

A heuristic approximation method for the Banzhaf index for voting games

Shaheen Fatima^{a,*}, Michael Wooldridge^b and Nicholas R. Jennings^c

^a*Department of Computer Science, Loughborough University, Loughborough, UK*

^b*Department of Computer Science, University of Oxford, Oxford, UK*

^c*School of Electronics and Computer Science, University of Southampton, Southampton, UK*

Abstract. The Banzhaf index is a well known and widely used index for measuring the power a player has in a voting game. However, the problem of computing this index is computationally hard. To overcome this problem, a number of approximation methods were developed for one majority voting games. While it may be possible to extend some of these to k -majority games (which are generalized versions of one majority games), to date, there has been no performance analysis of these methods in the context of the Banzhaf index for k -majority games. In this paper, we fill this gap, by first presenting an approximation method for the Banzhaf index for k -majority games. This is a heuristic method that uses randomization to estimate an approximate. We then show that this method is computationally feasible. Finally, we evaluate its performance by analyzing its error of approximation, and show how the error varies with k . Specifically, we show that the average percentage error increases from 15% for games with $k = 1$ to 30% for games with $k = 5$.

Keywords: Cooperative game theory, Banzhaf index, complexity, heuristic algorithms

1. Introduction

Coalition formation is a key form of interaction in multi-agent systems. It is the process of joining together two or more agents so as to achieve goals that individuals on their own cannot, or to achieve them more efficiently [25]. Often, in such situations, there is more than one possible way in which agents can form coalitions. Given this, the agents/players must decide upon the following:

D1: who to form a coalition with, and

D2: how to split the gains of cooperation between the members of a coalition.

In more detail, there will generally be many possible ways in which the agents can form coalitions. Moreover, which goals a coalition can achieve and how efficiently it can achieve them depends on the members that make up the coalition. Also, different agents may have different preferences for different goals. So each agent will want to join a coalition that is most beneficial to it. Given this, D1 is a crucial decision that an agent must make. Once a coalition is formed, the agents within it must decide upon D2, i.e., how they will split the joint gains between themselves.

In this context, cooperative game theory offers a number of solutions for the problems D1 and D2. Specifically, for D1, there are solutions such as *core* and *kernel* [25]. These solutions ensure that coalitions are stable, i.e., members (neither individuals nor subgroups) of a coalition do not have any incentive

*Corresponding author: Shaheen Fatima, Department of Computer Science, Loughborough University, Loughborough LE11 3TU, UK. E-mail: s.s.fatima@lboro.ac.uk.

to break away from the coalition to join another one. For the problem D2, there are solutions such as Shapley value [34,35] and Banzhaf index [3].

In order to use these solutions, a multi-agent interaction must first be represented as a *coalitional game*. A number of multi-agent systems researchers have used these games to model coalition formation. Examples for this include [7,12,32,33,36,37]. Once an interaction is represented as a coalitional game, it then becomes possible to apply game-theoretic solution concepts to them. However, in the context of multi-agent systems, the application of these solution concepts is not always straightforward. This is because it is computationally hard to find the game-theoretic solutions [9,30]. Therefore, a key issue, in the context of multi-agent systems, is to address the computational aspects of game-theoretic solutions.

In more detail, there are different forms of coalitional games [25]. A particular form of succinctly representable ones are *weighted voting games*. Although these games can be represented succinctly, the solutions to these games are hard to compute. Before we look at these solutions, let us first see what weighted voting games are. A weighted voting game is one in which each player is given a numeric weight, and a coalition takes the value 1 (i.e., is a *winning coalition*) if the sum of its weights exceeds a particular quota, and takes the value 0 (i.e., is a *losing coalition*) otherwise.

Weighted voting games are widely used in practice. For example, the voting system in the UK is a weighted voting game, and so is the voting system in the European Union [4]. In such games, the degree to which a player can influence the outcome of voting depends on the player's weights. This influence, also known as a player's voting power, is measured in terms of a *power index*. Although a player's power index is, in general, defined in terms of the weights, there are different ways of defining this power [15]. Amongst these, the most widely used ones include the *Shapley value* [34,35] and the *Banzhaf index* [3]. These two indices differ in terms of the way in which the index is computed: the Shapley value considers the 'order' in which the players join a coalition while the Banzhaf index does not. However, from a computational perspective, they are similar: for a given voting game, the problem of computing both is a #P-complete [9,30] problem. Roughly speaking, #P-complete problems are a class of problems that, although nobody has been able to prove it, seem *intractable*¹. Thus, it is practically infeasible to try to compute the Shapley value/Banzhaf index for these games.

Hence, in order to overcome this computational complexity a number of approximation methods have been proposed (see Section 6 for details). These include the Monte Carlo simulation method [23], the multi-linear extension (MLE) method [26,27], the modified MLE method [21], the random permutation method [40], and the randomized approximation methods [2,13]. These methods are mostly for one majority voting games.

While it may be possible to extend some of these to k -majority games (which are generalized versions of one majority games), to date, there has been no performance analysis of these methods in the context of the Banzhaf index for k -majority games. In this paper, we fill this gap by first presenting an approximation method for the Banzhaf index for k -majority games. This is a heuristic method that uses randomization to estimate an approximate. We then show that, for a game with n players, the time complexity of this method is $\mathcal{O}(kn^22^k)$. Finally, we evaluate its performance by analyzing its error of approximation, and show how the error varies with k . Specifically, we show that the average percentage error increases from 15% for games with $k = 1$ to 30% for games with $k = 5$.

The results of this research are significant not only in the context of coalition formation, but are also equally important in other contexts such as multi-party negotiations. To date, research in coalition

¹A problem is said to be intractable if it takes unreasonably long to solve it. This is the case, for example, if the time taken to solve it increases exponentially in the size of the input to the problem.

formation and multi-party negotiation has largely progressed independently of each other. However, there is a strong link between these two lines of research [29]: in the context of multi-party negotiation, the decision making problem D1 corresponds to the search for a negotiation partner and D2 corresponds to the development of a negotiated agreement. Also, [17] shows that the notion of a player's *power index*² for a coalitional game is closely related to the notion of the player's *utility* in a multi-party negotiation involving the N players. Specifically, the former is directly proportional to the latter. In other words, the higher a player's Shapley value/Banzhaf index, the higher its utility in the multi-party negotiation. Thus, by computing the Banzhaf index for a coalitional game we get an indication of the player's utilities from negotiation.

There are a number of other applications of these power indices. These include analysis of voting structures of several bodies such as European Union Council of Ministers, and the International Monetary Fund [20,22]. Another application of power indices is cost sharing schemes [39]. Apart from this, voting games find application in number of multi-agent settings. For example, consider a computational grid environment in which a set of limited resources must be allocated to a subset of agents that compete for them. A key decision in such a setting is to decide what set of resources to make available on the grid. Assume we are considering whether or not to introduce a new resource R. This decision depends on the agents' requirements for R: some agents may benefit from its introduction while others may not. In such a scenario, different agents have different preferences for R and they must find a way of resolving these differences. In such a context, a weighted voting game allows agents to resolve these differences and reach an agreement on whether or not to introduce R. In a situation like this, a player would want to know how much power it has in influencing the outcome of voting and would therefore need to compute its Banzhaf index.

Yet another domain where the Banzhaf index could be used is in resource allocation in multi-agent systems. In such scenarios, there are limited resources available and many agents contending for them. In this context, the Banzhaf index can be used as an indication of the individual agent's shares of the available resources.

Although the Banzhaf index has the potential to be used across this wide spectrum of application domains, in reality, its use is not straightforward because of the complexity involved in computing it. By presenting a computationally feasible approximation method, this paper adds a practical dimension to the Banzhaf index and facilitates its use in multi-agent encounters.

The rest of the paper is organised as follows. Section 2 provides the background to voting games and power indices. In Section 3 we present our approximation method. Section 4 (Section 5) provides an analysis of the *standard error (relative error)* of approximation. Section 6 discusses related literature and Section 7 concludes.

2. Background

We define the following terms: coalitional game, weighted voting game (the two forms of weighted voting games i.e., one-majority and k -majority games), and the Banzhaf index.

²For a coalitional game, a player's 'power index' is a measure of how much power it has in influencing the outcome of the game [15]. The Shapley value and the Banzhaf index are two ways of measuring this power.

2.1. Coalitional game

A coalitional game $\langle N, v \rangle$, consists of [25]:

1. a finite set, $N = \{1, 2, \dots, n\}$ of n players, and
2. a function, v , that associates with every non-empty subset S of N (i.e., a *coalition*) a non-negative real number $v(S)$ that indicates its worth. The domain for v is the set of all possible subsets of N and the range is the set of all non-negative real numbers.

For each coalition S , $v(S)$ is the total payoff that is available for division among the members of S .

2.2. Weighted one-majority voting game

For the players in N , let w_i denote the weight for player i . Also let $q \in \mathbb{R}_+$ denote a real number, and let $w(S) = \sum_{i \in S} w_i$ for any coalition S . Here, q is the *quota* of the game. Then, a weighted one-majority voting game is a game $\langle N, v \rangle$ such that [25]:

$$v(S) = \begin{cases} 1 & \text{if } w(S) \geq q \\ 0 & \text{otherwise} \end{cases}$$

This game is denoted as $\langle q; w_1, \dots, w_n \rangle$. A coalition with value zero is called a ‘losing coalition’ and with value one a ‘winning coalition’.

A special case of such games is a *simple majority voting game*. For such games, each player has unit weight and the quota is $\lceil \frac{|N|+1}{2} \rceil$. There are a number of real-world applications of the simple majority voting game such as the electoral process used in the UK and many other countries.

2.3. Weighted k -majority voting game

These are generalized versions of one-majority games and are defined as follows [4]. For the set of n players, a weighted k -majority game (denoted³ $v_1 \wedge \dots \wedge v_k$) is a game where the t th ($1 \leq t \leq k$) component $v_t = \langle q^t; w_1^t, \dots, w_n^t \rangle$ is a one-majority voting game. The value function for the k -majority game is defined as follows:

$$(v_1 \wedge \dots \wedge v_k)(S) = \begin{cases} 1 & \text{if } w^t(S) \geq q^t \text{ for } 1 \leq t \leq k \\ 0 & \text{otherwise} \end{cases}$$

where $w^t(S) = \sum_{i \in S} w_i^t$. Thus, in a k -majority game there are k different one-majority games. All these k component games have the same set of players. A coalition in a k -majority game is winning if it wins in each of the k component games.

An example of a k -majority game is the voting system of the enlarged European Union [4]. This is a three-majority voting game. The three component games are for the *countries*, the *population*, and the *commissionaries*. Here, a coalition is winning if it has the support of a majority of countries of the EU, a majority of the population of the EU, and majority of the commissionaries of the EU.

³The ‘ \wedge ’ symbol in $v_1 \wedge \dots \wedge v_k$ is not a ‘logical and’ symbol. It is just a notation for representing the components of k -majority games.

2.4. Banzhaf index

The *Banzhaf index* for a voting game is a way of measuring a player's voting power. A player's power is his ability to turn a losing coalition into a winning one. The *Banzhaf index* has two versions: the *absolute Banzhaf index* and the *normalised Banzhaf index*. These are defined as follows [3].

For a voting game $G = \langle N, v \rangle$, player i 's Banzhaf index is defined in terms of its *marginal contribution*. The *marginal contribution* of player i to coalition S with $i \notin S$ is a function $\Delta_i v$ defined as follows:

$$\Delta_i v(S) = v(S \cup \{i\}) - v(S)$$

In words, a player's marginal contribution to a coalition S is the increase in the value of S as a result of i joining it. Note that for a voting game $(\langle q; w_1, \dots, w_n \rangle)$, a player's marginal contribution is either zero or one. This is because the value of any coalition is either zero or one. If a player's entry to a coalition changes it from losing to winning, then the player's marginal contribution for that coalition is one; otherwise it is zero.

A player's Banzhaf index is defined in terms of the number of coalitions for which it is the *swing* player. The player $i \in N$ is called a *swing* player for a coalition S if S is losing but $S \cup \{i\}$ is winning. Let T_i denote the set of all possible coalitions for which $i \in N$ is a swing player, i.e., for $x \in T_i$, x is losing but $x \cup \{i\}$ is winning. Then, for player $i \in N$, let η_i denote the number of coalitions for which it is swing player, i.e.,

$$\eta_i = |T_i| \tag{1}$$

The two versions of Banzhaf index are defined by expressing η_i over different denominators.

Definition 1 For player i , the *absolute Banzhaf index* (β_i) is defined as [3]:

$$\beta_i = \eta_i / 2^{n-1} \tag{2}$$

Definition 2 For player i , the *normalized Banzhaf index* (λ_i) is defined as [3]:

$$\lambda_i = \eta_i / \sum_{i=1}^n \eta_i \tag{3}$$

Note that the normalized Banzhaf index sums to unity over the players: $\sum \lambda_i = 1$. The following example illustrates how the Banzhaf index is computed.

Example 1 Let $N = \{1, 2, 3\}$ and $w_1 = 1, w_2 = 2, w_3 = 3, q = 4$. For this example, consider a coalition of two players: 1 and 2. Depending on the order in which the two players join, the coalition can form in one of two ways: player 1 followed by player 2, or player 2 followed by player 1. However, for the Banzhaf index only the set of players in a coalition matters, not the order in which they joined the coalition. Now, there are three winning coalitions: $\{1, 3\}$, $\{2, 3\}$, and $\{1, 2, 3\}$. Players 1 and 2 are swing players in $\{1, 3\}$, players 2 and 3 are swing players in $\{2, 3\}$, and player 3 is the swing player in $\{1, 2, 3\}$. So player 1 is swing player once and so is player two, but player 3 is swing player thrice. Consequently, the player's absolute Banzhaf indices are: $\beta_1 = 1/4, \beta_2 = 1/4$, and $\beta_3 = 3/4$. And their normalized Banzhaf indices are: $\lambda_1 = 1/5, \lambda_2 = 1/5$, and $\lambda_3 = 3/5$.

Having illustrated how the exact Banzhaf index is computed, we are now ready to present our approximation methods for it. In what follows, we will show how to approximate β_i . Using this method, it is straightforward to compute λ_i . In more detail, since both β_i and λ_i are defined in terms of η_i (see Eqs (2) and (3)), we will show how to find an approximate for η_i . An approximate for both β_i and λ_i can then easily be computed in terms of an approximate η_i .

3. Approximating the Banzhaf index

We will first show how to compute an approximate Banzhaf index for one-majority games and then for k -majority games. The methods we will describe are an extension of those given in [11]. Specifically [11], gives an approximation for the Shapley value for k -majority game. Below, we will explain how these methods can be extended to find an approximation for the absolute Banzhaf index for k -majority games. In Section 3.1, we consider one-majority games and then, in Section 3.2, k -majority games.

3.1. One-majority games

We know from Section 2.4, that computing β_i requires computing i 's marginal contribution to all possible coalitions S_X of size X for $1 \leq X \leq n - 1$ and summing them. Recall from Eq. (1), that this sum is η_i . However, computing η_i by computing i 's marginal contribution to all possible coalitions S_X of size X for $1 \leq X \leq n - 1$ and summing them is computationally hard: it takes $\mathcal{O}(\sum_{X=1}^{n-1} \binom{n}{X})$ time because $\sum_{X=1}^{n-1} \binom{n}{X}$ coalitions need to be considered. Thus instead of computing η_i by considering all these coalitions, we show how to find an approximation for η_i .

In more detail, this approximation is done as follows. Let $\bar{\eta}_i$ denote an approximation for η_i and $E\Delta_i^X$ an approximation for the mean of i 's marginal contributions to all coalitions of size X . Now, we know that the total number of players is n . So there are $n - 1$ players excluding player i and therefore there are $\binom{n-1}{X}$ possible coalitions of size X that i can join. Given this, the sum of i 's marginal contributions to all coalitions of size X will be $\binom{n-1}{X} E\Delta_i^X$. In order to compute $E\Delta_i^X$, we use the following method given⁴ in [11,13] ([13] is an extension of [11]). This method is based on the following approximation rule \mathcal{R}_1 [16] which works as follows. Let the players' weights in N be defined by 'any' probability distribution function. Irrespective of the actual form of this function, let μ be the mean weight of players and ν the variance in weights. Now, from this set (N), if we randomly draw a sample (i.e., a coalition) of size X , then the approximate sum of the players' weights in the sample is given by the following rule [16]:

\mathcal{R}_1 : If a random sample of size X is drawn from any distribution with mean μ and variance ν , then the sample mean has an approximate Normal distribution, \mathcal{N} , with mean $X\mu$ and variance $\frac{\nu}{X}$ (the larger the n the better the approximation⁵).

We use the above rule to determine $\bar{\eta}_i$ as follows. For player i with weight w_i , the marginal contribution to a random sample/coalition of size X is one if the mean weight for the sample is greater than or equal to $a = (q - w_i)/X$ but less than $b = (q - \epsilon)/X$ (where ϵ is an infinitesimally small quantity). Otherwise, its marginal contribution is zero. Now, as per the rule \mathcal{R}_1 , an approximate mean probability with which i is the swing player for samples of size X is the area under the normal distribution $\mathcal{N}(X\mu, \nu/X)$ between the limits a and b . We use this probability to approximate i 's mean marginal contribution (i.e., $E\Delta_i^X$) to all coalitions of size X . Thus, we have:

$$E\Delta_i^X = \frac{1}{\sqrt{(2\pi\nu/X)}} \int_a^b e^{-X \frac{(x-X\mu)^2}{2\nu}} dx. \quad (4)$$

⁴This is because both the Shapley value and the Banzhaf index are defined in terms of 'marginal contribution', i.e., both require computing $E\Delta_i^X$.

⁵Also, for large X , any measurement done on a sample drawn with replacement is the same as that for a sample drawn without replacement [16].

Given $E\Delta_i^X$, player i 's total approximate marginal contribution to all coalitions of size X is $\binom{n-1}{X} \times E\Delta_i^X$. Having shown how to compute $E\Delta_i^X$, we will now show how to compute $\bar{\eta}_i$. To do so, we let $\bar{\eta}_i^X$ denote an approximate number of coalitions of size X for which i is the swing player. Note that $\bar{\eta}_i^X$ is nothing but i 's total approximate marginal contribution to all coalitions of size X . Thus we have:

$$\bar{\eta}_i^X = \binom{n-1}{X} \times E\Delta_i^X \quad (5)$$

Given this, and the fact that η_i is the number of coalitions for which i is swing player (see Eq. (1)), an approximate number of coalitions (of all possible sizes) for which i is swing player is:

$$\bar{\eta}_i = \sum_{X=1}^{n-1} \bar{\eta}_i^X \quad (6)$$

We will now show how to obtain i 's approximate Banzhaf index in terms of $\bar{\eta}_i$. To do so, let $\bar{\beta}_i$ denote an approximation for β_i . We obtain $\bar{\beta}_i$ by substituting Eq. (6) in Eq. (2):

$$\bar{\beta}_i = \bar{\eta}_i / 2^{n-1} \quad (7)$$

3.2. k -Majority voting game

We will now show how to extend the above method to k -majority games. To do so, we need some data structures and functions. We begin by introducing these. Let Com denote a two-dimensional array with k columns and a varying number of rows. We will specify the number of rows as we need them. Then let $combinations(\{1, \dots, k\}, i)$ denote a function that takes a vector (such as $\{1, \dots, k\}$) and a non-negative integer (such as $i \leq k$) as arguments and returns all combinations of $\{1, \dots, k\}$ of size i . Assume that all these combinations of size i are stored in the vector Com . Then Com will have $\binom{k}{i}$ rows. Each row of this vector contains a combination of i items which are stored in the first i columns of Com . And let $Com(y, \{i+1, \dots, k\})$ denote the columns $i+1$ to k of the y th row. Later, we will show what these columns from $i+1$ to k contain.

Now, if we let S_X denote a coalition of size X , then, as per Section 2.3, the marginal contribution of player i to S_X for the k -majority game $v_1 \wedge \dots \wedge v_k$ is 1 if the following conditions hold:

- C1: i is swing player for S_X in at least one of the k games, and
- C2: S_X is winning for the remaining of the k games.

As before, we find a player's approximate total marginal contribution to all coalitions of size X in terms of an approximate probability. To do so, we use the following notation. For game v_h and player i , let $PL(h, i, X)$ denote an approximate mean probability that i is the swing player in coalitions of size X . Also, for game v_h , let $PW(h, X)$ denote an approximate mean probability that a coalition of size X is winning. Then, for $1 \leq j \leq k$ and $1 \leq row \leq \binom{k}{j}$, let $Term(row, X, j, i)$ denote an approximate mean probability that:

- P1: i is swing player (for coalitions of size X) for the games $v_{Com(row,1)}, \dots, v_{Com(row,j)}$, and
- P2: coalitions of size X are winning for the games $v_{Com(row,j+1)}, \dots, v_{Com(row,k)}$.

In other words, $Term(row, X, j, i)$ is:

$$Term(row, X, j, i) = \left(\prod_{f=1}^j PL(Com(row, f), i, X) \times \prod_{g=j+1}^k PW(Com(row, g), X) \right) \quad (8)$$

where

$$Com = combinations(\{1, \dots, k\}, j) \quad \text{and} \quad (9)$$

$$Com(r, \{j + 1, \dots, k\}) = \{1, \dots, k\} - Com(r, \{j + 1, \dots, k\}).$$

Then, let $Sum(X, j, i)$ denote an approximate mean probability that i is swing player in coalitions of size X . Given $Term(row, X, j, i)$, and the conditions $C1$ and $C2$, $Sum(X, j, i)$ will be:

$$Sum(X, j, i) = \sum_{row=1}^{\binom{k}{i}} Term(row, X, j, i) \quad (10)$$

Then, let $kE\Delta_i^X$ denote i 's approximate mean marginal contribution to all coalitions of size X . Given $Sum(X, j, i)$ and the fact that $1 \leq j \leq k$, we get:

$$kE\Delta_i^X = \sum_{j=1}^k Sum(X, j, i) \quad (11)$$

We will now show how to compute $PL(h, i, X)$ and $PW(h, X)$. For the game v_h , let $w(h, i)$ denote player i 's weight, $\mu(h)$ the mean weight of the players, $\nu(h)$ the variance in their weights, and $q(h)$ the quota. From \mathcal{R}_1 , we know that $PL(h, i, X)$ is the area under the normal distribution $N(\mu(h), \nu(h))$ between the limits $a = q(h) - w(h, i)$ and $b_1 = q(h) - \epsilon$:

$$PL(h, i, X) = \frac{1}{\sqrt{(2\pi\nu(h)/X)}} \int_a^{b_1} e^{-X \frac{(x - \mu(h))^2}{2\nu(h)}} dx \quad (12)$$

and $PW(h, X)$ is the area under the normal distribution $N(\mu(h), \nu(h))$ between the limits $q(h)$ and ∞ :

$$PW(h, X) = \frac{1}{\sqrt{(2\pi\nu(h)/X)}} \int_{q(h)}^{\infty} e^{-X \frac{(x - \mu(h))^2}{2\nu(h)}} dx \quad (13)$$

If we let $k\bar{\eta}_i^X$ denote i 's approximate total marginal contribution to all coalitions of size X , then as per Eq. (1) we get:

$$k\bar{\eta}_i^X = \binom{n-1}{X} \times kE\Delta_i^X \quad (14)$$

where $kE\Delta_i^X$ is as computed in Eq. (11). Substituting Eq. (11) in Eq. (14) we get $k\bar{\eta}_i^X$. Then let $k\bar{\eta}_i$ denote i 's approximate total marginal contribution to coalitions of all possible sizes (i.e., $1 \leq X \leq n-1$). Since there are $\binom{n-1}{X}$ possible coalitions of size X and $1 \leq X \leq n-1$, we get:

$$k\bar{\eta}_i = \sum_{X=1}^{n-1} k\bar{\eta}_i^X \quad (15)$$

For player i , let $\bar{\beta}_i^k$ denote an approximate Banzhaf index. Substituting Eq. (15) in Eq. (2), we get $\bar{\beta}_i^k$:

$$\bar{\beta}_i^k = k\bar{\eta}_i/2^{n-1} \quad (16)$$

The above steps are described in Algorithm 1. The ‘for’ loop in Line 1 is for computing the Banzhaf index for each of the n players and is therefore repeated n times. For each player, the for loop in Line 3 computes an approximate marginal contribution. Each player’s marginal contribution is computed for all possible coalitions of size X where $1 \leq X \leq n - 1$. So the ‘for’ loop in Line 3 is repeated $n - 1$ times.

Algorithm 1 BanzhafIndex-KMG(N, k, n, q, μ, ν, w)

N : The set of players

n : The number of players in N

k : The number of weighted voting games

w : A matrix of the n columns and k rows with $w(x, y)$ ($1 \leq x \leq n$ and $1 \leq y \leq k$) being player x ’s weight for the component game t

q : A vector of k elements, the element $q(t)$ is the quota for the component game t

μ : A vector of k elements, the element $\mu(t)$ being the mean weights of the players in N for the component game t

ν : A vector of k elements, the element $\nu(t)$ being the mean weights of the players in N for the component game t

1: **for** $i = 1$ to n **do**

2: $Sum \leftarrow 0$

3: **for** $X = 1$ to $n - 1$ **do**

4: **for** $c = 1$ to k **do**

5: $Com \leftarrow combinations(\{1, 2, \dots, k\}, c)$

6: **for** $temp = 1$ to $\binom{k}{c}$ **do**

7: $Com(temp, \{c + 1, \dots, k\}) \leftarrow \{1, \dots, k\} - Com(temp, \{1, \dots, c\})$

8: **end for**

9: **for** $d = 1$ to $\binom{k}{c}$ **do**

10: $Term \leftarrow 1$

11: **for** $h = 1$ to c **do**

12: $a \leftarrow (q(Com(d, h)) - w(Com(d, h), i))/X; b_1 \leftarrow (q(Com(d, h)) - \epsilon)/X$

13: $Term \leftarrow Term \times \int_a^{b_1} e^{-X \frac{(x - X\mu^g)^2}{2\nu^g}} dx$

14: **end for**

15: **for** $h = c + 1$ to k **do**

16: $a \leftarrow (q(Com(d, h)) - w(Com(d, h), i))/X; b_2 \leftarrow \infty$

17: $Term \leftarrow Term \times \int_a^{b_2} e^{-X \frac{(x - X\mu^g)^2}{2\nu^g}} dx$

18: **end for**

19: $Sum \leftarrow Sum + Term$

20: **end for**

21: **end for**

22: $kE\Delta_i^X \leftarrow Sum$

23: $k\bar{\eta}_i^X \leftarrow kE\Delta_i^X \times \binom{k}{c}$

24: $k\bar{\eta}_i \leftarrow k\bar{\eta}_i + k\bar{\eta}_i^X$

25: **end for**

26: $T \leftarrow T + k\bar{\eta}_i$

27: **end for**

28: **for** $i=1$ to n **do**

29: $\bar{\beta}_i^k \leftarrow \eta_i/2^{n-1}$

30: **end for**

31: **return** $\bar{\beta}^k$

In Line 5, the vector Com is set by invoking the *combinations* function. This function takes a set (for example $\{1, \dots, k\}$) and an integer (for example c) as inputs and returns all combinations of $\{1, \dots, k\}$

of size c . There are $\binom{k}{c}$ such combinations, and they are returned in the vector Com . Thus Com has $\binom{k}{c}$ rows and c columns. Here $1 \leq c \leq k$. Hence the 'for' loop in Line 4 is repeated k times.

The 'for' loop in Line 6, fills the columns $c + 1, \dots, k$ of Com . Specifically, in Line 7, these columns are set to $\{1, \dots, k\} - Com(temp, \{1, \dots, c\})$. The vector Com is now used to check the conditions $C1$ and $C2$ described earlier in this section. This is done in the 'for' loop of Line 9. Here, $Term$ and Sum are computed as per Eqs (8) and (10) respectively. In Lines 22, 23 and 24, we compute $kE\Delta_i^X$, $k\bar{\eta}_i^X$, and $k\bar{\eta}_i$ respectively. This is done as per Eqs (11), (14) and (15) respectively. Finally, Line 29 computes $\bar{\beta}_i^k$ (for $1 \leq i \leq n$) as per Eq. (16).

Analyzing the time complexity of Algorithm 1, we see that the 'for' loops in Lines 3, 4 and 9 are repeated $n - 1$, k , and $\binom{k}{c}$ times (where $1 \leq c \leq k$) respectively. Line 5 takes $\mathcal{O}(k)$ time [10,18]. Lines 7 takes $\mathcal{O}(k)$ time. So the time taken by the loop in Line 6 is $\mathcal{O}(k\binom{k}{c})$. The Lines 10, 12, 13, 16, 17 and 19 take $\mathcal{O}(1)$ time. Thus the time to execute the loop in Line 4 is $\mathcal{O}(k \sum_{c=1}^k \binom{k}{c} \times \mathcal{O}(1)) = \mathcal{O}(k2^k)$. Given this, the time to execute the loop in Line 3 is $\mathcal{O}(nk2^k)$ (since the time to execute the loop in Line 3 is n times the time to execute the loop in Line 4). Note that this is the time to compute a single player's Banzhaf index. So the time to execute it n times for the n players (as is done in the loop in Line 1) is $\mathcal{O}(kn2^k)$ (this is the time to compute the Banzhaf index for all the n players).

Note that time complexity of our approximation method is polynomial in n and exponential in k . However, for most practical games, n may be large but k is small number, typically two or three [4]. This explains the computational feasibility of Algorithm 1.

Having analyzed the time complexity of our approximation method, we will now evaluate it on the basis of its approximation error. To this end, Section 4 provides an analysis in terms of *standard* error and Section 5 in terms of *relative* error. The motivation for analyzing both standard and relative errors is as follows. The former gives an indication of the 'uncertainty' in the approximation (see Section 4 for details on uncertainty of an approximation) while the latter shows how far an approximate is from its exact counterpart. In other words, the relative error requires computing the exact Banzhaf index. Since computing the exact is #P-complete, we can only do so for small n . Thus, although, the relative error shows the magnitude of error, it can be computed only for small n . In contrast, the standard error does not require computing the exact Banzhaf index and can therefore be easily determined for large n . Thus, our analysis for standard error shows the *uncertainty* in the approximation for large and small games, while our analysis of relative error shows the *magnitude* of error for small games.

4. Error analysis: Standard error

We first provide a brief overview of *standard* error and then derive the formula for measuring the standard error in our approximation method for the Banzhaf index.

In general, the concept of error relates to an estimate/approximation made of a quantity which has an exact value [5,38]. One cannot determine exactly how far off the estimate is from the exact value (except for very small problem instances); if this could be done, it would be possible to just give the exact value. Thus, error has to do with *uncertainty* in measurements that nothing can be done about. However, although it is not possible to do anything about such an error, it can be characterized in terms of two essential components [5,38]:

1. a numerical value giving the best 'estimate' (or approximation) possible of the quantity measured, and
2. the degree of *uncertainty* associated with this estimated value.

So if the estimate of a quantity is x and the uncertainty is $e(x)$, then, with ‘reasonable’ certainty, one could say that the quantity would lie in $x \pm e(x)$. The question then is what do we mean by ‘reasonable’ certainty? There are different ways of defining the meaning of ‘reasonable’. One of these is standard error; the lower this error, the more reasonable is the estimate. This error is defined as follows [38].

4.1. Standard error

Let x be a quantity we want to measure but which cannot be measured exactly, it can only be approximated/estimated. If x_1, \dots, x_N denote N estimates of the same quantity x , and ν_x the variance, then ν_x characterizes the average uncertainty in the separate measurements x_1, \dots, x_N . Our best estimate (denoted \bar{x}) is their mean, i.e., $\bar{x} = (\sum_{i=1}^N x_i)/N$. Given that ν_x is the average uncertainty in the separate measurements, the uncertainty in \bar{x} is $\sqrt{\nu_x}/\sqrt{N}$ [38]. This quantity is called the *standard deviation of the mean* or the *standard error* [5,38]. Thus, if we use $e(\bar{x})$ to denote this standard error, we have:

$$e(\bar{x}) = \sqrt{\nu_x}/\sqrt{N} \quad (17)$$

Here, our aim is to determine the standard error in the Banzhaf index. For this, we let $e(E\Delta_i^X)$ denote the standard error in the approximate marginal contribution for player i (see Eq. (4)). We determine this error as follows. In Eq. (4), a and b are the lower and upper limits for mean weight for a coalition of size X . Let $e(a)$ and $e(b)$ denote the standard error in a and b respectively. Then, since the standard deviation of the mean weight is ν/X , as per Eq. (17), we get $e(a) = e(b) = \sqrt{\nu}/X$.

Given this, the error in Eq. (4) is:

$$e(E\Delta_i^X) = \frac{1}{\sqrt{(2\pi\nu)/X}} \times \left(\int_{a-e(a)}^{b+e(b)} e^{-X \frac{(x-X\mu)^2}{2\nu}} dx - \int_a^b e^{-X \frac{(x-X\mu)^2}{2\nu}} dx \right) \quad (18)$$

On the basis of the above error, we find the error in the Banzhaf index by using the following error propagation rules. Let x and y be two estimates with errors $e(x)$ and $e(y)$ respectively. If z is a function of x and y , then the error in z is obtained using the following error propagation rules [38]:

\mathcal{R}_2 If $z = x + y$, the error in z is:

$$e(z) = e(x) + e(y)$$

\mathcal{R}_3 If $z = \alpha x$ where the constant α has no error, then the error in z is:

$$e(z) = |\alpha|e(x)$$

\mathcal{R}_4 If $z = x \times y$, the error in z is:

$$\frac{e(z)}{|z|} = \frac{e(x)}{|x|} + \frac{e(y)}{|y|}$$

\mathcal{R}_5 If $z = x/y$, the error in z is:

$$\frac{e(z)}{|z|} = \frac{e(x)}{|x|} + \frac{e(y)}{|y|}$$

4.2. Standard error for k -majority games

Let $e(PL(h, i, X))$ and $e(PW(h, X))$ denote the errors in $PL(h, i, X)$ and $PW(h, X)$ respectively (see Eqs (12) and (13)). We will compute $e(PL(h, i, X))$ and $e(PW(h, X))$ in the same way as we did $e(E\Delta_i^X)$ in Eq. (18). This gives the following errors:

$$e(PL(h, i, X)) = \frac{1}{\sqrt{(2\pi\nu(h)/X)}} \left(\int_{a+e(a)}^{b_1+e(b_1)} e^{-X \frac{(x-X\mu(h))^2}{2\nu(h)}} dx - \int_a^{b_1} e^{-X \frac{(x-X\mu(h))^2}{2\nu(h)}} dx \right) \quad (19)$$

$$e(PW(h, X)) = \frac{1}{\sqrt{(2\pi\nu(h)/X)}} \left(\int_{q(h)-e(q(h))}^{\infty} e^{-X \frac{(x-X\mu(h))^2}{2\nu(h)}} dx - \int_{q(h)}^{\infty} e^{-X \frac{(x-X\mu(h))^2}{2\nu(h)}} dx \right) \quad (20)$$

Let $e(Term(row, X, j, i))$ denote the error in $Term(row, X, j, i)$. Applying the rule \mathcal{R}_4 to Eq. (8), we get $e(Term(X, i, j))$ as follows:

$$\frac{e(Term(row, X, j, i))}{Term(row, X, j, i)} = \sum_{f=1}^j \frac{e(PL(Com(row, f), i, X))}{PL(Com(row, f), i, X)} + \sum_{g=j+1}^k \frac{e(PW(Com(row, g), X))}{PW(Com(row, g), X)} \quad (21)$$

Applying the rule \mathcal{R}_2 to Eq. (10), we get $e(Sum(X, j, i))$ as follows:

$$e(Sum(X, j, i)) = \sum_{j=1}^{\binom{k}{c}} e(Term(row, X, j, i)) \quad (22)$$

Applying the rule \mathcal{R}_2 to Eq. (11), we get $e(kE\Delta_i^X)$ as follows:

$$e(kE\Delta_i^X) = \sum_{j=1}^k e(Sum(X, j, i)) \quad (23)$$

Applying the rule \mathcal{R}_3 to Eq. (14), we get $e(k\bar{\eta}_i^X)$ as follows:

$$e(k\bar{\eta}_i^X) = \binom{n-1}{X} \times e(kE\Delta_i^X) \quad (24)$$

Applying the rule \mathcal{R}_2 to Eq. (15), we get the error $e(k\bar{\eta}_i)$ as follows:

$$e(k\bar{\eta}_i) = \sum_{X=1}^{n-1} e(k\bar{\eta}_i^X) \quad (25)$$

Applying the rule \mathcal{R}_3 to Eq. (16), we get $e(\bar{\beta}_i^k)$ as follows:

$$e(\bar{\beta}_i^k) = e(k\bar{\eta}_i)/2^{n-1} \quad (26)$$

From the above analysis, we conclude the following. As per Eq. (23), $e(kE\Delta_i^X)$ increases with k . This is because $e(Sum(X, i, j)) \geq 0$. Consequently, as per Eq. (26), $e(\bar{\beta}_i^k)$ also increases with k .

5. Error analysis: Relative error

Here we evaluate the performance of the proposed approximation methods in terms of their *relative* approximation errors. While standard error (measured in Section 4), is a measure of the uncertainty in an approximation, relative error is a measure of the difference between an approximate and its corresponding exact [1].⁶ Now, since the proposed approximation methods are randomized methods, the relative error may be different for different problem instances (i.e., for different voting games). Thus measuring the relative error for a single problem instance may not give a proper indication of the performance of the method. So we experimentally measure the relative error for a wide range of voting games and then find the average error.

The aim of these experiments is as follows. First, we want to show, on average, how far an approximate is from the corresponding exact. Second, we want to show how this error changes with k .

The experimental setting is as follows. We generate a range of voting games by varying the number of players between $n = 3$ and $n = 9$. It must be noted that the reason for having at most 9 players is that we are computing the exact solution for #P-complete problems. We also vary k between $k = 1$ and $k = 5$. For each n and k , we generate random weights in a given range. Note that, for each n and k , a player has k weights, one for each of the k components of the game. Let $\beta_i^k(j, n)$ denote i 's exact Banzhaf index for the j th game. Likewise, let $\bar{\beta}_i^k(j, n)$ denote i 's approximate Banzhaf index for the j th game. Lastly, let $E_j(\bar{\beta}_i^k(j, n))$ denote the *percentage relative error* in the Banzhaf index for the j th game. This error is defined as follows:

$$E_j(\bar{\beta}_i^k(j, n)) = \frac{1}{n} \sum_{i=1}^n 100 \times \frac{\text{abs}(\beta_i^k(j, n) - \bar{\beta}_i^k(j, n))}{\text{max}\{\beta_i^k(j, n), \bar{\beta}_i^k(j, n)\}} \quad (27)$$

If we let $E(\bar{\beta}_i^k(n))$ denote the average of these errors for the 25 randomly generated games, i.e., we have :

$$E(\bar{\beta}_i^k(n)) = \frac{1}{25} \sum_{j=1}^{25} E_j(\bar{\beta}_i^k(j, n)) \quad (28)$$

Recall that, for each k , the number of players is between $n = 3$ and $n = 9$. Thus, for each k , we have seven different values of n . So the average percentage error ($E(\bar{\beta}^k)$) across all these n is computed as follows:

$$E(\bar{\beta}^k) = \frac{1}{7} \sum_{a=1}^9 E(\bar{\varphi}^k(a)) \quad (29)$$

The results of these experiments are summarized in Fig. 1. This figure demonstrates the following facts:

1. For a given k , on average, how far the Banzhaf index is from the corresponding exact. For instance, consider $k = 1$ for which the average error is 15%. This means that, for voting games with $3 \leq n \leq 9$ and $k = 1$, the relative error is, on average 15%. The standard deviation in the error is 7%.

⁶Clearly, computing relative error for the proposed approximation methods requires computing the exact Banzhaf index.

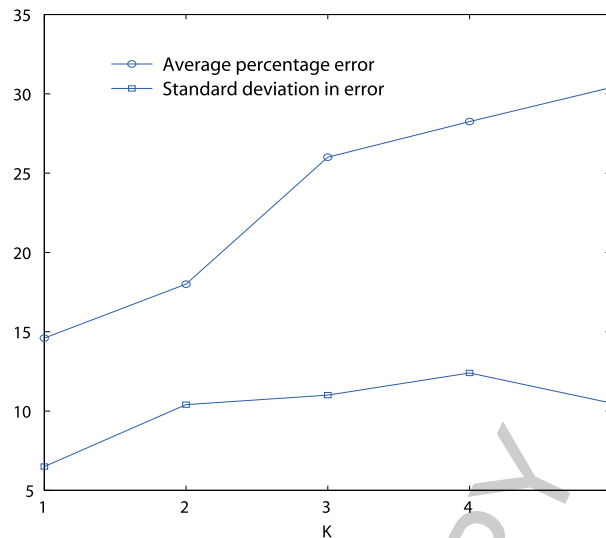


Fig. 1. The average percentage error ($E(\beta^k)$) in the Banzhaf index and the standard deviation in the error. (Colours are visible in the online version of the article; <http://dx.doi.org/10.3233/MGS-2012-0194>)

- How the error in the approximate Banzhaf index changes with k . As shown in the figure, the average percentage error increases in k . This is logical since, for a given k , the proposed method require computing the approximate Banzhaf index in terms of k component approximations. In other words, the error accumulates and therefore increases with k . Note that, the aim of these experiments is not just to demonstrate the fact that error increases in k but also to show by what ‘magnitude’ the error increase. For instance, as per Fig. 1, the magnitude of increase in error from $k = 1$ to $k = 2$ is 2.5%. The standard deviation also increased with k . But the increase in the standard deviation was less than the increase in error. When k increased from $k = 1$ to $k = 5$, the error increased from 15% to 30%, and the standard deviation increased from 7 to 12.

Also, this result regarding the increase in error with k matches the conclusion made in Section 4.1 for the standard error. Note that the standard error is a measure of the degree of uncertainty (measured in terms of the standard deviation) in an approximation. However, unlike the relative percentage error (measured in this section) it does not require measuring the difference between an approximate and its corresponding exact. Despite the difference in the way the standard and relative errors are measured, they both were found to increase in k .

- What can we say about the error for larger games, i.e., those with more than 9 players? Recall that a main aim of these experiments is to show how far an approximate is from its corresponding exact value. Since this requires computing the exact value, we had to limit our study to games with a small number of players ($n \leq 9$). Nevertheless, these results are important because they shed light on how the proposed approximation methods will perform for larger games. Since the proposed methods are randomized, and we know that the approximation error for such methods decreases with the size of a problem instance, we can conclude that for voting games for a given k and $n > 9$, the error will be less than that in Fig. 1 for the same k . Likewise, the standard deviation will also be lower.

Before closing this section, we would like to re-iterate the difference between standard error and relative error. If the standard error in an estimate x is $e(x)$, it means that, with reasonable certainty, the

exact value lies in $x \pm e(x)$. However, there is a small chance that the exact lies outside this range. In contrast, the relative error is measured with reference to the corresponding exact. In other words, the advantage of relative error is that it is a better indication of approximation error but its drawback is that it is hard to determine this error for large games. This explains our motivation for analyzing the relative error for small games and standard error for large games.

6. Related work

A number of methods have been proposed for measuring voting power. These include [3,6,8,28,31,34]. A survey of these methods can be found in [15,19]. Since the indices are hard to compute, the following methods have been proposed to approximate them [24]:

1. Monte Carlo simulation method [23]
2. Multi-linear extension (MLE) method [26,27]
3. Modified MLE method [21]
4. Random permutation method [40]
5. Bachrach et al approximation method [2]
6. Randomized approximation method [13]

The earliest approximation method was proposed by Mann and Shapley [23]. This method is based on Monte Carlo simulation and estimates the Shapley value from a random sampling of coalitions. The advantage is its linear time complexity. But a key issue here is that it does not give details of how the samples are to be drawn. Also, there is no assessment of the accuracy of this method, and the method is for one-majority games.

The MLE approximation method was proposed in [26] for computing the Shapley value and this method was later extended [27] to approximate the Banzhaf index for one-majority games. Again, while the advantage of this method is its linear time complexity, it is for one-majority games and there is no evaluation of its approximation error.

Leech [21] modified Owen's MLE method for the Shapley value and the Banzhaf index by trading-off computational time in order to improve the error of approximation. In more detail, this modified MLE method combines the essential features of direct enumeration and MLE in order to improve the accuracy of the MLE method. Specifically, the players are divided into two subsets: major players with large weights $L = \{1, 2, \dots, l\}$, and minor players $N - L$. This combined method treats the major players using enumeration as in the direct approach, but treats minor players using Owen's MLE approximation technique. Large values of l will improve accuracy, but will also increase computation time. The advantage of this method is that it generates a better approximation than Owen's MLE method, but its disadvantage is that it has exponential time complexity: $\mathcal{O}(2^l)$.

Zlotkin and Rosenschein [40] proposed a *random permutation mechanism* for approximating the Shapley value. In this method, the players choose a random permutation and form the full coalition, one player after another, according to the chosen permutation. Here each player gets a utility equal to its contribution to the coalition at the time of joining it. If each permutation has equal chance of being chosen, then this mechanism gives each player an expected utility equal to its Shapley value. This method requires the players to agree on an all-or-nothing deal. The advantage of this method is its linear time complexity. However, for weighted voting games, getting the players to agree on an all-or-nothing deal may be an issue because, for these games, there is only one swing player for each possible coalition. So only one player gets a utility of one and all others get zero utility.

Using a random sampling approach, Bachrach et al provide an analysis of the error bounds and minimum number of samples required to achieve a given accuracy [2]. In more detail, they give randomized approximation methods for power indices such as the Shapley value, which can be used for simple coalitional games. They show that their approximation methods approach the optimal, and give lower bounds for both deterministic and randomized approaches to computing the Shapley value and the Banzhaf index. This work focuses on simple coalitional games, but does not deal with the performance of the approximation method in the context of k -majority games.

An approximation method for the Shapley value for k -majority games was given in [13]. This method was then extended in [14] for computing the Banzhaf index. The difference between [14] and this paper is that the former provides a theoretical comparative analysis of the approximation methods for the Shapley value and the Banzhaf index. The latter focuses on the Banzhaf index part and extends it by including an experimental analysis of the approximation error.

Before concluding, we emphasize the significance of k -majority voting games relative to one-majority games. Games with $k > 1$ are more general in the sense that they provide a means of determining a winning coalition on the basis of not just a single rule but more than one rule. For example, $k > 1$ for the voting game used for elections to Nice European Council of ministers [4]. Here the winning coalition of countries is determined not just on the basis of its total weight but also on the basis of the number of countries in the coalition. Hence our focus on k -majority games. Furthermore, unlike existing methods, we demonstrate how our method perform in terms of the *standard error* and also in terms of the *relative error* of approximation.

7. Conclusions and future work

The Banzhaf index is a well known and widely used index for measuring the power a player has in a voting game. However, the problem of computing it is #P-complete. To overcome this problem, we presented a heuristic randomized method for computing an approximate Banzhaf index for k -majority voting games. The complexity of this method is $\mathcal{O}(kn^22^k)$ where n is the number of players and k is the number of component games. We also experimentally analyzed this method in terms of its approximation error and showed that the error ranges between 15% for random games with $k = 1$ to 30% for $k = 5$.

The Banzhaf index finds application in a number of multi-agent settings. These include analysis of voting structures, cost sharing and resource allocation, and utility analysis in multi-party negotiations. By providing computationally feasible methods for approximating the Banzhaf index, our work facilitates its application in such multi-agent settings.

Apart from the Banzhaf index, there are other definitions of power indices such as Coleman index and Deegan-Packel index. Future work will deal with computing these for k -majority games.

Acknowledgment

The first author acknowledges the EPSRC grant EP/G000980/1. The authors are thankful to the reviewers for their helpful comments.

References

- [1] G. Ausiello, P. Crescenzi, G. Gambosi, V. Kann, A. Marchetti-Spaccamela and M. Protasi, Complexity and approximation: Combinatorial optimization problems and their approximability properties, *Springer*, 2003.

- [2] Y. Bachrach, E. Maarkakis, E. Reznik, A.D. Procaccia, J. Rosenschein and A. Saberi, Approximating power indices – theoretical and empirical analysis, *Journal of Autonomous Agents and Multi-Agent Systems* **20**(2) (2010), 105–122.
- [3] J.F. Banzhaf, Weighted voting doesn't work: A mathematical analysis, *Rutgers Law Review* **19** (1965), 317–343.
- [4] J.M. Bilbao, J.R. Fernandez, N. Jimenez and J.J. Lopez, Voting power in the European Union enlargement, *Journal of Operational Research* **143** (2002), 181–196.
- [5] P.V. Bork, H. Grote, D. Notz and M. Regler, Data analysis techniques in high energy physics experiments, *Cambridge University Press*, 1993.
- [6] J.S. Coleman, Control of collectives and the power of a collectivity to act, in: *Social Choice Gordon and Breach*, J.S. Coleman, ed., 1971. IOS press, Amsterdam.
- [7] V. Conitzer and T. Sandholm, Complexity of constructing solutions in the core based on synergies among coalitions, *Artificial Intelligence Journal* **170** (2006), 607–619.
- [8] J. Deegan and E.W. Packel, A new index for power of simple n -person games, *International Journal of Game Theory* **7**(2) (1978), 113–123.
- [9] X. Deng and C.H. Papadimitriou, On the complexity of cooperative solution concepts, *Mathematics of Operations Research* **19**(2) (1994), 257–266.
- [10] G. Ehrlich, Loopless algorithms for generating permutations, combinations and other combinatorial configurations, *Journal of the Association for Computing Machinery* **20**(3) (1973), 500–513.
- [11] S.S. Fatima, M. Wooldridge and N.R. Jennings, A randomized method for the shapley value for the voting game, in: *Proceedings of the Sixth International Conference on Autonomous Agents and Multi-Agent Systems*, 2007, pp. 955–962. IOS press, Amsterdam.
- [12] S.S. Fatima, M. Wooldridge and N.R. Jennings, Automated analysis of weighted voting games, in: *Proceedings of the Thirteenth International Conference on Electronic Commerce*, 2011. IOS press, Amsterdam.
- [13] S.S. Fatima, M. Wooldridge and N.R. Jennings, A linear approximation method for the shapley value, *Artificial Intelligence Journal* **172** (2008), 1673–1699.
- [14] S.S. Fatima, M. Wooldridge and N.R. Jennings, An approximation method for power indices for voting games, in: *Innovations in Agent-Based Complex Automated Negotiations*, T. Ito, M. Zhang, V. Robu, S. Fatima, T. Matsuo and H. Yamaki, eds, 2010, pp. 179–193. IOS press, Amsterdam.
- [15] D. Felsenthal and M. Machover, The measurement of voting power: Theory and practice, problems and paradoxes, *Public Choice* **102** (2000), 373–376.
- [16] A. Francis, Advanced level statistics, *Stanley Thornes Publishers*, 1979.
- [17] J. Greenberg and R.L. Cohen, Equity and justice in social behavior, *Academic Press*, 1982.
- [18] D.E. Knuth and J.L. Szwarcfiter, A structured program to generate all topological sorting arrangements, *Information Processing Letters* **2** (1974), 153–157.
- [19] D. Leech, An empirical comparison of the performance of classical power indices, *Political Studies* **50**(1) (2002), 1–22.
- [20] D. Leech, Voting power in the governance of the international monetary fund, *Annals of Operations Research* **109** (2002), 375–399.
- [21] D. Leech, Computing power indices for large voting games, *Management Science* **49**(6) (2003), 831–837.
- [22] M. Machover and D.S. Felsenthal, The treaty of nice and qualified majority voting, *Social Choice and Welfare* **18**(3) (2001), 431–464.
- [23] I. Mann and L.S. Shapley, Values for large games iv: Evaluating the electoral college by monte-carlo techniques, *Technical Report The RAND Corporation, Santa Monica*, 1960.
- [24] T. Matsui and Y. Matsui, A survey of algorithms for calculating power indices of weighted majority games, *Journal of Operations Research* **43**(1) (2000), 71–86.
- [25] M.J. Osborne and A. Rubinstein, A course in game theory, *The MIT Press*, 1994.
- [26] G. Owen, Multilinear extensions of games, *Management Science* **18**(5) (1972), 64–79.
- [27] G. Owen, Multilinear extensions and the Banzhaf value, *Naval Research Logistic Quarterly* **2** (1975), 741–750.
- [28] L.S. Penrose, The elementary statistics of majority voting, *Journal of the Royal Statistical Society* **109** (1946), 53–57.
- [29] T.P. Jeffrey, A.M. Elizabeth and A.N. Margaret, Interest alignment and coalitions in multiparty negotiation, *The Academy of Management Journal* **41**(1) (1998), 42–54.
- [30] K. Prasad and J.S. Kelly, NP-completeness of some problems concerning voting games, *International Journal of Game Theory* **19** (1990), 1–9.
- [31] D.W. Rae, Decision rules and individual values in constitutional choice, *American Political Science Review* **163** (1969), 40–56.
- [32] T. Rahwan and N.R. Jennings, An algorithm for distributing coalitional value calculations among cooperating agents, *Artificial Intelligence Journal* **171** (2007), 535–567.
- [33] T. Sandholm and V. Lesser, Coalitions among computationally bounded agents, *Artificial Intelligence Journal* **94**(1) (1997), 99–137.

- [34] L.S. Shapley, A value for n person games, in: *The Shapley value*, A.E. Roth, eds, 1988, pp. 31–40. University of Cambridge Press, Cambridge. IOS press, Amsterdam.
- [35] 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.
- [36] O. Shehory and S. Kraus, A kernel-oriented model for coalition-formation in general environments: Implementation and results, in: *In Proceedings of the National Conference on Artificial Intelligence (AAAI-96)*, 1996, pp. 131–140.
- [37] O. Shehory and S. Kraus, Methods for task allocation via agent coalition formation, *Artificial Intelligence Journal* **101**(2) (1998), 165–200.
- [38] J.R. Taylor, An introduction to error analysis: The study of uncertainties in physical measurements, *University Science Books*, 1982.
- [39] H.P. Young, Cost allocation, in: *The Handbook of Game Theory, Volume 2*, R.J. Aumann and S. Hart, eds, 1994, pp. 1193–1235. IOS press, Amsterdam.
- [40] G. Zlotkin and J. Rosenschein, Coalition cryptography and stability: Mechanisms for coalition formation in task oriented domains, in: *In Proceedings of the National Conference on Artificial Intelligence (AAAI-94)*, 1994, pp. 432–437. IOS press, Amsterdam.

AUTHOR COPY