf{Bel}(A, \mathsf{Do}(B, \alpha, t_1, C_\alpha \wedge N) \Rightarrow \neg\phi), t)] \Rightarrow$
$\mathsf{Bel}(A, \mathsf{InfoNeed}(B, N, t_1, C_n), t)$, *where*
$C_n = C_\phi \wedge C_\alpha \wedge \mathsf{Pot.Int.To}(B, \alpha, t, t', C_\alpha) \wedge \mathsf{Int.Th}(B, \phi, t, t'', C_\phi)$.

Axiom 10 (Goal-Protection Information Needs—Type 2).
$\forall A \in TA, B \in TA, \phi, C_\phi, N, t, t'' > t, \forall G \in TB, \alpha, t_1 < t'' \cdot$
$[\mathsf{Bel}(A, \mathsf{Int.Th}(B, \phi, t, t'', C_\phi), t) \wedge$
$\mathsf{Bel}(A, \exists C' \cdot \mathsf{Pot.Int.To}(G, \alpha, t, t_1, C'), t) \wedge$
$\mathsf{Bel}(A, \exists \Theta_\alpha \cdot \mathsf{Do}(G, \alpha, t_1, \Theta_\alpha) \Rightarrow \neg\phi, t) \wedge$
$\mathsf{Bel}(A, P, t)] \Rightarrow$
$\mathsf{Bel}(A, \mathsf{InfoNeed}(B, N, t_1, C_n), t)$, *where*
$P = \exists \beta, \Theta_\beta, t_b < t_1 \cdot [\mathsf{Bel}(B, N, t_b) \wedge \mathsf{Do}(B, \beta, t_b, \Theta_\beta)] \Rightarrow$
$\nexists R, \Theta_\alpha \cdot \mathsf{CBA}(G, \alpha, R, t_1, \Theta_\alpha)$,
$C_n = C_\phi \wedge \mathsf{Int.Th}(B, \phi, t, t'', C_\phi) \wedge [\exists C' \cdot \mathsf{Pot.Int.To}(G, \alpha, t, t_1, C')]$.

Fig. 11. The axioms for deriving goal-protection information needs.

Axiom 11 (Goal-escape information need).
$\forall A, B \in TA, \phi, C_\phi, t, t' \geqslant t, \forall N \in Needs(C_\phi)\cdot$
$\mathsf{Bel}(A, \mathsf{Int.Tx}(B, \phi, t, t', C_\phi), t) \Rightarrow \mathsf{Bel}(A, \mathsf{InfoNeed}(B, N, t', C_n), t)$, *where*
$C_n = C_\phi \wedge \mathsf{Int.Tx}(B, \phi, t, t', C_\phi)$.

Fig. 12. The axiom for deriving goal-escape information needs.

a.m. on that day (i.e., $\mathsf{Pot.Int.To}(Eric, meeting, t, 9, last(1h))$). His assistant will take the schedule conflict (here N is *conflict*(*talk_happened*, *meeting*)) as Eric's information need. Knowing the conflict, Eric can postpone or cancel the lab meeting.

The second type of goal-protection information needs is related to external threats: lacking such information, an agent may not be able to fulfill its chosen goal because agents in a different team are acting in a way that would thwart its goal. Knowing the threat information, the agent could respond in a timely manner to nullify the plan or intention of the other team.

Axiom 10 in Fig. 11 says that agent A will assume agent B needs N to deal with an external threat, if A knows (1) B has a chosen goal ϕ; (2) an agent G in an opposite team potentially intends to do action α, the doing of which will make ϕ impossible; (3) G would not be able to perform α successfully if B knows N and performs some action β in a timely manner. The context of the information need consists of agent B's intention, the embedded context of B's intention, and agent G's potential intention.

It is worth noting that the anticipating agent A need not know which action agent B will choose to respond to the coming threat. Thus, Axiom 10 leaves open the possibility of searching for recipes/plans to avoid the threat. On the other hand, the axiom offers B the flexibility of choosing one from several possible reactions. Axiom 10 will further elicit the anticipation of action-performing information needs, once it becomes clear to agent A that agent B will adopt a particular action (Int.To) to deal with the threat.

5.4. Goal-escape information needs

It could be the case that if an agent did not know that the context or escape condition had changed status, the agent might take actions that would foil the mission of the whole team. Axiom 11 in Fig. 12 states that if agent A believes that agent B has a goal (Int.Th or Int.To), A will assume that B needs the information described by N, which is generated from the context of B's intention. The context of the information need consists of B's intention and the context of B's intention. Int.Tx in Axiom 11 refers to either Int.Th or Int.To, and ϕ refers to either a proposition or an action, respectively.

From this axiom, it can be proved[22] that teammates can anticipate each other's goal-escape information needs related to their team intentions (e.g., $\mathsf{Int.Th}(T_1, \phi, t, t', C_\phi)$ is an intention of team T_1).

The Joint Intentions theory includes a provision that agents who become aware of certain conditions will adopt certain goals. In accord with the Joint Intentions theory,

22 Assume that joint intentions imply individual intentions [23].

Axiom 11 allows an agent to anticipate others' information needs regarding the context conditions of their goals. Such anticipation may or may not result in communicative actions, depending on whether the agent can possibly do those helpful behaviors.

5.5. Self-reflection on information needs

Being aware of its own information needs, an agent could, instead of passively waiting for others' help, choose to proactively request assistance from teammates or subscribe its information needs from a known information provider. However, as we mentioned before, usually an agent may not be able to know its own information needs by reflection for various reasons. For instance, due to lack of expertise or observability, an agent may have difficulty inferring all the information needs by itself that are relevant to making a certain decision or protecting a certain goal.

But under some contexts, an agent can anticipate its own information needs from the already committed intentions. For instance, when A and B refer to the same agent, Axiom 6.1 states that an agent can derive its own information needs when the agent intends to do some action but lacks the pre-requisite information. Similarly, when A and B refer to the same agent, Axiom 11 states that an agent needs to know all the information relevant to the context of the committed intention.

Knowing its own information needs is not enough; the agent has to know whom to ask. Hence, teammates' anticipation described above and proactive assistance to be studied later play a critical role in cases where the information needer is not aware of its information needs or does not know whom to ask.

5.6. Discussion

Given the complexity of the problem of anticipating others' information needs, it would be cumbersome to have one axiom apply to all the situations. Table 1 summarizes the scope of reasoning covered by the axioms given in this section, where Axiom 9′ and Axiom 10′ refer to axioms similar to Axiom 9 and Axiom 10, respectively, for the combination of Pot.Int.Th and Int.Th.

The types of information needs characterized by the axioms are by no means complete; the vacant fields in Table 1 reveal the potential directions to be explored in the future. On the other hand, the given axioms are not redundant either. For instance, to anticipate B's

Table 1
Anticipating information needs from intentions

Intentions	Intentional context	Action preconditions	Recipe knowledge
Int.To	Axiom 11	Axiom 6.1	
Int.Th	Axiom 11	Lemma 1	
Pot.Int.To		Axiom 6.2	
Pot.Int.Th			
Pot.Int.To+ Int.Th			Axioms 8, 9, 10
Pot.Int.Th+ Int.Th			Axioms 9′, 10′

information needs, both Axiom 8 and Axiom 9 require that agent A believe that agent B's adopted intention and potential intentions. The difference is that the adopted intention and potential intentions in Axiom 8 are highly related (consistent) while the adopted intention and the potential intention in Axiom 9 are typically competitive. Examples can be given to show that the ability to anticipate others' information needs would be weakened if any of the axioms were removed from the framework.

6. Commitment to other's information needs

When an agent recognizes the information needs of its teammates by being informed or by anticipating, it will consider providing help if that would not foil the fulfillment of the adopted intentions or reduce the performance of the whole team. An important issue here is how to relate an agent's belief about the information needs of teammates to intentions to help. One may be tempted to establish this linkage using an axiom similar to Axiom 1: If (1) agent A believes that agent B has an information need, (2) A believes that B does not have the information, and (3) the performance of some action β can lead to B's awareness of the information, then A will consider doing β. However, this seemingly intuitive approach has two drawbacks: (1) it requires the action β be explicitly prescribed, and (2) it does not explicitly specify that agent A should be persistent in its helpful commitment to the information needs.

One more general approach is to make abstract rather than specific the commitments for satisfying others' information needs, postponing the specific commitments (and their reconciliation) to later stages. In this way, the commitment to providing help can be clearly separated from the decisions on how to provide help. We conjecture that this would improve flexibility in implementing agent teams with multiple proactive behaviors.

Let B_A be the belief base of agent A; then $\mathsf{B}_A \models p$ represents that p is a logical consequence of B_A. For any agent A and need-expression N, function $info(A, N)$ returns the information with respect to N evaluated by A:

$$info(A,N) \triangleq \begin{cases} N & \text{if } \mathsf{B}_A \models N, \text{ and } N \text{ is a proposition,} \\ \neg N & \text{if } \mathsf{B}_A \models \neg N, \text{ and } N \text{ is a proposition,} \\ Refer(N,Q) & \text{if } N = (iota\ ?\vec{x}\ p(?\vec{x})), \\ & Q \in \Sigma = \{\theta \cdot ?\vec{x} : \mathsf{B}_A \models \theta \cdot p, \theta \text{ is most general} \\ & \ \text{substitution (mgs)}\}, \text{ and } \Sigma \text{ is singleton,} \\ Refer(N,Q) & \text{if } N = (any\ ?\vec{x}\ p(?\vec{x})), \\ & Q \in \Sigma = \{\theta \cdot ?\vec{x} : \mathsf{B}_A \models \theta \cdot p, \theta \text{ is mgs}\} \neq \emptyset, \\ Refer(N,\Sigma) & \text{if } N = (all\ ?\vec{x}\ p(?\vec{x})), \\ & \Sigma = \{\theta \cdot ?\vec{x} : \mathsf{B}_A \models \theta \cdot p, \theta \text{ is mgs}\}. \end{cases}$$

$info(A, N)$ is undefined in the following cases: (1) N is a proposition, but neither $\mathsf{B}_A \models N$ nor $\mathsf{B}_A \models \neg N$ holds. In this case, the information related to N is unknown to agent A. (2)$N = (iota\ ?\vec{x}\ p(?\vec{x}))$ but Σ is not a singleton. In this case, a unique solution is required but agent A finds more than one solution for $p(?\vec{x})$. (3)$N = (any\ ?\vec{x}\ p(?\vec{x}))$ but $\Sigma = \emptyset$. In this case, agent A finds no solution for $p(?\vec{x})$. In cases where $N = (any\ ?\vec{x}\ p(?\vec{x}))$ and $|\Sigma| > 1$, a randomly selected element of Σ is returned.

Axiom 12 (ProAssist). $\forall A, B \in TA, N, C_n, t, t' > t \cdot$
$\mathsf{Bel}(A, \mathsf{InfoNeed}(B, N, t', C_n), t) \Rightarrow$
$[has.info(A, N, t) \Rightarrow \mathsf{Pot.Int.Th}(A, \mathsf{Bel}(B, info_t(A, N), t'), t, t', C_n) \vee$
$\neg has.info(A, N, t) \Rightarrow \mathsf{Pot.Int.Th}(A, has.info(B, N, t'), t, t', C_n)]$.

Fig. 13. The axiom for proactive assist.

Usually function *info* is evaluated at a certain time point. We thus use $info_t(A, N)$ to denote the information returned when agent A evaluates N at time t. Predicate $has.info(A, N, t)$ is true when $info_t(A, N)$ is defined, and false otherwise.

Axiom 12 in Fig. 13 says that when an agent comes to know another agent's information needs, the first agent will adopt an attitude of potential intention-that toward "the other's belief about the needed information". It is worth noting that even if A is unaware of the information needed by B, A can still adopt an intention which might lead it to engage other agents in providing help (e.g., by forwarding the information need to another agent).

In Axiom 12 we use Pot.Int.Th rather than Pot.Int.To because Pot.Int.To requires the agent to adopt a specific action to help the needer while Pot.Int.Th offers the agent flexibility in choosing how to help. Note that A and B could refer to the same agent. In this case, from the semantics of InfoNeed, the truth value of $has.info(A, N, t)$ must be false. Then, the axiom would allow an agent to adopt a potential intention, which would further stimulates the agent to consider means-ends reasoning to help itself. Axiom 12 relates information needs with potential intentions-that. It, together with Axiom 3, specifies how an agent chooses appropriate actions to satisfy its own or others' information needs.

Being aware of others' information needs does not always lead to helping actions. Many factors (e.g., an agent is simply too busy) may prevent an agent from adopting the commitment. This is the reason why instead of Int.Th we choose Pot.Int.Th, which offers agents the flexibility of deciding whether to help. Once the Pot.Int.Th is upgraded to Int.Th, the agent is committed to retrying until either the information needer is satisfied or the information need is no longer relevant.

Furthermore, if an agent has an intention-that concerning some other agent's information need, then Axiom 3 implies that agent may eventually adopt Pot.Int.To's to fulfill that information need. This enables our framework to specify the situations in which an agent could reflect on its helping behaviors, yet leaves open the agent's commitment to such behaviors. When an agent faces multiple opportunities to assist, it will not be restricted to committing to a specific helping action.

7. Proactive inform

Up to now, we have discussed how agents anticipate others' information needs and how agents choose to help others with their information needs by adopting appropriate intentions and potential intentions. By Axiom 3, we also know that agents will eventually perform certain actions to fulfill their commitments to helping others. In this and the next section, we will introduce two kinds of communicative actions that can be used to fulfill an agent's commitments regarding others' information needs.

The communicative actions to be introduced are different from those in the existing literature (e.g., Inform [21,22]) in at least two ways. First, the new identified communicative actions are *need-driven* performatives in the sense that the speaker is aware of the addressee's information needs prior to performing these actions. Such need-driven semantics has never been explicitly captured before. For instance, in FIPA [1], even though "ask" intuitively conveys the speaker's need to the addressee, such a need is not captured in the semantics of ask (defined in terms of *query-if*). Similarly, the addressee's reply to an "ask" should be derived from its awareness of the asking agent's needs, rather than being simply treated as a reactive act (i.e., modeled as an "inform" regarding the result of a query to its belief base). Of course, the replying agent knows implicitly that the asking agent *needs* to know the thing it asked, but such an implicit reflection on another's needs is still weak: the replying agent does not know the purpose or context of the asking agent's needs. Generally, as far as communicative acts are concerned, deliberative semantics would be preferable to reactive semantics for at least two reasons. First, the well-adopted idea of performative-as-attempt promotes mentalistic characterization of communicative acts; this can be better leveraged by providing deliberative semantics for communicative acts. Second, information needs are more stable, and thus more valuable than information itself. Explicitly capturing information needs makes it possible for agents to commit persistently to satisfying these information needs: an agent can proactively deliver information whenever the information changes.

Second, the newly identified communicative actions allow the flow of information needs as well as the exchange of information. This becomes possible in our framework due to the introduction of the concept of information needs. The flow of information needs offers three benefits:

(1) It can be used by the addressee agent to account for the communication behaviors of the speaking agent and by the speaking agent to establish certain expectations of the possible response from the addressee agent. A speaking agent may also want to initiate conversations to confirm that the anticipated information needs do reflect the real needs of the addressee;
(2) It enables agents in a team to better establish and evolve an "approximate mental model" regarding others' information needs. Such a mental model is important for further enhancing a team of agents in their intelligent information exchange; and
(3) The contexts of information needs allow agents to make certain inferences about the activities being pursued by the information needers. This is useful for recognizing teammates' plans (recipes), explaining teammates' intention-shifting, and for better anticipating teammates' information needs.

7.1. ProInform

We will use ProInform (i.e., Proactive Inform) to refer to the new communicative act to be defined. One may be tempted to define it using compositionality of speech acts like:

$$[\mathsf{Bel}(A, \mathsf{InfoNeed}(B, N, t', C), t) \wedge (I = \mathit{info}(A, N))]; \mathsf{Inform}(A, B, \epsilon, I, \ldots).$$

Definition 10. $\mathsf{ProInform}(A, B, \epsilon, I, N, t, t_a, t', C_n) \triangleq$
$[(t_a < t') \wedge (I = \mathit{info}(A, N))]?; \mathsf{Attempt}(A, \epsilon, P, Q, C_n, t, t_a)$, where
$P = \mathsf{Bel}(B, I, t')$,
$Q = \exists t'' \cdot (t \leqslant t'' < t_a) \wedge \mathsf{MB}(\{A, B\}, \psi, t'')$, where
$\psi = \exists C_p, \exists t_b \cdot (t'' \leqslant t_b < t_a) \wedge \gamma \wedge \mathsf{Int.Th}(A, \mathsf{Bel}(B, \phi, t_b), t, t_b, C_p)$, where
$\phi = \mathsf{Bel}(A, I, t) \wedge \mathsf{Bel}(A, \mathsf{InfoNeed}(B, N, t', C_n), t)$,
$\gamma = [C_p \supseteq \{C_n, \mathsf{Bel}(A, \mathsf{InfoNeed}(B, N, t', C_n), t),$ (c1)
$\mathsf{Bel}(A, I, t), (I = \mathit{info}(A, N)),$ (c2)
$\mathit{pos}(N) \Rightarrow [\mathsf{Bel}(A, \mathsf{UBif}(B, I, t), t) \vee \mathsf{CBel}(A, B, I, t)]\}]$. (c3)

Fig. 14. The definition of proactive-inform.

However, this definition does not explicitly convey the receiver's information need as an account of the speaker's communicative act, even though the definition requires the speaker to believe the communicated information is related to the receiver's information needs. As we mentioned above, agents having mutual awareness of their information needs can improve the effectiveness of proactive communication.

For this reason, we define ProInform in Fig. 14[23] by extending the semantics of Inform with additional requirements on the speaker's awareness of and willingness to convey the anticipated information needs of the addressee. Thus, the speaker's belief about the addressee's need for the information is explicitly included as a part of the mental states being communicated.

The ultimate goal of ProInform is to let B at time t' believe the information I related to the need-expression N. According to the definition of InfoNeed, A knows that B will need the information related to N by time t' (e.g., t' may be the last opportunity for B to perform some action in order to achieve some goal). This means that A, to be helpful, has to deliver the relevant information no later than t'; otherwise such a ProInform makes no sense.[24] Normally, B can get the information delivered by A before t'. In such cases, B will be lucky to find out at t' that the information B needs next is already available. Thus, to let B believe I at t' is actually A's lowest expectation in using ProInform to help B. After that, B may continue to believe the information, or drop it if it conflicts with newly acquired information.

The honest effort of ProInform is to establish a mutual belief about the speaker's goal to let the addressee know that (1) the speaker knows the information being communicated, and (2) the speaker knows the addressee needs the information. The mutual belief is established at some time t'' before t_a; the uncertainty of t'' is due to the uncertainty of the factors like the delay of communication, the reliability of communication, etc. For the same reason, the Int.Th in the mutual belief to be established also refers to an uncertain time point t_b. Time t_b is somewhere between t'' and t_a because A is intending to change B's beliefs after the establishment of the mutual belief rather than before. We could replace t_b with t, and then the content of the mutual belief would be an intention at t. Consequently, B

[23] Refer to Section 3.1 for the definitions of CBel and UBif, and refer to Section 4.2 for the definition of *pos*.
[24] Notice that in the definition $t_a < t'$. This ensures that ProInform is only performed to satisfy others' information needs in the future.

would have to adopt new beliefs at or after t'' based on what A wanted B to adopt at a past time t. The approach used in the definition is more straightforward.

The definition of ProInform involves two contexts: C_n and C_p. C_n refers to the context of the information need under concern; it is also used as the context of the Attempt. C_p is the 'actual' context of ProInform. The context of a proactive communicative act plays an important role in specifying the semantics of the act, and in allowing agents involved in a conversation session to interpret each other's communication behavior. Before talking about C_p, we first examine in general what the context of a proactive performative may be composed of.

First of all, proactive communicative acts are performed only when it is necessary. Their contexts should capture appropriate "escape" conditions; the attempting agent (the initiator or the addressee) could discharge its duty of achieving the communicative goal whenever it realizes that the escape conditions no longer hold. Such excuses are typically related to the information need under consideration because proactive communicative acts are *per se* driven by information needs. This may include the context of the information need and the speaker's belief of the information need. Both are necessary because the truth value of either one cannot be derived from the other. The expiration time of the information need also implicitly establishes an escape condition for proactive performatives.

Second, like domain actions, communicative acts cannot be performed if the associated constraints are not satisfied. The constraints of a proactive communicative act may include: (1) the beneficiary agent does not have the information to be delivered or has the wrong information; (2) the sending agent either has the information or knows how to acquire the information (e.g., by requesting from a known provider). Also, an agent may consider certain personalized constraints such as the threshold on the possibility that the needer can get the information from other teammates, and the tradeoff between the benefits of communication and the potential side-effects (e.g., being overheard by opponent agents, slowing down its individual activities, etc.).

Optionally, the context of a proactive performative may specify the expected communication delay, the frequency of retry, and the conversation policy, which circumscribes the potential responses from the receiver as well as the speaker's reactions to a reply. The context may also specify common assumptions related to social relationships such as agent sincerity in communication, agent activeness (extraversion, agreeableness, etc. [33]), agent cooperativeness.

Table 2 summarizes the compositions of contexts for proactive performatives. Both *constraints* and *escape conditions* are essential parts. The optional part can serve as en-

Table 2
Contexts of proactive performatives: the composition

Escape conditions	Constraints (example)	Optional (example)
• the expiration time of the InfoNeed hasn't come • the context of the InfoNeed holds • the speaker believes the InfoNeed	• the beneficiary holds no or wrong beliefs about the info to be delivered • the speaker holds or knows how to get the information	• conversation policy • expected delay • sincerity • activeness

hancements for the semantics; it could even be extended to help interpret the meanings of indirect proactive speech acts. In this paper, the optional part is left open when a performative context is specified.

Now we come to the performative context of ProInform. C_p includes the context of the information need (C_n) and A's belief of the information need, which together serve as escape conditions in C_p. This means, agent A will abandon the intention Int.Th (embedded in the mutual belief to be established) whenever A believes the context C_n no longer holds or A no longer believes B will need I. In addition, since the Int.Th will be established as a mutual belief, C_n, as a part of C_p, may also be exploited by B to justify the information need anticipated by A: B may inform A that B never held the intentions from which A managed to derive the information need. Thus, A can discharge its help by dropping the incorrect information need. This shows how the context of information needs plays a role in specifying the semantics of communicative acts driven by information needs.

Lines *c2* and *c3* in C_p are constraints. It does not make sense for agent A to ProInform information I to agent B if A currently does not believe I. When the need-expression N refers to a proposition, agent A ProInforms information I only when A believes B does not have I or B holds a wrong belief of I.

The above described components of C_p only establish the minimum requirements on C_p. The complete composition of C_p is implementation-dependent. The performative context C_p justifies the behavior of an agent who uses ProInform. For instance, suppose ProInform is implemented in a multi-agent system using a component that reasons about the information needs of teammates and a communication plan involving sending, receiving confirmation, and re-sending if confirmation is not received. An agent can choose to abandon such a communication plan during execution if the agent realizes the context of the addressee's information need is no longer true.

7.2. *A conversation protocol for* ProInform

Intentional semantics of performatives is desirable because humans' choice of commitments to communicative acts really involves reasoning about the beliefs, intentions, and abilities of other agents. However, reliable logical reasoning about others' private beliefs and goals is technically difficult. Practical agent systems typically employ various assumptions to simplify this issue. One promising approach is to frame the semantics of performatives using publicly shared protocols or conversation policies. Conversation policies, serving as constraints on the potentially unbounded universe of semantically coherent message sequences [41], make it easier for the agents involved in a conversation to model and reason about each other. In particular, conversation policies can restrict agents' attention to a smaller set of possible responses which otherwise could be larger.

To design protocols for ProInform, we start with the potential responses of the addressee to a ProInform. An "acceptance" response is what the initiator of ProInform most wants to bring about because this is exactly the best reward for its helping behavior. However, an addressee may disregard or even explicitly reject a ProInform for many reasons. For instance, an addressee may prefer to keep what it already has if that conflicts with the information received from the speaker of ProInform. The semantics of ProInform also has

Definition 11 (*Responses to* ProInform).
$\mathsf{WAcceptInfo}(B, A, \epsilon, I, N, t, t_a, t', C_n) \triangleq \mathsf{Inform}(B, A, \epsilon, \psi, t, t_a)$, where
$\psi = \mathsf{Bel}(B, I, t) \wedge \mathsf{Bel}(B, \neg\mathsf{InfoNeed}(B, N, t', C_n), t)$;
$\mathsf{SAcceptInfo}(B, A, \epsilon, I, N, t, t_a, t', C_n) \triangleq \mathsf{Inform}(B, A, \epsilon, \psi, t, t_a)$, where
$\psi = \mathsf{Bel}(B, I, t) \wedge \mathsf{Bel}(B, \mathsf{InfoNeed}(B, N, t', C_n), t) \wedge$
$\quad \mathsf{Int.Th}(B, \mathsf{Bel}(B, I, t'), t, t', C_n)$;
$\mathsf{WRejectInfo}(B, A, \epsilon, I, N, t, t_a, t', C_n) \triangleq \mathsf{Inform}(B, A, \epsilon, \psi, t, t_a)$, where
$\psi = \neg\mathsf{Bel}(B, I, t) \wedge \mathsf{Bel}(B, \mathsf{InfoNeed}(B, N, t', C_n), t)$;
$\mathsf{SRejectInfo}(B, A, \epsilon, I, N, t, t_a, t', C_n) \triangleq \mathsf{Inform}(B, A, \epsilon, \psi, t, t_a)$, where
$\psi = \neg\mathsf{Bel}(B, I, t) \wedge \mathsf{Bel}(B, \neg\mathsf{InfoNeed}(B, N, t', C_n), t)$.

Fig. 15. The responses to proactive-inform.

direct impacts on the receiver. For instance, a ProInform may be rejected simply because the receiver does not agree on the information need anticipated by the speaker.

As shown in Fig. 15, we define two kinds of acceptance in response to ProInform: WAcceptInfo (accept the information but refuse the information need) and SAcceptInfo (accept the information and the information need), and define two kinds of refusal in response to ProInform: WRejectInfo (reject the information but accept the information need) and SRejectInfo (reject the information and the information need).

By WAcceptInfo, an accepting agent B attempts to inform the listening agent A that it accepts the information I from A but rejects the information need anticipated by A. In this case, B may think information I will be useful in its other activities.

SAcceptInfo carries stronger semantics: the accepting agent B attempts to let the listening agent A know that it really adopts the information and will commit to maintaining the information up to the time the need expires. Note that even though B believes I at the time of doing SAcceptInfo, I may change between t and t'. In the case that A or some other agent observes such a change, the agent may perform another ProInform to B. If this happens, agent B needs to drop the obsolete Int.Th before performing another SAcceptInfo.

By WRejectInfo, agent B attempts to inform agent A, the speaker of ProInform that B believes it has the information need anticipated by A, but does not believe I. In such a case, I may conflict with B's existing beliefs related to N, and B chooses to persist in what it believes.

SRejectInfo carries stronger semantics: the rejecting agent B attempts to inform the listening agent A that it believes neither the information I nor the information need anticipated by A. This may enable A to revise its model about B's information needs, as well as discharge A from further helping B regarding N.

Fig. 16 shows a conversation protocol involving ProInform using a Petri-Net representation [38]. One of the criteria in designing this protocol is that it should be able to enrich team intelligence in proactive information delivery by considering not only the exchange of information but also the flow of information needs.

The states in Fig. 16 are labeled $s0$ to $s6$ and each state transition is labeled by a communicative act. The sink states $s4$, $s5$, and $s6$ are possible final states, where $s5$ is the main final state representing the ideal execution of the protocol. The context of ProInform can be recorded in the *start* state $s0$ and the goal (i.e., let B know the information related to N)

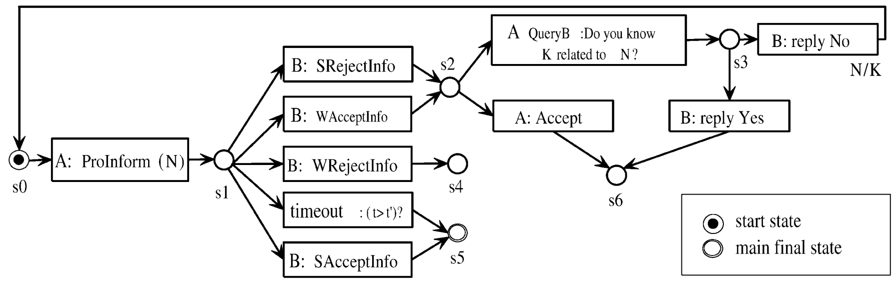

Fig. 16. A conversation protocol involving ProInform regarding need N.

can be recorded in the main *final* state, $s5$. In the beginning, an initiator A proactively informs agent B a piece of information related to need-expression N. Agent B may respond with one of five possible choices:

(1) B strongly accepts A's ProInform by performing SAcceptInfo;
(2) B strongly accepts A's ProInform simply by keeping silent if timeout is commonly assumed by both parties as strong acceptance;
(3) B performs WRejectInfo in response to A's ProInform, and the protocol terminates at state $s4$. In this case, the protocol may be extended such that A will persuade B to update its information, or send B newly acquired information related to N;
(4) B performs SRejectInfo in response to A's ProInform;
(5) B performs WAcceptInfo in response to A's ProInform.

In the first two cases, the protocol terminates and A can discharge its helpful commitment to B regarding the information related to N. In the last two complicated cases, agent A will keep trying to help B recognize its information need related to N. For instance, assuming that B could not recognize N as its information need due to a lack of inference knowledge, and knowing that K is closer than N to B's purpose (e.g., performing some action), A will take K as B's new information need and perform another ProInform with respect to information need K. Such a recursive process may terminate when A chooses to accept B's refusal, or B clarifies to A that its refusal is not due to a lack of certain inference knowledge (e.g., the information need N anticipated by A is simply wrong). In these two cases, the protocol terminates at state $s6$, where the initiator of ProInform might revise its belief about B's information needs.

It is easy to show that the protocol is complete in the sense that no undischarged commitments are left behind.

7.3. *Some properties of* ProInform

Theorem 1. *Successful performance of the* ProInform *act establishes a mutual belief between the sender and the addressee that the sender believes the delivered information and the sender believes that the addressee needs the delivered information. Formally,*

$$\models \mathsf{SuccDone}(A, \mathsf{ProInform}(A, B, \epsilon, I, N, t, t_a, t', C_n)) \Rightarrow \mathsf{MB}(\{A, B\}, \mathsf{Bel}(A, I, t) \wedge \mathsf{Bel}(A, \mathsf{InfoNeed}(B, N, t', C_n), t), t_a).$$

Proof. (1) Assume $\mathsf{SuccDone}(A, \mathsf{ProInform}(A, B, \epsilon, I, N, t, t_a, t', C_n))$.

(2) By (1), the definition of SuccDone (Definition 7 in Section 3.5) and the definition of ProInform (cf. Fig. 14), there exists a time $t_1 < t_a$ such that $\mathsf{MB}(\{A, B\}, \psi, t_1)$, where

$$\psi = \exists C_p, \exists t_b \cdot (t_1 \leqslant t_b < t_a) \wedge \gamma \wedge \mathsf{Int.Th}(A, \mathsf{Bel}(B, \phi, t_b), t, t_b, C_p),$$

where

$$\begin{aligned}
\gamma = [C_p \supseteq \{&C_n, \mathsf{Bel}(A, \mathsf{InfoNeed}(B, N, t', C_n), t),\\
&\mathsf{Bel}(A, I, t), (I = \mathit{info}(A, N)),\\
&\quad \mathit{pos}(N) \Rightarrow [\mathsf{Bel}(A, \mathsf{UBif}(B, I, t), t) \vee \mathsf{CBel}(A, B, I, t)]\}],\\
\phi &= \mathsf{Bel}(A, I, t) \wedge \mathsf{Bel}(A, \mathsf{InfoNeed}(B, N, t', C_n), t)\\
&\equiv \mathsf{Bel}(A, I \wedge \mathsf{InfoNeed}(B, N, t', C_n), t)\\
&\equiv \mathsf{Bel}(A, \delta, t), \quad \text{where}\\
\delta &= I \wedge \mathsf{InfoNeed}(B, N, t', C_n).
\end{aligned}$$

(3) A is assumed to be sincere, thus by Axiom 2 (cf. Fig. 3) we have

$$\begin{aligned}
&\mathsf{Int.Th}(A, \mathsf{Bel}(B, \mathsf{Bel}(A, \delta, t), t_b), t, t_b, C_p) \Rightarrow\\
&\qquad [\mathsf{Bel}(A, \delta, t) \wedge \mathsf{Int.Th}(A, \mathsf{Hold}(\delta, t_b), t, t_b, C_p)].
\end{aligned}$$

(4) From (2) and (3) we have $\mathsf{MB}(\{A, B\}, \mathsf{Bel}(A, \delta, t), t_1)$.

(5) By Assumption 2 (cf. Section 3.1), we can conclude that

$$\mathsf{MB}(\{A, B\}, \mathsf{Bel}(A, \delta, t), t_a). \qquad \square$$

Note that in the proof of Theorem 1, it does not matter whether agents A and B can agree on a context C_p and a specific time point for t_b. Moreover, for each A and B, the value of t_b may be different in different possible worlds.

Theorem 2. *Successful performance of a* ProInform *with respect to I and N followed by a successful* SAcceptInfo *by the addressee of* ProInform *establishes a mutual belief between the two agents that the information I is true and the addressee of* ProInform *really needs N. Formally* ($t_0 < t_1 < t_2 < t_3 < t'$),

$$\begin{aligned}
\models\ &\mathsf{SuccDone}(A, \mathsf{ProInform}(A, B, \epsilon_1, I, N, t_0, t_1, t', C_n)) \wedge\\
&\mathsf{SuccDone}(B, \mathsf{SAcceptInfo}(B, A, \epsilon_2, I, N, t_2, t_3, t', C_n))\\
&\Rightarrow \mathsf{MB}(\{A, B\}, I \wedge \mathsf{InfoNeed}(B, N, t', C_n), t_3).
\end{aligned}$$

Proof. (1) Assume that

$$\mathsf{SuccDone}(A, \mathsf{ProInform}(A, B, \epsilon_1, I, N, t_0, t_1, t', C_n))$$

and

$$\mathsf{SuccDone}(B, \mathsf{SAcceptInfo}(B, A, \epsilon_2, I, N, t_2, t_3, t', C_n))$$

hold.

(2) By applying Theorem 1, we have

$$\mathsf{MB}(\{A, B\}, \mathsf{Bel}(A, I, t_0) \wedge \mathsf{Bel}(A, \mathsf{InfoNeed}(B, N, t', C_n), t_0), t_1).$$

(3) By Proposition 1 and the definition of SAcceptInfo, we have

$$\mathsf{MB}(\{B, A\}, \mathsf{Bel}(B, I, t_2) \wedge \mathsf{Bel}(B, \mathsf{InfoNeed}(B, N, t', C_n), t_2), t_3).$$

(4) By Assumption 3 (cf. Section 3.5), A's commitments in ProInform and B's commitments in SAcceptInfo prevent them from changing beliefs about I and the information need before t_3. Thus, we can conclude that

$$\mathsf{MB}(\{A, B\}, I \wedge \mathsf{InfoNeed}(B, N, t', C_n), t_3). \qquad \Box$$

From Theorem 2 we can draw a conclusion that the protocol shown in Fig. 16 is correct in the sense that successful execution of ProInform and SAcceptInfo can achieve the goal of the protocol.

Similarly we can prove the following results.

Theorem 3. (1) *Successful performance of a* ProInform *with respect to I and N followed by a successful* WAcceptInfo *by the addressee of* ProInform *establishes a mutual belief between the two agents that the information I is true*;

(2) *Successful performance of a* ProInform *with respect to I and N followed by a successful* WRejectInfo *by the addressee of* ProInform *establishes a mutual belief between the two agents that the addressee of* ProInform *really needs N*;

(3) *Successful performance of a* ProInform *with respect to I and N followed by a successful* SRejectInfo *by the addressee of* ProInform *can only establish a mutual belief of the addressee* (*of* ProInform)*'s belief regarding the information I and the need N. Formally* $(t_0 < t_1 < t_2 < t_3 < t')$,

$$\begin{aligned}
(1)\ \ &\models \mathsf{SuccDone}(A, \mathsf{ProInform}(A, B, \epsilon_1, I, N, t_0, t_1, t', C_n)) \wedge \\
&\quad \mathsf{SuccDone}(B, \mathsf{WAcceptInfo}(B, A, \epsilon_2, I, N, t_2, t_3, t', C_n)) \\
&\quad \Rightarrow \mathsf{MB}(\{A, B\}, [I \wedge \mathsf{Bel}(B, \neg\mathsf{InfoNeed}(B, N, t', C_n), t_2)], t_3), \\
(2)\ \ &\models \mathsf{SuccDone}(A, \mathsf{ProInform}(A, B, \epsilon_1, I, N, t_0, t_1, t', C_n)) \wedge \\
&\quad \mathsf{SuccDone}(B, \mathsf{WRejectInfo}(B, A, \epsilon_2, I, N, t_2, t_3, t', C_n)) \\
&\quad \Rightarrow \mathsf{MB}(\{A, B\}, [\neg\mathsf{Bel}(B, I, t_2) \wedge \mathsf{InfoNeed}(B, N, t', C_n)], t_3), \\
(3)\ \ &\models \mathsf{SuccDone}(A, \mathsf{ProInform}(A, B, \epsilon_1, I, N, t_0, t_1, t', C_n)) \wedge \\
&\quad \mathsf{SuccDone}(B, \mathsf{SRejectInfo}(B, A, \epsilon_2, I, N, t_2, t_3, t', C_n)) \\
&\quad \Rightarrow \mathsf{MB}(\{A, B\}, [\neg\mathsf{Bel}(B, I, t_2) \wedge \mathsf{Bel}(B, \neg\mathsf{InfoNeed}(B, N, t', C_n), t_2)], t_3).
\end{aligned}$$

Theorem 4. *A* ProInform *can be performed even when the receiver does not realize it needs the information. Formally,*

$$\begin{aligned}
&\mathsf{SuccDone}(A, \mathsf{ProInform}(A, B, \epsilon, I, N, t, t_a, t', C_n)) \wedge \\
&\quad \neg\mathsf{Bel}(B, \mathsf{InfoNeed}(B, N, t', C_n), t)
\end{aligned}$$

is satisfiable.

Theorem 4 can be proved by constructing two possible-world structures (one for time point t and one for t'' ($t'' < t_a$) when the honest effort of ProInform is established) and by showing the possibility of the transition from the first structure to the second.

We define a meta-predicate $CUPP(A, \phi)$ to represent that agent A can upgrade the potential intentions regarding ϕ to intentions. For example, suppose $\phi = \mathsf{Bel}(D, I, t')$, then $CUPP(A, \phi)$ represents that agent A can upgrade potential intentions such as $\mathsf{Pot.Int.Th}(A, \mathsf{Bel}(D, I, t'), t, t', C_n)$ to an intention $\mathsf{Int.Th}(A, \mathsf{Bel}(D, I, t'), t, t', C_n)$. $CUPP$ is used to abstract away the details about how agents reconcile conflicts.

Theorem 5. *If agent A believes information I related to B's need N, it will consider helping B with I using* ProInform. *Formally,*

$$\models [\mathsf{Bel}(A, \mathsf{InfoNeed}(B, N, t', C_n), t) \wedge \mathsf{Bel}(A, I, t) \wedge (I = info(A, N)) \wedge$$
$$\neg\mathsf{Bel}(A, \mathsf{Bel}(B, I, t'), t) \wedge CUPP(A, \mathsf{Bel}(B, I, t'))] \Rightarrow$$
$$\exists t_1, t_2, C' \cdot \mathsf{Pot.Int.To}(A, \mathsf{ProInform}(A, B, \epsilon, I, N, t_1, t_2, t', C_n), t, t_1, C').$$

Proof. (1) Assume

$$\neg\mathsf{Bel}(A, \mathsf{Bel}(B, I, t'), t), \quad \mathsf{Bel}(A, I, t), \ (I = info(A, N)),$$
$$\mathsf{Bel}(A, \mathsf{InfoNeed}(B, N, t', C_n), t), \quad \text{and} \quad CUPP(A, \mathsf{Bel}(B, I, t')).$$

(2) $has.info(A, N, t)$ follows from the assumption $(I = info(A, N))$.

(3) By Axiom 12 (cf. Fig. 13) and $\mathsf{Bel}(A, \mathsf{InfoNeed}(B, N, t', C_n), t)$, we have

$$\mathsf{Pot.Int.Th}(A, \mathsf{Bel}(B, I, t'), t, t', C_n).$$

(4) By the assumption that $CUPP(A, \mathsf{Bel}(B, I, t'))$ holds, the potential intention $\mathsf{Pot.Int.Th}(A, \mathsf{Bel}(B, I, t'), t, t', C_n)$ can be upgraded to $\mathsf{Int.Th}(A, \mathsf{Bel}(B, I, t'), t, t', C_n)$.

(5) Agents are assumed to have the capabilities of doing communicative actions. Also, from Theorem 2 we know that A's ProInform followed by B's SAcceptInfo can make $\mathsf{Bel}(B, I, t')$ true. Thus there exist t_1 and t_2 such that

$$\mathsf{Lead}(A, \mathsf{ProInform}(A, B, \epsilon, I, N, t_1, t_2, t', C_n), \mathsf{Bel}(B, I, t'), t, t_1, \Theta).$$ [25]

(6) From the assumption $\neg\mathsf{Bel}(A, \mathsf{Bel}(B, I, t'), t)$, (4), (5), and Axiom 3, we can derive $\mathsf{Pot.Int.To}(A, \mathsf{ProInform}(A, B, \epsilon, I, N, t_1, t_2, t', C_n), t, t_1, \Theta \wedge C_n)$. This completes the proof. □

Note that in Theorem 5, the context C' of A's potential intention is actually composed of the context C_n of B's information need and the constraints of performing ProInform.

8. Proactively subscribe information needs

In Section 5, we examined how an agent may anticipate other teammates' information needs based on their shared mental models. While an agent may be able to anticipate

[25] Refer to Fig. 5 for the definition of Lead.

certain information needs of teammates, this is not always reliable in dynamic domains, and sometimes the whole team might have to pay the price for a delay in information sharing. A complementary means is to allow agents to reasonably share their information needs with their teammates.

Information needs is a kind of meta-level information. After proclaiming its information needs to another teammate, an agent typically wants to receive either a firm commitment or refusal from that addressee. For instance, suppose agent A needs weather forecast information for a particular area in a battle space for a certain time period, and agent B is one of the weather information providers known to A. To let B know its information need, A actually intends that B commit to delivering the relevant information during the time period. In other words, A is expecting a confirmation from B regarding whether B can satisfy the information need. Such a confirmation is critical because if refused, agent A could proactively gather the needed information using alternative means (e.g., by requesting another weather information provider). Without B's confirmation, agent A will be left with a hard decision on whether to request help from another teammate because requesting multiple teammates may result in redundant information delivery.

Hence, the essence of "informing an information need" is not just information sharing, but more "expecting the addressee to adopt a commitment to satisfying the information need under concern". We take this as a criterion to see whether compositionality of speech acts suffice to capture such semantics.

The first attempt is to treat "informing an information need" as a special case of informing information. It can thus be defined in terms of Inform as:

$$\mathsf{Inform}(A, B, \epsilon, \mathsf{InfoNeed}(A, N, t', C_n), t, t_a). \tag{8.1}$$

Defined as such, agent A merely informs agent B of its information need N. Even though B does accept such an Inform (i.e., $\mathsf{Bel}(B, \mathsf{InfoNeed}(A, N, t', C_n), t_a)$ holds), according to Axiom 12, B will only consider sending–rather than adopting a commitment to sending—A the information described by N unless irrelevant.

One might also be tempted to model it using Request, since the speaker is expecting the addressee to perform a certain communicative action. In doing so, Request and Inform may be composed as:

$$\mathsf{Request}(A, B, \epsilon, \mathsf{Inform}(B, A, \epsilon', \mathit{info}(B, N), t_1, t_2), t, t_a, \Theta). \tag{8.2}$$

However, defined as such, B is required to send information based on its beliefs at exactly the time it performs Inform (the evaluation of $\mathit{info}(B, N)$ works like a query to a database server). Composition (8.2) even does not allow B to know A's information need. B thus is under no obligation to send A the relevant information when it becomes available.

One may also want to compose Request together with ProInform as:

$$\mathsf{Request}(A, B, \epsilon, \mathsf{ProInform}(B, A, \epsilon', I, N, t_1, t_2, t', C_n), t, t_a, \Theta). \tag{8.3}$$

Like (8.2), by accepting this Request, agent B only makes a one-time response to A's information need rather than a long-term commitment until t'. Moreover, (8.3) requires that agent B already know A's information need N (the context of ProInform). In such cases, we know from Axiom 12 that agent B, whenever possible, will consider helping A without being requested. Then, the Request in (8.3) is actually of no use.

To make the semantics of the performative in our mind as general as possible, we further consider the sharing of information needs where three parties are involved. As the size of a team or the complexity of domain tasks increases, the mental model about information needs of teammates may vary significantly among members of the team. For instance, as a team scales up in size, the team is often organized into subteams, each of which may be further divided into smaller subteams, and so on. In such cases, team knowledge might be distributed among several subteams. Hence, agents in one subteam might not be able to anticipate the information needs of agents in other subteams because they may not share the resources for doing so, such as the subteam process, the plans, task assignments, etc. To enable information sharing among subteams, some agents in a subteam are often designated as the points of contact with other subteams. For instance, an agent who simultaneously participates in the activities of two subteams can be designated as the broker agent of the two subteams. These broker agents play a key role in informing agents outside the subteam about the information needs of agents in the subteam. Such an observation motivates us to introduce a proactive performative involving three parties, by which a broker agent A is expecting a known information provider D to commit to satisfying a third-party agent B's information need. When A and B are the same agent, the semantics is reduced to two-party subscription of information needs.

8.1. Third-party Subscribe

As shown in Fig. 17, we define 3PTSubscribe in terms of Attempt and ProInform. $\mathsf{3PTSubscribe}(A, B, D, \epsilon, N, t_1, t_2, t_3, C_n)$ means that agent A, acting as a broker, subscribes from agent D information need N under the context C_n on behalf of agent B until time t_3.

The ultimate goal of 3PTSubscribe is to let B at time t_3 believe what D believes at t_3 about the information related to the need-expression N. This goal might be unachievable because B's and D's beliefs are out of the control of A, and because the information related to N may be changing from time to time.

Definition 12. $\mathsf{3PTSubscribe}(A, B, D, \epsilon, N, t_1, t_2, t_3, C_n) \triangleq$
$(t_1 < t_2 < t_3)?;\ \mathsf{Attempt}(A, \epsilon, P, Q, C_n, t_1, t_2)$, where
$P = \mathsf{Bel}(B, \mathit{info}_{t_3}(D, N), t_3)$,
$Q = \exists t'' \cdot (t_1 \leqslant t'' < t_2) \wedge \mathsf{MB}(\{A, D\}, \rho, t'')$, where
$\rho = \exists C_p, \exists t_b \cdot (t'' \leqslant t_b < t_2) \wedge \gamma \wedge \mathsf{Int.Th}(A, \psi \wedge \phi, t_1, t_b, C_p)$, where
$\gamma = [C_p \supseteq \{C_n, \mathsf{Bel}(A, \mathsf{InfoNeed}(B, N, t_3, C_n), t_1),$ (c1)
$\mathsf{Bel}(A, \mathit{has.info}(D, N, t_1), t_1),\ \neg \mathit{has.info}(A, N, t_1),$ (c2)
$\neg \mathsf{Bel}(A, \mathsf{Bel}(D, \mathsf{InfoNeed}(B, N, t_3, C_n), t_1), t_1)\}],$ (c3)
$\psi = \mathsf{Bel}(D, \mathsf{Bel}(A, \mathsf{InfoNeed}(B, N, t_3, C_n), t_1), t_b)$,
$\phi = \mathsf{Int.Th}(D, \delta, t_b, t_b, C_n)$, where
$\delta = \forall t' \leqslant t_3, I \cdot [\mathit{BChange}(D, N, t') \wedge (I = \mathit{info}_{t'}(D, N))] \Rightarrow$
$\exists t_a, t_c \cdot \mathsf{Int.To}(D, \mathsf{ProInform}(D, B, \epsilon', I, N, t_a, t_c, t_3, C_n), t', t_a, C_n)$,
$\mathit{BChange}(D, N, t) \triangleq \mathit{info}_t(D, N) \neq \mathit{info}_{t-1}(D, N)$.

Fig. 17. The definition of third-party subscribe.

The intermediate effect of 3PTSubscribe is to establish a mutual belief between A and D that A intends that (1) ψ: D believes that A believed B has an information need N by t_3 under the context C_n, and (2) ϕ: D intends that whenever acquiring new information related to N (i.e., D's belief about N has changed, which is represented by $BChange(D, N, t)$), D intends to send the information to B by ProInform as long as B still needs it. The mutual belief is established at some time t'' before t_2; the uncertainty of t'' is due to the uncertain factors such as delays of communication, the reliability of communication. For the same reason, the Int.Th in the mutual belief to be established also refers to an uncertain time point t_b. t_b is somewhere between t'' and t_2 because A is intending to change B's beliefs after rather than before the establishment of the mutual belief.

Similar to ProInform, the performative context C_p includes the context of the information need (C_n) and A's belief of the information need, which together serve as escape conditions in C_p. In addition, since A's intention will be established as a mutual belief, the context of the information need will be known to the addressee (agent D). This allows agent D to avoid delivering unneeded information when the context C_n no longer holds. Lines $c2$ and $c3$ in C_p are constraints. If $has.info(A, N, t_1)$ held, A would have performed ProInform rather than 3PTSubscribe. A cannot perform 3PTSubscribe if no agent known to A can be the potential information provider regarding N. Also, A will perform 3PTSubscribe only to those potential providers who, in A's opinion, do not believe in the information need known to A. For those providers who know B's information need, A would assume they will help B as far as possible without A's 3PTSubscribe.

We now compare 3PTSubscribe with the approaches identified in the beginning of this section. When applied to three parties, (8.2) can be upgraded to:

$$\mathsf{Request}(A, D, \epsilon, \mathsf{Inform}(D, B, \epsilon', info(B, N), t_1, t_2), t, t_a, \Theta). \qquad (8.2')$$

(8.3) can also be restructured to involve three parties:

$$\mathsf{Request}(A, D, \epsilon, \alpha, t_1, t_2, \Theta); \mathsf{ProInform}(A, B, \epsilon', I, N, t_3, t_4, t', C_n), \qquad (8.3')$$

where α is the action that A requests D to do (e.g., Inform), and the performance of α will result in A's awareness of information I.

Compared to (8.2′), in (8.3′) agent A can get the information needed by B as a by-product. However, neither (8.2′) nor (8.3′) is equivalent to 3PTSubscribe in semantics because neither of them allows the sharing of information needs. Nevertheless, (8.2′) and (8.3′) are useful in certain cases. For instance, if the information needed is static, (8.2′) is better than 3PTSubscribe, because the former relieves the information-providing agent from monitoring I for detecting changes.

Definition 12 characterizes the semantics of third-party subscribing information needs. In particular, when the broker agent A and the information needer B refer to the same agent, i.e., $\mathsf{3PTSubscribe}(A, A, D, \epsilon, N, t_1, t_2, t_3, C_n)$, it means agent A issues a subscription request on its behalf to an information service provider D regarding N.

8.2. Conversation protocol of 3PTSubscribe

To design protocols for 3PTSubscribe, we start with the potential responses of the addressee to a 3PTSubscribe. As shown in Fig. 18, we define two kinds of acceptance and two kinds of refusal in response to 3PTSubscribe.

Definition 13 (*Responses to* 3PTSubscribe).
$\mathsf{WAcceptSub}(D, B, A, \epsilon, N, t_1, t_2, t_3, C_n) \triangleq \mathsf{Inform}(D, A, \epsilon, \psi, t_1, t_2)$, where
$\psi = \mathsf{Bel}(D, \mathsf{InfoNeed}(B, N, t_3, C_n), t_1)$;
$\mathsf{SAcceptSub}(D, B, A, \epsilon, N, t_1, t_2, t_3, C_n) \triangleq \mathsf{Inform}(D, A, \epsilon, \psi, t_1, t_2)$, where
$\psi = \mathsf{Bel}(D, \mathsf{InfoNeed}(B, N, t_3, C_n), t_1) \wedge \mathsf{Int.Th}(D, \delta, t_1, t_1, C_n)$, where
$\delta = \forall t' \leqslant t_3, I \cdot [BChange(D, N, t') \wedge (I = info_{t'}(D, N))] \Rightarrow$
$\quad \exists t_a, t_c \cdot \mathsf{Int.To}(D, \mathsf{ProInform}(D, B, \epsilon', I, N, t_a, t_c, t_3, C_n), t', t_a, C_n)$;
$\mathsf{SRejectSub}(D, B, A, \epsilon, N, t_1, t_2, t_3, C_n) \triangleq \mathsf{Inform}(D, A, \epsilon, \psi, t_1, t_2)$, where
$\psi = \mathsf{Bel}(D, \neg\mathsf{InfoNeed}(B, N, t_3, C_n), t_1)$;
$\mathsf{WRejectSub}(D, B, A, \epsilon, N, t_1, t_2, t_3, C_n, C') \triangleq \mathsf{Inform}(D, A, \epsilon, \psi, t_1, t_2)$, where
$\psi = \neg\mathsf{Bel}(D, \mathsf{InfoNeed}(B, N, t_3, C_n), t_1) \wedge \mathsf{Bel}(D, \mathsf{InfoNeed}(B, N, t_3, C'), t_1)$.

Fig. 18. The responses to third-party subscribes.

In the definition of WAcceptSub, an agent D tells the originator of 3PTSubscribe that D only accepts the information need but refuses to make a commitment to serving the information need. In the definition of SAcceptSub, an agent D tells the originator of 3PT-Subscribe that D not only accepts the information need but also adopts a commitment to the information need.

WAcceptSub may be used when agent D is prevented from making the strong commitment due to more urgent things. Most likely, D cannot make a commitment to helping B because D is not an information provider of N as A imagined. In such a case, D's reply can be taken as an indirect speech act, from which A may infer that D cannot provide the information relevant to N. However, there may exist other reasons. For instance, D may be simply too busy. Thus, the acceptance of the information need offers agent D the opportunity of helping agent B later, and with the flexibility of deciding when and how to provide help. For instance, D could help B by using ProInform or even issuing a 3PTSubscribe to yet another information provider.

SAcceptSub carries a stronger semantics: the accepting agent D attempts to let the listening agent A know that D accepted B's information need known from A, and D adopted an intention (Int.Th) at t_1 to help B whenever necessary. Such an instant intention corresponds to the intention that A intended D to adopt within the mutual belief that A attempted to establish in performing 3PTSubscribe.

The addressee may reject a 3PTSubscribe if it simply does not believe in the information need anticipated by the originator of 3PTSubscribe (in this case, it is meaningless for the addressee of 3PTSubscribe to make a commitment to provide help); or the addressee disagrees with the originator of 3PTSubscribe on the context of the information need. We call the former refusal SRejectSub and the later WRejectSub.

Upon receiving a SRejectSub, the agent A may revise its model of B's information needs, or still hold the information need and issue another 3PTSubscribe toward another information provider.

The receiver of a WRejectSub can refine its model of B's information need (i.e., change the context). By reflection, A may be able to improve its capability of anticipating B's information needs in the future. After receiving a WRejectSub, A may perform another

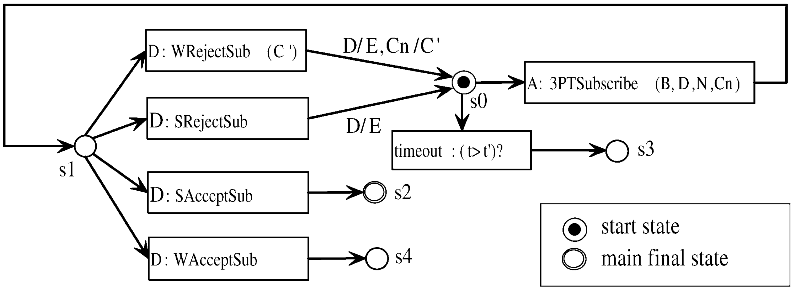

Fig. 19. A conversation protocol involving 3PTSubscribe.

3PTSubscribe to D or another known provider with the context of the information need changed to C'.

Fig. 19 describes a conversation protocol involving 3PTSubscribe. The states are labeled $s0$ to $s4$, where $s2$, $s3$, and $s4$ are possible final states and $s2$ is the main final state of the protocol. Initially, an initiator agent A performs 3PTSubscribe toward potential information provider D with respect to need-expression N and the context C_n (the protocol may not be triggered at all if A keeps silent). Agent D may respond with one of four possible choices:

(1) D strongly accepts A's 3PTSubscribe by performing SAcceptSub. In this case, the protocol terminates and A can discharge its helpful commitment to B;
(2) D performs a WAcceptSub in response to A's 3PTSubscribe, and the protocol terminates at state $s4$. In this case, the protocol may be extended such that A will persuade D to make a commitment to B's information need;
(3) D performs SRejectSub. In this case, agent A may perform another 3PTSubscribe with the potential information provider D replaced by some other agent E;
(4) D performs WRejectSub. In this case, agent A may perform another 3PTSubscribe with the information need context C_n replaced by C' and probably D replaced by some other information provider E.

The recursive process involved in the last two cases can terminate when A chooses to keep silent at state $s0$ (timeout). Then, at state $s3$, agent A will retract its belief of B's information need regarding N.

8.3. *Some properties of* 3PTSubscribe

Theorem 6. *Successful performance of the* 3PTSubscribe *act establishes a mutual belief between the sender and the addressee that the sender believes the delivered information need. Formally,*

$$\models \mathsf{SuccDone}(A, \mathsf{3PTSubscribe}(A, B, D, \epsilon, N, t_1, t_2, t_3, C_n)) \Rightarrow \mathsf{MB}(\{A, D\}, \mathsf{Bel}(A, \mathsf{InfoNeed}(B, N, t_3, C_n), t_1), t_2).$$

Proof. (1) Assume $\mathsf{SuccDone}(A, \mathsf{3PTSubscribe}(A, B, D, \epsilon, N, t_1, t_2, t_3, C_n))$.

(2) By (1), the definition of SuccDone (Definition 7 in Section 3.5), the definition of 3PTSubscribe (cf. Fig. 17), and the possible world semantics of Int.Th and MB, there exists a time $t'' < t_2$ such that $\mathsf{MB}(\{A, D\}, \rho, t'')$, where

$$\rho = \exists C_p, \exists t_b \cdot (t'' \leqslant t_b < t_2) \wedge \gamma \wedge \mathsf{Int.Th}(A, \psi, t_1, t_b, C_p),$$

where

$$\psi = \mathsf{Bel}(D, \mathsf{Bel}(A, \mathsf{InfoNeed}(B, N, t_3, C_n), t_1), t_b),$$
$$\gamma = [C_p \supseteq \{C_n, \mathsf{Bel}(A, \mathsf{InfoNeed}(B, N, t_3, C_n), t_1), \mathsf{Bel}(A, \mathit{has.info}(D, N, t_1), t_1), \neg \mathit{has.info}(A, N, t_1), \neg\mathsf{Bel}(A, \mathsf{Bel}(D, \mathsf{InfoNeed}(B, N, t_3, C_n), t_1), t_1)\}].$$

(3) A is assumed to be sincere, thus by Axiom 2 in Fig. 3 we have

$$\mathsf{Int.Th}(A, \mathsf{Bel}(D, \mathsf{Bel}(A, \mathsf{InfoNeed}(B, N, t_3, C_n), t_1), t_b), t_1, t_b, C_p) \Rightarrow [\mathsf{Bel}(A, \mathsf{InfoNeed}(B, N, t_3, C_n), t_1) \wedge \mathsf{Int.Th}(A, \mathsf{Hold}(\mathsf{InfoNeed}(B, N, t_3, C_n), t_b), t_1, t_b, C_p)].$$

(4) From (2) and (3) we have $\mathsf{MB}(\{A, D\}, \mathsf{Bel}(A, \mathsf{InfoNeed}(B, N, t_3, C_n), t_1), t'')$.

(5) By Assumption 2 (cf. Section 3.1), we can conclude that

$$\mathsf{MB}(\{A, D\}, \mathsf{Bel}(A, \mathsf{InfoNeed}(B, N, t_3, C_n), t_1), t_2). \qquad \square$$

Theorem 7. *Successful performance of a* 3PTSubscribe *with respect to B and N followed by a successful* SAcceptSub *by the addressee of* 3PTSubscribe *establishes a mutual belief between the two agents that the addressee of* 3PTSubscribe *comes to believe B will need N and adopts a commitment to helping B. Formally* ($t_0 < t_1 < t_2 < t_3 < t'$),

$$\models \mathsf{SuccDone}(A, \mathsf{3PTSubscribe}(A, B, D, \epsilon_1, N, t_0, t_1, t', C_n)) \wedge \mathsf{SuccDone}(D, \mathsf{SAcceptSub}(D, B, A, \epsilon_2, N, t_2, t_3, t', C_n)) \Rightarrow \mathsf{MB}(\{A, D\}, \mathsf{InfoNeed}(B, N, t', C_n) \wedge \delta, t_3), \quad \textit{where}$$
$$\delta = \forall t \leqslant t', I \cdot [\mathit{BChange}(D, N, t) \wedge (I = \mathit{info}_t(D, N))] \Rightarrow \exists t_a, t_c \cdot \mathsf{Int.To}(D, \mathsf{ProInform}(D, B, \epsilon', I, N, t_a, t_c, t', C_n), t, t_a, C_n).$$

Proof. (1) Assume that

$$\mathsf{SuccDone}(A, \mathsf{3PTSubscribe}(A, B, D, \epsilon_1, N, t_0, t_1, t', C_n))$$

and

$$\mathsf{SuccDone}(D, \mathsf{SAcceptSub}(D, B, A, \epsilon_2, N, t_2, t_3, t', C_n))$$

hold.

(2) By applying Theorem 6, we have

$$\mathsf{MB}(\{A, D\}, \mathsf{Bel}(A, \mathsf{InfoNeed}(B, N, t', C_n), t_0), t_1).$$

(3) By Proposition 1 and the definition of SAcceptSub, we have

$$\mathsf{MB}(\{D, A\}, \mathsf{Bel}(D, \mathsf{InfoNeed}(B, N, t', C_n), t_2), t_3).$$

(4) By Assumption 3 (cf. Section 3.5), A's commitments in 3PTSubscribe and D's commitments in SAcceptSub prevent them from changing beliefs about B's need before t_3. Thus, by the positive introspection of Bel, we can conclude that

$$\mathsf{MB}(\{A, D\}, \mathsf{InfoNeed}(B, N, t', C_n), t_3).$$

(5) By Proposition 1 and the definition of SAcceptSub, we have

$$\mathsf{MB}(\{D, A\}, \mathsf{Int.Th}(D, \delta, t_2, t_2, C_n), t_3),$$

where

$$\delta = \forall t \leqslant t', I \cdot [BChange(D, N, t) \wedge (I = info_t(D, N))] \Rightarrow$$
$$\exists t_a, t_c \cdot \mathsf{Int.To}(D, \mathsf{ProInform}(D, B, \epsilon', I, N, t_a, t_c, t', C_n), t, t_a, C_n).$$

The theorem is thus proved from (4) and (5). □

From Theorem 7, we can conclude that the protocol shown in Fig. 19 is correct in the sense that successful execution of 3PTSubscribe and SAcceptSub can achieve the goal of the protocol.

Similarly, we can prove the following results.

Theorem 8. (1) *A successful performance of* 3PTSubscribe *with respect to agent B and N followed by a successful performance of* WAcceptSub *by the addressee of the* 3PTSubscribe *establishes a mutual belief between the two agents about B's information need regarding N*;

(2) *A successful performance of* 3PTSubscribe *with respect to agent B and N followed by a successful* WRejectSub *by the addressee of the* 3PTSubscribe *establishes a mutual belief between the two agents that the addressee of the* 3PTSubscribe *believes that N will be needed by B under a different context*;

(3) *A successful performance of* 3PTSubscribe *with respect to agent B and N followed by a successful* SRejectSub *by the addressee of the* 3PTSubscribe *establishes a mutual belief between the two agents that the addressee of the* 3PTSubscribe *believes that B will not need N.*

Formally ($t_0 < t_1 < t_2 < t_3 < t'$),

$$(1)\ \models \mathsf{SuccDone}(A, \mathsf{3PTSubscribe}(A, B, D, \epsilon_1, N, t_0, t_1, t', C_n)) \wedge$$
$$\mathsf{SuccDone}(D, \mathsf{WAcceptSub}(D, B, A, \epsilon_2, N, t_2, t_3, t', C_n))$$
$$\Rightarrow \mathsf{MB}(\{A, D\}, \mathsf{InfoNeed}(B, N, t', C_n), t_3),$$

$$(2)\ \models \mathsf{SuccDone}(A, \mathsf{3PTSubscribe}(A, B, D, \epsilon_1, N, t_0, t_1, t', C_n)) \wedge$$
$$\mathsf{SuccDone}(D, \mathsf{WRejectSub}(D, B, A, \epsilon_2, N, t_2, t_3, t', C_n, C'))$$
$$\Rightarrow \mathsf{MB}(\{A, D\}, \neg\mathsf{Bel}(D, \mathsf{InfoNeed}(B, N, t', C_n), t_2), t_3) \wedge$$
$$\mathsf{MB}(\{A, D\}, \mathsf{Bel}(D, \mathsf{InfoNeed}(B, N, t', C'), t_2), t_3),$$

$$(3)\ \models \mathsf{SuccDone}(A, \mathsf{3PTSubscribe}(A, B, D, \epsilon_1, N, t_0, t_1, t', C_n)) \wedge \mathsf{SuccDone}(D, \mathsf{SRejectSub}(D, B, A, \epsilon_2, N, t_2, t_3, t', C_n)) \Rightarrow \mathsf{MB}(\{A, D\}, \mathsf{Bel}(D, \neg\mathsf{InfoNeed}(B, N, t', C_n), t_2), t_3).$$

Theorem 9. *A* 3PTSubscribe *with respect to some information need N can be performed toward an agent even when the agent actually does not believe any information relevant to N. Formally,*

$$\mathsf{SuccDone}(A, \mathsf{3PTSubscribe}(A, B, D, \epsilon, N, t_1, t_2, t_3, C_n)) \wedge \neg\mathsf{Bel}(D, \mathit{has.info}(D, N, t_1), t_1)$$

is satisfiable.

Proof. We construct a possible-world structure K_1 at t_1 that satisfies the context of 3PT-Subscribe and $\neg\mathsf{Bel}(D, \mathit{has.info}(D, N, t_1), t_1)$. Let the real world w_0 be the world when the 3PTSubscribe is being performed. Let w_1 and w_2 be the worlds that are both belief and intention accessible by A, and let w_2 and w_3 be the worlds that are belief accessible by D. Let $\mathsf{InfoNeed}(B, N, t', C_n)$ be true at w_1, w_2 and w_3, and let $\mathit{info}(D, N)$ be defined at w_1 and w_2, but not at w_3. Similarly, we can construct a structure K_2 at t'' ($t'' < t_2$) when the honest effort of 3PTSubscribe is established. The transition from K_1 to K_2 is straightforward since communication is reliable. Then, $\mathsf{3PTSubscribe}(A, B, D, \epsilon, N, t_1, t_2, t_3, C_n)$ and $\neg\mathsf{Bel}(D, \mathit{has.info}(D, N, t_1), t_1)$ can both be satisfied by this model. □

Theorem 10. *Suppose* 3PTSubscribe *is the only means considered by all teammates in reacting to others' information needs. There may exist a loop that prevents teammates from helping an information needer.*

Proof. We construct a loop using only 3PTSubscribe and WAcceptSub. Suppose that agent B will need the information described by N before t' and agent A is aware of this information need, that is, $\mathsf{InfoNeed}(B, N, t', C_n)$ and $\mathsf{Bel}(A, \mathsf{InfoNeed}(B, N, t', C_n), t)$ hold. Suppose that A managed to reconcile its potential intentions and eventually performed a 3PTSubscribe at t_A to agent D, whom A believed is a potential provider of N. From Theorem 9, we know that agent D may not be able to directly satisfy B's needs. Thus, it is possible for D to perform a WAcceptSub to A—discharging A from retrying to help B—and then perform a 3PTSubscribe at t_D to agent E, who is a potential provider of N from D's perspective. Such a process may be repeated until some agent, say M, who was requested to help B using 3PTSubscribe, also performed a 3PTSubscribe to agent A. Then, agents $A, D, E, \ldots, M, A$ form a loop, and all of them were discharged from helping B by the WAcceptSub performed by the successive teammate in the loop. □

However, the loops as described in Theorem 10 can be easily avoided if the information about the first initiator of 3PTSubscribe is maintained as a part of the performative context and, before initiating another 3PTSubscribe, each agent checks to avoid circularities.

Theorem 11 states that an agent could assist its teammates by adopting a potential intention-to regarding 3PTSubscribe.

Theorem 11. *If agent A does not have any information related to N but it believes that agent D had the information, it will consider helping B using* 3PTSubscribe. *Formally,*

$$\models [\mathsf{Bel}(A, \mathsf{InfoNeed}(B, N, t', C_n), t) \wedge \neg has.info(A, N, t) \wedge \\ \mathsf{Bel}(A, has.info(D, N, t), t) \wedge \neg\mathsf{Bel}(A, has.info(B, N, t'), t) \wedge \\ CUPP(A, has.info(B, N, t'))] \Rightarrow \\ \exists t_1, t_2, C' \cdot \mathsf{Pot.Int.To}(A, \mathsf{3PTSubscribe}(A, B, D, \epsilon, N, t_1, t_2, t', C_n), t, t_1, C').$$

Proof. (1) Assume

$$\neg\mathsf{Bel}(A, has.info(B, N, t'), t), \neg has.info(A, N, t), \\ \mathsf{Bel}(A, has.info(D, N, t), t), \mathsf{Bel}(A, \mathsf{InfoNeed}(B, N, t', C_n), t),$$

and

$$CUPP(A, has.info(B, N, t')).$$

(2) Since $\neg has.info(A, N, t)$ and $\mathsf{Bel}(A, \mathsf{InfoNeed}(B, N, t', C_n), t)$, by Axiom 12 in Fig. 13 we have that

$$\mathsf{Pot.Int.Th}(A, has.info(B, N, t'), t, t', C_n)$$

holds.

(3) From $CUPP(A, has.info(B, N, t'))$, the potential intention

$$\mathsf{Pot.Int.Th}(A, has.info(B, N, t'), t, t', C_n)$$

can be upgraded to $\mathsf{Int.Th}(A, has.info(B, N, t'), t, t', C_n)$.

(4) Since $\mathsf{Bel}(A, has.info(D, N, t), t)$ holds, from Theorem 7 we know that A's 3PT-Subscribe followed by D's SAcceptSub may lead to B's belief of the information described by N because D will send B the relevant information whenever necessary. Thus there exist t_1 and t_2 such that

$$\mathsf{Lead}(A, \mathsf{3PTSubscribe}(A, B, D, \epsilon, N, t_1, t_2, t', C_n), has.info(B, N, t'), t, t_1, \Theta).$$

(5) By Axiom 3 in Fig. 5, (1), (3), and (4) we have

$$\mathsf{Pot.Int.To}(A, \mathsf{3PTSubscribe}(A, B, D, \epsilon, N, t_1, t_2, t', C_n), t, t_1, \Theta \wedge C_n).$$

Here, the context of A's potential intention is composed of the context C_n of B's information need and the constraints of performing 3PTSubscribe. This completes the proof. □

9. Discussion

9.1. The role of observability in reasoning about others' beliefs

An agent needs to hold certain beliefs about the addressees when performing ProInform and 3PTSubscribe. For instance, ProInform is performed under the context that the speaker A has beliefs $\mathsf{CBel}(A, B, I, t)$ and $\mathsf{Bel}(A, \mathsf{UBif}(B, I, t), t)$, while 3PTSubscribe is performed under the context that the speaker A has belief $\mathsf{Bel}(A, \mathit{has.info}(D, N, t), t)$. However, belief reasoning itself is extremely difficult in general [8,9,13,95], and implemented agent systems typically employ various *ad hoc* assumptions to simplify this issue. In particular, an agent A can approximately reason about another agent B's beliefs by leveraging its knowledge of B's observability.

Let $\mathit{CObs}(B, I, C_I)$ represent that agent B can observe the truth value of proposition I if the constraint C_I holds. Suppose agent A believes $\mathit{CObs}(B, I, C_I)$ (this can be achieved by allowing agents to share their observability). Then A could approximately model B's belief regarding I using certain assumptions of which the following is one:

Assumption 4. $\mathsf{Bel}(A, \mathit{CObs}(B, I, C_I), t) \wedge \mathsf{Bel}(A, C_I, t) \Rightarrow \mathsf{Bel}(A, \mathsf{Bif}(B, I, t), t)$.

It says that if agent A believes agent B can observe information I and the corresponding constraint is satisfiable, A could assume that B is aware of I. This assumption reduces the likelihood that agent A provides unnecessary help to B. For instance, the derived belief prevents A from using ProInform to help B.

The following assumption indicates a second use of observability for belief reasoning: inferring teammates' lack of beliefs from their lack of observabilities. Let $\mathit{Obs}(B)$ denote the set of propositions that agent B can potentially observe, i.e.,

$$\mathit{Obs}(B) = \{I \mid \exists C_I \cdot \mathit{CObs}(B, I, C_I)\}.$$

Assumption 5. (1) $\mathsf{Bel}(A, I \notin \mathit{Obs}(B), t) \Rightarrow \mathsf{Bel}(A, \mathsf{UBif}(B, I, t), t)$.
(2) $\mathsf{Bel}(A, \mathit{CObs}(B, I, C_I), t) \wedge \mathsf{Bel}(A, \neg C_I, t) \Rightarrow$
$\mathsf{CBel}(A, B, I, t) \vee \mathsf{Bel}(A, \mathsf{UBif}(B, I, t), t)$.

Assumption 5.1 says that agent A could assume that agent B is unaware of information I if A knows that I is beyond B's observability; Assumption 5.2 says that even if I is indeed among those that B can observe, in cases where the constraint for B to observe I does not hold from A's perspective, A still assumes that B is unaware of I, or B's belief about I conflicts with A's belief.

However, an agent may establish incorrect models of other agents' mental states by inferring their lack of beliefs only based on their lack of observability. For instance, in Assumption 5.1, agent B might have acquired I from some other agents even though it lacks the observability regarding I. In Assumption 5.2, it is possible that A and B's beliefs about I coincide, or it may even be the case that it is A's belief about I is incorrect, not B's. To bring the assumptions closer to the reality, additional constraints could be added as the premise, such as "A is the only agent in the team that can observe I," and "A knows that other teammates are all too busy to help B".

But on the other hand, by assuming B's unawareness of I or the existence of belief-conflict regarding I, agent A could choose appropriate communicative actions (e.g., ProInform, 3PTSubscribe) to help B. It is true that more than one agent may be able to anticipate B's information need, which might result in redundant help. However, from the whole team's viewpoint, redundant helps in information delivery are sometimes useful, with each serving as a backup to the rest.

9.2. Implications

Our proposed framework for specifying proactive information delivery behaviors in agent teamwork has several implications.

First, it allows an agent to deliver needed information to teammates who could not have requested the information themselves due to their limited sensing capabilities or their incomplete knowledge about the distributed environment. Thus, it formally specifies the proactive information delivery behavior embodied in effective human teams. Software agents empowered with such capabilities can be used to better simulate, train, or support the information fusion, interpretation, and decision-makings of agent teams that may include human agents in the loop.

Second, even though broadcast can be used to deliver information, it would result in an overwhelming amount of information for agents to process. In this new information age, the information that a team needs to filter, fuse, and interpret under time pressure increases at a rapid speed as the domain complexity increases. For instance, the US Army estimates that, without filter and fusion at lower echelon levels, more than 600,000 reports will need to be processed every hour by a team of brigade battle staff under the vision of the digitized Objective Force. Similarly, a team of anti-terrorism analysts needs to filter, analyze, and fuse overwhelming amounts of information from a wide variety of information sources (e.g., satellite images, intelligence reports and radio transmissions). Each member of these teams needs to make decisions under time pressures. Delivering only the information relevant to the needs of teammates promises to enable teammates to make better decisions without overloading them.

Third, agents committed to others' information needs will continuously monitor the environment to detect changes relevant to the information needs. In addition, agents can automatically terminate their "monitoring" activity for a teammate's information need when the need becomes irrelevant (e.g., the context of the need is no longer valid). This is very important in improving the flexibility and rationality of agents implemented for collaborative information-pushing.

Fourth, our proposed framework supports not only the exchanges of information but also the flows of information needs. This will enable agents in a team to establish and evolve a "shared mental model" regarding others' information needs. Such a shared mental model is valuable for further enhancing a team of agents in their intelligent information exchange.

Fifth, the ways of anticipating others' information needs proposed in this paper lay the foundations for developing algorithms for agents to dynamically reason about information needs of their teammates [100]. For instance, RPD (Recognition-primed decision-making), proposed by Klein [54], is a well-known naturalistic decision making model and has been widely adopted in implementing decision-support systems. When RPD is used in a teamwork setting [36], algorithms for anticipating the decision-making information needs can be developed by following Axiom 8.

Finally, intelligent proactive information delivery is a critical issue in large agent infrastructures like Grid [51], where joint activities may involve trans-architecture teams of agents such as STEAM [91], CAST [99] and D'Agents [40]. It is highly desirable to provide well-defined semantics for proactive communicative actions used in these agent infrastructures, as well as mechanisms for accessing shared ontologies. Other proactive performatives may be identified from the interaction of teammates with heterogeneous architectures. Their semantics can be given in the same way as for ProInform and 3PTSubscribe.

10. Comparisons

We compare our work with the related literature from four aspects.

10.1. Information needs

Proactive information delivery behavior has long been recognized by researchers studying indirect speech acts in the field of human discourse understanding [3,4,64]. Psychological studies about human teams have also identified proactive information delivery as one of the key behaviors of effective teamwork [32,68,86]. For instance, in a study by Dickinson and McIntyre [32], "recognize other members' need for certain information" is listed as part of the ATOM teamwork dimensions. The identification of users' information needs and the shifting of their information needs are considered as critical issues in developing user-oriented information systems such as decision support systems [2]. However, the concept of information needs has never been formally characterized before in agent teamwork settings. This paper not only studied the properties of information needs and categorized the information needs in agent teamwork, it also connects the anticipated information needs to potential commitments so that agents could choose appropriate communicative actions to satisfy teammates' information needs.

10.2. The SharedPlans theory

Even though the SharedPlans theory was originally motivated by certain problems within human discourse understanding, the concept of shared plans actually provides a foundation for theories of collaborative agent behaviors and has been successfully applied to study general teamwork problems [66,77,91,99]. On the one hand, we adopted the SharedPlans theory as one of the cornerstones of our framework because it provides a clean model of shared team processes, which is critical in enabling agents to anticipate teammates' information needs. On the other hand, the work in this paper can be taken as an extension of the SharedPlans formalism. By exploring the potential communication-related axioms, this paper moves a step toward the goal established by Grosz and Kraus [42]: to develop a more complete set of communication axioms in SharedPlans theory for establishing requisite mutual beliefs and for ensuring the satisfaction of intentions-that. Of course, it is unlikely that a single set of axioms will cover all eventualities because communication is inherently context-dependent [72]. Our goals are to establish a framework for proactive information delivery behavior and, in the future, to fully examine how need-driven communications may affect the performance of teams with both human and software agents.

Our work is related to Lochbaum's work on *knowledge precondition subdialogues*, where it was shown that SharedPlans provide a more detailed account of an agent's motivations for producing an utterance or initiating a discourse [65,67]. In particular, an agent's (information needer's) reflection on its lack of knowledge about an action to be performed initiates an information-seeking dialogue; knowing such a desire, the hearer tries to help the speaker to acquire the necessary knowledge. While Lochbaum's approach of using the SharedPlans theory can be extended to cover information exchange regarding physical preconditions and constraints, it relies too much on discourse understanding, in that it requires that the information provider be able to infer the speaker's needs from the *preceding* utterances. The more interesting behavior, which this paper is trying to cover, is that an agent could anticipate another's needs and *push* the relevant information selectively without being asked directly or indirectly.

10.3. Semantics of performatives

Research on speech acts can be traced to Austin's work [5], which was later extended by Searle [79]. In early 1990s, Cohen and Levesque proposed the idea of "performative-as-attempt" [22] and modeled speech acts as actions of rational agents in the framework of intentions [20]. Henceforward, this has been adopted as the standard way of assigning mentalistic semantics to communicative acts. For instance, Arcol [11], KQML [58], and FIPA's ACL [1] are the representatives of agent communication languages proposed so far. The strictly declarative semantics of performatives in these languages are all framed in terms of mental attitudes. For example, Arcol uses performance conditions to specify the semantics of communicative acts. KQML adopts a more operational approach by using preconditions, postconditions and completion conditions. FIPA ACL is heavily influenced by Arcol, wherein the semantics of performatives are specified by feasibility preconditions and rational effect, both of which are formulas of a semantic language SL.

The semantics of communicative acts defined in this paper also draw heavily on Cohen and Levesque's seminal work. However, our work is distinguished from the others by emphasizing need-driven communications. That is, prior to delivering information to other agents, an agent has to know explicitly (at least from its own perspective) that the information to be sent is what the receiving agent will need in its pursuit of certain team or individual goals. Need-driven communication is partially supported in Arcol. For instance, in Arcol, if agent A is informed that agent B needs some information, A would supply that information as if A had been requested by B. Here, the *inform* is actually treated as an indirect *request*. However, the need-driven communication in Arcol works in a reactive rather than proactive way. In contrast, in our framework, both are allowed due to the support for reasoning about teammates' information needs. More specifically, the semantics of ProInform and 3PTSubscribe rely on the performer's awareness of the beneficiary agents' information needs. In addition, our approach allows agents to make long-term commitments regarding others' information needs. In other words, being aware of agent B's information need, agent A will try to update B about the relevant information whenever A observes a change.

ProInform (proactive inform) is comparable to the performative *tell* in KQML although they are not equivalent *per se*. Both *tell* and ProInform require an agent to only offer solicited information to others. The modal operator *WANT* in KQML, which stands for the psychological state of desire, plays the same role as InfoNeed. However, the semantics of

WANT is left open for generality. InfoNeed can be viewed as an explicit way of expressing information needs under certain contexts.

Both 3PTSubscribe and the performative *broker_one* in KQML [58] involve three parties (but have different semantics). 3PTSubscribe is initiated by a broker agent, who needs to know the other two parties. An agent cannot perform 3PTSubscribe if it does not know any potential information provider regarding the information need under concern. This is desirable to encourage intelligent and efficient communications. Comparatively, the speaker of *broker_one* only needs to know the broker agent. This is more flexible because the broker agent can decide the addressee of the embedded speech act later. While *broker_one* can be simulated using Inform and Request, 3PTSubscribe cannot be easily simulated in KQML.

The performative *Proxy* in FIPA [1] is defined in terms of Inform (i.e., the sending agent informs the recipient that the sender wants the receiver to select target agents denoted by a given description and to perform the embedded communicative act to them). While *Proxy* captures a rather weaker third-party semantics, Huber et al. [46] defined a stronger third-party semantics for PROXY and PROXY-WEAK. Both PROXY and PROXY-WEAK are based on Request. PROXY imposes significant commitments on the intermediate agent while PROXY-WEAK reduces the burden placed upon the intermediate agent. "PROXY of an Inform" and "PROXY-WEAK of an Inform" are different from 3PTSubscribe. PROXY of an Inform requires the middle agent to believe the information that the speaker wants him/her to forward to the target agent. Even though PROXY-WEAK of an Inform relaxes this requirement, both still require that the speaker already hold the information to be delivered. 3PTSubscribe, focusing on information needs, applies to situations where the speaker does not have the information needed by others.

More recently, social agency is emphasized as a complement to mental agency due to the fact that communication is inherently public [82]. This requires the social construction of communication be treated as a first-class notion rather than as a derivative of the mentalist concepts. For instance, in a study by Singh [83], speech acts are defined in terms of social commitments, which are obligations relativized to both the beneficiary agent and the whole team as the social context. Kumar [55] argued that joint commitments may simulate social ones because PWAG entails a social commitment provided that the persistent goal is made public. The semantics of ProInform and 3PTSubscribe adopt a richer notion of context, which includes the context of the information need under concern. Thus, an agent could stop providing information once the context is no longer valid. The context can also be enriched to specify protocols in force, as suggested by Smith et al. [84], and even social constraints. This enables agents to take the public perspective (e.g., team goals) into consideration while intending to perform a communicative act.

Compositionality is useful in defining meta-level performatives in terms of elementary ones. For instance, it is shown that ASK (regarding Yes–No questions) could be defined in terms of Request and Inform [24]. However, as we have shown, the semantics of ProInform and 3PTSubscribe cannot be simply defined using compositionality. The semantics of communicative acts has also been studied from a team's point of view [25,56]. However, to thoroughly investigate the semantics of *proactive* communicative acts used in teamwork settings requires an agent to be able to reason about teammates' information needs. Our work in this paper is the first effort toward this end.

10.4. Conversation protocols

Communicative acts are not simply individual actions; they should be understood as part of an ongoing social interaction [82]. To fully understand the ties between the semantics of communicative acts and patterns of these acts, conversation policies or protocols have been studied heavily in the ACL field [18,55,57,72,84,96]. However, conversation protocols involving proactive performatives have been neglected. Although the protocols proposed in this paper are rather simple, they not only help in investigating the characteristics of proactive communications enabled by proactive communicative acts, but they also offer a guide to exploring more complex protocols that support proactive information delivery behaviors.

Conversation protocols are traditionally specified using finite state machines [83,84]. Enhanced Dooley graphs [71], Colored Petri Nets [26], and a Landmark-based representation [55] have also been proposed to specify a richer semantics for protocols. For instance, in a Landmark-based representation, a protocol (family) is specified as a sequence of waypoints (landmarks) that must be followed in order to accomplish the goal associated with that protocol. Concrete protocols are realized by specifying action expressions for each landmark transition such that performing the action expressions can result in the landmark transitions [55]. In this paper we only considered concrete protocols, which are viewed as patterns of communicative acts, and their semantics tie to those of the involved individual acts. Petri-nets, as a modeling tool of parallel behaviors, is used to specify the protocols because the petri-net representations can be easily translated into shared plans through which the teammates can coordinate their communicative behaviors.

11. Summary

In this paper, we presented a formal framework for the proactive information delivery behaviors in agent teamwork. The main contribution of this paper is three-fold.

First, we studied the key concept of the framework–information need. In particular, we used reference expressions to represent information and information need expressions; clarified the concept of information need by introducing a modal operator InfoNeed, examining the properties of InfoNeed, and exploring its relationships with other mental modal operators; analyzed levels of information needs based on the idea of social inference trees; formally identified four types of information needs prevalent in agent teamwork; and proposed and justified the axioms for anticipating others' information needs based on shared team processes. Such formal treatment enables agent systems to explicitly represent and reason about information needs. Furthermore, it may allow a team of agents to establish shared mental models regarding their information needs.

Second, we established a formal foundation for proactive information delivery behaviors. The framework mainly consists of 8 axioms. Axiom 3 characterizes chains of helping behaviors in large agent teams. Axiom 5 allows an agent who intends others to be involved in a team activity to release them from the obligations whenever the intentional context no longer holds. Axioms 6, 8, 9, 10, and 11 state how to anticipate teammates' action-performing information needs, decision-making information needs, goal-protection information needs, and goal-escape information needs. Axiom 12 relates information needs to potential intentions-that. Together with axiom 3, it allows an agent to choose appropriate actions to satisfy its own or others' information needs. In general, the framework formally specifies two important teamwork behaviors: to anticipate teammates' information needs and to proactively help teammates with their information needs. The framework not only provides a better understanding of the underlying assumptions required to justify the proactive behavior, but also provides a coherent basis for the specification and design of agent teams with proactive information delivery capabilities.

Third, while several research groups (e.g., CAST [99]) have been employing the Cohen–Levesque semantics in systems implemented upon the SharedPlans theory, there has been lacking a formal grounding of that semantics in the SharedPlans theory. This paper filled this gap by re-formulating the Cohen–Levesque semantics of communicative acts using the SharedPlans formalism. Based on this, we formally provided semantics for two proactive communicative acts (i.e., ProInform and 3PTSubscribe) and analyzed proactive communication in multi-agent systems using the developed formalism. In particular, the semantics focuses on the deliberation about others' information needs and allows the information needs to be transferred as meta-level information. We also examined the properties of the two proactive performatives and designed a conversation protocol for each. The protocols based on the semantics of proactive performatives are useful in analyzing and understanding the proactive information flows at different abstract levels in teamwork settings. Agents using the protocols are able to establish a shared mental model regarding teammates' information needs; the shared mental model could further enhance team performance in terms of intelligent information exchange.

There are several important issues that deserve further studies. For instance, indirect speech acts occur prevalently in human discourse. Similarly, indirect information needs is also a worthwhile topic in agent teamwork supporting information exchange. Further research is required to extend the current social inference tree approach to thoroughly explore this interesting field.

An agent may get overloaded by adopting too many commitments. It is worthwhile to investigate the effects on team performance of different ways by which an agent resolves the conflicts between helpful commitments (e.g., proactive communicative actions) and its own responsibilities. Proactive information delivery behaviors among teammates improve team intelligence but may inevitably introduce redundant information exchanges because multiple agents in a team might deliver the same piece of information to the information needer. How to reduce redundant information deliveries among teammates with proactive information delivery capabilities also remains to be elucidated.

In addition, the formal semantics of InfoNeed and the in-depth analysis of the computational complexity of reasoning within the framework are also left for future studies.

Acknowledgements

This research has been supported by AFOSR MURI grant No. F49620-00-1-0326. We would like to thank the referees for their valuable suggestions and comments. We thank

Thomas R. Ioerger at the Texas A&M University for the earlier discussions regarding the topics. We are also very grateful to Margaret Hopkins for her help in improving the writing.

References

[1] FIPA: Agent Communication Language Specification, http://www.fipa.org/, 2002.
[2] M.J. Albers, Goal-driven task analysis: improving situation awareness for complex problem-solving, in: Proceedings of the 16th Annual International Conference on Computer Documentation, Quebec, Canada, 1998, pp. 234–242.
[3] J. Allen, Recognizing intentions from natural language utterances, in: M. Brady, R.C. Berwick (Eds.), Computational Models of Discourse, MIT Press, Cambridge, MA, 1983, pp. 107–166.
[4] J. Allen, C. Perrault, Analyzing intention in utterances, Artificial Intelligence 15 (1980) 143–178.
[5] J. Austin, How to Do Things with Words, Oxford University Press, Oxford, 1962.
[6] K. Bach, Communicative intentions, plan recognition, and pragmatics: comments on Thomason and on Litman and Allen, in: P. Cohen, J. Morgan, M. Pollack (Eds.), Intentions in Communication, MIT Press, Cambridge, MA, 1990, pp. 389–400.
[7] M. Beetz, D. McDermott, Declarative goals in reactive plans, in: J. Hendler (Ed.), Proceedings of the First International Conference on Artificial Intelligence Planning Systems(AIPS92), AAAI Press, 1992, pp. 3–12.
[8] S. Brainov, T. Sandholm, Reasoning about others: Representing and processing infinite belief hierarchies, in: Proceedings of the ICMAS'00, 2000, pp. 71–78.
[9] A. Brandenburger, E. Dekel, Hierarchies of beliefs and common knowledge, J. Economic Theory 59 (1993) 189–198.
[10] M.E. Bratman, Intention, Plans, and Practical Reason, Harvard University Press, 1987.
[11] P. Breiter, M. Sadek, A rational agent as a kernel of a cooperative dialogue system: Implementing a logical theory of interaction, in: Proceedings of ECAI-96 Workshop on Agent Theories, Architectures, and Languages, Springer, Berlin, 1996, pp. 261–276.
[12] G. Brown, Characterizing indirect speech acts, Amer. J. Computational Linguistics 6 (3–4) (1980) 150–166.
[13] M. Cadoli, M. Lenzerini, The complexity of propositional closed world reasoning and circumscription, J. Comput. System Sci. 48 (1994) 255–310.
[14] J.A. Cannon-Bowers, E. Salas, A framework for developing team performance measures in training, in: M. Brannick, et al. (Eds.), Team Performance Assessment and Measurement: Theory, Methods, and Applications, Lawrence Erlbaum Associates, Hillsdale, NJ, 1997.
[15] J.A. Cannon-Bowers, E. Salas, S.A. Converse, Cognitive psychology and team training: Training shared mental models and complex systems, Human Factors Soc. Bull. 33 (1990) 1–4.
[16] J.A. Cannon-Bowers, E. Salas, S.A. Converse, Shared mental models in expert team decision making, in: Individual and Group Decision Making, Castellan, NJ, 1993, pp. 221–246.
[17] B. Chaib-Draa, F. Dignum, Trends in agent communication language, Computational Intelligence 18 (2) (2002) 89–101.
[18] A. Chopra, M. Singh, Nonmonotonic commitment machines, in: F. Dignum (Ed.), Advances in Agent Communication, in: Lecture Notes Artif. Intell., vol. 2922, Springer, Berlin, 2004, pp. 183–200.
[19] P.R. Cohen, The need for identification as a planned action, in: Proceedings of the Seventh International Joint Conference on Artificial Intelligence, Vancouver, BC, 1981.
[20] P.R. Cohen, H.J. Levesque, Intention is choice with commitment, Artificial Intelligence 42 (1990) 213–261.
[21] P.R. Cohen, H.J. Levesque, Performatives in a rationally based speech act theory, in: Proceedings of the 28th Annual Meeting of the Association for Computational Linguistics, 1990, pp. 79–88.
[22] P.R. Cohen, H.J. Levesque, Rational interaction as a basis for communication, in: Intentions in Communication, MIT Press, Cambridge, MA, 1990, pp. 221–255.
[23] P.R. Cohen, H.J. Levesque, Teamwork, Nous 25 (4) (1991) 487–512.
[24] P.R. Cohen, H.J. Levesque, Communicative actions for artificial agents, in: Proceedings of the International Conference on Multi-Agent Systems, AAAI Press, 1995, pp. 65–72.

[25] P.R. Cohen, H.J. Levesque, I.A. Smith, On team formation, in: G. Holmstrom-Hintikka, R. Tuomela (Eds.), Contemporary Action Theory, vol. 2: Social Action, Kluwer, Dordrecht, 1997, pp. 87–114.
[26] R. Cost, Y. Chen, T. Finin, Y. Labrou, Y. Peng, Using coloured petri nets for conversation modeling, in: F. Dignum, M. Greaves (Eds.), Issues in Communication, Springer, Berlin, 2000, pp. 178–192.
[27] J. Crant, Proactive behavior in organization, J. Management 26 (3) (2000) 435–462.
[28] J. Crant, T. Bateman, Charismatic leadership viewed from above: The impact of proactive personality, J. Organizational Behavior 21 (2000) 63–75.
[29] G. Cybenko, B. Brewington, The foundations of information push and pull, in: D. O'Leary (Ed.), Mathematics of Information, Springer, Berlin, 1998.
[30] E. Davis, Knowledge preconditions for plans, J. Logic Comput. 4 (5) (1994) 721–766.
[31] K. Decker, V. Lesser, Designing a Family of Coordination Algorithms, AAAI Press, 1995, pp. 73–80.
[32] T. Dickinson, R. McIntyre, A conceptual framework for teamwork measurement, in: M.T. Brannick, E. Salas, C. Prince (Eds.), Team Performance Assessment and Measurement: Theory, Methods and Applications, Erlbaum, Hillsdale, NJ, 1997, pp. 19–44.
[33] J. Digman, Personality structure: Emergence of the five-factor model, Annu. Rev. Psychol. 41 (1990) 417–440.
[34] F. Dignum, M. Greaves, Issues in Agent Communication, Lecture Notes Artif. Intell., vol. 1916, Springer, Berlin, 2000.
[35] R. Fagin, J.Y. Halpern, Y. Moses, M. Vardi, Reasoning About Knowledge, MIT Press, Cambridge, MA, 1995.
[36] X. Fan, S. Sun, M. McNeese, J. Yen, Extending the recognition-primed decision model to support human-agent collaboration, in: Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multi Agent Systems (AAMAS), ACM Press, 2005.
[37] X. Fan, J. Yen, The semantics of potential intentions, in: Proceedings of the Twentieth National Conference on Artificial Intelligence (AAAI-05), AAAI Press, 2005.
[38] H.J. Genrich, K. Lautenbach, System modeling with high-level petri nets, Theoret. Comput. Sci. 1 (1981) 109–136.
[39] J. Giampapa, K. Sycara, Team-oriented agent coordination in the RETSINA multi-agent system, Technical Report CMU-RI-TR-02-34, Robotics Institute, Carnegie Mellon University, 2002.
[40] R. Gray, G. Cybenko, D. Kotz, R. Peterson, D. Rus, D'agents: Applications and performance of a mobile-agent system, Software, Practices and Experience 32 (6) (2002) 543–573.
[41] M. Greaves, H. Holmback, J. Bradshaw, What is a conversation policy?, in: Issues in Agent Communication, Springer, Berlin, 2000, pp. 118–131.
[42] B. Grosz, S. Kraus, Collaborative plans for complex group actions, Artificial Intelligence 86 (1996) 269–358.
[43] B. Grosz, S. Kraus, The evolution of sharedplans, in: A. Rao, M. Wooldridge (Eds.), Foundations and Theories of Rational Agencies, 1999, pp. 227–262.
[44] B. Grosz, C. Sidner, Plans for discourse, in: P. Cohen, J. Morgan, M. Pollack (Eds.), Intentions in Communication, MIT Press, Cambridge, MA, 1990, pp. 417–444.
[45] J. Grunig, Information and decision making in economic development, J. Quarterly (1969) 565–575.
[46] M.J. Huber, S. Kumar, P.R. Cohen, D.R. McGee, A formal semantics for proxycommunicative acts, in: J.-J.C. Meyer, M. Tambe (Eds.), Intelligent Agents VIII, ATAL 2001, in: Lecture Notes Comput. Sci., vol. 2333, Springer, Berlin, 2002.
[47] L. Hunsberger, Making shared plans more concise and easier to reason about, in: J. Müller, M.P. Singh, A.S. Rao (Eds.), Proceedings of the 5th International Workshop on Intelligent Agents V: Agent Theories, Architectures, and Languages (ATAL-98), vol. 1555, Springer, Heidelberg, 1999, pp. 81–98.
[48] D.R. Ilgen, D.A. Major, J.R. Hollenbeck, D.J. Sego, Team research in the 1990's, in: M. Chemers, R. Ayman (Eds.), Leadership Theory and Research: Perspectives and Directions, Academic Press, San Diego, 1993, pp. 245–270.
[49] L. Jauch, K. Kraft, Strategic management of uncertainty, Acad. Management Rev. 11 (4) (1986) 777–790.
[50] N.R. Jennings, Controlling cooperative problem solving in industrial multi-agent systems using joint intentions, Artificial Intelligence 75 (2) (1995) 195–240.
[51] M. Kahn, C. Cicalese, CoABS grid scalability experiments, Autonomous Agents and Multi-Agent Systems 7 (1–2) (2003) 171–178.

[52] D. Kinny, M. Ljungberg, A.S. Rao, E. Sonenberg, G. Tidhar, E. Werner, Planned team activity, in: C. Castelfranchi, E. Werner (Eds.), Artificial Social Systems—MAAMAW-92, in: Lecture Notes Artif. Intell., vol. 830, Springer, Heidelberg, 1992, pp. 226–256.

[53] B. Kirkman, B. Rosen, Beyond self-management: Antecedents and consequences of team empowerment, Acad. Management J. 42 (1999) 58–74.

[54] G.A. Klein, Recognition-primed decisions, in: W.B. Rouse (Ed.), Advances in Man-Machine Systems Research, vol. 5, JAI Press, Greenwich, CT, 1989, pp. 47–92.

[55] S. Kumar, M.J. Huber, P.R. Cohen, D.R. McGee, Toward a formalism for conversation protocols using joint intention theory, Computational Intelligence 18 (2) (2002) 174–228.

[56] S. Kumar, M.J. Huber, D.R. McGee, P.R. Cohen, H.J. Levesque, Semantics of agent communication languages for group interaction, in: Proceedings of 17th National Conference on Artificial Intelligence (AAAI 2000), Austin, TX, AAAI Press/MIT Press, 2000, pp. 42–47.

[57] Y. Labrou, Standardizing agent communication, in: V. Marik, O. Stepankova (Eds.), Multi-Agent Systems and Applications (Advanced Course on Artificial Intelligence), 2001, pp. 74–97.

[58] Y. Labrou, T. Finin, Semantics and conversations for an agent communication language, in: M. Huhns, M. Singh (Eds.), Readings in Agents, Morgan Kaufmann, San Mateo, CA, 1998, pp. 235–242.

[59] Y. Labrou, T. Finin, Semantics for an agent communication language, in: M. Wooldridge, A. Rao, M. Singh (Eds.), Intelligent Agents IV: Agent Theories, Architectures and Languages, in: Lecture Notes Comput. Sci., vol. 1365, Springer, Berlin, 1998, pp. 209–215.

[60] T. LaPorte, P. Consolini, Working in practice but not in theory: Theoretical challengers of high-reliability organizations, J. Public Administrative Res. Theory 1 (1) (1991) 19–47.

[61] J. Larson, C. Christensen, Groups a problem-solving units: Toward a new meaning of social cognition, British J. Social Psychol. (1993) 5–30.

[62] H.J. Levesque, P.R. Cohen, J. Nunes, On acting together, in: Proceedings of AAAI-90, Boston, MA, Morgan Kaufmann, San Mateo, CA, 1990, pp. 94–99.

[63] Z. Lin, K. Carley, Proactive or reactive: An analysis of the effect of agent style on organizational decision-making performance, Intelligent Systems in Accounting, Finance and Management 2 (1993) 271–289.

[64] D.J. Litman, J.F. Allen, Discourse processing and commonsense plans, in: Intentions in Communication, MIT Press, Cambridge, MA, 1990, pp. 365–388.

[65] K.E. Lochbaum, A model of plans to support inter-agent communication, in: Proceedings of the AAAI Workshop on Planning for Inter-agent Communication, Seattle, WA, 1994, pp. 26–32.

[66] K.E. Lochbaum, Using collaborative plans to model the intentional structure of discourse, PhD thesis, Harvard University, Technical Report TR-25-94. Cambridge, MA, 1994.

[67] K.E. Lochbaum, The use of knowledge preconditions in language processing, in: C. Mellish (Ed.), Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence, San Francisco, Morgan Kaufmann, San Mateo, CA, 1995, pp. 1260–1266.

[68] R. McIntyre, E. Salas, Measuring and managing for team performance: emerging principles from complex environments, in: R. Guzzo, E. Salas (Eds.), Team Effectiveness and Decision Making in Organizations, Jossey-Bass, San Francisco, 1995, pp. 149–203.

[69] L. Morgenstern, Knowledge preconditions for actions and plans, in: Proceedings of IJCAI-87, Milan, Italy, 1987, pp. 867–874.

[70] OAIS. Reference model for an open archival information system, http://www.ccsds.org/documents/pdf/CCSDS-650.0-R-1.pdf, 1999.

[71] C. Parunak, Visualizing agent conversations: Using enhanced Dooley graphs for agent design and analysis, in: Proceedings of 2nd International Conference on Multi-Agent Systems (ICMAS-96), 1996, pp. 275–282.

[72] J. Pitt, A. Mamdani, A protocol-based semantics for an agent communication language, in: Proceedings of IJCAI-99, Stockholm, Sweden, 1999, pp. 486–491.

[73] M.E. Pollack, Plans as complex mental attitudes, in: P. Cohen, J. Morgan, M. Pollack (Eds.), Intentions in Communication, MIT Press, Cambridge, MA, 1990, pp. 77–103.

[74] C. Porter, J. Hollenbeck, D. Ilgen, A. Ellis, B. West, H. Moon, Towards understanding backing up behaviors in work teams: The role of personality and the legitimacy of need for backing up others, J. Appl. Psychol. 88 (2003) 391–403.

[75] A. Rao, M. Georgeff, Formal models and decision procedures for multi-agent systems, Technical Report 61, Australian Artificial Intelligence Institute, Melbourne, Australia, 1995.

[76] J.R. Rentsch, R.J. Hall, Members of great teams think alike: A model of team effectiveness and schema similarity among team members, in: M.M. Beyerlein, D.A. Johnson (Eds.), Advances in Interdisciplinary Studies of Work Teams, in: Self-Managed Work Teams, vol. 1, JAI Press, Greenwich, CT, 1994, pp. 223–262.
[77] C. Rich, C. Sidner, Collagen: When agents collaborate with people, in: Proceedings of the International Conference on Autonomous Agents (Agents'97) 1997, pp. 284–291.
[78] W. Rouse, J. Cannon-Bowers, E. Salas, The role of mental models in team performance in complex systems, IEEE Trans. Syst. Man Cybernet. 22 (6) (1992) 1296–1308.
[79] J. Searle, Speech Acts: An Essay in the Philosophy of Language, Cambridge University Press, Cambridge, 1969.
[80] J. Searle, Indirect speech acts, in: P. Cole, J. Morgan (Eds.), Syntax and Semantics. III. Speech Acts, Academic Press, New York, 1975, pp. 59–82.
[81] J.R. Searle, How performatives work, Linguistics and Philosophy 12 (1989) 535–558.
[82] M.P. Singh, Agent communication languages: Rethinking the principles, IEEE Computer 31 (12) (1998) 40–47.
[83] M.P. Singh, A social semantics for agent communication languages, in: F. Dignum, M. Greaves (Eds.), Issues in Agent Communication, Springer, Heidelberg, 2000, pp. 31–45.
[84] I. Smith, P. Cohen, J. Bradshaw, M. Greaves, H. Holmback, Designing conversation policies using joint intention theory, in: Proceedings of the Third International Conference on Multi-Agent Systems (ICMAS98), 1998, pp. 269–276.
[85] I.A. Smith, P.R. Cohen, Toward a semantics for an agent communications language based on speech-acts, in: Proceedings of the Annual Meeting of the American Association for Artificial Intelligence (AAAI-96), Portland, OR, AAAI Press, 1996, pp. 24–31.
[86] K.A. Smith-Jentsch, J.H. Johnson, S.C. Payne, Measuring team-related expertise in complex environments, in: J.A. Cannon-Bowers, E. Salas (Eds.), Making Decisions Under Stress: Implications for Individual and Team Training, American Psychological Association, 2000, pp. 61–87.
[87] L. Sterling, E. Shapiro, The Art of Prolog, MIT Press, Cambridge, MA, 1986.
[88] P. Stone, M. Veloso, Task decomposition, dynamic role assignment, and low-bandwidth communication for real-time strategic teamwork, Artificial Intelligence 110 (1999) 241–273.
[89] K. Sycara, M. Lewis, Forming shared mental models, in: Proceedings of the 13th Annual Meeting of the Cognitive Science Society, 1991, pp. 400–405.
[90] K. Sycara, M. Lewis, Modeling teams of specialists, in: Proceedings of the 27th Hawaii International Conference on System Sciences, IEEE Society Press, 1994, pp. 271–280.
[91] M. Tambe, Towards flexible teamwork, J. Artificial Intelligence Res. 7 (1997) 83–124.
[92] G. Tidhar, C. Heinze, M. Selvestrel, Flying together: Modeling air mission teams, J. Appl. Intelligence 8 (1998) 195–218.
[93] R. Tuomela, K. Miller, We-intentions, Philos. Stud. 53 (2–3) (1988) 367–389.
[94] United States Air Force Scientific Advisory Board, Report on Building the Joint Battlespace Infosphere, vols. 1, 2, SAB-TR-99-02, December 1999.
[95] J. Vidal, E. Durfee, The impact of nested agent models in an information economy, in: Proceedings of the ICMAS'96, 1996, pp. 377–384.
[96] B. Vitteau, M.-P. Huget, Modularity in interaction protocols, in: F. Dignum (Ed.), Advances in Agent Communication, in: Lecture Notes Artif. Intell., vol. 2922, Springer, Berlin, 2004, pp. 291–309.
[97] M. Wooldridge, N.R. Jennings, Towards a theory of cooperative problem solving, in: Proc. Modelling Autonomous Agents in a Multi-Agent World (MAAMAW-94), Odense, Denmark, 1994, pp. 15–26.
[98] M. Wooldridge, N.R. Jennings, Intelligent agents: theory and practice, Knowledge Engineering Review 10 (2) (1995) 115–152.
[99] J. Yen, J. Yin, T. Ioerger, M. Miller, D. Xu, R. Volz, CAST: Collaborative agents for simulating teamworks, in: Proceedings of IJCAI'2001, Seattle, WA, 2001, pp. 1135–1142.
[100] J. Yin, M.S. Miller, T.R. Ioerger, J. Yen, R.A. Volz, A knowledge-based approach for designing intelligent team training systems, in: Proceedings of the Fourth International Conference on Autonomous Agents, 2000, pp. 427–434.