論文 / 著書情報
Article / Book Information

Title	Monte Carlo Methods for Calculating Shapley-Shubik Power Index in Weighted Majority Games
Authors	Yuto Ushioda, Tomomi Matsui
Citation	Proceedings of WAAC2018
Pub. date	2018, 8

Monte Carlo Methods for Calculating Shapley-Shubik Power Index in Weighted Majority Games

Yuto Ushioda and Tomomi Matsui

Graduate School of Engineering, Tokyo Institute of Technology

Abstract

This paper addresses Monte Carlo algorithms for calculating the Shapley-Shubik power index in weighted majority games. First, we analyze a naive Monte Carlo algorithm and discuss the required number of samples. We then propose an efficient Monte Carlo algorithm and show that our algorithm reduces the required number of samples as compared to the naive algorithm.

1 Introduction

The analysis of power is a central issue in political science. In general, it is difficult to define the idea of power even in restricted classes of the voting rules commonly considered by political scientists. The use of game theory to study the distribution of power in voting systems can be traced back to the invention of "simple games" by John von Neumann and Oskar Morgenstern in their 1944 classic book titled *Theory of Games and Economic Behavior* [28]. A simple game is an abstraction of the constitutional political machinery for voting.

In 1954, Shapley and Shubik [25] proposed the specialization of the Shapley value [24] to assess the a priori measure of power of each player in a simple game. Since then, the Shapley-Shubik power index (S-S index) has become widely known as a mathematical tools for measuring the relative power of the players in a simple game.

In this paper, we consider a special class of simple games, called *weighted majority games*, which constitute a familiar example of voting systems. Let N be a set of players. Each player $i \in N$ has a positive integer voting weight w_i as the number of votes or weight of the player. The quota needed for a coalition to win is a positive integer q. A coalition $N' \subseteq N$ is a *winning coalition*, if $\sum_{i \in N'} w_i \geq q$ holds; otherwise, it is a *losing coalition*.

The difficulty involved in calculating the S-S index in weighted majority games is described in a book [11] by Garey and Johnson without proof (see p. 280, problem [MS8]). Deng and Papadimitriou [8] showed the problem of computing the S-S index in weighted majority games to be #P-complete. Prasad and Kelly [22] proved the NP-completeness of the problem of verifying the positivity of a given player's S-S index in weighted majority games. The problem of verifying the asymmetricity of a given pair of players was also shown to be NP-complete [19]. It is known that even approximating the S-S index within a constant factor is intractable unless P = NP [9].

There are variations of methods for calculating the S-S index. These include algorithms based on the Monte Carlo method [16, 18, 10, 6, 1, 7], multilinear extensions [20, 14], dynamic programming [5, 15, 17, 18, 26], generating functions [3], binary decision diagrams [4], the Karnaugh map [23], relation algebra [2], or the enumeration technique [13]. A survey of algorithms for calculating power indices in weighted voting games is presented in [18].

This paper addresses Monte Carlo algorithms for calculating the S-S index in weighted majority games. In the following section, we describe the notations and definitions used in this paper. In Section 3, we analyze a naive Monte Carlo algorithm (Algorithm A1) and extend some results obtained in the study reported in [1]. In Section 4, we propose an efficient Monte Carlo algorithm (Algorithm A2) and show that our algorithm reduces the required number of samples as compared to the naive algorithm. Table 1 summarizes the results of this study, where $(\varphi_1, \varphi_2, \dots, \varphi_n)$ denotes the S-S index and $(\varphi_1^{\mathrm{A}}, \varphi_2^{\mathrm{A}}, \dots, \varphi_n^{\mathrm{A}})$ denotes the estimator obtained by Algorithm A1 or A2.

Table 1: Required Number of Samples.

Property	Required number of samples	
	Algorithm A1 (naive algorithm)	Algorithm A2 (our algorithm)
$\Pr\left[\left\lvert\varphi_i^{\mathrm{A}} - \varphi_i\right\rvert < \varepsilon\right] \geq 1-\delta$	$\dfrac{\ln 2 + \ln(1/\delta)}{2\varepsilon^2}$ (shown by Bachrach et al. [1])	$\dfrac{\ln 2 + \ln(1/\delta)}{2\varepsilon^2}\left(\dfrac{1}{i^2}\right)$ (under Assumption 1)
$\Pr\left[\forall i \in N, \left\lvert\varphi_i^{\mathrm{A}} - \varphi_i\right\rvert < \varepsilon\right] \geq 1-\delta$	$\dfrac{\ln 2 + \ln(1/\delta) + \ln n}{2\varepsilon^2}$	$\dfrac{\ln 2 + \ln(1/\delta) + \ln 1.129}{2\varepsilon^2}$
$\Pr\left[\dfrac{1}{2}\sum_{i \in N}\left\lvert\varphi_i^{\mathrm{A}} - \varphi_i\right\rvert < \varepsilon\right] \geq 1-\delta.$	$\dfrac{n\ln 2 + \ln(1/\delta)}{2\varepsilon^2}$	$\dfrac{n''\ln 2 + \ln(1/\delta)}{2\varepsilon^2}$

An integer n'' denotes the size of a maximal player subset with mutually different weights.

2 Notations and Definitions

In this paper, we consider a special class of cooperative games called *weighted majority games*. Let $N = \{1, 2, \dots, n\}$ be a set of *players*. A subset of players is called a *coalition*. A weighted majority game G is defined by a sequence of positive integers $G = [q; w_1, w_2, \dots, w_n]$, where we may think of w_i as the number of votes or the weight of player i and q as the quota needed for a coalition to win. In this paper, we assume that $0 < q \leq w_1 + w_2 + \cdots + w_n$.

A coalition $S \subseteq N$ is called a *winning coalition* when the inequality $q \leq \sum_{i \in S} w_i$ holds. The inequality $q \leq w_1 + w_2 + \cdots + w_n$ implies that N is a winning coalition. A coalition S is called a *losing coalition* if S is not winning. We define that an empty set is a losing coalition.

Let $\pi : \{1, 2, \dots, n\} \to N$ be a permutation defined on the set of players N, which provides a sequence of players $(\pi(1), \pi(2), \dots, \pi(n))$. We denote the set of all the permutations by Π_N. We say that the player $\pi(i) \in N$ is the *pivot* of the permutation $\pi \in \Pi_N$, if $\{\pi(1), \pi(2), \dots, \pi(i-1)\}$ is a losing coalition and $\{\pi(1), \pi(2), \dots, \pi(i-1), \pi(i)\}$ is a winning coalition. For any permutation $\pi \in \Pi_N$, $\mathrm{piv}(\pi) \in N$ denotes the pivot of π. For each player $i \in N$, we define $\Pi_i = \{\pi \in \Pi_N \mid \mathrm{piv}(\pi) = i\}$. Obviously, $\{\Pi_1, \Pi_2, \dots, \Pi_n\}$ becomes a partition of Π_N. The S-S index of player i, denoted by φ_i, is defined by $|\Pi_i|/n!$. Clearly, we have that $0 \leq \varphi_i \leq 1$ $(\forall i \in N)$ and $\sum_{i \in N} \varphi_i = 1$.

Throughout this paper, we assume the following property.

Assumption 1 *The set of players is arranged to satisfy* $w_1 \geq w_2 \geq \cdots \geq w_n$.

Clearly, this assumption implies that $\varphi_1 \geq \varphi_2 \geq \cdots \geq \varphi_n$.

3 Naive Algorithm and its Analysis

In this section, we describe a naive Monte Carlo algorithm and analyze its theoretical performance.

Algorithm A1

Step 0: Set $m := 1$, $\varphi'_i := 0$ $(\forall i \in N)$.

Step 1: Choose $\pi \in \Pi_N$ uniformly at random.
Put the random variable $I^{(m)} := \mathrm{piv}(\pi)$.
Update $\varphi'_{I^{(m)}} := \varphi'_{I^{(m)}} + 1$.

Step 2: If $m = M$, then output φ'_i/M $(\forall i \in N)$ and stop.
Else, update $m := m + 1$ and go to Step 1.

For each permutation $\pi \in \Pi_N$, we can find the pivot $\mathrm{piv}(\pi) \in N$ in $\mathrm{O}(n)$ time. Thus, the time complexity of Algorithm A1 is bounded by $\mathrm{O}(M(\tau(n) + n))$ where $\tau(n)$ denotes the computational effort required for random generation of a permutation.

We denote the vector (of random variables) obtained by Algorithm A1 by $(\varphi_1^{\mathrm{A1}}, \varphi_2^{\mathrm{A1}}, \ldots, \varphi_n^{\mathrm{A1}})$. The following theorem is obvious.

Theorem 1 *For each player $i \in N$, $\mathrm{E}\left[\varphi_i^{\mathrm{A1}}\right] = \varphi_i$.*

The following theorem provides the number of samples required in Algorithm A1.

Theorem 2 *For any $\varepsilon > 0$ and $0 < \delta < 1$, we have the following.*

(1) [1] *If we set $M \geq \dfrac{\ln 2 + \ln(1/\delta)}{2\varepsilon^2}$, then each player $i \in N$ satisfies that*

$$\Pr\left[\left|\varphi_i^{\mathrm{A1}} - \varphi_i\right| < \varepsilon\right] \geq 1 - \delta.$$

(2) *If we set $M \geq \dfrac{\ln 2 + \ln(1/\delta) + \ln n}{2\varepsilon^2}$, then*

$$\Pr\left[\forall i \in N, \left|\varphi_i^{\mathrm{A1}} - \varphi_i\right| < \varepsilon\right] \geq 1 - \delta.$$

(3) *If we set $M \geq \dfrac{n \ln 2 + \ln(1/\delta)}{2\varepsilon^2}$, then*

$$\Pr\left[\frac{1}{2}\sum_{i \in N}\left|\varphi_i^{\mathrm{A1}} - \varphi_i\right| < \varepsilon\right] \geq 1 - \delta.$$

The distance measure $\frac{1}{2}\sum_{i \in N}\left|\varphi_i^{\mathrm{A1}} - \varphi_i\right|$ appearing in (3) is called the *total variation distance.*

4 Efficient Algorithm

In this section, we propose a new algorithm based on the hierarchical structure of the partition $\{\Pi_1, \Pi_2, \ldots, \Pi_n\}$. First, we introduce a map $f_i : \Pi_i \to \Pi_N$ for each $i \in N \setminus \{1\}$. For any $\pi \in \Pi_i$, $f_i(\pi)$ denotes a permutation obtained by swapping the positions of players i and $i-1$ in the permutation $(\pi(1), \pi(2), \ldots, \pi(n))$. Because $w_{i-1} \geq w_i$ (Assumption 1), it is easy to show that the pivot of $f_i(\pi)$ becomes the player $i-1$. The definition of f_i directly implies that $\forall \{\pi, \pi'\} \subseteq \Pi_i$, if $\pi \neq \pi'$, then $f_i(\pi) \neq f_i(\pi')$. Thus, we have the following.

Lemma 1 *For any $i \in N \setminus \{1\}$, the map $f_i : \Pi_i \to \Pi_{i-1}$ is injective.*

When an ordered pair of permutations (π, π') satisfies the conditions that $\pi \in \Pi_i$, $\pi' \in \Pi_j$, $i \leq j$, and $\pi = f_{i-1} \circ \cdots \circ f_{j-1} \circ f_j(\pi')$, we say that π' is an *ancestor* of π. Here, we note that π is always an ancestor of π itself. Lemma 1 implies that every permutation $\pi \in \Pi_N$ has a unique ancestor, called the *originator*, $\pi' \in \Pi_j$ satisfying that either $j = n$ or its inverse image $f_{j+1}^{-1}(\pi') = \emptyset$. For each permutation $\pi \in \Pi_N$, $\mathrm{org}(\pi) \in N$ denotes the pivot of the originator of π; i.e., $\Pi_{\mathrm{org}(\pi)}$ includes the originator of π.

Now, we describe our algorithm.

Algorithm A2

Step 0: Set $m := 1$, $\varphi'_i := 0$ $(\forall i \in N)$.

Step 1: Choose $\pi \in \Pi_N$ uniformly at random.
Put the random variable $L^{(m)} := \mathrm{org}(\pi)$.
Update $\varphi'_i := \begin{cases} \varphi'_i + 1/L^{(m)} & (\text{if } 1 \leq i \leq L^{(m)}), \\ \varphi'_i & (\text{if } L^{(m)} < i). \end{cases}$

Step 2: If $m = M$, then output φ'_i/M $(\forall i \in N)$ and stop.
Else, update $m := m + 1$ and go to Step 1.

For each permutation $\pi \in \Pi_N$, we can find the originator $\mathrm{org}(\pi) \in N$ in $\mathrm{O}(n)$ time. Thus, the time complexity of Algorithm A2 is also bounded by $\mathrm{O}(M(\tau(n)+n))$ where $\tau(n)$ denotes the computational effort required for random generation of a permutation.

We denote the vector (of random variables) obtained by Algorithm A2 by $(\varphi_1^{\mathrm{A2}}, \varphi_2^{\mathrm{A2}}, \ldots, \varphi_n^{\mathrm{A2}})$. The following theorem is obvious.

Theorem 3 (1) *For each player $i \in N$, $\mathrm{E}\left[\varphi_i^{\mathrm{A2}}\right] = \varphi_i$.*
(2) *For each pair of players $\{i, j\} \subseteq N$, if $\varphi_i > \varphi_j$, then $\varphi_i^{\mathrm{A2}} \geq \varphi_j^{\mathrm{A2}}$,*
(3) *For each pair of players $\{i, j\} \subseteq N$, if $\varphi_i = \varphi_j$, then $\varphi_i^{\mathrm{A2}} = \varphi_j^{\mathrm{A2}}$.*

The following theorem provides the number of samples required in Algorithm A2.

Theorem 4 *For any $\varepsilon > 0$ and $0 < \delta < 1$, we have the following.*
(1) *For each player $i \in N = \{1, 2, \ldots, n\}$, if we set $M \geq \dfrac{\ln 2 + \ln(1/\delta)}{2\varepsilon^2 i^2}$, then*

$$\Pr\left[\left|\varphi_i^{\mathrm{A2}} - \varphi_i\right| < \varepsilon\right] \geq 1 - \delta.$$

(2) *If we set $M \geq \dfrac{\ln 2 + \ln(1/\delta)}{2\varepsilon^2}$, then*

$$\Pr\left[\forall i \in N, \left|\varphi_i^{\mathrm{A2}} - \varphi_i\right| < \varepsilon\right] \geq 1 - 2\sum_{i=1}^{n}\left(\frac{\delta}{2}\right)^{i^2} = 1 - 2\left(\left(\frac{\delta}{2}\right) + \left(\frac{\delta}{2}\right)^4 + \left(\frac{\delta}{2}\right)^9 + \cdots + \left(\frac{\delta}{2}\right)^{n^2}\right).$$

(3) *If we set* $M \geq \dfrac{|N^*| \ln 2 + \ln(1/\delta)}{2\varepsilon^2}$, *then*

$$\Pr\left[\frac{1}{2}\sum_{i \in N}\left|\varphi_i^{\mathrm{A2}} - \varphi_i\right| < \varepsilon\right] \geq 1 - \delta,$$

where $N^* = \{i \in N \setminus \{n\} \mid \varphi_i > \varphi_{i+1}\} \cup \{n\}$, *i.e.,* $|N^*|$ *is equal to the size of the maximal player subset, the S-S indices of which are mutually different.*

The following corollary provides an approximate version of Theorem 4 (2). Surprisingly, it says that the required number of samples is irrelevant to n (number of players).

Corollary 1 *For any* $\varepsilon > 0$ *and* $0 < \delta' < 1$, *we have the following. If we set* $M \geq \dfrac{\ln 2 + \ln(1/\delta') + \ln 1.129}{2\varepsilon^2}$, *then*

$$\Pr\left[\forall i \in N, \left|\varphi_i^{\mathrm{A2}} - \varphi_i\right| < \varepsilon\right] \geq 1 - \delta'.$$

Here, we note that $\ln 2 \simeq 0.69314$ and $\ln 1.129 \simeq 0.12133$.

In a practical setting, it is difficult to estimate the size of N^* defined in Theorem 4 (3), since the problem of verifying the asymmetricity of a given pair of players is NP-complete [19]. The following corollary is useful in some practical situations.

Corollary 2 *For any* $\varepsilon > 0$ *and* $0 < \delta < 1$, *we have the following. If we set* $M \geq \dfrac{n'' \ln 2 + \ln(1/\delta)}{2\varepsilon^2}$, *then*

$$\Pr\left[\frac{1}{2}\sum_{i \in N}\left|\varphi_i^{\mathrm{A2}} - \varphi_i\right| < \varepsilon\right] \geq 1 - \delta,$$

where $n'' = |\{i \in N \setminus \{n\} \mid w_i > w_{i+1}\} \cup \{n\}|$, *i.e.,* n'' *is equal to the size of a maximal player subset with mutually different weights.*

The game of the power of the countries in the EU Council is defined by $G = [q; 10, 10, 10, 10, 8, 5, 5, 5, 5, 4, 4, 3, 3, 3, 2]$, where $q = 62$ or $q = 65$ [3]. In this case, $n = 15$ and $n'' = 6$. A weighted majority game defined by Owen [21] for a voting process in United States has a vector of weights

$$[270; 45, 41, 27, 26, 26, 25, 21, 17, 17, 14, 13, 13, 12, 12, 12, 11, \underbrace{10, \ldots, 10}_{4\text{ times}}, \underbrace{9, \ldots, 9}_{4\text{ times}}, 8, 8, \underbrace{7, \ldots, 7}_{4\text{ times}}, \underbrace{6, \ldots, 6}_{4\text{ times}}, 5, \underbrace{4, \ldots, 4}_{9\text{ times}}, \underbrace{3, \ldots, 3}_{7\text{ times}}],$$ where $n = 51$ and $n'' = 19$.

5 Conclusion

In this paper, we analyzed a naive Monte Carlo algorithm (Algorithm A1) for calculating the S-S index denoted by $(\varphi_1, \varphi_2, \ldots, \varphi_n)$ in weighted majority games. By employing the Bretagnolle-Huber-Carol inequality [27]. we estimated the required number of samples that gives an upper bound of the total variation distance.

We also proposed an efficient Monte Carlo algorithm (Algorithm A2). The time complexity of our algorithm is equal to that of the naive algorithm (Algorithm A1). Our algorithm has the property that the obtained estimator $(\varphi_1^{\mathrm{A2}}, \varphi_2^{\mathrm{A2}}, \ldots, \varphi_n^{\mathrm{A2}})$ satisfies

$$\text{both [if } \varphi_i < \varphi_j \text{ then } \varphi_i^{\mathrm{A2}} \leq \varphi_j^{\mathrm{A2}}\text{] and [if } \varphi_i = \varphi_j \text{ then } \varphi_i^{\mathrm{A2}} = \varphi_j^{\mathrm{A2}}\text{].}$$

We also proved that, even if we consider the property $\Pr\left[\forall i \in N, \left|\varphi_i^{\mathrm{A2}} - \varphi_i\right| < \varepsilon\right] \geq 1 - \delta$, the required number of samples is irrelevant to n (the number of players).

References

[1] Bachrach, Y., Markakis, E., Resnick, E., Procaccia, A. D., Rosenschein, J. S., and Saberi, A. Approximating power indices: theoretical and empirical analysis. *Autonomous Agents and Multi-Agent Systems*, **20** (2010), 105–122.

[2] Berghammer, R., Bolus, S., Rusinowska, A., and De Swart, H. A relation-algebraic approach to simple games. *European Journal of Operational Research*, **210** (2011), 68–80.

[3] Bilbao, J. M., Fernandez, J. R., Losada, A. J., and Lopez, J. J. Generating functions for computing power indices efficiently. *Top*, **8** (2000), 191–213.

[4] Bolus, S. Power indices of simple games and vector-weighted majority games by means of binary decision diagrams. *European Journal of Operational Research*, **210** (2011), 258–272.

[5] Brams, S. J. and P. J. Affuso, P. J. Power and size; a new paradox. *Mimeographed Paper*, 1975.

[6] Castro, J., Gómez, D., and Tejada, J. Polynomial calculation of the Shapley value based on sampling. *Computers & Operations Research*, **36** (2009), 1726–1730.

[7] Castro, J., Gómez, D., Molina, E., and Tejada, J. Improving polynomial estimation of the Shapley value by stratified random sampling with optimum allocation. *Computers & Operations Research*, **82** (2017), 180–188.

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

[9] Elkind, E., Goldberg, L. A., Goldberg, P., and Wooldridge, M. Computational complexity of weighted threshold games. In *Proc. of the National Conference on Artificial Intelligence*, AAAI Press, 718–723, 2007.

[10] Fatima, S. S., Wooldridge, M., and Jennings, N. R. A linear approximation method for the Shapley value *Artificial Intelligence*, **172** (2008), 1673–1699.

[11] Garey, M. R. and Johnson, D. S. Computers and Intractability: A Guide to the Theory of NP-Completeness. WH Freeman, 1979.

[12] Hoeffding, W. Probability inequalities for sums of bounded random variables, *Journal of the American Statistical Association*, **58** (1963), 13–30.

[13] Klinz, B. and Woeginger, G. J. Faster algorithms for computing power indices in weighted voting games. *Mathematical Social Sciences*, **49** (2005), 111–116.

[14] Leech, D. Computing power indices for large voting games. *Management Science*, **49** (2003), 831–837.

[15] Lucas, W. F. Measuring power in weighted voting systems. Brams, S. J., Lucas, W. F., and Straffin, P. D. (eds.): Political and Related Models, Springer, 183–238, 1983.

[16] Mann, I. and Shapley, L. S. Values of large games. IV: evaluating the electoral college by Montecarlo techniques. *Technical Report, The RAND Corporation*, **RM-2651**, 1960.

[17] Mann, I. and Shapley, L. S. Values of large games. VI: Evaluating the electoral college exactly. *Technical Report, The RAND Corporation*, **RM-3158-PR**, 1962.

[18] Matsui, T. and Matsui, Y. A survey of algorithms for calculating power indices of weighted majority games. *Journal of the Operations Research Society of Japan*, **43** (2000), 71–86.

[19] Matsui, Y. and Matsui, T. NP-completeness for calculating power indices of weighted majority games. *Theoretical Computer Science*, **263** (2001), 305–310.

[20] Owen, G. Multilinear extensions of games. *Management Science*, **18** (1972), 64–79.

[21] G. Owen, G. Game Theory. Academic press, 1995.

[22] Prasad, K. and Kelly, J. S. NP-completeness of some problems concerning voting games. *International Journal of Game Theory*, **19** (1990), 1–9.

[23] Rushdi, A. M. A. and Ba-Rukab, O. M. Map calculation of the Shapley-Shubik voting powers: An example of the European Economic Community. *International Journal of Mathematical, Engineering and Management Sciences (IJMEMS)*, **2** (2017), 17–29.

[24] Shapley, L. S. A value for n-person games. Kuhn, H. W. and Tucker, A. W. Tucker (eds.), Contributions to the Theory of Games II, Princeton University Press, 307–317, 1953.

[25] Shapley, L. S. and Shubik, M. An algorithm for evaluating the distribution of power in a committee system. *American Political Science Review*, **48** (1954), 787–792.

[26] Uno, T. Efficient computation of power indices for weighted majority games. In *Proc. of International Symposium on Algorithms and Computation (ISAAC)*, LNCS **7676**, Springer, 679–689, 2012.

[27] van der Vaart, A. W. and Wellner, J. A. Weak Convergence and Empirical Processes: with Applications to Statistics, Springer, 1989.

[28] von Neumann, J. and Morgenstern, O. Theory of Games and Economic Behavior. Princeton Univ. Press, 1944.