

# 自律エージェントの協調活動のプランニング

伊藤 嘉邦      安西 祐一郎

慶應義塾大学 理工学部

自律的なエージェントが、プラン交換等の通信なしで、他のエージェントの行動からエージェントの集団による協調作業のモデルを構築し、そのモデルに基づいて自らのプランを決定していく方法について考察し、このような能力を持つCAPエージェントを提案する。CAPエージェントは、協調作業のモデル化機能、モデルの妥当性管理機能を持ち、モデル構築においてはゴール-サブゴール階層を用いたモデル化方法、モデルの妥当性管理においては、ATMSに似た方法を用いる。

## Planning for Cooperative Activity by Autonomous Agent

Yoshikuni Itoh    Yuichiro Anzai

Faculty of Science and Technology, Keio University

3-14-1, Hiyoshi, Kohoku-ku, Yokohama 223, Japan

In this paper we consider how an autonomous agent construct the model of cooperative work done by group of agents without exchanging their plans and decide its own plan following the model, and we propose the CAP agent. The CAP agent has the ability of generating the model of cooperative work and maintaining it. It utilizes goal-subgoal hierarchy to generate the model and the ATMS-like method to maintain it.

## 1 はじめに

我々の世界では、他の人々と互いに協力しあいながら活動をしていく場面がたいへん多い。例えば、家の引越しをする場合を考えてみよう。誰かが、テーブルの片方を持ち上げたら、すぐに手の空いている者がもう片方を持ち上げる。家の中から家具を一つ運び出し終えたら、すぐに次の仕事を探して取り掛かる。後からやってきた人も、他の人々の作業の様子を見た上で、自分の協力できるところから自主的に作業に参加する。このような作業を行なう際に、他人との通信はごく少ない。また、今後の行動のプランを他人と交換して、それを参考に自分の行動のプランを作成するといったこともほとんど行なわれない。各作業者は、他の人との情報交換を通してではなく、周囲の人の行動の観察から今、身の回りで進んでいる協調作業のモデルを自分自身で自分の内部に構築し、それに従って自分の行動を決定する。もちろん、作業中のリソースの確保(段ボール箱を他人に使われないようにしておく等)、行動の競合回避(ドアでかち合った際の譲り合い)などのため、他人との通信が全く不必要とは言えない。しかし、このような協調作業で自分の行なう作業、すなわち協調作業を行なっているグループのゴールの中での自分の実行するサブゴールを決定する方法は、リーダーが明示的に各人の分担を決定する集中制御方式か、他人の行動の観察から構築した協調作業のモデルに基づいて自律的に自分の作業を決定する方式、あるいはその両者の組み合わせによって行なわれることが多いと考えられる。そこで、本研究では、自律的なエージェントが、プランの交換等の通信なしで、他のエージェントの行動からエージェントの集団による協調作業のモデルを構築し、それに基づいて自らのプランを決定していく方法について考察する。

## 2 背景

### 2.1 マルチエージェントの協調活動に関する研究

マルチエージェントシステムにおいて、エージェント同士の協調作業を実現するため、複数のエージェントを集中制御機構なしで協調作業を行なわせる研究が盛んに行なわれてきた [Gasser, 89]。

Multistage Negotiation [Conry, 86] は、Contract net [Davis, 83] のプロトコルを発展させたものである。Multistage Negotiation では、複数のエージェントによって分散管理された通信網でのコネクション経路を決定するため、エージェント間で要求と自分の持つ情報を繰り返して交換していく。これを協調作業のためのプランニングに応用すると、各エージェントが、互いに他のエージェントへ仕事の依頼を行ない、依頼を受け取ったエージェントは、自分自身の制約に照らし合わせて返答を返すという手続きを繰り返しながら、各エージェントのゴールを決定していくことになる。しかしながら、Multistage Negotiation では、プランニングとアクションのフェイズが分離しており、エージェント全体でのプランに対する合意が形成されるまで、各エージェントはアクションを起こせない。また、他のエージェントが一時的に選択したゴールの通達から、自らのゴールを決定する一般的な方法は明確になっていない。

Partial Global Plan (PGP) [Durfee, 87] は、各エージェントが自分のプランを他のエージェントと交換し、複数エージェントのプランを合わせた partial global plan を作成する。PGP は、他のエージェントのプランとの整合性を保つのに効果的だが、一般的に他のエージェントと協調していくための自分のゴールを決定する手法は明らかになっていない。

このような研究は、エージェントが他エージェントとの情報交換(プランの交換等)を通じて、エージェントが他のエージェントを考慮したグローバルな視野に立ったプランを構築していく手法が中心であった。この場合、協調作業を行なうエージェントは互いに、予めエージェントの設計者によって規定された協調的なアルゴリズム通りに振舞うことが暗

黙の前提となっていることが多い。エージェントから見れば、他のエージェントのプランニングのアルゴリズムを熟知し、かつ他のエージェントが設計された通りに合理的に振舞うことを所与の前提として、自分のプランニングをすることになる。従って、研究の焦点は、より効率的なタスク割り当て、競合回避、通信プロトコルなどにあった。

## 2.2 本研究の位置づけ

本研究では、他のエージェントの内部的なアルゴリズムや通信のプロトコルを知らないエージェントが、他のエージェントとの協調作業を行なっていく方法について考察する。本研究では、基本的にエージェント間でプラン交換等の通信を行なわない。もちろん、通信は重要な手段であり、通信を行なえばより効率の良い協調作業の遂行ができる可能性がある。しかしながら、通信は、必ず受信エージェントの通信に反応した行動を期待して、あるいは受信エージェントの通信に対する行動を予測して行なわれるため、ここにシステム的设计者によって形作られた協調作業を行ない易くするための仕組みが入り込むことになる。すなわち、通信を行なうエージェントは、他のエージェントの内部的なアルゴリズムを知っていると考えられる場合が多い。また、通信は非常に有効になる状況がある反面、不安定な側面もあり、通信エラーや通信不能に陥ることも多い。従って、頑健なエージェントを設計するためには、通信なしでの協調作業にも対応できるようにするべきであろう。また、複数の人間と複数のエージェントが協調作業をしていく場面を考えてみる [Nakauchi, 91a][Nakauchi, 91b]。すると、従来のような、エージェント間での詳細なプランの交換を人間に対して行なうことは現実的でないことに気付く。人間の内部的なアルゴリズムをエージェントが知ることも不可能であり、エージェントは人間に対して、受信に対する人間の反応を正確に予測して通信を行なうことはできない。従って、人間と入り交じって協調作業を行なうエージェントには、通信に頼らずに協調作業を進めていける仕組みが必要である。通信なしでのエージェントの相互作用に関する研究として、[Rosenschein, 85][Genesereth, 86]

がある。[Genesereth, 86] では、それぞれ2種類のアクションを取り得る2つのエージェントの、合理的な振舞い(アクションの選択)について分析している。本研究では、エージェントが、

- エージェントは、協調作業の中での自分のゴール、プランを自律的に決定する。
- エージェント間でのプラン交換等の通信を行なわない。エージェントは、他のエージェントの実行しているアクションのみを知り得る。
- エージェントは、他のエージェントの内部的なアルゴリズムを知らない。
- エージェントの周囲の環境は常に変化するので、これに対処できるプランニングを行なう。

といった要件を満たした上で、他のエージェントと協調作業を進めていくためのプランニングの方法について考察する。

## 3 本研究の概略

前節で述べたような要件を満たして、エージェントが協調作業のためのプランニングを行なうためには、周囲のエージェントのアクションの観察から今、自分の周りで進んでいる協調作業のモデルを生成し、そのモデルに合わせて自らのゴールを自主的に設定し、そのゴールに到達するプランを作成する能力がエージェントに必要であると考えられる。本研究では、このような能力を持つエージェントであるCAP(Cooperative Action Planner) エージェントを提案する。CAP エージェントは、特に次の2つの機能を持つ。

協調作業のモデル化機能 周囲のエージェントのアクションの時系列の観察から、これらのエージェントによって進められている協調作業のモデルを構築する機能。CAP エージェントは、このモデルに従って自分のゴールを自主的に設定する。

協調作業のモデルの妥当性管理機能 協調作業のモデル化機能によって構築されたモデルの妥当性

を常に監視し、後にモデルが妥当でないことが判明した場合には、このモデルに基づいて作成された自分のプランを撤回する機能。

協調作業のモデル化については、本論文では、協調作業のゴール-サブゴール階層を用いたモデル化を取り挙げる。エージェントは、周囲のエージェントの振舞い(アクション)から、それらのエージェントのゴールを推測する。更に、それらのエージェントのゴールが、エージェントのグループのゴールのサブゴールを担っていると仮定して、エージェントのグループのゴールを推測する。このように、サブゴールからの上位ゴールを推測するためには、エージェントは、ゴール-サブゴール関係に関する知識を豊富に持たなくてはならない。本論文で提案するエージェントは、1階層のゴール-サブゴール木のフラグメントをゴール-サブゴール関係に関する知識として多数所持する。このフラグメントは、1個のゴールを根にして、それを満たすためのサブゴール、あるいは具体的なアクションを葉に持った1階層の木として表される(図1上)。葉にあたるサブゴールやアクションは、半順序関係が付けられたネットワークとして表される。条件部には、そのフラグメントのゴールを実行する際の前提条件を記述する。この前提条件は、実行時に使用される。フラグメントの構造は、[Firby, 87]におけるRAPに似ている。エージェントは、このフラグメントを組み合わせて、協調作業のゴール-サブゴールのモデル化を行なう(図1下)。

協調作業のモデルの妥当性管理機能は、

- 協調作業のモデルは限られたデータから作成された推測であるため、その後獲得された新しいデータでモデルの正しさを検証していく必要。
- 環境は常に変化しており、ある時点で構築した協調作業のモデルが、ずっと有効であるとは限らない。従って、現時点でのモデルの有効性を確認していく必要。

の2点によりエージェントに必要な機能である。協調作業のモデルは、1階層のゴール-サブゴール木のフ

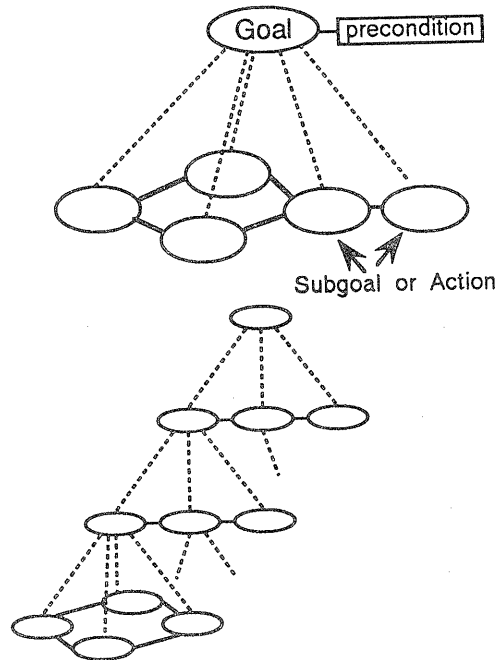


図1: ゴール-サブゴール木のフラグメント

ラグメントを組み合わせて作成されるので、このモデル全体のある一部分でモデルの有効性が失われると、その影響が組み合わされた木を伝って伝播していく可能性がある。また、このモデルを基に作成した自分のプランも部分的に無効化しなくてはならない。このような、フラグメント間での依存関係を明確に管理し、モデルの無効化を効率良く実行するため、本研究では、ATMS(Assumption-based Truth Maintenance System)[de Kleer, 86]に似た方法を用いる。

## 4 協調作業のためのプランニング

ここでは、これまで述べたような協調作業を実現するCAPエージェントの内部構成と、そのアルゴリズムについて述べる。

### 4.1 CAP エージェント

CAP エージェントは、モデル構築部、環境監視部、モデル検証部、データベース部、実行部の5つ

の部分から構成される (図 2)。

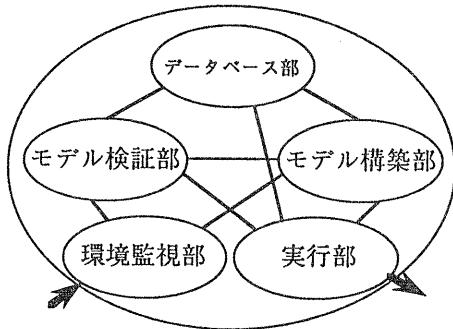


図 2: CAP エージェント

**モデル構築部** 周囲のエージェントの協調作業のモデルを構築する。モデル構築部は、他のエージェントのアクションの履歴に、ゴール-サブゴール木のフラグメントの葉部分を部分マッチさせる。こうして推測された他エージェントのゴールに、更に再帰的にフラグメントを部分マッチさせ、エージェントのグループのゴールを導出する。マッチしたフラグメントは、協調作業のモデリングに関する論拠として、モデル検証部に登録する。

**環境監視部** 他のエージェントのアクションを監視し、履歴を作成する。

**モデル検証部** モデル構築部の作成したモデルの妥当性を常に検証する。環境監視部が観察したモデル構築後の他のエージェントのアクション、モデル構築部が推測したゴールを、モデル構築部によって論拠として登録されたフラグメントに適合させてみる。アクションがうまく適合しなかった場合、無効化するフラグメントを決定してから、モデル構築部を起動する。

**データベース部** モデル構築部が使用する 1 階層のゴール-サブゴール木のフラグメントを蓄えておく。

**実行部** モデル構築部が構築したモデルに基づいて自分のゴールを決定し実行する。フラグメントが組み合わさってできたゴール-サブゴール階層からなるモデルの内、まだ実行されていないサブゴールを選び、自分のゴールとする。このゴールを実行するため、このゴールをフラグメントの根の部分にマッチさせ、このフラグメントを用いてゴールをサブゴールに展開する。サブゴールを再帰的に展開していき、プリミティブなアクションとなったところで実際に実行する。

CAP エージェントの各機能部分は、並列に実行される。従って、プランの実行中でも、環境の変化に柔軟に対処できる。

## 4.2 モデルの妥当性管理

CAP エージェントは、作業記憶として、次の 4 つを持つ。

- **cw-model 協調作業モデル** CAP エージェントが構築した協調作業のモデルを表す。モデル構築部により管理される。推測されたグループの協調作業のゴールがここに含まれる。
- **hypothesis-db 仮説データベース** 協調作業モデルを構築する際に使用されたフラグメントをここに登録しておく。モデル検証部により管理される。登録されたフラグメントは、そのサブゴール部分の実行状況とマッチ率、このフラグメントが登録された際の依存関係を表すラベルと共に管理される。
- **exec-queue 実行キュー** CAP エージェントが自分のプランをここに構築し格納しておく。実行部により管理される。実行キュー内のプランは、モデル検証部により無効化されることがある。
- **history ヒストリー** 他エージェントのアクションの履歴を記録しておく。環境監視部により管理され、モデル検証部により利用される。

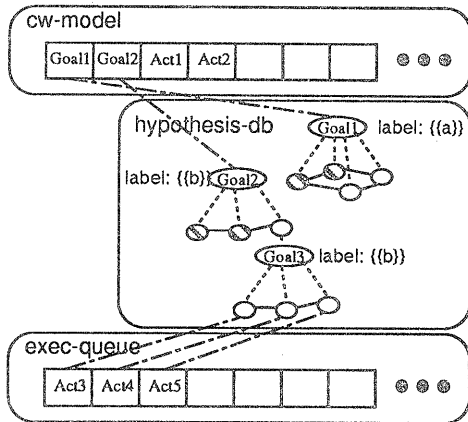


図 3: CAP エージェントの作業記憶

モデル検証部では、モデル構築部で他エージェントのゴール推測、グループのゴール推測に利用されたフラグメントと、実行部でゴール-サブゴール展開に利用されたフラグメントを hypothesis-db 内に管理している。モデル構築部や、実行部が hypothesis-db に、フラグメントを登録すると、フラグメントは、ATMS におけるノード (node) の様に扱われる。従って、フラグメントには、ATMS のノード同様にラベル (label) 付けされる。モデル構築部でフラグメントの葉を部分マッチさせた際、マッチングした部分が他エージェントのアクションのみから成るフラグメントは、仮説 (assumption) として扱われる。マッチングした部分にサブゴールを含むフラグメントのラベルは、サブゴールを担っているフラグメントのラベルから計算される。この計算には、サブゴールを担っているフラグメントを前件とした正当化 (justification) を実行する。実行部によるゴール-サブゴール展開によって登録されたフラグメントのラベルの内容は、上位ゴールのフラグメントのラベルの内容をそのまま継承する。モデル検証部により、hypothesis-db 内のあるフラグメントが無効となった場合、無効になったフラグメントのラベルを矛盾 (nogood) とする。この

結果、ラベルが空になったフラグメントは、次の2つの内いずれかの処理を行なった後、hypothesis-db から消去される。

- モデル構築部により登録されたフラグメントであった場合、フラグメントの葉の部分の内、既にマッチしている部分を cw-model に書き戻す。
- 実行部により登録されたフラグメントであった場合、exec-queue から、このフラグメントの葉部分であるアクションを消去する。

### 4.3 アルゴリズム

CAP エージェントの各機能部分におけるアルゴリズムを簡単に記す。

#### 4.3.1 環境監視部

1. 他エージェントのアクションを観察する。
2. 観察された他エージェントのアクションを history 及び cw-model に付け加える。

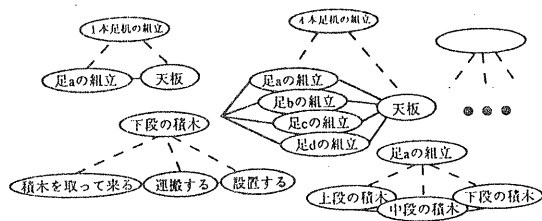
#### 4.3.2 モデル構築部

1. cw-model にフラグメントの葉を部分マッチさせる。
2. cw-model 中の部分マッチした部分を削除し、フラグメントの根と置き換える。
3. 置き換えたフラグメントを hypothesis-db に登録する。この時、フラグメントのラベルは、cw-model から削除されたフラグメントを前件とした正当化 (justification) によって計算される。

#### 4.3.3 モデル検証部

1. history に新しく追加されたアクション、あるいはモデル構築部が仮定したゴールを hypothesis-db 内のフラグメントに当てはめてみる。当てはまった場合は、そのフラグメントのマッチ率を更新する。

2. 当てはまるフラグメントが無かった場合は、アクションのマッチ率が最も低いフラグメントを選び、このフラグメント及びこのフラグメントに依存したフラグメントを `cw-model` に戻す。また、`exec-queue` からは取り除く(前節を参照)。この時、依存関係は、フラグメントに付けられたラベルから導く。



CAPエージェントの所持しているフラグメント

#### 4.3.4 実行部

1. `exec-queue` が空ならば、`hypothesis-db` から、未実行のサブゴールを1つ選び、`exec-queue` へ入れる。
2. `exec-queue` から1つ取り出す。アクションならば、それを実行する。
3. サブゴールならば、それを根とするフラグメントを持ってきて、その条件部を調べ、実行する必要があるかどうか判断する。
4. 実行する必要があるならば、そのフラグメントを用いゴール - サブゴール展開をして、`exec-queue` に入れる。この時用いたフラグメントを `hypothesis-db` に登録する。ラベルの内容は、上位ゴールのフラグメントのラベルの内容を継承する。

#### cw-modelの変化

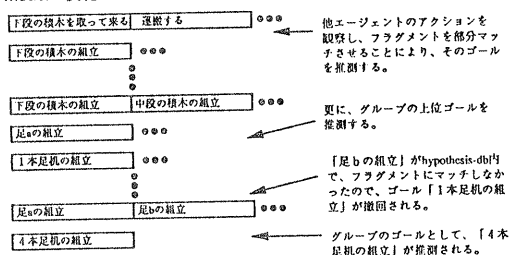


図 5: CAP エージェントの実行例

## 5 具体例

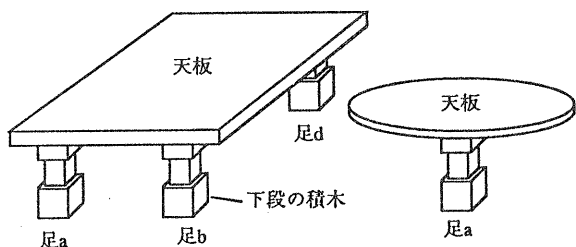


図 4: 例

CAP エージェントの振舞いを示すための例として、いくつかのエージェントが図 4左のような4本

足の積木のテーブルを組み立てようとしている状況を考えてみる。CAP エージェントは他エージェントのアクションを観察し、`cw-model` 上で、この観察したアクションにフラグメントを当てはめることにより、図 5に示すようなゴール推測をする。この例では、エージェントは、誤って図 4右のような1本足のテーブルをゴールとして推測してしまっているが、次の他エージェントによる足bの組み立てを観察することにより、モデルの修正が行なわれる。

## 6 おわりに

本論文では、自律的なエージェントが、プラン交換等の通信なしで、他のエージェントの行動からエージェントの集団による協調作業のモデルを構築し、そのモデルに基づいて自らのプランを決定していく方法について考察し、このような能力を持つ CAP エージェントを提案した。CAP エージェントは、協調作業のモデル化機能、モデルの妥当性管理機能を持ち、モデル構築においてはゴール-サブゴール階

層を用いたモデル化方法、モデルの妥当性管理においては、ATMS に似た方法を用いた。

今後の課題として、モデル化能力の向上、特にエージェント間の協調関係等に注目したモデル化を取り入れること、複数モデルの構築と比較機能、ネゴシエーションなどの通信を行なう上層機構との融合が考えられる。現在、CAP エージェントによる協調作業のシミュレーションとチューニングを行ない、更にこれらの課題を考慮して発展させていくことを試みている。

## 参考文献

- [Gasser, 89] Gasser, L. and Huhns, M. N. (eds.), *Distributed Artificial Intelligence, volume II*, Morgan Kaufmann Publishers, Inc., 1989.
- [Davis, 83] Davis, R. and Smith, R. G., "Negotiation as a Metaphor for Distributed Problem Solving," *Artificial Intelligence*, Vol.20, No.1, pp.63-109, Jan. 1983.
- [Conry, 86] Conry, S. E., Meyer, R. A. and Lesser, V. R., "Multistage Negotiation in Distributed Planning," *COINS Tech. Rep. 86-67, Dep. of Computer and Inf. Sciences, Univ. of Massachusetts*, Dec. 1986.
- [Durfee, 87] Durfee, E. H. and Lesser, V. R., "Using Partial Global Plans to Coordinate Distributed Problem Solvers," *Proc. of the Tenth Int. Joint Conf. on Artificial Intelligence*, pp.875-883, Aug. 1987.
- [Nakauchi, 91a] Nakauchi, Y., Itoh, Y., Sato, M. and Anzai, Y., "Modeling and Implementation of Multiagent Interface System for Computer-Supported Cooperative Work," *Ergonomics*, 1991 (in press).
- [Nakauchi, 91b] Nakauchi, Y., Itoh, Y., Sato, M. and Anzai, Y., "Michele: A multiagent interface architecture for distributed open environments," *Proc. of Technology of Object Oriented Languages and Systems (Tools '91)*, pp.61-69, Mar. 1991.
- [Rosenschein, 85] Rosenschein, J. S. and Genesereth, M. R., "Deals Among Rational Agents," *Proc. of the Ninth Int. Joint Conf. on Artificial Intelligence*, pp.91-99, Aug. 1985.
- [Genesereth, 86] Genesereth, M. R., Ginsberg, M. L. and Rosenschein, J. S., "Cooperation without Communication," *Proc. of fifth National Conf. on Artificial Intelligence*, pp.51-57, Aug. 1986.
- [de Kleer, 86] de Kleer, J., "An Assumption-based TMS," *Artificial Intelligence*, Vol.28, pp.127-162, 1986.
- [Firby, 87] Firby, R. J., "An Investigation into Reactive Planning in Complex Domains," *Proc. of sixth National Conf. on Artificial Intelligence*, pp.202-206, July 1987.