重み付き多数決ゲームにおける投票力指数の計算について

松井知己　東京大学大学院工学系研究科計数工学専攻
松井泰子　東海大学理学部情報数理学科

Calculating Power Indices of Weighted Majority Games

Tomomi Matsui　University of Tokyo, Bunkyo-ku, Tokyo 113, Japan.
tomomi@misojiro.t.u-tokyo.ac.jp
Yasuko Matsui　Tokai University, Hiratsuka-shi, Kanagawa, 259-1292, Japan.
yasuko@ss.u-tokai.ac.jp

For measuring an individual's voting power of a voting game, some power indices are proposed. In this paper, we discuss the problems for calculating Shapley-Shubik index, Banzhaf index and Deegan-Packel index of weighted majority games.

Key Words: weighted majority game, power index, NP-complete, dynamic programming, enumeration problem, isotonic regression, Monte Carlo method.

1　重みつき多数決ゲーム

重みつき多数決ゲームは、投票者がそれぞれ何票かの票を持ち、その多数決によって決定が行なわれるゲームである。投票者の集合を $N = \{1,2,\ldots,n\}$ と表し、投票者 i の持つ票数を正整数 w_i としたとき、正整数の列 $G = [q; w_1, w_2, \ldots, w_n]$ で重みつき多数決ゲームを表す。ただし q は勝つために必要な票数であり、$(1/2)\sum_{i=1}^{n} w_i < q \leq \sum_{i=1}^{n} w_i$ を満たしているとする。本稿では、投票者は票数の多い順に整列されているとする、 すなわち $w_1 \geq w_2 \geq \cdots \geq w_n$ が成り立っていると仮定する。

投票者集合 N の任意の部分集合を提携と呼ぶ。ある提携 S が、 $\sum_{i\in S} w_i \geq q$ を満たす時、 S は勝利提携であると呼び、勝利提携すべてを集めた族を $\mathcal{W}(G)$、あるいは単に $\mathcal{W}$、と書く。勝利提携でない提携を敗北提携と呼ぶ。勝利提携族の中で極小のものを、極小勝利提携と呼ぶ。任意の提携族 $\mathcal{F} \subseteq 2^N$ に対し、 $\mathcal{F}$ 中の極小集合すべてを集めた族を $\min \mathcal{F}$ と書く。これにより、極小勝利提携族は $\min \mathcal{W}$ と書くことができる。任意の投票者 i に対し、 i を含む勝利提携すべての族を $\mathcal{W}^i_+$、 i を含まない勝利提携すべての族を $\mathcal{W}^i_-$ と記す。明らかに $\forall S \in \mathcal{W}^i_-,\ S \cup \{i\} \in \mathcal{W}^i_+$ が成り立ち、 これより $|\mathcal{W}^i_-| \leq |\mathcal{W}^i_+|$ が成り立つ。投票者集合 N 上に定義された重みつき多数決ゲーム G に対し、特性関数 $v_G : 2^N \to \{0,1\}$ は

$$v_G(S) = \begin{cases} 1 & (S \in \mathcal{W}), \\ 0 & (S \notin \mathcal{W}), \end{cases}$$

と定義される。特性関数についても、混乱のおそれの無い時は、単に v と記す。本稿では、任意の集合 S と要素 e に対し、 $S \cup \{e\}$ を単に $S + e$ と書く。また、 $S \setminus \{e\}$ を単に $S - e$ と書く。

投票者集合 N 上に定義された、相異なる 2 つの重みつき多数決ゲーム G_1, G_2 において、 $\mathcal{W}(G_1) = \mathcal{W}(G_2)$ が成り立つならば、どのような提携が投票に勝利しうるのかという観点からは、 2 つのゲームは同等である。ゆえに、投票者の持つ票数が、その投票者の影響力を表しているとは限らない。極端な例としては、整数 $k \geq 2$ に

対して $q = \lfloor((n-1)k+1)/2\rfloor + 1$ を用いて定義される n 人の投票者からなるゲーム $[q;k,k,\ldots,k,1]$ がある。このゲームは、投票者数が奇数ならば、票数 1 を持つ投票者 n の持つ影響力は他の投票者とまったく変わらない。しかしながら、投票者数が偶数ならば、投票者 n は影響力を全く持たない、すなわち投票者 n を含まない任意の提携 S に対し $v(S) = v(S+n)$ が成り立つ。

投票ゲームのより詳しい解説については、[12,17,19,20] 等を参照されたい。

2 投票力指数

勝利提携族の観点から定義される投票者の影響力を捉える投票力指数がいくつか提案されている。本節では、Shapley-Shubik 指数 [15,16]、Banzhaf 指数 [1]、Deegan-Packel 指数 [5] を定義する。これらの定義の詳細な意味付けについては [5,6,12,20] 等を参照されたい。

Shapley-Shubik 指数: Shapley-Shubik 指数 (S-S 指数) は、Shapley によって定義された協力ゲームの解である Shapley 値を、Shapley と Shubik が投票ゲームに適用したものである [15,16]。投票者の集合 $\{1,2,\ldots,n\}$ の順列は $n!$ 個存在する。投票者のある順列 $\sigma = (\sigma_1,\sigma_2,\ldots,\sigma_n)$ に対し、提携 $\{\sigma_1,\sigma_2,\ldots,\sigma_{i-1}\}$ が敗北提携であり、提携 $\{\sigma_1,\sigma_2,\ldots,\sigma_{i-1},\sigma_i\}$ が 勝利提携のとき、投票者 σ_i は順列 σ におけるピヴォットと呼ぶ。すべての順列が等確率で起こると仮定したときの、各投票者のピヴォットとなる期待値を、各投票者の Shapley-Shubik 指数 (S-S 指数) と呼ぶ。投票者 i の S-S 指数を φ_i と表す。投票者 i の S-S 指数は、

$$\varphi_i = (1/n!)\sum\{(|S|-1)!(n-|S|)! : S \in \mathcal{W}, S-i \notin \mathcal{W}\},$$

と表すことも出来る。票数が $w_1 \geq w_2 \geq \cdots \geq w_n$ を満たすならば、$\varphi_1 \geq \varphi_2 \geq \cdots \geq \varphi_n$ が成り立つことは容易に示される。

投票者数が多い重みつき多数決ゲームについては、中心極限定理に基づいた S-S 指数の近似的な計算法が提案されているが、これについては [12,20] 等を参照されたい。

Banzhaf 指数: Banzhaf 指数 (Bz 指数) は、法律家 Banzhaf によって 1965 年に定義された指数である [1]。その後 Dubey と Shapley [6] によって、 公理的特徴付けがなされている。投票者の提携は全部で 2^n 存在する。ある提携 S に対して、投票者 i がみずからの投票態度を賛成から反対に、あるいは反対から賛成に変えることにより、投票の結果を変えることができるとき、投票者 i は提携 S のスウィングと呼ぶ。すべての提携が等確率で起こると仮定したときの、各投票者のスウィングとなる期待値を、各投票者の Bz 指数と呼ぶ。投票者 i の Bz 指数を β_i と書く。特性関数を用いるならば、投票者 i の Bz 指数 β_i は、

$$\beta_i = (1/2^{n-1})\sum\{v(S+i)-v(S) : S \subseteq N-i\}$$

と書くことができる。投票者 i の Bz 指数は、 Dahlingham 指数と一致しており $\beta_i = (1/2^{n-1})(|\mathcal{W}_+^i| - |\mathcal{W}_-^i|)$, と表わすこともできる [6]。 Bz 指数の総和は 1 になるとは限らない点に注意されたい。票数が $w_1 \geq w_2 \geq \cdots \geq w_n$ を満たすならば、$\beta_1 \geq \beta_2 \geq \cdots \geq \beta_n$ が成り立つ。

投票者数が多い重みつき多数決ゲームについては、中心極限定理に基づいた Bz 指数の近似的な計算法が提案されている [12,20]。

Deegan-Packel 指数: 投票者 i の Deegan-Packel 指数 (D-P 指数) γ_i は、

$$\gamma_i = \begin{cases} (1/|\min \mathcal{W}|) \sum \{1/|S| : i \in S \in \min \mathcal{W}\} & (\text{ if } \{S : i \in S \in \min \mathcal{W}\} \neq \emptyset), \\ 0 & (\text{ if } \{S : i \in S \in \min \mathcal{W}\} = \emptyset), \end{cases}$$

と定義される。 D-P 指数はその総和が 1 となる。 D-P 指数は、票数が $w_1 \geq w_2 \geq \cdots \geq w_n$ を満たしても、 $\gamma_1 \geq \gamma_2 \geq \cdots \geq \gamma_n$ が成り立つとは限らない。

3 投票力指数の計算の複雑さ

投票者集合 N 上の重みつき多数決ゲーム G が与えられたとして、いくつかの概念を定義する。ある投票者 i が、 $\forall S \subseteq N - i, v(S) = v(S + i)$ を満たすとき、投票者 i はダミーであると呼ぶ。投票者 i, j が、 $\forall S \subseteq N \setminus \{i, j\}, v(S + i) = v(S + j)$ を満たすとき、投票者 i と j は対称であるという。

定理 1 [7,11]重みつき多数決ゲーム $[q; w_1, \ldots, w_n]$ において、 $w_1 \geq w_2 \geq \cdots \geq w_n$ が成り立っているものとする。次の 2 つの問題は *NP*- 完全である。

(1) 投票者 n がダミーでないことを判定せよ。

(2) 投票者 1 と 2 が対称でないことを判定せよ。

証明：概略のみを記す。投票者 n がダミーでない必要十分条件は、投票者 n を含む極小勝利提携が存在することである。投票者 1 と 2 が対称でなければ、 $w_1 \geq w_2$ より、極小勝利提携で投票者 1 を含み投票者 2 を含まないものが存在する。ゆえに問題 $(1)(2)$ の両方とも、答が YES の際に多項式サイズの保証があり、クラス NP に属する。

NP- 完全性の証明には、分割問題を多項式時間帰着する。分割問題とは、正整数の集合 $S = \{a_1, a_2, \ldots, a_k\}$ が与えられたとき、 $\sum_{i \in S'} a_i = (1/2)M$ を満たす $S' \subseteq S$ の存在を問う問題である、ただし $M = \sum_{i \in S} a_i$ である [7,9]。 $(1)(2)$ を証明するためには、 $G_1 = [q_1; a_1, a_2, \ldots, a_n, 1]$ と $G_2 = [q_2; M + 1, M, a_1, \ldots, a_n]$ というゲームを構築する。必要票数 q_1, q_2 は、過半数をとれば勝利提携となるように設定する。すると、 (a) 分割問題の解が YES である、 (b) G_1 において投票者 n がダミーでない、 (c) G_2 において投票者 1 と 2 が対称でない、の 3 つの条件は同値である。 □

上記の定理より、入力サイズが非常に大きな重みつき多数決ゲームに対し、投票者の持つ影響力の分析は困難であることが予想される。また問題 (2) の NP- 完全性より、投票者を票数の多い者に限っても、影響力の分析は困難と予想される。上記の定理中 (1) の問題の NP- 完全性については、 Garey and Johnson [7] においても言及されているが (p.280, 問題 [MS8] 参照) 証明は省略されている。

S-S 指数、 Bz 指数、 D-P 指数はどれも、投票者がダミーである事と、対応する指数が 0 となる事が必要十分条件である。これより以下が導かれる [7,11]。

系 1 重みつき多数決ゲーム $[q; w_1, \ldots, w_n]$ は、 $w_1 \geq w_2 \geq \cdots \geq w_n$ を満たしているとする。このとき投票者 n に対する *S-S* 指数 φ_n、 *Bz* 指数 β_n、 *D-P* 指数 γ_n を計算する問題は、どれも *NP*- 困難である。

Garey and Johnson [7] (p.280, 問題 [MS8]) では、 上記の結果の内 S-S 指数についてのみ記述されている。

S-S 指数と Bz 指数は、 2 人の投票者が対称である事と対応する指数が等しいことは必要十分条件なので、以下の性質が成り立つ [11]。

系 **2** 重みつき多数決ゲーム $[q; w_1, \ldots, w_n]$ は、 $w_1 \geq w_2 \geq \cdots \geq w_n$ を満たしているとする。このとき投票者 1 と 2 に対する S-S 指数 (φ_1, φ_2)、 Bz 指数 (β_1, β_2)、を計算する問題は、どちらも NP- 困難である。

上記の性質より、票数の大きな投票者に限って S-S 指数や Bz 指数を求める問題も、効率的に解くことは困難であると予想される。

S-S 指数を求める問題については、以下の結果も知られている [13]。

定理 **2** 重みつき多数決ゲーム $[q; w_1, \ldots, w_n]$ において、 $w_1 \geq w_2 \geq \cdots \geq w_n$ が成り立っているものとする。このとき、投票者 n に対する S-S 指数 φ_n、 Bz 指数 β_n を計算する問題は、どちらも $\#P$- 完全である。

この定理は [13] において、 S-S 指数の際について証明されているが、 Bz 指数についても全く同様に証明できる。

最も票数の大きな投票者と最も票数の小さな投票者の影響力の比較については、次の定理が成り立つ。

定理 **3** 重みつき多数決ゲーム $[q; w_1, \ldots, w_n]$ において、 $w_1 \geq w_2 \geq \cdots \geq w_n$ が成り立っているものとする。このとき、以下の条件は互いに必要十分である。
(1) 投票者 1 と投票者 n は対称である。
(2) 任意の 2 人の投票者は対称である。
(3) ある自然数 k が存在して $\mathcal{W} = \{S \subseteq N : |S| \geq k\}$ が成り立つ。
(4) ある自然数 k が存在して

$$w_{n-(k-1)} + \cdots + w_{n-1} + w_n \geq q > w_1 + w_2 + \cdots + w_{k-1} \tag{1}$$

を満たす。(上式の最左辺は k 項の和であり、 最右辺は $k-1$ 項の和であることに注意されたい。)
証明： (1) と (2) が必要十分であるのは明らか。また (3) から (2) も明らか。次に (2) から (3) を導く。整数 k を $k = \min\{|S| : S \in \mathcal{W}\}$ と定義し、 最小値を達成する提携を S^* とする。任意の敗北提携 S' と投票者の対 (i, j) において、 全ての投票者対が対称であることから $i \in S' \not\ni j$ ならば $S' - i + j \notin \mathcal{W}$ である。ある敗北提携 S' が $|S'| \geq k$ を満たすならば、 敗北提携で S^* を含むものを作ることができるため、 矛盾が導かれる。ゆえに (3) が成り立つ。 (3) から (4) は明らか。最後に条件 (4) が成り立つと仮定して (3) を導く。要素数が k 以上の任意の提携 S に対し、 $\sum_{i \in S} w_i \geq w_{n-(k-1)} + \cdots + w_n \geq q$ が成り立ち、明らかに $S \in \mathcal{W}$ である。次に要素数が k 未満の任意の提携 S' に対し、 $\sum_{i \in S'} w_i \leq w_1 + \cdots + w_{k-1} < q$ となり S' は敗北提携となる。 □

上記の定理より、最も票数の多い投票者と最も票数の少ない投票者が対称であるかは、$\mathrm{O}(n)$ 時間の計算で判定することができる。

4 動的計画法

本節では、任意の投票者 i について $w_i < q$ を仮定する。また投票者を、その票数に応じて分類したものを $N_1, N_2, \ldots, N_z$ とする、すなわち

$$\begin{aligned}
&N_1 \cup N_2 \cup \cdots \cup N_z = N, \\
&1 \leq \forall x < \forall y \leq z,\ \forall i \in N_x,\ \forall j \in N_y,\ w_i > w_j, \\
&1 \leq \forall x \leq z,\ \forall i, \forall j \in N_x,\ w_i = w_j,
\end{aligned}$$

が成り立っている。さらに $|N_x|$ を n_x と書き、 N_x 中の各投票者の票数を $\overline{w}_x$ と書く。

4.1 前処理

本節では、動的計画法による前処理について述べる。前処理の目的は次の 2 つである; (1) ダミーの投票者すべての集合を求める、 (2) 極小勝利提携の総数を求める。ダミー投票者の集合が求められた際は、それらの投票者を抜いてしまっても、他の投票者の S-S 指数、 Bz 指数、 D-P 指数はどれも変わらない。ゆえに、本節以降の動的計画法を実行する際は、本節の算法を予め実行し、ダミー投票者を (あれば) 取り除く事によって計算を軽減するのが望ましい。また (2) の結果は、次節の算法において D-P 指数を計算する際に用いる。

ダミーの投票者集合を求めるには、以下の定理を用いる。

定理 4 重みつき多数決ゲーム $[q; w_1, \ldots, w_n]$ において、 $w_1 \geq w_2 \geq \cdots \geq w_n$ が成り立っているとする。投票者 j がダミーである必要十分条件は $\alpha_j + w_j + w_{j+1} + \cdots + w_n < q$ である。ただし

$$\alpha_j = \max\left\{ \sum_{i \in S} w_i \ : S \subseteq \{1, 2, \ldots, j-1\},\ S \notin \mathcal{W} \right\}$$

と定義する。

証明：不等式 $\alpha_j + w_j + w_{j+1} + \cdots + w_n \geq q$ が成り立つと仮定して、投票者 j がダミーでないことを示す。仮定より、不等式

$$\alpha_j + w_j + w_{j+1} + \cdots + w_{k-1} < q \leq \alpha_j + w_j + w_{j+1} + \cdots + w_k$$

を満たす投票者 $k \in \{j, j+1, \ldots, n\}$ が存在する。すると、

$$\alpha_j + w_{j+1} + \cdots + w_k \leq \alpha_j + w_j + w_{j+1} + \cdots + w_{k-1} < q \leq \alpha_j + w_j + w_{j+1} + \cdots + w_k$$

が成り立ち、投票者 j はダミーではない。

次に投票者 j がダミーでないと仮定する。すると、ある提携 $S \subseteq N - j$ が存在して $S \notin \mathcal{W}$ かつ $S + j \in \mathcal{W}$ が成り立つ。提携 $S' = S \cap \{1, 2, \ldots, j-1\}$ と定義すると、不等式

$$\sum_{i \in S'} w_i \leq \sum_{i \in S} w_i < q \leq \sum_{i \in S} w_i + w_j \leq \sum_{i \in S'} w_i + w_j + w_{j+1} + \cdots + w_n$$

が成り立つ。値 α_j の定義より $\sum_{i \in S'} w_i \leq \alpha_j$ が成り立つことから、

$$q \leq \sum_{i \in S'} w_i + w_j + w_{j+1} + \cdots + w_n \leq \alpha_j + w_j + w_{j+1} + \cdots + w_n$$

が満たされる。 □

上記の定理より、 $(\alpha_2, \alpha_3, \ldots, \alpha_n)$ が求まればダミー投票者を特定することができる。各値 α_j $(j = 2, 3, \ldots, n)$ を求める問題はナップサック問題と呼ばれ、擬似多項式時間算法の存在が知られている (例えば [3,18] 参照) 。

任意の非負整数の対 w と x に対し、提携族

$$\{S \subseteq N : \sum_{i \in S} w_i = w,\ S \cap N_x \neq \emptyset,\ S \cap N_{x+1} = \cdots = S \cap N_z = \emptyset,\}$$

の要素数を $c(w,x)$ と表すと、 極小勝利提携の数は $|\min \mathcal{W}| = \sum_{x=1}^{z} \sum_{w=q}^{q+\overline{w}_x-1} c(w,x)$, として計算される。また投票者 i がダミーである必要十分条件は、 $i \in N_x$ ならば

$$\alpha'_x + \overline{w}_x n_x + \overline{w}_{x+1} n_{x+1} + \cdots + \overline{w}_z n_z < q$$

となる。ただし

$$\alpha'_x = \max\{w : 0 \leq w \leq q-1 \text{ かつ } 1 \leq \exists y \leq x-1, c(w,y) > 0\}$$

である。

動的計画法を用いて $\{c(w,x) : 0 \leq w \leq q-1,\ 1 \leq x \leq z\}$ を求める以下の解法が構築できる。解法の第 x 反復終了時において、 c^* の値が $c(w,1) + c(w,2) + \cdots + c(w,x)$ となっており、 α^* の値が α'_x となっている。

begin
  for $w = 1,2,\ldots,q-1$ **do** $c(w) := 0$;
  $c(0) := 1$; $c^* := 0$; $\alpha^* := 0$; $w' := w_1 + \cdots + w_n$;
  for $x = 1,2,\ldots,z$ **do** {
    $w' := w' - \overline{w}_x * n_x$;
    for $w = \alpha^*, \alpha^*-1, \ldots, 0$ **do** {
      if $c(w) > 0$ **then** {
        $c' := 1$; (以下の反復では $c' = {}_{n_x}\mathrm{C}_y$ が成り立っている)
        for $y = 1,2,\ldots,n_x$ **do** {
          $c' := c' * (n_x - y + 1)/y$;
          if $w + \overline{w}_x * y \leq q-1$ **then** {
            $c(w + \overline{w}_x * y) := c(w + \overline{w}_x * y) + c(w) * c'$;
            $\alpha^* := \max\{\alpha^*, w + \overline{w}_x * y\}$
          };
          else if $q \leq w + \overline{w}_x * y \leq q-1+\overline{w}_x$ **then** $c^* := c^* + c(w) * c'$
    } } };
    if $\alpha^* + w' \leq q-1$ **then** {**output** $(c^*, x+1)$; **STOP** }
  };
  output $(c^*,$ "no dummy player")
end

上記の算法が数値の対 $(c^*, x+1)$ を出力して終了したときは、 c^* は極小勝利提携の総数を表し、 $N_{x+1} \cup N_{x+2} \cup \cdots \cup N_z$ が ダミー投票者の集合である。この算法の時間計算量は $\mathrm{O}(nq)$、記憶容量は $\mathrm{O}(n+q)$ である。

4.2　投票力指数を求める動的計画法

S-S 指数と Bz 指数については、これを計算する動的計画法が提案されている [4,10]。

以下では、 [4,10] の動的計画法を改訂することで、 S-S 指数、 Bz 指数、 D-P 指数の 3 つの指数を同時に、擬似多項式時間で求める動的計画法を提案する。

投票者 i に対し、提携族

$$\{S \subseteq N-i : \sum_{i \in S} w_i = w,\ |S| = t,\ S \cap N_x \neq \emptyset,\ S \cap N_{x+1} = \cdots = S \cap N_z = \emptyset,\}$$

の要素数を $c(w,t,x)$ と表す。すると、集合 N_y に含まれている投票者 i の指数は、

$$\begin{aligned}
\varphi_i &= \sum_{t=1}^{n-1} \sum_{w=q-w_i}^{q-1} \sum_{x=1}^{z} \frac{t!(n-t-1)!}{n!} c(w,t,x), \\
\beta_i &= (1/2^{n-1}) \sum_{t=1}^{n-1} \sum_{w=q-w_i}^{q-1} \sum_{x=1}^{z} c(w,t,x), \\
\gamma_i &= \left(\sum_{x=1}^{y} \sum_{w=q-w_i}^{q-1} \sum_{t=1}^{n-1} \frac{c(w,t,x)}{t+1} + \sum_{x=y+1}^{z} \sum_{w=q-w_i}^{q-1-w_i+w_x} \sum_{t=1}^{n-1} \frac{c(w,t,x)}{t+1} \right) \frac{1}{|\min \mathcal{W}|},
\end{aligned}$$

として計算される ($i \in N_y$ の性質は γ_i 計算の際用いられている) 。すると以下の算法を構築できる。

begin
　for $(w,t) \in \{1,2,\ldots,q-1\} \times \{1,2,\ldots,t\}$ **do** $c(w,t) = 0$;
　$c(0,0) := 1$; $t^* := 0$; $\varphi^* := 0$; $\beta^* := 0$; $\gamma^* := 0$;
　for $t = 0,1,\ldots,n-1$ **do** $\alpha(t) := 0$;
　for $x = 1,2,\ldots,z$ **do** $\{$
　　$y' := |N_x - i|$;
　　for $t = t^*, t^*-1, \ldots, 0$ **do** $\{$
　　　for $w = \alpha(t), \alpha(t)-1, \ldots, 0$ **do** $\{$
　　　　if $c(w,t) > 0$ **then** $\{$
　　　　　$c' := 1$; (以下の反復では $c' =_{y'} \mathrm{C}_y$ が成り立っている)
　　　　　for $y = 1,2,\ldots,y'$ **do** $\{$
　　　　　　$c' := c' * (y'-y+1)/y$;
　　　　　　if $w + \overline{w}_x * y \leq q-1$ **then** $\{$
　　　　　　　$c(w + \overline{w}_x * y, t+y) := c(w + \overline{w}_x * y, t+y) + c(w,t) * c'$;
　　　　　　　$\alpha(t+y) := \max\{\alpha(t+y), w + \overline{w}_x * y\}$
　　　　　　$\}$;
　　　　　　if $q - w_i \leq w + \overline{w}_x * y \leq q-1$ **then** $\{$
　　　　　　　$\varphi^* := \varphi^* + (t+y)!(n-t-y-1)!\, c(w,t) * c'$;
　　　　　　　$\beta^* := \beta^* + c(w,t) * c'\}$;
　　　　　　if $q - w_i \leq w + \overline{w}_x * y \leq \min\{q-1, q-1-w_i+\overline{w}_x\}$ **then** $\{$
　　　　　　　$\gamma^* := \gamma^* + (1/(t+y+1)) * c(w,t) * c'\}$
　　$\}\ \}\ \}\ \}$;
　　$t^* := t^* + y'$

};
output $(\varphi^*/n!,\ \beta^*/2^{n-1},\ \gamma^*/|\min \mathcal{W}|)$
end

この算法の時間計算量は $\mathrm{O}(n^2q)$、記憶容量は $\mathrm{O}(nq)$ である。

5 列挙算法

本節では各指数を計算する列挙算法について議論する。

以下では、本節で用いる記号を定義する。正の実数ベクトル $\boldsymbol{a} = (a_1, a_2, \ldots, a_k)$ と実数 b に対し、$\mathcal{F}(\boldsymbol{a}, b) \equiv \{F \subseteq \{1, 2, \ldots, k\} : \sum_{i \in F} a_i \geq b\}$ と定義する。ベクトル $\boldsymbol{f}(\boldsymbol{a}, b) \equiv (f_1, f_2, \ldots, f_k)$ は $f_i = |\{S \in \mathcal{F}(\boldsymbol{a}, b) : |S| = i\}|$ を満たすものとする。

5.1 D-P 指数を求める列挙算法

極小勝利提携を実際に全列挙すれば、D-P 指数は容易に計算できる。本節では極小勝利提携の高速な列挙算法について議論する。

本節では、正整数ベクトル $\boldsymbol{a} = (a_1, a_2, \ldots, a_n)^T$ と整数 b が与えられたとして、集合族 $\mathcal{F}(\boldsymbol{a}, b)$ の極小集合を全て列挙する問題の解法を記述する。下記の解法の基本的なアイデアは、以下の定理を用いた分割統治法である。

定理 5 正の実数ベクトル $\boldsymbol{a} = (a_1, a_2, \ldots, a_n)^T$ と実数 b は、$a_1 \geq a_2 \geq \cdots \geq a_n > 0$ と $a_1 + a_2 + \cdots + a_n \geq b$ を満たしているとする。このとき

$$\mathcal{F}^1_+ \equiv \{S : 1 \in S \in \mathcal{F}(\boldsymbol{a}, b)\},\ \ \mathcal{F}^1_- \equiv \{S : 1 \notin S \in \mathcal{F}(\boldsymbol{a}, b)\},$$

と定義する。すると $\emptyset \notin \mathcal{F}(\boldsymbol{a}, b)$ ならば、$\min \mathcal{F}(\boldsymbol{a}, b)$ は $\min \mathcal{F}^1_+$ と $\min \mathcal{F}^1_-$ の直和となっている。

証明： 定義より明らかに、$\min \mathcal{F}(\boldsymbol{a}, b) \subseteq \min \mathcal{F}^1_+ \cup \min \mathcal{F}^1_-$ と $\min \mathcal{F}(\boldsymbol{a}, b) \supseteq \min \mathcal{F}^1_-$ が成り立つ。もし $\exists S \in \min \mathcal{F}^1_+ - \min \mathcal{F}(\boldsymbol{a}, b)$ を仮定すると、$S - 1 \in \mathcal{F}(\boldsymbol{a}, b)$ であり、$\sum_{i \in S-1} a_i \geq b$ が成り立っている。$a_1 \geq \cdots \geq a_n$ より $\exists j \in S - 1$ ならば $\sum_{i \in S-j} a_i \geq \sum_{i \in S-1} a_i \geq b$ より $S \in \min \mathcal{F}^1_+$ に矛盾する。ゆえに $S = \{1\}$ であるが、$S - 1 = \emptyset \in \mathcal{F}(\boldsymbol{a}, b)$ が導かれ、矛盾。 □

上記定理中の集合族 $\mathcal{F}^1_+$ と $\mathcal{F}^1_-$ は、$\boldsymbol{a}' = (a_2, a_3, \ldots, a_n)$ とすると、

$$\mathcal{F}^1_+ = \{S + 1 : S \in \min \mathcal{F}(\boldsymbol{a}', b - a_1)\},\ \ \mathcal{F}^1_- = \mathcal{F}(\boldsymbol{a}', b)$$

と書くことが出来る。これを用いて以下の算法を構築することが出来る。

Input: 正整数ベクトル $\boldsymbol{a} = (a_1, a_2, \ldots, a_n)^T$ と正整数 b 、
ただし $a_1 \geq a_2 \geq \cdots \geq a_n$ かつ $a_1 + a_2 + \cdots + a_n \geq b$ を満たす。
Output: 集合族 $\mathcal{F}(\boldsymbol{a}, b)$ の全ての極小集合。
begin(main routine)
ENUMERATE $(\emptyset, 1, b)$
end

procedure ENUMERATE (S', n', b')
  begin
    if $n' = n$ **then** {**output** $S' \cup \{n\}$; **return** };
    else {
      if $(a_{n'+1} + \cdots + a_n) \geq b'$ **then** ENUMERATE $(S', n'+1, b')$;
      if $a_{n'} \geq b'$ **then** {**output** $S' \cup \{n'\}$};
      else ENUMERATE $(S' \cup \{n'\}, n'+1, b' - a_{n'})$;
      return }
  end

上記の再帰呼び出し算法は、極小集合を 1 つ出力するか **procedure** ENUMERATE を少なくとも 1 回再帰呼び出しする。 **procedure** ENUMERATE を連続して $n+1$ 回以上再帰呼び出しする事は無いので、 **procedure** ENUMERATE が実行される回数は、出力する極小集合の数の n 倍以下となる。 **procedure** ENUMERATE の各行の実行時間は、上記算法中の $(a_{n'+1} + \cdots + a_n)$ の値を常に更新しながら注意深く保持することにより、 $\mathrm{O}(1)$ 時間で実装することができる。ゆえに上記の算法の総実行時間は $\mathrm{O}(n|\min \mathcal{F}(\boldsymbol{a}, b)|)$、必要記憶容量は $\mathrm{O}(n)$ である。

上記の算法を用いて極小勝利提携が出力されるたびに、各投票者の D-P 指数を更新することにより、投票者全員分の D-P 指数を時間計算量 $\mathrm{O}(n|\min \mathcal{W}|)$、記憶容量 $\mathrm{O}(n)$ で計算することができる。

5.2 S-S 指数と Bz 指数を求める列挙算法

本節では投票者 i の S-S 指数と Bz 指数を計算する列挙算法を提案する。投票者 i に対応する要素のみを取り除いた票数ベクトル $(w_1, w_2, \ldots, w_{i-1}, w_{i+1}, \ldots, w_n)^T$ を改めて $\boldsymbol{w}' = (w'_1, w'_2, \ldots, w'_{n-1})^T$と表し、 $\boldsymbol{f}^+ = \boldsymbol{f}(\boldsymbol{w}', q)$ と $\boldsymbol{f}^- = \boldsymbol{f}(\boldsymbol{w}', q - w_i)$ と書く。すると定義より明らかに

$$\varphi_i = \sum_{t=1}^{n-1} \frac{t!(n-t-1)!}{n!}(f_t^+ - f_t^-), \quad \beta_i = (1/2^{n-1}) \sum_{t=1}^{n-1} (f_t^+ - f_t^-)$$

が成り立つ。

以下では正整数ベクトル $\boldsymbol{a} = (a_1, a_2, \ldots, a_k)$ と整数 b が与えられたとして、ベクトル $\boldsymbol{f}(\boldsymbol{a}, b)$ を求める算法を記述する。下記算法中では、 n に対する 2 項係数ベクトルを θ_k と表す、すなわち $\theta_k = ({}_k\mathrm{C}_0, {}_k\mathrm{C}_1, \ldots, {}_k\mathrm{C}_k)^T$ である。また k 次元のゼロベクトルを $\mathbf{0}_k$ と書く。

Input: 正整数ベクトル $\boldsymbol{a} = (a_1, a_2, \ldots, a_n)^T$ と正整数 b、
  ただし $a_1 \geq a_2 \geq \cdots \geq a_n > 0$ かつ $a_1 + a_2 + \cdots + a_n \geq b$ を満たす。
Output: ベクトル $\boldsymbol{f}(\boldsymbol{a}, b)$。
begin(main routine)
  SHELLING $(1, b)$;
end
function SHELLING (n', b')
  begin

if $n' = n$ **then return** $((0,1))$;
else $\{$
if $(a_{n'+1} + \cdots + a_n) \geq b'$ **then** $\boldsymbol{f}^1 :=$ SHELLING $(n'+1, b')$; **else** $\boldsymbol{f}^1 := \mathbf{0}_{n-n'}$;
if $a_{n'} \geq b'$ **then** $\boldsymbol{f}^2 := \theta_{n-n'}$; **else** $\boldsymbol{f}^2 :=$ SHELLING $(n'+1, b'-a_{n'})$;
$\boldsymbol{f}' := (f_1^1, f_2^1 + f_1^2, f_3^1 + f_2^2, \ldots, f_{n'}^1 + f_{n'-1}^2, f_{n'}^2)$;
return $(\boldsymbol{f}')$ $\}$
end

上記の再帰呼び出し算法も、 **procedure** SHELLING が実行される回数は、 極小集合の数の n 倍以下となる。 2 項係数表があれば **function** 中の各行の計算時間は $\mathrm{O}(n)$ で抑えられる。ゆえに総計算量は $\mathrm{O}(n^2|\min \mathcal{F}(\boldsymbol{a}, b)|)$ で抑えられる。必要記憶容量は、再帰呼び出しの深さが最大 n であることから長さ n 以下のベクトルを最大 n 本保持出来れば良い。

上記の算法を用いて $\boldsymbol{f}^+$ と $\boldsymbol{f}^-$ を求める計算量は $\mathrm{O}(n^2|\min \mathcal{W}_+^i| + n^2|\min \mathcal{W}_-^i|) = \mathrm{O}(n^2|\min \mathcal{W}|)$ となる。ただし 2 項係数表を作る前処理として $\mathrm{O}(n^2)$ の時間が必要である。以上より、投票者 i の S-S 指数と Bz 指数を、時間計算量 $\mathrm{O}(n^2|\min \mathcal{W}|)$、記憶容量 $\mathrm{O}(n^2)$ で計算することができる。

6 モンテカルロ法

本節では、ランダムサンプリングに基づいた S-S 指数の最尤推定法について議論する。

6.1 S-S 指数を求めるモンテカルロ法

S-S 指数の定義より、投票者の順列すべてが等確率で起こると仮定したときの、各投票者のピヴォットとなる期待値が、 S-S 指数である。

投票者の順列をランダムに p^* 個発生させたとき、各々の投票者がピヴォットになる回数を表す確率変数を $P_1, P_2, \ldots, P_n$ と表す。確率変数 $P_1, \ldots, P_n$ の同時分布は多項分布となり、 $\Pr[P_1 = p_1, P_2 = p_2, \ldots, P_n = p_n] = \dfrac{p!}{p_1! p_2! \cdots p_n!} \varphi_1^{p_1} \varphi_2^{p_2} \cdots \varphi_n^{p_n}$ で与えられる、ただし $\boldsymbol{p} = (p_1, \ldots, p_n)^T$ は総和が p^* となる非負整数の組とする。

上記の多項分布の定義式を尤度関数としたとき、標本 $\boldsymbol{p} = (p_1, p_2, \ldots, p_n)^T$ から得られる、投票者 i の S-S 指数の最尤推定値は p_i/p^* である。

投票者は票数にしたがって整列されているとする、すなわち $w_1 \geq w_2 \geq \cdots \geq w_n$ という仮定より、各投票者の S-S 指数は $\varphi_1 \geq \varphi_2 \geq \cdots \geq \varphi_n$ を満たしているが、各投票者の推定値 p_i/p^* は上記のような単調性を満たすとは限らない。そのため上記の単調性を保った最尤推定を行なう必要がある。 S-S 指数の単調性を満たす最尤推定値は以下の最適化問題の最適解として求められる、

$$\begin{aligned} \mathrm{MLE}(\boldsymbol{p}) : \text{maximize} \quad & p_1 \log x_1 + p_2 \log x_2 + \cdots + p_n \log x_n \\ \text{subject to} \quad & x_1 \geq x_2 \geq \cdots \geq x_n \geq 0, \qquad (2) \\ & x_1 + x_2 + \cdots x_n = 1, \qquad (3) \end{aligned}$$

ただし上記の目的関数は対数尤度の定数項を取り除いたものである。以下ではこの最適化問題の解法について議論する。

任意の非負ベクトル $\boldsymbol{p} = (p_1, \ldots, p_n)^T$ に対して、ベクトル $\boldsymbol{p}^{\mathrm{c}} = (p_0^{\mathrm{c}}, p_1^{\mathrm{c}}, \ldots, p_n^{\mathrm{c}})^T$ は $p_0^{\mathrm{c}} = 0$ と $p_i^{\mathrm{c}} = p_1 + \cdots + p_i$ を満たすベクトルとし、ベクトル $\boldsymbol{p}$ の累積ベクトルと呼ぶ。また任意の非負ベクトル $\boldsymbol{p} = (p_1, p_2, \ldots, p_n)^T$ に対して、関数 $f : [0, n] \to \mathbf{R}$ は、 $f(0) = 0$ かつ $\forall i \in \{1, \ldots, n\},\ f(i) \geq p_1 + p_2 + \cdots + p_i$ を満たすすべての連続凹関数の下限を達成する関数とし、このときベクトル

$$(f(1) - f(0), f(2) - f(1), \ldots, f(n) - f(n-1))^T$$

を $\widehat{\boldsymbol{p}} = (\widehat{p}_1, \widehat{p}_2, \ldots, \widehat{p}_n)^T$ と書き、 $\widehat{\boldsymbol{p}}$ を $\boldsymbol{p}$ の単調回帰 (isotonic regression) と呼ぶ。

問題 MLE において、以下の定理が成り立つ。

定理 6 正ベクトル $\boldsymbol{p} = (p_1, \ldots, p_n)^T$ に対して、 $(1/p^*)\widehat{\boldsymbol{p}}$ は最適化問題 $MLE(\boldsymbol{p})$ の最適解となる。ただし $p^* = p_1 + p_2 + \cdots + p_n$ である。

証明： 単調回帰の定義より、 $(\widehat{p}_i^{\mathrm{c}} - p_i^{\mathrm{c}})$ が正ならば $\widehat{p}_i = \widehat{p}_{i+1}$ である。ゆえに $i = 1, 2, \ldots, n-1$ に対し、 $(\widehat{p}_i^{\mathrm{c}} - p_i^{\mathrm{c}})(\log \widehat{p}_i - \log \widehat{p}_{i+1}) = 0$ が成り立つ。これより、

$$\begin{aligned} 0 &= \sum_{i=1}^{n-1} (\widehat{p}_i^{\mathrm{c}} - p_i^{\mathrm{c}})(\log \widehat{p}_i - \log \widehat{p}_{i+1}) + (\widehat{p}_n^{\mathrm{c}} - p_n^{\mathrm{c}}) \log \widehat{p}_n \\ &= (\widehat{p}_1^{\mathrm{c}} - p_1^{\mathrm{c}}) \log \widehat{p}_1 + \sum_{i=2}^{n} (-(\widehat{p}_{i-1}^{\mathrm{c}} - p_{i-1}^{\mathrm{c}}) + (\widehat{p}_i^{\mathrm{c}} - p_i^{\mathrm{c}})) \log \widehat{p}_i \\ &= (\widehat{p}_1 - p_1) \log \widehat{p}_1 + \sum_{i=2}^{n} (\widehat{p}_i - p_i) \log \widehat{p}_i = \sum_{i=1}^{n} (\widehat{p}_i - p_i) \log \widehat{p}_i \end{aligned}$$

が成り立つ。すなわち、 $\sum_{i=1}^n p_i \log \widehat{p}_i = \sum_{i=1}^n \widehat{p}_i \log \widehat{p}_i$ となる。

問題 $\mathrm{MLE}(\widehat{\boldsymbol{p}})$ の最適解について議論する。制約式 (2) を問題 $\mathrm{MLE}(\widehat{\boldsymbol{p}})$ より取り除いた緩和問題の KKT 条件は

$$\widehat{p}_i / x_i - \lambda = 0 \ \ (i = 1, \ldots, n), \quad x_1 + x_2 + \cdots + x_n = 1$$

となる。ゆえに $\boldsymbol{x} = (1/p^*)\widehat{\boldsymbol{p}} = (\widehat{p}_1/p^*, \ldots, \widehat{p}_n/p^*)^T$ と $\lambda = p^*$ は KKT 条件を満たし、この緩和問題の最適解である。解 $(1/p^*)\widehat{\boldsymbol{p}}$ が制約式 (2) を満たすことから $(1/p^*)\widehat{\boldsymbol{p}}$ は $\mathrm{MLE}(\widehat{\boldsymbol{p}})$ の最適解でもある。これより $\mathrm{MLE}(\boldsymbol{p})$ の任意の許容解 $\boldsymbol{x}$ において、 $\boldsymbol{x}$ は $\mathrm{MLE}(\widehat{\boldsymbol{p}})$ の許容解でもあるため、 $\sum_{i=1}^n \widehat{p}_i \log(\widehat{p}_i/p^*) \geq \sum_{i=1}^n \widehat{p}_i \log x_i$ が成り立つ。

値 $(\widehat{p}_{i-1}^{\mathrm{c}} - p_{i-1}^{\mathrm{c}})$ は定義より非負である。また、 $\mathrm{MLE}(\boldsymbol{p})$ の任意の許容解 $\boldsymbol{x}$ において、制約式 (2) より $\log x_i - \log x_{i+1} \geq 0$ が成り立つ。これより、

$$0 \leq \sum_{i=1}^{n-1} (\widehat{p}_i^{\mathrm{c}} - p_i^{\mathrm{c}})(\log x_i - \log x_{i+1}) + (\widehat{p}_n^{\mathrm{c}} - p_n^{\mathrm{c}}) \log x_n = \sum_{i=1}^{n} (\widehat{p}_i - p_i) \log x_i$$

が成り立つ。すなわち、 $\sum_{i=1}^n \widehat{p}_i \log x_i \geq \sum_{i=1}^n p_i \log x_i$ が示された。

以上より、 $\mathrm{MLE}(\boldsymbol{p})$ の任意の許容解 $\boldsymbol{x}$ において、 次の不等式

$$\begin{aligned} \sum_{i=1}^{n} p_i \log(\widehat{p}_i/p^*) &= \sum_{i=1}^{n} p_i \log \widehat{p}_i - \sum_{i=1}^{n} p_i \log p^* = \sum_{i=1}^{n} \widehat{p}_i \log \widehat{p}_i - \sum_{i=1}^{n} \widehat{p}_i \log p^* \\ &= \sum_{i=1}^{n} \widehat{p}_i \log(\widehat{p}_i/p^*) \geq \sum_{i=1}^{n} \widehat{p}_i \log x_i \geq \sum_{i=1}^{n} p_i \log x_i \end{aligned}$$

が成立し、 $(1/p^*)\widehat{\boldsymbol{p}}$ は $\mathrm{MLE}(\boldsymbol{p})$ の最適解となる。 □

単調回帰については文献 [2] に詳しい議論がされており、上記の定理についてもより一般的な設定で証明されている。

単調回帰ベクトルは、 2 次元座標平面上の点集合 $Q = \{(i, q_i) : i = 0, 1, \ldots, n\}$ の凸包を求めることにより、容易に計算される。集合 Q 中の点は、バケット整列法により $\mathrm{O}(n)$ 時間で、 第一座標の値によって整列することができる。第一座標の値で整列した後は、 Graham の走査法 [8] により $\mathrm{O}(n)$ 時間で凸包を求めることができる （例えば [14] 3.3.2 節参照) 。単調回帰ベクトルは、 [2] に述べられている Pool-Adjacent-Violators algorithm でも求めることができる。この算法も、注意深く実装することにより計算時間は $\mathrm{O}(n)$ となる。

6.2 Bz 指数を求めるモンテカルロ法

Bz 指数の定義より、投票者の提携すべてが等確率で発生すると仮定したときの、各投票者のスウィングとなる確率が Bz 指数である。

提携をランダムに p^* 個発生させたとき、各々の投票者がスウィングになる回数を表す確率変数を $P_1, P_2, \ldots, P_n$ と表す。 Bz 指数の単調性を満たす最尤推定値は以下の最適化問題の最適解として求められる、

$$\begin{aligned} \mathrm{MLE'}(\boldsymbol{p}) : \text{maximize} \quad & \sum_{i=1}^{n} (p_i \log x_i + (p^* - p_i) \log(1 - x_i)) \\ \text{subject to} \quad & 1 \geq x_1 \geq x_2 \geq \cdots \geq x_n \geq 0, \end{aligned} \tag{4}$$

ただし上記の目的関数は対数尤度の定数項を取り除いたものである。このとき、以下の定理が成り立つ。

定理 7 正ベクトル $\boldsymbol{p} = (p_1, \ldots, p_n)^T$ に対して、ベクトル $(1/p^*)\widehat{\boldsymbol{p}}$ は最適化問題 $MLE'(\boldsymbol{p})$ の最適解となる、ただし $p^* = p_1 + p_2 + \cdots + p_n$ である。

証明： 証明の概略は、定理 6とほぼ同じである。以下では $h(\boldsymbol{p}, \boldsymbol{x}) = \sum_{i=1}^{n} (p_i \log x_i + (p^* - p_i) \log(1 - x_i))$ と置く。

単調回帰の定義より、 $(\widehat{p}_i^{\mathrm{c}} - p_i^{\mathrm{c}})$ が正ならば $\widehat{p}_i = \widehat{p}_{i+1}$ である。さらに、問題 MLE' の目的関数は $\boldsymbol{p}$ に関して線形であることから、 $h(\widehat{\boldsymbol{p}}, \widehat{\boldsymbol{p}}) = h(\widehat{\boldsymbol{p}}, \boldsymbol{p})$ が成り立つ。ベクトル $(1/p^*)\widehat{\boldsymbol{p}}$ は、問題 $\mathrm{MLE'}(\widehat{\boldsymbol{p}})$ の最適解となるのは明らかであろう。問題 $\mathrm{MLE'}(\boldsymbol{p})$ の任意の許容解 $\boldsymbol{x}$ は $\mathrm{MLE'}(\widehat{\boldsymbol{p}})$ の許容解でもあるため、 $h(\widehat{\boldsymbol{p}}, \widehat{\boldsymbol{p}}) \geq h(\widehat{\boldsymbol{p}}, \boldsymbol{x})$ が成り立つ。値 $(\widehat{p}_{i-1}^{\mathrm{c}} - p_{i-1}^{\mathrm{c}})$ は定義より非負である。 $\mathrm{MLE'}(\boldsymbol{p})$ の任意の許容解 $\boldsymbol{x}$ において、 $x_i \geq x_{i+1}$ より $\log x_i - \log(1 - x_i) \geq \log x_{i+1} - \log(1 - x_{i+1})$ である。これより、

$$\begin{aligned} 0 \quad \leq \quad & \sum_{i=1}^{n-1} (\widehat{p}_i^{\mathrm{c}} - p_i^{\mathrm{c}})((\log x_i - \log(1 - x_i)) - (\log x_{i+1} - \log(1 - x_{i+1}))) \\ + \quad & (\widehat{p}_n^{\mathrm{c}} - p_n^{\mathrm{c}})(\log x_n - \log(1 - x_n)) = \sum_{i=1}^{n} (\widehat{p}_i - p_i)(\log x_i - \log(1 - x_i)) \end{aligned}$$

が成り立つ。すなわち、 $h(\widehat{\boldsymbol{p}}, \boldsymbol{x}) \geq h(\boldsymbol{p}, \boldsymbol{x})$ が示された。

以上より、 $(1/p^*)\widehat{\boldsymbol{p}}$ は $\mathrm{MLE'}(\boldsymbol{p})$ の最適解となる。 □

7　おわりに

本稿では、重みつき投票ゲームの投票力指数の計算についての話題をまとめた。投票力指数の計算は県議会、国会、株主総会等様々な状況において必要とされているが、効率の良い算法が無いために大きな議決機関についてはその計算は困難である。このため、より効率の良い解法、または近似算法の開発が望まれている。また現実問題の要請としては、票数の大きな投票者数人と、票数の非常に小さな投票者多数からなる重みつき多数決ゲームについて、指数を高速に計算する算法の開発が求められている。衆参両議院のような代議員制度に見られる、２段階投票システムに対する投票力指数の計算についても、まだ研究の余地は十分に残されていると感じられる [12,20,19]。投票力指数は、未来の様々なシナリオについて計算することも多く、良く似通った票数からなるゲームを繰り返し解く状況での、効率的な算法の開発も今後の課題として残されている。投票力指数の計算は、最適化問題ではなく数え挙げ問題の構造を持っており、最適化問題に対する従来の発見的解法や近似解法のアイデアを利用することが困難であり、効率的な算法の開発には新たな知見が必要であろう。

References

[1] J.F.Banzhaf III, *Weighted voting doesn't work*, Rutgers Law Review, 19 (1965), 317-343.

[2] R.E.Barlow, D.J.Bartholomew, J.M.Bremner and H.D.Brunk, *Statistical Inference under Order Restrictions*, John Wiley and Sons, 1972.

[3] R.E.Bellman, *Notes on the theory of dynamic programming IV – maximization over discrete sets*, Naval Research Logistics Quarterly, 3 (1956), 67-70.

[4] S.J.Brams and P.J.Affuso, *Power and size; a new paradox*, mimeographed paper, 1975.

[5] J.Deegan and E.W.Packel, *A new index of power for simple n-person games*, International Journal of Game Theory, 7 (1978), 113-123.

[6] P.Dubey and L.S.Shapley, *Mathematical properties of the Banzhaf power index*, Mathematics of Operations Research, 4 (1979), 99-131.

[7] M.R.Garey and D.S.Johnson, *Computers and Intractability: a guide to the theory of NP-completeness*, W.H.Freeman and Company, 1979.

[8] R.L.Graham, *An efficient algorithm for determining the convex hull of a finite planar set*, Information Processing Letters, 1 (1972),132-133.

[9] R.M.Karp, *Reducibility among combinatorial problems*, in R.E.Miller and J.W.Thatcher eds., Complexity of Computer Computations, Plenum Press, New York, 1972, 85-103.

[10] W.F.Lucas, *Measuring power in weighted voting systems*, in S.J.Brams, W.F.Lucas and P.D.Straffin eds., *Political and related models*, Springer-Verlag, 1983, 183-238.

[11] Y.Matsui and T.Matsui, *NP-completeness for calculating power indices of weighted majority games*, *METR98-01*, Technical Report, Department of Mathematical Engineering and Information Physics, Faculty of Engineering, University of Tokyo, 1998.

[12] G.Owen, *Game Theory, 3rd ed.*, Academic Press, 1995.

[13] X.Deng and C.H.Papadimitriou, *On the complexity of cooperative solution concepts*, Mathematics of Operations Research, 19 (1994), 257-266.

[14] F.P.Preparata and M.I.Shamos, *Computational Geometry*, Springer-Verlag, 1985; 浅野孝夫, 浅野哲夫 訳, 「計算幾何学入門」, 総研出版, 1992.

[15] L.S.Shapley, *A value for n-person games*, in *Contributions to the Theory of Games, Vol.II*, in H.W.Kuhn and A.W.Tucker eds., Annals of Mathematics Studies, 28, Princeton University Press, 1953, 307-317.

[16] L.S.Shapley and M.Shubik, *A method for evaluating the distribution of power in a committee system*, American Political Science Review, 48 (1954), 787-792.

[17] 岡田章, 「ゲーム理論」, 有斐閣, 1996.

[18] 今野浩, 鈴木久敏 編, 「整数計画法と組合せ最適化」, 日科技連, 1982.

[19] 鈴木光男, 武藤滋夫, 「協力ゲームの理論」, 東京大学出版, 1985.

[20] 武藤滋夫, 小野理恵, 「投票システムのゲーム分析」, 日科技連, 1998.