

多論点交渉問題における

エージェントの公開情報の調整に基づく交渉手段 Using Autonomous Threshold Adjustment to Enable Multi-Party Negotiations with Multiple Interdependent Issues

藤田 桂英†
Katsuhide Fujita

伊藤 孝行†
Takayuki Ito

1. まえがき

実世界の交渉問題では、複数の項目が論点として存在することから、マルチエージェント研究分野においても、多論点交渉問題が重要な研究課題となってきた。複数の論点を対象とした研究はこれまでに行われているが([1]~[4])既存の研究では論点の独立性が仮定されており、エージェントの効用は線形の効用関数として表現可能であった。しかし実世界の問題では複数の論点が全て独立していることは稀であり、複数の論点が相互依存関係にある場合が多い。そこで非線形関数をもつエージェントのためのオークションを基とした交渉プロトコルが提案されている[5]。まず、エージェントは自分の効用空間においてサンプリングを行い、調整して局所解を発見する。そして、それをもとにエージェントの効用の大きい領域に対して入札の生成を行う。その後、メディエータが全ての入札の組み合わせを考慮し、最終的な合意案を発見する手法である。

しかし、この手法に対してエージェントの個人効用の公開については議論されていなかった。本来、エージェントの個人効用はできる限りメディエータに対して公開されるべきではない。本論文では入札の際の閾値調整を行うことで公開範囲を調整し、交渉する手法を提案する。まず、エージェントは共通の初期閾値を用いて、既存のプロトコル[5]で交渉を行う。その後、メディエータは各エージェントにどれだけ自分の効用の情報を公開しているかに基づいて下げ幅を指定し、閾値を下げるように命令する。そして、各エージェントはその閾値を用いて再び交渉を行い、解が発見される、もしくは解がないものと判断されるまでこの操作を繰り返す。

本論文の構成を以下に示す。2. ではそれぞれのエージェントがもつ非線形で多論点を扱う効用空間と既存の交渉手法について述べる。次に3. では閾値調整を用いてエージェントが個人の効用情報を公開しないようにする方法を提案する。そして4. で実験的に提案する手法の評価を行い、最後に今後の課題について議論する。

2. 非線形効用関数に基づく交渉

2.1 複雑な効用空間

本論文では、 N 個のエージェントが合意形成を試みる交渉の状況を考える。論点が M 個存在し、個々の論点を $i_j \in I$ と表す。論点 i_j は $[0, X]$ の範囲の整数を値として持つ。

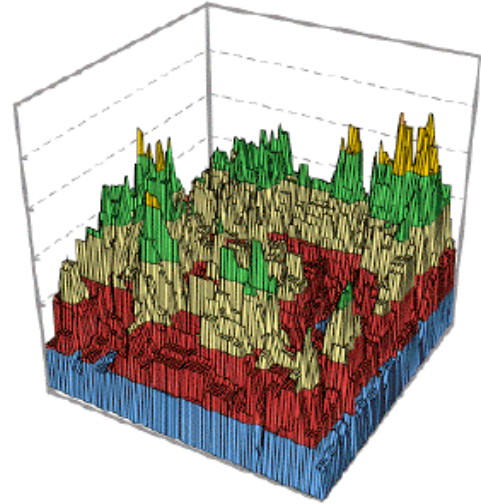


図1. 効用空間の例

交渉の結果得られる合意案は、各値のベクトル $\vec{s} = (s_1, \dots, s_M)$ として表現される。

エージェントの効用関数は制約を用いて表現する。 l 個の制約が存在するとし、個々の制約は $c_k \in C$ と表す。制約は、単一、もしくは複数の次元(論点)に関して制約充足条件となる値の範囲、および効用値を持つ。交渉に参加する全てのエージェントは、個々にユニークな制約集合を持っている。

合意 \vec{s} に関するエージェントの効用を $u_i(\vec{s}) = \sum_{c_k \in C, \vec{s} \in x(c_k)} w_i(c_k, \vec{s})$ と定義する。ここで、 $x(c_k)$ は制約 c_k を充足可能な合意案の集合である。この効用表現により、凹凸のある非線形の効用空間が形成される。本論文での効用空間とは、各論点を取りうる値のあらゆる組み合わせについて、効用関数によって得られる効用値を空間状にプロットして得られるグラフを意味し、空間の次元数は論点数+1となる。図1に非線形の効用関数の例を示す。この効用空間では2つの論点を対象としており3次元グラフとなる。また各論点の値域は $[0, 99]$ で、単項制約(1つの論点しか関係しない制約)、二項制約(2つの論点に関係する制約)がそれぞれ50個と100個あるとする。図に示す通り、非線形の効用空間は山と谷が入り組んだ複雑なものとなる。

† 名古屋工業大学 (Nagoya Institute of Technology)

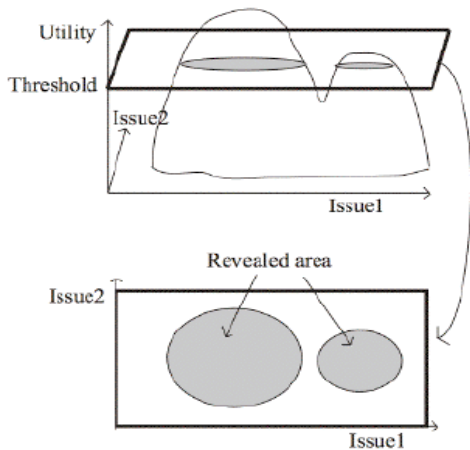


図2. 公開範囲

2.2 オークションに基づく交渉プロトコル

オークションに基づく交渉プロトコルは以下の4つのステップからなる。

[Step1: サンプリング] 効用の高い合意案が存在する範囲を特定するために、エージェントが、それぞれの効用空間でサンプリングを行う。エージェントは一様分布に従う乱数に基づいてあらかじめ決まった数の合意案候補を取得する。ここで取得するサンプル数が少なすぎる場合、エージェントが高効用のサンプルを見逃してしまい、不十分な効用しか得られない可能性があることに注意が必要である。

[Step2: サンプリング調整] Step1 で得られたサンプルが局所最適解を的確に取られていることが保証されない。そこで、各エージェントはシミュレーテッドアニーリング [6] に基づく非線形の最適化を行い、サンプリングポイント周辺の局所最適解の発見を試みる。

[Step3: 入札の生成] エージェントは、サンプルを調整して得られた各合意案 \vec{s} に関して効用を計算する。ここで効用とは合意案 \vec{s} を充足する制約の効用の総和である。効用が閾値を越えている場合に限りエージェントは入札を生成する。生成する入札は合意案 \vec{s} と同一の効用が得られる周辺全てをカバーするように生成される。

[Step4: Winner Determination] メディエータは全ての入札の組み合わせを考慮し、最終的な合意を決定する。具体的には、まず、各入札が持つ論点に関する値の範囲の共通部分を求める。ここでは各エージェントにつき1つの入札を採用し、互いに無矛盾であるものとする。2つ以上の合意案が見つかった場合、メディエータはそれらの合意案の中で最も効用が高いものを最終的な合意として選択する。メディエータは社会効用を最大化する入札の組み合わせを見つける際に枝刈り付きの幅優先探索を行う。

このアプローチを用いることで、最適解の発見を保証できることは容易に説明できる。全てのエージェントが網羅

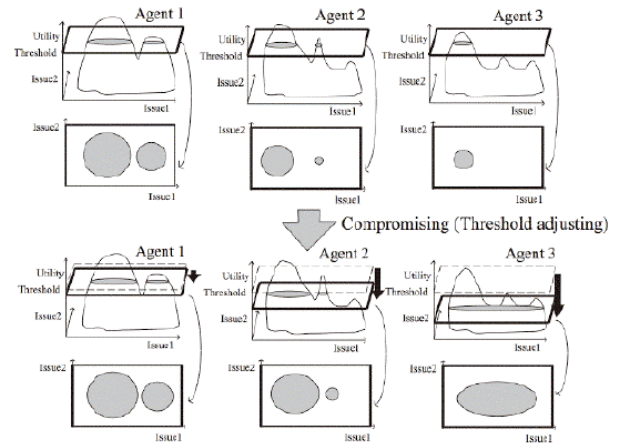


図3. 閾値調整プロセス

的にサンプリングを行い、かつ閾値を0に設定しているのならば、エージェントは自身の全ての効用空間を入札によって表現することができる。そして全てのエージェントはメディエータに入札を渡し、それを用いて網羅的な探索を行うので、最適解の発見は保証されるのである。しかし、入札生成と最適解の発見の際の計算コストは効用空間のサイズに依存する。現実的な適用を考えた場合、すべての空間を対象としてしまうと計算機上で実行できる計算コストを超えてしまう。そこでサンプル数を抑え、エージェントが生成する入札数に制限を設けることで現実的な範囲で停止させる必要がある。つまりこのアプローチが現実的に適用可能なのは、問題空間つまり探索空間が非常に小さい時のみであり、理想的な場合のみといえるのである。

これまでの手法 [5] ではすべてのエージェントの閾値はメディエータの指定された同一の数値であった。しかし、今から提案する閾値調整プロトコルを用いた場合は他のエージェントを考慮したエージェントごとに異なる数値の閾値となる。

3. 閾値調整メカニズム

このメカニズムの前提として多くの範囲をメディエータに公開しているエージェントはあまり多くの範囲を公開していないエージェントに対してより多くの範囲を公開するように説得を行えるという考えがある。つまり、もし合意に達しなかった場合は、公開範囲の少ないエージェントは合意が形成できるように、より多くの範囲を公開するようすべきといえる。ここでいう公開範囲とは図2が示すように、ある閾値においてどれほどエージェント自身が持つ効用空間をメディエータに対して公開したかである。

図3に3エージェントにおける閾値調整のプロセスを示している。矢印の上の部分が閾値調整をする前の閾値と公開範囲を示していて、下の部分が閾値調整した後の閾値と公開範囲を示している。この場合、閾値調整をする前の公開範囲が狭い Agent 3 に注目してほしい。閾値調整をした後 Agent 3 の公開範囲は3エージェントの中で最大となっている。このように公開範囲が少ないエージェントは閾値の下げ幅を調整することで、次のラウンドでは他のエージェントより大きな範囲を公開することになるのである。閾値調整の際の下げ幅は全体のうち個々のエージェントがどれくらいの割合を公開しているかを基にして、メディエータが決定する。そして、エージェント間で合意が形成され

るかもしくは合意が発見できないと判断されるまでこの操作が繰り返される。

閾値調整メカニズムの詳細を次に示す：

`bid_generation_with_SA()` は前章の Step1~3 を行う関数。
`search_solution()` は前章の Step4 操作を行う関数である。

Ar : Area Range of each agent ($Ar = \{Ar_0, Ar_1, \dots, Ar_n\}$)

```

1. procedure threshold_adjustment()
2. loop:
3.    $i := 1, B := \emptyset$ 
4.   while  $i < |Ag|$  do
5.     bid_generation_with_SA(Thi, V, SN, T, Bi)
6.      $maxSolution := search\_solution(B)$ 
7.     if find  $maxSolution$ 
8.       break loop
9.     elseif all agent can lower the threshold
10.     $i := 1$ 
11.     $SumAr := \sum_{i \in |Ag|} Ar_i$ 
12.    while  $i < |Ag|$  do
13.       $Th_i := Th_i - C * (SumAr - Ar_i) / SumAr$ 
14.       $i := i + 1$ 
15.    end while
16.  else
17.    break loop
18.  return  $maxSolution$ 

```

上のアルゴリズムは前章での Step1, Step2, Step3, Step4 を利用している。かつての論文[5]ではこれらのステップを何度も繰り返すことはしなかったが、この閾値調整を用いた手法では何度もこれらのステップを繰り返すことを新たに提案した。

4. 評価実験

4.1 実験の設定

本実験では、ランダムに生成された効用関数をエージェント間の交渉を 100 回試行した結果の平均値を取る。

最適率を求める実験で、網羅的な探索を行った場合、問題がスケールアップするにつれて計算量的困難が生じる。そこで、最適値に近似した値を求めるために、シミュレーテッドアニーリング(SA)を用いることとし、比較の為の基準として用いる。SA は初期の温度を 50 度として、2500 回の繰り返し処理を経て温度を 0 まで下げることとし、初期解はランダムに選択する。

また、公開範囲の実験では、効用空間における格子状の 1 点の効用値を公開した時、1 だけ公開したと指標を定義して評価を行った。つまりもし 100 点公開したら 100 だけ

効用を公開したということとなる。

本実験におけるパラメータは以下の通りである

- エージェント数：3
- 論点数：2~10
- 論点の値域：[0, 9]
- 制約数：10 (単項制約)、5 (二項制約)、5 (三項制約)。すなわち、単行制約を除く各次元の制約数は 5
- 制約の最大効用：100 × (論点数)。従って多くの論点に関して条件を満たす制約は、平均して、効用がより高くなる。本設定は、多くの問題領域において妥当だと考えられている。例えばミーティングスケジューリングにおいて、各論点を、個々の参加者のスケジューリングの可否として定義した場合、より多くの参加者に関する制約が、より少数の参加者に関する制約より重要とするのは、自然な設定である。
- 制約の最大範囲：7。この設定の下では、例えば、以下の制約が妥当なものとして生成される。(論点 1, 論点 2, 論点 3) = ([2, 6], [2, 9], [1, 3])。
- サンプル数：(論点数) × 200
- 個々のエージェントがサンプリングの際に行うシミュレーテッドアニーリングでは初期の温度を 30 とし、処理を 30 回繰り返す。サンプリング時のシミュレーテッドアニーリングの目的はサンプリングポイント周辺の(局所的な)最適解を得ることである。そのため、探索範囲が拡大しすぎないように初期の温度設定を低めにし、処理回数も少なく抑えてある。
- 閾値調整を使用した方法では初期の閾値を 900 として下げていき、閾値が 200 になったら、それ以降は下げない。
- 閾値調整を行わない方法では入札の際の閾値を 200 としている。
- 閾値調整を行う方法では下げ幅決める式として $50 * (SumAr - Ar_i) / SumAr$ を使用した。($SumAr$: 全エージェントの公開範囲の和、 Ar_i : エージェント i の公開範囲)
- 最適率を求める実験における閾値調整を行わない手法と、公開範囲に関する実験の閾値調整を行わない手法の数の制限をかけている場合は 1 エージェント辺りの上限を $\sqrt[3]{6400000}$ (N: エージェント数) とする。これは全てのエージェントの入札数の積が計算機上で行える限界 640000 を越えないようにするためである。また、公開範囲を求める実験では論点数が 6 以上になると計算機上で行える限界を超えてしまうため行う場合はできない。そのため制限がない場合の数値は理想的な場合として実験している。
- 本実験のためのプログラムは JAVA2(1.5) で記述し、Mac OS X 10.4 が動作している iMac(Core2Duo 2.33GHz・メモリ 1.5GB) 上で実験を行った。

4.2 実験の設定

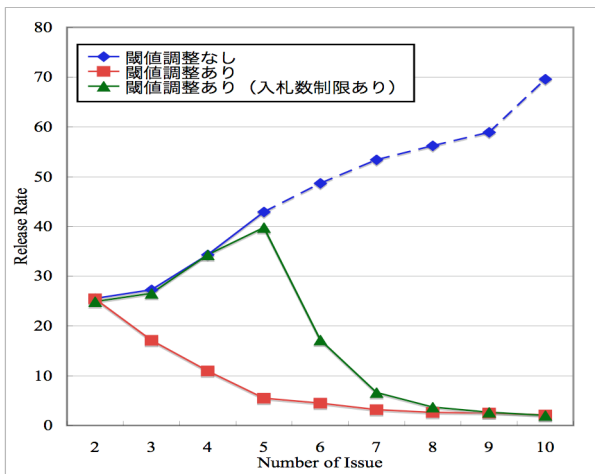


図4. 公開範囲

図4を見てほしい。図4は入札数の制限なしで閾値調整を行ったグラフ、入札数の制限ありで閾値調整を行ったグラフ、そして閾値調整を行った場合のグラフが示されている。またこのグラフでは(公開率) = (公開範囲)/(全体の効用空間の範囲)を比較の方法として用いた。

まず入札数の制限なしで閾値調整を行っていない手法に注目してほしい。この手法では論点数の増加に従って公開率が増加している。これから分かるようにもし入札数の制限や閾値調整を行わないと他の方法に比べて無駄に情報を公開していることが分かる。

次に入札数制限ありで閾値調整を行っている手法に注目してほしい。このグラフでは論点数5より大きくなると公開率が下がっている。これは論点数5以上になると入札数が制限されるからである。しかし、閾値調整をした場合に比べて論点数が3〜7の範囲では無駄に自分の効用の情報を公開していることが分かる。

一方閾値調整を実施した場合、公開率は他の閾値調整を実施しない方法に比べて押さえられている。このように閾値調整を行うことで無駄に自分の効用情報を公開することを防ぎながら交渉を行えることが分かる。

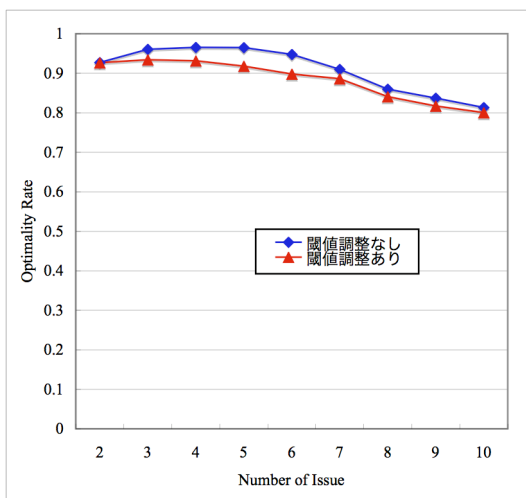


図5. 最適率

次に図5を見てほしい。図5においては閾値調整を行わない場合と閾値調整を行う場合の最適率の比較を行っている。この最適率のグラフから閾値調整を行った場合と行わない場合では差はほとんど無いといえる。しかし論点数が3〜7周辺では閾値調整を行った場合の最適率が少し下がっている。これはあるエージェントが高い効用を持ち、その部分で他のエージェントが低い効用を持つ場合、閾値調整をしなければ発見できるが、閾値調整をすることでその組み合わせを見逃してしまうからである。しかし、このようになる場合はあまり多くなく、そのときの効用もあまり大きくないため最適率に大きな差がでない。

これらの実験から閾値調整を行う手法を用いることで最適率の大きな変化なしにエージェントの公開範囲を最低限にすることが可能であることが分かった。

5. おわりに

本論文では公開範囲調整に基づくマルチエージェントにおける多論点交渉問題を提案した。人の代わりとして動作するエージェントにとって自分の効用をできる限り公開しないことはより現実的な設定である。そこで閾値を調整し最適率を大きく下げることなく公開範囲を最低限にすることができた。

今後の課題として交渉の成功率の増加があげられる。閾値調整の手法を用いた時の成功率は閾値調整を行わない場合の閾値と交渉をやめると判断する閾値が同じならば全く同等である。しかし、エージェント数や論点数が増加した場合の成功率はあまり高くなく、解が存在しうるのに解が発見できない場合が起こる。このような場合を減らす手法の提案が必要といえる。

文献

- [1] P.Faratin, C.Sierra and N.R. Jennings: "Using similarity criteria to make issue trade-offs in automated negotiations", *Artificial Intelligence*, pp. 142:205--237 (2002).
- [2] L.K. Soh and X.Li: "Adaptive, confidence-based multiagent negotiation strategy", *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems(AAMAS2004)* (2004).
- [3] S.Fatima, M.Wooldridge and N.R. Jennings: "Optimal negotiation of multiple issues in incomplete information settings", *Proc. of Autonomous Agents and Multi-Agent Systems (AAMAS2004)* (2004).
- [4] R.Y.K.Lau: "Towards genetically optimised multi-agent multi-issue negotiations", *Proceedings of the 38th Annual Hawaii International Conference on System Sciences (HICSS05)* (2005).
- [5] T.Ito, H.Hattori, and M.Klein. Multi-issue negotiation protocol for agents: Exploring nonlinear utility spaces In *JCAI-2007*, pp.1347-1352, 2007
- [6] S.J.Russell and P.Norving: "Artificial Intelligence : A Modern Approach", Prentice Hall(2002)