

ATMS を用いた分散制約充足問題の解法†

横尾 真** 石田 亨**

本論文では、複数のエージェントによる協調的な問題解決である分散協調問題解決の新しいモデルを提案する。このモデルでは、問題解決を行うエージェントが仮説に基づく推論を行い、推論結果を Assumption-based Truth Maintenance System (ATMS) を用いて管理する。エージェント間では、仮説に基づく推論結果および同時に成立しえない仮説の組合せに関する情報 (nogood) が通信される。本論文では、このモデルが複数のエージェントが関係する制約充足問題 (分散制約充足問題) を解くために有効であることを示す。分散制約充足問題の一例である、通信ネットワークにおける通信路のプランニング問題に対するこのモデルの適用について述べる。分散制約充足問題の解法として、全体の解を取りまとめるエージェントが存在する階層的な解法と、取りまとめるエージェントが存在しない非階層的な解法の2種類を想定する。それぞれの解法について、ATMS の適用方法を示す。また、ATMS の適用により、階層的な解法では部分解の間の制約条件を求める処理が不要となること、非階層的な解法では不適切な資源の割当てが回避されることにより、効率的に分散制約充足問題を解くことが可能になることを示す。

1. ま え が き

分散協調問題解決とは複数の自律的なエージェントによる協調的な問題解決である。これまでに分散協調問題解決システムを構築するためのいくつかのツール、言語等が提案されている^{7),8),10)}。本研究の目的は、これらのツール、言語等に付加可能な、分散協調問題解決における基礎的なメカニズムを提供することである。本稿では、複数のエージェントが関係する制約充足問題 (分散制約充足問題) を解くことが、分散協調問題解決のさまざまな問題において必要であると考へ、ATMS (Assumption-based Truth Maintenance System)³⁾ を分散協調問題解決へ適用することを検討する。

ATMS は問題解決システム (Problem-Solver) が仮説 (Assumption) を用いた推論を行う際に、推論結果の管理を行うメカニズムである。ATMS は制約を満足する解の探索を必要とする問題に有効であり、設計、診断、計画問題などの多くの分野への適用が検討されている^{5),12)}。

本稿では次のような分散協調問題解決のモデルを提案する。

モデル: 複数のエージェントが独立に ATMS を持ち、各エージェントは仮説に基づく推論を行い、互いにメッセージを通信しながら問題解決を行う (図 1)。

本稿では以下、ATMS および ATMS を用いた分散協調問題解決のモデルについて説明する。さらに、

分散制約充足問題の一例として、通信ネットワークにおける通信路のプランニング問題に対するこのモデルの適用について述べる。

一般に分散制約充足問題の解法として、全体の解を取りまとめるエージェントが存在し、他のエージェントは問題の一部分を解き、取りまとめるエージェントが部分解を統合して全体の解を求める階層的な解法と、取りまとめるエージェントを置かず、各エージェントが対等に通信し合い、全体として合意のとれた状態になるように調整し合う非階層的な解法の2種類が存在する。それぞれの解法について ATMS の適用方法を示す。階層的な解法では、ATMS の適用によってエージェントが部分解の間の制約条件を求める処理が不要になり、全体の処理が効率化されることを示す。非階層的な解法では、ATMS の適用によってエージェントが不適切な資源の割当てを防ぎ、効率的に分散制約充足問題を解くことが可能になることを示す。

さらに、階層型の問題解決における nogood の通信の削減方法、非階層型の問題解決の課題について考察する。

2. ATMS を用いた分散協調問題解決のモデル

2.1 ATMS

ATMS は次の3つによって特徴付けられる。

- 複数の環境 (environment) を持つことによる多重世界機能
- 環境間でのデータの共有による重複した推論の回避

† Solving Distributed Constraint Satisfaction Problems Using the ATMS by MAKOTO YOKOO and TORU ISHIDA (NTT Communications and Information Processing Laboratories).

** NTT 情報通信処理研究所

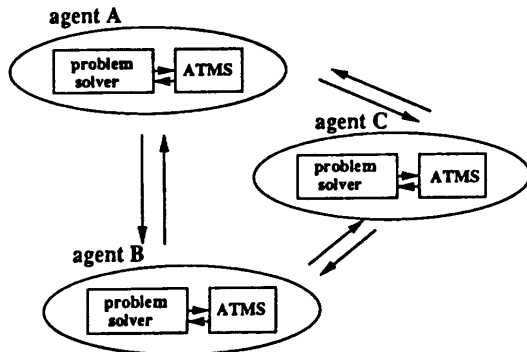


図 1 分散協調問題解決における ATMS の利用方法
Fig. 1 Using the ATMS in DAI problems.

● 矛盾の効率的な回避

以下、それぞれについて説明する。

ATMS は複数の環境を管理する。1つの環境はいくつかの仮説の組合せであり、仮説の組合せの包含関係 (サブセット, スーパーセットの関係) によってラティスを作る (図 2)。データ α が環境 E の仮説 $\{H_1, H_2, \dots, H_n\}$ から導かれるとき、データ α は環境 E で成立するという。1つの環境は、その仮説の組がすべて真となるような1つの世界に対応すると考えられる。

データ α が環境 E で成立するとき、 α は E のスーパーセットとなるような環境でも同様に成立する。このため、データ α が成立する、サブセット、スーパーセットの関係にある複数の環境は、そのうちの最も小さい環境によって代表させることができる。ATMS はすべてのデータに、そのデータが成立する極小の環境の集合を関連付ける。この環境の集合をラベルと呼ぶ。ラベルが環境の集合であるのは、同じデータが互いに包含関係にない仮説の組から導かれるためである。データが成立する環境をラベルという形で表現することにより、環境間でのデータの共有が実現される。

データ α がいくつかのデータ $\beta_1, \beta_2, \dots, \beta_n$ を前提として導かれるとき、 α のラベルは $\beta_1, \beta_2, \dots, \beta_n$ のラベルから計算される。すなわち、 α がどの世界で成立するかは、前提が成立する世界によって決定される。このデータ間の依存関係、 $\beta_1, \beta_2, \dots, \beta_n \rightarrow \alpha$ をジャスティフィケーションと呼ぶ。環境間でデータが共有されるため、異なる環境で重複した推論を行うことが回避される。

環境 E で矛盾が生じた場合、 E は **nogood** (制約条件違反) として登録される。 E のスーパーセットであるすべての環境も同様に **nogood** となる。**nogood** となった環境はすべてのデータのラベルから取り除かれる。このことは矛盾によって以前の推論結果が取り消

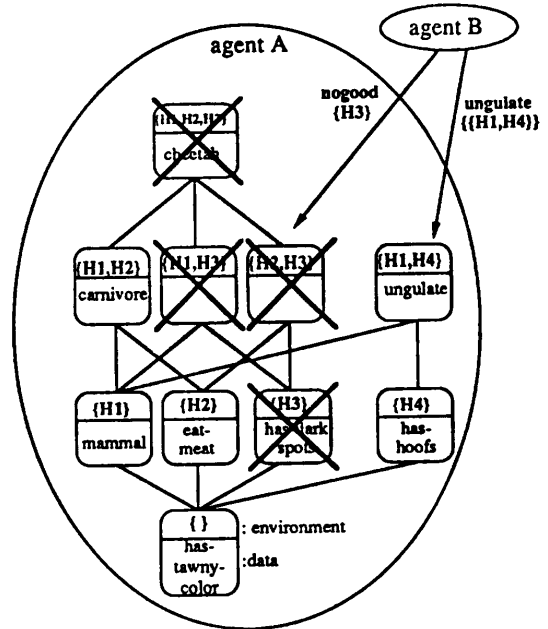


図 2 環境のラティスとエージェント間通信
Fig. 2 Environment lattice and communication between agents.

されることを意味する。問題解決システムは **nogood** となった環境では推論を行わない。

2.2 モデル

本稿で提案する分散協調問題解決のモデル* では、複数のエージェントがそれぞれ独立に ATMS を持ち、以下の情報を交換し合う。これらの情報は各エージェントの管理する ATMS の環境に影響を与える。

推論結果とそのラベル: 各エージェントの推論結果は、ラベルが付けられて他のエージェントに送信される。推論結果が送信され、その推論結果の成立する環境が受信側のエージェントにとって未知である場合には、適切な新しい環境が作られる。図 2 で、ある動物が **ungulate** (有蹄類) であるという推論結果が送信され、受信側のエージェントがその推論結果の成立する環境 $\{H_1, H_4\}$ を管理していなかった場合、新たに環境 $\{H_1, H_4\}$ が作られる。

nogood: **nogood** が他のエージェントから送信された場合、図 2 で、**nogood** $\{H_3\}$ が送信された場合、この **nogood** のスーパーセットの環境、すなわち $\{H_1, H_3\}$, $\{H_2, H_3\}$, $\{H_1, H_2, H_3\}$ は **nogood** となる。

ATMS を用いた推論を行い、仮説に基づく推論結果と **nogood** を互いに通信し合うことにより、各エー

* 同様のモデルが文献 9) によって提案され、解釈型のエキスパートシステムへの適用が検討されている。本研究は文献 9) の研究とは独立に進められたものである。

ジェントは、他のエージェントの推論結果を利用し、かつ2.1節のATMSの特徴を生かして推論を行うことが可能になる。本論文では以下、仮説に基づく推論結果とnogoodを互いに通信し合うことにより、エージェント間で得られる効果について述べる。仮説に基づく推論結果とnogoodを互いに通信し合うことにより期待される効果は、複数のエージェント間で矛盾を効率的に回避することである*。

3. 分散プランニング問題の概要

本章では、前章で示したモデルの通信ネットワークにおける通信路のプランニング問題への適用について述べる。図3に通信ネットワークの例を示す。このネットワークは、複数のサイト、各サイトを結ぶ通信リンクによって構成される。各サイトはいくつかの地域的に分割された領域に分かれており、それぞれの領域には独立なエージェントが存在し、領域内のサイトのコントロールを行う。各エージェントは、自分の領域内に関してのみ詳しい情報を持っている。

このような通信ネットワークにおいて、エージェントの共同作業により、通信路のプランニング、トラヒックの制御を行う研究が文献1), 2)などで行われている。本稿では、複数のエージェントによる通信路のプランニング問題を想定する。目的は、要求に応じてサイトとサイトを結ぶ通信路を確保することである。例え

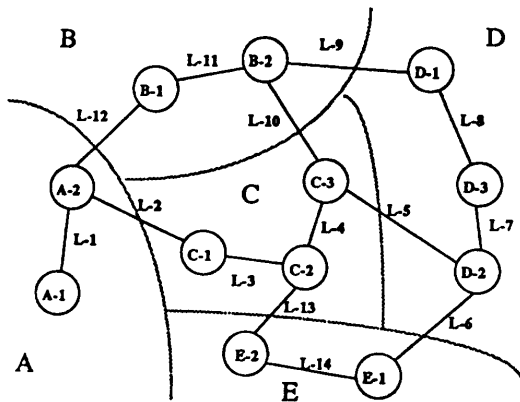


図3 通信ネットワークの例

Fig. 3 Example of communication network.

*2.1節に述べたように、ATMSのもうひとつの重要な効果として、重複した推論の回避が存在する。一方、分散協調問題解決における重要な問題として、他のエージェントが既に行った処理の重複を避けることがある。しかしながら、ATMSの利用は、直接的にはこの問題の答とはなっていない。他のエージェントが既に行った処理の重複を避けるためには、エージェント間での推論結果の共有が行われるように推論結果の通信がなされなければならない。また、異なるエージェントが同時に同じ処理をしないようにするためには、推論のプランの交換等が必要となる。

表1 プランの断片
Table 1 Plan fragments.

Agent	Goal	Plan fragment	Resource used
A	goal-1	1A	L-1, L-2
		2A	L-1, L-12
	goal-2	3A	L-12
B	goal-1	1B	L-10, L-11, L-12
	goal-2	2B	L-10, L-11, L-12
C	goal-1	1C	L-2, L-3, L-4, L-5
		2C	L-5, L-10
	goal-2	3C	L-5, L-10
		4C	L-4, L-10, L-13
D	goal-1	1D	L-5, L-7, L-8
	goal-2	2D	L-5, L-6
E	goal-2	1E	L-6
		2E	L-13, L-14

表2 プランの候補
Table 2 Alternative plans.

goal-1	
goal-1/plan-1	1A, 1C, 1D
goal-1/plan-2	2A, 1B, 2C, 1D
goal-2	
goal-2/plan-1	3A, 2B, 3C, 2D, 1E
goal-2/plan-2	3A, 2B, 4C, 2E

ば、図3のA-1とD-1を接続したいという要求(以下goal-1と呼ぶ)、A-2とE-1を接続したいという要求(以下goal-2と呼ぶ)の2つに対して、goal-1, goal-2を共に満足させるような、各要求に対する通信リンクの割当てを決定する。各通信リンクの収容可能な通信路の数には制限がある。

この問題を解くためには、まず各エージェントでそれぞれのゴールを達成するプランの一部(プランの断片と呼ぶ)を作成し、その後すべてのゴールを満足するような各エージェントのプランの断片の組合せを求めればよい。表1に各エージェントの求めるプランの断片の例を示す。例えば、エージェントBはgoal-1に関して、Aとの境界からCとの境界に達するためのプランの断片1Bを立てる。表2にプランの断片によって構成される全体としてのプランを示す。全体として、goal-1, goal-2を達成するのにそれぞれ2つのオルターナティブが存在する。

この問題は、通信路を確保するという要求(複数存在する)に対して、要求を満たし、かつ資源の制限を満足するように各エージェントの持つ資源の割当てを決定する問題であり、制約充足問題の一種である。制約充足問題を解くことはプランニング問題のみでなく、設計、診断などのさまざまな問題で必要とされる。

制約充足問題は次のように定式化される。変数の有限集合 $I = \{X_1, X_2, \dots, X_n\}$ および制約の集合が存在し、変数はそれぞれ値を値域 D_1, D_2, \dots, D_n からとる。 I 中の k 変数間の制約 $c(X_{i_1}, \dots, X_{i_k})$ は、直積 $D_{i_1} \times \dots \times D_{i_k}$ の部分集合であり、互いに整合のとれた変数の値の組を表す。制約充足問題の解を求めることは、すべての制約を満たす変数の値の組を求めることである。分散制約充足問題とは、制約充足問題の変数、制約に関する知識が複数のエージェントに分散されている制約充足問題である。

一般に分散制約充足問題を解くのに次の2つのアプローチが存在する。

階層的解法: 全体の解を取りまとめるエージェントを置き、その他のエージェントは問題の一部分を解く。取りまとめるエージェントは部分解を統合して全体の解を求める。

非階層的解法: 全体の解を取りまとめるエージェントを置かず、各エージェントが対等に通信し合い、全体として合意がとれた状態になるように調整し合う。

通信路のプランニング問題では、各エージェントがプランの断片を求め、そのプランの断片を1つのエージェントに集めて最終的な解を導くのが階層的な解法であり、取りまとめるエージェントを置かず全体としてゴールを満足するようなプランの断片の選択を決定するのが非階層的な解法である。

これらの2つの解法にはトレードオフが存在する。一般に階層的な解法の方が非階層的な解法よりも簡単である。階層的な解法では、問題のうち局所的な知識のみを必要とする処理を各エージェントに割り当て、その他の処理は取りまとめるエージェントで行えばよい。非階層的な解法で制約充足問題を解くアルゴリズムで、解が存在する場合に必ず解が求められることを保証するものはまだ提案されていない。一方、階層的な解法では取りまとめるエージェントの負荷、エージェント間の通信量が大きくなるのが問題となる。

以下、それぞれのアプローチについて、ATMS を用いた解法、ATMS を用いることによる利点、課題について述べる。

4. 階層的な解法

4.1 概要

階層的な解法では全体の解を取りまとめるエージェントを置き、その他のエージェントは問題の一部分を解く。前述の2つの要求 goal-1, goal-2 に対して、取

りまとめるエージェントでこの要求を副問題に分割して、各エージェントに割り当てる。取りまとめるエージェントでは各領域のおおまかな接続状況だけを認識しており、goal-1 は A-C-D か A-B-C-D という経路で、goal-2 は A-B-C-D-E か A-B-C-E という経路で接続するというプランを立て、各エージェントにこれらの経路を実現する詳細なプラン(プランの断片)を立てさせる。各エージェントはプランの断片およびそのプランの断片に関する制約条件を取りまとめるエージェントに送信する。エージェントBは、回線の制約から、goal-1 のためのプランの断片 1B と、goal-2 のためのプランの断片 2B は両立できないという情報を送信する。取りまとめるエージェントでは、各エージェントから送られたプランの断片を組み合わせて、全体として制約条件を満たす解を求める。

4.2 ATMS の適用方法

ATMS の適用方法は次のとおりである。

- あるゴールのために、ある資源が利用可能であるということ仮説とおく。
- プランの断片に、そのプランの断片が成立するための資源に関する条件をラベルとして付加する。
- 資源の制限による制約を *nogood* として記述する。

Cは goal-1 のために L-2, L-3, L-4, L-5 をつなぐというプランの断片 1C を立てるが、このプランの断片はこれらの資源(回線)が利用可能であるという仮説のもとに成立している。goal-1 のために L-2 が使用可能であるという仮説を H_{L2-g1} と書くことにすれば、この 1C のラベルは $\{H_{L2-g1}, H_{L3-g1}, H_{L4-g1}, H_{L5-g1}\}$ となる。資源の制約から L-5 が1つのゴールにしか利用できないとすれば、*nogood* $\{H_{L5-g1}, H_{L5-g2}\}$ が取りまとめるエージェントに送信される。

ATMS を利用することの利点は次のとおりである。

- *nogood* で表現される本質的な制約が通信されるため、本質的な制約から導かれる副次的な制約を求める必要がない。

具体的には、本質的な制約とは資源の制約であり、副次的な制約とは、資源の制約から導かれる、部分解(プランの断片)の間の制約である。

分散協調問題解決の特徴は、他のエージェントで得られた推論結果をもとに、さらに推論が行われる点である。他のエージェントで得られた推論結果を利用するために、推論結果の成立する条件等の通信が必要な場合が存在する。

具体的には、取りまとめのエージェントは部分問題を解くエージェントで得られた部分解を統合して全体の解を求める。その際に、取りまとめのエージェントには、部分解のみでなく、部分解の間の制約条件が通信されることが必要である。

ATMS を用いない場合、各エージェントは、資源の制約を用いて、部分解の間の制約を求め、取りまとめのエージェントに送信しなければならない。例えば、エージェントCでは goal-1 のためのプランの断片 1C, 2C と goal-2 のためのプランの断片 3C, 4C の組合せ 1C-3C, 1C-4C, 2C-3C, 2C-4C が、資源の制約を満たしているかどうかをチェックする。制約を満たしていなければその情報を、 $\text{conflict}\{1C, 3C\}$ のように表現し、取りまとめのエージェントに送信する。同様な処理がエージェントA, B, Dでも必要であり、合計8通りの組合せのチェックが各エージェントで必要である。L-11, L-5 が1つの通信路しか収容できないとすると、 $\text{conflict}\{1B, 2B\}$, $\text{conflict}\{1C, 3C\}$, $\text{conflict}\{2C, 3C\}$, $\text{conflict}\{1D, 2D\}$ が組合せのチェック中に発見され、取りまとめのエージェントに送られる。

ATMS を用いる場合、部分解はそのラベルと共に送信され、取りまとめのエージェントに、*nogood* で表現される資源の制約が送信される。例えば、取りまとめのエージェントには *nogood* $\{H_{L11-g1}, H_{L11-c2}\}$, *nogood* $\{H_{L5-g1}, H_{L5-c2}\}$ が送られる。この *nogood* は、部分解の間の制約を求めるために必要な本質的な制約を表している。ATMS を用いる場合には具体的なプランの断片の組合せのチェックを行う必要はなく、前述の8通りの組合せのチェックが不要になる。

5. 非階層的な解法

5.1 概要

階層的な解法では、局所的な知識のみを用いてはできない処理は、取りまとめのエージェントに情報を集中させることによって対処できるが、非階層的な解法では全体から情報を集めるエージェントが存在しないため、一般に階層的な解法よりも難しい。非階層的な解法の1つとして、マルチステージネゴシエーションが文献2)で提案されている。この方法は制約充足問題を一種の探索問題とみなし、複数のエージェントによる探索のためのエージェント間のプロトコルおよび各エージェントにおける処理の手順を定めたものである。

表2に示すような全体のプランの一部分を各エー

ジェントが認識している状態を仮定する。これは各エージェントが部分問題を解き終えた状態に相当する。この状態で、各エージェントは次の情報のみを認識している。

1. 自分の持つプランの断片
2. 隣接するエージェントの持つプランの断片と自分のプランの断片との接続関係

例えばエージェントAは、goal-2 に関して唯一のプランの断片 3A のみを認識しており、それがBの持つプランの断片 2B と接続されることを認識している。しかしながら、エージェント全体として2つの可能なプランがあることは認識していない。

マルチステージネゴシエーションは、取りまとめのエージェントなしで複数のゴールを満足するプランの断片の組合せを求めるための、エージェント間通信の枠組を定めたものであり、次のように記述される。

Step 1: ゴールとなっている通信の起点であるエージェントが行動を開始し、可能なプランの1つに対して一時的な資源の割当てを行う。

Step 2: エージェントは他のエージェントに、自分のプランの断片の選択が他のエージェントと整合が取れていることの確認を要求する。

Step 3: 各エージェントはメッセージキューを調べる。メッセージは確認要求および確認要求に対する返答である。

Step 4: 各エージェントは確認要求および確認要求に対する返答を調べ、これらを反映して改めて一時的な資源の割当てを行う。一時的な資源の割当てを変更した場合、その変更を反映したメッセージを関連するエージェントに送信する。確認要求が満足できない場合は、できない原因となった制約条件を、確認要求を送信したエージェントに通信する。

Step 5: 3~4 の処理を収束するまで繰り返す。

注意すべきことは、マルチステージネゴシエーションは一般的な通信の枠組のみを定めたものであり、資源の割当て方法、制約条件の表現方法、通信された制約条件の利用方法等については定められていないことである。本論文では以下、ATMS を用いたマルチステージネゴシエーションの具体化の方法、その利点について述べる。

5.2 ATMS の適用方法

次のような方法で ATMS を利用する。

- ゴールおよび各エージェントの持つプランの断片

のオルターナティブを仮説とする。

- プランの断片の接続関係をジャスティフィケーションで表現する。
- プランの断片にラベルを付ける。
- 一時的な資源の割当ては、環境を選択することによって行われる。選択された環境で成立するプランの断片に対して資源の割当てを行う。
- 環境はエージェントの認識しているすべてのゴールを含み、かつ nogood でないものが選択される。

例えば、表2に示す全体のプランの一部分を各エージェントが認識している状態で、エージェントAは2つのゴールがあり、goal-1には2つのオルターナティブがあることを認識している。エージェントAはそれぞれのゴールに対応する仮説 H_{G1} , H_{G2} およびゴールを達成するためのオルターナティブに対応する仮説 H_{1A} , H_{2A} を持つ。プランの断片 1A のラベルは $\{H_{G1}, H_{1A}\}$ に、2A のラベルは $\{H_{G1}, H_{2A}\}$, 3A のラベルは $\{H_{G2}\}$ となる。また、choose $\{H_{1A}, H_{2A}\}$ という宣言をする。この宣言の意味は、仮説 H_{1A}, H_{2A} のどちらか1つが必ず成立するという意味である。詳細は文献4)を参照されたい。

AとBの間の、ATMSを用いたネゴシエーションの経過のトレースを表3に示す。以下、ネゴシエーションの経過の説明を行う。

Stage 1: エージェントAはgoal-1を達成するための2つのオルターナティブから2Aを選択し、環境 $\{H_{G1}, H_{G2}, H_{2A}\}$ を選択する (Step 1)。2Aと3Aに対して一時的な資源の割当てを行い、Bに対して確認要求を出す (Step 2)。

Stage 2: Bにおいて、2A, 3Aはそれぞれ1B, 2Bが達成されることを要求する。1B, 2Bは資源の制限より同時に満足することはできないため、nogood $\{H_{G1}, H_{G2}, H_{2A}\}$ が得られる。1Bのラベルは $\{H_{G1}, H_{2A}\}$ であるため、 H_{G1}, H_{G2} を含み、かつ1Bが成立する環境はnogoodであるため選択できない。これは、1Bを満足させることにすると、goal-1, goal-2の両方を満足させることが不可能であることを意味する。Bは環境 $\{H_{G1}, H_{G2}\}$ を選択し、Aに返答を送信する (Step 3, 4)。

Stage 3: Aではnogood $\{H_{G1}, H_{G2}, H_{2A}\}$ とchoose $\{H_{1A}, H_{2A}\}$ から、ハイパリゾレーションルール⁴⁾を適用することにより、1Aのラベルを $\{H_{G1}$,

表3 ネゴシエーション中の通信の概要
Table 3 Summary of communication in negotiation.

A	B
2A $\{H_{G1}, H_{2A}\}$; OK? → B	
3A $\{H_{G2}\}$; OK? → B	
	A → OK? 2A $\{H_{G1}, H_{2A}\}$ A → OK? 3A $\{H_{G2}\}$
	2A implies 1B 3A implies 2B 1B ∧ 2B → conflict
	nogood $\{H_{G1}, H_{G2}, H_{2A}\}$ → A 3A is OK → A
B → nogood $\{H_{G1}, H_{G2}, H_{2A}\}$	
B → 3A is OK 1A $\{H_{G1}, H_{1A}, H_{G2}\}$; OK? → C	

$H_{1A}\}$ から $\{H_{G1}, H_{G2}\}, \{H_{G1}, H_{1A}\}$ に変更する*。このラベルの変更により、1Aはgoal-1, goal-2の両方を達成するためには必要不可欠であることが表現されている。Aは環境 $\{H_{G1}, H_{G2}\}$ を選択し、1A, 3Aに資源の割当てを行い、Cに確認要求を出す (Step 3, 4)。

マルチステージネゴシエーションにATMSを用いることの利点は以下のようにまとめられる。

- 確認要求、制約条件がゴール、オルターナティブを表す仮説によって表現されるため、他のゴールの実現を妨げる、不適切な資源の割当てを防ぐことができる。

例えば、エージェントBで1B, 2Bのどちらかを選択する際に、ATMSを用いない場合には、判断を行う明確な基準が存在しないため、1Bを選択する可能性があり、1Bを選択した場合、後で適切なバックト

*詳細には、ここで用いているハイパリゾレーションルールは、H6ルールと呼ばれるものであり、choose $\{A1, A2, \dots, Ai, \dots\}$ で、 β のラベルが λ であるとき、すべての*i*に対して、 $\{Ai\} \cup \alpha_i$ がnogoodもしくは $\{Ai\} \cup \alpha_i \in \lambda$ で、 $A_j, j \neq i \notin \alpha_i$ が成立する際に、 β のラベルを $\{U\alpha_i\} \cup \lambda^*$ に変更するものである。ただし λ^* は λ から、 $\{U\alpha_i\}$ のスーパーセットを削除したものである。すなわち、choose $\{H_{1A}, H_{2A}\}$ で1Aのラベルが $\{H_{G1}, H_{1A}\}$ であるとき、 $\{H_{1A}\} \cup \{H_{G1}\} \in \{\{H_{G1}, H_{1A}\}\}$ であり、 $\{H_{2A}\} \cup \{H_{G1}, H_{G2}\}$ がnogoodであるため、1Aのラベルが $\{H_{G1}, H_{G2}\}, \{H_{G1}, H_{1A}\}$ に変更される。

ラック等の処理が行われないと, goal-2 を達成できなくなってしまう. ATMS を用いることで, 2B を達成することが goal-2 の達成のために必要不可欠であることが通信され, 1B を選択する可能性が排除される.

別の可能性として, B が 2B を選択したとする. ATMS を用いない場合には, A に対して 1B を選択しなかった原因として, $2A \wedge 2B \rightarrow \text{conflict}$ が送信される. この情報のみでは, エージェント A は, 2A が goal-2 と両立できないこと, 1A が 2つのゴールを達成するために必要不可欠であること等は結論できない. ATMS とハイパゾリューションルールの適用によって, 1A が 2つのゴールを達成するために必要不可欠であることが, ラベルの変更によって表現される.

6. 考 察

本章では ATMS の分散制約充足問題への適用に関する考察を行う. まず, 階層型の問題解決で問題となる nogood の通信の削減について述べ, 次に, 非階層型の問題解決でのマルチステージネゴシエーションの課題について述べる.

6.1 nogood の通信の削減

本節では, 階層的な解法における nogood の通信の削減方法について述べる. 非階層的な解法では, 本論文で提案する ATMS を用いたマルチステージネゴシエーションにより, エージェント間での nogood の通信方法が詳細に決定される. 一方, 階層的な解法では, 部分問題を解くエージェントで得られた nogood が取りまとめのエージェントに送信されるが, 正しい解を得るためには必ずしもすべての nogood が送信される必要はない. 取りまとめのエージェントが処理のボトルネックになることを避けるためには, 必要な nogood のみが送信されることが望ましい.

本節では, データ依存グラフ^{11),13)}を用いて, 各エージェントが行う推論の内容を表現し, このデータ依存グラフを解析することにより, 不必要な nogood の通信を削減する方法を示す. データ依存グラフは次のプリミティブからなる有向グラフである.

Aノード: 仮説の集合

Dノード: 仮説以外のデータの集合

Jノード: ジャスティフィケーションの集合

JノードからDノードへの有向枝: Jノードの要素がDノードの要素を結論として導くことを示す.

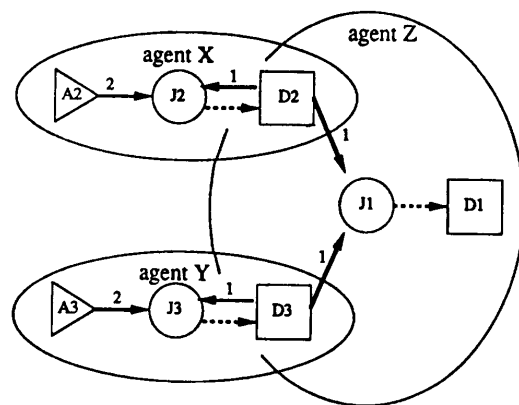
Aノード, DノードからJノードへの有向枝: J

ノードの要素がA, Dノードの要素を前提として用いることを示す. この有向枝はJノードの要素のジャスティフィケーションが参照するA, Dノードの要素の個数の最大値(最大参照データ数)でラベル付けされる.

仮説, 仮説以外のデータ, ジャスティフィケーションは, それぞれいくつかの共通要素を持たない集合に分類され, それらの集合にノードが割り当てられる. 集合の選び方は任意である. ただし, 集合の選び方によって後述する解析の処理量と精度が異なる(集合が小さい方が精度は良いが処理量は大きくなる). 一例として, 1つの推論ルールによって作られるジャスティフィケーションの集合, ルールの条件部, 結論部に対応する仮説, 仮説以外のデータの集合に対して, それぞれ1つのノードを割り当てることが考えられる.

各ノードは1つないし複数のエージェントに属する. Dノード, Jノードは, その要素の少なくとも1つを持つエージェントに属し, Aノードはその要素の少なくとも1つを, 他のエージェントから推論結果の通信を受ける以前に持つエージェントに属する. 図4に, データ依存グラフの例を示す. データ依存グラフはシステム設計者によって与えられることも, エージェントの共同作業によって作成することも可能である.

次の条件が成立すれば, エージェントXで得られた nogood N は, エージェントXが以前にNおよびNのスーパーセットをラベルを持つ推論結果を他のエージェ



▷: A-node, a set of assumptions

□: D-node, a set of non-assumption data

○: J-node, a set of justifications

→: addition of data by the element of J-node

⇨: reference of data by the element of J-node and maximal number of data references

図4 データ依存グラフの例

Fig. 4 Example of data dependency graph.

ントに送信していない場合に、他のエージェントに対して通信が不要になる。

条件1: *nogood* N はエージェント X に局所的な A ノード A_{x1}, A_{x2}, \dots の要素によって構成される。エージェント X に局所的な A ノードとは、X のみに属する A ノードである。

条件2: 他のエージェントに属する任意の J ノード J_i において、 J_i が参照する D, A ノードのうち、 A_{x1}, A_{x2}, \dots から到達可能なものは高々1つであり、かつその参照を示す有向枝の最大参照データ数が1である。

これらの条件により、N および N のスーパーセットの環境が、他のエージェントで作られないことが導かれる。図4でエージェント X の導く任意の *nogood* N は、X に局所的な A ノード A2 の要素から構成される。また、A2 から到達可能なノードを参照する J ノードは J1 のみであり、その参照するノードで A2 から到達可能なものは D2 のみで、参照を示す有向枝の最大参照データ数が1であるため条件1, 2が満たされる。エージェント Y で導かれる *nogood* についても同様である。

LAN で結合された3台の Symbolics 3640 上で、データ依存グラフが図4で表現されるエージェントの構成を用いて、6 queens を3つのエージェントによって解いた。*nogood* の通信の削減を行う場合と行わない場合の比較を行った結果、部分問題を解くエージェントで得られるすべての *nogood* の通信が削減され(合計 92 個)、解を得るために要する時間が 86% に減少した。この問題のように、各エージェントが唯一の部分問題を解き、共有される仮説がない場合には、*nogood* の通信の削減効果が大きい。

6.2 マルチステージネゴシエーションの課題

非階層型の問題解決での、マルチステージネゴシエーションによる解法は完全ではない。すなわち、すべてのゴールを達成できる解が存在する場合でも解が見つけられない場合があり、また、制約が強すぎて解が存在しない場合でも、解が存在しないことを発見できない場合がある。これらは、各エージェントが不適切な資源の割当てを行い、かつ、その割当てが不適切であることが ATMS を用いたマルチステージネゴシエーションによって認識できないという場合である。

ATMS の利用によって、各エージェントは、単独で他のゴールの実行を不可能にする資源の割当てを行わないことが保証される。しかしながら、複数のエー

表4 プランの候補 (解が発見できない例)
Table 4 Alternative plans. (A solution can not be found.)

goal-1	
goal-1/plan-1	1A, 1F
goal-1/plan-2	2A, 1D, 2F
goal-2	
goal-2/plan-1	1B, 2D, 1G
goal-2/plan-2	2B, 1E, 2G
goal-3	
goal-3/plan-1	1C, 2E, 1H
goal-3/plan-2	2C, 2H

ジェントの資源の割当ての組合せにより、他のゴールの実行が不可能になる場合が存在する。

例えば、8つのエージェント間で3つのゴールが存在し、表4のようなプランの断片が存在すると仮定する。エージェント A が 2A, D が 1D, C が 1C, E が 2E を選択し、D, E で $1D \wedge 2D \rightarrow \text{conflict}$, $1E \wedge 2E \rightarrow \text{conflict}$ であるとする、B で goal-2 が達成できなくなる。1D, 2E のラベルをそれぞれ $\{H_{C1}, H_{2A}\}$, $\{H_{C3}, H_{1C}\}$ とすると、B が2つの可能なプランが達成できないことから、*nogood* $\{H_{C1}, H_{C2}, H_{C3}, H_{2A}, H_{1C}\}$ が得られる。エージェント A, C のどちらかがプランの断片のオルターナティブを選択すれば、3つのゴールを達成できるが、*nogood* $\{H_{C1}, H_{C2}, H_{C3}, H_{2A}, H_{1C}\}$ は、エージェント A, C の選択が単独で不適切であることは導かない。この状態を避けるためには、エージェント A-C 間で何らかの通信により調整を行う必要がある。

完全性を保証するようにマルチステージネゴシエーションを改良することは今後の課題である。

7. む す び

本稿では分散協調問題解決の新しいモデルとして、問題解決を行うエージェントが仮説に基づく推論を行い、仮説に基づく推論結果を ATMS を用いて管理し、エージェント間で仮説に基づく推論結果および *nogood* を通信し合うというモデルを提案した。さらに、分散制約充足問題の一例として、通信ネットワークにおける通信路のプランニング問題に対するこのモデルの適用について述べた。全体の解を取りまとめるエージェントが存在する階層的な解法と、取りまとめるエージェントが存在しない非階層的な解法について、ATMS の適用方法を示した。ATMS の適用によって階層的な解法では部分解の間の制約条件を求める処理が不要になり、非階層的な解法では不適切な資源

の割当てを防ぐことによって、それぞれ効率的に分散制約充足問題を解くことが可能になることを示した。

今後の課題として、非階層的な解法で、解が存在する場合に必ず解を求めることができるアルゴリズムについて検討を行っている。

謝辞 本研究の機会を与えて頂いた NTT 情報通信処理研究所の中野良平主幹研究員に感謝します。また、ATMS を提供して頂き、熱心に御討論頂いた同研究所の桑原和宏研究主任に感謝します。

参 考 文 献

- 1) Adler, M. R., Davis, A. B., Weihmayer, R. and Worrest, R. W.: Conflict-Resolution Strategies for Non-Hierarchical Distributed Agents, in USC Computer Research Institute Technical Report CRI-88-41 (Gasser, L. ed.), University of Southern California (1988).
- 2) Conry, S. E., Meyer, R. E. and Lesser, V. R.: Multistage Negotiation in Distributed Planning, in *Readings in Distributed Artificial Intelligence* (Bond, A. H. and Gasser, L. eds.), pp. 367-384, Morgan Kaufman (1988).
- 3) de Kleer, J.: An Assumption-based TMS, *Artif. Intell.*, Vol. 28, pp. 127-162 (1986).
- 4) de Kleer, J.: Extending the ATMS, *Artif. Intell.*, Vol. 28, pp. 163-196 (1986).
- 5) de Kleer, J. and Williams, B.: Diagnosing Multiple Faults, *Artif. Intell.*, Vol. 32, pp. 97-129 (1987).
- 6) Durfee, E. H. and Lesser, V. R.: Using Partial Global Plans to Coordinate Distributed Problem Solvers, *Proc. IJCAI-87*, pp. 875-883 (1987).
- 7) Gasser, L., Braganza, C. and Herman, N.: Implementing Distributed AI Systems Using MACE, *Proc. CAIA-87*, pp. 315-320 (1987).
- 8) Lesser, V. R. and Corkill, D. D.: The Distributed Vehicle Monitoring Testbed: A Tool for Investigating Distributed Problem Solving Networks, *AI Magazine*, Vol. 4, No. 3, pp. 15-33 (1983).
- 9) Mason, C. and Johnson, R.: DATMS: A

Framework for Distributed Assumption-based Reasoning, in USC Computer Research Institute Technical Report CRI-88-41 (Gasser, L. ed.), University of Southern California (1988).

- 10) Yonezawa, A. and Tokoro, M. (eds.): *Object-Oriented Concurrent Programming*, MIT Press (1986).
- 11) 石田: プロダクションシステムの並列実行可能性の解析, 電子情報通信学会論文誌, Vol. J71-D, No. 3, pp. 567-576 (1988).
- 12) 石塚, 松田: 仮説推論システムの拡張知識表現と概念学習機構, 人工知能学会誌, Vol. 3, No. 1, pp. 94-102 (1988).
- 13) 横尾, 石田: 分散協調問題解決における ATMS の利用, 第 2 回人工知能学会全国大会, pp. 141-144 (1988).

(平成元年 2 月 1 日受付)

(平成元年 10 月 11 日採録)



横尾 真 (正会員)

1962年生。1984年東京大学工学部電子工学科卒業。1986年同大学院修士課程修了。同年NTTに入社。現在NTT情報通信処理研究所、知識処理研究部勤務。知識処理に関する研究に従事。分散AI、制約充足問題等に興味を持つ。人工知能学会、ソフトウェア科学会各会員。



石田 亨 (正会員)

昭和28年生。昭和51年京都大学工学部情報工学科卒業。昭和53年同大学院修士課程修了。同年日本電信電話公社に入社。昭和58~59年、コロンビア大学客員研究員。現在、NTT情報通信処理研究所知識処理研究部勤務。工学博士。プログラミング言語、知識処理、並列処理、分散処理に興味を持つ。人工知能学会、ソフトウェア科学会、AAAI各会員。