

マルチステージネゴシエーションにおけるゴール間競合の検出†

桑原和宏††* VICTOR R. LESSER†††

分散ネットワークにおいて複数のエージェントが資源の割当を協調して行うマルチステージネゴシエーションにおけるグローバルなゴール間の競合の検出手法を提案する。マルチステージネゴシエーションではどのエージェントも完全なグローバルな情報を持たず、各エージェントは自分の所における局所的な資源の割当を他のエージェントに通信し、お互いにその影響を交換しあうことによって全体として整合のとれた資源の割当を行う。この時、局所的な影響をどのように表現し通信したらよいか問題となる。特に資源の割当の制約が強すぎてすべてのグローバルゴールを満たすことができない場合がでてくるため、エージェント間で交換する情報に基づいてグローバルゴール間の競合関係が正しく検出できるようにする必要がある。ここではマルチステージネゴシエーションの扱う問題の定式化を与える。そこではグローバルゴールはいくつかのグローバルプランを持ち、そのうちの一つを実行することによりそのグローバルゴールは満足される。各エージェント内での資源割当における選択を基にしてグローバルプランを表現し、それに基づいて各エージェント内での競合情報を表現し、エージェント間で交換することによって、グローバルゴール間の競合を正しく認識できるようにした。さらにその競合情報はグローバルゴールを満足させる解の探索にも有効である。

1. はじめに

複数のエージェント間での協調を実現するためのモデルの一つとしてネゴシエーションがある。Durfeeらは協調問題解決におけるネゴシエーションを“関連する情報の秩序だった交換を通して、矛盾や不確実性を減少させることによって、共通の物の見方またはプランの合意の程度を改善すること”⁵⁾と定義している。このようなネゴシエーションの枠組として契約ネットプロトコル^{4), 8), 9)}, マルチステージネゴシエーション²⁾などが提案されている。

契約ネットプロトコルは複数のエージェント間でタスクの割当を実現するものである。具体的にはタスクのアナウンス、入札、落札という形でタスクをエージェント間に割り当てる。一方、マルチステージネゴシエーションでは局所的な情報をエージェント間で交換しあうことによってエージェント間で協調的に資源の割当を行う。契約ネットプロトコルでは一般に各エージェントは自分の独自の判断で入札、落札を行う。したがってタスクの割当に関してお互いにグローバルな制約がある場合、全体として整合性のとれた割当を実現するのが難しい。これに対し、マルチステージネゴ

シエーションでは局所的な決定の影響を繰り返して交換するため、数多くの分散している資源割当においてもその割当の間でグローバルな制約がある場合においても全体として適当な割当を実現することができる。

文献 2) ではマルチステージネゴシエーションの概念的な提案がされている。各エージェントは他のエージェントに対して自分の局所的な選択の是非を問い合わせ、その問い合わせに対して自エージェントにおけるその選択の影響を返答する。このようなメッセージの交換を通して各エージェントはお互いに満足する選択に到達するという枠組が示されている。そこでは単に枠組が提案されただけで、その前提となる条件、対象としている問題が明確には定義されてはいなかった。また、ネゴシエーションの過程で実際にどのような形式で情報を交換すべきかも十分には明らかになっていなかった。

その後の文献³⁾ではエージェントの局所的な選択がグローバルにどのような影響を与えるか、その評価の手法を提案している。そこではエージェント内における局所的な選択とグローバルなゴールとの競合関係を Exclusion Set という形で表し、それをエージェント間で交換することによってゴール間の競合の検出を試みている。しかし、そこで提案された手法では一つのゴールを満足するのにグローバルにみて複数の方法(後述するグローバルプランに相当)がありうることを考慮に入れていないため、ゴール間の競合関係が正しく認識できない場合がある⁷⁾。

本論文ではまず、文献 2) で提案されているマルチ

† Goal Conflict Detection in the Multistage Negotiation by KAZUHIRO KUWABARA (NTT Communications and Information Processing Laboratories) and VICTOR R. LESSER (University of Massachusetts at Amherst).

†† NTT 情報通信処理研究所

††† マサチューセッツ大学アムハースト校

* 現在 NTT コミュニケーション科学研究所

NTT Communication Science Laboratories

ステージネゴシエーションの概念を基にして、その扱っている問題の定式化を与え、マルチステージネゴシエーションがどのような条件の下で何を解こうとしているかを明確にする。次にその定式化に基づいてどのような情報をエージェントが交換すべきかを明らかにする。そして、その交換された情報を用いてどのようにゴール間の競合を検出するかを述べる。

2. マルチステージネゴシエーション

2.1 探索空間

マルチステージネゴシエーションにおいては図1に示すようにグローバルなゴールがいくつか存在する。グローバルゴールはいわば全体として解くべき目標である。グローバルゴール g_i はグローバルプラン $gp_{i,j}$ の集合を持つ。グローバルプランとはグローバルゴールを満足するためのエージェントにまたがる仮想的なプランである。グローバルプランはエージェントが局所的に解けるいくつかのサブゴールに分解される。ここで $sg_{i,j,k}$ をグローバルプラン $gp_{i,j}$ に対する k 番目のサブゴールとする。それぞれのサブゴールはある唯一のエージェントに属している。一つのサブゴールがいくつかのグローバルプランに同時に属する場合もある。エージェントはそのサブゴールを満足するためのいくつかの方法、すなわちプランフラグメント $pf_{i,j,k,l}$ を持つ。プランフラグメントはある唯一のサブゴールに属するものとする。

これらの関係は論理表現を用いると次のようになる。

$$g_i \Leftrightarrow \bigvee_j gp_{i,j}$$

$$gp_{i,j} \Leftrightarrow \bigwedge_k sg_{i,j,k}$$

$$sg_{i,j,k} \Leftrightarrow \bigvee_l pf_{i,j,k,l}$$

ここで $g_i, gp_{i,j}, sg_{i,j,k}, pf_{i,j,k,l}$ はそれぞれが満足される、または選択される時に真、そうでない時に偽の値をとるものとする。

2.2 エージェントの情報

各エージェントは自分のところのサブゴールとそれを満たすプランフラグメントの集合、また、そのサブゴールのグローバルゴールは知っているもののそのサブゴールがどのグローバルプランに属するかは知らないという状況を考える。これはエージェントの集合全体としてのゴールが定まっていて、各エージェントがそのゴールに対してどのようなことをすべきかがわかっているが、しかし、エージェントの間の関係をどのように取って全体としてうまくゴールを満足できるかがわかっていないという場合と考えることができる。

このためエージェントは自分のサブゴールがどのグローバルプランに属するかを求める必要がでてくる。

2.3 制約

ここでは資源の割当に関し、エージェント内に閉じた資源使用の制約（エージェント内制約）、エージェントにまたがった資源使用の制約（エージェント間制約）の二種類の制約を考える。

- エージェント内制約：プランフラグメントを実行するためには一定の資源を確保する必要があり、資源の量には限りがあることから同時に選択できるプランフラグメントの組には制限がでてくる。具体的にはプランフラグメントの集合 PF が同時に選択できるためには

$$\sum_{pf_i \in PF} needs(pf_i, r_j) \leq copies(r_j) \text{ (for all } j \text{)}$$

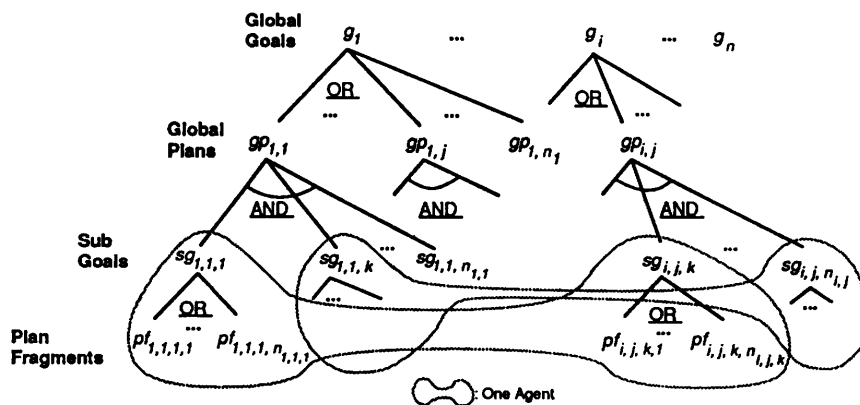


図1 マルチステージネゴシエーションにおける探索空間
Fig. 1 Global search space in the multistage negotiation.

が成立する必要がある。ここで $needs(pf_i, r_j)$ は pf_i の実行に必要な資源 r_j の数、また、 $copies(r_j)$ は資源 r_j のエージェントにおいて使用可能な数を表す。

- **エージェント間制約**: 複数のエージェントに関係する資源使用の制約であり、例えばエージェント A で資源 ra が使用される時、 B では資源 rb が使用されなければいけないというようなものである。ここで自エージェントとエージェント間制約があるようなエージェントを関連するエージェントと呼ぶ。エージェントはすべての自分と関連するエージェントを知っているものとする。

2.4 起動エージェント

グローバルゴール g_i は、その g_i の解の探索を開始するエージェント（これを起動エージェントと呼ぶ）を一つ持っているとする。起動エージェントは g_i を *primary goal (p-goal)* として扱い、その解の探索の責任を持つ。また、起動エージェントには g_i のすべてのグローバルプランに関してそれに属するサブゴールが一つ以上含まれているという仮定をおく。したがって g_i を満足するためには起動エージェントにおいてなんらかのサブゴールが満足されなければならないことになる。

2.5 ネゴシエーション途中の問題設定の変更

ここではネゴシエーションの途中においては問題の設定条件が変化しないという仮定をおく。すなわち、途中でエージェントが追加削除されたり、グローバルゴールが追加削除されたりすることはないとしている。

2.6 例題

マルチステージネゴシエーションを通信ネットワークに応用した例を示す。この応用ではグローバルゴールはある地点からある地点への通信路を設定することに相当する。エージェントにはそれぞれある地域が割り当てられ、その地域内の通信路の設定に責任を持つ。資源はエージェント地域内の中継点 (Station) または地域間の境界点の間を結ぶ通信路のパスに対応する。また、サブゴールは各エージェント内における地域間の境界点から他の境界点（または、最終目的の通信路の始点ないし

は終点）の間を結ぶ通信路の設定に相当する。いくつかのサブゴールが接続された最終目的の始点と終点間を結ぶグローバルな通信路がグローバルプランに相当する。

ここでのエージェント内制約とは各通信路の容量の制約に基づく制約である。通信路という性格から地域間の境界点に達する通信路はその境界点の両側でそれぞれ通信路が割り当てられる必要がある。すなわち、エージェント間における資源の割当の制約で、これがエージェント間制約になる。

図2に $g1, g2, g3$ の三つのグローバルゴールがある通信ネットワークの例⁶⁾を示す。ここでは八つのエージェント A, B, C, D, E, F, G, H がある。それぞれのエージェントは各々の地域の通信路の設定を行う。 $g1, g2, g3$ の起動エージェントはそれぞれ A, B, C である。ここで $g1$ とはエージェント A から F に到達する通信路を求めるもので、また、 $g2$ はエージェント B から G に到達する通信路を求めるもので、また、 $g3$ はエージェント C から H に到達する通信路を求めるものである。

グローバルプランとそれを構成するサブゴールを表1に示す。この例の場合、 $g1$ に対して一つ、 $g2, g3$ に対してそれぞれ二つのグローバルプランがある。それぞれのエージェントは表2に示すような資源、サブゴールおよびそのプランフラグメントを持っている。表には各プランフラグメントが必要とする資源の数もあわせて示してある。この例では一つのサブゴ

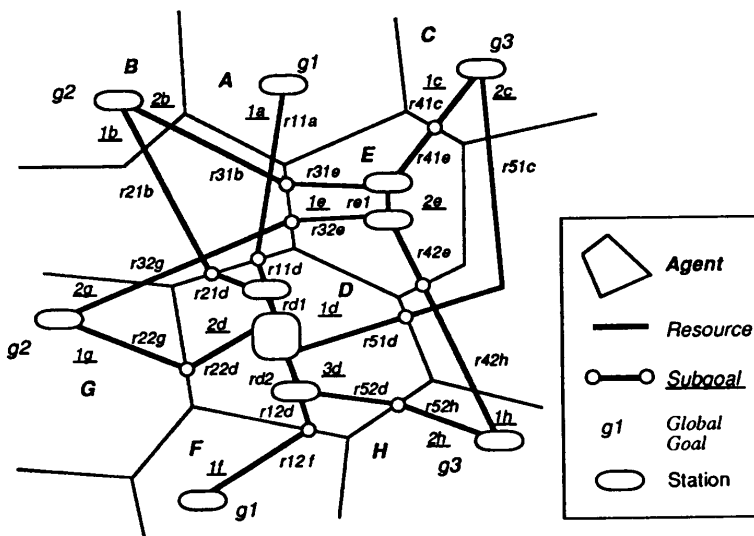


図2 例における通信ネットワーク
Fig. 2 Communication network example.

ルに対し一つのプランフラグメントが存在している。
この例では次のようなエージェント内制約がある。

- エージェント *D*: プランフラグメント *p1d* と *p2d* はともに資源 *rd1* を使用するため同時には選択できない。また、プランフラグメント *p1d* と *p3d* はともに資源 *rd2* を使用するため同時には選択できない。
- エージェント *E*: プランフラグメント *p1e* と *p2e* はともに資源 *re1* を使用するため同時には選択できない。

ここでのエージェント間制約は同時に使用しなけれ

ばならない資源の組があるということである。この例では *r11a* と *r11d*, *r21b* と *r21d*, *r31b* と *r31e*,

表 1 例におけるグローバルプラン
Table 1 Global plans in the example.

Global goal	Global plan	Subgoals
<i>g1</i>	<i>gp11</i>	<i>1a-1d-1f</i>
<i>g2</i>	<i>gp21</i> <i>gp22</i>	<i>1b-2d-1g</i> <i>2b-1e-2g</i>
<i>g3</i>	<i>gp31</i> <i>gp32</i>	<i>1c-2e-1h</i> <i>2c-3d-2h</i>

表 2 例における各エージェントの資源, サブゴール, プランフラグメント
Table 2 Resources, subgoals, and plan fragments in the example.

Agent A				Agent F			
Goal	Subgoal	pf	<i>r11a</i>	Goal	Subgoal	pf	<i>r12f</i>
Resource count			1	Resource count			1
<i>g1</i>	<i>1a</i>	<i>p1a</i>	1	<i>g1</i>	<i>1f</i>	<i>p1f</i>	1

Agent B					Agent G				
Goal	Subgoal	pf	<i>r21b</i>	<i>r31b</i>	Goal	Subgoal	pf	<i>r22g</i>	<i>r32g</i>
Resource count			1	1	Resource count			1	1
<i>g2</i>	<i>1b</i>	<i>p1b</i>	1		<i>g2</i>	<i>1g</i>	<i>p1g</i>	1	
	<i>2b</i>	<i>p2b</i>		1		<i>2g</i>	<i>p2g</i>		1

Agent C					Agent H				
Goal	Subgoal	pf	<i>r41c</i>	<i>r51c</i>	Goal	Subgoal	pf	<i>r42h</i>	<i>r52h</i>
Resource count			1	1	Resource count			1	1
<i>g3</i>	<i>1c</i>	<i>p1c</i>	1		<i>g3</i>	<i>1h</i>	<i>p1h</i>	1	
	<i>2c</i>	<i>p2c</i>		1		<i>2h</i>	<i>p2h</i>		1

Agent D										
Goal	Subgoal	pf	<i>r11d</i>	<i>r21d</i>	<i>r51d</i>	<i>rd1</i>	<i>rd2</i>	<i>r12d</i>	<i>r22d</i>	<i>r52d</i>
Resource count			1	1	1	1	1	1	1	1
<i>g1</i>	<i>1d</i>	<i>p1d</i>	1			1	1	1		
<i>g2</i>	<i>2d</i>	<i>p2d</i>		1		1			1	
<i>g3</i>	<i>3d</i>	<i>p3d</i>			1		1			1

Agent E							
Goal	Subgoal	pf	<i>r31e</i>	<i>r41e</i>	<i>re1</i>	<i>r32e</i>	<i>r42e</i>
Resource count			1	1	1	1	1
<i>g2</i>	<i>1e</i>	<i>p1e</i>	1		1	1	
<i>g3</i>	<i>2e</i>	<i>p2e</i>		1	1		1

$r41c$ と $r41e$, $r51c$ と $r51d$, $r12d$ と $r12f$, $r22d$ と $r22g$, $r52d$ と $r52h$, $r32e$ と $r32g$, $r42e$ と $r42h$ の資源の組はそれぞれ同時に使用しなければならない。

3. 競合関係の表現

前章においてマルチステージネゴシエーションが扱っている問題を定義した。ネゴシエーションを進めていく上でグローバルゴール、グローバルプラン、サブゴール間の競合関係を各エージェントが認識することが必要である。本章では競合関係の表現形式を述べ、さらに次章でどのように競合関係を計算するかを述べる。

3.1 Nogood Goal Set

グローバルゴールが複数存在する時、問題の制約が強すぎてあるグローバルゴールを満足させると他のグローバルゴールを満足させられないという場合が出てくる。このようなグローバルゴール間の関係を示したものが Nogood Goal Set である。Nogood Goal Set を論理表現を用いて次のように表現する。

$$\bigvee \neg q_i$$

例えば

$$\neg q_1 \vee (\neg q_2 \wedge \neg q_3)$$

は q_1 をあきらめるかまたは q_2 および q_3 を同時にあきらめる必要があることを示す。最終的には Nogood Goal Set が求められればグローバルゴール間の関係がわかり、例えば制約が強すぎるような場合でもどのグローバルゴールをあきらめればよいかかわかる。もし、同時にすべてのグローバルゴールが満足できるのであれば Nogood Goal Set は空となる。

グローバルゴールを満足させるにはそのグローバルプランのうち一つを満足させればよい。したがって Nogood Goal Set を求めるためにはグローバルプランのレベルでの関係を調べる必要がある。また、このようなグローバルプランに関する情報はエージェントの間に分散して存在するのでエージェントの間で情報を交換する必要がある。

3.2 Goal Exclusion Set

起動エージェントは自分の p-goal に関して他のグローバルゴールのグローバルプランとの相互関係を求め、それを交換しあうことで Nogood Goal Set を求めることができる。ここでグローバルゴール q_i と競合関係にあるグローバルプラン $qp_{j,k}$ ($j \neq i$) を Goal Exclusion Set と呼ぶ。この競合関係の意味は次のよ

うになる。

$$\neg q_i \vee (\bigvee_{j \neq i} \neg qp_{j,k})$$

Goal Exclusion Set は起動エージェントの間で交換され、この情報をもとに起動エージェントは Nogood Goal Set を計算する。

3.3 Exclusion Set

グローバルゴールの Goal Exclusion Set はそのグローバルゴールに含まれるサブゴールの Exclusion Set から計算される。このサブゴール $sq_{i,j,k}$ の Exclusion Set とはそのサブゴールと競合関係にあるグローバルプランの集合である。この競合関係の意味は次のようになる。

$$\neg sq_{i,j,k} \vee (\bigvee_{l \neq i} \neg qp_{l,m})$$

Exclusion Set はサブゴールに対してそれと競合関係にあるグローバルプランを表したものであり、その競合は自分のエージェントの中で発生するものと他のエージェントにおける競合によって起こるものとがある。それを区別して表す時は Exclusion Set をさらに細かく次のように分類する。まず自分の中に閉じた競合だけに基づく Exclusion Set を Local Exclusion Set と呼ぶ。これに対して他のエージェントにおける競合が直接の原因になるものも含めて考える時は Induced Exclusion Set と呼ぶ。

4. Nogood Goal Set の計算

前章で定義した Nogood Goal Set を求める手法を以下説明する。マルチステージネゴシエーションにおいて各エージェントはどのサブゴールを選択したかを関連するエージェントに送信する。ここでどのサブゴールを選択したかを通信する際にその選択したサブゴールのグローバルプランに関する情報（後述する Choice List）を付け加える。エージェントはその情報を基に自分のサブゴールの Local Exclusion Set を計算する。さらに、自エージェントのサブゴールの Local Exclusion Set から相手のエージェントの選択したサブゴールの Exclusion Set の情報を計算し、それを相手のエージェントに通知する。

4.1 Conflict Set

サブゴールの Local Exclusion Set を求めるために、そのサブゴールとは同時には選択できないサブゴールの集合（これを Conflict Set と呼ぶ）を考える。まず、その準備としてプランフラグメントの集合 $P_k = \{pf_i\}$ は以下の条件が成り立つ時 compatible

と定義する*。

$$\sum_{P_i \in P_k} needs(p_i, r_j) \leq copies(r_j) \text{ (for all } j)$$

また, SG_A をエージェント A のサブゴールの集合とする。サブゴール sg_i のグローバルゴールを $goal(sg_i)$ で表す。さらにサブゴール sg_i のプランフラグメントの集合を $P(sg_i)$ とする。

ここで SG_A の部分集合で, その要素のサブゴールのグローバルゴールがすべて相異なる集合, SSG_k を考える。すなわち,

$$SSG_k = \{sg_i \mid goal(sg_i) \neq goal(sg_j) \text{ for all } i, j\}$$

SSG_k が compatible とは SSG_k のサブゴールのすべてを満たす compatible なプランフラグメントの集合が存在することと定義する。すなわち,

$$SSG_k \text{ が compatible} \Leftrightarrow$$

$$\exists P_i [P_i \text{ が compatible}$$

$$\wedge \forall sg_i \in SSG_k [(P(sg_i) \cap P_i) \neq \emptyset]]$$

論理表現を用いると $SSG_k = \{sg_1, sg_2, \dots, sg_n\}$ が compatible でないということは次のように表せる。

$$\neg \left(\bigwedge_{sg_i \in SSG_k} sg_i \right)$$

ここで, サブゴール sg_i を含み, かつ, compatible でないすべての SSG_k の集合, SS_{sg_i} を考える。すなわち,

$$SS_{sg_i} = \{SSG_k \mid sg_i \in SSG_k$$

$$\wedge SSG_k \text{ が compatible でない}\}$$

この集合の意味は

$$\bigwedge_{SSG_k \in SS_{sg_i}} \neg \left(\bigwedge_{sg_j \in SSG_k} sg_j \right)$$

である。 $SSG_k \in SS_{sg_i}$ を含んでいるのでこの式は次のように変形できる。

$$\neg sg_i \vee \left(\bigwedge_{SSG_k \in SS_{sg_i}} \bigvee_{sg_j \in SSG_k, j \neq i} (\neg sg_j) \right)$$

sg_i の Conflict Set, CS_{sg_i} は

$$\left(\bigwedge_{SSG_k \in SS_{sg_i}} \bigvee_{sg_j \in SSG_k, j \neq i} (\neg sg_j) \right)$$

をサブゴールの否定の選言標準形に変換した形で表現する。

4.2 グローバルプランの表現

Conflict Set から Local Exclusion Set を計算するためにはサブゴールがどのグローバルプランに属しているかを記述できるようにする必要がある。ここではサブゴールの性質として Choice List というものを導入し, グローバルゴールと Choice List の組で

サブゴールの属するグローバルプラン (場合によっては複数の場合もある) を表現する。サブゴール sg_i の Choice List を $cl(sg_i)$ と表し, sg_i の属するグローバルプランを次のように書く。

$$\langle goal(sg_i), cl(sg_i) \rangle$$

この記述をここではゴール記述と呼ぶ。

エージェントはどのようなグローバルプランがあるかはあらかじめ知らず, Choice List をネゴシエーションの過程で段階的に作成することによってどのようなグローバルプランがあるかを認識できるようになる。

4.2.1 Choice List の構成要素

Choice List は各エージェントにおけるいくつかあるサブゴールのうちどのサブゴールを選択したかを表したもので, 具体的には *local-id* の選言標準形で表される。*local-id* とはそれぞれのエージェントでどのサブゴールを選んだかという選択を表すものである。ここでは起動エージェントにおける選択と起動エージェントでないエージェントにおける選択とを区別して考える。

起動エージェントにおける選択を表す *local-id* は次のような形式で表される。

$$(agent \ index \ total-number)$$

ここで *agent* は起動エージェント, $index (1 \leq index \leq total-number)$ はその選択が全部で何通りかある選択枝のうち何番目のものかを表し, そして *total-number* は選択枝の全体の数を表す。 *total-number* が含まれているのはどのエージェントにおいてもすべての選択枝と競合しているかを容易に識別できるようにするためである。もし, エージェントの選択枝が一つしかない場合は *index* と *total-number* は省略され, 単に

$$(agent)$$

と書かれる。

一方, 起動エージェント以外のエージェントにおける選択を表す *local-id* は次のような形式で表される。

$$(entry-id \ index \ total-number)$$

ここで *entry-id* はエージェントの内部テーブルのエントリを示す ID である。この内部テーブルとは他のエージェントから送られてくる選択の確認のメッセージに対して, 自エージェント内で選択するサブゴールの候補を得るのに用いられるものである。このメッセージには選択されたサブゴールとサブゴールを選択した時に割り当てられた資源のうちエージェント間制約に関係する資源が含まれている。内部テーブルの各エントリはグローバルゴール (メッセージ中のサブゴー

* 文献⁹⁾ではサブゴールの考え方がないため, こことは違った意味で compatible という言葉を用いている。

ルから求められる) とエージェント間制約の資源をキーとしてそれに対する自エージェント内で選択可能なサブゴールのリストを与える。index はサブゴールのリストの中で選択されたサブゴールが何番目のものかを示す。また、total-number はその候補のリストの要素の数を示す。ただし、候補が一つしかない場合は local-id そのものが省略される。

4.2.2 Choice List の生成

まず、起動エージェントは自分の p-goal を調べ、どのサブゴールを用いるかを求め、それを関連するエージェントに送信する。この時の選択を表した local-id がそのまま選択されたサブゴールの Choice List となる。

起動エージェント以外のエージェントにおいては他のエージェントからその仮の選択の確認のメッセージを受け取った時はまず、その内部テーブルを参照して自分の中のどのサブゴールを選択するか決める。この時メッセージに含まれているサブゴールを sg_o 、自エージェント内での選択を表す local-id を lid_h 、自エージェント内で選択されたサブゴールを sg_h とする。この時、 sg_h の Choice List, $cl(sg_h)$ は次のように更新される。

$$cl(sg_h) = cl(sg_h) \vee (cl(sg_o) \wedge lid_h)$$

ここで自エージェント内での候補が一つしかなく local-id が省略された時は lid_h は真として計算する。また、サブゴールの Choice List の初期値は偽の値を持つものとして計算する。

形式的には Choice List は次のようになる。

$$choice-list = \vee \wedge local-id$$

$$local-id = (agent [index total-number])$$

または

$$(entry-id [index total-number])$$

となる。ここで [,] は省略される場合があることを表す。

4.2.3 Choice List 簡単化ルール

Choice List は local-id の選言標準形で表現され、さらに同じ選択に関してすべてのインデックスが集められた時、それらは次の簡単化ルールに従ってまとめられる。

$$\begin{aligned} & \vee_{index=1, \dots, total-number} (\{agent|entry-id\} \\ & \hspace{10em} index \ total-number) \\ & \Rightarrow (\{agent|entry-id\}) \hspace{10em} (1) \end{aligned}$$

ただし、ここで $\{agent|entry-id\}$ は agent か entry-id のどちらかであることを示す。local-id は total-

number を含んでいるのでこの簡単化はどのエージェントにおいても行うことができる。

ここで Choice List を選言標準形に変換した時そのすべての連言要素が“(agent)”の形式の表現を含む時、その Choice List を ‘p-goal を含む’ と呼ぶ。これは起動エージェントにおけるすべての選択を含むことを意味する。問題設定の仮定から起動エージェントにはすべてのグローバルゴールに関してそれに属するサブゴールが一つ以上含まれているので、起動エージェントにおけるすべての選択とはすべてのグローバルプランの選択を意味していることになる。

4.2.4 Choice List 計算の例

図3のような通信路の一部では Choice List は次のようになる。起動エージェント A にサブゴール 1a と 2a がある。この時、2a の Choice List は

$$cl(2a) = (A \ 2 \ 2)$$

となる。エージェント B にはそのサブゴールと同じグローバルプラン (この場合二つある) に属するサブゴール 1b と 2b がある。これらのサブゴールの Choice List は

$$cl(1b) = (A \ 2 \ 2) \wedge (B \ 1 \ 1 \ 2)$$

$$cl(2b) = (A \ 2 \ 2) \wedge (B \ 1 \ 2 \ 2)$$

となる。ただし、B1 はエージェント B において 2a と接続する場合の entry-id である。さらにエージェント C にはエージェント B のサブゴール 1b, 2b とそれぞれ同じグローバルプランに属するサブゴール 1c, 2c がある。ここではエージェント C において B でサブゴールが決定してしまえば選択の余地が残されていないので Choice List は変化しない。すなわち、

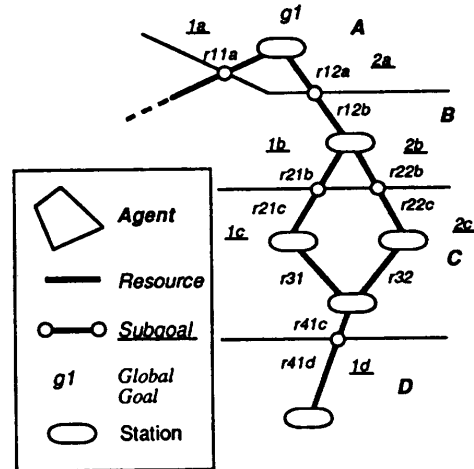


図3 通信ネットワークの別例 (部分)
Fig. 3 Another example of communication network (in part).

$$cl(1c)=cl(1b)$$

$$cl(2c)=cl(2b)$$

となる。ここでまずエージェント C でサブゴール $1c$ が選択されたとするとエージェント D ではサブゴール $1d$ が選択され、 $1d$ の Choice List は

$$cl(1d)=cl(1b)$$

となる。さらにエージェント C で新たにサブゴール $2c$ が選択されたとするとサブゴール $1d$ の Choice List は

$$\begin{aligned} cl(1d) &= cl(1b) \vee cl(2b) \\ &= ((A \ 2 \ 2) \wedge (B \ 1 \ 1 \ 2)) \vee ((A \ 2 \ 2) \wedge (B \ 1 \ 2 \ 2)) \end{aligned}$$

となり、(1)の簡単化ルールを適用すると

$$cl(1d) = (A \ 2 \ 2) \wedge (B \ 1)$$

となる。ここで $1d$ は二つのグローバルプランに属しており、この例からもわかるように Choice List はそのような場合に対してもコンパクトな表現形式を与える。

4.3 Exclusion Set の結合

前節で述べたようにサブゴールの属するグローバルプランはゴール記述を用いて表現される。Exclusion Set はサブゴールと競合関係にあるグローバルプランを表したものであり、具体的にはゴール記述の否定の選言標準形で表現する。

この時 Exclusion Set において同じグローバルゴールに関してはゴール記述は次のようなマージルールに従ってまとめる。

$$\begin{aligned} \vee \wedge \neg \langle g_i, cl(sg_i) \rangle &\Rightarrow \neg \langle g_i, \wedge \vee cl(sg_i) \rangle \\ (2) \end{aligned}$$

ここでゴール記述の Choice List が 'p-goal を含む' であれば、そのゴール記述も同様に 'p-goal を含む' という。もし、ゴール記述が 'p-goal を含む' であればそれはグローバルゴールと意味的には等価である。

4.3.1 Local Exclusion Set

Local Exclusion Set はエージェント内に閉じた競合を表したものである。サブゴール sg_i の Local Exclusion Set, LES_{sg_i} は sg_i の Conflict Set, CS_{sg_i} に現れるサブゴールをそのゴール記述で置き換えたものである。すなわち、 $CS_{sg_i} = \vee \wedge \neg sg_i$ である時、

$$LES_{sg_i} = \vee \wedge \neg \langle goal(sg_i), cl(sg_i) \rangle$$

となる。

4.4 Induced Exclusion Set

サブゴール sg_i に対する Induced Exclusion Set, IES_{sg_i} は sg_i を選択した時の局所的かつ非局所的な影響を表しているものと考えることができる。Induced

Exclusion Set は最初は Local Exclusion Set と同じ値に設定され、他のエージェントから情報を受け取るに従い変化していく。この時、どのような Exclusion Set の情報を他のエージェントに送ればよいか、そして、他のエージェントから送られてきた Exclusion Set の情報をどのように結合すればよいかが問題となる。

4.4.1 送信する Exclusion Set 情報の計算

エージェント X のサブゴール sg_x に対して、エージェント A では sg_x と同じグローバルプランに属する n 個のサブゴール sg_1, \dots, sg_n があるとする。 X が sg_x の Induced Exclusion Set を計算するためには A は sg_1, \dots, sg_n の Induced Exclusion Set を結合し、 X に通信する必要がある。

エージェント A では次の式が成り立つ。

$$\neg sg_x \vee IES_{sg_i}, (1 \leq i \leq n) \quad (3)$$

もし、 sg_x が最終的な解として選択されるならば sg_i のどれかが選択されることになる。逆にもし sg_i のどれかが最終解で選択されるならば、 sg_x も選択されることになる。したがって、次の式が成り立つ。

$$sg_x \Leftrightarrow \vee_i sg_i \quad (4)$$

(4)から

$$\neg sg_x \vee (\vee_i \neg sg_i) \quad (5)$$

(3)と(5)から次の式が導ける。

$$\neg sg_x \vee (\vee_i IES_{sg_i})$$

したがって、エージェント A は次の Exclusion Set 情報、 ES_A をエージェント X に送ればよい。

$$ES_A = \vee_i IES_{sg_i}$$

これはエージェント X からみると A に仮想的な一つのサブゴール sg_A が存在し、

$$sg_A \Leftrightarrow sg_x$$

$$\neg sg_A \vee ES_A$$

の関係にあると考えることができる。

4.4.2 受け取った Exclusion Set 情報の合成

エージェント A においてサブゴール sg_x の Induced Exclusion Set を計算するのにエージェント X_i ($1 \leq i \leq n$) とエージェント間制約があり、これらのエージェントから Exclusion Set 情報 ES_{X_i} をもらう必要があるとする。この時これらの情報をまとめ、 sg_x の Induced Exclusion Set を更新する必要がある。

それぞれのメッセージに含まれる Exclusion Set 情報、 ES_{X_i} はエージェント X_i の仮想的なサブゴール sg_{X_i} と次のような関係がある。

$$\neg sg_{x_i} \vee ES_{x_i} \quad (6)$$

ここで sg_a が選択された時, すべての i に対し sg_{x_i} も同時に選択される. 逆にすべての i に対し sg_{x_i} が最終的に選択されるならば sg_a も同時に選択されることになる. したがって, 次の式が成り立つ.

$$sg_a \Leftrightarrow \bigwedge_i sg_{x_i} \quad (7)$$

(7)より,

$$\neg sg_a \vee (\bigwedge_i sg_{x_i}) \quad (8)$$

(6)と(8)から,

$$\neg sg_a \vee (\bigwedge_i ES_{x_i})$$

が導ける. したがって sg_a の Induced Exclusion Set, IES_{sg_a} は次のようになる.

$$IES_{sg_a} = (\bigwedge_i ES_{x_i}) \wedge LES_{sg_a}$$

4.5 Goal Exclusion Set

ここでグローバルゴール g_i がその起動エージェントに n 個のサブゴール sg_1, \dots, sg_n を持っているとする. これらの関係は次のように表現できる.

$$g_i \Rightarrow \bigvee_i sg_i \quad (9)$$

ここで IES_{sg_i} を sg_i の Induced Exclusion Set とする. すなわち,

$$\neg sg_i \vee IES_{sg_i} \quad (10)$$

(9)から,

$$\neg g_i \vee (\bigvee_i \neg sg_i) \quad (11)$$

(10)と(11)から,

$$\neg g_i \vee (\bigvee_i IES_{sg_i})$$

したがってグローバルゴール g_i の Goal Exclusion Set, GES_{g_i} は次のように計算される.

$$GES_{g_i} = \bigvee_i IES_{sg_i}$$

GES_{g_i} は g_i を解くこと以外のグローバルゴールのグローバルプランへの影響を示しているといえる.

4.6 Nogood Goal Set

起動エージェントはまず自分のグローバルゴールに関し Goal Exclusion Set を計算し, それを他の関連する起動エージェントに送る. 送られてきた Goal Exclusion Set は起動エージェントの中で Induced Goal Exclusion Set に追加されていく. この Induced Goal Exclusion Set とは起動エージェントの一つできるもので, 他の起動エージェントから送られる Goal Exclusion Set を蓄えるものである. ここで Goal Exclusion Set を送る相手の起動エージェントとはこ

の Induced Goal Exclusion Set の中に出てくるグローバルゴールを p-goal として持つ起動エージェントのことである. なお自己エージェント内で求められた Goal Exclusion Set もそのエージェント内の Induced Goal Exclusion Set に追加される.

起動エージェント A に対する Induced Goal Exclusion Set, $IGES_A$ は次のように計算される.

$$IGES_A = (\bigwedge_{g_i \in IG_A} (\neg g_i \vee GES_{g_i})) \wedge (\bigwedge_{g_j \in PG_A} (\neg g_j \vee GES_{g_j}))$$

ここで IG_A は起動エージェント A の p-goal と直接, 間接的に関係するグローバルゴール, すなわち, $IGES_A$ に出現する, 自分の p-goal 以外のすべてのグローバルゴールの集合, PG_A はエージェント A の p-goal の集合を示す.

各起動エージェントは, この Induced Goal Exclusion Set に次のような操作を施すことにより Nogood Goal Set を計算する.

1. Induced Goal Exclusion Set を選言標準形

$$F_1 \vee F_2 \vee \dots \vee F_n$$

に直す. ただし,

$$F_i = (\bigwedge_j \neg g_j) \wedge (\bigwedge_k GES_{g_k})$$

である.

2. 選言標準形中の各連言 F_i に次の操作を施す.

- (a) 同じグローバルゴールに関して(2)のマージルに依りまとめる.
- (b) 各 Choice List に関して(1)の簡単化ルールにしたがって簡単化する.
- (c) 'p-goal を含む' ゴール記述をグローバルゴールに置き換える.
- (d) もし, 各連言 F_i 中に一つでもグローバルゴールが含まれれば F_i の中からグローバルゴールでないものを取り除く. これは一般に命題 a, b, c の間には次の式が成り立つことによる.

$$(a \wedge b) \vee c \Rightarrow a \vee c$$

3. ここでグローバルゴール以外のものが残ったならば Nogood Goal Set は空とする. そうでなければ書き換えられた各連言 F_i' に関して

$$F_i' \Rightarrow F_j' \quad (i \neq j)$$

となる F_i' をすべて取り除く. これは Nogood Goal Set に関して最小のもの, すなわち, 満足されないグローバルゴールが最小になる場合を考えるためである.

表 3 例における Choice List, Exclusion Set
Table 3 Choice lists and exclusion sets in the example.

Agent	Global goal	Subgoal	Choice list	Local exclusion set	Induced exclusion set
A	g1	1a	(A)		$\neg\langle g2, (B 1 2)\rangle \wedge \neg\langle g3, (C 2 2)\rangle$
B	g2	1b	(B 1 2)		$\neg\langle g1, (A)\rangle$
		2b	(B 2 2)		$\neg\langle g3, (C 1 2)\rangle$
C	g3	1c	(C 1 2)		$\neg\langle g2, (B 2 2)\rangle$
		2c	(C 2 2)		$\neg\langle g1, (A)\rangle$
D	g1	1d	(A)	$\neg\langle g2, (B 1 2)\rangle \wedge \neg\langle g3, (C 2 2)\rangle$	
		2d	(B 1 2)	$\neg\langle g1, (A)\rangle$	
		3d	(C 2 2)	$\neg\langle g1, (A)\rangle$	
E	g2	1e	(B 2 2)	$\neg\langle g3, (C 1 2)\rangle$	
		2e	(C 1 2)	$\neg\langle g2, (B 2 2)\rangle$	
F	g1	1f	(A)		
G	g2	1g	(B 1 2)		
		2g	(B 2 2)		
H	g3	1h	(C 1 2)		
		2h	(C 2 2)		

4.7 例題

第2章の通信路の例題における Nogood Goal Set を計算してみる。

この場合の各エージェントにおける Choice List, Exclusion Set は最終的には表3のようになる。そして、各起動エージェントにおける Goal Exclusion Set は表4のようになる。この Goal Exclusion Set は他の起動エージェントに送られて、結局すべての起動エージェントの Induced Goal Exclusion Set, IGES は

$$\begin{aligned}
 IGES &= (\neg g1 \vee (\neg\langle g2, (B 1 2)\rangle \\
 &\quad \wedge \neg\langle g3, (C 2 2)\rangle)) \\
 &\quad \wedge (\neg g2 \vee \neg\langle g1, (A)\rangle \\
 &\quad \vee \neg\langle g3, (C 1 2)\rangle) \\
 &\quad \wedge (\neg g3 \vee \neg\langle g2, (B 2 2)\rangle \\
 &\quad \vee \neg\langle g1, (A)\rangle) \\
 &= (\neg g1 \vee \langle g2, (B 1 2)\rangle) \\
 &\quad \wedge (\neg g1 \vee \langle g3, (C 2 2)\rangle) \\
 &\quad \wedge (\neg g2 \vee \neg\langle g1, (A)\rangle \\
 &\quad \vee \neg\langle g3, (C 1 2)\rangle) \\
 &\quad \wedge (\neg g3 \vee \neg\langle g2, (B 2 2)\rangle \\
 &\quad \vee \neg\langle g1, (A)\rangle)
 \end{aligned}$$

となる。各起動エージェントはこれを次のように変形する。

選言標準形に変換 (ステップ1)

表 4 例における Goal Exclusion Set
Table 4 Goal exclusion set in the example.

Agent	p-goal	Goal exclusion set
A	g1	$\neg\langle g2, (B 1 2)\rangle \wedge \neg\langle g3, (C 2 2)\rangle$
B	g2	$\neg\langle g1, (A)\rangle \vee \neg\langle g3, (C 1 2)\rangle$
C	g3	$\neg\langle g2, (B 2 2)\rangle \vee \neg\langle g1, (A)\rangle$

$$\begin{aligned}
 IGES &= (\neg g1 \wedge \neg g2 \wedge \neg g3) \vee \dots \\
 &\quad \vee (\neg g1 \wedge \neg\langle g1, (A)\rangle) \vee \dots \\
 &\quad \vee (\neg\langle g2, (B 1 2)\rangle \wedge \neg\langle g3, (C 2 2)\rangle \\
 &\quad \quad \wedge \neg g2 \wedge \neg\langle g2, (B 2 2)\rangle) \vee \dots \\
 &\quad \vee (\neg\langle g2, (B 1 2)\rangle \wedge \neg\langle g3, (C 2 2)\rangle \\
 &\quad \quad \wedge \neg\langle g3, (C 1 2)\rangle \wedge \neg g3) \vee \dots
 \end{aligned}$$

グローバルゴールに関してマージ (ステップ 2a)

$$\begin{aligned}
 &\Rightarrow (\neg g1 \wedge \neg g2 \wedge \neg g3) \vee \dots \\
 &\quad \vee (\neg g1 \wedge \neg\langle g1, (A)\rangle) \wedge \dots \\
 &\quad \vee (\neg\langle g2, (B 1 2)\rangle \wedge (B 2 2) \\
 &\quad \quad \wedge \neg\langle g3, (C 2 2)\rangle \wedge \neg g2) \vee \dots \\
 &\quad \vee (\neg\langle g2, (B 1 2)\rangle \\
 &\quad \quad \wedge \neg\langle g3, (C 1 2)\rangle \wedge (C 2 2) \\
 &\quad \quad \wedge \neg g3) \vee \dots
 \end{aligned}$$

Choice List の簡単化 (ステップ 2b)

$$\begin{aligned}
 &\Rightarrow (\neg g1 \wedge \neg g2 \wedge \neg g3) \vee \dots \\
 &\quad \vee (\neg g1 \wedge \neg\langle g1, (A)\rangle) \vee \dots \\
 &\quad \vee (\neg\langle g2, (B)\rangle \wedge \neg\langle g3, (C 2 2)\rangle \\
 &\quad \quad \wedge \neg g2) \vee \dots
 \end{aligned}$$

$$\begin{aligned} & \vee(\neg q_2, (B\ 1\ 2)) > \\ & \wedge \neg q_3, (C) > \wedge \neg q_3) \vee \dots \end{aligned}$$

グローバルゴールへの置き換え (ステップ 2c)

$$\Rightarrow (\neg q_1 \wedge \neg q_2 \wedge \neg q_3) \vee \dots \vee \neg q_1 \vee \dots$$

$$\vee(\neg q_2 \wedge \neg q_3, (C\ 2\ 2)) > \vee \dots$$

$$\vee(\neg q_2, (B\ 1\ 2)) > \wedge \neg q_3) \vee \dots$$

グローバルゴール以外のものの除去 (ステップ 2d)

$$\Rightarrow (\neg q_1 \wedge \neg q_2 \wedge \neg q_3) \vee \dots$$

$$\vee \neg q_1 \vee \dots \vee \neg q_2 \vee \dots \vee \neg q_3 \vee \dots$$

最小化 (ステップ 3)

$$\Rightarrow \neg q_1 \vee \neg q_2 \vee \neg q_3$$

このことはこの例題では制約が強すぎてすべてのグローバルゴールを同時には満たすことができなく、どれか一つのグローバルゴールをあきらめることが必要なことを示している。

5. 議 論

ここで提案した手法ではエージェントのすべての情報を一か所に集中する場合に比べ、起動エージェントに Exclusion Set という形式で抽象化された情報が集まるので他のエージェントの内部に関して知る必要がないという利点がある。また、逆にエージェントはその内部情報を洗いざらい他のエージェントに通信する必要もない。起動エージェントが Goal Exclusion Set を計算するため起動エージェントに対する負荷が大きくなるおそれがあるものの、グローバルな競合関係を認識するためには、ここで述べたような情報を集める必要がある。

また、各エージェントにおいてその Exclusion Set の内容から自分の局所的な選択の影響を認識することができ、無駄な試みを減らすことが可能になる。例えば Nogood Goal Set からどのグローバルゴールを同時に満足すべきかを決定した後、どのグローバルプランを用いてそのグローバルゴールを満足させるか決定する時 Goal Exclusion Set からグローバルゴールと競合関係にあるグローバルプランが何かわかるので、その情報より選択すべきグローバルプランを決めることが可能になる。さらにエージェント内でどのサブゴールを選択するか決める場合、サブゴールの Induced Exclusion Set からどのグローバルプランと競合しているかわかるので、競合関係にあるサブゴールを避けることができる。

また、制約が強すぎる場合は Nogood Goal Set か

らどのグローバルゴールを解いたらよいかを決めることができる。ここでは Nogood Goal Set が求まった後はどのグローバルゴールを解くべきかは起動エージェントの間で合意されている何らかの基準で一意に決まるものと仮定している。例えば解くべきグローバルゴールの数を最大にする、または、グローバルゴールにエージェント間で合意のとれたユーティリティ (効用) があらかじめ決まっているとして、ユーティリティの総和が最大になるように決めるなどである。例題では Nogood Goal Set が $\neg q_1 \vee \neg q_2 \vee \neg q_3$ であり、仮に q_1, q_2, q_3 のユーティリティがそれぞれ 10, 20, 30 だとすると、 q_1 をあきらめることによって q_2 と q_3 が満足でき、解くグローバルゴールのユーティリティの和を 50 と最大にすることができる。したがって q_1 をあきらめるべきと決めることができる。もし、このようなエージェント間で合意のとれた基準がない場合は、どのグローバルゴールを解くかについて Nogood Goal Set の情報に基づいて起動エージェントの間で別のネゴシエーションを行う必要がでてくる。

6. おわりに

本論文ではマルチステージネゴシエーションを定式化し、エージェント間で交換すべき情報の形式を明らかにした。ゴール記述というものを取り入れグローバルプランをコンパクトに表現した。グローバルプランの情報をやりとりすることにより、ゴールの競合が複数のエージェントにまたがっている場合でも正しく競合関係が認識できる。

さらにここで述べた情報の交換を基にネゴシエーションを進めるための 3 フェーズプロトコルを開発している^{1),6)}。具体的にはエージェントがお互いに勝手に解を見つけようとする非同期探索フェーズとそのフェーズで解が見つからなかった時にエージェント間で不足している情報を交換し Nogood Goal Set を完全に求める協調探索フェーズとその結果制約が強すぎてすべてのグローバルゴールが満足できないことがわかった時に解くべきグローバルゴールを決定する過制約解消フェーズからなる。そこでは各エージェントが Local Exclusion Set が完全に求まったことを認識し、認識した時点で acknowledgement を通信することにより、起動エージェントがその Goal Exclusion Set, Nogood Goal Set が完全に求まったかどうかを認識できるようなメッセージのやりとりを決めている。フェーズ間の移行に際しては Goal Exclusion Set, No-

good Goal Set が完全に求まったかどうかを調べている。これらについてはまた別に稿を改めて論じたい。

また、マルチステージネゴシエーションは分散制約充足と呼ばれる種類の問題を取り扱っているとみることができる¹⁰⁾。分散制約充足問題における他の手法との定量的な評価も含めた比較も今後の課題の一つである。

また、ここでの問題設定ではネゴシエーションの途中で問題の条件が変化しないことを仮定している。状況が変化する環境での適用を考えるとネゴシエーションの途中でグローバルゴールが新たに追加されたり、また、制約が変化したりすることに対処しなければならない。このような状況の変化に対応していくようにマルチステージネゴシエーションを拡張していくのも今後の課題の一つである。

謝辞 ご討論いただいた Clarkson University の Prof. Susan E. Conry, Prof. Robert A. Meyer に感謝します。また草稿に多くの有益なコメントをいただいた NTT コミュニケーション科学研究所の石田亨主幹研究員、赤埴淳一研究主任、榎肅之研究主任に感謝します。

参 考 文 献

- 1) Conry, S. E., Kuwabara, K., Lesser, V. R. and Meyer, R. A.: Multistage Negotiation for Distributed Constraint Satisfaction, (to appear in *IEEE Trans. Systems, Man and Cybernetics*).
- 2) Conry, S. E., Meyer, R. A. and Lesser, V. R.: Multistage Negotiation in Distributed Planning, Bond, A. H. and Gasser, L. (eds.), *Readings in Distributed Artificial Intelligence*, pp. 367-384, Morgan Kaufmann (1988).
- 3) Conry, S. E., Meyer, R. A. and Pope, R. P.: Mechanisms for Assessing Nonlocal Impact of Local Decisions in Distributed Planning, Huhns, M.N. and Gasser, L. (eds.), *Distributed Artificial Intelligence, Volume II*, pp. 245-258, Morgan Kaufmann (1989).
- 4) Davis, R. and Smith, R. G.: Negotiation as a Metaphor for Distributed Problem Solving, *Artif. Intell.*, Vol. 20, No. 1, pp. 63-109 (1983).
- 5) Durfee, E. H., Lesser, V. R. and Corkill, D. D.: Trends in Cooperative Distributed Problem Solving, *IEEE Trans. Knowledge and Data Engineering*, Vol. 1, No. 1, pp. 63-83 (1989).
- 6) 桑原和宏: マルチステージネゴシエーションによる分散資源割当, 1990年代の分散処理シンポジウム, pp. 87-96, 情報処理学会 (1990).
- 7) Kuwabara, K. and Lesser, V. R.: Extended Protocol for Multistage Negotiation, *9th Workshop on Distributed Artificial Intelligence*, pp. 129-161 (1989).
- 8) Smith, R. G.: The Contract Net Protocol: High-Level Communication and Control in a Distributed Problem Solver, *IEEE Trans. Comput.*, Vol. 29, No. 12, pp. 1104-1113 (1980).
- 9) Smith, R. G. and Davis, R.: Frameworks for Cooperation in Distributed Problem Solving, *IEEE Trans. Systems, Man and Cybernetics*, Vol. 11, No. 1, pp. 61-70 (1981).
- 10) Yokoo, M., Ishida, T. and Kuwabara, K.: Distributed Constraint Satisfaction for DAI Problems, *10th International Workshop on Distributed Artificial Intelligence* (1990).

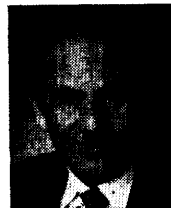
(平成2年12月12日受付)

(平成3年7月8日採録)



桑原 和宏 (正会員)

1959年生。1982年東京大学工学部電気工学科卒業。1984年同大学院電子工学専門課程修士課程修了。同年日本電信電話公社入社。以来知識ベースシステム、分散協調問題解決などの研究に従事。1988年9月より一年間 University of Massachusetts at Amherst 客員研究員。現在、NTT コミュニケーション科学研究所主任研究員。



Victor R. Lesser

Victor R. Lesser received his B. A. in Mathematics from Cornell University in 1966, and the M. S. and Ph. D. degrees in Computer Science from Stanford University in 1969 and 1972, respectively. Prior to his arrival at University of Massachusetts in 1977 where he is a Professor of Computer Science, he spent five years as a research computer scientist at Carnegie-Mellon University, where he was responsible for the system architecture of the Hearsay-II Speech Understanding System. His major research focus is on the control and organization of complex AI systems. Professor Lesser is a fellow of the American Association of Artificial Intelligence (AAAI), and has done extensive research in the areas of blackboard systems, distributed AI, and real-time AI. He has also made contributions in the areas of computer architecture, diagnostics, intelligent user interfaces, parallel AI, and plan recognition.