

# 複雑な環境におけるマルチエージェント協調方式の設計

## Design of Coordination of Multi-Agent System under Complicated Environment

大竹 麗央      西田 豊明  
OHTAKE Leo    NISHIDA Toyooki

東京大学 大学院情報理工学系研究科  
Graduate school of Information Science and Technology, The University of Tokyo.

In this paper, we define the “waiter problem” to deal with coordination in multi-agent system under complicated environment like real world. In the waiter problem, waiter agents must provide effectively liquor and relishes for the participants in a buffet-style dinner party. We introduce ‘chip’ and ‘contract’ to make coordination easy to happen in this problem, and implement the problem with InViWo platform.

### 1. はじめに

昨今のコンピュータ技術の急速な発展に伴い、我々の日常生活のさまざまな部分にコンピュータが入り込んできている。このような人工物が多数集まり、お互い通信することで協調的にサービスを提供してくれるならば、ユーザの要求に対して柔軟に対応することが出来ると考えられる。このため、現実世界のような複雑な環境における、マルチエージェントの協調技術の開発が望まれる。

マルチエージェント協調の研究はこれまで、エージェント間の通信言語や相互作用のあり方、～に関するものが主流であったが、その大半が追跡問題 [Benda 1985] などの比較的単純な問題設定におけるものが多く、現実世界のような複雑な環境にその成果を適用することはほとんど不可能であると考えられる。一方で、最近では RoboCup Soccer [RoboCup Web] や RoboCup Rescue [Kitano 2000] のような、現実世界に近い複雑な環境におけるマルチエージェント協調の標準問題が提示されている。これらの問題では実時間でのプランニングや通信、環境に関する不確実性の処理、さらには環境の動的変化への追従といった非常に多くの要素を考慮せねばならないが、マルチエージェント協調の問題としては従来から一気に難しくなった感があり、従来の研究成果を生かす意味でも、両者の中間となるような標準問題が必要であると考えられる。

現実世界のような複雑な環境での協調を扱うためには、主に次に挙げる点を考慮する必要がある。

- 環境の動的変化に対する追従
- 通信、環境の不確実性に対する処理
- 実時間でのプランニング
- 異種エージェント間の協調

このうち、本研究では「環境の動的変化に対する追従」に着目し、複雑な環境におけるマルチエージェント協調の標準問題としてウェイター問題を定義し、ウェイター問題における協調方式の設計を行った。

以下、2章ではウェイター問題の定義を行う。3章ではウェイター問題を解くために“チップ”、“報酬”という2概念を導

連絡先: 東京大学 大学院情報理工学系研究科 電子情報学専攻  
西田・黒橋研究室 大竹麗央, 〒113-8656 文京区本郷7-3-1 工学部3号館230号室, Tel/Fax: 03-5841-8758, Mail: leo@kc.t.u-tokyo.ac.jp

出し、協調のための行動を設計する。4章で InViWo プラットフォームを用いた実装法について述べ、最後に5章で本稿をまとめる。

### 2. ウェイター問題

本研究で扱う協調問題は、立食パーティにおけるウェイターエージェント（以下、エージェント）の協調動作に関するもので、以下ウェイター問題と呼ぶことにする。本節では、ウェイター問題を定義する。エージェントはパーティ参加者の好みのお酒やおつまみを効率よく提供するという使命を持っている。

#### 2.1 ウェイター問題の定義

パーティでは、お酒やおつまみはカウンタに置いてある。お酒やおつまみは一定時間ごとに補充されるが、数や種類には限りがある。エージェントはカウンタから適切なお酒やおつまみを選んで持ち、パーティ参加者に渡す。パーティ参加者はランダムに移動しており、自らカウンタでお酒やおつまみを受け取ることはしない。パーティ参加者とエージェントの間、およびエージェント同士は互いに通信を行うことが出来るが、全体を統括する中央集権のエージェントは設けず、各エージェントはそれぞれ自分の観測した情報に基づいて自律的に行動する。

このとき、限られた時間内でパーティ参加者にできるだけよいサービスを提供するためには、エージェントはどのように行動せねばならないか。

#### 2.2 ウェイター問題の特徴

我々は現実世界での人工物の協調動作への適用を念頭に入れ、ウェイター問題を設計した。現実世界での問題の特徴として、その複雑さが挙げられる。ウェイター問題に存在する複雑さの要素としては、以下の点が挙げられる。

- クライアント（パーティ参加者）、資源（お酒・おつまみ）という2種類の競合が存在する
- エージェント間の行動（品物を取りに行く、品物を渡しに行く）が非同期である

特に後者は、環境が動的に変化するという現実世界の特徴を反映している。例えば、あるエージェント A があるパーティ参加者 P にワインを持っていく最中に、P が別のエージェント B からビールを受け取ってしまったとすると、A は手持ちのワインを別のパーティ参加者に渡さねばならなくなる。エージェントはお酒やおつまみを運ぶ際、常にこのような事態を想定しておかねばならない。

## 2.3 得られた解の評価

ウェイター問題の解は、エージェントの行動系列である。この解のよさ、すなわち、エージェントがどれだけよいサービスを提供したかを評価するには、各サービスの評価をする必要がある。

### 提供されたサービスの評価

パーティ参加者はすべてのサービス（この場合、エージェントがお酒やおつまみを持ってくること）が終了した段階で、自分が受けた一連のサービスを評価する。この評価には、1回のサービスの内容だけでなく、そのサービスがどのようなタイミングで起こったのかも含まれるべきである。しかしこのタイミングは基本的に、前のサービスを受けてからの時間間隔、およびサービスを受けた際の自分の状況に依存すると考えられるため、これはサービスが生じた段階で十分評価可能であると考えられる。したがって、パーティ参加者のサービスに対する評価は、サービス1回ごとの評価の線形和で表すことが出来る。

パーティ参加者は1回サービスを受けるたびに、次に挙げる2点を評価基準としてそのサービスを評価する。

(a) サービスの内容

(b) サービスのタイミング

(a) については、エージェントが持って来たお酒やおつまみの種類が自分の好みとどれだけ合っているか、量は自分にとって適量か、などの観点から評価できる。また、(b) については、パーティ参加者にとって、エージェントにお酒やおつまみを持ってきてもらいたいというタイミングは異なる。例えば、他のパーティ参加者と談笑している間は、お酒やおつまみを持ってきてほしくないという人もいるだろうし、いつも手元に何らかのお酒やおつまみを持っていたいという人もいるだろう。

エージェントがパーティ参加者にお酒やおつまみを1回持ってくるときのサービスの評価は、上に挙げた2点についてそれぞれ数値化し、重みつき線形和をとることで、算出できると考えられる。

### サービス全体の評価

前節で述べたサービスの評価値をもとに、パーティ参加者全員に提供されたサービスのよさを評価する。

最も単純な評価方法は、各パーティ参加者に提供されたサービスの平均をとることである。しかし、次の2つの解は、サービスのよさが同じといえるだろうか？

(A) パーティ参加者1人が評価値100のサービスを受け、残り9人は全くサービスを受けなかった

(B) 10人のパーティ参加者全員が評価値10のサービスを受けた

両者とも、サービス評価の平均は同じ10であるが、内容は全く異なる。社会的な平等性を考えれば、(B)よりも(A)の方がよいサービスと言えよう。この平等性を測る尺度としては、サービス評価値の分散あるいは標準偏差を利用することが考えられる。

したがって、パーティ参加者全員に提供されたサービスのよさは、サービス評価の平均と標準偏差<sup>\*1</sup>の両方を用いるべきで

\*1 ここで、分散ではなく標準偏差を用いるのは、平均と次元を合わせるためである。

ある。しかし、両者の重要度比率はスケージングの違いにより大きく変動すると考えられるため、両者の重みつき線形和をとるという手法の適用は現実的ではない。

そこで、平等性はウェイター問題の解として認める際の必要条件として用いる。すなわち、サービス評価値の標準偏差がある閾値以下のもののみを解として認め、その条件の下でできるだけサービス評価値の平均値を高いものを求める。

## 2.4 設定すべきパラメータ

ウェイター問題を実装する際に設定すべきパラメータには、以下のようなものがある。

- パーティ参加者の好み
- カウンタの品揃えの更新頻度および更新内容
- エージェントの能力
  - － 視野の広さ
  - － 移動速度
  - － 1回に運べる量

また、各エージェントは以上の情報をすべて知っているものとする。

## 2.5 ウェイター問題における協調行動

ウェイター問題において、パーティ参加者によりよいサービスを提供するためには、エージェントは互いに協調せねばならない。ここで言う“協調”とは、エージェントが他のエージェントの行動を考慮に入れた上で、自らの行動を決定することを指している。

ウェイター問題におけるエージェントの協調行動として、最初に考えられるのが、パーティ参加者の分担である。例えば、複数のエージェントが同一のパーティ参加者に対してサービスを提供している一方で、誰にもサービス提供をされないパーティ参加者がいるという状況は、望ましいとは言えない。エージェントは、長時間にわたってサービスを受けていないパーティ参加者がいないように、互いに分担してパーティ参加者にサービスを提供する必要があると言える。

しかしその一方で、同一パーティ参加者に複数のエージェントが立て続けにサービスをする必要がある場面も考えられる。例えば、ワインとチーズをセットで味わうのが好きなパーティ参加者に対して、ワインしか持っていないエージェントがサービスを提供したとすると、その直後にチーズを持ったエージェントがサービスを提供することで、パーティ参加者のサービスに対する評価はより高くなるだろう。このような、エージェント間の連携も、協調行動として考えられる。

以上のような、エージェント間の協調行動をどのように実現するかが、ウェイター問題を解く上でのポイントになるだろう。

## 3. チップと契約の導入

本研究では、ウェイター問題において、エージェント間の協調を実現する手法として、“チップ”と“契約”という2概念を導入する。

### 3.1 チップ

エージェントがパーティ参加者にサービスを提供する際、パーティ参加者はそのサービスの内容に応じた額のチップをエージェントに支払う。具体的には、2.3節で述べた評価値に比例してチップ額を決定する。

チップを導入することで、エージェントはより高額のチップをもらうための行動を選択する、というエージェントの設計指針ができる。これにより、パーティ参加者によりよいサービスが提供されると考えられる。

### 3.2 契約

パーティ参加者とエージェントは互いに契約を結ぶことができる。これは、パーティ参加者にロボットが特定のお酒やおつまみを持って来る、その際に所定のチップ額を支払う、という内容である。パーティ参加者は、契約したお酒やおつまみをそのエージェントが持ってきた場合、優先的にその所有権を主張することができる。また、契約を破棄することもできる。この場合、前払いしたチップは戻らない。一方、エージェントはその契約を遵守しようとし、契約したお酒やおつまみを他のパーティ参加者に渡すことはしない。万一、契約が履行できない事態となったときは、その旨を通知し、前払いでもらったチップを返還する。

契約を結ぶメリットはパーティ参加者とエージェントの双方に存在する。パーティ参加者にとっては、複数契約からよりよいサービスを選択することが出来る。一方、エージェントにとっては、契約なしで持って行った場合に比して、チップの支払いが保証される。

### 3.3 パーティ参加者の行動

エージェントの協調行動を設計するためには、パーティ参加者がどのような行動をとるかや予め考察しておく必要がある。パーティ参加者がとる行動を、以下の4つの観点から考察した。

#### (1) エージェントに支払うチップ額

チップはパーティ参加者がよりよいサービスを提供してもらうための対価として導入されたものであり、それゆえパーティ参加者は、提供されたサービスの品質に合ったチップ額を決定する必要がある。ここで、サービスの品質とは、持って来るお酒やおつまみがどれだけ自分の好みと合っているか、およびそれを持ってくるタイミングの2点をさす。また、契約時に払われる前金については、支払いの段階でタイミングを予測することは難しいため、契約したお酒、おつまみがどれだけ自分の好みと合っているかという観点のみで評価するのが妥当であろう。

#### (2) エージェントに持ちかける契約の内容とタイミング

内容については、自分の好みのものを要請することが多いだろうが、なかなかサービスが実行されなければ「何でもいから」という内容になるだろう。また、タイミングとしては、常に持ちかける、手持ちがなくなったときのみ持ちかける、といったことが考えられる。

#### (3) エージェントから契約を持ちかけられたときの対処

サービスの内容に依存するが、諾否の基準として、一度に結ぶ契約の総数を制限するか、契約の評価が一定以上ならば必ず受け入れるかという2つが考えられる。ここで、契約の評価基準には、契約したサービス内容におけるチップの支払額を用いればよいだろう。

#### (4) 契約外のお酒やおつまみを持ってこられたときの対処

必ず受け入れる／受け入れないの他に、ある基準以上のサービス内容（前述のチップ額で評価可能）であれば受け入れる、また契約している場合には、提携中の契約より内容がよければ受け入れる、あるいは提携中の契約がなかなか完了しそうになれば受け入れる、といったことが考えられる。

### 3.4 エージェントの行動

エージェントの行動は大きく分けて「契約行動」「移動」「品物獲得」「品物譲渡」の4つに分けることが出来る。それぞれの行動の内容を決定する要因は「手持ちの品物の有無」「契約の内容」「契約を持ちかけるパーティ参加者」「カウンタの品揃え」「チップ額」「他エージェントの行動」がある。

いま、契約はエージェントがパーティ参加者に持ちかけるのみとし、また両者の位置に関係なく瞬時に結べるものとする。このとき、各状態におけるエージェントの行動は、図1のようになる。

発動条件	行動	エージェントの状態	行動内容
パーティ参加者との契約なし	契約	(条件なし)	一番好みの品を契約
			それ以外の品を契約
			契約しない
常時	移動	手持ちなし	カウンタへ
		手持ちあり、契約なし	一番近いパーティ参加者へ
		手持ちあり、契約不一致	契約外のパーティ参加者へ
		手持ちあり、契約と一致	契約したパーティ参加者へ
カウンタの近くにいる	品物獲得	契約あり、品物揃い	契約の品物をとる
		契約あり、品物不揃い	揃うまで待つ
		契約なし	置いてある品物をとる
パーティ参加者の近く、品物あり	品物譲渡	契約あり	契約の品を渡す
		契約なし	一番好みの品を渡す

図 1: ウェイターエージェントの行動分類

この図より、複数の行動が選択肢として考えられる状態が存在することが分かる。このような状態でも、候補のうちどの行動をとるかやエージェントごとに決定しておき、実際に得られた解から最適な行動を判断する、という方法が考えられる。

## 4. InViWo プラットフォームを用いた実装

前章までで述べてきたウェイター問題を計算機上で実装する方法について述べる。

### 4.1 InViWo プラットフォーム

InViWo プラットフォームは Richard によって開発された、エージェント間のインタラクションを簡易に記述できるマルチエージェントのシミュレータである [Richard 2001]。InViWo プラットフォームでは、エージェントの知覚器、行動決定器、行動器をモジュール化し、Marvin Language という記述言語により、それらのモジュールを簡易に記述できるようになっている (図 2)。

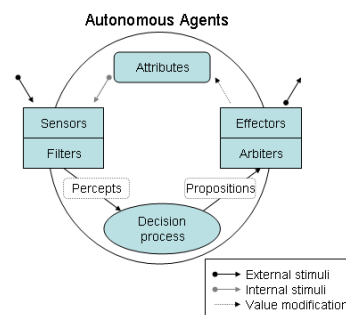


図 2: InViWo における自律エージェントの構成

## 4.2 実装におけるアプローチ

現在は、InViWoプラットフォームにてウェイター問題の実装を行っている段階であり、一刻も早く実装を終え、提案手法の評価、考察を行う予定である。

ウェイター問題の実装を困難にしている点としては、主に以下の2点が挙げられるが、各項目でそれぞれ述べたような解決法をとることにしている。

### (1) 「契約行動」の生起するタイミングの不確定性

エージェントの一連のサービス行動において、「移動→品物獲得→移動→品物譲渡」という行動遷移は自明であるが、このどこに契約行動が入るかは確定しない。この点で自由度が非常に高くなるが、本研究では、パーティ参加者は常に誰かと契約をしていることが望ましいと考え、契約終了時に次の契約を結ぶことで、「契約行動」の位置を固定化する。

### (2) 有限時間内にサービスを行わなければならないという要請

従来の協調問題では、できるだけ早い問題解決が望まれるものの、問題解決にかかる時間は最悪で無限時間を想定していた。しかし本問題では、有限時間内にサービスを行わなければならないという要請があり、エージェントは資源が動的に生成されるなかでこの制約を満たされなければならない。このため、例えば、サービス終了までの時間が短い場合、エージェントは「待つ」という行動を行わない、というようなヒューリスティックな戦略を導入する必要がある。

## 5. おわりに

本稿では、複雑な環境におけるマルチエージェント協調の標準問題としてウェイター問題を定義した。その上で、チップと契約という2概念を導入してエージェントを協調する方式を設計し、InViWoプラットフォームにてウェイター問題を実装する手法について述べた。

## 参考文献

- [Benda 1985] Benda M., Jagannathan V. and Dodhiawalla R.: On Optimal Cooperation of Knowledge Sources, Technical Report BCS-G 2010-28, Boeing AI Center, 1985.
- [Kitano 2000] H. Kitano: Robocup rescue: A grand challenge for multi-agent system. In Proceedings of the Third International Conference on Multi-Agent Systems(ICMAS-2000), 2000.
- [Richard 2001] Nadine Richard, Philippe Codognet, and Alain Grumbach: The InViWo Toolkit: Describing Autonomous Virtual Agents and Avatars. Proceedings of IVA'01(Intelligent Virtual Agents) workshop, 2001.
- [RoboCup Web] RoboCup Webpage:  
<http://www.robocup.org/>