

## 7F-8 分散協調型問題解決における知識表現モデル

村田努 柴田知博 長木正宏 井内稔 山崎晴明  
山梨大学

## 1. はじめに

現在、人工知能による問題解決には主に単体で動く問題解決エージェントが用いられているが、世の中に存在する解決すべき問題には、問題解決の過程において、多種多様な側面を持つものや、複数の問題解決フェイズを持つものがある。このような性質を持つ問題を解決するためには、異なる推論の手段や知識表現を持つ複数のエージェントを用いたほうが有効であると考えられる。また、複数の計算機に処理させることによる信頼性の向上、処理の分散による負荷の軽減が期待される。そこで現在、分散協調型問題解決と呼ばれる分野の研究が盛んに行われている。<sup>1)</sup>

今回我々は、異なった知識ベースを持つ複数の問題解決エージェントの協調により問題解決を行う分散協調型問題解決システムの試作を行った。

## 2. 分散協調型問題解決システムの概要

図1に今回我々が試作したシステムの構成を示す。推論を行うエージェントに3台、通信の制御及びユーザインターフェースを担当するプロトコ

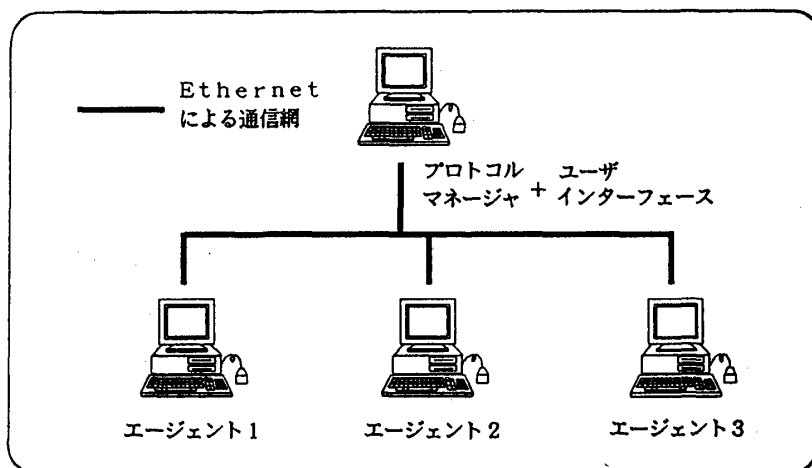


図1. システム構成図

ルマネージャに1台のPCをそれぞれ割り当てる。

本来、各エージェントは協調プロトコルを用いて他のエージェントと直接情報を交換するのであるが、現試作に於いては管理の都合上プロトコルマネージャなる集中管理機構を設けている。

エージェント群はそれぞれ異なる知識表現ベース<sup>2) 3) 4)</sup> (意味ネットワーク、プロダクション、フレーム) による推論機能を持ち、それぞれの特徴を活かした推論を実行可能である。また各々のエージェントはEthernetによりプロトコルマネージャと接続されておりプロトコルマネージャを介して他のエージェントと通信を行うことが可能である。

ユーザインターフェースでは各エージェントへの指示、及びエージェントからの応答をビジュアルに表示することが可能である。

## 3. エージェント間の協調

単一エージェントによる問題解決の方法では、そのエージェントが所有できる知識の量や分野などの制限により広範囲に及ぶ問題解決が困難となる。

複数のエージェントを用いることにより、同一問題でも異なる視点から検討できること、複数分野の知識を統合して推論することが可能になることなどにより問題解決能力が向上する。

以下では協調して推論を行うためのプロセスと、共通表現の必要性について述べる。

Knowledge representation models on Cooperative Distributed Problem Solving

Tsutomu MURATA, Tomohiro SHIBATA, Masahiro CHOKI, Minoru IUCHI, Haruaki YAMAZAKI  
Yamanashi Univ.

### 3.1. 協調動作

各エージェントはプロトコルマネージャから問題を与えられた後、以下のような動作を行う。

- ① 自力で推論可能な範囲の推論を行う。この時、定期的にプロトコルマネージャへアクセスし、自分宛ての情報があるか照会を行う。
- ② ①において自分宛ての情報があった場合、その処理を優先して実行する。その際、現在までの推論履歴と解決すべき問題を保存し、後で処理を再開できるようにしておく。
- ③ 自力で推論が不可能な状態に陥った場合、プロトコルマネージャを介して他のエージェントに問い合わせを行う。
- ④ ③において他に推論可能な問題が存在するならばその処理を実行し、なければ他のエージェントからの解答を待つ。
- ⑤ 他のエージェントからの解答があった場合、その解答を受け取ったことにより再開できる推論と、問い合わせを行った時点で新しく始めた推論とを比較し、より有益であると判断したほうの推論を行う。
- ⑥ 問題に対するゴールを見つけるまで①から⑤を繰り返し、見つけたら推論を終了する。

### 3.2. 共通表現形式

各エージェントの内部表現が異なるため、そのままでは他のエージェントとの情報の交換を正しく行うことが不可能である。そこで各エージェントが共通に理解可能な表現が必要となる。

今、共通表現形式を $R_c$ 、表現形式 $R_x$ から表現形式 $R_y$ に変換する処理系を $T_{xy}$ と表現すると、2つのエージェントA、B間の通信は、エージェントAの内部表現形式を $R_A$ 、エージェントBの内部表現形式を $R_B$ とすると、以下ようになる。

(図2)

- ① エージェントA内の変換系 $T_{Ac}$ において内部表現形式 $R_A$ を共通表現形式 $R_c$ に変換を行う。
- ② エージェントAからエージェントBへ送信する。
- ③ エージェントB内の変換系 $T_{cB}$ において共通表現形式 $R_c$ で記述されている情報を自己の内部表現形式 $R_B$ に変換する。

共通表現形式を用いることの利点は、新たにエージェントを追加する時のコストを低くおさえるこ

とが可能なことである。新たに追加するエージェントZに自己の内部表現と共通表現形式とを相互に変換できる処理系 $T_{cz}$ 、 $T_{zc}$ を用意することにより、既存のエージェントを変更する必要なくシステムに追加可能である。

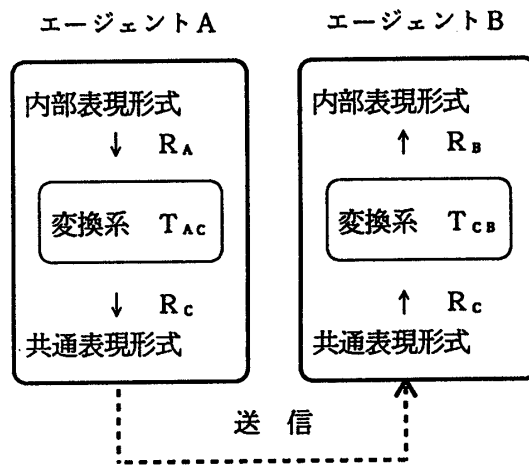


図2. 共通表現形式の必要性

### 4. 今後の課題

今回試作したシステムではエージェント間での協調は行われているが完全に分散されていない。プロトコルマネージャが通信を一元管理しているため、プロトコルマネージャに機能障害などが生じた場合、協調させることが不可能となる。またエージェント数が増加した場合、レスポンスの低下が生じる。以上のことから各エージェントに分散型のプロトコルマネージャを組み込むことが必要である。

### 参考文献

- [1] E. H. Durfee, V. R. Lesser, D. D. Corkill: "Trends in Cooperative Distributed Problem Solving", IEEE transactions on knowledge and data engineering, VOL. 1, NO. 1, March (1989).
- [2] C. F. Chabris原著, 岩谷宏訳: 「TurboCによる人工知能」, 工学社 (1990).
- [3] 小林重信: 「知識工学」, 昭晃堂 (1986).
- [4] 上野晴樹, 石塚満: 「知識の表現と利用」, オーム社 (1987).