

エージェント間自動交渉に基づく集約的共同デザイン支援システムの試作

奥村 命†

藤田 桂英†, ††

伊藤 孝行†, †††, ††††

†名古屋工業大学 情報工学科/産業戦略工学専攻

††MIT スローン経営大学院

†††東京大学政策ビジョン研究センター

†††† 科学技術振興機構さきがけ

1 はじめに

近年、地球温暖化問題対策、多国籍企業のグローバルな商品開発など、世界中から多くの人が同時に参加して議論をする必要がある場面が増えている。大規模な数の意見を集約し、合意を形成するためには、自動交渉エージェントによる効率的なコラボレーション支援が効果的に考えられる。しかし、マルチエージェントの自動交渉機構に関する多くの既存研究 [1, 2] では、エージェントの完全な効用空間が仮定されているが、現実世界ではユーザーの効用空間を完全に取得することは困難である。本論文では、ユーザーの効用空間抽出機構を提案し、マルチエージェントの自動交渉機構に基づいたコラボレーティブな合意形成支援システムを試作する。一例として、公園デザインの作成を支援するシステムを取りあげる。

2 関連研究

複数論点交渉問題の合意形成支援に関する多くの既存研究 [1, 2] では効用空間を仮定しているため、効用空間抽出機構が必要である。本論文ではユーザーの効用空間を抽出する機構を提案する。

効用空間の抽出機構として、文献 [3] では、トレードオフとなる属性を取り出す方法を提案している。システムは異なる2つの属性間に存在するトレードオフ関係をユーザーと対話することで抽出し、数値化する。しかし、[3]の手法では、3つ以上の複数の属性間に依存関係が存在する場合や、属性間依存が非線形である場合などの複雑な効用空間の抽出を行うことができない。本手法では、このような複雑な効用空間の抽出を行うことができる。

3 コラボレーティブな公園設計支援システム

本論文では、コラボレーティブな合意形成支援システムの例として、公園デザインの共同設計支援システムを試作開発する。本システムの基本ステップは以下の [Step1]~[Step4] からなる。

[Step1] ユーザーの意見、好みの収集機構

本ステップでは、サンプリング点を選出し、対応する設計モデル（公園デザイン）に対するユーザー評価を得ることで、ユーザーの好みを抽出する。

[Step2] ユーザーの効用空間形成機構

4章で述べる効用空間形成手法を用い、[Step1]で得たサンプリングから各ユーザーの好みを表す効用空間を形成する。

[Step3] エージェントによる自動交渉機構

[Step2]で抽出した各ユーザーの効用空間を、ユーザー

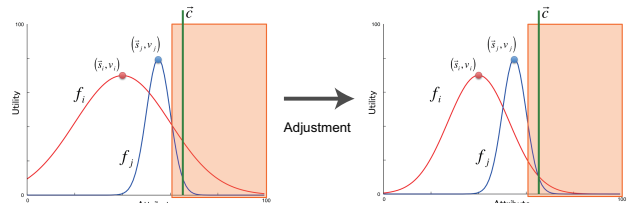


図1: 基本関数が別の基本関数の下側にある状態 (ケース1)

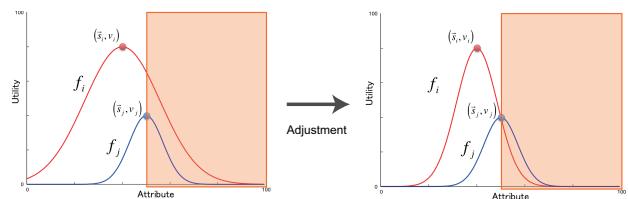


図2: 基本関数の大部分が別の基本関数の下側にある状態 (ケース2)

の代理となって自動交渉を行うエージェントに与え、自動交渉技術 [1, 2] を用い、ユーザー達にとって好ましい合意案を発見する。

[Step4] 合意案提案機構

得られた合意案（属性空間上の点）に対応する設計モデルを生成し、ユーザーへ提出する。

4 効用空間抽出機構

本章で提案する効用空間抽出機構は、3章で説明した設計支援システムの [Step1], [Step2] に相当する。本機構では設計モデルをプログラムにより生成し、生成された設計モデルに対するユーザー評価を得ることにより、ユーザーの好みを表す効用空間を抽出する。

本論文で提案する効用空間の形成手法では、ベースとなる単純な形状をした基本関数を用いて、複雑な効用空間を表す関数を表す。サンプリング点における効用値をユーザー評価値とし、サンプリング点から離れるほど効用値が減少するような形状をした基本関数を用意する。サンプリング点で発生した基本関数の山を組み合わせることで、効用空間全体の複雑な凹凸形状を表現する。今回は基本関数としてガウス関数を採用した。i番目のサンプリング点が s_i で、 s_i に対するユーザー評価値が v_i であるとき、 s_i に対応する基本関数 f_i を式 (1) のように定義する。なお、 d_i は f_i の広がり具合を表すパラメータである。

$$f_i(\vec{x}) = v_i \cdot \exp\left(-\frac{(\vec{x} - \vec{s}_i)^2}{d_i}\right) \quad (1)$$

サンプリング点から効用空間を形成する最も単純な方法は、サンプリング点を滑らかに繋ぐ方法である。しかし、サンプリング点間の距離が大きい場合、サンプリング点の間で効用値を高く評価してしまう可能性がある。本手法では、基本関数の最大値を取ることで、本

†Mikoto OKUMURA †, ††Katsuhide FUJITA †, †††, ††††Takayuki ITO
 †Dept. of Computer Science/Master of Techno-Business Administration, Nagoya Institute of Technology
 ††Sloan School of Management, Massachusetts Institute of Technology
 †††Todai Policy Alternatives Research Institute
 ††††JST PREST

問題を解決する。\$N\$ 個のサンプリング点 \$s_1, \dots, s_N\$ が得られたとき、効用関数 \$U\$ を式 (2) のように定義する。

$$U(\vec{x}) = \max_{i=1, \dots, N} f_i(\vec{x}) \quad (2)$$

本手法ではサンプリング点から離れた点、即ち情報が不確かな点では、効用値が低くなるように設計しているため、サンプリングが不足している点で合意を形成する可能性を抑制することができる。一方、サンプリングが不足している段階では合意形成が難しいという問題がある。しかし、多くの交渉相手が合意を求めると点をサンプリングすることで本問題は解決する。

単純に式 (2) によって効用関数 \$U\$ を求めた場合、図 1, 図 2 に示す問題が発生してしまう。どちらの例でも、図中の矩形領域では、サンプリング点 \$s_i\$ よりも \$s_j\$ に近い点では、\$f_j\$ の値を採用すべきであるが、効用関数では \$f_i\$ が採用されてしまう。以下に示す 2 手法では、基本関数 \$f_i\$ の広がり具合を表す変数 \$d_i\$ を変更することで、本問題を解決する。

【手法 1】ケース 1: 基本関数が別の基本関数の下側にある状態 (図 1) を解決する。

本手法では、図 1 中の左側のような状態では、\$d_i\$ を調節することで、図 1 の中の右側のように修正する。具体的には、2つの異なるサンプリング点 \$s_i, s_j\$ について、\$f_i(s_j) > f_j(s_j)\$ ならば、式 (3) によって \$f_i(s_j) = f_j(s_j)\$ となるように \$d_i\$ を変更する (\$i \neq j, \max f_i(\vec{x}) \ge \max f_j(\vec{x})\$)。

$$d_i = \frac{(s_j^i - s_i^i)^2}{\ln \frac{v_i}{v_j}} \quad (3)$$

【手法 2】ケース 2: 基本関数の大部分が別の基本関数の下側にある状態 (図 2) を解決する。

本手法では、図 2 中の左側のような状態では、\$d_i\$ を調節することで、図 2 の右側のように修正する。具体的には、2つの異なるサンプリング点 \$s_i, s_j\$ について、\$f_i(\vec{c}) > f_j(\vec{c})\$ ならば、式 (4) によって \$f_i(\vec{c}) = f_j(\vec{c})\$ となるように \$d_i\$ を変更する (\$i \neq j, d_i > d_j\$)。

$$d_i = \frac{(\vec{c} - s_i^i)^2}{\frac{k}{2} - \ln \frac{v_i}{v_j}} \quad (4)$$

なお、\$\vec{c} = s_j + k \sqrt{\frac{d_j}{2}} \vec{u}\$、\$\vec{u} = \frac{1}{\|s_j - s_i\|} (s_j - s_i)\$ で、\$\vec{u}\$ は \$s_i\$ から \$s_j\$ へ向かう方向を表す単位ベクトルであり、\$\vec{c}\$ は変更の基準点を指す (\$k \in \mathbb{R}^{++}\$)。基準点 \$\vec{c}\$ はパラメータ \$k\$ によって変化し、\$k\$ が大きいほど \$s_j\$ から離れた点で本手法を実行する。例えば、図 2 の場合、\$k\$ が大きくなると \$\vec{c}\$ が右方向へ移動する。本論文で示すシステムでは、\$k = 2\$ としている。

本論文で提案する効用空間の抽出機構は、以下に示す [Step1]~[Step3] で構成される。

【Step1】サンプリングポイントの選出
ユーザーによる評価を行う属性空間上の点を選出する。今回はランダムにサンプリング点を選出する。

【Step2】ユーザーによる評価
選出した属性空間上の点に対応する設計モデルに対するユーザーからの評価を得る。評価方法としては、投票方式や採点方式などが考えられるが、今回は採点方式を採用し、0~100 の整数値で評価する。

【Step3】効用空間の形成
【Step2】により \$i\$ 番目に新しく得られた評価値から、基

本関数 \$f_i\$ を生成する。基本関数の広がり具合を表す変数 \$d_i\$ は \$D_0\$ で初期化される。新規基本関数 \$f_i\$ と既存の各基本関数 \$f_j\$ に対して、手法 1, 手法 2 を順に適用し、\$d_i, d_j\$ を調節する (\$0 \le j < i\$)。なお、本論文で示すシステムでは、\$D_0 = 500\$ としている。以降、[Step1]~[Step3] を予め決められた期間中繰り返す。

5 評価と考察

本システムの評価者実験を行った。属性は自然度と遊具充実度の 2 つを設定し、被験者 11 人で 2 度交渉を行った。各属性は 0~100 の実数値を取る。交渉時間をそれぞれ 1 度目は 10 分、2 度目は 5 分に設定した。2 度目の交渉時間が 1 度目よりも短い理由は、被験者がシステムの操作に慣れたためである。

交渉の結果は、1 度目は {自然度=44, 遊具充実度=90} で評価値平均が 79.09, 誤差平均が 15.78, 差分平均が 9.56 であった。2 度目は {自然度=51, 遊具充実度=85} で評価値平均が 79.36 で、誤差平均が 16.60 で、差分平均が 9.2 であった。評価値平均とは、被験者が交渉結果として得られたデザインに対して 0~100 の整数値で点数付けしたもので、4 章の [Step2] で説明したユーザーによる評価ステップと同様に評価する。誤差平均とは、各ユーザーの評価値と抽出した効用関数で交渉結果を評価した値との誤差の平均を取った値である。差分平均とは、各ユーザーの評価値から抽出した効用関数で交渉結果を評価した値を引いた値の平均を取った値である。

両交渉結果で高い評価値平均を得ており、多くのユーザーが納得する合意案を形成できたことが分かる。各交渉の誤差平均が 15.78, 16.60 であることから、効用空間抽出機構の精度は良くないものの、ユーザーの好みの傾向を抽出することはできていることが分かる。また、各交渉の差分平均が 9.56, 9.20 であることから、本効用空間抽出手法が全体的に効用値を低く推測していることが分かる。本効用空間抽出手法は、情報の不確かさを考慮し、サンプリングが不足している点で合意形成する可能性を抑制している。本方針はサンプリングが十分に得られた領域では、サンプリング点から離れても効用値の減少しない効用空間抽出手法を適応するなど、属性空間上のサンプリング点の分布を考慮した効用空間抽出手法を考える必要がある。

6 まとめ

本論文では、マルチエージェントの自動交渉機構に基づいたコラボレーティブな合意形成支援システムの一例として、公共空間設計支援システムを実装し、評価者実験により本システムが多くのユーザーが納得する合意案形成に有用であることを示した。

参考文献

- [1] S. Shaheen Fatima, Michael Wooldridge, and Nicholas R. Jennings. An analysis of feasible solutions for multi-issue negotiation involving nonlinear utility functions. In AAMAS (2), pages 1041-1048, 2009.
- [2] Ito, T., Hattori, H., and Klein, M.: Multi-issue Negotiation Protocol for Agents: Exploring Nonlinear Utility Spaces, in Proc. of IJCAI-2007, pp. 1347-1352 (2007)
- [3] X Luo, N. R. Jennings, and N. Shadbolt. Acquiring user strategies and preferences for negotiating agents: A default then adjust method. International Journal of Human Computer Studies, 64(4):304-321, 2006.