Hindawi Publishing Corporation
Journal of Applied Mathematics
Volume 2013, Article ID 838694, 15 pages
http://dx.doi.org/10.1155/2013/838694

Research Article

Proactive Communicating Process with Asymmetry in Multiagent Systems

Jiafang Wang,[1] Zhiyong Feng,[1] and Chao Xu[2]

[1] *School of Computer Science and Technology, Tianjin University, Tianjin 300072, China*
[2] *School of Computer Software, Tianjin University, Tianjin 300072, China*

Correspondence should be addressed to Chao Xu; chaoxu@tju.edu.cn

Received 6 March 2013; Accepted 29 April 2013

Academic Editor: Xiaoyu Song

This paper presents a formalized communicating process for dealing with information asymmetry between agents. A proactive process can improve the efficiency of dealing with asymmetry by allowing agents to take the initiative of communication in a goal-oriented way. In the process, by reasoning on belief and intention about the world and figuring out the information needed, the agent proactively requests information from another agent when asymmetry exists between them. Considering that agents may take advantage of information asymmetry by hiding information, the process also includes a model based on game theory to restrict the hiding behaviour. The work presented here not only introduces a definition of information asymmetry from cognitive perspective but also proposes a way to deal with it by communication in MAS. In addition, this paper presents some basic ideas on designing proactive mechanisms in cooperation between agents.

1. Introduction

Information asymmetry exists when a party or parties possess greater informational awareness relative to other participating parties, and this information is pertinent to effective participation in a given situation [1]. In Multiagent System (MAS), agents represent entities with different interests, so information asymmetry could bring benefits to some agents in team work and poor results for some other agents. This paper presents a formalized communicating process, where agents can deal with information asymmetry by reasoning on their knowledge about the world and figure out the information needed when facing asymmetry.

To deal with issues caused by information asymmetry, probability and statistical mechanisms are usually employed [2]. Such mechanisms usually relay on history of interaction between agents. But in some situations, like at beginning of the interaction, such historical information is unavailable. From the cognitive view, if an agent can take proactive action to figure out what information is lacking in cooperation, the agent can request that information directly from agents who possess such information, and it can also take strategies to restrict information hiding (even cheating) by considering the context. Then information asymmetry can be solved in a proactive manner.

Proactive behaviour is considered as one of the key characteristics in software agents. Proactivity usually refers to the ability of agent to make conscious decision without being told to [3, 4], which means that agents will take actions to help each other according to some common goals without being instructed. On such willing to help assumption, researches for proactive behaviour usually ignore the issue of information asymmetry, such as share plan or joint intention [5–10]. However, we can treat "eliminating information asymmetry" as a common goal in teamwork and the communication process can then be modelled from the cognitive view.

Researches have also been conducted on proactive behaviour in human organisations including in the area of feedback seeking and issue selling [11–23]. These researches show that proactive communication between people is helpful to resolve the information asymmetry. From the view of information economics [2], researchers also have proposed several models to analyse the problem of how to get optimum contract under information asymmetry. In this paper, we intend to combine the work about proactive behaviour modelling in MAS, proactive communication, and game theory together and then to provide an efficient way of dealing with information asymmetry between agents in teamwork.

The work described here firstly introduces a formalized description about the communication process of dealing with information asymmetry from the cognitive point of view. Secondly, by combining a game theory based model with the communication process, information hiding is restricted according to context. Finally, the work proposed here provides some basic ideas of designing proactive communication process between agents. In a scenario of information asymmetry, the agent that needs information takes the initiative to identify the information and requests information from the agent that owns it. Such a proactive manner can be used to deal with some other problems in communication, such as trust establishment. During trust establishment the trustor can proactively collect information from the trustee, rather than just waiting to observe the behaviour of the trustee or waiting for information from the third party.

2. The Proactive Communicating Process for Dealing with Information Asymmetry

To facilitate the next discussion, a simple scenario of information asymmetry is introduced first. During the development of software, requirements of a customer change over time. Generally speaking, a customer is usually not familiar with the technologies, while the developer is not familiar with business requirements. Thus information asymmetry exists between the customer and the developer. The developer may take the advantage of information asymmetry to refuse a new requirement in order to gain unreasonable benefits.

In this section, the formal description of communication for dealing with information asymmetry in a proactive manner is presented. Information asymmetry and related processes are expressed with mental attitudes of agents. These mental attitudes are described with modal operators like Bel, Int. To/Th, Attempt, Inform, and Request, which are proposed in Joint intention, SharedPlans and the work of proactive information exchange [3–8, 24–26].

Sometimes information asymmetry exists in some scenarios which participants do not even realise. The process presented here focuses on how to deal with information asymmetry in a proactive manner. So we assume that communicating participants realise that information asymmetry exists in their cooperation. In the previous scenario, the customer should consider how to deal with information asymmetry proactively, in order to add the new requirement without paying unreasonable cost. Meanwhile the developer should consider how to deal with request of customer and take advantage of information asymmetry.

2.1. The Definition Information Asymmetry in the Communication Process. First different roles that two agents play in asymmetry are deserved to be discussed. Without special statement, this paper uses R to present the agent that owns information and P to present the agent short of information. Here information asymmetry means that for certain proposition p, there is an agent P that does not believe p being true or false, and meanwhile there is another agent R that does believe p being true or false, or forms such belief by reasoning on its mental attitudes and knowledge base. Suppose that prop (P) and prop (R) are the set of propositions that P and R own in their mental attitudes and knowledge base, respectively, and Rules is the set of rules of two agents, and rules are all written with horn clauses. Then we define Rules (p) as the set of propositions appeared in the rules like

$$p_1 \wedge p_2 \wedge \cdots p_{n-1} \wedge p_n \rightarrow p \text{ or}$$
$$p_1 \wedge p_2 \wedge \cdots p_{n-1} \wedge p_n \wedge p \rightarrow \perp.$$

We assume that in cooperation P needs to form the belief about proposition p based on R's belief on p. Here P is the agent short of information and R is the agent with information. However, R needs to get some information from P to complete the reasoning process of forming belief about p. If R cannot provide all information P needs, but needs P to provide some information to help R get information needed by P, for these parts of information needed by R, there is role exchange between P and R. R then becomes the agent short of information and P becomes the agent with information. Such a process can be described as in Figure 1.

Now we introduce the presentation of information asymmetry in the communication process. First, two participants of communication should be included in the representation of information asymmetry, as well as the role of each participant: who needs information and who provides.

Second, information asymmetry relates certain propositions which form intentions, beliefs, and other mental attitudes of agents. For instance, in the software scenario, the proposition in the representation of information asymmetry should be “the customer intends the developer to implement a new requirement.”

Third, information asymmetry exists under certain context. For instance, in the scenario, if the developer and the customer belong to the same company and the developer is subordinate of the customer, the developer should tell the customer the feasibility of a new requirement. Then it is not necessary to deal with information asymmetry in such situation.

With the previous discussion, information asymmetry can be presented as the following:

$$AsymInfo(P, R, p, inputVar, outputVar,\ P\_Role,$$
$$R\_Role, C_{Asym}).$$

In the definition, P and R are agents involved in information asymmetry; P_Role and R_Role are the roles that two agents take in the asymmetry. For the sake of convenience, we use poor to represent 0, and rich to present 1.

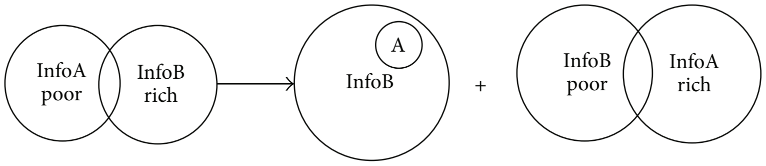
inputVar (inputVar$'$) is the set of input variables of P (R). outputVar (outputVar$'$) is the set of output variables of P (R). C_{Asym} is the context constraints of information asymmetry. The semantics of AsymInfo operator is illustrated with the following axiom.

As for some proposition p:

$p_1 \wedge p_2 \wedge \cdots \wedge p_i \wedge \cdots \wedge p_j \wedge \cdots \wedge p_n \rightarrow p$

P does not have true value for each p_i except p_j

R has true value for each p_i except p_j, which can be gotten from P



Suppose that P needs true value of p, making it lacking information in asymmetry
but before R gives P the information about each p_i, R needs true value of p_j first
then P and R exchange roles in asymmetry

Figure 1: Role exchange between two agents.

Axiom 1.

$Bel(P, AsymInfo(P, R, p, inputVar, outputVar, poor, rich, C_{Asym}), t) \Rightarrow MB(\{P, R\}, \exists p' \in Rules(p) \neg Bel(P, Bel(R, p', t), t) \wedge Bel(R, Bel(R, p', t), t), t) \vee MB(\{P, R\}, \exists p' \in Rules(p) \neg Bel(P, Bel(R, \neg p', t), t) \wedge Bel(R, Bel(R, \neg p', t), t), t)$.

$Bel(R, AsymInfo(P, R, p, inputVar', outputVar', poor, rich, C_{Asym}), t) \Rightarrow MB(\{P, R\}, \exists p' \in Rules(p) \neg Bel(P, Bel(R, p', t), t) \wedge Bel(R, Bel(R, p', t), t), t)$
$\vee MB(\{P, R\}, \exists p' \in Rules(p) \neg Bel(P, Bel(R, \neg p', t), t)$
$\wedge Bel(R, Bel(R, \neg p', t), t), t)$.

The axiom says that at current time t, if P (or R) believes that information asymmetry about proposition p exists between it and R (or P), there must exist some proposition p' in Rules(p) (of each agent's own) that R believes being true but P does not believe that R believes p' being true, or R believes being false but P does not believe that R believes p' being false. Table 1 lists the references about the Bel and MB operators. With this axiom, it can be assumed that R has ability to provide P with information related with asymmetry.

In the following section, the formal description of communication process is presented in detail, and many problems will be discussed. The notations to be used are listed in Table 1.

2.2. The Communication Process from the General View. In modern control theory, state space equation is a common tool to model and analyze dynamic characteristics of systems. A state space equation can be expressed as

$$\begin{aligned} X(t+1) &= F(X(t), U(t), t) \quad \text{state equation}, \\ Y(t) &= G(X(t), U(t), t) \quad \text{output equation}. \end{aligned} \tag{1}$$

The equation is composed of following components: a set of state variables to describe behaviours of system and a set of input variables and a set of output variables. These variables make up the state equation and the output equation. Consider that agents communicate with each other to deal with information asymmetry. Each agent has its own internal states composed by its mental attitudes. An agent sends some variables to another agent and requests for answers. These variables indicate what information P needs. The target agent receives the variables and finally gives out answers with reasoning process. This situation is similar to state space equation discussed in control theory.

Based on the previous analysis, the process of dealing with information asymmetry can be described with following state space equations

$$\begin{aligned} &\text{for the agent short of information:} \\ &\quad X_{\text{poor}}(t+1) = F\left(X_{\text{poor}}(t), U_{\text{poor}}(t), t\right), \\ &\quad Y_{\text{poor}}(t) = G\left(X_{\text{poor}}(t), U_{\text{poor}}(t), t\right); \\ &\text{for the agent with information:} \\ &\quad X_{\text{rich}}(t+1) = F'\left(X_{\text{rich}}(t), Y_{\text{poor}}(t), t\right), \\ &\quad Y_{\text{rich}}(t) = \alpha \mathrm{G}'\left(X_{\text{rich}}(\mathrm{t}), Y_{\text{poor}}(t), t\right), \\ &\quad U_{\text{poor}}(t+1) \subseteq \alpha Y_{\text{rich}}(t). \end{aligned} \tag{2}$$

Here, poor and rich in subscript of the equation represent two agents involved in asymmetry, with different roles. $X(t)$ is a set which includes mental attitudes of an agent, like beliefs and intentions. $U(t)$ and $Y(t)$ are sets of input and output variables, which correspond to inputVar and outputVar in the operator of AsymInfo. Here $X_{\text{poor}}(t+1)$ and $X_{\text{rich}}(t+1)$ mean that the states of agent are updated for the next round of communication. $U_{\text{poor}}(t+1)$ means that after poor gets the output variables $\alpha Y_{\text{rich}}(t+1)$ from rich, input variables of poor are updated. In the equation, F, G, F', and G' correspond to the reasoning process as follows.

F: For the agent P that is short of information, F represents the establishment of reasoning process for dealing with information asymmetry and identifying input variables in $U_{\text{poor}}(t)$.

Table 1: Summary of notations.

Notation	Meaning
$\mathrm{prop}(P)$	$\mathrm{prop}(P)$ stands for proposition that agent A owns in its mental attitudes or knowledge base
Rule	The set of rules of agent
$\mathrm{Rules}(p)$	$\{p_i \mid 1 \le i \le n, p_1 \wedge p_2 \wedge \cdots \wedge p_i \wedge \cdots \wedge p_n \rightarrow p \in \mathrm{Rule}\}$
$\mathrm{Bel}(A, p, t)$	Agent A believes that proposition p holds at time t [4, 6, 26, 27]
$\mathrm{MB}(\{A, B\}, p, t)$	Both agents A and B believe that proposition p holds at time t [4, 6, 26, 27]
$\mathrm{Int.To}(A, \alpha, t, t_\alpha, C_\alpha)$	At time t, agent A intends to do α at time t_α in the context C_α [3, 6]
$\mathrm{Int.Th}(A, p, t, t', C_p)$	Agent A at time t intends that p holds at t' under the intentional context C_p [3, 6]
$\mathrm{Pot.Int.To}(A, \alpha, t, t_\alpha, C_\alpha)$	Agent A has a potential intention to do α. Compare with Int.to, potential intention means an agent is not fully committed, and A needs to find out a plan to do α before changing potential intention into intention [3, 6]
$\mathrm{Pot.Int.Th}(A, p, t, t', C_p)$	Agent A at time t has a potential intention that p holds at t' under the intentional context C_p [3, 6]
$\mathrm{CBA}(A, \alpha, R_\alpha, t_\alpha, \Theta_\alpha)$	Agent A is able to bring about single-agent action α at t_α under constraints Θ_α by following recipe R_α. Here a recipe is composed of action decomposition and constraints [3]
$\mathrm{Do}(A, \alpha, t, C_\alpha)$	Agent A performs action α at time under constraints Θ_α. [3]
$\mathrm{Attempt}(A, \varepsilon, P, Q, C_n, t, t_1)$	Agent A has only a limited commitment (potential intention) to the ultimate goal P by executing ε, while having a full-fledged intention to achieve Q [3]
$\mathrm{Request}(A, B, \varepsilon, \alpha, t, t_a, \Theta_\alpha)$	Agent A's attempt to make both A and B believe that A intends that B commits to performing the action ε [3]
$\mathrm{Inform}(A, B, \varepsilon, p, t, t_a)$	Agent A's attempt to establish a mutual belief with agent B about the A's goal to let the addressee know p is stand [3]
$\mathrm{constr}(C)$	Constraints component of context C [3]
$\mathrm{post}(\varepsilon)$	A conjunction of propositions that describe the effects of ε [3]
CONF	Conflictions exist between actions or propositions [3]

G: For the agent P that is short of information, G represents the process of identifying output variables in $Y_{\text{poor}}(t)$ with input variables and internal state.

F': For the agent R that is with information, F' represents the process of updating R's internal states after R receives input variables from P

G': For the agent R that is with information, G' represents the establishment of reasoning process for finding true values for input variables in $U_{\text{poor}}(t)$

α: In the process of G', α represents the process of hiding information in the $Y_{\text{rich}}(t)$ (such a process can be included in the process of G'. Here we use a separate operator in order to emphasize such a process).

Equation (2) shows that in communication process, input and output variables define what information needs to be exchanged between two agents and they are closely related to asymmetry between two agents. Definition 1 gives out the formal definition of input and output variables of both agents.

Definition 1. For agent P that is short of information, input variable is defined as

$$\text{for all } \mathrm{var} \in \{(\mathit{prop}_i, \mathit{true\_value}_i) \mid 1 \le i \le n, \mathit{prop}_i \in \mathit{prop}(P), \mathit{true\_value} = \mathit{true} \mid \mathit{false} \mid \mathit{unknown}\},$$

$$\mathit{true\_value}_i = \mathit{unknown}, \text{ if and only if } \mathit{Bel}(P, Bt) = \mathit{false} \text{ and } \mathit{Bel}(P, \mathit{Bel}(R, \neg \mathit{prop}_i, t), t) = \mathit{false};$$

for agent P that is short of information, output variable is defined as

$$\text{for all } \mathrm{var} \in \{(\mathit{prop}_i, \mathit{true\_value}_i) \mid 1 \le i \le n, \mathit{prop}_i \in \mathit{prop}(P), \mathit{true\_value} = \mathit{true} \mid \mathit{false} \mid \mathit{unknown}\},$$

$$\mathit{true\_value}_i = \mathit{true}, \text{ if and only if } \mathit{Bel}(P, \mathit{prop}_i, t) = \mathit{true},$$

$$\mathit{true\_value}_i = \mathit{false}, \text{ if and only if } \mathit{Bel}(P, \neg \mathit{prop}_i, t) = \mathit{true};$$

for agent R that is with information, output variable is defined as

$$\text{for all } \mathrm{var} \in \{(\mathit{prop}_i, \mathit{true\_value}_i) \mid 1 \le i \le n, \mathit{prop}_i \in \mathit{prop}(R), \mathit{true\_value} = \mathit{true} \mid \mathit{false} \mid \mathit{unknown}\},$$

$$\mathit{true\_value}_i = \mathit{true}, \text{ if and only if } \mathit{Bel}(R, \mathit{prop}_i, t) = \mathit{true},$$

$$\mathit{true\_value}_i = \mathit{false}, \text{ if and only if } \mathit{Bel}(R, \neg \mathit{prop}_i, t) = \mathit{true},$$

$$\mathit{true\_value}_i = \mathit{unknown}, \text{ if and only if } \mathit{Bel}(R, \mathit{prop}_i, t) = \mathit{false} \text{ and } \mathit{Bel}(R, \neg \mathit{prop}_i, t) = \mathit{false}.$$

Then input variables of P are related to what information P wants from R and are defined with P's belief about R's belief about a proposition prop. Input variable (prop, unknown) means that P does not believe that R believes prop is true and P also does not believe R believes prop is false. Output variables of P are some beliefs of its own that P wants to tell R in communication. These variables may help R to get information that P needs and they are helpful to avoid R to request them from P again.

As for input variables of R, they are elements in union set of input and output variables sending by P. For output variables of R, they are beliefs of R which are requested by P as what P defined as input variables. Output variable (prop, true) of R means that R believes prop is true, and output variable (prop, false) means that R believes prop is false. Output variable (prop, unknown) means that R does not believe prop is true or false.

The whole process is presented by Figure 2. First, agent short of information (P) finds out that information asymmetry would bring negative influence on cooperation between itself and agent with information (R). First P identifies some input variables related to information asymmetry. In the process of identifying input variables, a process of reasoning is established (the tree in second part of the Figure 2). The reasoning process and related mental attitudes of agent correspond to F and $X_{\text{poor}}(t)$ in (2). At the same time, the output variables ($Y_{\text{poor}}(t)$) with initial value are also identified (with input variables $U_{\text{poor}}(t)$ included). Then output variables are sent to R to begin a communicating process to deal with information asymmetry (Rule 2 in Figure 2).

After output variables received, R begins to construct its own state space. The input and the output variables of R are constrained by the output and input variables of P (as shown in (2)). Mental attitudes of R are updated with input variables at first. In order to get the true_values of variables in $U_{\text{poor}}(t)$, R will start a reasoning process (the tree in third part of the Figure 3). After true_values of variables are gotten, R puts these variables into the set of output variables and hides information as needed. Then output variables of R are sent to P (CommuResponse in the Figure 2), and P analyzes whether true values of variables are hidden and updates its mental attitudes.

The information hiding during communication is constrained by the method based on game theory. R can have a set of strategies by defining to hide a set of variable in its output. After P gets inputVar from R (output variables of R), it can have a set of strategies by judging each variable as hidden or not. P can also define payoffs for strategies. If proper mechanisms for the game are designed, P and R can get equilibrium about their game on information hiding.

2.3. The Communication Process for Dealing with Information Asymmetry in Detail

2.3.1. Start of the Process. To facilitate introduction in following section, formal definitions of modal operators Attempt, Inform, and Request are listed as follows [3]. The semantics of Inform and Request are given by choosing appropriate formulas to substitute in the definition of Attempt [3, 28]. Related operators and predications are listed in Table 1. Here we use "=" to present "defined as."

Definition 2. $Attempt(P, \varepsilon, U, V, C_n, t, t_1) = \phi?: \varepsilon$, *where*

$$\phi = [\neg Bel(P, U, t) \land Pot.Int.Th(P, U, t, t_1, C_n) \land Int.Th(P, V, t, t_1, \neg Bel(P, U, t) \land C_n) \land Int.To(P, \varepsilon, t, t, \psi)], \text{ where}$$

$$\psi = Bel(P, post(\varepsilon) \Rightarrow V, t) \land Pot.Int.Th(P, U, t, t_1, C_n).$$

$Inform(P, R, \varepsilon, p, t, t_\alpha) = (t < t_\alpha)?;\ Attempt(P, \varepsilon, U, V, C_p, t, t_\alpha)$, *where*

$$U = MB(\{P, R\}, p, t_\alpha),$$

$$V = \exists t''\ (t \le t'' < t_\alpha) \land MB(\{P, R\}, \psi, t''),$$

$$C_p = Bel(P, p, t) \land Bel(P, \neg Bel(R, p, t), t), \text{ where}$$

$$\psi = \exists t_b\ (t'' \le t_b < t_\alpha) \land$$
$$Int.Th(P, Bel(R, Bel(P, p, t), t_b), t, t_b, C_p).$$

$Request(P, R, \varepsilon, \alpha, t, t_\alpha, \Theta_\alpha) = (t < t_\alpha)?;\ Attepmt(P, \varepsilon, U, V, C_p, t, t_\alpha)$, *where*

$$U = Do(R, \alpha, t_\alpha, \Theta_\alpha),$$

$$V = \exists t''\ (t \le t'' < t_\alpha) \land MB(\{P, R\}, \psi, t''),$$

$$C_p = Bel(P, \exists R_\alpha CBA(R, \alpha, R_\alpha, t_\alpha, \Theta_\alpha), t) \land Int.Th(P, Do(R, \alpha, t_\alpha, \Theta_\alpha), t, t_\alpha, \Theta_\alpha), \text{ where}$$

$$\psi = \exists t_b\ (t'' \le t_b < t_\alpha) \land$$
$$Int.Th(P, Int.To(R, \alpha, t_b, t_\alpha, C_p \land Helpful(R)), t, t_b, C_p).$$

In the definition of $Attempt(P, \varepsilon, U, V, C_n, t, t_1)$, U represents some ultimate goal that may or may not be achieved by the attempt and V represents what it takes to make an honest effort. The definition of $Inform(P, R, \varepsilon, p, t, t_\alpha)$ says that at current time t, P wants R to believe that p is true with event ε before time t_α. The definition of $Request(P, R, \varepsilon, \alpha, t, t_\alpha, \Theta_\alpha)$ says that at the current time t, P wants R to execute α with event ε before time t_α.

With these definitions, the process of dealing with information asymmetry will be discussed in detail. The first question is how to start and who will initiate the process of communication. According to Axiom 1 in Section 2.1, two agents believe that information asymmetry about proposition p exists between them. If both of them have reached an agreement on true value of p, it may be unnecessary to deal with asymmetry between them. They just need to act as what they both agree. But when some conflicts about p appear as Axiom 1 suggests that inconsistency between both agents' belief exists, both agents should consider finding out the relation between conflicts and information asymmetry. And the process of dealing asymmetry should be taken into consideration.

Another question is who will initiate the process. It seems that both the agent short of information and the agent with information may be aware of conflicts caused by asymmetry,

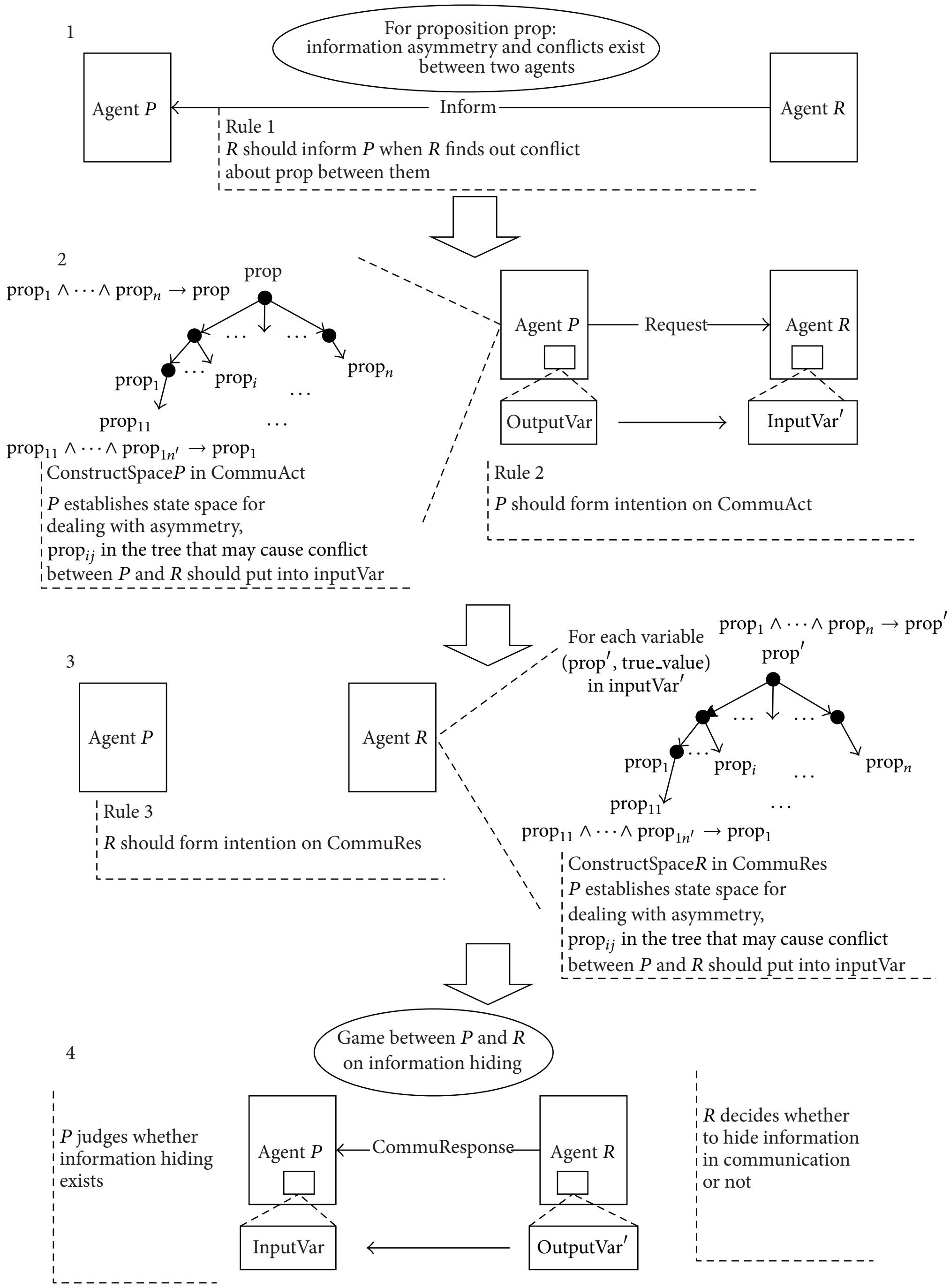
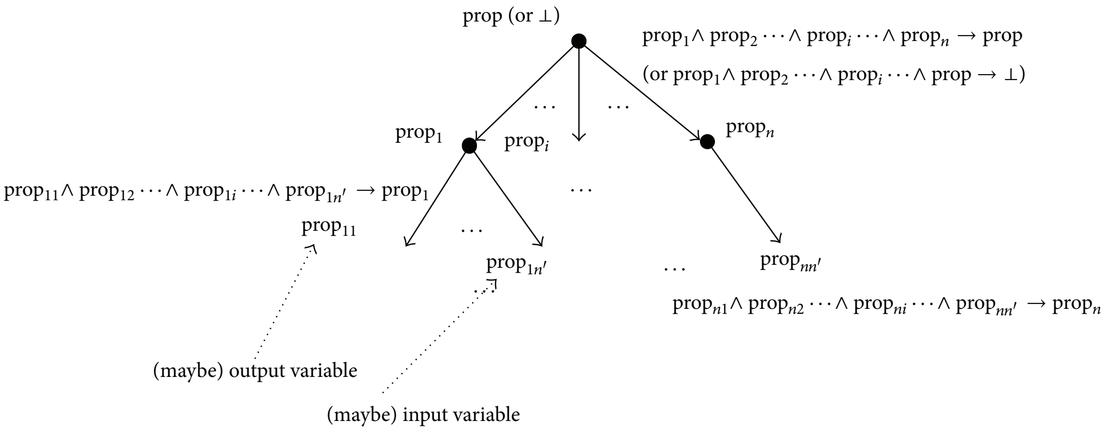

1
For proposition prop: information asymmetry and conflicts exist between two agents
Agent P
Inform
Agent R
Rule 1
R should inform P when R finds out conflict about prop between them
2
$\text{prop}_1 \wedge \cdots \wedge \text{prop}_n \rightarrow \text{prop}$
prop
prop_1
prop_i
prop_n
prop_{11}
$\text{prop}_{11} \wedge \cdots \wedge \text{prop}_{1n'} \rightarrow \text{prop}_1$
ConstructSpace$P$ in CommuAct
P establishes state space for dealing with asymmetry, prop_{ij} in the tree that may cause conflict between P and R should put into inputVar
Agent P
Request
Agent R
OutputVar
InputVar′
Rule 2
P should form intention on CommuAct
3
Agent P
Agent R
Rule 3
R should form intention on CommuRes
For each variable (prop′, true_value) in inputVar′
$\text{prop}_1 \wedge \cdots \wedge \text{prop}_n \rightarrow \text{prop}'$
prop′
prop_1
prop_i
prop_n
prop_{11}
$\text{prop}_{11} \wedge \cdots \wedge \text{prop}_{1n'} \rightarrow \text{prop}_1$
ConstructSpace$R$ in CommuRes
P establishes state space for dealing with asymmetry, prop_{ij} in the tree that may cause conflict between P and R should put into inputVar
4
Game between P and R on information hiding
P judges whether information hiding exists
Agent P
CommuResponse
Agent R
InputVar
OutputVar′
R decides whether to hide information in communication or not

Figure 2: The framework on dealing with information asymmetry in a proactive manner.

and both of them can start the process of dealing. But the agent who initiates the process should identify input variables, which constrains the choices of output variables of another agent and the reasoning process of both agents. This paper assumes that agent short of information (P) should initiate the process, as it can find out propositions whose true values it is not aware of in reasoning process. These propositions are candidates of input variables. When information asymmetry on p exists between R and P, we define a rule to make sure that the agent with information (R) will inform P about a contradiction of beliefs or intentions on proposition p between it and P.

Rule 1.

$$Bel(R, Int.Th(P, Bel(R, p, t_p), t, t_p, C_p), t) \wedge Bel(R, \neg Bel(R, p, t_p), t) \wedge Bel(R, AsymInfo(P, R, p, poor, rich, inputVar, outputVar, C_p), t) \Rightarrow Int.\ To(R, Inform(R, P, \varepsilon, Bel(R, \neg Bel(R, p, t_p), t), t, t_{inform}),$$



prop (or ⊥)
prop$_1$∧ prop$_2$ ⋯ ∧ prop$_i$ ⋯ ∧ prop$_n$ → prop
(or prop$_1$∧ prop$_2$ ⋯ ∧ prop$_i$ ⋯ ∧ prop → ⊥)
prop$_1$
prop$_i$
prop$_n$
prop$_{11}$∧ prop$_{12}$ ⋯ ∧ prop$_{1i}$ ⋯ ∧ prop$_{1n'}$ → prop$_1$
prop$_{11}$
prop$_{1n'}$
prop$_{nn'}$
prop$_{n1}$∧ prop$_{n2}$ ⋯ ∧ prop$_{ni}$ ⋯ ∧ prop$_{nn'}$ → prop$_n$
(maybe) output variable
(maybe) input variable

Figure 3: Reasoning tree in state space establishment.

$$t, t_{inform}, C_p)\text{—}part\ 1;$$

$$Bel(R, \exists t' < t_p\ Int.Th(P, Int.To(R, p, t', t_p, C'_p), t, t',$$

$$C_p), t) \wedge Bel(R, \forall t'' < t \neg Int.To(R, p, t'', t_p, C'_p), t) \wedge$$

$$Bel(R, AsymInfo(P, R, p, poor, rich, C_p), t) \Rightarrow$$

$$Int.To(R, Inform(R, P, \varepsilon, Bel(R, \forall t'' < t_p \neg Int.To$$

$$(R, p, t'', t_p, C'_p), t), t, t_{inform}), t, t_{inform}, C')\text{—}part\ 2.$$

Suppose that at time t information asymmetry about proposition p exists between P and R. Rule 1 includes two situations. First, at time t if R believes that P intends that R will believe p is true at time t_p ($t_p > t$), but at time t, R believes that it won't believe p is true at time t_p, R should intend to inform P that R will not believe p is true at time t_p. Second, if R believes P intends that R will intend to do p at some time before t_p under context C'_p, but R believes that it won't intends to do p under context C'_p at any time before t_p, R should intend to inform P such belief. Inform should be finished before t_{Inform}. t_{Inform} is certain time after t, and it can be defined by R according to the requirement of concrete scenario.

In Rule 1 we use beliefs about $Int.Th(P, Bel(R, p, t_p), t, t_p, C_p)$ and $\exists t' < t_p$, $Int.Th(P, Int.To(R, p, t', t_p, C'_p), t, t', C_p)$, because such beliefs can be gotten when P informs R about its intentions, but it is hard for P to get beliefs like $Bel(R, Bel(P, p, t), t')$ directly.

As described in part 1 of Figure 2, if R uses Rule 1 to inform P about some conflicts, P can be aware of conflicts between P and R. Here assumption 1 is defined to show how P forms belief of conflict between it and R.

Assumption 3. P believes that there exist conflicts between its intention that R should perform some action p or intend some proposition p to hold and R's unwillingness of performing p or intending p be hold as

$$Bel(P, prop1, t) \ \wedge\ Bel(P, prop2, t) \ \Rightarrow\ Bel(P, CONF(prop1, prop2, t, t, C_{prop1}, C_{prop2}), t),$$

$$Bel(P, prop3, t) \ \wedge\ Bel(P, prop4, t) \ \Rightarrow\ Bel(P, CONF(prop3, prop4, t, t, C_{prop3}, C_{prop4}), t),$$

where

$$prop1 = Int.Th(P, Bel(R, p, t_p), t, t_p, C_p),$$

$$prop2 = (\exists t''\ (t'' \le t_{inform}) \wedge Int.Th(R, Bel(P, Bel(R, \neg Bel(R, p, t_p), t), t''), t, t'', C_p)),$$

$$prop3 = \exists t' < t_p, Int.Th(P, Int.To(R, p, t', t_p, C'_p), t, t', C_p),$$

$$prop4 = (\exists t''\ (t'' \le t_{inform}) \wedge Int.Th(R, Bel(P, Bel(R, \forall t''' < t_p \neg Int.To(R, p, t''', t_p, C'_p), t), t''), t, t'', C_p)).$$

prop1 stands for that P intends proposition "R believes p being hold at time t_p" being hold. prop2 stands for that R intends proposition "at some time t" before t_{inform}, and P believes that R does not believe that p holds at time t_p". prop3 stands for that P intends that "R intends to do p at time t_p" being hold. prop4 stands for that R intends proposition "at some time t" before t_{inform}, and P believes that R believes that R will not intend to do p at any time before t_p" being hold. Meta-predicate $CONF(\alpha, \beta, T_\alpha, T_\beta, \Theta_\alpha, \Theta_\beta)$ represents situations in which actions or propositions conflict with each other [6]. Function constr (C_α) denotes the constraints components of the context C_α [6].

Theorem 4. *In Rule 1, if R's belief about P's intention is consistent with P's intention, successful performance of Inform in Rule 1 will make P believe that conflicts exist between it and R.*

Proof. (1) As R's belief about P's intention is consistent with P's intention, in Rule 1, if

$$\begin{aligned}&Bel(R, Int.Th(P, Bel(R, p, t_p), t, t_p, C_p), t) \text{ or}\\&Bel(R, \exists t' < t_p,\ Int.Th(P, Int.To(R, p, t', t_p,\\&C'_p), t, t', C_p), t) \text{ holds at current time } t,\ P\\&\text{also has beliefs as follows:}\end{aligned}$$

$$\begin{aligned}&Bel(P, Int.Th(P, Bel(R, p, t_p), t, t_p, C_p), t) \text{ or}\\&Bel(P, \exists t' < t_p,\ Int.Th(P, Int.To(R, p, t', t_p,\\&C'_p), t, t', C_p), t).\end{aligned}$$

Then at some time t',

$$\begin{aligned}&prop1 = Int.Th(P, Bel(R, p, t_p), t, t_p, C_p) \text{ and}\\&prop2 = \exists t' < t_p,\ Int.Th(P, Int.To(R, p, t', t_p, C'_p),\\&t, t', C_p) \text{ hold (time } t' \text{ is decided by } P \text{ as necessary).}\end{aligned}$$

(2) With Rule 1, R forms the intention of Inform. If the performance of Inform is successful, with Definition 2 there exists some time t' that P and R reach a mutual belief like

$$\begin{aligned}&prop3 = MB(\{P, R\}, (\exists t'' \ (t' \le t'' \le t_{inform}) \wedge\\&Int.Th(R, Bel(P, Bel(R, \neg Bel(R, p, t_p), t), t''), t, t'',\\&C_p)), t') \text{ or}\\&prop4 = MB(\{P, R\}, (\exists t'' \ (t' \le t'' \le t_{inform}) \wedge\\&Int.Th(R, Bel(P, Bel(R, \forall t''' < t_p \neg Int.To(R, p, t''', t_p,\\&C'_p), t), t''), t, t'', C_p)), t').\end{aligned}$$

With Assumption 3, P gets a conflict between prop1 and prop3, or prop2 and prop4. □

After P is aware of conflicts, P should consider initiating a process of dealing with information asymmetry in the following situations, as we define in Rule 2.

Rule 2.

$$\begin{aligned}&Bel(P, Int.Th(R, Bel(P, q, t_{Bel}), t_{Int}, t_{Bel},\\&C_{Int\_Th})), t) \wedge\\&\exists p \in prop(P),\ Bel(P, p, t) \wedge Bel(P, CONF(Bel(P,\\&p, t),\\&Bel(P, q, t_{Bel}), t, t_{Bel}, constr(C_p), constr(C_q)), t) \wedge\\&Bel(P, AsymInfo(P, R, q, poor, rich, inputVar,\\&outputVar, C_{Asym}), t)\\&\Rightarrow Pot.Int.To(P, CommuAct(P, R, q, inputVar,\\&outputVar, t_{exeP}, t_{comP}, t_{comR}, t_{res}, C_{comm}), t, t_{exeP},\\&C_{comm}),\\&where\ t < t_{exeP} < t_{comP} < t_{comR} < t_{res} < t_{Bel};\end{aligned}$$

$$\begin{aligned}&Bel(P, Int.Th(R, Bel(P, q, t_{Bel}), t_{Int}, t_{Bel}, C_{Int\_Th})), t) \wedge\\&\exists p \in prop(P),\ Int.Tx(P, p, t, t_p, C_p) \wedge Bel(P,\\&CONF(Int.Tx(P, p, t, t_p, C_p), Bel(P, q, t_{Bel}),\\&t, t_{Bel}, constr(C_p), constr(C_q)), t) \wedge Bel(P,\\&AsymInfo(P, R, q, poor, rich, inputVar, outputVar,\\&C_{Asym}), t) \Rightarrow Pot.Int.To(P, CommuAct(P, R, q,\\&inputVar, outputVar, t_{exeP}, t_{comP}, t_{comR}, t_{res},\\&C_{comm}), t, t_{exeP}, C_{comm}),\\&where\ t < t_{exeP} < t_{comP} < t_{comR} < t_{res} < t_{Bel};\end{aligned}$$

$$\begin{aligned}&Bel(P, Int.Th(R, Int.To(P, q, t_{Int\_To}, t_{Int\_To\_Fin},\\&C_{Int\_To}), t_{Int}, t_{Int\_To}, C_{Int\_Th}), t) \wedge \exists p \in prop(P)\\&Bel(P, p, t) \wedge Bel(P, CONF(Bel(P, p, t),\\&Int.To(P, q, t_{Int\_To}, t_{Int\_To\_Fin},\\&C_{Int\_To}), t, t_{Int\_To}, constr(C_p), constr(C_q)), t) \wedge\\&Bel(P, AsymInfo(P, R, q, poor, rich, inputVar,\\&outputVar, C_{Asym}), t) \Rightarrow Pot.Int.To(P,\\&CommuAct(P, R, q, inputVar, outputVar,\\&t_{exeP}, t_{comP}, t_{comR}, t_{res}, C_{comm}), t, t_{exeP}, C_{comm}),\\&where\ t < t_{exeP} < t_{comP} < t_{comR} < t_{res} < t_{Int\_To};\end{aligned}$$

$$\begin{aligned}&Bel(P, Int.Th(R, Int.To(P, q, t_{Int\_To}, t_{Int\_To\_Fin},\\&C_{Int\_To}), t_{Int}, t_{Int\_To}, C_{Int\_Th})), t) \wedge \exists p \in prop(P),\\&Int.Tx(P, p, t, t_q, C_q) \wedge Bel(P, CONF(Int.Tx(P, p, t,\\&t_q, C_q),\\&Int.To(P, q, t_{Int\_To}, t_{Int\_To\_Fin}, C_{Int\_To}), t, t_{Int\_To},\\&constr(C_p), constr(C_q)), t) \wedge Bel(P, AsymInfo(P, R, q,\\&poor, rich, inputVar, outputVar, C_{Asym}), t) \Rightarrow\\&Pot.Int.To(P, CommuAct(P, R, q, inputVar,\\&outputVar, t_{exeP}, t_{comP}, t_{comR}, t_{res}, C_{comm}),\\&t, t_{exeP}, C_{comm}),\\&where\ t < t_{exeP} < t_{comP} < t_{comR} < t_{res}.\end{aligned}$$

Here, Int.Tx stands for Int.Th or Int.To. Rule 2 says that at time t, suppose that P believes that at time t_{Int}, R intends that P will believe that some proposition q is true or P will intend to do q. At this time P also believes that some proposition p is true or it will intend to do p. However, conflicts between p and q, as well as information asymmetry, exist between P and R in P's opinion. Then at time t, P should form potential intention to execute CommuAct at time t_{exeP} under context C_{comm}. C_{comm} includes C_{Asym} and P's belief about conflict.

Here we use potential intention because P should reconcile intention on CommuAct with other intentions that P has already adopted. We define CommuAct as follows.

Definition 5.

$$\begin{aligned}&CommuAct(P, R, p, inputVar,\\&outputVar, t, t_{comP}, t_{comR}, t_{res}, C_{comm})\\&= ((t < t_{comP})?;\\&ConstructSpaceP(p, inputVar, outputVar, t))?;\\&Request(P, R, \varepsilon, CommuResponse(R, P, p, inputVar',\end{aligned}$$

$$\mathit{outputVar}', t_{comR}, t_{res}, \Theta_{CommuRes}), t, t_{comR}, \Theta_{CommuRes}), \text{ where } \mathit{inputVar} \subseteq \mathit{outputVar}' \wedge \mathit{outputVar} \subseteq \mathit{inputVar}' \in \Theta_{CommuRes}.$$

At time t, execution of CommuAct means that before time $t_{\mathrm{com}P}$, P executes action ConstructSpaceP. If ConstructSpaceP is done successfully, P requests R to execute CommuResponse at $t_{\mathrm{com}R}$ and makes response before time t_{res}. ConstructSpaceP is responsible for establishing reasoning process for dealing with information asymmetry. This definition will be discussed in detail later.

2.3.2. Dealing Process. In Definition 5, CommuAct is executed at time t defined as an act like this: before $t_{\mathrm{com}P}$ agent P executes ConstructSpaceP, which constructs state space according to the proposition p and identifies input and output variables. If ConstructSpaceP is executed successfully, P requests R to execute CommuResponse at time $t_{\mathrm{com}R}$, which should be finished before t_{res}. inputVar is a set of input variables of P, and outputVar is a set of output variables of P. outputVar will be sent to R with action ε. inputVar$'$ and outputVar$'$ are input and output variables of R. $\mathit{inputVar} \subseteq \mathit{outputVar}' \wedge \mathit{outputVar} \subseteq \mathit{inputVar}'$ says that when P sends outputVar to R with action Request, inputVar$'$ should include outputVar; when R sends outputVar$'$ to P with action CommuResponse, outputVar$'$ should include input variables. Request and CommuResponse can be implemented with Agent Communication Language.

First input and output variables need further discussion. As what we have discussed earlier, R will hide information when it uses CommuResponse to send outputVar$'$ to P. Then R gets true value of var $= (\mathit{prop}, \mathit{truth\_value}) \in \mathit{outputVar}'$, true or false, and information hiding can be defined as the following:

$$\mathit{truth\_value} = \mathit{unknown}, \text{ if and only if } \mathit{Bel}(B, \mathit{prop}, t) = \mathit{true},$$
$$\mathit{truth\_value} = \mathit{unknown}, \text{ if and only if } \mathit{Bel}(B, \neg \mathit{prop}, t) = \mathit{true}.$$

Here we do not consider cheating between two agents, and we define cheating as follows:

$$\mathit{truth\_value} = \mathit{false}, \text{ if and only if } \mathit{Bel}(B, \mathit{prop}, t) = \mathit{true},$$
$$\mathit{truth\_value} = \mathit{true}, \text{ if and only if } \mathit{Bel}(B, \neg \mathit{prop}, t) = \mathit{true}.$$

So we define an assumption about information hiding as follows.

Assumption 6.

$$\textit{For all } \mathrm{var} = (p, \mathit{true\_value}) \in \mathit{outputVar},$$
$$\mathit{true\_value} = \mathit{true} \wedge \forall q \in \mathit{prop}(R),$$
$$\mathit{Bel}(R, q, t) \wedge (\mathit{Bel}(R, \neg \mathit{CONF}(\mathit{Bel}(R, q, t), \mathit{Bel}(P, p, t), t, t, \mathit{constr}(C_q), \mathit{constr}(C_p)), t) \vee \mathit{Bel}(R, \neg \mathit{CONF}(\mathit{Int.Tx}(R, q, t, t_q, C_q), \mathit{Bel}(P, p, t), t, t, \mathit{constr}(C_q), \mathit{constr}(C_p)), t)) \Rightarrow \mathit{Bel}(R, \mathit{Bel}(P, p, t), t);$$
$$\mathit{true\_value} = \mathit{false} \wedge \forall q \in \mathit{prop}(R),\ \mathit{Bel}(R, q, t) \wedge (\mathit{Bel}(R, \neg \mathit{CONF}(\mathit{Bel}(P, \neg p, t), \mathit{Bel}(R, q, t_q), t, t, \mathit{constr}(C_p), \mathit{constr}(C_q)), t) \vee \mathit{Bel}(R, \neg \mathit{CONF}(\mathit{Bel}(P, \neg p, t), \mathit{Int.\ Tx}(R, q, t, t_q, C_q), t, t, \mathit{constr}(C_p), \mathit{constr}(C_q)), t)) \Rightarrow \mathit{Bel}(R, \mathit{Bel}(P, \neg p, t), t);$$

$$\textit{for all } \mathrm{var} = (p, \mathit{true\_value}) \in \mathit{outputVar}',$$
$$\mathit{true\_value} = \mathit{true} \wedge \forall q \in \mathit{prop}(P),$$
$$\mathit{Bel}(P, q, t) \wedge (\mathit{Bel}(P, \neg \mathit{CONF}\ (\mathit{Bel}(P, q, t), \mathit{Bel}(R, p, t), t, t, \mathit{constr}(C_q), \mathit{constr}(C_p)), t) \vee \mathit{Bel}(P, \neg \mathit{CONF}\ (\mathit{Int.Tx}(P, q, t, t_q, C_q), \mathit{Bel}(R, p, t), t, t, \mathit{constr}(C_q), \mathit{constr}(C_p)), t)) \Rightarrow \mathit{Bel}(P, \mathit{Bel}(R, p, t), t);$$

$$\mathit{true\_value} = \mathit{false} \wedge \forall q \in \mathit{prop}(P),$$
$$\mathit{Bel}(P, q, t) \wedge (\mathit{Bel}(P, \neg \mathit{CONF}(\mathit{Bel}(R, \neg p, t), \mathit{Bel}(P, q, t_q), t, t, \mathit{constr}(C_p), \mathit{constr}(C_q)), t) \vee \mathit{Bel}(P, \neg \mathit{CONF}(\mathit{Bel}(R, \neg p, t), \mathit{Int.Tx}(P, q, t, t_q, C_q), t, t, \mathit{constr}(C_p), \mathit{constr}(C_q)), t)) \Rightarrow \mathit{Bel}(P, \mathit{Bel}(R, \neg p, t), t).$$

The assumption says that after R received the set of output variables, for each variable var in outputVar, if true_value of var is true (or false) and R believes that no conflict will appear between $\mathit{Bel}(P, p, t)$ (or $\mathit{Bel}(P, \neg p, t)$) and other propositions that R believes being true or intends to (or intends that p being true), R will believe P believes that p is true (or false).

According to Definition 1, for each output variable (prop, true_value) in outputVar which is sent by P to R, (prop, true) stands for $\mathit{Bel}(P, \mathit{prop}, t)$ and (prop, false) stands for $\mathit{Bel}(P, \neg \mathit{prop}, t)$. For each output variable (prop, true_value) in outputVar$'$ which is sent by R to P, (prop, true) stands for $\mathit{Bel}(R, \mathit{prop}, t)$ and (prop, false) stands for $\mathit{Bel}(R, \neg \mathit{prop}, t)$. Take first part of Assumption 6 as example, it says that after R receives output variables from P, for each output variable (prop, true) R believes that its beliefs have no conflict with $\mathit{Bel}(P, \mathit{prop}, t)$, R chooses to believe that $\mathit{Bel}(R, \mathit{prop}, t)$. In other words, $\mathit{Bel}(R, \mathit{Bel}(P, p, t), t)$ holds. Here C_p and C_q stand for context constraints related with p and q, respectively. As for the situation that *true_value* of var is unknown, it gets involved with information hiding, which will be discussed later.

Then the process of CommuAct will be presented in detail. According to Rule 2, a process of dealing with information asymmetry is initiated because conflicts occur between P and R. Such conflicts happen because in the process of deducing p, beliefs and intentions of both agents have conflicts. Consider that R has more information related to p. Before P gets the information that it needs, P should establish reasoning trees about p with its own mental attitudes and rules and find out beliefs and intentions in the tree that P

considers inconsistent with R. These beliefs and intentions are candidates of input variables. And these reasoning trees are state spaces of P for the process of dealing with information asymmetry. They also correspond to F of poor in (2) (in Section 2.2). P can also choose some beliefs and intentions from reasoning trees as output variables. In P's opinion, these variables can help R to get true values of input variables.

We assume that rules of P and R are written with horn clause. That is to say rules follow the schema like $prop_1 \land prop_2 \cdots \land prop_i \cdots \land prop_n \rightarrow prop$ or $prop_1 \land prop_2 \cdots \land prop_i \cdots \land prop \rightarrow \perp$. Then the reasoning tree can be presented with Figure 3. In a reasoning tree, each node is a proposition of the P. Suppose that information asymmetry that is related to proposition prop exists between P and R. In the rule $prop_1 \land prop_2 \cdots \land prop_i \cdots \land prop_n \rightarrow prop$ and $prop_1 \land prop_2 \cdots \land prop_i \cdots \land prop \rightarrow \perp$, prop should be a belief of P. For rule $prop_1 \land prop_2 \cdots \land prop_i \cdots \land prop_n \rightarrow prop$, prop is the root of the tree, and each $prop_i$ is a child node of prop. For rule $prop_1 \land prop_2 \cdots \land prop_i \cdots \land prop \rightarrow \perp$, $\perp$ is the root of the tree, and each $prop_i$ is child node. Every node o in the tree, except the leaf nodes, will have a rule like $prop_{o1} \land prop_{o2} \cdots \land prop_{oi} \cdots \land prop_{on} \rightarrow prop_o$, and each proposition $prop_{oi}$ in the left part of the rule will be the child node of o. prop may have many reasoning trees at one time.

In the reasoning process presented in the previous tree, for the prop in the information asymmetry, some propositions in the tree of prop may be inconsistent with beliefs or intentions of R. And the inconsistency of these propositions may hinder P and R in getting consistent result of prop. So in P's opinion, R's beliefs about these propositions are what P needs. These propositions are candidates of input variables.

Assume that a conflict of proposition prop $Bel(P, q, t_q)$ appears between P and R. R intends that P will believe q at time t_q, while P believes that it won't believe q at t_q. If P has some rules like $prop_1 \land prop_2 \cdots \land prop_i \cdots \land Bel(P, q, t_q) \rightarrow \perp$ or $prop_1 \land prop_2 \cdots \land prop_i \cdots \land prop_n \rightarrow Bel(P, q, t_q)$, such rules can be used to establish the reasoning tree of the process of dealing with information asymmetry. For each $prop_i$ in the rules, P can also find out rules like $prop_{i1} \land prop_{i2} \cdots \land prop_{ii} \cdots \land prop_{in'} \rightarrow prop_i$ and add them into the reasoning tree. Repeating with such a recursion process until no more rules added, the state space of the process of dealing with asymmetry is established. However, there may be rules like $prop'_{i1} \land prop'_{i2} \cdots \land prop'_{ii} \cdots \land prop_{ij} \rightarrow \perp$ for $prop_{ii}$ in the tree. Although such rules do not appear in the reasoning tree, propositions in these rules can also be considered as input variables. P can also choose some propositions, or even rules in the reasoning tree as output variables. Such a process can be implemented with backward chaining algorithm [29], and this paper takes ConstructSpaceP as a basic action here.

In the part 2 of Figure 2, if action CommuAct is successful, R will be aware of the input variables of P with Request in CommuAct.

Assumption 7. As propositions in the previous reasoning tree, propositions $p' \in Rules(p)$ that fulfill the following conditions are taken as input variables and should be put into inputVar of P:

$$\begin{aligned}
&Bel(P, p' \in Rules(p) \neg Bel(P, Bel(R, p', t_{p'}), t) \\
&\land \neg Bel(P, Bel(R, \neg p', t_{p'}), t), t) \Rightarrow \\
&Bel(P, \mathrm{var} = (p', unknown) \in inputVar, t), \\
&Bel(P, p' \in Rules(p) \neg Bel(P, Bel(R, \neg p', t_{p'}), t) \\
&\land Bel(P, Bel(R, \neg p', t_{p'}), t), t) \Rightarrow \\
&Bel(P, \mathrm{var} = (\neg p', unknown) \in inputVar, t).
\end{aligned}$$

The assumption says that for proposition p' in Rules(p), P believes that P does not believe R believes p' is true and P also does not believe that R believes p' is false, and then P believes variable var = (p', unknown) is input variable.

With ConstructSpaceP, P will have input and output variables in inputVar and outputVar, respectively, with the state space being established. Then P will request R to execute act, CommuResponse, and expect for reply for each input variables.

Axiom 2. R gets output variables of P when R believes that P intends R to execute CommuResponse:

$$\begin{aligned}
&Bel(R, \exists t_1 < t_{comR}, Int.Th(P, \\
&Int.To(R, CommuResponse, t_1, t_{comR}, \Theta_{commuRes}), \\
&t, t_1, \Theta_{commuRes}), t) \Rightarrow inputVar' = inputVar \\
&where\ CommuResponse = \\
&CommuResponse(R, P, p, inputVar', outputVar', \\
&t_{comR}, t_{res}, \Theta_{commuRes}).
\end{aligned}$$

Axiom 2 says that when R believes that P intends that "at certain time t_1 before $t_{\mathrm{com}R}$, R intends to execute CommuResponse at time t_1," R gets input variables in inputVar and puts them into inputVar$'$. According to Rule 2, P requests R to execute CommuResponse and sends input variables with the request. If P's request is received by R successfully, R is aware of P's intention of expecting R to intend to execute CommuResponse.

Theorem 8. *In Rule 2, successful performance of CommuAct will make R get the input variables of P.*

Proof. (1) If the performance of CommuAct in Rule 2 is successful, as earlier introduced, the state space of dealing with information asymmetry is set up. With Assumption 7, P gets input variables and puts them into inputVar.

(2) As CommuAct is successfully accomplished, Request in CommuAct is also successful. According to the definition of Request, if $Request(P, R, \varepsilon, \alpha, t_\alpha, \Theta_\alpha)$ is success, there exists a time t'' such that

$$\begin{aligned}
&MB(\{P, R\}, \psi, t) \text{ holds, where} \\
&\psi = \exists t_b < t_a,\ Int.Th(P, Int.To(R, \alpha, t_b, t_a, C_p), \\
&t, t_b, C_p). \text{ Then according to the} \\
&Request(P, R, \varepsilon, CommuResponse(R, P, p, \\
&inputVar', outputVar', t_{comR}, t_{res},), \\
&\Theta_{CommuRes} t, t_{comR}, \Theta_{CommuRes}).
\end{aligned}$$

In Definition 5, α in the ψ is actually CommuResponse, and P and R can get the following mutual beliefs at some time t_{past}:

$$\begin{aligned}&MB(\{P,R\},\exists t_1 < t_{comR}, Int.Th(P, Int.To(R,\\&CommuResponse, t_1, t_{comR}, \Theta_{commuRes}), t_{past},\\&t_1, \Theta_{commuRes}), t_{past}),\\&where\\&CommuResponse = CommuResponse(R, P, p,\\&inputVar', outputVar', t_{comR}, t_{res}, \Theta_{commuRes}).\end{aligned}$$

With Axiom 2, R finally gets input variables in inputVar and put them into inputVar′. □

The mutual belief means that after request P and R both believe that at some time t_{past} before $t_{\text{com}R}$, P wants "R intends to execute CommuResponse at some time t_1" to hold before t_1. The definition of CommuResponse is shown as Definition 9.

Definition 9.

$$\begin{aligned}&CommuResponse(R, P, p, inputVar',\\&outputVar', t, t_{res}, \Theta_{commuRes}) = \textit{for all } \text{var} = (prop, true\_value) \in outputVar',\\&Inform(R, P, \varepsilon, Bel(R, prop, t), t, t_{res}),\\&\textit{if and only if } Hold(\Theta_{commuRes}, t) \wedge true\_value = true,\\&Inform(R, P, \varepsilon, Bel(R, \neg prop, t), t, t_{res}),\\&\textit{if and only if } Hold(\Theta_{commuRes}, t) \wedge true\_value = false.\end{aligned}$$

Executing CommuResponse at time t means that for each output variable var in outputVar′, R informs P about its belief in the proposition in var before time t_{res} under context constraint $\Theta_{commuRes}$. As for variables with true_value being unknown, after outputVar′ are sent to P, they are still with true_value being unknown.

Before R executes CommuResponse, there is still some work which needs to be done, including constructing a process of reasoning for input variables in inputVar′ and finding out true value for each input variable. Similar to P, an act also needs to be defined for R (Definition 10).

Definition 10.

$$\begin{aligned}&CommuRes(P, p, inputVar',\\&outputVar', t, t_{comR}, t_{res}, \Theta_{commu})\\&= ((t < t_{comR})?;\\&ConstructSpaceR(p, inputVar', outputVar', t))?;\\&CommuResponse(R, P, p, inputVar', outputVar',\\&t_{comR}, t_{res}, \Theta_{commuRes}).\end{aligned}$$

Definition 10 says that before $t_{\text{com}R}$ agent R constructs its state space for the process of dealing with information asymmetry with action ConstructSpaceR, then if ConstructSpaceR is successful, R will execute CommuResponse at time $t_{\text{com}R}$, which should be finished before t_{res}. inputVar′ and outputVar′ are sets of input and output variables of R, respectively. The action ConstructSpaceR has some similarity to ConstructSpaceP and will be discussed later.

Rule 3 requires R to form a potential intention on CommuRes after the request about CommuRes received from P.

Rule 3.

$$\begin{aligned}&Bel(R, Int.Th(P, Int.To(R, CommuResponse,\\&t, t_{comR}, \Theta_{commuRes}), t_{past}, t, \Theta_{commuRes}), t) \wedge\\&Bel(R, AsymInfo(P, R, p, inputVar, outputVar,\\&poor, rich, C_p), t) \Rightarrow Pot.Int.to(R,\\&CommuRes(P, p, inputVar', outputVar',\\&t_{exeR}, t_{comR,} t_{res}, \Theta_{commu}), t, t_{exeR}, \Theta_{commu}),\\&where\ t_{past} \leq t < t_{exeR} < t_{comR} < t_{res},\\&CommuResponse = CommuResponse(R, P, p,\\&inputVar', outputVar', t_{comR}, t_{res}, \Theta_{commuRes}).\end{aligned}$$

The rule says that at current time t, R believes that some time earlier (at time t_{past}), P intended that at time t, R would intend to execute CommuResponse at time $t_{\text{com}R}$, and R also believes that information asymmetry exists between itself and P. Then R should have a potential intention to execute action CommuRes at time $t_{\text{exe}R}$. This time $t_{\text{exe}R}$ is a certain time after t. It is defined by R according to the concrete scenario.

If the Request for CommuResponse is successful, R will have belief at some time t as follows:

$$\begin{aligned}&Bel(R, Int.Th(P, Int.To(R, CommuResponse, t,\\&t_{comR}, \Theta_{commuRes}), t_{past}, t, \Theta_{commuRes}), t),\\&CommuResponse = CommuResponse(R, P, p,\\&inputVar', outputVar', t_{comR}, t_{res}, \Theta_{commuRes}).\end{aligned}$$

Then if R believes information asymmetry about p exists between P and R, R should have a potential intention on CommuRes.

Similar to action ConstructSpaceP of P, R uses action ConstructSpaceR in Definition 9 to establish the state space for the process of dealing with information asymmetry. ConstructSpaceR is mainly responsible for the following tasks: updates the mental attitudes with input variables in inputVar′, gets true values for each input variables, and puts these input variables in to outputVar′ after the true_value of each input variables is got by reasoning. It is also responsible for deciding strategies of hiding for output variables.

First, input variables in inputVar′ from P contain beliefs of P about the proposition in each variables. ConstructSpaceR will update mental attitudes of R with such beliefs. With Assumption 6, for each input variable (that is output variable of P) with its true_value being true (or false), R believes that P believes the proposition in the input variable being true (or false). In the process that R updates its own mental attitudes with input variables, the problem of belief revision is involved. Many works have been done on this problem, and some algorithms have been proposed [30–33]. As belief revision is a complex topic, it will be discussed in our future work. ConstructSpaceR is also responsible for judging whether P hides information in CommuAct. However, since P is the agent short of information, it seems that P is short of motivation to hide information in the process of requiring information from R. For simplicity, this paper does not discuss the situation that P hides information in communication with R.

Second, after belief revision finished, R finds out the true value for each input variable. The process is similar to ConstructSpaceP; for variables whose true values can not be gotten directly from beliefs of R, the reasoning tree for each variable is established to see whether the true value of variables can be gotten from R's own mental attitudes and rules. Backward chaining algorithm can also be employed here. For those variables which can not be gotten by the reasoning process, their true_value are set to be unknown. Then these input variables are put into outputVar$'$, waiting for sending back to P.

In parts 3 and 4 of Figure 2, if action CommuRes is successful, P will get responses of the input variables from R.

Axiom 3. When P believes that R intends P to believe R's belief about the true value of input variables, P puts such beliefs in inputVar as input variables with true value:

$$Bel\{P, \exists t_b\ (t_b < t_{res}) \wedge Int.Th(R, Bel(P, Bel(R, p, t), t_b), t, t_b, \Theta_{commuRes}), t\} \wedge Bel(P, (p, unknown) \in inputVar, t) \Rightarrow Bel(P, (p, true) \in inputVar, t) \wedge Bel(P, (p, unknown) \notin inputVar, t),$$

$$Bel\{P, \exists t_b\ (t_b < t_{res}) \wedge Int.Th(R, Bel(P, Bel(R, \neg p, t), t_b), t, t_b, \Theta_{commuRes}), t\} \wedge Bel(P, (p, unknown) \in inputVar, t) \Rightarrow Bel(P, (p, false) \in inputVar, t) \wedge Bel(P, (p, unknown) \notin inputVar, t).$$

The first part says that if P believes that R intends that "at some time t_b, P believes that R believes p" holds, and (p, unknown) belongs to inputVar, P updates (p, unknown) to (p, true) in inputVar. The second part says that if P believes that R intends that "at some time t_b, P believes that R believes p" holds, and (p, unknown) belongs to inputVar, P updates (p, unknown) to (p, true) in inputVar.

Theorem 11. *In Rule 3, successful performance of CommuRes will make P get the output variables of R.*

Proof. (1) According to Rule 3, R forms intention on CommuRes when the proposition

$$Bel(R, Int.Th(P, Int.To(R, CommuResponse, t, t_{comR}, \Theta_{commuRes}), t_{past}, t, \Theta_{commuRes}), t) \wedge Bel(R, AsymInfo(P, R, p, inputVar, outputVar, poor, rich, C_p), t)$$

holds. Then with Axiom 2, R gets input variables of P and puts them into inputVar$'$.

With Assumption 6, the input variables in inputVar$'$ are changed to beliefs of R. And with necessary process of beliefs revision, R updates its mental attitudes. At the same time the state space of dealing with information asymmetry is set up.

(2) As CommuRes is successfully accomplished, ConstructSpaceR and CommResponse are successful. As ConstructSpaceR is successful, R gets true values of input variables in inputVar$'$.

According to Definition 2, if $Inform(A, B, \varepsilon, p, t, t_\alpha)$ Inform is successful, the following proposition holds:

$$\exists t''\ (t \le t'' < t_\alpha) \wedge MB(\{A, B\}, \psi, t''),$$
$$where\ \psi = \exists t_b\ (t'' \le t_b < t_\alpha) \wedge$$
$$Int.Th(A, Bel(B, Bel(A, p, t), t_b), t, t_b, C_p).$$

Then if Inform in CommuResponse is successful, P and R can get the following mutual beliefs at some time t_{past}:

$$MB\{\{P, R\}, \exists t_b\ (t_b < t_{res}) \wedge$$
$$Int.Th(R, Bel(P, Bel(R, p, t), t_b), t, t_b, \Theta_{commuRes}), t\}.$$

With Axiom 3, for each input variables in inputVar$'$, P updates true values of input variables with input variables in inputVar$'$. □

We leave one step in CommuRes for further discussion. Before sending inputVar$'$ to P, R will consider the hiding strategies for these variables and a game between R and P will begin. Before such a step, we should notice that after P receives inputVar$'$ from R, the true value of some variables may be "unknown." Then P can also choose to establish a reasoning process about these variables with some other rules rather than used in current process of establishing state space. It means that P establishes a new state space for variables and P can also start a communicating process as aforementioned. Such a situation is much similar with decomposition in Hierarchical Task Network (HTN) [29]. At first the process of dealing with information asymmetry can be regarded as some high level tasks. Then P receives input variables from R and decides to set up new state space for certain variables and start a communicating process. This new process can be regarded as the high level task decomposing to low level task.

2.3.3. Game in the Process of Dealing with Information Asymmetry in a Proactive Manner. After the second step of ConstructSpaceR, R has output variables in the set outputVar$'$. As information asymmetry exists between P and R, R will consider taking advantage of asymmetry by hiding true values of output variables. Suppose that elements in outputVar$'$ are as follows:

$$\{(prop_1, true), \{prop_2, false\}, \{prop_3, unknow\}\}.$$

According to the strategy of hiding introduced in Section 2.2, R can adopt the following strategies for the outputVar$'$:

$$strategy1: \{0, 0, 1\},\ and\ after\ hidden,$$
$$outputVar' = \{(prop_1, unknown),$$
$$(prop_2, unknown), (prop_3, unknown)\},$$
$$strategy2: \{0, 1, 1\},\ and\ after\ hidden,$$

$outputVar' = \{(prop_1, unknown), (prop_2, false),$
$(prop_3, unknown)\},$
$strategy3\colon \{1, 0, 1\},$ *and after hidden*,
$outputVar' = \{(prop_1, true), (prop_2, unknown$
$(prop_3, unknown)\},$
$strategy4\colon \{1, 1, 1\},$ *and after hidden*,
$outputVar' = \{(prop_1, true), (prop_2, false),$
$(prop_3, unknown)\}.$

The decision of R on whether to hide information depends on the result of game on information hiding between R and P. With Assumption 6, after P receives outputVar$'$ (that is input variables of P) from R, for each variable with true_value being true (or false), P believes that R believes the proposition in the input variable being true (or false). And these variables will be directly used to revise beliefs of P. But for the variables with true_value to be unknown, P will wonder whether the true_values of these variables are hidden, or R really has no idea about the true_values of these variables. If P considers some true values of variables are hidden, P will punish R in order to force R to eliminate hiding behaviours and promote cooperation. In such situation, game theory is employed by both P and R to make decisions in the game. Here, R first decides how to hide information, and then P analyzes what hiding strategy may be adopted by R, so it is reasonable for P and R to use dynamic game model [34] to support their decision. Besides, according to whether P and R are willing to publish payoff in the game, P and R can choose to use complete or incomplete information game model [34]. Here we use dynamic game with complete information.

As the previous example, R's complete strategy set is {strategy1, strategy2, strategy3, strategy4}, and correspondingly P's strategy set is $\{(0,0,0), (0,0,1), (0,1,0), (0,1,1), (1,0,0), (1,0,1), (1,1,0), (1,1,1)\}$. If R adopts strategy4 and then gets inputVar$'$ like $\{(prop_1, true), (prop_2, false), (prop_3, unknown)\}$, P can possibly adopt strategies like $(1,1,0)$ or $(1,1,1)$. Generally speaking, the game between P and R can be presented with the game tree in Figure 4.

In Figure 4, the dotted line presents information set with many nodes [34], which means that P knows its turn to make decision, but it does not know which node in the set it resides; in other words, it does not know the decision of R. At the terminal nodes [34] of the tree, the payoffs for each strategy are labelled with P_{ij} (payoff of P) and R_{ij} (payoffs of R). Such payoffs depend on the concrete scenario. And they influence result of the game.

First, R should take the punishment from P into consideration, as P may find out that R hides information in the process of dealing with asymmetry. Second, R should consider the possibility that P will act as what R expects if R hides some variables. For instance, in the scenario of Section 2.1, after P requests for the true value of "new requirement will bring big change on what has been finished," R decides to tell P that "it is hard to tell whether or not" (which means unknown) instead of the truth that "we have already defined some interfaces to guarantee no big change will be brought about" (which means false). Before such decision, R may consider whether P will choose to give up new requirement or give R more time and more money. The different choice of P will bring R with different payoff.

As for P, different strategies will lead P to take different actions. For instance, P should consider whether to add new requirement when it is short of information about whether new requirement will bring big change on the overall work. So the payoff for each strategy depends on action P will take, and it also depends on its punishment on R. Here P should be careful because that mistaken punishment on R will cause R to punish P back.

After the game is over, P revises its beliefs with input variables, and a process of dealing with information asymmetry in a proactive manner is finished. Here we define the term "the game between P and R effectively constrains information hiding between P and R" as follows. In the game between P and R, R finally chooses not to hide information in communication, and P considers that R does not hide information in communication. As different game mechanisms can be employed when facing different scenarios, in communication between P and R the situations like R hides information regardless of punishment or P has wrong judgment about information hiding may appear. In such situations the process of dealing with information asymmetry will be influenced by information hiding. This problem needs to be discussed in game mechanism design, and we do not take it into consideration here.

Theorem 12. *If P gets response of input variables from R successfully and the game between P and R effectively constrains information hiding between P and R, P's belief about information asymmetry no longer holds.*

Proof. After successful performance of CommuAct and CommuRes, P gets responses for input variables from R. According to Assumption 6, P believes that R is not cheating for input variables. As the game between P and R effectively constrains information hiding between P and R, P believes that R does not hide information in communication. Then each variable in inputVar falls into the following categories.

when var = (prop, true), with the definition of input/output variables, P gets beliefs like

$$Bel(P, Bel(R, prop, t_p), t).$$

when var = (prop, false), with the definition of input/output variables, P gets beliefs like

$$Bel(P, Bel(R, \neg prop, t_p), t).$$

when var = (prop, unknown), with the definition of input/output variables, P gets beliefs like

$$Bel(P, \neg Bel(R, prop, t_p) \wedge \neg Bel(R, \neg prop, t_p), t).$$

Consider that operator Bel follows KD45 axioms of model logic, with P also getting beliefs as follows:

$$Bel(P, Bel(R, Bel(R, prop, t_p), t), t),$$
$$Bel(P, Bel(R, \neg prop, t_p), t),$$

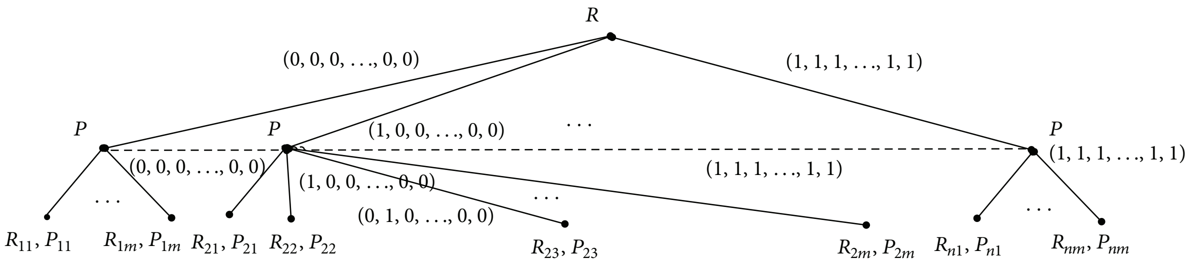

R
$(0, 0, 0, \ldots, 0, 0)$
$(1, 1, 1, \ldots, 1, 1)$
P
P
$(1, 0, 0, \ldots, 0, 0)$
$\ldots$
P
$(1, 1, 1, \ldots, 1, 1)$
$(0, 0, 0, \ldots, 0, 0)$
$(1, 0, 0, \ldots, 0, 0)$
$(1, 1, 1, \ldots, 1, 1)$
$\ldots$
$(0, 1, 0, \ldots, 0, 0)$
$\ldots$
$\ldots$
R_{11}, P_{11}
R_{1m}, P_{1m}
R_{21}, P_{21}
R_{22}, P_{22}
R_{23}, P_{23}
R_{2m}, P_{2m}
R_{n1}, P_{n1}
R_{nm}, P_{nm}

Figure 4: Dynamic game in dealing with information asymmetry.

$$Bel(P, \neg Bel(R, Bel(R, prop, t_p), t) \wedge \neg$$
$$Bel(R, Bel(R, \neg prop, t_p), t), t).$$

According to Assumption 7, belief

$$Bel(P, \exists p' \in Rules(p)\ \neg Bel(P, Bel(R, p', t_{p'}), t) \wedge Bel(R, Bel(R, p', t_{p'}), t), t) \vee Bel(P, \exists p' \in Rules(p)\ \neg Bel(P, Bel(R, \neg p', t_{p'}), t) \wedge Bel(R, Bel(R, \neg p', t_{p'}), t), t).$$

no longer holds. □

Then with Axiom 1, P's belief about information asymmetry no longer holds. With information in inputVar and the process of belief revision, P can complete the reasoning process that it sets up for dealing with information asymmetry.

3. Summary

Information asymmetry brings about problems like adverse selection and negative influence on cooperation between agents. This paper presents a proactive communication process for dealing with information asymmetry in MAS. The main contributions of this paper are as follows. First, a formal description of the communication process for dealing with asymmetry is presented. Previous works pay less attention to the process of dealing with information asymmetry, such as how to start the process, or what the communicating process like. In the work presented here, agent short of information takes initiative to identify and request for the needed information, and detailed steps of communication are also defined.

Second, by combining the communication process with a game theory based model, the work presented here provides a more flexible and effective way to deal with asymmetry between two agents. The game between two agents guarantees that information hiding can be constrained. On the one hand, the game between agents allows agents to take advantage of information asymmetry by hiding information in communication; on the other hand, it also constrains the decisions of each agent according to their interests, respectively.

At last, the process proposed here provides some basic ideas of designing proactive communication process between agents. In the situation of information asymmetry, the agent short of information takes initiative to identify the information it needs and requests information from the agent with information. Such a proactive manner can be used to deal with some other problems in communication, such as in the situation of trust establishment.

Still several important issues deserve further studies. First, more concrete semantic should be considered for information asymmetry. Information asymmetry may be caused by many factors in organization. These factors should be taken into consideration. As the study of information asymmetry has been conducted in information economics, some models will be helpful when agents construct state space and identify input and output variables.

Second, in the process of establishing state space, the controllability and observability of the state space should be taken into consideration. On the one hand, an agent expects another agent who it is communicating with to act as expected, and on the other hand, an agent does not expect to be controlled by other agents. So the choice of the input and the output variables is critical. Too much information exposed will break autonomous of the agent. Too less information exposed will make cooperation less effective.

Third, we mainly focus on information asymmetry between two agents. If three or more agents are involved in information asymmetry, the organization of these agents may influence the process of dealing with information asymmetry. Such situations will be considered in our future work.

Acknowledgments

This work is partially supported by the National 973 Foundation of China (no. 2013CB329304), the National 985 Foundation of China, National Science Foundation of China (nos. 61070202 and 61222210), and the National 863 Program of China (no. 2013AA013204).

References

[1] G. Clarkson, T. E. Jacobsen, and A. L. Batcheller, "Information asymmetry and information sharing," *Government Information Quarterly*, vol. 24, no. 4, pp. 827–839, 2007.

[2] W. z. J. A. Mirrlees, *The Collection of the Theses of James A. Mirrlees*, The Commercial Press, Beijing, China, 1997.

[3] X. Fan, J. Yen, and R. A. Volz, "A theoretical framework on proactive information exchange in agent teamwork," *Artificial Intelligence*, vol. 169, no. 1, pp. 23–97, 2005.

[4] M. Wooldridge, *The logical modelling of computational multi-agent systems [Ph.D. thesis]*, Manchester University, Manchester, UK, 1992.

[5] P. R. Cohen and H. J. Levesque, “Teamwork,” *Nous*, vol. 25, no. 4, pp. 487–512, 1991.

[6] B. J. Grosz and S. Kraus, “Collaborative plans for complex group action,” *Artificial Intelligence*, vol. 86, no. 2, pp. 269–357, 1996.

[7] B. J. Grosz and S. Kraus, “The evolution of SharedPlans,” in *Foundations and Theories of Rational Agency*, A. Rao and M. Wooldridge, Eds., pp. 227–262, Springer, New York, NY, USA, 1998.

[8] H. J. Levesque, P. R. Cohen, and J. H. T. Nunes, “On acting together,” in *Proceedings of the 8th National Conference on Artificial Intelligence*, Boston, Mass, USA, 1990.

[9] 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, Wash, USA, 1994.

[10] K. E. Lochbaum, *Using collaborative plans to model the intentional structure of discourse [Ph.D. thesis]*, Harvard University, Cambridge, Mass, USA, 1994.

[11] F. Anseel and F. Libvens, “A within-person perspective on feedback seeking about task performance,” *Psychologica Belgica*, vol. 46, no. 4, pp. 283–300, 2006.

[12] L. G. Aspinwall and S. E. Taylor, “A stitch in time: self-regulation and proactive coping,” *Psychological Bulletin*, vol. 121, no. 3, pp. 417–436, 1997.

[13] L. G. Aspinwall, “The psychology of future-oriented thinking: from achievement to proactive coping, adaptation, and aging,” *Motivation and Emotion*, vol. 29, no. 4, pp. 203–235, 2005.

[14] T. S. Bateman and J. M. Crant, “Proactive behavior: meaning, impact, recommendations,” *Business Horizons*, vol. 42, no. 3, pp. 63–70, 1999.

[15] J. E. Dutton, S. J. Ashford, R. M. O’Neill, E. Hayes, and E. E. Wierba, “Reading the wind: how middle managers assess the context for selling issues to top managers,” *Strategic Management Journal*, vol. 18, no. 5, pp. 407–425, 1997.

[16] J. M. Crant, “Proactive behavior in organizations,” *Journal of Management*, vol. 26, no. 3, pp. 435–462, 2000.

[17] M. Frese and D. Fay, “Personal initiative (PI): An active performance concept for work in the 21st century,” *Research in Organizational Behavior*, vol. 23, pp. 133–187, 2001.

[18] A. Hwang and J. B. Arbaugh, “Virtual and traditional feedback-seeking behaviors: underlying competitive attitudes and consequent grade performance,” *Decision Sciences Journal of Innovative Education*, vol. 4, no. 1, pp. 1–28, 2006.

[19] P. E. Levy, M. D. Albright, B. D. Cawley, and J. R. Williams, “Situational and individual determinants of feedback seeking: a closer look at the process,” *Organizational Behavior and Human Decision Processes*, vol. 62, no. 1, pp. 23–37, 1995.

[20] G. J. Ruder, *The relationship among organizational justice, trust, and role breadth self-efficacy [Ph.D. thesis]*, Virginia Polytechnic Institute and State University, Falls Church, Va, USA, 2003.

[21] K. D. Stobbeleir, “A self-determination model of feedback-seeking behavior in organizations,” Vlerick Leuven Gent Management School Working Paper Series, 2006.

[22] D. VandeWalle, “A goal orientation model of feedback-seeking behavior,” *Human Resource Management Review*, vol. 13, no. 4, pp. 581–604, 2003.

[23] D. VandeWalle, G. N. Challagalla, S. Ganesan, and S. P. Brown, “An integrated model of feedback-seeking behavior: disposition, context, and cognition,” *Journal of Applied Psychology*, vol. 85, no. 6, pp. 996–1003, 2000.

[24] M. Huth and M. Ryan, *Logic in Computer Science: Modelling and Reasoning about Systems*, Cambridge University Press, Cambridge, UK, 2000.

[25] X. Mao, *Agent-Oriented Software Development*, Tsinghua University Press, Beijing, China, 2005.

[26] M. Wooldridge and N. R. Jennings, “Intelligent agents: theory and practice,” *Knowledge Engineering Review*, vol. 10, no. 2, pp. 115–152, 1995.

[27] S. Kraus and D. Lehmann, “Knowledge, belief and time,” *Theoretical Computer Science*, vol. 58, no. 1–3, pp. 155–174, 1988.

[28] S. Kumar, M. J. Huber, P. R. Cohen, and D. R. McGee, “Toward a formalism for conversation protocols using joint intention theory,” *Computational Intelligence*, vol. 18, no. 2, pp. 174–228, 2002.

[29] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, Prentice Hall, Englewood Cliffs, NJ, USA, 2nd edition, 2002.

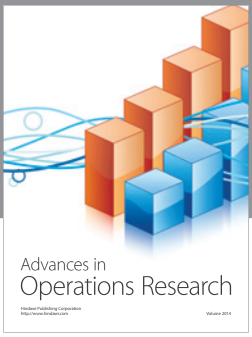
[30] A. F. Dragoni, P. Giorgini, and L. Serafini, “Updating mental states from communication,” in *Proceedings of the 7th International Workshop on Intelligent Agents VII Agent Theories Architectures and Languages (ATAL ’00)*, Boston, Mass, USA, 2000.

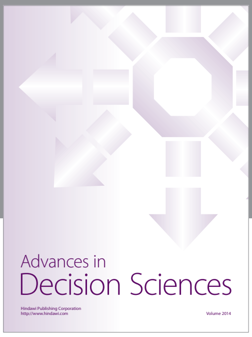
[31] Y. Jin, M. Thielscher, and D. Zhang, “Mutual belief revision: semantics and computation,” in *Proceedings of the 22nd AAAI Conference on Artificial Intelligence and the 19th Innovative Applications of Artificial Intelligence Conference*, pp. 440–445, Vancouver, British Columbia, Canada, July 2007.

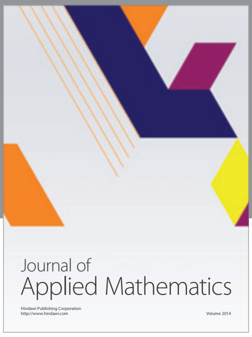
[32] R. Wassermann, “An algorithm for belief revision,” in *Proceedings of the 7th International Conference Principles of Knowledge Representation and Reasoning*, 2000.

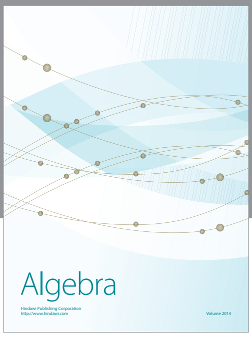
[33] D. Zhang, Z. Zhu, and S. Chen, “Default reasoning and belief revision: a syntax-independent approach,” *Journal of Computer Science and Technology*, vol. 15, no. 5, pp. 430–438, 2000.

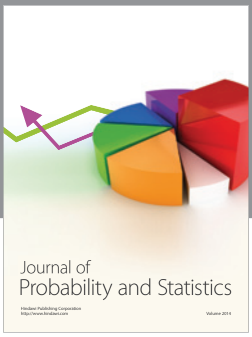
[34] R. Gibbons, *A Primer in Game Theory*, Financial Times/Prentice Hall, London, UK, 1992.


Advances in Operations Research

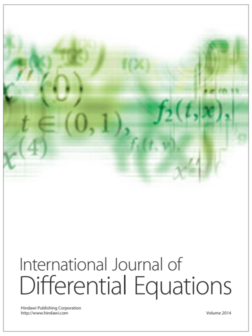

Advances in Decision Sciences


Journal of Applied Mathematics

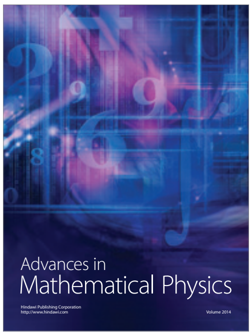

Algebra

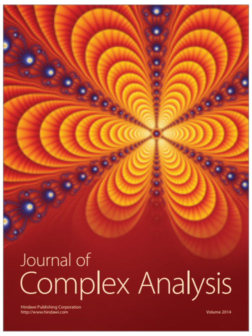

Journal of Probability and Statistics

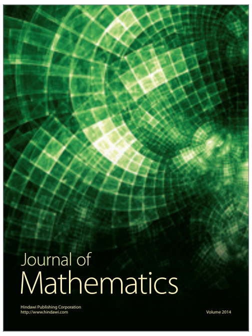
The Scientific World Journal

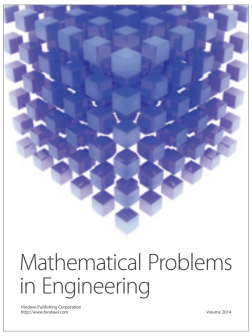

International Journal of Differential Equations

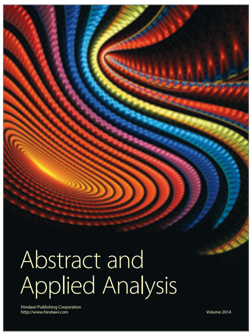
International Journal of Combinatorics

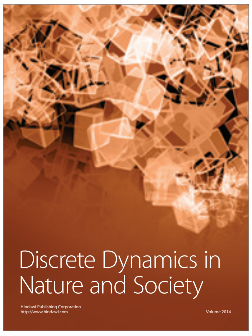

Advances in Mathematical Physics

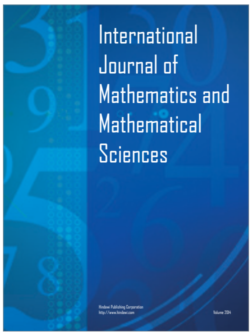

Journal of Complex Analysis


Journal of Mathematics


Mathematical Problems in Engineering

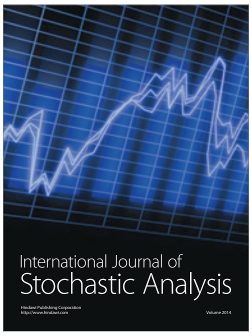

Abstract and Applied Analysis


Discrete Dynamics in Nature and Society


International Journal of Mathematics and Mathematical Sciences

Journal of Discrete Mathematics

Journal of Function Spaces


International Journal of Stochastic Analysis

Journal of Optimization