1B2-3

開環境での協力ゲームにおける解の簡略記述法の検討
A Compact Representation Scheme for Coalitional Games in Open Anonymous Environment

大田　直樹[*1] Naoki Ohta　Vincent Conitzer[*2]　Tuomas Sandholm[*2]　丸小野　公己[*1] koki maruono　岩崎　敦[*1] Atsushi Iwasaki
横尾　真[*1] Makoto Yokoo

[*1]九州大学 Kyushu University　[*2]Carnegie-Mellon University

Coalition formation is a key aspect of automated negotiation among self-interested agents. In order for coalitions to be stable, a key question that must be answered is how the gains from cooperation are to be distributed. Previous research has revealed that traditional solution concepts are vulnerable to various manipulations in open anonymous environments such as the Internet. To address this, a solution concept called the anonymity-proof core, which is robust to such manipulations, was developed. However, representing solutions in the anonymity-proof core (or similar concepts) requires space exponential in the number of agents. This paper proposes a compact representation of the outcome function. This compact representation scheme can express the outcome functions in the anonymity-proof core. Furthermore, this paper develops another new solution concept called the anonymity-proof nucleolus that can be expressed in this compact representation. The anonymity-proof nucleolus is uniquely determined.

1.　はじめに

提携を組むということは，自動化された利己的な主体（エージェント）の持つ重要な性質である．エージェント間の提携が成立した場合，我々は提携を結んだエージェントの集合が得た利得をどのように分配するかを考える必要がある．

協力ゲーム理論は，複数のエージェントが協力する際の，利得の分配方法に関する研究分野である．この研究分野においては，シャープレイ値，コア，最小コアそして仁といった解概念と呼ばれる利得の分配方法が提案されている [鈴木 81, 岡田 96, Schmeidler 69, Neumann 47]．さらに，これらの解概念は，マルチエージェントシステムにおいて，実際に利用されている [Zlotkin 94, Shehory 98, Conitzer 03, Conitzer 04].

また近年のインターネットの普及により，複数の企業，組織が動的，迅速に提携を構成することが可能／必要となったことから，協力ゲーム理論の適用分野は今後さらに拡大していくことが予想される．しかし最近の研究により，シャープレイ値やコアといった代表的な解概念を用いた場合，エージェントはインターネットの様な開環境の下で実行できる不正な操作を行うことで，利得を増加させることが判明した．そのためこれらの操作に耐性を持つ，匿名の開環境で適用可能な解概念，匿名操作不可能コアが提案された [Yokoo 05].

しかし匿名操作不可能コアのような，匿名の開環境の下で適用可能な解概念を用いた場合，解となる配分の記述量がエージェントの数に対して指数的に増加するという問題が生じる．そのため本論文ではシナジー提携集合（SCG）[Conitzer 03] を用いて，解となる配分の記述量を減らす方法を提案する．この方法を用いることにより，シナジー提携集合の要素数が少ない場合，解となる配分を簡略に表記できる．加えて匿名操作不可能コアの解となる配分が，簡略表記可能となることを示す．

さらに匿名操作不可能コアは，その解が一意に求まらないことや，存在しない場合がある．こうした問題に対して，本論文では新しい解概念である匿名操作不可能仁を提案する．この解概念は匿名操作不可能コアと同じく，匿名の開環境の下，エージェントが行う操作に耐性を持ち，かつ解の簡略表記が可能という特徴を持つ．さらに匿名操作不可能仁は，解が一意に求まる事など，いくつかの特徴を持つことを示す．

連絡先: 大田　直樹，九州大学，812-8581 福岡県福岡市東区箱崎 6-10-1，(092)642-3882，oota@lang.is.kyushu-u.ac.jp

2.　前提条件の定義

近年，協力ゲームの解概念は，特性関数を用いて利得の配分を決定している．特性関数 w とは，任意の提携（ゲームに参加するエージェントの集合）X を引数とする関数であり，w(X) は X が協力し合うことにより，得る利得の量を示す.

しかし匿名の開環境でエージェントが実行可能な操作を定式化するためには，通常のエージェントの集合に関して定義された特性関数で与えられる情報のみでは不十分であることが示されている．このため，我々はエージェントの持つスキルという概念を導入する．スキルとはエージェントの持つ能力を限界まで細分化したものである．特性関数をエージェントの集合に対してではなく，スキルの集合によって定義することにより，匿名の開環境でエージェントが実行可能な操作を明確に定義することが可能となる.

定義 1 (スキルとエージェント)**.** それぞれのエージェントが持つ，個々の分割不可能な技能をスキルと呼ぶ．各エージェントはそれぞれ 1 つ，または複数のスキルを持つ．また各スキルはそれぞれ固有の名称を持ち，同じ機能を持つスキルもそれぞれ区別することが可能であるとする.

定義 2 (スキルの特性関数)**.** スキルの特性関数 $v : 2^T \to \Re$（Tはスキルの全体集合）は，任意のスキルの集合 S に対して，S を持つエージェントが協力した場合に得られる利得 $v(S)$ を与える関数である.

本論文ではエージェントの特性関数の代わりに，スキルの特性関数を用いるものとする.

匿名の開環境下において，エージェントは架空名義の利用，共謀といった不正行為を行うことで，利得を増加させることが

出来ることが示されている．これらの操作を防止する方法として，エージェントに直接配分を与えるのではなく，スキルに与えられる配分をもとに（既存の解概念をスキルに対して用いることは可能），エージェントに与える配分を決定する方法が提案されている．この時エージェントに与える利得は，そのエージェントが持つスキルに与えられる利得の総和である．しかしこの方法を用いた場合，エージェントは自分が持つスキルを隠蔽することで，不正に利得を増加させることが出来る．

3. 匿名操作不可能コア

本章では，最近提案された新しい解概念である匿名操作不可能コアについて説明する [Yokoo 05]．この解概念は，エージェントが行う操作に対し耐性を持つ解概念である．

ここからはゲームに参加しうるスキルの集合を T と定義する．匿名の開環境で適用可能な解概念は，エージェントに対する配分ではなく，利得関数を解とする．利得関数 π とは，任意のスキルの集合 $S \subseteq T$ 及び，$s \in S$ を満たす任意のスキル s を引数とし，$\pi(s, S)$ と表現される．これはゲームに参加したスキルの集合が S であった場合，s が受け取る利得を示す．

利得関数 π から，エージェントに与える配分を決定することにより，架空名義の利用及び，共謀を防止出来る．そのため匿名の解環境で適用可能な利得関数（以下，匿名操作不可能な利得関数）は，次のように定義できる．

定義 3 (匿名操作不可能な利得関数)**.** 匿名操作不可能な利得関数 π は，次の条件を満たす利得関数である．

$$\forall S, S \subseteq T, \sum_{s \in S} \pi(s, S) = v(S). \tag{1}$$

$\forall S, \forall S', \forall S'', S'' \subseteq S' \subset S \subseteq T,$

$$\sum_{s \in S''} \pi(s, S') \leq \sum_{s \in S'' \cup (S \setminus S')} \pi(s, S). \tag{2}$$

これらの条件式の内，式（1）は，ゲームの結果生じた利得が，もれなく各参加スキルに分配されることを示し，式（2）は，スキルを隠蔽する誘因が生じないことを示す．

匿名操作不可能な利得関数のうち，任意のスキルの集合が独自に提携をくみ，自分たちだけでゲームをする誘因を生じさせないような利得関数の集合を，匿名操作不可能コアという．

定義 4 (匿名操作不可能コア)**.** 匿名操作不可能コアとは，次の条件を満たす匿名操作不可能な利得関数 π の集合である．

- $\forall S \forall S' \subseteq S$, $\sum_{s \in S'} \pi(s, S) \geq v(S')$

例 1. 参加しうるスキルの集合 $T = \{a, b, c, d, e\}$，下のようなスキルの特性関数 v をもつゲームを考える．

- $v(\{a, b, c, d, e\}) = 2$
- $v(\{a, b, d, e\}) = v(\{b, c, d, e\}) = 2,$
- $v(\{a, b, c, d\}) = v(\{a, b, c, e\}) = v(\{a, c, d, e\}) = 1,$
- $v(\{a, b, c\}) = v(\{a, b, d\}) = v(\{a, b, e\}) = v(\{a, d, e\}) = v(\{b, c, d\}) = v(\{b, c, e\}) = v(\{b, d, e\}) = v(\{c, d, e\}) = 1,$
- $v(\{a, b\}) = v(\{b, c\}) = v(\{d, e\}) = 1,$
- *For any subset* $S \subset \{a, b, c, d, e\}$, $v(S) = 0.$

匿名操作不可能コアは利得の配分を利得関数の形で定義するので，このゲームの匿名操作不可能コア π は次のような利得関数の集合となる．

- $\pi(b, \{a, b, \ldots\}) = \pi(b, \{b, c, \ldots\}) = 1,$
- $\pi(d, \{d, e, \ldots\}) = p$, $\pi(e, \{d, e, \ldots\}) = q$, 但し $p \geq 0, q \geq 0, p + q = 1,$
- 上に含まれないスキル s とスキルの集合 $S \subseteq \{a, b, c, d, e\}$ の組 (s, S) について，$\pi(s, S) = 0.$

例 1 で示している通り，匿名操作不可能コアとなる利得関数を定義する場合，定義される利得関数の引数の組の数は 90 となる．このように匿名操作不可能コアとなる利得関数を定義するには，ゲームに参加しうるスキルの総数 n に対し，$n(2^{n-1})$ と指数的な表記量が必要になるという問題がある．

4. 利得関数の簡略表記

本章では，前章でしめした利得関数の表記量の問題を解決するために，利得関数の表記量を減らす方法を提案する．本章では先に提案されたシナジー提携集合（synergy coalition group 以下 SCG）[Conitzer 03] を用いて，簡略表記を行う．シナジー提携集合とは，その提携が組むことで新たな価値が生み出されるような提携の集合である．

定義 5 (シナジー提携集合)**.** シナジー提携集合とは，提携が組まれることで価値が増加するスキルの提携の集合であり，下の式を満たすスキルの集合 S の集合である．

- $\bigcup_j S_j = S$ 及び全ての S_j は共通部分を持たないという条件を満たす，任意のスキルの集合の集合 $\forall \{S_1, \ldots, S_j, \ldots\}$，について $v(S) > \sum_{S_j} v(S_j)$ が成立．

例えば例 1 の協力ゲームにおける SCG は $[\{a, b\}, \{b, c\}, \{d, e\}]$ となる．

あるゲームの特性関数の全ては，そのゲームの SCG の各要素を引数とする特性関数の値から求めることが出来る．SCG に含まれないスキルの集合 S について，$v(S)$ は次のように求めることが出来る．

- $v(S) = \max\{\sum_{1 \leq j \leq r} v(S_j)$, 但し $\bigcup_{1 \leq j \leq r} S_j = S$ 及び全ての S_j は共通部分を持たない．$\}$

本論文では SCG を用いて汎シナジー提携集合（generalized synergy coalition group 以下 GSCG ）を提案する．

定義 6 (汎シナジー提携集合)**.** 汎シナジー提携集合 $GSCG$ とは以次の条件を満たすスキルの集合の集合である．

- $\forall S$, *if* $S \in SCG$, *then* $S \in GSCG$,
- $\forall S_1 \in GSCG, \forall S_2 \in GSCG$, *if* $S_1 \cap S_2 \neq \emptyset$, *then* $S_1 \cup S_2 \in GSCG$.

例えば例 1 の協力ゲームにおける $GSCG$ は SCG に $\{a, b, c\}$ を加えたものである．

ここからは，利得関数の簡略表記法について述べる．まず最初に簡略表記法に用いる概念であるスキルの集合の $GSCG$ への写像について定義する．次にスキルの集合の $GSCG$ への写像を用いて，利得関数を簡略に表記する方法を説明する．

定義 7 (スキルの集合の GSCG への写像). あるスキルの集合 S の $GSCG$ への写像 P_S は次のように定義される:

$$\begin{aligned} P_S = \ & \{C \mid C \in GSCG, C \subseteq S, \\ & \text{and } \forall C', \text{where } C \subset C' \subseteq S, C' \notin GSCG\} \end{aligned}$$

例えば例 1 のゲームにおいて，スキルの集合 $\{a,b,c,d,e\}$ の $GSCG$ への写像は，$[\{a,b,c\},\{d,e\}]$ となる．

次に GSCG への写像を用いて，利得関数を簡略に表記する方法を説明する．まずは簡略表記利得関数 π_c を定義する．

定義 8 (簡略表記利得関数). 簡略表記利得関数 π_c は，第 2 引数に $S \in GSCG$ を満たす任意のスキルの集合 S，第 1 引数に $s \in S$ を満たす任意のスキル s をとる関数である．$\pi_c(s,S)$ は実際にゲームに参加したスキルの集合が S である時，スキル s が受け取る利得である．

簡略表記利得関数は，利得関数に比べ，少ない表記量で表現できる．例えば例 1 について，簡略表記利得関数 $\pi(s,S)$ の引数は，第 2 引数 S の取りうるスキルの集合は，$\{a,b\}$, $\{a,c\}$, $\{d,e\}$, $\{a,b,c\}$ 第 1 引数 s の取りうるスキルは $s \in S$ を満たすスキルである．このように簡略表記利得関数の引数の組の数は 9 であり，利得関数の引数の組の数 90 に比べ，小さくなっていることが分かる．

このように簡略表記利得関数は，利得関数より少ない表記量で表現することが出来る．しかし全ての利得関数が，簡略表記利得関数で表現できるとは限らない．そのためここからは，簡略表記関数で表現できる利得関数を，簡略表記可能な利得関数として定義する．

定義 9 (簡略表記可能な利得関数). 任意のスキルの集合 S 及び，$s \in S$ について次の条件を満たす特性関数 π は，簡略表記利得関数 π_c によって簡略表記可能となる．

$$\pi(s,S) = \begin{cases} \pi_c(s,P) & \text{where } s \in P \text{ and } P \in P_S \\ & \text{if such } P \text{ exists,} \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

上の定義より分かるように，任意の簡略表記利得関数は，ある利得関数を簡略表記したものである．

利得関数の中に，匿名操作不可能な利得関数が存在するように，簡略表記利得関数の中にも，匿名操作不可能な簡略表記利得関数が存在する．匿名操作不可能な簡略表記利得関数は，匿名操作不可能な利得関数を簡略表記したものである．つまり匿名操作不可能な簡略表記利得関数は，匿名の開環境で適用可能な簡略表記関数といえる．

定義 10 (匿名操作不可能な簡略表記利得関数). 匿名操作不可能な簡略表記利得関数 π_c とは次の条件を満たす簡略表記利得関数である：

- $B \in GSCG$ を満たす任意の B, $S' \subset S \subseteq B$ を満たす任意の S, S' 及び，$B \setminus (S \setminus S')$ の $GSCG$ への写像，$P_{B \setminus (S \setminus S')}$ について

$$\sum_{P \in P_{B \setminus (S \setminus S')}} \sum_{s \in (S' \cap P)} \pi_c(s,P) \le \sum_{s \in (B \cap S)} \pi_c(s,B).$$

が成立する．

つまり匿名操作不可能な簡略表記利得関数とは，スキルを隠蔽する誘因を持たない簡略表記関数のことである．

定理 1. 匿名操作不可能な簡略表記利得関数 π_c によって，簡略表記される利得関数 π は匿名操作不可能な利得関数である．

証明. ある匿名操作不可能な簡略表記利得関数 π_c によって表現される利得関数 π がスキルを隠蔽する誘因を持つと仮定し，この仮定の下では矛盾が生じることを示し，そのため，匿名操作不可能な簡略表記利得関数 π_c によって表現される利得関数 π はスキルを隠蔽する誘因がない，すなわち π は匿名操作不可能な利得関数であるということを示す．以下の条件を満たす，互いに共通部分を持たないスキルの集合 S_1, S_2, S_3 が存在すると仮定する：

$$\sum_{s \in S_1} \pi(s, S_1 \cup S_3) > \sum_{s \in (S_1 \cup S_2)} \pi(s, S_1 \cup S_2 \cup S_3).$$

次に証明を簡単にするために，$S_1 \cup S_2 \cup S_3$ の GSCG への写像は唯一つのスキルの集合 G によってのみ構成されると仮定する．$S_1 \cup S_2 \cup S_3$ の GSCG への写像が複数のスキルの集合によって構成される場合，それぞれのスキルの集合について，以下の議論を適用することで，証明出来る．

$s \in (S_1 \cap G)$ を満たす任意のスキル s について，$G \setminus (G \cap S_2)$ の GSCG への写像に s を含むスキルの集合が一つ存在する．そのスキルの集合を P と定義する．ただしそのようなスキルの集合が存在しない時，P は空集合とする．次に $S_1 \cup S_3$ の GSCG への写像について考える．この時 s を含むスキルの集合が，$S_1 \cup S_3$ の GSCG への写像について最大で一つ存在する．そのスキルの集合を P' と定義する．ただしそのようなスキルの集合が存在しない時，P' は空集合とする．

以下 $P = P'$ が成立することを示す．もし P' が P に含まれないスキルを所持している場合, $G \cup P'$ は GSCG の要素であり $S_1 \cup S_2 \cup S_3$ の部分集合でもあるため，G が $S_1 \cup S_2 \cup S_3$ の GSCG への写像という仮定に反する．一方もし P が P' に含まれないスキルを所持している場合, $P \cup P'$ は GSCG の要素であり，かつ $S_1 \cup S_3$ の部分集合でもあるため，P' が $S_1 \cup S_3$ の GSCG への写像という仮定に反する．

従って定義 8 より以下の式が成立する．:

$$\begin{aligned} \textstyle\sum_{s \in (S_1 \cup S_2)} \pi(s, S_1 \cup S_2 \cup S_3) = & \textstyle\sum_{s \in ((S_1 \cup S_2) \cap G)} \pi_c(s,G) \\ \ge & \textstyle\sum_{s \in (S_1 \cap G)} \pi_c(s,P), \end{aligned}$$

$$\begin{aligned} \textstyle\sum_{s \in S_1} \pi(s, S_1 \cup S_3) = & \textstyle\sum_{s \in (S_1 \cap G)} \pi_c(s,P') \\ = & \textstyle\sum_{s \in (S_1 \cap G)} \pi_c(s,P). \end{aligned}$$

これらの式より，$\sum_{s \in S_1} \pi(s, S_1 \cup S_3) \le \sum_{s \in (S_1 \cup S_2)} \pi(s, S_1 \cup S_2 \cup S_3)$，となるがこれらは仮定に矛盾するため，$\sum_{s \in S_1} \pi(s, S_1 \cup S_3) > \sum_{s \in (S_1 \cup S_2)} \pi(s, S_1 \cup S_2 \cup S_3)$ が成立する． □

次に匿名操作不可能コアに含まれる利得関数 π を表現する，簡略表記利得関数 π_c が満たすべき条件について述べる．

定義 11 (SCG に対するノンブロッキング条件). SCG に対するノンブロッキング条件を満たす簡略表記利得関数 π_c とは $B \in GSCG$ を満たす任意のスキルの集合 B および $S \in SCG, S \subseteq B$ を満たす任意のスキルの集合 S について $\sum_{s \in S} \pi_c(s,B) \ge v(S)$ を満たす簡略表記利得関数とする．

定理 2. SCG に対するノンブロッキング条件を満たす，匿名操作不可能な簡略表記利得関数を π_c とする．この π_c によって表現される利得関数 π は，匿名操作不可能コアに含まれる．

紙幅の都合上，証明は省略するが，利得関数を簡略表記することにより，表記量をかなり減らすことが出来る．例えば例 1 において，ノンブロッキング条件を満たす簡略表記利得関数の引数の組は 9 であるが，もし簡略表記をしない場合，利得関数の引数の組は 80 になる．

ここからはある協力ゲームの匿名操作不可能コアが非空である場合，必ず匿名操作不可能かつ SCG に対するノンブロッキング条件を満たす，簡略表記利得関数が存在することを示す．

定理 3. ある協力ゲームについて，匿名操作不可能コアが非空である場合，匿名操作不可能コアに含まれ，かつ簡略表記可能な利得関数 π が存在する．

証明. $\forall C \in GSCG, \forall s \in C, \pi_c(s, C) = \pi(s, C)$ を満たす簡略表記利得関数 π_c について考える．この時 π が匿名操作不可能な利得関数であるため，π_c は匿名操作不可能な簡略表記利得関数となる．従って，定理 2 より，π_c によって表現される利得関数は匿名操作不可能コアに含まれる． □

5. 匿名操作不可能仁

本章では本論文で新しく提案する解概念，匿名操作不可能仁について説明を行う．匿名操作不可能仁は，従来の仁の定義をもとにした，簡略表記利得関数に対する解概念である．この解概念は匿名の開環境に適用可能かつ簡略表記可能であるといった性質を持つ．さらにこの解概念は仁の性質を受け継いだ解概念である．具体的な性質については後に定理として示す．但し紙幅の都合上，証明は全て省略する．

ここからは匿名操作不可能仁の定義を行う．まず匿名操作不可能仁の定義を行うために必要な，不満と簡略不満ベクトルという概念を定義する．

定義 12 (不満)**.** $B \in GSCG$ を満たす任意のスキルの集合 B 及び，$S \subseteq B$ を満たしかつ，SCG の要素である，または，要素数が *1* であるスキルの集合 S について考える．参加スキルが B の時，S が任意の簡略表記利得関数 π_c にもつ不満 $f(\pi_c, S, B)$ とは，次の式により定義される：
$f(\pi_c, S, B) = v(S) - \sum_{s \in S} \pi_c(s, B)$

定義 13 (簡略不満ベクトル)**.** 簡略不満ベクトルとは，任意の簡略表記利得関数 π_c から求めることの出来る全ての不満を，降順に並べたものである．

ここからは匿名操作不可能仁の定義とその性質を示す．

定義 14 (匿名操作不可能仁)**.** 匿名操作不可能仁とは，辞書式順序で最も小さい簡略不満ベクトルを持つ匿名操作不可能な簡略表記利得関数である．

定理 4. 匿名操作不可能仁となる利得関数は，一つのゲームに対して複数存在しない．

スペースの都合上，証明は省略する．

定理 5. 匿名操作不可能仁は常に存在する．

スペースの都合上，証明は省略する．但しここの証明は文献 [Yokoo 05] の，Theorem 7 の証明と同じものである．

これらの性質に加え，匿名操作不可能仁となる利得関数は，匿名操作不可能コアに含まれる（但し，匿名操作不可能コアが存在する時のみ），同じ性能を持つスキルの組には，それぞれ同じ利得を与えるといった特徴を持つ．

6. おわりに

匿名操作不可能な解概念は匿名の開環境において適用可能な解概念である．しかし匿名操作不可能な解概念の解は利得関数となり，その表記量が参加しうるスキルの数に対して，指数的に増加するという問題がある．

本論文では，匿名操作不可能コア等の匿名操作不可能な解概念の解を，簡略に表記する方法を提案した．さらに，匿名操作不可能コアの解が常に一意に求まるとは限らないという問題を解決するために簡略表記可能である匿名操作不可能な解概念，匿名操作不可能仁を提案した．また匿名操作不可能仁は，どのような協力ゲームにおいても，解が唯一つ求まることや，同じ能力を持つスキルに対し，同じ利得を与えること等，仁と似たような性質を持つことを示した．

参考文献

[Conitzer 03] Conitzer, V. and Sandholm, T.: Complexity of determining nonemptiness of the core., pp. 613–618 (2003)

[Conitzer 04] Conitzer, V. and Sandholm, T.: Computing Shapley values, manipulating value division schemes, and checking core membership in multi-issue domains., pp. 219–225 (2004)

[Neumann 47] Neumann, von J. and Morgenstein, O.: *Theory of games and economic behavior* (1947)

[Schmeidler 69] Schmeidler, D.: The nucleolus of a characteristic function game., *Society for Industrial and Applied Mathematics Journal of Applied Mathematics*, Vol. 17, pp. 1163–1170 (1969)

[Shehory 98] Shehory, O. and Kraus, S.: Methods for task allocation via agent coalition formation., *Artificial Intelligence*, Vol. 101, No. 1–2, pp. 165–200 (1998)

[Yokoo 05] Yokoo, M., Conitzer, V., Sandholm, T., Ohta, N., and Iwasaki, A.: Coalitional games in open anonymous environments., pp. 509–515 (2005),（「匿名の開環境下における協力ゲームについて」，情報処理学会論文誌，掲載予定, (2006)）

[Zlotkin 94] Zlotkin, G. and Rosenschein, J. S.: Coalition, cryptography and stability: Mechanisms for coalition formation in task oriented domains., pp. 432–437 (1994)

[岡田 96] 岡田 章：ゲーム理論, 有斐閣 (1996)

[鈴木 81] 鈴木 光男：ゲーム理論入門, 共立全書 (1981)