

複数エージェント間交渉のための Misrepresentation Game 実装フレームワークの試作

西 将宏^{1,a)} 福田 直樹^{2,b)}

概要: Gratch らの Misrepresentation game は、効用を相互に公表しながら公平な配分を行う交渉であるにもかかわらず、特定の条件下において嘘をつくことで正直な返答よりも効用を得られる場合のある、特殊な交渉ゲームの一つである。本稿では、マルチエージェントネゴシエーションにおける misrepresenting agent の実装や分析に適したフレームワークおよび関連するソフトウェアの試作について述べる。本フレームワークおよび試作ソフトウェアを用いることで、エージェント数や論点数、エージェントの交渉手法、対象とする問題領域の設定などを自由に組み替えて実行しその振る舞い等を解析することが容易になることが期待される。

NISHI MASAHIRO^{1,a)} FUKUTA NAOKI^{2,b)}

1. はじめに

Gratch らの Misrepresentation game [1] は、効用を相互に公表しながら公平な配分を行う交渉であるにもかかわらず、特定の条件下において嘘をつくことで正直な返答よりも効用を得られる場合のある、特殊な交渉ゲームの一つである。本稿では、この Misrepresentation game を解析するソフトウェアフレームワークの試作について述べる。Misrepresentation game となる交渉を複数者間で行う場合に、Misrepresenting agent が複数存在するとき起こる問題や、新たな Misrepresenting の手法を発見するための分析支援を目指す。

2. Misrepresentation game

文献 [1] によれば、Misrepresentation game は、Lu と Boutilier らの、選好公開型の公平な財の分割手法であると考えられていた Preference-elicitation を用いた交渉 [3] において、fixed-pie lie と呼ばれる手法を用いることで、正直な申告と比較して高い効用を得られる場合があるという点に着目し、一見公平に見える財の分割方法における正直で

ない回答による利益追求問題である。

文献 [3] における Preference-elicitation は minimax-regret の方針によって考え出された手法であり、相手に選好を質問することで交渉を進めていく。Preference-elicitation に基づく交渉例を示す。ここでは、エージェント A とエージェント B が、りんごとバナナとメロンの 3 種類の財の割当て (論点数:3) を交渉すると仮定する。ランダムに選ばれたエージェント (ここでは便宜上エージェント A が選ばれたとする) が、エージェント B に対して論点を比較する (たとえば、バナナとメロンのどちらが好きか? という質問を行う) 方法で選好を尋ねる。ここで、論点を比較するのは、人間にとっては、ある論点に対して自分が得られる効用を明確に表現するのが必ずしも可能とは限らないからである [3]。論点比較により選好を尋ねられたエージェント B は、質問に対して、論点に対する優先順位を (たとえば、メロンよりバナナが好き、のように) 返答をする。エージェント B の返答を確認した後、エージェント A は自分の選好を公開する。エージェント A の質問が終わると、今度はエージェント B がエージェント A に対して同じように質問を行う。このように選好を相互に公開していき、十分な数の選好が公開された段階で、両エージェントに対して公平な割り当てを計算し、最適な割り当てを行うことで交渉が終了する。

論点は有限であり、比較による選好公開もまた有限であるため、Preference-elicitation に基づく交渉は必ず終了す

¹ 静岡大学情報学部
Hamamatu, Shizuoka 432-8561, Japan

² 静岡大学大学院情報学領域
Hamamatu, Shizuoka 432-8561, Japan

a) cs13074@s.inf.shizuoka.ac.jp

b) fukuta@cs.inf.shizuoka.ac.jp

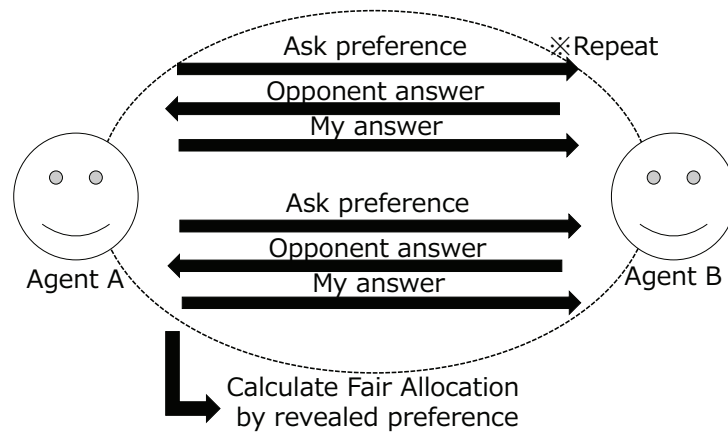


図 1 Preference-elicitation に基づく公平な財の分配プロトコル

ることが保証されている [3].

Misrepresentation game では、上記の交渉において、Misrepresenting agent が以上の流れにおける選好公開を偽ることで、本来得られる効用よりも多くの効用を得る。例えば、先ほどの例と同様に、バナナ、りんご、メロンの 3 論点が存在し、エージェント A とエージェント B (Misrepresenting agent) が存在すると仮定する。得られる効用は好みの順に 3,2,1 とする。Misrepresenting agent の選好はメロン > バナナ > りんごと仮定する。すなわち、バナナよりメロンが好きで、りんごよりバナナが好きである、という選好である。いまここで、あらかじめ一回選好を尋ね、相手はりんご > バナナという選好を持つことが分かっている。この時考えられる相手の選好としてはりんご > バナナ > メロン、りんご > メロン > バナナ、メロン > りんご > バナナの 3 つである。

例を図 2 に示す。ここで、Misrepresenting agent は fixed-pie lie という、相手と同じ選好を宣言する嘘を用いる [1]。すなわち、Misrepresenting agent は、fixed-pie lie に従って、自身の選好をりんご > バナナと申告し、申告結果に従って財を配分すると、相手の選好がりんご > バナナ > メロンの場合は、りんごを相手エージェントに渡し、残りをすべて Misrepresenting agent が受け取ることで、見かけ上はお互いに効用 3 を得る。しかし本来の選好はメロン > バナナ > りんごであるため、この Misrepresentation agent が得た効用は、メロンの 3 とバナナの 2 を合計した 5 となる。相手の選好がりんご > メロン > バナナの場合も同様の配分で、相手エージェントにりんごを渡し残りをすべて Misrepresenting agent が受け取ることで、見かけ上はお互いに効用 3 を得るが Misrepresenting agent の本来の選好は異なるため効用 5 を得る。相手の選好がメロン > りんご > バナナの場合は、相手にりんご 3/4 個、メロン 1/2 個を渡し、残りをすべて Misrepresenting agent が受け取ることで、見かけ上お互いに効用 3 を得る。しかし Misrepresenting agent の本来の選好で効用を計算する

と $3 \cdot 1/2 + 2 + 1 \cdot 1/4$ で 3.75 の効用を得ることが分かる。以上のように選好を偽ることで一方的に高い効用を得ることが Misrepresentation game の特徴的な性質である。

3. Alternating-offer protocol に基づく交渉分析フレームワークとの差異

3.1 GENIUS フレームワーク

複数者間交渉を分析するツールに、GENIUS がある [10]。GENIUS に用いられている中心的なプロトコルの 1 つである。Alternating-offer protocol は文献 [4], [5] で提唱されている二者間交渉のためのプロトコルであり、相手の提案に対して “Accept”, “Offer”, “EndNegotiation” の 3 つの行動のうち 1 つを選択し合意を図るプロトコルである。以下にそれぞれの行動について示す。

- Accept: 相手エージェントから提案された合意案を受け入れる行動である。通常、相手の合意案に対して評価関数を用い、得られる効用が受け入れるエージェントの閾値 (求める効用) 以上の時に選択する。二者間交渉においては、相手の提案に対しこの行動を選択することで合意を形成し交渉を終了する。
- Offer: 相手エージェントに対し合意案を提案する行動である。この行動を選択した時、既に提案されている相手エージェントの合意案は破棄される。
- EndNegotiation: エージェントが交渉そのものを放棄する行動である。二者間交渉においては、どちらかのエージェントがこの行動を選択した時、お互いのエージェントが得られる効用は 0 となり合意が形成されずに交渉が終了する。

エージェントは相手の提案に対し、以上に示した 3 つの行動のうち 1 つを選択することで合意を形成、または交渉そのものが放棄されるまで交渉を続ける。

Alternating-offer protocol 下における交渉の概略を図 3 に示す。エージェント A, エージェント B が交渉を行うと仮定する。最初にどちらかが合意案を提案する。ここでは、

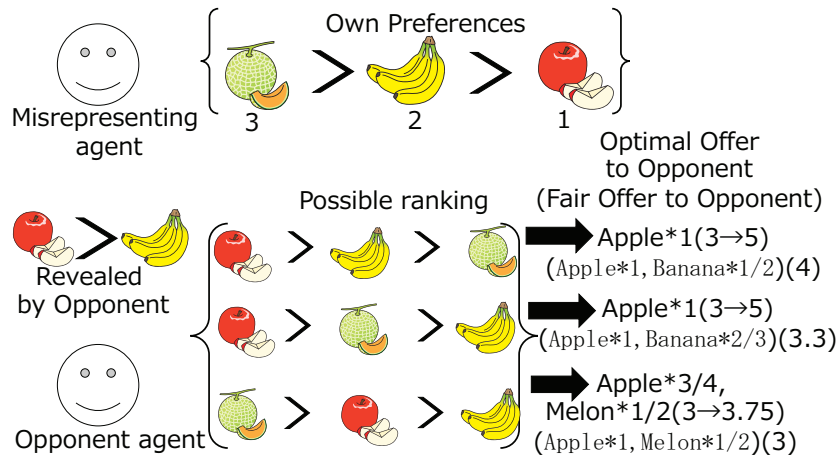


図 2 Misrepresentation game の例

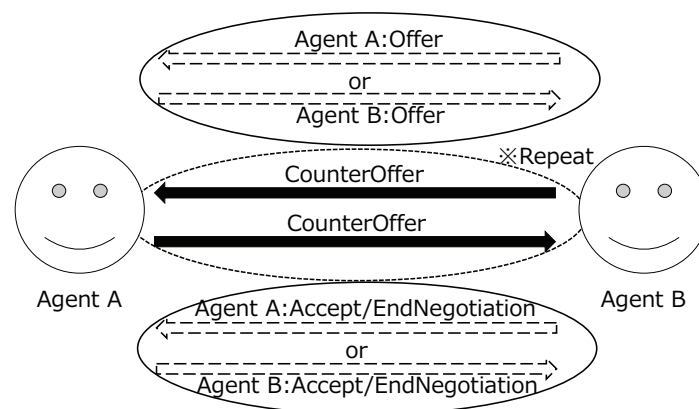


図 3 Alternating-offers protocol における交渉

どちらが最初に提案するかはランダムであるとする。説明の便宜上、エージェント A が最初に Offer を選択すると、評価関数を用いて提案から得られる効用を求める。ここで得られる効用が求める効用以上であれば Accept を選択し、そうでなければエージェント A に対して Offer を選択する。Accept を選択した場合は合意を形成し、エージェント A の提案から得られる効用をお互いのエージェントが受け取る。Offer を選択した場合はエージェント A の提案を破棄し、エージェント B が新しい合意案を提案する。この場合はエージェント B の提案に対し、エージェント A の行動選択に移る。Accept, Offer の他にも EndNegotiation を選択することもでき、この場合はお互いに得られる効用は 0 となり、交渉が終了する。これら一連の流れをどちらかが Accept または EndNegotiation を選択するまで続ける。ただしお互いに譲歩した提案を行わない場合を考慮して時間制限を設ける。時間制限が来た場合も EndNegotiation 同様、両者の得られる効用は 0 となり合意が形成されずに交渉が終了する。

3.2 Misrepresentation game と Alternating-offer protocol における交渉の違い

Alternating-offer protocol[4], [5]では、交互に Offer を用いて合意を図るプロトコルである。Misrepresentation game では相手に尋ねる行為 (Ask とする) が Offer に当たる行動であるが、Offer が自分の選好を反映した提案を行うのに対し、Ask は質問のみであり自身の選好をその段階では一切公開しないところが異なる。また Alternating-protocol における EndNegotiation や Accept といった行動は Misrepresentation game には存在せず、公開された選好を元に場により公平な割り当てが決まり、それをもって合意が形成される。すなわち、Alternating-offer protocol における自身の合意案には自身の選好と相手の選好に対する妥協の両者が織り込まれており、自身の選好の提示が正直な申告であったか否かによる差異を検証するという必要性が、現状の GENIUS では本質的に存在しない。このため、本研究では、Misrepresentation game の分析に必要な要素を適切に扱うことができるフレームワークを提案する。

4. 本フレームワークの設計における検討事項

文献 [10] によると、自動交渉フレームワークのデザイン

に必要な要素としては、(1) 新しいエージェントのデザインをサポートし、交渉戦略の有効性についての洞察を提供すること、(2) さまざまな交渉戦略の構成要素の評価と組み合わせを容易にすること、(3) 我々に、既存の最先端のエージェントのデザインを異なる要素に分解することを可能にすること、の3つが挙げられており、これらの思想をもとに設計されたのが GENIUS である。

GENIUS は、General Environment for Negotiation with Intelligent multi-purpose Usage Simulation、つまりインテリジェントな多目的利用シミュレーションと交渉のための汎用環境を目指して作られたもので、自動交渉エージェントの国際競技会 ANAC(Automated Negotiating Agents Competition) に用いられてきた自動交渉エージェントフレームワークである。GENIUS は、効率的な自動交渉エージェントの設計そのものの単なる支援を中心的な検討課題にしてはならず、自動交渉エージェントの戦略の設計、及び評価を容易にするための環境として発表され、それまでの ANAC における上位進出エージェントの持つ戦略の各部分に対する意義を独立して検証するなどの点で、興味深い考察を示している [10]。GENIUS は以下要素から構成される。

- (a) 分析モジュール：分析モジュールは、異なる評価指標を使用して結果を分析できるようにする。
- (b) リポジトリ：リポジトリには、GENIUS に組み込まれた3つの分析モジュールに対応した、3つの異なる交渉のモジュールが含まれている。(1) 交渉シナリオは、問題領域において、少なくとも2つの選好プロフィールと交渉領域からなる。交渉シナリオが指定されている時、GENIUS はシナリオ上の問題空間の交渉結果を分析することができる。(2) 交渉エージェントは API を用いて実装する。エージェントは API を用いて、交渉環境に関する情報を得ることができる。(3) 交渉プロトコルは、1対1と複数間の両方である。GENIUS は、プロトコルに応じて、公平性や、social welfare などの交渉特性を評価するための交渉分析を提供することができる。
- (c) ロギング：交渉状況の記録を取る。
- (d) シミュレーションコントロール：シミュレーションコントロールとロギングモジュールは、研究者に、シミュレーションをコントロールすることや、そのデバッグ、及び詳細化された情報の入手を可能にする。

本フレームワークの目的は Misrepresentation game の分析支援であり、GENIUS における検討をもとに、分析支援をするために以下の目的を達成することを目標とする。

- シミュレーション：Misrepresentation game の発見、及び検証をする上では、網羅的なシミュレーションを行う必要がある。この網羅的検証を手手で計算するのは負担が大きいため、フレームワーク内に実装されて

いるのが理想であると考える。

- ロギング：Misrepresentation game は、どのような場合に正直な返答よりも利益が多く得られるのかを確認するために、どのように交渉が進んでいる場合に起こり得るかを検証する必要がある。そのためには、交渉のログを確認できれば分析を行いやすい。シミュレーション及びロギングは、文献 [10] の“新しいエージェントのデザインをサポートし、交渉戦略の有効性についての洞察を提供すること”の思想をもとに設計している。
- 問題設定：本フレームワークは、論点の数、エージェントの数、Misrepresentation agent の数、選好を自由に入力することができる。このように幅広い設定が可能なのは、Misrepresentation game が、どのような問題領域で発生するか、どの程度の一般性があるか、再現性があるかなどを検証するためである。これは文献 [10] の“さまざまな交渉戦略の構成要素の評価と組み合わせを容易にする”の思想をもとに設計している。

本フレームワークには、Misrepresentation game が可能かどうかを検証することはしても、その対策を提供はしない。これは、Misrepresentation game そのものが、正直に返答した時よりも高く効用を得られる場合があるのみで、必ずしも高い効用を得られるわけではないという性質や、文献 [1] においても書かれている、悪意のあるテクニックの理解やモデリングが、対策を提案するために必要な最初のステップであるからである。次に、これらの基本設計をもとに、分析支援のための更なるツールについて考察を行う。

5. 提案フレームワークの設計

本節では、本提案フレームワークの設計をするにあたって、必要となる機構について述べる。Preference-elicitation に基づいて交渉を行うエージェントの実装やその振る舞いの解析を容易にするためのフレームワークには、次の機構が必要となると考えられる。1つは、交渉場の制御機構である。エージェントがお互いに選好を尋ねる動作や公開された選好から割り当てを決める動作はエージェントの選好公開戦略そのものではなく、交渉の場そのものが計算すべきものという考え方もできる。このため、フレームワーク内にその実装を含め、共通の部品として使用することが好ましいと考えられる。

詳細な分析を行う際や、Misrepresentation agent を実装する際には、公開されている選好から考えられる(場として)公平な割り当てを計算する機構や、自分の返答によってどのように割り当てが変化するかを計算する機構を再利用できれば、分析のためのコーディングの負担を軽減できると考えられる。割り当てがどちらにどれくらい有利になるかを計算する機構を用意することで、Misrepresentation による利点が生じるような状況の発見を行うプログラムの

コーディングの容易化に役立つと考えられる。

以上の機構を用意することで、Misrepresenting agent の本質的な部分の実装以外について、フレームワーク内の実装を再利用しながら詳細な分析を行うことができるようにすることが、本フレームワークの設計において必要になると考えられる。

6. 実装に関する詳細

本節では、本フレームワークの実装の詳細について述べる。本フレームワークは、GENIUS と同じく、フレームワーク自身の提供する API に基づいて使用者が各自でエージェントプログラムを設計し、それを取り込んで実際の自動交渉を行わせるように設計しており、GENIUS に慣れた使用者がエージェントの設計を行いやすいように、Java で開発している。

本フレームワークは GUI 部分とシステム部分とエージェント間交渉実行部分に分かれている。GUI 部分は、CUI では直感的な操作が行いにくいような、インタラクティブな解析を行う場合のために用意しており、あらかじめ決められた手順での交渉の実行などではコマンドラインからの GUI を持たないプログラムとしての実行もできるようにしている。

フレームワークのメインとなる実行部分は、交渉そのものについての実行動作部分を実装しており、論点数やエージェントの数などの初期設定や、交渉の流れそのものを制御することが可能となっている。

エージェントの部分は、本フレームワークの提供する API およびひな形を利用することで、簡易なコードでエージェントを実装することができるような仕組みになっている。現状の実装は、選好の管理や、質問をされたときに返答するメソッドなどを実装した基本エージェントを用意し、自己中心的なエージェントや Misrepresentation agent はこれを継承して作成できるようにしている。

本フレームワークを用いることで、使用者は、この継承して作成されたエージェントのクラスに必要な応じてメソッドを追加することで任意のエージェントを作成することができる。

論点に関して、現在は、int 型の範囲 (2147483648 個まで) の論点数を扱うことができるようにしているが、実際に実行・解析可能な論点数に関しては、エージェントのコードの設計にも依存するため、現在検証を進めている。それぞれの論点に対しては選好を重みとして設定でき、論点に対する重みが大きいほど、高い選好であることを意味する。

エージェントとしては、利己的な正直エージェントと Misrepresentation エージェントを 1 体ずつ選択して交渉を行う部分を実装の出発点としている。現在は二者間での交渉を想定した設計になっており、三者間以上の交渉では、別途プロトコルを拡張することで実現される。

交渉部分は、Preference-elicitation に基づく財の公平配分プロトコルである。交渉結果としては、明らかになった選好を表示すること、配分の結果としての各自の効用を表示可能であり、Misrepresentation game となるような、本来の交渉と比べて多くの効用を得ることができているかについての分析は、Misrepresentation agent のかわりに正直なエージェントを用いて実行させた結果との比較により可能となる。

7. 試作フレームワークの実行例

図 4 は、試作中の本フレームワークにおいて Misrepresentation game となる財の公平割当て交渉をさせた例である。

まず、交渉の対象となる財 (論点) の数は、図 4 中 A 下部に直接入力するか、スピナーで調整することで、設定ができる。論点の数は少なくとも 1 以上である。論点の数が決まったら、図 4 中 A 右下の “Set” を押すことで論点の数を決定する。ここでは、後述のとおり、手動での入力のほか、論点数に応じて自動的に各論点に対応した効用空間を生成可能としているが、必要に応じてあらかじめ定義した問題空間 (ドメイン) を読み込むことも可能である。

次に、交渉を行うエージェントのプログラムを追加する。エージェントは図 4 中下部の “Add Agent” を押すことで、本フレームワークに読み込み可能なエージェントプログラムの一覧から必要なものを追加できる。“Remove Agent” を押すことで、すでに追加したエージェントを交渉で使用しないように削除することができる。Misrepresentation agent のプログラムは、“Add MisrepresentationAgent” を押すことで追加、“Remove MisrepresentationAgent” を押すことで削除することができる。通常のエージェントも Misrepresentation agent も任意の数追加することができるが、Misrepresentation game としての交渉を行うためには、少なくとも通常のエージェント 1 体と、Misrepresentation agent1 体が必要である。

論点に対する重み付け (選好) を図 4 中 C 周辺の入力欄に直接入力する場合、数値またはスピナーで調整することで入力することができる。この値は論点に対する選好を示しており、この値が大きいほど、その論点を優先することを示す。Preference-elicitation に基づく公平な財の分配交渉においては、質問に対する答えとして、2 つの論点を比べ、ここで決定した選好の値が大きい方が好みである、と返答する。選好を決定する動作は Misrepresentation agent についても同様であるが、ここでの選好の値とは異なる返答をすることができる。

これらの一連の交渉条件を設定した後、図 4 中 D 下部の “Start Negotiation” を押すことで交渉が開始される。交渉そのものは自動的に行われ、右上の “Result” で交渉の結果を、右下の “Log” で交渉の様子を確認することができる。

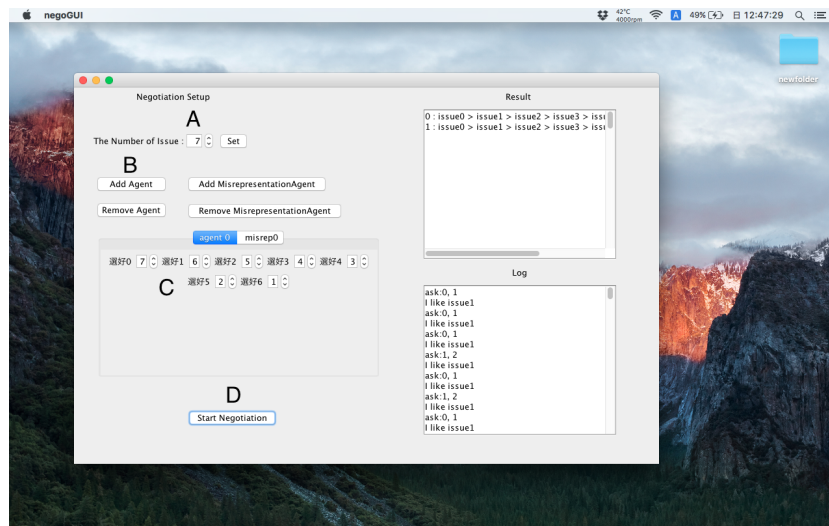


図 4 試作中のソフトウェアフレームワークの動作例

8. おわりに

Misrepresentation game 解析ソフトウェアフレームワークについては、以下の評価手法が考えられる。1つは、ANAC で使われているエージェント間交渉フレームワークである GENIUS と対比させ、Misrepresentation game の分析において、本研究との差異を定性的に比較評価することを試みることである。もう1つの評価方法の観点として、Misrepresentation game に関する研究である [1] では、複数間で行う Misrepresentation game についての検討結果は示されていない点に着目できる。複数間交渉における Misrepresentation game を行う場合はどのようなプロトコルを用いるのか、fixed-pie lie が使用可能なのか、使えないならば、Misrepresentation game が成立するにはどのような条件が必要なのかは、今後解析する必要がある。本フレームワークを用いて3者間交渉の条件下での解析を進めることで、Misrepresentation game となるような行動を取ることが可能かどうか、fixed-pie lie などの戦略をどういった場面に用いるのが効果的であるかを明らかにすることが、もう一つの本研究における評価方法の1つとして考えられる。Misrepresentation game が成立するような条件の生成が可能であれば、論点数やエージェント数の増加にもなってそれを可能とする条件がどのように変わるかも、検討の余地がある。これにより、エージェント数の増加に応じて、正直に返答した時よりも多く得られる効用はどの程度になるのかも検証可能となると考えられる。特に、Misrepresentation game を行える場合に、得られる効用がマイナスになる場合があるかどうか、平均してどの程度の効用を得られるのかといったことについても検証したいと考えている。

謝辞

本研究の一部は、科学研究費補助金基盤研究 (B)15H02972 からの支援による。

参考文献

- [1] Gratch, J., Nazari Z., Johnson, E.: The Misrepresentation Game: How to win at negotiation while seeming like a nice guy. In: Proceedings of the 15th International Conference on Autonomous Agents and Multiagent Systems (AAMAS2016), pp728-737 (2016)
- [2] Lin, R., Kraus, S., Baarslag, T., Tykhonov, D., Hindriks, K., Jonker, C.M.: Genius: an integrated environment for supporting the design of generic automated negotiators. *Comput. Intell.*30(1), pp.48-70,(2014).
- [3] Li, T., Boutilier, C.: Robust approximation and incremental elicitation in voting protocols. 22nd International Joint Conference on Artificial Intelligence, pages 287-293 (2011)
- [4] Rubinstein, A.: Perfect equilibrium in a bargaining model. *Economica* 50(1), 97-109 (1982)
- [5] Rubinstein, A.: A bargaining model with incomplete information about time preferences. *Economica* 53(5), 1151-1172 (1985)
- [6] Faratin, P., Sierra, C., Jennings, N.R.: Using similarity criteria to make issue trade-offs in automated negotiations. *Artif. Intell.* 142, 205-237 (2002)
- [7] Fatima, S.S., Wooldridge, M., Jennings, N.R.: Multi-issue negotiation under time constraints. In: Proceedings of the First International Conference on Autonomous Agents and Multi-agent Systems (AAMAS 2002), pp143-150. New York, NY, USA (2002)
- [8] Kraus, S.: Strategic Negotiation in Multiagent Environments. MIT Press (2001)
- [9] Osborne, M.J., Rubinstein, A.: Bargaining and Markets (Economic Theory, Econometrics, and Mathematical Economics). Academic Press (1990)
- [10] Baarslag, T.: What to Bid and When to Stop. SIKS Dissertation Series No.2014-26. (2014)
- [11] K, Fujita.: Automated Negotiating Agent with Strategy Adaption for Multi-times Negotiations. *Recent Advances in Agent-based Complex Automated Negotiation, Studies in Computational Intelligence* 638. (2016)