Belief Coordination by Default

Yasuhiro Katagiri
ATR Media Integration & Communications Research Laboratories
2-2 Hikaridai Seika Soraku Kyoto 619-02 Japan
E-mail: katagiri@mic.atr.co.jp

Abstract

Use of confirmations in dialogue is mundane and ubiquitous, but theoretically it has been a puzzle how confirmations serve their supposed end, namely to secure the establishment of mutual understandings between dialogue participants. We propose a solution to this confirmation puzzle by introducing the notion of *defeasible mutual knowledge*. The notion is based upon our intuition that mutual beliefs obtained in message exchanges are inherently of nonmonotonic nature in the sense that they are revocable by the negating evidences obtained later. We point out that there are two aspects to the notion of defeasible knowledge: optimism and error discovery. We give an analysis of defeasible knowledge within the framework of an epistemic temporal modal logic over distributed systems, and propose a set of conditions for defeasible mutual knowledge. We examine a simple message transmission example and show that confirmations are necessary to obtain defeasible mutual knowledge. We also examine organizational structures for defeasible mutual knowledge for larger groups.

Introduction

Use of confirmations is extremely mundane and almost ubiquitous in our daily human interactional activities. However, theoretically speaking, it has been a puzzle how confirmations serve their supposed function, namely to secure the establishment of mutual understandings among participants of interactional activities(Cohen & Levesque 1993). On the one hand, research typically exemplified on the coordinated attack problem in knowledge-based distributed systems(Halpern & Moses 1990) demonstrated that it is impossible to attain new mutual knowledge via communication through unreliable channels. On the other hand, if the communication channels were reliable, issuing informational messages would be sufficient to obtain mutual knowledge and there would be no need to rely on confirmations for the first place. If so, then why we use confirmations? We call this puzzle the *confirmation puzzle.*

On the face of the strong negative result on the attainability of mutual knowledge, several attempts have been made(Halpern & Moses 1990; Fagin *et al.* 1995; Neiger 1988) in distributed systems research to weaken the notion of mutual knowledge. Their main direction has been to weaken the requirement of simultaneity on occurrences of internal state transition events in obtaining mutual knowledge, but they have not yet successfully addressed the problem of arbitrary message loss type channel unreliabilities not uncommon in our daily communication settings.

We believe the status of mutual knowledge attained in our daily communication is of a default nature. When we speak to a hearer, we assume we have reached a mutual understanding unless the other party indicates otherwise. There have been several attempts, notably by (Perrault 1990), (Appelt & Konolige 1988) and (Thomason 1990), to incorporate the idea of defaults into conversational belief revision. But these studies haven't addressed the issue of confirmations. They are primarily concerned with coordinated belief revision based on the lack of negative information available at the time the revision is made. And as such they don't provide any account for why a positive confirmation is issued even when everything goes smoothly.

Basic picture behind these works is that of a reasoning agent trying to reason as much as possible about a given state of the world from incomplete information available to her. However, dialogue management operations available to the agents, such as confirmations, clarifications and repairs, are actions that modify the conversational states toward those that are more in line with originally intended states of mutual understanding. More adequate picture for capturing these phenomena would be an acting agent trying to maintain correspondence between public conversational states and individual informational states.

Dialogue as joint activity view(Clark 1992; Grosz & Sidner 1990; Cohen & Levesque 1991) also endorses a view of agents actively contributing to ongoing discourse, rather than a view of a hearer simply inferring meanings of what the speaker says. Clark(1993) points out the significance of the process of managing troubles or errors in dialogue modeling. Adopting an acting agent view enables us to address issues of discovery

and subsequent recovery from errors. There is usually a time delay between the time of an utterance and the subsequent time of signaling and discovery of errors in obtaining mutual understanding. An error once detected triggers a recovery operation, which leads to restoration of initially intended states. We argue in this paper that the solution to the confirmation puzzle lies in this delayed defeasibility. Even though it is impossible to attain real mutual knowledge, it is possible, with the employment of confirmations, to attain a version of mutual knowledge that takes into account this delayed defeasibility and recovery from errors.

Duality of Context

The concept of defeasiblity in reasoning has been extensively studied in research on nonmonotonic reasoning and belief revision. The notion of minimization plays a central role in many of these studies; works on circumscription(McCarthy 1986), chronological ignorance(Shoham 1986), logic of knowledge and justified assumptions(Lin & Shoham 1992), works on rationality postulates on belief revision and update(Alchourrón, Gärdenfors, & Makinson 1985; Katsuno & Mendelzon 1991). We think of minimization as a way to capture a double standard or duality of the role context plays in reasoning. The duality in question here is the contrast between the actual context on the one hand, and the idealized or the normal context on the other. Minimization in nonmonotonic reasoning provides us with a way of determining a set of legitimate conclusions a reasoning agent can draw from available information, if the context in which reasoning takes place is normal. Non-monotonic conclusions are not guaranteed to be correct if the actual context doesn't satisfy this normality condition. We also find similar but somewhat different duality of context in knowledge-based distributed systems.

Nonmonotonic reasoning

Let's take a look at minimization in circumscription and its associated normality condition on contexts. Normality condition will become more explicit when we recast the circumscriptive reasoning into epistemic terms. Shoham(1988) gives a way of capturing circumscription in terms of minimal knowledge entailment. Think of the following simple set of nonmonotonic reasoning rules about watching a movie on TV. You can watch a movie by turning on your TV provided that there are no abnormalities in the situation, which include the power line being unplugged and the TV being broken.

$$\texttt{TV-ON} \wedge \neg \texttt{ab} \supset \texttt{MOVIE} \quad (1)$$
$$\texttt{UNPLUGGED} \supset \texttt{ab}$$
$$\texttt{BROKEN} \supset \texttt{ab}$$

Circumscribing on ab is equivalent to adding the condition (2) below, and then minimizing on knowledge.

$$\texttt{ab} \supset K\texttt{ab} \quad (2)$$

The added condition (2) explicitly captures the normality condition on contexts. Drawing a nonmonotonic conclusion MOVIE from TV-ON under (1) through circumscription is legitimate only when the context satisfy (2), namely, all abnormalities must be known by the reasoning agent. The same condition can be derived with epistemic reconstruction of default logic formulation of nonmonotonic reasoning.

The type of normality conditions in nonmonotonic reasoning reflects emphasis on reasoning. They may reason about actions, but performance of actions itself is not within the picture. When performing actions, abnormalities may not always be known beforehand when deliberating on outcomes of actions. With action errors like slipping, fumbling and other types of unreliabilities, abnormalities show up only after actions are actually taken. To incorporate performance of actions and its associated possibility of errors, we will require a weaker version of the normality condition on contexts, namely that abnormalities will eventually be known to agents.

Knowledge Consistency

Duality of context, in a slightly different flavor, can also be found in knowledge-based distributed systems. We can assume two different systems, given a knowledge-based protocol. One system serves to give interpretations for knowledge conditions in the protocol, and the other system gives the result of the protocol execution. Neiger(1988) exploits this duality and proposes the notion of knowledge consistency, which intuitively amounts to the agent-wise indiscernibility between the two systems. Coexistence of these two systems is also a manifestation of duality of the ideal and the actual context for the agents. Neiger showed that satisfaction of a specification based only on agents' local states is equivalent between the ideal and the actual systems if they satisfy knowledge consistency conditions. As far as agent's local states are concerned, agents can assume the context is ideal when it is guaranteed to be knowledge consistent with the actual context. Knowledge consistency has been useful in certain applications, including weakening of the simultaneity requirement for common knowledge. But completely ignoring actual contexts misses the opportunity of error discovery and subsequent error correction, and it amounts to mere wishful thinking in some cases if important errors are left unnoticed.

Modal logical conception of knowledge is external and implicit(Fagin *et al.* 1995) in that it is actually a relational condition between agent states and environment states, and cannot be directly referred to in selecting actions by agents with incomplete grasp of their environments(Katagiri 1996). Duality of contexts is, from the point of view of the agents, a natural way to realize both reasoning and acting by default. Reasoning gives truthful information only when the supposition of ideal context is a true supposition, and act-

ing reveals deviation of actual context from the supposition and thereby providing opportunities to recover from errors.

A Distributed System Model for Agent-Environment Interaction

We give in this section basic definitions of distributed system model on which to develop our notion of defeasible mutual knowledge. We use standard concepts and definitions on distributed systems given in (Fagin *et al.* 1995). Our formalization emphasizes the role of environments in mediating between actions and knowledge states of agents.

A distributed system model assumes one or more agents executing their programs in an environment. Execution of programs can bring about changes both in environment and in agents. It is not guaranteed that execution of programs brings about a constant effect. One agent's program step may interfere with other agents' program step, and environment may also non-deterministically intervene program execution and change the outcome.

Basics

We define a distributed system to be a set of n agents $\{a_1, \ldots, a_n\}$ executing their programs in an environment e.

States A *system state* s can be described by a tuple of an environmental state and local states of each agents, $\langle s_e, s_1, \ldots, s_n \rangle$. A *local state* s_i of each agent a_i at a given instant is taken from a corresponding set of local states S^i. An environmental state s_e is similarly taken from a set S^e. The behavior of the system can be specified as transitions in the set of system states S. We will also write a_0 and s_0 for e and s_e, respectively, for expository convenience.

A *basic proposition* corresponds to a set of system states. We denote a set of system states in which a basic proposition p holds by S_p. A basic proposition is a *local state condition* σ_i of an agent a_i, if it is solely determined by a set of a_i's local states.

Protocols and transition functions We assume a set ACT_i of basic acts for each agent a_i. A *protocol* Π is a tuple of local protocol Π_i's. Π_i specifies which act to execute based on local states of a_i; $\Pi_i : S^i \rightarrow 2^{ACT_i}$. We assume that the environment may act non-deterministically, but other agents are deterministic, that is, the value of Π_i is a singleton set for $i \geq 1$. A *transition function* τ specifies the transition of the entire system state given all the acts executed by agents and the environment: $s_{t+1} = \tau(act_e, act_1, \ldots, act_n)(s_t)$. τ represents the outcome of actions of all the agents. This includes each agent's internal state change upon receiving information about actions performed.

Runs, points and systems A *run* r is a function $r : N \rightarrow S$ which gives a system state $r(t)$ for each time point t. We write $r_i(t)$ for the corresponding local state of a_i. We call $\langle r, t \rangle$ a *point* in a run r. We say for an agent a_i, points $\langle r, t \rangle$ and $\langle r', t' \rangle$ are *a_i-equivalent* if $r_i(t) = r'_i(t')$, and write $\langle r, t \rangle \sim_i \langle r', t' \rangle$. A *system* A is identified with a set of runs that corresponds to all executions of the protocol Π under the transition function τ.

Knowledge and time We regard a system of runs as a Kripke structure and introduce modalities of both knowledge and time.

Knowledge is a relational condition between agents' local states and entire system states. We assume S5 for knowledge, and write $K_{a_i}\varphi$ to represent that the agent a_i knows that φ. From knowledge modalities for individual agents, we define group knowledge E_G (everybody in the group G knows): $E_G\varphi \stackrel{\text{def}}{=} \bigwedge_{a_i \in G} K_{a_i}\varphi$. Mutual knowledge $MK_G\varphi$ in a group G is defined in terms of $E_G\varphi$ as the largest fixed point of the equation $X = E_G(\varphi \wedge X)$. Eventual group knowledge $E^{\diamond}_G\varphi$ is defined by weakening the condition of simultaneity of the knowing. Everybody in the group G eventually knows φ if and only if for every agent a_i in G there is a time point t_i such that a_i knows φ at t_i. Eventual mutual knowledge $MK^{\diamond}_G\varphi$ is similarly defined in terms of $E^{\diamond}_G\varphi$.

Since each run in a system has a linear temporal structure, we introduce temporal modalities, $\diamond$ and its dual $\Box$, in a standard way.

Satisfaction conditions

When φ holds at a point $\langle r, t \rangle$ in the system A, we say $\langle r, t \rangle$ in A satisfies φ and write $A, r, t \models \varphi$. We also write $A \models \varphi$ when all the points in A satisfies φ. We state the satisfaction conditions for formulas below.

- When p is a basic proposition, $A, r, t \models p$ iff $r(t) \in S_p$.
- $A, r, t \models \neg\varphi$ iff it is not the case that $A, r, t \models \varphi$.
- $A, r, t \models \varphi \wedge \psi$ iff $A, r, t \models \varphi$ and $A, r, t \models \psi$.
- $A, r, t \models K_{a_i}\varphi$ iff for all $\langle r', t' \rangle \in A$ if $\langle r, t \rangle \sim_i \langle r', t' \rangle$ then $A, r', t' \models \varphi$.
- $A, r, t \models \diamond\varphi$ iff there exists t' such that $t \leq t'$ and $A, r, t' \models \varphi$.

Knowledge consistency and beliefs

Recall that we related the notion of duality of context, ideal and actual, with a picture of agents reasoning and acting by default, who have the capability of incorporating possibilities of errors. We introduce the notion of *weak knowledge consistency*, by extending Neiger's notion of knowledge consistency, which prescribes the condition on the relationship between the actual system A and the ideal system I. We define the notion of beliefs based on the duality of A and I. The idea is that we require every run in the actual system A to be almost indiscernible for all the agents to some run

in the ideal system I, except for some states in which one or more agents notice the deviation and try to get back onto the right track. We define beliefs on top of this duality. An agent a_i believes φ iff φ holds in every a_i-equivalent system state in every ideal system I.

- Weak knowledge consistency of runs

  A run r is *weakly knowledge consistent* with another run r', $r \|_{\preceq} r'$, iff there is a monotone increasing mapping $\rho_i : N \to N$ for each agent a_i such that $\langle r, t\rangle \sim_i \langle r', \rho_i(t)\rangle$ for all i.
- Preference order on systems

  A system A_2 is *preferred* over another system A_1, $A_1 \sqsubseteq A_2$, iff A_2 is a subset of A_1, $A_1 \supseteq A_2$, and for all $r' \in A_1$ there exists $r \in A_2$ such that $r \|_{\preceq} r'$.

When A_2 is preferred over A_1, every run in A_1 has its counterpart with less deviations in A_2. Both weak knowledge consistency relation $\|_{\preceq}$ and preference relation $\sqsubseteq$ are partial order. For any local state condition σ and systems A_1, A_2, if $A_1 \sqsubseteq A_2$ and $A_2 \models \Diamond\sigma$ then $A_1 \models \Diamond\sigma$ also holds.

We define beliefs $B_{a_i}\varphi$ of an agent a_i relative to ideal systems I which are maximally preferred over the actual system A.

- $A, r, t \models B_{a_i}\varphi$ iff for all maximally preferred system I satisfying $A \sqsubseteq I$, for all $\langle r', t'\rangle \in I$, if $\langle r, t\rangle \sim_i \langle r', t'\rangle$ then $I, r', t' \models \varphi$.

An agent a_i believes that φ if φ holds in all a_i-equivalent system states with the assumption that the context is ideal. This definition of beliefs satisfies KD45 axioms. Group beliefs A_G, mutual beliefs MB_G and their eventual counterparts $A_G^\Diamond$ and $MB_G^\Diamond$ are defined in parallel to the definitions given for knowledge.

Defeasible Mutual Knowledge

Informal characterization

We think we can capture the defeasibility of mutual knowledge among acting agents by the following two complementary components. Agents jump to conclusions with incomplete information and act on them. Agents also recover from errors when mistakes are discovered later. We propose to characterize the notion of defeasible mutual knowledge with these two components.

Optimism: Defeasible mutual knowledge would amount to real mutual knowledge, if the environment were ideal.
Error discovery: In reality any deviation from mutual knowledge will jointly be noticed.

Conditions for defeasible mutual knowledge

We first propose a set of conditions for defeasible mutual knowledge in formal terms. We then argue that these conditions capture our intuition on defeasibility of mutual knowledge among acting agents, through examination of logical consequences derived from the proposed conditions.

Condition 1 (DMK condition) To achieve defeasible mutual knowledge of φ in a system A among members of a group G, there has to be a local state condition σ_i for each agent a_i, and they have to satisfy the following conditions:

- **Knowledge in ideal context:**

  For every agent $a_i \in G$, $I \models \sigma_i \Leftrightarrow K_{a_i}\varphi$.
- **Eventual notice of success:**

  For every agent $a_i \in G$, $I \models \Diamond\sigma_i$.
- **Simultaneity:**

  For any pair of agents a_i and a_j, $I \models \sigma_i \Leftrightarrow \sigma_j$.
- **Future error discovery:**

  There exists a subset G' of G such that for every agent $a_i \in G'$, $A \models \sigma_i \supset (\neg\varphi \supset \Diamond K_{a_i}\neg\varphi)$.

First three conditions correspond to the optimism component of defeasible mutual knowledge. If the context were ideal, members of the group have to undergo simultaneous internal state change at some time each of which amounts to the knowing of φ. The fourth condition corresponds to the error discovery component. It is necessary at least for some member(s) of the group to notice if an error has occurred in order to avoid a state of collective illusion. We call members of G' *sober* agents.

Aspect of optimism

We state several logical consequences of the DMK conditions in the form of propositions below. First we examine optimism. General cases can be described by the following proposition.

Proposition 1 If the DMK condition holds, then $A \models MB_G^\Diamond\varphi$

Proof From the equivalence of knowledge in I and beliefs in A, and simultaneity, it is easy to see that for arbitrary $a_i, a_j \in G$, $A \models \sigma_i \supset \Diamond B_{a_j}(\varphi \wedge \sigma_i)$. Hence, $A \models \sigma_i \supset A_G^\Diamond(\varphi \wedge \sigma_i)$. From the induction rule for eventual mutual beliefs, $A \models \sigma_i \supset MB_G^\Diamond\varphi$. From eventual notice of success it follows that $A \models MB_G^\Diamond\varphi$. ∎

So, if the DMK condition holds, an eventual mutual belief among member of G obtains. The following proposition shows that when a stronger condition holds, real mutual belief obtains even though nobody may notice when it happens.

Proposition 2 If, in addition to the DMK condition, $A \models \Diamond \bigwedge_{a_i \in G} \sigma_i$ holds, then $A \models \Diamond MB_G\varphi$.

Proof Let ψ_G to denote $\bigwedge_{a_i \in G} \sigma_i$. It is easy to see that $A \models \psi_G \supset A_G(\varphi \wedge \psi_G)$. From induction rule and the condition of the proposition, the conclusion $A \models \Diamond MB_G\varphi$ follows. ∎

Aspect of error discovery

For error discovery, the following proposition shows that the DMK condition guarantees an eventual mutual knowledge among sober agents in G' that errors will at least be noticed by some members of sober agents.

Proposition 3 If the DMK condition holds, then

$$A \models MK^{\Diamond}_{G'}(\neg\varphi \supset \bigvee_{a_i \in G'} \Diamond K_{a_i} \neg\varphi).$$

Proof Let θ_i be $\neg\varphi \supset \Diamond K_{a_i}\neg\varphi$, and $\theta_{G'}$ be $\bigvee_{a_i \in G'} \theta_i$. $\theta_{G'}$ is equivalent to $\neg\varphi \supset \bigvee_{a_i \in G'} \Diamond K_{a_i}\neg\varphi$. From the eventual notice of success condition, it follows for any $a_i, a_j \in G'$ that $A \models K_{a_i} \Diamond \sigma_j$. Let $\psi_{G'}$ be $\bigwedge_{a_i \in G'} \Diamond \sigma_i$. Using the error discovery condition it follows that $A \models \sigma_i \supset K_{a_i}(\theta_i \wedge \psi_{G'})$. Since $A \models \theta_i \supset \theta_{G'}$, it follows that $A \models \sigma_i \supset K_{a_i}(\theta_{G'} \wedge \psi_{G'})$. Since $\psi_{G'} \supset \Diamond\sigma_i$ for any a_i, $A \models \psi_{G'} \supset \Diamond K_{a_i}(\theta_{G'} \wedge \psi_{G'})$ for every a_i. it follows that $A \models \psi_{G'} \supset E^{\Diamond}_{G'}(\theta_{G'} \wedge \psi_{G'})$, and hence from the induction rule for mutual knowledge, $A \models \psi'_G \supset MK^{\Diamond}_{G'}\theta_{G'}$. Since eventual notice of success implies $A \models \psi_{G'}$, the conclusion of the proposition $A \models MK^{\Diamond}_{G'}\theta_{G'}$ follows. ‖

The following proposition is an easy corollary of the proposition 3. It shows that if knowledge is guaranteed even in the actual context, eventual mutual knowledge is guaranteed.

Proposition 4 If, in addition to the DMK condition, each agent $a_i \in G$ has a local state which corresponds to knowledge of φ not only in the ideal system I, but also in the actual system A, that is, $A \models \sigma_i \Leftrightarrow K_{a_i}\varphi$, then eventual mutual knowledge among all the members of the group G will obtain, $A \models MK^{\Diamond}_{G}\varphi$.

The above results show that in general it is only guaranteed that mutual knowledge among sober agents is of disjunctive nature, that is, only somebody notices the error. In order to obtain a stronger mutual knowledge on error discovery, we need a stronger condition.

Proposition 5 If a stronger version of the future error discovery condition, $A \models \neg\varphi \supset \Diamond K_{a_i}\neg\varphi$ for all sober agent $a_i \in G'$, holds, then

$$A \models MK_G(\neg\varphi \supset MK^{\Diamond}_{G'}\neg\varphi).$$

Proof Since the stronger error recovery condition $A \models \neg\varphi \supset \Diamond K_{a_i}\neg\varphi$ holds for every agent $a_i \in G'$, an error is guaranteed to become eventual group knowledge among G', that is, $A \models \neg\varphi \supset E^{\Diamond}_{G'}\neg\varphi$. The conclusion follows from the induction rule for mutual knowledge. ‖

This proposition shows that when error discovery is always guaranteed, then it is mutual knowledge among all the members of the entire group G that occurrences of errors will become eventual mutual knowledge among the sober agents in G'.

An Example

We examine a simple message transmission example, which is a simplified version of acknowledgment-based message transmission protocol in computer communications. We choose this example as a rough approximation of human communication with confirmations, in which we can see how defeasible mutual knowledge is realized by joint behaviors of a sender and a receiver. The example demonstrates that the use of confirmations is necessary to establish defeasible mutual knowledge.

Imagine two agents, a sender S and a receiver R, executing the following programs within the environment E.

S:	if	$\overline{\mathtt{m}_S}$	then	**send-m**
R:	if	$\mathtt{m}_R \wedge \overline{\mathtt{a}_R}$	then	**send-a**
E:	if	*true*	then	$+ \parallel -$

Local states of S and R both consist of two binary valued components **m** and **a**. They stand for messages and acknowledgments. We use subscripts to indicate to whom the local states belong. S executes an act of sending a message **send-m** whenever she is in states where $\mathtt{m}_S$ is false, and no action is performed otherwise. Similarly R executes an act of sending back an acknowledgment **send-a** whenever he is in states where $\mathtt{m}_R$ is true and $\mathtt{a}_R$ is false, and no action is performed otherwise. The environment E always acts nondeterministically to effect the success or failure of delivery of both the messages and the acknowledgments. We assume fairness in the behavior of the environment. So, it is not the case that neither success nor failure indefinitely continues in any execution. We assume the system starts from a state in which neither the messages nor the acknowledgments are sent out.

Transition function τ of the system is given in Table 1. Each element in the table shows the resultant state of an action on the top of the column performed in a state on the leftmost in the row. So, starting at the state labeled (A), if **send-m** is performed by S and no action is performed by R and the environment is cooperative, then both S and R switch their respective **m** state components, $\mathtt{m}_S$ and $\mathtt{m}_R$, to positive values, while retaining negative values for $\mathtt{a}_S$ and $\mathtt{a}_R$. This corresponds to a transition where both S and R come to think that the message is delivered. Unfilled portion of the table is irrelevant here since cases we are interested in are those where the system starts from a particular state (A).

Our target proposition for mutual knowledge is the proposition φ that both a message and an acknowledgment are successfully delivered. It is a conjunction of two propositions: a proposition p_m that a message has been delivered, and a proposition p_a that an acknowledgment has been delivered. We regard that both p_m and p_a become true once a message or an acknowledgment is successfully delivered and stay true even if the agents keep sending messages/acknowledgments to the

Table 1: Transition function τ for a message transmission system.

	$\begin{bmatrix} m \\ - \\ + \end{bmatrix}$	$\begin{bmatrix} m \\ - \\ - \end{bmatrix}$	$\begin{bmatrix} - \\ a \\ + \end{bmatrix}$	$\begin{bmatrix} - \\ a \\ - \end{bmatrix}$	$\begin{bmatrix} - \\ - \\ +/- \end{bmatrix}$
(A) $\begin{pmatrix} \overline{m_S} & \overline{a_S} \\ \overline{m_R} & \overline{a_R} \end{pmatrix}$	$\begin{pmatrix} m_S & \overline{a_S} \\ m_R & \overline{a_R} \end{pmatrix}$	$\begin{pmatrix} m_S & \overline{a_S} \\ \overline{m_R} & \overline{a_R} \end{pmatrix}$			
(B) $\begin{pmatrix} m_S & \overline{a_S} \\ m_R & \overline{a_R} \end{pmatrix}$			$\begin{pmatrix} m_S & a_S \\ m_R & a_R \end{pmatrix}$	$\begin{pmatrix} \overline{m_S} & \overline{a_S} \\ m_R & a_R \end{pmatrix}$	
(C) $\begin{pmatrix} m_S & a_S \\ m_R & a_R \end{pmatrix}$					$\begin{pmatrix} m_S & a_S \\ m_R & a_R \\ \overline{m_S} & \overline{a_S} \\ \overline{m_R} & \overline{a_R} \end{pmatrix}$
(D) $\begin{pmatrix} m_S & \overline{a_S} \\ \overline{m_R} & \overline{a_R} \end{pmatrix}$					
(E) $\begin{pmatrix} \overline{m_S} & \overline{a_S} \\ m_R & a_R \end{pmatrix}$	$\begin{pmatrix} m_S & \overline{a_S} \\ m_R & \overline{a_R} \end{pmatrix}$	$\begin{pmatrix} m_S & \overline{a_S} \\ m_R & a_R \end{pmatrix}$			
(F) $\begin{pmatrix} m_S & \overline{a_S} \\ m_R & a_R \end{pmatrix}$					$\begin{pmatrix} \overline{m_S} & \overline{a_S} \\ m_R & a_R \end{pmatrix}$

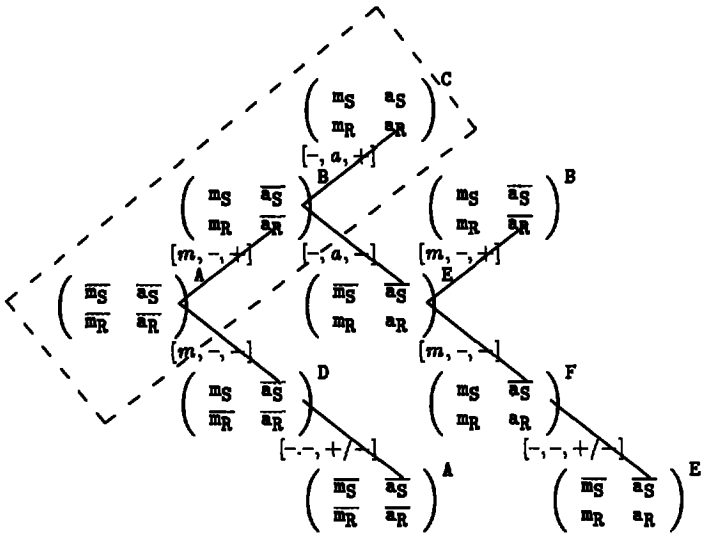

Figure 1: The actual system A and the ideal system I.

other parties.

The actual system A generated by the protocol and the transition function is shown in Figure 1. At a leaf system state, transition continues from the node within the tree which has the same label, except for the state labeled C. The state C repeats indefinitely in the system. The state C is the only state in which both p_m and p_a hold, and it is also the only state in which all the local state components for S and R are true.

The first point of this example is that the exchange of messages and acknowledgments here does not achieve mutual knowledge in its strict sense. Even though our target proposition φ is equivalent to $\mathtt{m}_S \wedge \mathtt{a}_S \wedge \mathtt{m}_R \wedge \mathtt{a}_R$, S and R does not establish mutual knowledge of φ in the entire system. If we take the transitive closure of the union of the a_i-equivalence of S and R, the resulting relation covers all the nodes in the system. Even eventual mutual knowledge does not obtain in A, since there is no local states of R that corresponds to knowledge of φ. But there is still an intuitive appeal in saying that some mutuality is achieved by the exchange in this example, since a variant of this type of exchange is actually employed in computer communications and this example definitely mimics certain important aspects of human use of confirmations in conversations.

The second point of this example is that we can show that the proposition φ becomes defeasible mutual knowledge between S and R in the sense described in the previous section. It is easy to see that we can take as the ideal system I the part of A surrounded by the dashed line in Figure 1. I is preferred over A because the only run in I is weakly knowledge consistent with all the runs in A, and I is obviously the most preferred system. Within I it is easy to see that local conditions $\sigma_S = \mathtt{m}_S \wedge \mathtt{a}_S$ for S and $\sigma_R = \mathtt{m}_R \wedge \mathtt{a}_R$ for R satisfy all the first three conditions in the DMK condition, e.g., knowledge in ideal context, eventual notice of success and simultaneity. Furthermore, fairness requirement on the environment ensures that both σ_S and σ_R will jointly be realized in the system state C eventually. From Proposition 2, we can say that S and R will eventually reach an actual mutual belief on φ. On the other hand, the entire system A satisfies the stronger version of the future error discovery condition. The local state condition $\overline{\mathtt{m}_S}$ serves to indicate S's discovery of errors about sharing of p_m, and the local state condition $\overline{\mathtt{a}_R}$ serves to indicate R's discovery of errors about sharing of p_a. From Proposition 5, we can say that S and R mutually know that an error will become eventual mutual knowledge between S and R.

The final point about this example is that defeasible mutual knowledge is attained through joint actions of

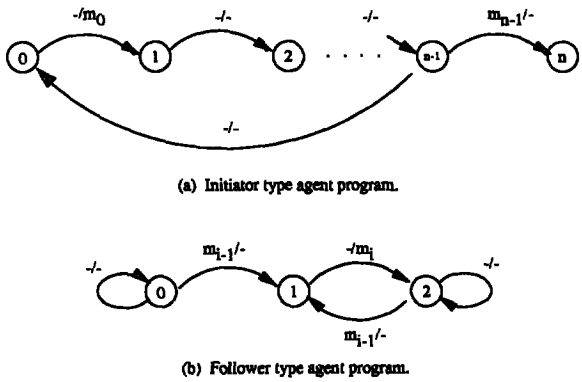

Figure 2: Two agent program types for belief coordination by message passing.

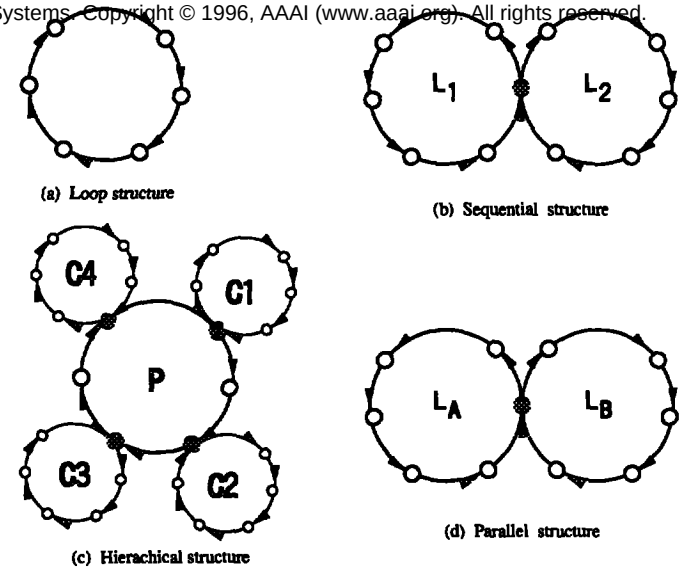

Figure 3: Organizational structures for belief coordination in large groups.

both S and R. As the protocol for S and R together with the transition function shows, not only they act for the end of establishing φ, but both of them act together to recover from errors when they find one. Both of the programs of S and R serve for these dual purposes. Defeasible mutual knowledge was only possible because both parties jointly take part in these achievement/error recovery processes. Confirmations are necessary to realize this joint involvement between speaker and hearer.

Organizational Structure for Belief Coordination

To establish defeasible mutual knowledge among a group of more than two agents, we need to have an organizational structure for message exchange. We will concentrate here on types of organizational structures for message exchanges, and exclude mutual knowledge obtained through copresence in a shared environment, which doesn't require message exchanges.

Generalizing the example in the previous section, we can see that a variety of organizational structures can be generated from two basic types of agent programs which correspond to the sender and the receiver in the example. Two basic types of agent programs are shown in Figure 2. The initiator type program is a generalization of the sender program to n member groups. The follower type program is the receiver program which can also work within larger groups. Nodes and arcs in the figure represent program states and program execution steps. Labels on arcs indicate input/output for the execution steps represented by the arcs.

We can also make complex programs out of these basic programs by combining programs. There are two types of operations for this purpose.

- Composition $p_1 \circ p_2$

  Two initiator type programs p_1 and p_2 can be concatenated by identifying the last node of p_1 with the first node of p_2 (COMP1). The resultant program is of the initiator type. An initiator program p_2 can be embedded into a follower program p_1 at the node 1 position of p_1 (COMP2). The resulting program is of the follower type.

- Product $p_1 \parallel p_2$

  The product $p_1 \parallel p_2$ operates on the direct product of the program states for p_1 and p_2, where p_1 and p_2 work on each components of the product states.

The basic organizational structure formed by these programs are loops, shown in Figure 3(a). The two party exchange in the last section is the simplest example of a loop. We need exactly one initiator in a loop. At least one initiator is necessary to start message exchange processes, while having two or more initiators in a loop doesn't help, since even if there were two or more initiators in a loop, each initiator would have to work as a follower for messages originated by other initiators in order for each initiators to know that their own messages successfully made it through the loop. The initiator reinitiates message sending cycle in case she doesn't get an appropriate success signal within a certain time bound.

We can generate various complex organizational structures out of loops. Three different types of organizational structures are shown in Figure 3(b)–(d). These structures are generated by corresponding complex programs constructed by the operations above.

(i) Sequential structure

The group G consists of several groups $L_1, L_2, \ldots$, each of which forms a loop. All subloops share one agent. The shared agent executes a program formed by composing initiator type programs with `COMP1` operations. Subloops execute sequentially.

(ii) Hierarchical structure

The group G is a union of a mother group P and several other subordinate groups C_j's. Some agents are members of both the mother group P and one of the subordinate groups C_j. Each of these shared agents executes a program formed by composing a follower program and an initiator type program with `COMP2` operation. She forwards a message in P only after she successfully completed the cycle in C_j.

(iii) Parallel structure

The group G consists of several subgroups $A, B, \ldots$, each of which forms a loop. All subloops share one agent. The shared agent executes a program formed by product operations. Subloops work in parallel, and they do not exhibit hierarchical dependencies.

Conclusions

We proposed a solution to the confirmation puzzle by introducing the notion of defeasible mutual knowledge for characterizing informational states of agents actively participating in joint activities. The notion utilizes the duality of context implicit in nonmonotonic reasoning and knowledge-based distributed systems research. We claimed that the notion captures our intuition about mutual understanding attained in our daily communication by separating out mutual knowledge in ideal setting and discovery of errors in actual environment. We formally stated conditions for defeasible mutual knowledge within an epistemic and temporal modal logical framework over distributed systems, and examined their logical consequences. We further demonstrated by an example that confirmations are a necessary part in obtaining defeasible mutual knowledge. We also showed that organizational structures in larger groups for defeasible mutual knowledge can be derived by extending the confirmation-based message exchange structure.

Our notion of defeasible mutual knowledge comes from a view of acting agents, and we believe that the work reported here provides us with a novel and better perspective for analysis of interplay between knowledge and actions in joint activities including everyday conversations.

References

Alchourrón, C. E.; Gärdenfors, P.; and Makinson, D. 1985. On the logic of theory changes: Partial meet contraction and revision functions. *Journal of Symbolic Logic* 50:510–530.

Appelt, D. E., and Konolige, K. 1988. A practical nonmonotonic theory for reasoning about speech act. In *Proceedings of the 26th Annual Meeting of the Association of Computational Linguistics.*

Clark, H. H. 1992. *Arenas of Language Use.* The University of Chicago Press.

Clark, H. H. 1993. Managing problems in speaking. In *International Symposium on Spoken Dialogue*, 181–184. Waseda University.

Cohen, P. R., and Levesque, H. J. 1991. Confirmations and joint action. In *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence*, 951–957.

Cohen, P. R., and Levesque, H. J. 1993. Preliminaries to a collaborative model of dialogue. In *International Symposium on Spoken Dialogue*, 261–266. Waseda University.

Fagin, R.; Halpern, J. Y.; Moses, Y.; and Vardi, M. Y. 1995. *Reasoning about Knowledge.* MIT Press.

Grosz, B. J., and Sidner, C. L. 1990. Plans for discourse. In Cohen, P. R.; Morgan, J.; and Pollack, M. E., eds., *Intentions in Communication.* MIT Press. chapter 20, 417–444.

Halpern, J. Y., and Moses, Y. 1990. Knowledge and common-knowledge in a distributed environment. *Journal of the ACM* 37(3):549–587.

Katagiri, Y. 1996. A distributed system model for actions of situated agents. In Seligman, J., and Westerståhl, D., eds., *Logic, Language and Computation.* Center for the Study of Language and Information, Stanford University. 317–332.

Katsuno, H., and Mendelzon, A. O. 1991. On the difference between updating a knowledge base and revising it. In *Proceedings of the Second International Conference on Principles of Knowledge Representation and Reasoning*, 387–394.

Lin, F., and Shoham, Y. 1992. A logic of knowledge and justified assumptions. *Artificial Intelligence* 57:271–289.

McCarthy, J. 1986. Applications of circumscription to formalizing common-sense knowledge. *Artificial Intelligence* 28:89–116.

Neiger, G. 1988. Knowledge consistency: a useful suspension of disbelief. In Vardi, M. Y., ed., *Proceedings of the Second Conference on Theoretical Aspects of Reasoning about Knowledge*, 295–308. Morgan Kaufmann.

Perrault, C. R. 1990. An application of default logic to speech act theory. In Cohen, P. R.; Morgan, J. L.; and Pollack, M. E., eds., *Intention in Communication.* MIT Press. 105–133.

Shoham, Y. 1986. Chronological ignorance: Time, nonmonotonicity and necessity. In *Proceedings of AAAI*, 389–393.

Shoham, Y. 1988. *Reasoning about Change: Time and Causation from the Standpoint of Artificial Intelligence.* MIT Press.

Thomason, R. H. 1990. Propagating epistemic coordination through mutual defaults I. In Parikh, P., ed., *Proceedings of the Second Conference on Theoretical Aspects of Reasoning about Knowledge*, 29–39. Morgan Kaufmann.