

# 合理的エージェントのコミュニケーション実現方式

高田 司郎<sup>†</sup> 新出 尚之<sup>††</sup> 五十嵐 新女<sup>†,††</sup> 間瀬 健二<sup>†</sup>

<sup>†</sup> ATR 知能映像通信研究所 <sup>††</sup> 奈良女子大学大学院人間文化研究科

## 概要

本稿では、人間または他のエージェントと協調して問題解決を行う合理的エージェントのコミュニケーション実現方式を提案する。合理的エージェントの実現には、BDI アーキテクチャを用い (1) エージェント間のコミュニケーションをプラン本体に自然に表現 (2) プランに未来指向的意図の形成条件の追加 (3) エージェントの置かれた状況や心的状態に応じて、コミットメント戦略の動的な選択などの拡張を提案し、この拡張 BDI アーキテクチャの有効性を応用事例で示す。

## Design of Rational Agents who Interact and Communicate with Other Agents

SHIRO TAKATA,<sup>†</sup> NAOYUKI NIDE,<sup>††</sup> YOSHIME IGARASHI<sup>†,††</sup> and KENJI MASE<sup>†</sup>

<sup>†</sup> ATR Media Integration & Communications Research Laboratories

<sup>††</sup> Graduate School of Human Culture, Nara Women's University

## Abstract

In this paper, we propose a design of rational agents that can interact and communicate with other agents on an extended BDI architecture. Then, we will show how effective it is by demonstrating a case based on such an BDI system that is extended: (1) to naturally describe communications with agents in the form of the body actions in a plan itself. (2) to add conditions of agents' formalization of future-directed intention to a plan, and (3) to enable agents to dynamically decide on a commitment strategy depending on the situation and the mental state in which they are.

## 1 はじめに

Bratman [1] の意図の理論によれば、意図は合理的行為者が自らの目標を達成するよう努力している期間の行為の決定に重要な役割を担っている。そして、意図は、その特性として、時が来れば予め意図したプランを実行するよう動機づける意欲的なコミットメントと、再考慮に抵抗し、意図したプランの空所を一貫性を持って埋めたり修正したりする実践的推論への入力として中心的な要素となるコミットメントを持っている。この意図の理論を合理的エージェントに適用すると、合理的エージェントは、人間から権利委譲された仕事に対して与えられた目標を達成するために、まず、熟考して選択したプランを意図として形成する。その後、どうにかして意図を持続しようとするコミットメントの支配の下に、そのプランを実施するための実践的な手段やサブプランを推論したり、その時が来ればそれらプランを意図的に実行する。

Rao と Georgeff [2, 3] はこの意図の理論を熟考型エージェントの内部アーキテクチャに適用した BDI アーキテクチャを提案している。

本稿では、人間または他のエージェントと協調して問

題解決を行う合理的エージェントのコミュニケーション実現方式を提案する。信念、目標、意図に基づく合理的エージェントの実現に、BDI アーキテクチャを用いる場合、まず、エージェント間のコミュニケーション実現に関する BDI アーキテクチャの問題点を挙げ、次に (1) エージェント間のコミュニケーションをプラン本体に自然に表現できるプラン (2) プランに、将来実行するという未来指向的意図 [1] の形成条件の追加 (3) 意図の持続性のタイプを決定するコミットメント戦略を、エージェントの置かれた状況や心的状態に応じて、動的に選択できる意図の持続機能、などの拡張を提案する。そして、この拡張 BDI アーキテクチャの有効性を応用事例で示す。

以下、2 章では BDI logic の概要を、3 章では BDI アーキテクチャの概要と問題点を、4 章では、これら問題点を解決するための拡張 BDI アーキテクチャの提案を、5 章では拡張 BDI アーキテクチャの有効性を示すための応用事例を、6 章では提案手法に関する考察を、それぞれ述べ、最後に 7 章でまとめる。なお、本稿を読むに当たっては一階述語論理と時相論理 [4] の初歩的な知識を仮定している。

## 2 BDI Logic

BDI logic [3, 5] は、エージェントの過去から未来に至る信念・目標（願望）・意図などの心的状態や行為などを自然に記述することができる時相論理体系の一つである。分岐時間命題時相論理の一つである CTL\* [6] に、信念・目標・意図といった心的状態を表す様相オペレータ BEL, GOAL, INTEND を導入し、さらに、述語論理に拡張した体系である。時相オペレータとしては X (nexttime), F (sometime), U (until), E (ある未来で), A (あらゆる未来で) などを持つ。Rao と Georgeff は Bratman [1] が分析した心的状態の整合性に関する公理や意図の持続性のタイプに関するコミットメント戦略などを、以下のように BDI logic を用いて自然に表現している [3]。

- 心的状態の整合性公理

- (A1)  $GOAL(\alpha) \supset BEL(\alpha)$
- (A2)  $INTEND(\alpha) \supset GOAL(\alpha)$
- (A3)  $INTEND(does(e)) \supset does(e)$
- (A4)  $INTEND(\phi) \supset BEL(INTEND(\phi))$
- (A5)  $GOAL(\phi) \supset BEL(GOAL(\phi))$
- (A6)  $INTEND(\phi) \supset GOAL(INTEND(\phi))$
- (A7)  $done(e) \supset BEL(done(e))$
- (A8)  $INTEND(\phi) \supset AF(\neg INTEND(\phi))$

- コミットメント戦略

- (A9A)  $INTEND(AF \phi) \supset A(INTEND(AF \phi) U BEL(\phi))$
- (A9B)  $INTEND(AF \phi) \supset A(INTEND(AF \phi) U (BEL(\phi) \vee \neg BEL(EF \phi)))$
- (A9C)  $INTEND(AF \phi) \supset A(INTEND(AF \phi) U (BEL(\phi) \vee \neg GOAL(EF \phi)))$

たとえば、(A5) は、直感的には「エージェントが  $\phi$  という状態の達成が目標であるならば、エージェントは  $\phi$  という状態の達成が目標であることを信じている」という意味である。

(A9A), (A9B), (A9C) は、それぞれ *blind*, *single-mind*, *open-mind* と呼ばれるコミットメント戦略である。直感的には、それぞれ「エージェントが  $\phi$  という状態をあらゆる未来でいつか達成するという意図を形成しているならば、あらゆる未来で、エージェントは  $\phi$  という状態を達成したと信じるまでその意図を持続する」、「エージェントが  $\phi$  という状態をあらゆる未来でいつか達成するという意図を形成しているならば、あらゆる未来で、エージェントは  $\phi$  という状態を達成した、または、 $\phi$  という

状態は達成不可能だと信じるまでその意図を持続する」、「エージェントが  $\phi$  という状態をあらゆる未来でいつか達成するという意図を形成しているならば、あらゆる未来で、エージェントは  $\phi$  という状態を達成したと信じるか、または、 $\phi$  という状態を目標から撤回するまでその意図を持続する」という意味である。

## 3 BDI アーキテクチャ

本章では、BDI アーキテクチャの概要とエージェント間のコミュニケーション実現に関する問題点を述べる。

### 3.1 BDI アーキテクチャの概要

BDI アーキテクチャ [2] は、限られた計算資源環境において動的に変化する環境を知覚し合理的に問題解決を行うために事前にプランを立てる熟考型エージェントの内部アーキテクチャである。エージェントの信念・目標・意図をそれぞれ陽に表現する動的なデータ構造 B, G, I やプランライブラリ、イベントキュー、およびそれらデータ構造を参照・更新するインタプリタから構成される。

プランライブラリは、複雑なプランに関する信念を B から分離したもので、タイプ、連想 (invocation) 条件 (プラン形成条件)、前提条件 (プラン実行条件)、add/delete list (プラン実行の成功時に B に加除される信念)、本体 (実現手段) などをスロットとして持つプランの集合である。プラン本体は、基本行為 (実行可能な行為) とサブゴール (当プランの部分目標となるプラン) を迎に持つ木構造をしている。

### 3.2 プランの問題点

従来の BDI アーキテクチャ [2] のプラン本体には、基本行為とサブゴールしか記述できないため、これらの“条件分岐”<sup>1</sup>や他のエージェントからのメッセージを待つ“同期”などを表現することができない。たとえば、エージェント間のコミュニケーションをプランとして記述しようとした場合、交換されるメッセージを連想条件に持つプランを複数に分けて記述するしかなく、エージェント間の本来連続した言語行為の関連が不明瞭となるばかりでなく不自然な記述になるという問題点がある。

また、プランのスロットには、このプランを将来実行するという未来指向的意図の形成条件としては連想条件しかなく<sup>2</sup>、しかも、サブゴールではないトップレベルの目標の追加 (g-add) は外部イベントとして外部から与えられるか、または、心的状態として目標 G に記述されて

<sup>1</sup>従来のプラン本体は木構造であるが、CTL\* の未来への時間分岐を表現しているに過ぎない。

<sup>2</sup>Rao らの BDI アーキテクチャは、まさに実行しようとしている、または、現在実行している現在指向的意図を主な対象としていると捉えている。そのため前提条件は現在指向的意図の実行条件と捉えている。

いることが仮定されている．このため，他のエージェントからのメッセージや自らの心的状態などの状況にマッチした目標や意図の形成をエージェント自身で決定するようなプラン記述ができない．特に，人間との自然言語対話システムなどでは，人間から明示的に目標が与えられることは少なく，文脈から対話プランの推論を行う必要があるが，現状の連想条件のみでは対応できない．

### 3.3 コミットメント戦略の問題点

Rao と Georgeff [3] が形式化した *blind*, *single-mind*, *open-mind* などのコミットメント戦略は「エージェントのタイプとしてエージェント単位に設定する」場合だけでなく，各エージェントの心的状態や自らの置かれた状況に応じて動的に選択した方が良い場合が考えられる．

たとえば，彼らの事例 [2] において，冷蔵庫にソーダ水が入っていればソーダ水を飲む．入っていなければ，水道の蛇口をひねって水を飲むことで「喉の乾きを潤す」という目標を達成するエージェントのタイプとしては *single-mind* コミットメント戦略が設定されている．我々は，この事例においても，以下のような状況に応じて動的にコミットメント戦略を設定することが必要であると考える．

(1) マラソンをした直後の状況では，“すぐに”「喉の乾きを潤したい」ので，サブゴールとして「ソーダ水を飲む」というプランを *open-mind* なコミットメント戦略で持続する意図として形成する．この意図を実施するための実践的なサブプランとして「冷蔵庫からソーダを取り出す」または「ソーダ水をコンビニまで買いに行く」を推論し，はじめに「冷蔵庫からソーダを取り出す」を選択して実行したが，冷蔵庫にはソーダ水がないのでこのサブゴールは失敗する．次に「ソーダ水をコンビニまで買いに行く」というサブプランは，“すぐに”「喉の乾きを潤したい」を満足しないので実行に移さず却下し，当初の「ソーダ水を飲む」というサブゴールを捨てる．次のサブゴールとして「水を飲む」というサブプランを *open-mind* なコミットメント戦略で持続する意図として形成し，その意図を実行するのが合理的である．

一方，(2) 恋人が訪ねてきている状況では，冷蔵庫にソーダ水がなければ「ソーダ水をコンビニまで買いに行く」というサブプランを実行して“必ず”「ソーダ水を飲む」という意図を持続させる *single-mind* なコミットメント戦略を選択するのが合理的である．

## 4 拡張 BDI アーキテクチャ

本章では，前章の BDI アーキテクチャの問題点を解決するために，プラン本体の拡張，意図形成条件の追加，および，動的コミットメント戦略を提案し，それらの実現手

```

Type: call-for-proposal
Invocation: g-add(:propose(M, 提案内容))
Precondition: true
Add List: {done(提案内容)}
Body:
  :propose(M, 提案内容){done(propose)}
  (:accept-proposal ?;
   if too-late then cancel
   else :inform(done(契約内容))
  | :reject-proposal ?; fail)

```

図 1: 契約ネットプロトコルのサブプラン例

法として，拡張 BDI アーキテクチャの基本設計を記述する．なお，本稿では，Agent Communication Language として，信念，願望，意図に基づく言語行為として基本通信行為が規定された FIPA [7] を使用する．FIPA の詳細については参考文献に譲る．

### 4.1 プラン本体の拡張

本節では，言語行為を用いて遂行されるエージェント間のコミュニケーションにおいて発生する条件分岐やメッセージを待つ同期などをプラン本体に表現できるように，プラン本体を，以下の「文」を辺とする木構造に拡張する．

```

文 ::= 単文 [{add/delete-list}]
      | { 文; 文; ... 文 }
      | if 条件 then 文 [else 文]
      | (待ち条件1 ?; 文1 | ... | 待ち条件n ?; 文n)
単文 ::= 基本行為 | サブゴール
待ち条件 ::= イベント

```

上記の文定義の最後の待ち条件などを記述する文は「待機文」と呼び，イベントキューに到着した他のエージェントからのメッセージなどの外部イベントまたは内部イベントと待ち条件<sub>1~n</sub> の順に検査を行い，最初に真となった待ち条件<sub>n</sub> に対応する文<sub>n</sub> を選択して実行するという文である．

拡張したプラン本体の記述例として FIPA-contact-net [7] のサブプラン記述例を図 1 に示す．この例は，マネージャ M からの「タスク告知」に対して「入札」するサブプランである．このエージェントが「提案内容」を添付して「入札」を行い，マネージャ M からの「落札」を待つ．そして，マネージャが落札した場合で，その連絡があまりに遅いときは cancel する．そうでないときはマネージャに「終了報告」を行うプランである．また，落札されなかったときは *fail* して，このサブプランは失敗する．本論文では，FIPA の言語行為は：で始まる基本行為として記述する．

この事例を，従来の BDI アーキテクチャで記述すると，

- 「propose(M, 提案内容)」を本体にもち *done(propose)* を add list に持つプラン

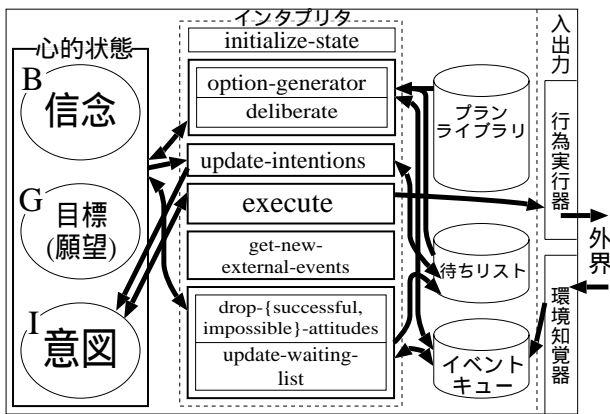


図 2: 拡張 BDI アーキテクチャ

- accept-proposal イベントが来てかつ前提条件 too-late が成立しているときに選択され、cancel を本体に持つプラン
- 同様に、前提条件  $\neg$  too-late が成立しているときに選択されて inform を本体に持つプラン
- reject-proposal イベントが来たとき選択されて fail を本体に持つプラン

の 4 つのプランに分けて記述する必要がある。これらが他のエージェントとの交渉を行う一連の言語行為であることは明示的に表現できないため、どのような意味があるのかもわかりにくい。これに対して、拡張 BDI アーキテクチャを用いてプラン本体に記述するとエージェント間の相互作用を自然に表現することができる。

以下、我々が提案する拡張 BDI アーキテクチャを図 2 に示し、プラン本体の拡張に伴う BDI アーキテクチャの実現方式を述べる。まず、BDI アーキテクチャの構成要素に、プラン本体の待機文を登録する「待ちリスト」を追加する。次に、以下の 3 個の関数をそれぞれ次のように拡張および追加する。option-generator は待ちリストに登録されている待機文の待ち条件と単一化できる外部イベントが来れば、その文をプラン本体に持つ意図を次に実行する意図の候補に加える。update-intentions はプラン本体の各文を以下のように評価する。

- 単文 (add/delete list が付いている文を含む) の場合は、その文を次に実行する文として意図 I を更新する。
- 複文の場合は、先頭の文を再帰的に評価する。
- if 文の場合は、条件の真偽に応じて、then 部/else 部のいずれかの文を再帰的に評価する。
- 待機文の場合は、単一化した待ち条件に対応する文を再帰的に評価する。そして、該当する待機文を待

ちリストから削除する。また、全ての待ち条件に単一化しない場合は何もしない。

最後に、実行中の意図の先頭に待機文が来ている場合、一回だけ待ちリストに登録する update-waiting-list を追加する。ただし、その意図が達成不能になっている場合は、待ちリストから該当する待機文を削除する。

## 4.2 意図形成条件の追加

他のエージェントからのメッセージ、自らの環境、および、心的状態などの状況にマッチする目標やその未来指向的意図をエージェント自身の判断で形成するために、プランに、意図形成条件 (FormCondition) を追加する。意図形成条件は、BDI logic を用いて表現し、意図形成の処理は 4.4 節の option-generator で行う。

## 4.3 動的コミットメント戦略

本節では、エージェントの置かれた状況や心的状態に応じて、意図の持続性のタイプを動的に決定するコミットメント戦略の実現方式を提案する。

### 4.3.1 コミットメント戦略の決定メカニズム

本稿では、エージェントの心的状態や自らが置かれた外部状況に応じて、形成する意図をどのコミットメント戦略で持続するのが合理的であるかという決定メカニズムについては今後の課題とする。たとえば、自然言語処理をベースとして、マラソンを終えた直後の状況では、“すぐに”「喉の乾きを潤したい」という目標の“すぐに”に注目して、この目標を達成するために形成する意図は blind で持続し、サブプラン (手段) は、“すぐに”実現できるサブゴールならどれでも良いと解釈して、open-mind で持続するように決定するようなメカニズムを考える。

### 4.3.2 戦略決定の単位と設定方式

コミットメント戦略は、プランおよびサブプラン単位に形成される意図を単位に決定する。そこで、図 3 のように、プランおよびサブプランに対して形成される意図スタックの中に、コミットメント戦略の属性を追加して、戦略を設定する。図 3 は、恋人が訪ねてきている状況で、冷蔵庫にソーダ水がなく、「ソーダ水をコンビニまで買いに行く」というサブプランを実行しようとしている例である。「ソーダ水を飲む」のコミットメント戦略には single-mind が、「ソーダ水をコンビニまで買いに行く」のコミットメント戦略には open-mind が、それぞれ設定されている。

また、コミットメント戦略の持続の処理は、次節の drop-successful-attitudes, drop-impossible-attitudes において述べる。

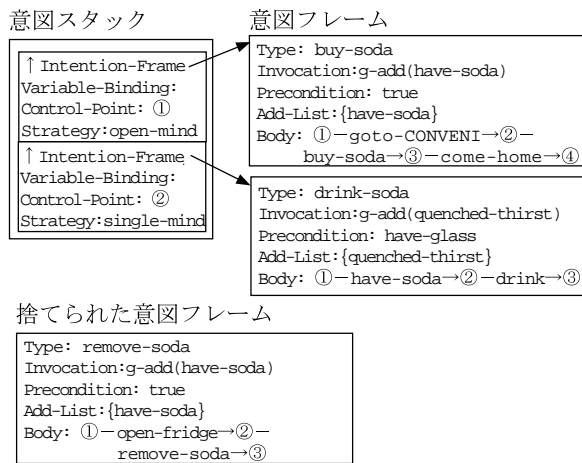


図 3: 意図スタックと意図フレーム

## 4.4 拡張 BDI アーキテクチャの基本設計

本節では、BDI アーキテクチャ [2] をベースに変更・拡張した「拡張 BDI アーキテクチャ」の基本設計を提案する。

心的状態  $B, G, I$  やプランの意図形成条件、前提条件、add/delete list、および、プラン本体の if 文の条件などは BDI logic を用いて表現する。

拡張 BDI アーキテクチャのインタプリタは、以下のループ構造とする。

```
initialize-state();
do
  options := option-generator(event-queue,
                              waiting-list, B, G, I);
  selected-option := deliberate(options,
                                B, G, I);
  update-intentions(selected-option,
                    B, G, I, waiting-list);
  execute(I);
  get-new-external-events();
  drop-successful-attitudes(B, G, I);
  drop-impossible-attitudes(B, G, I);
  update-waiting-list(I, waiting-list);
until quit.
```

### 4.4.1 option-generator(意図の形成)

option-generator は、event-queue のイベントを解析して、g-add イベント（目標の追加）の場合は、その目標を  $G$  に追加する。そして、連想条件とマッチし、さらに、意図形成条件が  $B$  から真と推論できるプランを未来指向的意図として  $I$  に形成する。また、waiting-list に登録された待機文の待ち条件と単一化できるイベントの場合は、その文を本体に持つ意図を次に実行可能な現在指向的意図 [1] の候補として options リストに追加する。さらに、

上記および既に形成されている未来指向的意図の前提条件が  $B$  から真と推論できる意図を options リストに追加する。

### 4.4.2 deliberate(意図の選択)

deliberate は、 $B, G, I$  やプランを選択するためのプランであるメタレベルプラン [2] を使って option-generator で選択された options リストの中から次に実行する意図を一つ selected-option に選択する。ただし、選択基準は個々のアプリケーションに依存する。

### 4.4.3 update-intentions(意図の更新)

update-intentions は、deliberate で選択された意図の本体から 4.1 節で述べた評価方法を用いて、次に実行すべき基本行為またはサブゴールを決定し、本体の次に実行すべき行為を指し示すように、その意図の制御情報を更新する。そして、waiting-list に登録された待機文の待ち条件を使用した場合は、その待機文を、waiting-list から消去する。また、次に実行すべき本体がサブゴールの場合は、このサブゴールを g-add 内部イベントとして event-queue に加え、このサブゴールが終了するまで、選択された意図を  $I$  中の意図スタックにプッシュしておく。

また、update-intentions では、4.3.1 節および 4.3.2 節にて述べた動的コミットメント戦略の決定とそれに伴う意図の更新を行う。

### 4.4.4 execute(意図の実行)

execute は、update-intentions で更新された制御情報を使って、次に実行する意図の本体が基本行為であれば、その基本行為を実行する。サブゴールであれば何もしない。

### 4.4.5 get-new-external-events

get-new-external-events は、環境知覚器に、event-queue への外部イベントの書き込み命令を発行する。

### 4.4.6 drop-\*-attitudes(心的状態の更新)

drop-successfull-attitudes および drop-impossible-attitudes は、達成・撤回したと信じる目標および対応する意図を、それぞれ  $G$  および  $I$  から取り除く。目標の達成・撤回は、意図  $I$  に形成されている全ての意図に対して設定されているコミットメント戦略が 2 節の blind, single-mind, または、open-mind によって、それぞれ、 $BEL(\phi)$ ,  $BEL(\phi) \vee \neg BEL(EF \phi)$ ,  $BEL(\phi) \vee \neg GOAL(EF \phi)$  を証明することで、推論する。また、サブゴールが終了した時は update-intentions でプッシュした意図を意図スタックからポップする。

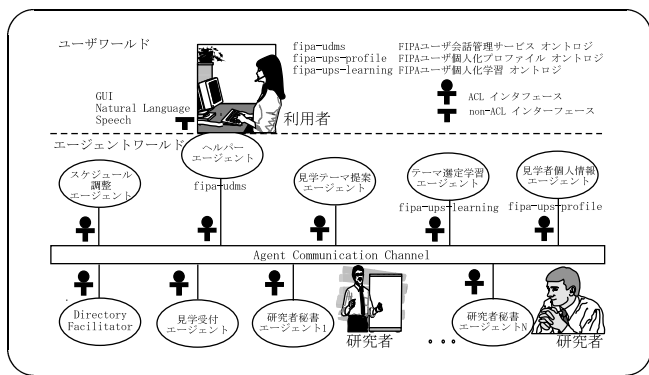


図 4: 研究所見学案内システム

#### 4.4.7 update-waiting-list

update-waiting-list は、4.1 節で述べたように待機文を waiting-lists へ登録する。

#### 4.4.8 拡張アーキテクチャの動作例

4.1 節の例文を本体に持つプランを意図として実行する拡張アーキテクチャの一連の動作概要を述べる。まず、execute において、先頭の propose を実行して、次に実行する文を次の待機文にする。そして、update-waiting-list において一回だけ待機文を待ちリストに登録して応答を待つ。ただし、その意図が達成不能になっている場合は、待ちリストから該当する待機文を削除する。次に、待ちリストの待機文に対応する他のエージェントからのメッセージが外部イベントとして到着して option-generator と deliberate により選択されると、update-intentions が待機文を評価する。たとえば、accept-proposal が来れば、too-late の真偽を判定して cancel か inform を意図 I に加え、それを execute で実行する。

## 5 応用事例

本章では、図 4 の研究所見学案内システム [8] における「研究者秘書エージェント」を事例として、4 章で提案したプラン本体の拡張が、合理的エージェントの他のエージェントとのコミュニケーションに有効であることを示す。

### 5.1 秘書エージェントのシナリオ

本節では、研究者秘書エージェントが、権利委譲をうけた研究者 A から与えられた目標を達成するために、見学テーマ提案エージェントとコミュニケーションを行う以下のようなシナリオを考える。研究者秘書エージェントは、研究者 A から「見学者が研究者の場合は彼の研究テーマである C-MAP について議論したい」という目標

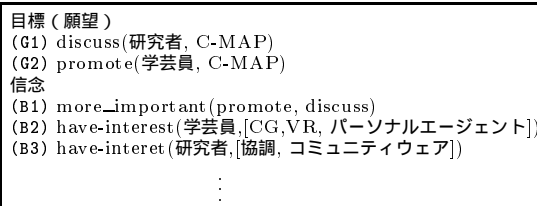


図 5: 秘書エージェントの心的状態

が与えられ、この目標を達成するために構築された対話プラン「研究者への見学対応」を意図として形成する。同様に、「博物館などの学芸員に対して、C-MAP を売り込みたい」という目標も与えられ、この目標を達成するために構築された対話プラン「学芸員への見学対応」を意図として形成する。その後、「見学テーマ提案エージェント」から、契約ネットプロトコル [9] を用いた「研究者 B に対する C-MAP 見学対応告知」があり、「研究者 B への見学対応」に「入札」して、見学テーマ提案エージェントからの「落札」を待っている。その間に、見学テーマ提案エージェントから、新たに、「学芸員 C に対する C-MAP 見学対応告知」があり、見学者の信念「議論よりは売り込みの方が重要である」に従って、研究者 B への見学対応より、学芸員 C への見学対応を優先すると判断する。そして、以前に「入札」した「研究者 B への見学対応」をキャンセルして、「学芸員 C への見学対応」に「入札」し、見学テーマ提案エージェントからの「落札」を待つ。

### 5.2 秘書エージェントの心的状態と対話プラン

秘書エージェントには、権利委譲した研究者 A が持っている信念、目標、対話プランを、図 5、図 6 のように登録する。(G1), (B1), (P3) は、それぞれ「見学者が研究者の場合は C-MAP に関する議論をしたい」という願望、「議論より売り込みの方が重要である」という信念、「見学者が研究者の場合は、C-MAP 見学対応を申し込む」という対話プランである。

### 5.3 秘書エージェントのコミュニケーション例

本節では、5.1 節のシナリオに基づいて、「秘書エージェント」の「見学テーマ提案エージェント」との拡張 BDI アーキテクチャを用いたコミュニケーション例を示す。

まず、研究者秘書エージェントは、自らに与えられた研究者 A の目標 (G1) を達成するために、対話プラン (P1) 「研究者への見学対応」を意図として形成する。そして、プラン本体のサブゴール appointment を達成するために、g-add 内部イベントを発行し、(P2) 「研究者への見学対応調整」を意図として形成する。研究者 A は病気ではないので、同様に、サブゴール propose を達成するために

```

(P1)
Type: 研究者への見学対応
Invocation: g-add(discuss(研究者, C-MAP))
Precondition: true
Add List: {EF discuss(研究者, C-MAP)}
Body:
  appointment(研究者, 日時)
  :inform(権利委譲した研究者, 研究者, 日時)
(P2)
Type: 研究者への見学対応調整
Invocation: g-add(appointment(研究者, 日時))
Precondition: ¬sick(権利委譲した研究者)
Add List: {appointment(研究者, 日時)}
Body:
  propose(研究者, C-MAP)
  if BEL(propose(研究者, C-MAP))
  then arrange(研究者, 日時)
  else fail
(P3)
Type: 研究者への見学対応交渉
Invocation: g-add(propose(研究者, C-MAP))
Precondition:
  received-cfp(見学テーマ提案AG, 研究者への対応)
Add List: {propose(研究者, C-MAP)}
Body:
  :propose(見学テーマ提案AG, 研究者への対応)
  (:accept-proposal ?;
  if ¬BEL (EF propose(研究者, C-MAP))
  then {cancel; fail}
  else :inform(見学テーマ提案AG,
               研究者向けC-MAP 概要)
  |:reject-proposal ?; fail)
(P4)
Type: 学芸員への見学対応
Invocation: g-add(promote(学芸員, C-MAP))
  :
  :
(P6)
Type: 学芸員への見学対応交渉
Invocation: g-add(propose(学芸員, C-MAP))
Precondition:
  received-cfp(見学テーマ提案AG, 学芸員への対応)
Add List: {propose(学芸員, C-MAP)}
Body:
  :propose(見学テーマ提案AG, 学芸員への対応)
  (:accept-proposal ?;
  :inform(見学テーマ提案AG, C-MAP活用方法)
  |:reject-proposal ?; fail)

```

図 6: 秘書エージェントの対話プラン例

(P3)「研究者への見学対応交渉」を意図として形成する。

同様に、与えられた目標 (G2) から対話プラン (P4)「学芸員への見学対応」、サブプラン (P5)「学芸員への見学対応調整」、サブプラン (P6)「学芸員への見学対応交渉」を順次、意図として形成する。

その後、見学テーマ提案エージェントからの見学対応告知により、連想条件にマッチするプラン (プラン記述は省略) によって received-cfp(見学テーマ提案AG, 研究者への対応) と received-cfp(見学テーマ提案AG, 学芸員への対応) が、信念に追加される。

そして、プラン (P3) の前提条件が B から真と推論でき、options リストに追加され、“研究者 B に対する C-MAP 見学対応告知” に対する入札 propose を実行する。対話プラン (P3) の待機文は、update-waiting-list によって待ちリストに書き込まれる。

その後、見学テーマ提案エージェントからの“研究者 B への C-MAP 見学対応入札を受理する” という外部イベントは待ちリスト中の対話プラン (P3) の待機文と単一化し

たので、options リストに選択される。また、プラン (P6) の前提条件が B から真と推論できるので、options リストに追加される。そして、deliberate において、「議論より売り込みの方が重要」という信念 (B1) を用いて (P6) を次に実行することを選択し、¬ (EF propose(研究者, C-MAP)) を信念に追加するようなメタレベルプラン [2] を適用する。このメタレベルプランの記述は省略する。そして、プラン (P6) の「学芸員への見学対応交渉」を実行することで学芸員 C への見学対応を入札する。

次サイクルの deliberate でプラン (P3) が選択され update-intentions で待機文の評価を行い、if 文の条件 ¬ (EF propose(研究者, C-MAP)) は B から真と推論でき、:cancel が次の execute で実行される。これらの交渉により、見学対応のダブルブッキングは回避された。

以上、秘書エージェントとテーマ提案エージェントとの合理的なコミュニケーション例を示し、拡張 BDI アーキテクチャで記述された対話プランの有効性を示した。

## 6 考察

### 6.1 拡張 BDI アーキテクチャの実装

Rao らも [2] にて、彼らが提案した BDI アーキテクチャにおいてマルチエージェントシステムの相互作用の様相を直接扱う方法を課題として取り上げている。我々の提案は、FIPA の基本通信行為やプロトコルを言語行為として、マルチエージェントシステムの相互作用をプラン本体に表現できるように BDI アーキテクチャを拡張したことで、その課題に答えた。現在、多重世界が容易に記述できる CGAEA [10] を用いて、BEL 以外の様相オペレータが記述された BDI logic の証明と動的コミットメント戦略を除いて、拡張 BDI アーキテクチャの実装を終了している。また、5 章の研究所見学案内システムを試作して、拡張 BDI アーキテクチャの検証もほぼ終了している。また、待機文に該当しない返答を受けた場合、現在の設計では副作用として蓄積され、しかも、エージェントとの相互作用も中断してしまう。そこで、形成された意図を識別記号として、返答メッセージに添付して受け、待機文に otherwise をつけるよう検討中である。

### 6.2 心的状態の整合性

拡張 BDI アーキテクチャにおける心的状態の更新は、2 章の BDI logic の心的状態の整合性公理を満足するように更新する必要がある。信念管理については、TMS [11] または ATMS [12] などの先行研究があるが、複数の様相オペレータを対象とした BDI logic には、そのままでは適用できない。ただし、2 章の (A4)(A5) の意図や目標の内省公理を用いれば、信念に関する整合性のみを対象と

すればよいと考えられるが、今後の検討を要する。

### 6.3 意図形成条件を用いた対話プランの推論

我々は、人間は合理的エージェントの一員と捉えているため、人間とのコミュニケーションを行うには、柔軟な自然言語に対応できる対話システムを各エージェント内に実現する必要がある [13]。その為、他のエージェントからの言語行為のプラン推論を行う必要がある。本稿での事例で提示したように、まず、典型的なプロトコルを種々のアプリケーションから集積し、プランとして登録することから始める。次に、これらの多くのプランの状況分析を信念として構築しプラン推論の足掛かりとする。

また、我々が提案した未来指向的意図の形成条件は、対話プラン解釈条件に該当すると考えられる。たとえば、秘書エージェントが旅行サービスを提供するとき、利用者とエージェントは「おはようございます」「おはようございます」「中津さんが出張します」「そうですか」「東京でホテルを探しているんだけど」「わかりました」...「新幹線の切符も取って欲しいのだけれど」「わかりました」などの対話が続く。「ホテルを探している」「切符を取って」などの利用者の要求が分かるまでは、ただ、文脈を蓄え、プランが決まってから、それまでの文脈を利用することになる。このシステムでは意図形成条件にて、文脈を利用者の発話を毎回チェックし、利用者の要求が理解できるかどうかのプログラムを実行する。そして、利用者要求が理解できればその要求に対するプランを意図として形成する。現在、このシステムを構築中である。

### 6.4 動的コミットメント戦略を用いた協調

動的なコミットメント戦略を使用することで、他の合理的エージェントと状況に依存した協調戦略の動的な選択に使用できるものと考えられる。たとえば、blind 戦略はどのような状況でも共同目標の達成を目指して協力する、single-mind 戦略は共同目標が実現可能だと信じている状況では協力する、open-mind 戦略は共同目標を信念などに基づいて撤回するまでは協力する、などの協調戦略を、共同目標単位に動的に選択することができる。この動的戦略を用いると、他のエージェントとの協調、妥協、拒否などに関する言語行為の選択に利用できるものと考えられる。実現方式は今後の課題である。

## 7 まとめ

本稿は、合理的エージェントの実現に Bratman [1] の意図の理論を応用して Rao と Georgeff [3, 2] が提案した BDI アーキテクチャを用いる場合、エージェント間のコミュニケーションの実現方式に関する BDI アーキテク

チャの問題点を挙げ (1) FIPA などの言語行為を用いてエージェント間のコミュニケーションをプラン本体に自然に表現できるプラン (2) プランに、将来実行するという未来指向的意図の形成条件の追加 (3) エージェントの置かれた状況や心的状態に応じて、blind, single-mind, open-mind などのコミットメント戦略を動的に選択できる意図の持続機能、などの拡張を提案した。そして、拡張 BDI アーキテクチャの有効性を応用事例で示した。今後の課題としては、コミットメント戦略の決定メカニズム、心的状態の信念管理、プロトタイプの構築・評価、プラン推論、および、他のエージェントとの協調、妥協、拒否などに関する言語行為の選択などが挙げられる。

## 参考文献

- [1] Michael E. Bratman. *Intention, Plans, and Practical Reason*. Harvard University Press, 1987. (角脇俊介, 高橋久一郎 (訳), 意図と行為 — 合理性, 計画, 実践的推論, 産業図書, 1994).
- [2] Munindar P. Singh, Anand S. Rao, and Michael P. Georgeff. Formal Methods in DAI: Logic-Based Representation and Reasoning. In *Multiagent Systems*, pp. 331–376. The MIT Press, 1999.
- [3] Anand S. Rao and Michael P. Georgeff. Modeling Rational Agents within a BDI-Architecture. In *Proc. of International Conference on Principles of Knowledge Representation and Reasoning*, pp. 473–484, 1991.
- [4] E. Allen Emerson. Temporal and Modal Logic. In *Handbook of Theoretical Computer Science*, Vol. B, pp. 997–1072. The MIT Press, 1990.
- [5] Anand S. Rao and Michael P. Georgeff. Decision Procedures for BDI Logics. *Journal of Logic and Computation*, Vol. 8, No. 3, pp. 292–343, 1998.
- [6] E. Allen Emerson and J. Srinivasan. Branching Time Temporal Logic. In J.W. de Bakker, W.P. de Roever, and G. Rozenberg, editors, *Linear Time, Branching Time and Partial Order in Logics and Models for Concurrency*, pp. 123–172. Springer-Verlag, 1989.
- [7] <http://www.fipa.org>.
- [8] 高田司郎, 五十嵐新女, 新出尚之, 榎本美香, 間瀬健二, 中津良平. マルチエージェント環境において意図的に言語行為を遂行する合理的エージェントの基本設計. SAA2000: ソフトウェアエージェントとその応用ワークショップ, pp. 125–132, 2000.
- [9] Reid G. Smith. The Contract Net Protocol: High-Level Communication and Control in a Distributed Problem Solver. *IEEE Trans. on Computers*, Vol. 29, No. 12, pp. 1104–1113, 1980.
- [10] <http://www.aist.go.jp/ETL/etl/captain/gaea/>.
- [11] Jon Doyle. A Truth Maintenance System. *Artificial Intelligence*, Vol. 12, pp. 231–272, 1979.
- [12] Johan de Kleer. An Assumption-based TMS. *Artificial Intelligence*, Vol. 28, pp. 127–162, 1986.
- [13] 高田司郎, 五十嵐新女, 中津良平. インターネット上における自然言語対話システムの分散処理について. 信学技報 NLC2000-9, pp. 63–70, 2000.