Vol.2020-ICS-200 No.4
2020/9/14

エージェントのタイプと参加確率を考慮した提携構造形成問題

島崎 啓介[1,a)] 上田 俊[1,b)]

概要：提携構造形成 (Coalition Structure Generation, CSG) 問題とは，協力ゲーム理論における基本的な問題のひとつであり，あるエージェントの集合を社会的余剰が最大化されるように，いくつかの提携に分割する問題である．従来の CSG 問題の表記量は，エージェント数に対して指数関数的に増加する．この問題を解決するアプローチのひとつとして，類似した能力をもつエージェントが複数存在するような状況を考え，エージェントのタイプを用いた特性関数の簡略表記法が提案されている．一方，確率提携構造形成問題 (Probabilistic Coalition Structure Generation, PCSG) 問題とは，各エージェントの提携への参加の有無が確率により与えられている CSG 問題である．本論文では，前述のタイプを用いた特性関数の簡略表記法と確率提携構造形成問題を組み合わせた問題に対するアルゴリズムを提案する．

1. はじめに

提携構造形成 (Coalition Structure Generation, CSG) 問題とは，協力ゲーム理論における基本的な枠組みの一つであり，あるエージェントの集合に社会的余剰，すなわち，すべての提携（グループ）における利得の総和が最大化されるように，いくつかの提携に分割する問題である．この問題は NP-完全な問題として広く知られている完全集合分割問題と等価な問題であり，代表的な応用例として，分散経路決定問題，マルチセンサーネットワーク，排水処理システム等が挙げられる．CSG 問題は，エージェントの集合及び，特性関数と呼ばれるブラックボックス関数により定義される．エージェントの部分集合は提携，その分割は提携構造と呼ばれる．また，各提携の利得は特性関数により与えられ，提携構造の利得は，すべての提携における利得の総和により求められる．従来の CSG 問題では，各提携の利得は特性関数により与えられていることを仮定しているため，その表記量は，エージェント数に対して指数関数的に増加する．この問題を解決するアプローチの一つとして特性関数の簡略表記法に関する研究 [2] がある．多数のエージェントの中には類似した能力（タイプ）をもつものが複数存在し，その数はそれほど多くない，すなわち，類似した能力をもつエージェントが複数存在するような状況を考え，エージェントのタイプを用いた特性関数の簡略表記法を提案している．

確率提携構造形成問題 (Probabilistic Coalition Structure Generation, **PCSG**) とは，各エージェントの提携への参加の有無が確率により与えられている CSG 問題である．実世界におけるエージェントの振る舞いは不確実な要素を多く含んでいる．例えば，スケジュールの都合上，ある提携への参加の有無が曖昧なエージェント等，実問題を考えた場合，提携成立の能否を考えることは重要である．PCSG 問題では，各提携で得られる利得の期待値を計算し，すべての提携における利得の期待値の総和が最大化されるような提携構造を求めることを目的としている．この問題では，各提携において得られる利得の期待値の計算法が重要となる．既存研究では，ある提携における利得の期待値は，すべての構成員が参加したときのみに与えられるとする計算法や，全てのシナリオ（欠席者の全組合わせ）を想定した計算法が提案されている．

本論文では，前述のタイプを用いた特性関数の簡略表記法と確率提携構造形成問題を組み合わせた問題に対するアルゴリズムを提案する．提案アルゴリズムは，動的計画法を用いて最適な提携構造を探索する．さらに，シミュレーション実験を行い，提案アルゴリズムの性能を分析した．

次章では，$\mathbf{PCSG}_t$ 問題について概説する．3 章では，提案アルゴリズムを紹介し，4 章では，実験結果について述べる．5 章では，結論と今後の課題について述べる．

1 佐賀大学, Saga University, Saga, 840–8502, Japan
a) 19704008@edu.cc.saga-u.ac.jp
b) sgrueda@cc.saga-u.ac.jp

2. タイプ付き確率的提携構造形成問題

2.1 提携構造形成問題

本章では，提携構造形成（Coalition Structure Generation）問題について概説する．CSG 問題とは，あるエージェントの集合を社会的余剰が最大化されるように，いくつかの提携 (グループ) に分割する問題である．はじめに，CSG の定義を与える．

- $A = \{a_1, a_2, \ldots, a_n\}$: エージェントの集合
- $C \subseteq A$: 提携
- $v(C)$: 提携 C の利得
- $CS = \{C_1, C_2, \ldots, C_{|CS|}\}$: 提携構造
- $V(CS) = \sum_{C_i \in CS} v(C_i)$: 提携構造 CS の利得

ある提携構造 CS が以下の条件を満たしているとき, CS は最適であるといい, CS^* と記述する．

$$\forall CS, V(CS) \leq V(CS^*).$$

例 1 3 人のエージェント $A = \{a_1, a_2, a_3\}$ が存在し，以下の特性関数が与えられる:

$$\begin{aligned} v(\{a_1\}) &= 200, & v(\{a_2\}) &= 200, \\ v(\{a_3\}) &= 300, & v(\{a_1, a_2\}) &= 500, \\ v(\{a_1, a_3\}) &= 500, & v(\{a_2, a_3\}) &= 500, \\ v(\{a_1, a_2, a_3\}) &= 700. \end{aligned}$$

このとき, 最適な提携構造 CS^* は, $CS^* = \{\{a_1, a_2\}, \{a_3\}\}$ であり，$V(CS^*) = 800$ である．

以下，**CSG** 問題における代表的な性質を示す．

性質 1 **CSG** 問題において，特性関数が優加法性を満たすとき，すべてのエージェントからなる提携構造が最適となる．また，特性関数が劣加法性を満たすとき，単独のエージェントからなる提携構造が最適となる．

CSG 問題は完全集合分割問題 [3] と等価な問題であり，一般に， NP-完全な問題として広く知られている [1]．但し，性質 1 より，特性関数が優加法性，もしくは，劣加法性を満たすとき，元の問題を解くことなしに，前者に関しては全体提携，後者では単独提携が最適な提携構造となり，多項式時間内で求解可能となる．

2.2 タイプ付き提携構造形成問題

本章では，上田らによって提案されたエージェントの能力に関するタイプに基づく特性関数の簡略表記法 [2] を紹介する．また簡略化されたエージェントのタイプ付き特性関数を用いた CSG 問題の定義を与える．文献 [2] では, 類似した能力をもつエージェントが複数存在するような状況を考え，エージェントのタイプを用いた特性関数の簡略表記法が提案されている．

- $T = \{t^1, \ldots, t^{|T|}\}$: タイプの集合
- $A^1, \ldots, A^{|T|}$: 各タイプのエージェントの集合
- $\psi = \langle n^1, \ldots, n^{|T|} \rangle$: 提携タイプ (coalition-type)
- $v^t(\psi)$: 提携タイプ ψ の利得
- $\lambda = \{\psi_1, \psi_2, \ldots, \psi_{|\lambda|}\}$: 提携構造タイプ
- $V^t(\lambda) = \sum_{\psi_i \in \lambda} v^t(\psi_i)$: 提携構造タイプ λ の利得

ある提携構造タイプ λ が以下の条件を満たしているとき，λ は最適であるといい，λ^* と記述する．

$$\forall \lambda, V^t(\lambda) \leq V^t(\lambda^*).$$

例 2 例 1 は $T = \{t^1, t^2\}, A^1 = \{a_1, a_2\}, A^2 = \{a_3\}$ とすることでタイプ付き提携構造形成問題として表せる:

$$\begin{aligned} &v^t(\langle 1, 0 \rangle) = 200, \quad v^t(\langle 0, 1 \rangle) = 300, \quad v^t(\langle 2, 0 \rangle) = 500, \\ &v^t(\langle 1, 1 \rangle) = 500, \quad v^t(\langle 2, 1 \rangle) = 700. \end{aligned}$$

このとき, 最適な提携構造タイプ λ^* は, $\lambda^* = \{\langle 2, 0 \rangle, \langle 0, 1 \rangle\}$ であり，$V^t(\lambda^*) = 800$ である．

2.3 確率的提携構造形成問題

本章では，確率的提携構造形成 (Probabilistic Coalition Structure Generation, **PCSG**) 問題について概説する．**PCSG** 問題とは，各エージェントの任意の提携への参加の有無が確率により与えられてる **CSG** 問題である．ここで，各エージェントの任意の提携への参加の有無は独立であるものとする．すなわち，あるエージェントの提携への参加の有無は，他のエージェントの参加の有無には依存しないものとする．はじめに，**PCSG** の定義を与える．

- $A = \{a_1, a_2, \ldots, a_n\}$: エージェントの集合
- $0 \leq p(a_i) \leq 1$: エージェント a_i の参加確率
- $C \subseteq A$: 提携
- $v_e(C) = v(C) \cdot \prod_{a_i \in C} p(a_i)$: 参加確率を考慮した提携 C の利得
- $CS = \{C_1, C_2, \ldots, C_{|CS|}\}$: 提携構造
- $V_e(CS) = \sum_{C_i \in CS} v_e(C_i)$: 参加確率を考慮した提携構造 CS の利得

ある提携構造 CS が以下の条件を満たしているとき，CS は最適であるといい，CS_e^* と記述する．

$$\forall CS, V_e(CS) \leq V_e(CS_e^*).$$

例 3 例 1 で，$p(a_1) = 0.5, p(a_2) = 0.5, p(a_3) = 0.9$ のとき，参加確率を考慮したそれぞれの提携の利得は:

$$\begin{aligned} v(\{a_1\}) &= 100, & v(\{a_2\}) &= 100, \\ v(\{a_3\}) &= 270, & v(\{a_1, a_2\}) &= 125, \\ v(\{a_1, a_3\}) &= 225, & v(\{a_2, a_3\}) &= 225, \\ v(\{a_1, a_2, a_3\}) &= 157.5. \end{aligned}$$

このとき，最適な提携構造 CS_e^* は，$CS_e^* = \{\{a_1\}, \{a_2\}, \{a_3\}\}$ であり，$V_e(CS_e^*) = 470$ である．

特性関数が優加法性または劣加法性を満たすような **PCSG** 問題において，以下の性質が成り立つ．

性質 2　PCSG 問題において，特性関数が優加法性を満たすとき，すべてのエージェントからなる提携構造が必ずしも最適になるとは限らない．しかし，特性関数が劣加法性を満たすときは，CSG のときと同様に，単独のエージェントからなる提携構造が最適となる．

2.4　タイプ付き確率的提携構造形成問題

本章では，エージェントのタイプを考慮した PCSG 問題を定義する．まず，タイプ付き提携構造形成問題 ($\mathsf{CSG_t}$) の定義を以下のように整理する．

- $A = \{a_1, a_2, \ldots, a_n\}$: エージェントの集合
- $T = \{t^1, \ldots, t^{|T|}\}$: タイプの集合
- $A^1, \ldots, A^{|T|}$: 各タイプのエージェントの集合
- $\psi = \langle n^1, \ldots, n^{|T|} \rangle$: 提携タイプ (coalition-type)
- $v^t(\psi)$: 提携タイプ ψ の利得
- $\lambda = \{\psi_1, \psi_2, \ldots, \psi_{|\lambda|}\}$: 提携構造タイプ
- $V^t(\lambda) = \sum_{\psi_i \in \lambda} v^t(\psi_i)$: 提携構造タイプ λ の利得
- $\mathsf{CSG_t} = \langle A, T, A^1, \ldots, A^{|T|}, v^t \rangle$

次に参加確率タイプを定義し，提携を行列 $\mathbb{C}$ として表現する定義を以下のように与える．

- $T_p = \{t_p^1, \ldots, t_p^{|T_p|}\}$: 参加確率タイプの集合
- $0 \leq p(t_p^i) \leq 1$: 参加確率タイプ t_p^i の参加確率
- $A_p^1, \ldots, A_p^{|T_p|}$: 各参加確率ののエージェントの集合
- $\mathbb{A} = \begin{bmatrix} A^{(1,1)} & \ldots & A^{(1,|T|)} \\ \ldots & \ldots & \ldots \\ A^{(|T_p|,1)} & \ldots & A^{(|T_p|,|T|)} \end{bmatrix}$: 全員が一つの提携に属する参加確率を考慮した提携タイプ
- $\mathbb{C} = \begin{bmatrix} n^{(1,1)} & \ldots & n^{(1,|T|)} \\ \ldots & \ldots & \ldots \\ n^{(|T_p|,1)} & \ldots & n^{(|T_p|,|T|)} \end{bmatrix}$: 参加確率を考慮した提携タイプ

ここで，$v_e^t(\mathbb{C})$ を参加確率を考慮した提携タイプ $\mathbb{C}$ の利得を与える特性関数とする．ψ を $\mathbb{C}$ に対応する提携タイプとすると，$v_e^t(\mathbb{C})$ は $v^t(\psi)$ から以下のように計算される:

$$v_e^t(\mathbb{C}) = v^t(\psi) \cdot \prod p(t_p^i)^{\sum n^{(t_p^i, t^j)}}$$

これを用いて，提携構造タイプとそれに対する利得を以下のように与える．

- $\mathbb{CS} = \{\mathbb{C}_1, \mathbb{C}_2, \ldots, \mathbb{C}_{|\mathbb{C}|}\}$: 参加確率を考慮した提携構造タイプ
- $V_e^t(\mathbb{CS}) = \sum_{\mathbb{C}_i \in \mathbb{CS}} v_e^t(\mathbb{C}_i)$: 参加確率を考慮した提携構造タイプ $\mathbb{CS}$ の利得

最後に，エージェントのタイプを考慮した PCSG 問題は以下で与えられる:

$$\mathsf{PCSG_t} = \langle A, T, T_p, p, A^{(1,1)}, \ldots, A^{(|T_p|,|T|)}, v^t \rangle$$

例 4　以下の表でタイプと参加確率タイプのそれぞれのエージェント数が与えられている:

	t^1	t^2
t_p^1	2	0
t_p^2	0	1

参加確率を考慮したそれぞれの提携の利得は以下のように与えられる:

$$v(\begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}) = 100, \qquad v(\begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}) = 270,$$

$$v(\begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix}) = 125, \qquad v(\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}) = 225,$$

$$v(\begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}) = 157.5.$$

このとき，最適な提携構造 $\mathbb{CS}^*$ は，
$\mathbb{CS}^* = \{\begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}\}$ であり，
$V_e^t(\mathbb{CS}^*) = 470$ である．

3.　提案アルゴリズム

以下が，本論文で提案する 2 つのアルゴリズムの基本的なアイデアである．

- 動的計画法アルゴリズムで求解する．
  - タイプ付き確率的提携構造形成問題をタイプ付き提携構造形成問題に変換する．
  - 動的計画法を用いたアルゴリズムで求解する．
- 整数計画問題として定式化し，求解する．

3.1　動的計画法アルゴリズム

$\mathsf{PCSG_t} = \langle A, T, T_p, p, A^{(1,1)}, \ldots, A^{(|T_p|,|T|)}, v^t \rangle$ を $\mathsf{CSG_t} = \langle \bar{A}, \bar{T}, \bar{A}^1, \ldots, \bar{A}^{|\bar{T}|}, \bar{v}^t \rangle$ に変換する．

- $\bar{A} = A$
- $\bar{T} = T_p \times T$
  - $\bar{T} = \{t^{(1,1)}, \ldots, t^{(|T_p|,|T|)}\}$
- $\bar{A}^1 = A^{(1,1)}, \ldots, \bar{A}^{|\bar{T}|} = A^{(|T_p|,|T|)}$
  (1) $|\bar{A}^{(i,j)}| = 0$ になるところは削除する．同時に，$t^{(i,j)}$ を $\bar{T}$ から削除する．
- $\bar{v}^t(\langle n^{(1,1)}, \ldots, n^{(|T_p|,|T|)} \rangle) = v^t(\langle \sum_{i=1}^{|T_p|} n^{(i,1)}, \ldots, \sum_{i=1}^{|T_p|} n^{(i,|T|)} \rangle) \cdot \prod_{i \in T_p} p(t_p^i)^{\sum_{j=1}^{|T|} n^{(i,j)}}$

変換後の $\mathsf{CSG_t} = \langle \bar{A}, \bar{T}, \bar{A}^1, \ldots, \bar{A}^{|\bar{T}|}, \bar{v}^t \rangle$ を、動的計画法アルゴリズムで解く．

例 5　変換の例を示す．

- $A = \{a_1, a_2, a_3\}$, $T = \{t^1, t^2\}$, $T_p = \{t_p^1, t_p^2\}$, $p(t_p^1) = 0.5$, $p(t_p^2) = 0.9$
- $\begin{pmatrix} |A^{(1,1)}| & |A^{(1,2)}| \\ |A^{(2,1)}| & |A^{(2,2)}| \end{pmatrix} = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix}$

- v^t は以下の通り:

$$v^t(\langle 1,0\rangle)=200,\quad v^t(\langle 0,1\rangle)=300,\quad v^t(\langle 2,0\rangle)=500,$$
$$v^t(\langle 1,1\rangle)=500,\quad v^t(\langle 2,1\rangle)=700.$$

この例を変換アルゴリズムに従って $\mathsf{CSG_t}$ に変換する．

(1) $\bar{A}=A=\{a_1,a_2,a_3\}$

(2) $\bar{T}=T_p\times T=\{\bar{t}^1,\bar{t}^2,\bar{t}^3,\bar{t}^4\}$
  - $\bar{t}^1=t_p^1\times t^1,\ \bar{t}^2=t_p^2\times t^1,\ \bar{t}^3=t_p^1\times t^2,\ \bar{t}^4=t_p^2\times t^2$

(3) 各タイプのエージェント数の計算
  (a) $|\bar{A}^1|=|A^{(1,1)}|=2$
  (b) $|\bar{A}^2|=|A^{(2,1)}|=0$
  (c) $|\bar{A}^3|=|A^{(1,2)}|=0$
  (d) $|\bar{A}^4|=|A^{(2,2)}|=1$

(4) $|\bar{A}^{(i,j)}|=0$ になるところを削除する．
  (a) $\bar{A}^2$ と $\bar{A}^3$ が削除される．
  (b) 同時に，$\bar{T}=\{\bar{t}^1,\bar{t}^4\}$ になる．

(5) 最後に提携タイプの利得を計算する．
  (a) $\bar{v}^t(\langle 1,0\rangle)=v^t(\langle 1,0\rangle)\cdot(0.5)^1\cdot(0.9)^0=200\cdot 0.5=100$
  (b) $\bar{v}^t(\langle 0,1\rangle)=v^t(\langle 0,1\rangle)\cdot(0.5)^0\cdot(0.9)^1=300\cdot 0.9=270$
  (c) $\bar{v}^t(\langle 2,0\rangle)=v^t(\langle 2,0\rangle)\cdot(0.5)^2\cdot(0.9)^0=500\cdot 0.25=125$
  (d) $\bar{v}^t(\langle 1,1\rangle)=v^t(\langle 1,0\rangle)\cdot(0.5)^1\cdot(0.9)^1=500\cdot 0.45=225$
  (e) $\bar{v}^t(\langle 2,0\rangle)=v^t(\langle 1,0\rangle)\cdot(0.5)^2\cdot(0.9)^1=700\cdot 0.225=157.5$

したがって，変換後は以下のようになる:

- $\bar{A}=\{a_1,a_2,a_3\},\ \bar{T}=\{\bar{t}^1,\bar{t}^4\},\ |\bar{A}^1|=2,\ |\bar{A}^4|=1$
- $\bar{v}^t$ は以下の通り:

$$\bar{v}^t(\langle 1,0\rangle)=100,\quad \bar{v}^t(\langle 0,1\rangle)=270,\quad \bar{v}^t(\langle 2,0\rangle)=125,$$
$$\bar{v}^t(\langle 1,1\rangle)=225,\quad \bar{v}^t(\langle 2,1\rangle)=157.5.$$

3.2 整数計画法アルゴリズム

$\mathsf{PCSG_t}=\langle A,T,T_p,p,A^{(1,1)},\dots,A^{(|T_p|,|T|)},v^t\rangle$ を整数計画問題として定式化する．

- 可能な $\mathbb{C}$ の集合を $\prod$ とし，任意の $\mathbb{C}\in\prod$ に対して $x\in[0,min\lfloor\frac{A^{(i,j)}}{n^{(i,j)}}\rfloor]$ を $\mathbb{C}$ をいくつ使うかを表す変数とする．
- 目的関数を設定する．
  - $maximize \sum_{\mathbb{C}\in\prod} v_e^t(\mathbb{C})\cdot x_{\mathbb{C}}$
- 制約を設定する．
  - $\sum_{\mathbb{C}\in\prod}\{(\sum_{k=1}^{|T_p|} n^{(k,i)})\cdot x_{\mathbb{C}}\}=|A^i|$
  - $\sum_{\mathbb{C}\in\prod}\{(\sum_{k=1}^{|T|} n^{(i,k)})\cdot x_{\mathbb{C}}\}=|A_p^i|$

4. 実験結果

本章では，提案アルゴリズムの性能を実験的に評価する．用いた計算機は，Intel(R) Core(TM)i5 − 8500U CPU 3.00GHz プロセッサーと 16.00GB のメモリを搭載した Windows 10 pro マシンである．

本実験では，以下のように定義した問題を用いる．

- エージェント数を 20, タイプ数を 2, 各タイプの参加確率を 0.9 と 0.5 と 0.1 と設定する．
- 以下のように 5 つのパターンを左から No.1〜No.5 と定義する．

$$\begin{pmatrix}8&8\\1&1\\1&1\end{pmatrix}\ \begin{pmatrix}6&6\\2&2\\2&2\end{pmatrix}\ \begin{pmatrix}3&3\\4&4\\3&3\end{pmatrix}\ \begin{pmatrix}2&2\\2&2\\6&6\end{pmatrix}\ \begin{pmatrix}1&1\\1&1\\8&8\end{pmatrix}$$

定義した 5 つのパターンに関してそれぞれ 100 問ずつ解き, 平均実行時間を測定する．

表 1 計測結果

No.	動的計画法アルゴリズム	整数計画法アルゴリズム
1	25	94
2	98	173
3	204	251
4	104	163
5	16	80

計測結果を表 1 に示す．実行時間は 100 問の平均であり，単位は [ms] (ミリ秒) である．すべてのパターンにおいて, 動的計画法アルゴリズムのほうが実行時間が短く，最大で約 4 倍の速さとなっている．特に，変換後の提携タイプ数が最も多くなるパターン 3 においても動的計画法アルゴリズムが速いため，動的計画法アルゴリズムの性能の高さが分かる結果となった．

5. おわりに

本論文では，タイプ付き確率的提携構造形成（$\mathbf{PCSG}_t$）問題のアルゴリズムについて提案した．結果として，今回の実験のすべてのパターンにおいて，整数計画法アルゴリズムより動的計画法アルゴリズムのほうが速く，最大で約 4 倍の差があるという結果になった．今後の課題として，動的計画法アルゴリズムを軸にしたアルゴリズムの開発や，エージェントが 20 人より多い場合でも速く解けるのか検証する必要がある．

参考文献

[1] T. Sandholm and V. R. Lesser. Coalitions among computationally bounded agents. *Artificial Intelligence*, 94(1-2):99–137, 1997.

[2] S. Ueda, M. Kitaki, A. Iwasaki, and M. Yokoo. Concise characteristic function representations in coalitional games based on agent types. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI)*, pages 393–399, 2011.

[3] D. Y. Yeh. A dynamic programming approach to the complete set partitioning problem. *BIT Numerical Mathematics*, 26(4):467–474, 1986.