Belief Updating by Communication Channel

Shingo Hagiwara[1], Mikito Kobayashi[1], and Satoshi Tojo[1]

School of Information and Science, Japan Advanced Institute of Science and Technology,
1–1 Asahidai, Nomi, Ishikawa 923-1292, Japan
{s-hagiwa,m-kobaya,tojo}@jaist.ac.jp,

Abstract. In this paper, we introduce the notion of communication channel into a multiagent system. We formalize the system in term of logic with Belief modality, where each possible world includes CTL. We represent the channel by a reserved set of propositional variables. With this, we revise the definition of *inform* of FIPA; if the channel exists the receiver agent surely comes to know the information whereas if not the action fails. According to this distinction, the current state in each world would diverge into two different states. We have implemented a computer system that works both for a prover and for a model builder. Given a fomula in a state in a possible world, the system proves if it holds or not, while if an *inform* action is initiated the system adds new states with branching paths.

1 Introduction

In this paper, we present a logic of communication in multiagents, together with its computer system.

There have been some logical approaches on the agent communication, however the communicability has rather been neglected . Though [1] treats this issue, it still lacks a sound formalization. The purpose of this study is to discuss how the communicability can be defined in terms of logic.

Among many researches on the multiagent system [2, 3, 4], our framework is based on Rao's research, where the Kripke semantics is given for BDI logic (*Belief*, *Desire* and *Intention*) and *CTL* (Computational Tree Logic), as the semantics seems useful to represent the changes of knowledge states of each agent. As for the communication [5, 6, 7], we fundamentally observe the protocol of FIPA (Foundation for Intelligent Physical Agents)[8, 5], where *inform* is an action of Dynamic Logic, and an agent must satisfy several prerequisites to transfer her knowledge to others.

In this paper, we will introduce a communication channel to represent communicability, particularly, we focus an *inform* action. Besides, we show a computer system which can update a model by the *inform* action and can prove formulae of the logic.

This paper consists of the followings. In Sec.2, we explain how the communication channel is defined, mentioning why it is not a modal operator but a

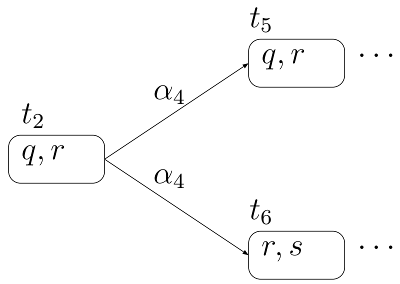

Fig. 1. A branching temporal structure

kind of proposition. In Sec.3, we propose a formalization in the logic of $B_{CTL/C}$, where we define the syntax and the Kripke semantics, in which we will also suggest how we should update a model, when the *inform* action is performed. In Sec.5, we explain our implementation. In the final section, we discuss some problems of our theory and summarize the contributions of this paper.

2 Communicability and Agent Action

In this section, we discuss what are an agent action and its execution, and how we should formalize communicability.

2.1 An Effect of a Communication Channel on an Agent Action

We regard that an agent can choose only one of those actions available at the given time, because we hypothesize that agent is a *Rational Agent*[9]. Therefore, she deliberately evaluates a precondition to act an action and performs it only if it is satisfied.

According to Wooldridge, "the *transitions between states* are *labeled* with actions."However, the result of an action may not be unique; as in Fig.1 the states of t_5 and t_6 show the results of the action. He has explained that "from state t_2, the agent has no choice about what action to perform – it must perform α_4, and either state t_5 or state t_6 will result." This implies that multiple different results may accompany an action.

Accordingly, an agent firstly recognizes the feasible actions, and then, selects an action from them, which immediately comes to the execution. As the action may possibly cause multiple results, multiple state transitions occur. When an agent executes a sequence of actions, because of the multiple different results, the state transitions may diverge. Namely, between the belief of an agent and the real situation, the branching structure of the state transitions may become different. We regard that one of the causes of such divergence could be reduced to the existence of the communication channel. For example, let us assume a situation that an agent sends packets to a receiver, via the Internet. As far as the operation has no trouble, the receiver surely would receive them. However, if some router *en route* intermitted, the operation may not be completed; even worse, the sender may not know whether the packets were successfully received.

Then, instead of the formalization of success/failure in communication, we propose the notion of secure *communication channel*.

2.2 Why communication channel is a proposition?

In this study, we will integrate the communication channel into FIPA's definition, which consists of *Feasibility Precondition* (FP) and *Rational Effect* (RE). FP consists of the preconditions (i.e. one or more propositions) which need to be satisfied before an agent can execute an action, and RE is the effects of the action.

Conceivably, there are many choices as to how we formalize a communication channel; (i) a predicate of first order logic, (ii) a modal operator, (iii) a Cartesian product of two agents, and (iv) a proposition. Here, we discuss the pros and cons of them.

Firstly, (i) the simplest method may be a predicate of FOL, where a channel predicate has two arguments of indices of the sender and the receiver agents, as $channel(i, j)$. Though this representation seems appropriate at a glance, this method requires to reform all of definitions of FIPA. Also, in this case, the universe of a model must be multiply sorted; a set of individuals and a set of agent indices. If so, we also need to provide quantifiers $(\forall, \exists)$ with sort declarations. Thus, we have judged that these issues would deteriorate the strictness of the preceding theories.

Next, (ii) we have considered to regard the channel as a modality, as $C_{ij}\varphi$ which means that φ could be possibly informed from the agent i to j. However, in this case, we should attach the operator C_{ij} to all the propositions to be transferred. In addition, when n agents are given, n^2 operators would be required, which unnecessarily complicates the logic. Moreover, in this case, an agent cannot *inform* the channel itself as a unit of knowledge. The last issue also gives the negative view for the formalization by (iii) i.e., by Cartesian product.

The final option, (iv) the formalization by a proposition, has its own problem. Let c_{ij} be a proposition that is the channel from the agent i to j. If it is a propositional variable, it could own its truth value arbitrarily regardless of the indices i and j and thus could not keep its semantics. On the contrary, if we assume that c_{ij} is a propositional constant, then n^2 constants, besides $\top$ and $\bot$, could be assigned to any propositional variables. All things considered, still we would like to stick to this final option as we would like to treat the channel itself as a payload of the *inform* action, defining a reserved set of channel variables. Moreover, since this method is not incompatible with the logic which is used in action definitions, the logic can be used without modification. In the future, we need to solve remaining problems and to consider better methods.

3 Logic of branching time and epistemic state with communication channel

In this section, we introduce a temporal epistemic logic system $B_{CTL/C}$ based on CTL for reasoning agent's epistemic states with communications. In the logic, an agent's epistemic state is possibly modified by one time step per a communication. Generally, when we consider a multiagent model, it is appropriate to include the branching time.

We summarize the formal definition of syntax and semantics of our logic of $B_{CTL/C}$,, in the following sections. The objective of the logic is to embed the notion of communication channel into the preceding logical framework of Rao [10, 4], i.e., BDI–CTL. However, as the first step to such integration, we restrict available modal operators only to B (belief) now. Further developments will be discussed in Conclusion.

3.1 Syntax of $B_{CTL/C}$

All temporal operators are formed by a pair of (A, E) and (F,X, etc...).

Definition 1. *(Signature) The language L consists of the following vocabulary*

P	*a set of propositional variables*
$Agent$	*a set of agents*
C	*a set of communication channel variables, where $C \subseteq Agent \times Agent$*

In addition, the symbols are used as following:

$\neg, \vee$	*the logical connectives*
$AX, AF, AG,$	*the propositional temporal operators*
B_α	*the propositional epistemic operators, where $\alpha \in Agent$*

Moreover, we define a formula as follows.

Definition 2. *Formula*
Let α be a propositional variable($\alpha \in P$), c_{ij} be a communication channel variable($c_{ij} \in C$ and $i, j \in Agent$). Then, α, c_{ij} are formulae. Let φ, ψ be formulae. Then, $B_i\varphi$, $AX\varphi$, $AF\varphi$, $AG\varphi$, $EX\varphi$, $EF\varphi$, $EG\varphi$, $\neg\varphi$ and $\varphi \vee \psi$ are formulae.

Finally, we define the abbreviated notations as follows:

$$\varphi \wedge \psi \equiv \neg(\neg\varphi \vee \neg\psi) \quad \varphi \supset \psi \equiv \neg\varphi \vee \psi$$
$$EX\varphi \equiv \neg AX\neg\varphi \qquad EF\varphi \equiv \neg AG\neg\varphi$$
$$EG\varphi \equiv \neg AF\neg\varphi$$

3.2 Semantics of $B_{CTL/C}$

Kripke semantics of $B_{CTL/C}$ respects Rao's one[4, 10], though his model includes epistemic operators of D, I which we have omitted.

A Kripke structure $B_{CTL/C}$ is defined as follow a tuple,

$$M = \langle W, \{T_w : w \in W\}, \{R_w : w \in W\}, \{B_i : i \in Agent\}, V, C\rangle$$

Here, W is a set of possible worlds, T_w is a set of states for each $w \in W$, R_w is a binary relation ($R_w \subseteq T_w \times T_w$, however seriality is not guaranteed in our model), v is a truth assignment to the primitive proposition, c is a truth assignment to the communication channel variables. Moreover, B_α is a set of accessibility relations($B_\alpha \subseteq W \times T_w \times W$, e.g. $(w, t, w') \in B_\alpha$), where α is an agent index. Here, if $(w, t, w') \in B_i$ and $t \in T_w$ hold, then $t \in T_{w'}$ holds. The accessibility relation B_i satisfies the axiom KD45 [1].

A satisfaction relation $\models$ is given as follows, where $(w_k, t_l) \models \varphi$ means that φ holds at t_l in w_k, for a possible world w_k and state t_l. Besides, *Path* and $p[i]$ denote a set of paths in a possible world and the i^{th} element of p from t, respectively.

$$\begin{array}{lll}
(w,t) \models \varphi & \Longleftrightarrow & v(w,t,\varphi) \in V \\
(w,t) \models c_{ij} & \Longleftrightarrow & c(w,t,c_{ij}) \in C, \textit{ where } i \textit{ and } j \textit{ are indices of agents} \\
(w,t) \models \neg\varphi & \Longleftrightarrow & (w,t) \not\models \varphi \\
(w,t) \models \varphi \vee \psi & \Longleftrightarrow & (w,t) \models \varphi \text{ or } (w,t) \models \psi \\
(w,t) \models B_j\varphi & \Longleftrightarrow & \forall w'' \in \{w' | (w,t,w') \in B_j\}, (w'',t) \models \varphi \\
(w,t) \models AX\varphi & \Longleftrightarrow & \forall p \in Path,\ (w, p[1]) \models \varphi \\
(w,t) \models AF\varphi & \Longleftrightarrow & \forall p \in Path,\ \exists i \geq 0,\ (w, p[i]) \models \varphi \\
(w,t) \models AG\varphi & \Longleftrightarrow & \forall p \in Path, \forall i \geq 0, (w, p[i]) \models \varphi
\end{array}$$

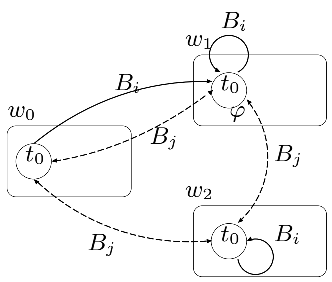

Fig. 2. An example of the operator B

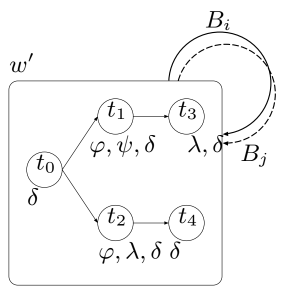

Fig. 3. An example of temporal operators

---

[1] For the operator B and an arbitrary formula φ, the following axioms hold. **K** : $B(\varphi \supset \psi) \supset (B\varphi \supset B\psi)$, **D** : $B\varphi \supset \neg B\neg\varphi$ (seriality), **4** : $B\varphi \supset BB\varphi$ (transitivity) and **5** : $\neg B\neg\varphi \supset B\neg B\neg\varphi$ (Euclidean).

Example 1. Fig.2 is an example of the operator B. In Fig.2, $(w_0, t_0) \vDash B_i\varphi$, $(w_2, t_0) \nvDash B_i\varphi$ and $(w_0, t_0) \vDash B_j\neg B_i\varphi$. Namely, '$B$' is equivalent to the operator '$\Box$' on the accessibility relations. Moreover, in Fig.3 belief-accessibility relations exist in all of the states in w'., $(w', t_0) \vDash AX\varphi$, $(w', t_0) \vDash EX\psi$, $(w', t_0) \vDash AF\lambda$, $(w', t_0) \vDash AG\delta$, and $(w', t_0) \models AXB_i\varphi$.

4 *inform*

The original definition of *inform* of FIPA[8, 5] is as follows.

Definition 3. *FIPA Inform Act*

$\langle i, \mathit{inform}(j, \varphi)\rangle$

$$\mathit{feasibility\ pre\text{-}condition}: B_i\varphi \wedge \neg B_i(\mathit{Bif}_j\varphi \vee \mathit{Uif}_j\varphi)$$
$$\mathit{rational\ effect}: B_j\varphi$$

A formula $B_j\varphi$ means that an agent j believes φ, and a formula $U_j\varphi$ means that an agent j is uncertain about φ but the agent supposes that φ is more likely than $\neg\varphi$. Also, $\mathit{Bif}_j\varphi$ and $\mathit{Uif}_j\varphi$ are the abbreviations of $B_j\varphi \vee B_j\neg\varphi$ and $U_j\varphi \vee U_j\neg\varphi$, respectively.

First, we exclude the epistemic operator U from the FIPA's definition because U has not been strictly formalized in terms of logic. Then, we revise the *inform* action as follows.

Definition 4. *inform Act(Revised)*

$\langle i, \mathit{inform}(j, \varphi)\rangle$

$$\mathit{feasibility\ pre\text{-}condition}: B_i\varphi \wedge \neg B_i(\mathit{Bif}_j\varphi) \wedge B_ic_{ij}$$
$$\mathit{rational\ effect}: (B_iB_j\varphi),\ \mathit{or}\ (B_iB_j\varphi \wedge B_j\varphi \wedge B_jB_i\varphi)$$

In the revised definition, we added B_ic_{ij} to FP because we supposed that a sender agent should recognize the communication channel. Moreover, we changed FIPA's RE to $(B_iB_j\varphi)$, $\mathit{or}(B_iB_j\varphi \wedge B_j\varphi \wedge B_jB_i\varphi)$. $B_iB_j\varphi$ means that intended transfer could hopefully be fulfilled, and $B_iB_j\varphi \wedge B_j\varphi \wedge B_jB_i\varphi$ implies that knowledge actually arrived. Therefore, the formalization could represent two different cases at the next time as follows. In case $B_j\varphi$ is not ensured, a branch appears.

In both of Fig.4 and Fig.5, an *inform* action takes place in the state t_1. However, Fig.4 means that the communication channel actually exists, and Fig.5 means that it is not actually guaranteed. Namely, the definition of our *inform* action represents these two cases at the same time.

5 A model-building system for $B_{CTL/C}$

In this section, we explain the model-building system of $B_{CTL/C}$, implemented in **SWI-Prolog** [11]. We show that the model-building system evaluates the truth values of logical formulae. Also, it works as a model builder, adding new epistemic states in each world.

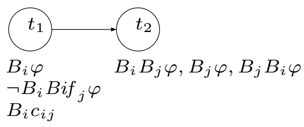

Fig. 4. An example of non branching

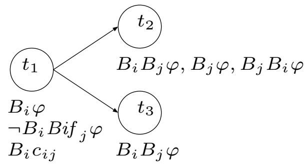

Fig. 5. An example of branching

5.1 Procedure of *prove*

First, we give the procedure $prove(w, t, \varphi)$ which proves φ in the given state t and in the possible world w. It returns the result as to whether $(w, t) \models \varphi$ holds or not.

Algorithm 1 $prove(w, t, \varphi)$
Let W be a set of worlds, T_w a set of states, R_w the relation among states, V a truth assignment of propositional variables, C a truth assignment of communication channel variables, B_α a set of belief accessible relation (α is an index of an agent), respectively. Then $w \in W$, $t \in T$. In addition, ψ and χ are subformulae of φ. The sequence of the procedure is as follows:

1 if $\varphi \equiv \psi \wedge \chi$, then execute $prove(w, t, \psi)$ and $prove(w, t, \chi)$. If both return '`YES`', then return '`YES`', else '`NO`'.
2 if $\varphi \equiv \psi \vee \chi$, then execute $prove(w, t, \psi)$ and $prove(w, t, \chi)$. If one of them returns '`YES`', then return '`YES`', else '`NO`'.
3 if $\varphi \equiv \neg\psi$, then execute $prove(w, t, \psi)$. If it returns '`NO`', then return '`YES`', else '`NO`'.
4 if $\varphi \equiv \psi \supset \chi$, then execute $prove(w, t, \neg\psi \vee \chi)$. If it returns '`YES`', then return '`YES`', else '`NO`'.
5 if $\varphi \equiv AX\psi$, then execute $\forall(t, w, t') \in T_w$, $prove(w, t', \psi)$. If all of them return '`YES`', then return '`YES`', else '`NO`'.
6 if $\varphi \equiv AG\psi$, then execute $prove(w, t, \psi)$ and $T' = \{t' | t'$ is reachable from t with transitivity, $t' \in T_w\}$, $\forall t' \in T'$, $prove(w, t', \psi)$. If all of them return '`YES`', then return '`YES`', else '`NO`'.
7 if $\varphi \equiv AF\psi$, then execute $prove(w, t, \psi)$. If it returns '`YES`', then return '`YES`', else $\forall(t, w, t') \in T_w$, $prove(w, t', AF\psi)$, and if all of them return '`YES`', then return '`YES`', else return '`NO`'.
8 if $\varphi \equiv EX\psi$, then execute $prove(w, t, \neg AX\neg\psi)$. If it returns '`YES`', then return '`YES`', else '`NO`'.
9 if $\varphi \equiv EG\psi$, then execute $prove(w, t, \neg AF\neg\psi)$. If it returns '`YES`', then return '`YES`', else '`NO`'.
10 if $\varphi \equiv EF\psi$, then execute $prove(w, t, \neg AG\neg\psi)$. If it returns '`YES`', then return '`YES`', else '`NO`'.
11 if $\varphi \equiv B_i\psi$, then, for $\forall(w, t, w') \in B_i$, execute $prove(w', t, \psi)$. If all of them return '`YES`', then return '`YES`', else '`NO`'.

12 $prove(w,t,\varphi)$ does not fall into 1–11 rules, if $(w,t,\varphi) \in V$ or $(w,t,\varphi) \in C$, then return '`YES`', else '`NO`'.

In this algorithm, each rule just resolves φ into subformulae ψ and χ with some operators. Moreover, in a cycle on B_i and R_w, our program does not check nodes which was checked at once. Therefore, the algorithm necessarily halts.

5.2 Procedure of *inform*

Here, we give the algorithm of the *inform* action as follows. However, in this implementation, φ is restricted to a propositional variable or a communication channel variable, because φ possibly has temporal operators. If φ has temporal operators, then we need to consider additional states which are beyond the next time step in updating the model. Moreover, for $F\varphi$, we cannot determine the state in which φ holds.

Algorithm 2 $inform(w,t,i,j,\varphi)$

Let W be a set of worlds, T_w a set of states, R_w the relation among states, v a truth assignment of propositional variables, c a truth assignment of communication channel variables, *Agent* a set of agents, and B a set of belief accessible relation, respectively. Also, let *FP* be our feasible pre-condition formula $B_i\varphi \wedge \neg B_i Bif_j\varphi \wedge B_i c_{ij}$. Then $w \in W$, $t \in T$ and $i, j \in Agent$. The procedure becomes as follows.

1 If $(w,t) \models FP$ hold, then goes to 2, else the system returns '`NO`', and ends.
2 $\forall w \in W, T'_w = T_w \cup \{t', t''\}$, where t' and t'' are new states which are added to the model. And, $\forall w \in W$, $R'_w = R_w \cup \{{}_tR_{wt'}, {}_tR_{wt''}\}$, where ${}_tR_{wt'}$ and ${}_tR_{wt''}$ are new state transitions.
3 $W^1 = \{w'' | (w,t,w') \in B_i, (w',t,w'') \in B_j\}$, if φ is a proposition then go to 4, else φ is a communication channel, then go to 5.
4 $V_1 = \{v(w^1,t',\varphi), v(w^1,t'',\varphi) | w^1 \in W^1\}$ go to 6.
5 $C_1 = \{c(w^1,t',\varphi), c(w^1,t'',\varphi) | w^1 \in W^1\}$ go to 6.
6 $W^2 = \{w' | (w,t',w') \in B_j\}$, , if φ is a proposition then go to 7, else φ is a communication channel, then go to 8.
7 $V_2 = \{v(w^2,t',\varphi) | w^2 \in W^2\}$ go to 9.
8 $C_2 = \{c(w^2,t',\varphi) | w^2 \in W^2\}$ go to 9.
9 $V'{=}V \cup V_1 \cup V_2$, $C'{=}C \cup C_1 \cup C_2$,
10 $M' = \langle W', T'_w, R'_w, Agent, V', C' \rangle$ is the updated model.

This procedure firstly tries to prove FP as follows.

$$?-\mathtt{prove}(\mathtt{world},\mathtt{state},\mathtt{B_i}\varphi \wedge \neg\mathtt{B_i}(\mathtt{B_j}\varphi \vee \mathtt{B_j}\neg\varphi) \wedge \mathtt{B_i}\mathtt{c_{ij}})$$

This query valuates whether the agent i can select (execute) the action. If the result is '`YES`', then the system adds two new states, each of which represents

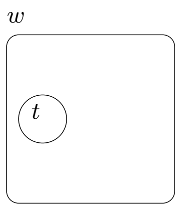

Fig. 6. before performing the *inform* action

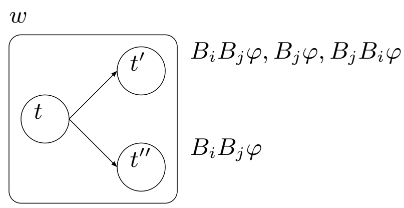

Fig. 7. after performing the *inform* action

a new state dependent on whether knowledge was transferred or not, as Fig.6 and 7. Then, the accessible relations between t' and t'' are copied from t. Finally, $B_iB_j\varphi$ is asserted in t', and $B_iB_j\varphi$, $B_j\varphi$ and $B_jB_i\varphi$ are asserted in t''.

Note that t' is the case in witch the communication channel exists, whereas in case t'' the channel does not in Fig.7.

Our model-building system ensures that two new states are added to the current state, although c_{ij} (let i and j be indices of a sender agent and a receiver, respectively) holds in the state from which the action is performed, because c_{ij} is headed by B_i; in our model-building system, the proposition without an epistemic operator is meaningless.

5.3 An execution example

We assume a situation where three computers, a sender, a receiver and DNS (Domain Name Server) exist. Here, we show an example of an *inform* action, where indices of s, r and d mean *sender*, *receiver* and *dns* in this description, and '`##`', '`&&`', '`=>`' and '`~`' mean $\vee$, $\wedge$, $\supset$ and $\neg$, respectively and '`bel(i, p)`' means B_ip in each execution log.

Firstly, the sender tries to send the packet p. However, the sender does not believe in the communication channel from the sender to the receiver, and thus, the action fails (FP is not satisfied). Secondly, the DNS informs the sender of the existence of the channel. Hence, the sender could inform the receiver of p. Then, only t_1 exists in each world in the initial model shown in Fig.8.

We show some computer screens for the example. Firstly, when the model-building system reads the file of the model definition, it compiles the definition to the middle data, and checks the accessible relations of belief in the model if they satisfy **KD45**. If the system finds errors, then it outputs the error relations and fixes automatically. The model-building system could not choose an appropriate serial access because it could not decide the accessibility relation which is defined by '$\Diamond$' automatically. We intentionally use an incomplete model which does not have the belief accessibility relations from w_5 to w_5 and from w_{17} to w_{16} as follows.

```
 ?- model_checking(dns).
Agent = agent(sender) : State = state(1)
```

```
axiom D is not satisfied on
 [world(5), world(17)]
axiom 4 OK
axiom 5 is not satisfied on
 [world(14), world(16)]
add relation in AXIOM 4
add relation in AXIOM 5
SYSTEM >data(dns, relation(belief, agent(sender),
   node(state, world(17), state(1)),
   node(state, world(17), state(1))))
SYSTEM >data(dns, relation(belief, agent(sender),
   node(state, world(16), state(1)),
   node(state, world(16), state(1))))
Agent = agent(dns) : State = state(1)
axiom D OK
axiom 4 OK
axiom 5 OK
Agent = agent(receiver) : State = state(1)
axiom D OK
axiom 4 OK
axiom 5 OK
Yes
```

.

Next, we check the receiver's knowledge about whether she knows p or not as follows.

```
 ?- prove(world(1),state(1),bel(receiver,p)).
 NO
```

In the state t_1 in the world w_1, she does not know it.

Then, we show the *inform* action; the sender agent tries to inform the receiver of p.

```
 ?- inform(world(1),state(1),sender,receiver,p).
SYSTEM >knowledge is atom, ok
SYSTEM >INFORM: CANNNOT EXECUTE!
YES
```

Since the sender does not believe in a channel from the sender to the receiver, this action fails. Next, DNS informs the sender of the existence of the channel.

```
 ?- inform(world(1),state(1),dns,sender,channel(sender,receiver)).
SYSTEM >knowledge is atom, ok
SYSTEM >FP bel(dns, channel(sender, receiver))&&
        ~bel(dns, bel(sender, ~channel(sender, receiver))
        ##bel(sender, channel(sender, receiver)))&&
        bel(dns, channel(dns, sender)) is satisfied
SYSTEM >INFORM: AGENT dns -[channel(sender, receiver)]
        -> AGENT sender
```

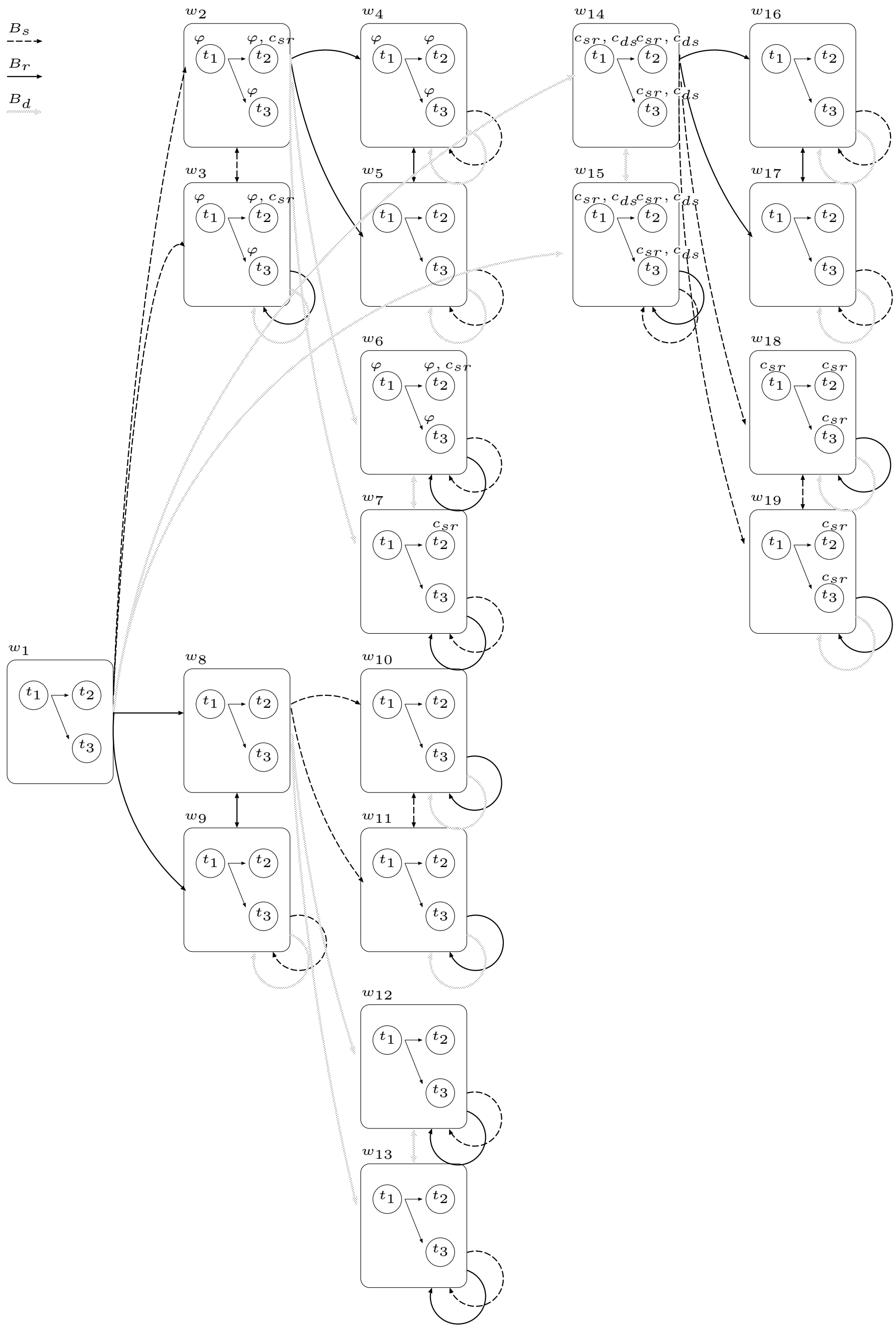

Fig. 8. An model for Example of Sec.5.3 after the *inform* action

```
SYSTEM >Effect: bel(dns, bel(sender, channel(sender, receiver)))
SYSTEM >Effect: bel(sender, channel(sender, receiver))
SYSTEM >Effect: bel(sender, bel(dns, channel(sender, receiver)))
Yes
```

Since the *dns* agent believes in the channels from the *dns* to the sender and from the sender to the receiver, the action succeeds. Thus, the states t_2 and t_3 are added to all the worlds, and c_{sr} was added to the state t_2 in the world $w_2, w_3, w_6, w_7, w_{19}$ and t_3 in w_{19} in the model of Fig.8. Then, the *dns* agent comes to believe in $B_d B_s c_{sr}$ in the state t_2 and t_3, and the sender comes to believe in $B_s c_{sr}$ and $B_s B_d c_{sr}$ in t_2 in w_1. Therefore, t_2 means the result that the communication channel actually exists, and t_3 means that it does not. In case the channel actually exists in the state t_1 in the world w_1, then the model-building system can erase t_3, though we have not implemented the erasing function yet. In future, we are going to revise it so as to prune unless branches, given the actual data of channels.

Next, the sender retries to inform the receiver of p, as follows.

```
 ?- inform(world(1),state(2),sender,receiver,p).
SYSTEM >knowledge is atom, ok
SYSTEM >FP bel(sender, p)&& ~bel(sender, bel(receiver, ~p)
 ##bel(receiver, p))&&bel(sender, channel(sender, receiver))
        is satisfied
SYSTEM >INFORM: AGENT sender -[p]-> AGENT receiver
SYSTEM >Effect: bel(sender, bel(receiver, p))
SYSTEM >Effect: bel(receiver, p)
SYSTEM >Effect: bel(receiver, bel(sender, p))
Yes
```

Here, the action succeeds. Then, the states t_4 and t_5 which are accessible from t_2 are added in the similar way to the above action (*ex facto*, t_6 and t_7 also need to be added to t_3; however, in t_3, since c_{sr} was not recognized by the sender agent, we do not need to consider it.), and p is added to t_4 in w_5, w_8, w_9, w_{10} and w_{11}, and to t_5 in w_5. Then, $B_s B_r p$ holds in t_4 and t_5 in w_1, and $B_r p$ and $B_r B_s p$ hold in t_4 in w_1.

Finally, we check the receiver's knowledge as follows:

```
 ?- prove(world(1),state(4),bel(receiver,p)).
Yes
 ?- prove(world(1),state(1),'EF' bel(receiver,p)).
Yes
```

Therefore, we could confirm that she came to believe p on t_5 in w_1, and this means that communication channels exist both on transitions from t_1 to t_3 and from t_3 to t_5 in w_1.

6 Conclusion

In this paper, we introduced the notion of *communication channel* in BDI–CTL logic, and revised the definition of the *inform* action. We have developed a computer system that could prove whether a given formula belongs to $B_{CTL/C}$ or not. Also, performing the *inform* action, the system updates the model by adding new states and new transitions.

In order to improve the definition of *inform*, we changed the contents of FP and RE. In FP, we introduced the recognition of communication channel $B_i c_{ij}$, and thus the agent could perform the action based on it. In RE, we provided the recognition of $B_i B_j \varphi$ and $B_j B_i \varphi$ for agents; they represent that agents could recognize the post state, after they performed the action. RE was denoted by $(B_i B_j \varphi)$, *or* $(B_i B_j \wedge B_j \varphi \wedge B_j B_i \varphi)$; the '*or*' implies that either the channel exists or not. This post-conditions are similar to the operator '$\oplus$' of Wooldridge[12], though our operation is independent of any specific computer system.

As for the formalization of the communication channel, we employed the way of *channel propositional variables*. Thus, we could transfer the channel itself as a piece of knowledge to other agents.

By the way, the *inform* action in the current system could inform only an atomic formula. Otherwise, an agent can inform the other agent of $\neg B\varphi \wedge \varphi$, and if $\neg B_j \varphi \wedge \varphi$ was informed from the agent i to j then j would believe to $B_j(\neg B_j \varphi \wedge \varphi)$; a contradiction. However, since it is meaningless that the agent i informs $B_j \varphi$ to the agent j, we should prohibit such an action.

As a future work, we need to include the epistemic operators of *Intention* and *Desire* into the logic, for an agent to initiate the *inform* action. With the intention modality, we may be able to represent an autonomous communication in agents. Thus far, we have only treated the *inform* action. According to the FIPA, there are many actions in the multiagent system. Among them, we especially need to implement the *request* action. Together with *inform* and *request*, we will be able to represent the *confirm* action where agents can conceive the existence of communication channels between them.

References

[1] R.M. van Eijk, F. S. de Boer, W. van der Hoek, J. J. Ch. Meyer: Process algebra for agent communication: A general semantic approach. In Huget, M., ed.: Communication in Mulitiagent Systems - Agent Communication Languages and Conversation Policies. Volume 2650., Springer-Verlag (2003) 113–128

[2] A. E. Emerson, J. Srinivasan: Linear time, branching time and partial order in logics and models for concurrency. In J. W. de Bakker, W. P. de Roever, Rozenberg, G., eds.: Branching time temporal logic. (1989) 123–172

[3] A. E. Emerson, J. Y. Halpern: Decision procedures and expressiveness in the temporal logic of branching time. In: Proceedings of the 14th Annual ACM Symposium. (1982) 169–180

[4] A. S. Rao, M. P. Gergeff: Decision procedures for bdi logics. Journal of Logic and Computation **9**(3) (1998) 293–342

[5] FIPA. Foundation for Intelligent Physical Agents: Communicative act library specification (2002) http://www.fipa.org.

[6] P. R. Cohen, H. J. Levesque: 8. In: Rational interaction as the basis for communication. MIT Press, Cambridge (1990) 221–255

[7] T. Finin, D. McKay, R. Fritzson, R. McEntire: KQML : An Information and Knowledge Exchenge Protocol. In: Knowledge Building and Knowledge Sharing. Ohmsha and IOS Press (1994)
[8] FIPA Foundation for Intelligent Physical Agents: Fipa 97 part 2 version 2.0: Agent communication language specification (1997) http://www.drogo.cselt.it/fipa.org.
[9] Wooldridge, M.: Reasoning about Rational Agent. The MIT Press (2000)
[10] A. S. Rao, M. P. Gergeff: Modeling rational agents within a bdi-architecture. In: Proceeding of International Conference on Principles of Knowledge Representation and Reasoning. (1991)
[11] : SWI-Prolog Version 5.6.2. (2006) University of Amsterdam, http://www.swi-prolog.org/.
[12] Wooldridge, M., Fisher, M., Huget, M.P., Parsons, S.: Model checking multi-agent systems with mable. In: AAMAS'02, ACM (2002)