Auton Agent Multi-Agent Syst (2010) 20:105–122
DOI 10.1007/s10458-009-9078-9

Approximating power indices: theoretical and empirical analysis

Yoram Bachrach · Evangelos Markakis · Ezra Resnick · Ariel D. Procaccia · Jeffrey S. Rosenschein · Amin Saberi

Published online: 14 February 2009

Abstract Many multiagent domains where cooperation among agents is crucial to achieving a common goal can be modeled as coalitional games. However, in many of these domains, agents are unequal in their power to affect the outcome of the game. Prior research on weighted voting games has explored power indices, which reflect how much "real power" a voter has. Although primarily used for voting games, these indices can be applied to any simple coalitional game. Computing these indices is known to be computationally hard in various domains, so one must sometimes resort to approximate methods for calculating them. We suggest and analyze randomized methods to approximate power indices such as the Banzhaf power index and the Shapley–Shubik power index. Our approximation algorithms do not depend on a specific representation of the game, so they can be used in *any* simple coalitional game. Our methods are based on testing the game's value for several sample coalitions. We show that no approximation algorithm can do much better for general coalitional games, by providing lower bounds for both deterministic and randomized algorithms for calculating power indices. We also provide empirical results regarding our method, and show that it typically achieves much better accuracy and confidence than those required.

A preliminary version of this paper appeared in [2]. The previous version contained a theoretical analysis of our suggested methods for approximating power indices. This version contains new empirical analysis of the performance of these methods, for both the Banzhaf and Shapley–Shubik power indices, based on simulations of several types of voting games, as well as a discussion of these new results and their significance.

Y. Bachrach (✉) · E. Resnick · A. D. Procaccia · J. S. Rosenschein
School of Engineering and Computer Science, The Hebrew University, Jerusalem, Israel
e-mail: yorambac@gmail.com

E. Markakis
Center for Mathematics and Computer Science (CWI), Amsterdam, The Netherlands

E. Markakis
Athens University of Economics and Business, Athens, Greece

A. Saberi
Department of Management Science and Engineering, Stanford University, Palo Alto, CA, USA

Keywords Power indices · Power index · Coalitional games · Shapley value · Banzhaf power index · Shapley–Shubik power index · Power index approximation

1 Introduction

Cooperation is critical to many types of interaction among self-interested agents. In many domains, agents require one another in order to achieve their goals. When the outcomes, achieved by a coalition of agents, can be described in terms of success or failure, we can model the system as a simple coalitional game.

One example of such a domain is the case of coalitions among voting agents. When voting on a candidate under a certain protocol, agents may work together so that a certain candidate will be chosen. A well-known game-theoretic model of cooperation in voting domains is that of weighted voting games. Each of the players in such a game has a weight, and a coalition of players wins the game if the sum of the weights of its participants exceeds a certain quota.

One question that arises is that of *measuring the power* different agents possess in cooperative settings. In some domains, agents are unequal in their power to affect the outcome of the game, and the weight of a player in a voting game does not necessarily reflect the actual power that player has over the decisions of a larger group.

A common interpretation of the power an agent possesses is that of its a priori probability of having a significant role in the game. A power index measures the ability a certain agent has to affect the result of the game; thus, power indices reflect how much "real power" an agent has. Two prominent power indices are the Shapley–Shubik power index [29] and the Banzhaf power index [4]. Although power indices have mainly been considered in the context of weighted voting systems, it is possible to apply them to any simple coalitional game.

Power indices have several uses. The most prominent of these is measuring political power in decision-making bodies. As discussed in Sect. 3, power index-based analysis has been applied to several political bodies. Another application of power indices is cost-sharing schemes and cost allocation [32]. Due to their applicability to such real-world problems, calculating power indices is an important problem. Unfortunately, these indices are generally hard to compute, even in restricted domains.

We suggest and analyze approximation algorithms for calculating power indices of a given agent, for any simple coalitional game. We consider a query regarding the value of any coalition to be the basic operation. Our method randomly samples coalitions, and tests whether the given agent is critical in the sample. We use the samples to estimate the Banzhaf power index, or more precisely, to build a confidence interval for it. Our procedure returns an interval $[l, r]$ (where $l, r \in \mathbb{R}$), so that the correct Banzhaf index lies inside the interval with high probability $1 - \delta$ (where δ is the maximal allowed error probability).

The required number of samples (and thus the running time of the algorithm) depends on two parameters: the desired accuracy of the procedure, as defined by the width of the interval, and the desired confidence level, or the probability that the correct value actually lies outside the interval, namely δ. In fact, we later show that it is easy to trade off one parameter for the other, and provide the equations linking these parameters.

Although we handle the Banzhaf power index, we show that slight changes in the algorithm allow it to approximate the Shapley–Shubik index as well. We show that no other algorithm can obtain much better results for simple coalitional games, by providing lower bounds for both deterministic and randomized algorithms for calculating the Banzhaf power index. We first provide a lower bound on the number of queries required for a deterministic algorithm with a given accuracy. We then use the Minimax principle of Yao [31] to show that no randomized algorithm can achieve superpolynomial approximation. Finally, we present results of an empirical evaluation of our procedure, showing that it typically achieves much better accuracy and confidence than those required, for both the Banzhaf and Shapley–Shubik power indices.

The theoretical and empirical results in this paper have several applications. In political domains involving decision-making bodies, where only an approximate power index analysis might be required, our results enable one to obtain good approximations of power indices. Also, many multiagent systems where agents make joint decisions use weighted voting (or some weighted voting variant which can be modeled as a simple coalitional game). In such domains, one sometimes wants to obtain many power index results, and the algorithms in this paper show how to do so effectively, since our theoretical and empirical bounds show how to obtain accurate results in short amounts of time.

1.1 Structure of the paper

The paper proceeds as follows. In Sects. 2.1 and 2.2 we give some background concerning coalitional games and power indices. In Sect. 3 we discuss some related work regarding power indices and the computational complexity of calculating power indices, as well as approximate methods for calculating these indices. In Sect. 4 we discuss our method of approximating power indices using a sampling technique. Section 5 shows that the results achieved by the sampling technique are in a sense the best possible, since no deterministic algorithm can achieve comparable accuracy with a polynomial number of queries, and since no randomized algorithm can achieve superpolynomial accuracy. Section 6 contains an empirical analysis of our suggested method, and provides results regarding its actual accuracy and confidence (rather than the theoretical bounds on these). We conclude in Sect. 7.

2 Model and definitions

We now give several definitions regarding coalitional games and power indices, required for the rest of this paper.

2.1 Coalitional games

A coalitional game is composed of a set of n agents, I, and a function mapping any subset (coalition) of the agents to a real value $v : 2^I \to \mathbb{R}$. In a *simple* coalitional game, v only gets values of 0 or 1 ($v : 2^I \to \{0, 1\}$). We say a coalition $C \subseteq I$ *wins* if $v(C) = 1$, and say it *loses* if $v(C) = 0$.[1]

An agent i is *critical* in a winning coalition C (sometimes also called a "swinger" or "pivot") if the agent's removal from that coalition would make it a losing coalition.

A critical agent has a strong influence on the result of the game, so this property is related to various measures of power. Two approaches to measuring the power of individual agents in simple coalitional games are the Banzhaf index and the Shapley–Shubik index.

[1] In many such settings the coalitional function v is *monotone* in the sense that if $A \subseteq B$ then $v(A) \leq v(B)$. Another common assumption, especially in weighted voting domains, is that it is impossible for both a coalition and its complement to win, so if $v(C) = 1$ then $v(I \setminus C) = 0$.

2.2 The Banzhaf index and the Shapley–Shubik index

The Banzhaf index depends on the number of coalitions in which an agent is critical, out of all the possible coalitions that contain the agent. The Banzhaf index is given by $\beta = \beta(v) = (\beta_1(v), \ldots, \beta_n(v))$ where

$$\beta_i(v) = \frac{1}{2^{n-1}} \sum_{S \subset I | i \in S} [v(S) - v(S \setminus \{i\})]$$

The Shapley–Shubik power index is simply the name of the Shapley value when applied in a setting of a simple coalitional game. Denote by π a permutation (reordering) of the agents, so π is a reversible function $\pi : \{1, \ldots, n\} \to \{1, \ldots, n\}$, and by Π the set of all possible such permutations. Denote by $S_\pi(i)$ the predecessors of i in π, so $S_\pi(i) = \{j | \pi(j) < \pi(i)\}$. The Shapley–Shubik index is given by $sh(v) = (sh_1(v), \ldots, sh_n(v))$ where

$$sh_i(v) = \frac{1}{n!} \sum_{\pi \in \Pi} [v(S_\pi(i) \cup \{i\}) - v(S_\pi(i))]$$

The definition of the Banzhaf index reflects the assumption that each coalition has an equal probability of occurring, while the Shapley–Shubik index reflects the assumption that any *ordering* of the agents entering the coalition has an equal probability of occurring.

3 Related work

The Banzhaf index emerged directly from the study of voting in decision-making bodies. A first version of the Banzhaf power index was introduced in [4]. This version is called the *normalized* Banzhaf index. It measures the proportion of coalitions in which a player is a swinger, out of all *winning* coalitions, and normalizes the indices of all the agents, so they sum up to 1. The normalized Banzhaf index is defined as:

$$\widetilde{\beta}_i = \frac{\beta_i(v)}{\sum_{k \in I} \beta_k}$$

The normalized Banzhaf index was analyzed in [11], where it was shown that this normalization lacks certain desirable properties, and the more natural Banzhaf index was introduced.

A similar power index, the Shapley–Shubik power index, originated in work on game theory. In his seminal paper, Shapley [28] considered coalitional games and the fair allocation of the utility gained by the grand coalition. The Shapley–Shubik index [29] is simply an application of the Shapley value to simple coalitional games.

Both the Shapley–Shubik and the Banzhaf indices have been widely studied. Straffin [30] has shown that each index reflects certain conditions in a voting body, relating to how a coalition is formed. Reference [17] describes the axioms that characterize both the Banzhaf and Shapley–Shubik power indices, along with several others.

Various power indices, including the Banzhaf power index and the Shapley–Shubik power index, have been surveyed in [5,9,15,18]. These indices were applied in an analysis of the voting structures of several bodies, including the European Union Council of Ministers and the IMF [19,21]. Reference [6] considers a power index-based analysis of the European Union, and shows that such analysis is computationally difficult, as brute-force techniques for computing these indices are not always tractable. Another application of power indices is cost-sharing schemes: the Shapley value is well known for its use in establishing the fair sharing of costs [32].

The naive implementation of calculating the Banzhaf index is exponential, since the number of possible coalitions is exponential in the number of agents. The situation is even worse for the Shapley–Shubik power index. For n agents, there are $n!$ permutations to consider. Using Stirling's approximation,[2] this means there are about $O(2^{n \log n})$ permutations to check. Calculating power indices in time polynomial in the number of agents can only be achieved in very specific and restricted domains.

Reference [10] shows that computing the Shapley–Shubik index in weighted majority games is #P-complete. Similar results [25,27] show that calculating both the Banzhaf and Shapley–Shubik indices in weighted voting games is NP-complete. The problem of power-index comparison is studied in [12], and is shown to also be hard in general.

Bachrach and Rosenschein [3] have considered the problem of calculating the Banzhaf power index in *network flow games*. This is a game-theoretic model of a network reliability problem. In this game, each agent controls an edge in a network flow graph, and a coalition of agents wins if it manages to allow a certain flow between a source vertex and a target vertex. Bachrach and Rosenschein [3] have shown that in this specific domain, calculating the Banzhaf power index is #P-complete, but gave a polynomial algorithm for a certain restricted case, namely of connectivity games in bounded layer graphs.

The example above shows that restricting the domain may allow one to find ways of overcoming the computational difficulty of calculating power indices in the general case. However, the hardness results for various *general* domains indicate that in order to calculate power indices one must either restrict the domain, or *approximate* the power index.

One approach, using generating functions, has been suggested in [23] and analyzed in [7]. That method trades time complexity for storage space. Owen [26] describes methods for computing the power indices exactly, based on the multilinear extension (MLE) of a game. These have similar complexity issues as direct enumeration, but serve as a basis for approximation techniques presented in [20], which allows trading off time and approximation quality.

A Monte-Carlo approach to approximating the Shapley value has been suggested in [22]. That paper only considered the Shapley–Shubik power index, and not the Banzhaf index, and did not provide a complete confidence interval statistical analysis.[3] We apply a similar approach to approximating both indices, and provide a more rigorous statistical analysis. We do this by investigating the required number of samples for building a *confidence interval* with a given confidence level and accuracy. We also complement these results by providing lower bounds on the number of required samples, for both deterministic and randomized algorithms for calculating power indices.

The performance of an approximation method should be evaluated in terms of two criteria: its *time complexity* and its quality (the accuracy of the procedure, and the confidence it guarantees).

A randomized method for calculating the Shapley–Shubik power index of weighted voting games has been suggested in [13,14]. Although that work provided an analysis of the statistical error in such a procedure, it only considered the Shapley–Shubik power index in weighted voting games, and the evaluation of the approximation error was carried out empirically.

Reference [24] contains a survey of algorithms for calculating the power indices of weighted majority games. It discusses a Monte-Carlo approach for calculating the Banzhaf

[2] See [1] for more information regarding Stirling's approximation of factorials.

[3] We use Hoeffding's inequality [16], which was introduced in 1963, a few years after [22] was published.

power index, but only shows how to calculate the maximum likelihood estimator. Reference [24] does not show how to build a confidence interval for a given confidence level. It focuses on weighted voting games, in which comparing the power of two agents is simple: in weighted voting games a player with a higher weight cannot have a smaller power index than a player with a smaller weight. Our work also considers general simple games where such a simple way of comparing power indices does not exist, and shows how to rank agents according to their power indices in such domains.

To our knowledge, no work has considered lower bounds for power index approximation algorithms. In this paper, we provide such bounds, showing that our randomized approach outperforms any deterministic algorithm in terms of accuracy (given a certain number of samples), and that no randomized algorithm can achieve significantly better results than those achieved by our algorithms. These lower bounds hold for general coalitional games, where the structure of the coalitional function is unrestricted. For restricted domains, it may be possible to provide better deterministic algorithms, or better randomized algorithms.

4 Approximating power indices by sampling

We suggest a method for approximating the power indices of a certain agent in a simple game. The basic operation our algorithms use is a query regarding the value of a coalition. Calculating β_i, the Banzhaf index for agent i, is performed by randomly sampling *coalitions containing player* i, and estimating β_i by the proportion of the sampled coalitions where agent i is critical. When sampling coalitions, each sample has a probability of β_i of being a coalition where agent i is critical, so we can approximate β_i by taking into consideration several such samples.

The number of samples determines the accuracy of this procedure: for a given $\epsilon > 0$, the probability δ of missing the correct value β_i by more than ϵ depends on the number of samples used. The algorithm we propose determines the required sample size according to the required confidence level (δ, the probability of error) and approximation accuracy (ϵ, the maximal allowed distance from the correct value).

4.1 Randomly sampling coalitions

The suggested approximation algorithm relies on a sampling procedure. The Banzhaf sampling procedure returns a random coalition C which contains player i. This can easily be done by randomizing a bit using the uniform distribution over $\{0, 1\}$ for each of the other agents, and always setting the bit for agent i to 1.

Once the random sample is returned, we simply check if agent i is critical in that sample. Agent i is critical in the returned coalition C if:

$$v(C) - v(C \setminus \{i\}) = 1$$

We denote this as $\mathit{Critical}(i, C)$.

It is easy to see from the definition of the Banzhaf index that the probability that agent i is critical in a random coalition that contains it is exactly its Banzhaf power index, so:

$$Pr_{C|i \in C}(\mathit{Critical}(i, C)) = \beta_i$$

Let C_j be a random coalition containing agent i, as defined previously. Let X_j be the random variable which is 1 if agent i is critical in C_j, and 0 otherwise.

4.2 Estimating the power index

When attempting to estimate the power index of an agent i, we can randomly sample coalitions in the way described above. For each such sample we can check if agent i is critical in that coalition. Given k such sample coalitions, we can get an estimator for β_i.

Lemma 1 *Let $C_1, \ldots, C_k$ be a set of k randomly sampled coalitions, and let $X_1, \ldots, X_k$ be the series of k Bernoulli trials (random trials with a boolean result), as defined above. Let X be the number of successes in this series of Bernoulli trials, $X = \sum_{j=1}^{k} X_j$. Then the* maximum likelihood estimator *for β_i is:*

$$\hat{\beta}_i = \frac{X}{k}$$

This estimator is unbiased.

Proof As discussed in Sect. 4.1, each such X_j is a single Bernoulli trial, and $Pr(X_j = 1) = \beta_i$ and $Pr(X_j = 0) = 1 - \beta_i$. $X_1, \ldots, X_k$ is a series of k such Bernoulli trials. X is the number of successes in this series of Bernoulli trials, $X = \sum_{j=1}^{k} X_j$, and thus has the binomial distribution $X \sim B(k, \beta_i)$. Since the X_j's are independent but identical Bernoulli trials, the *maximum likelihood estimator* for β_i is $\hat{\beta}_i = \frac{X}{k}$. This estimator is known to be unbiased for the binomial distribution. □

4.3 A confidence interval for the power index

The estimator $\hat{\beta}_i$ by itself does not provide a bound on the probability that this value is approximately correct. Given a sample $X_1, \ldots, X_k$ of k such Bernoulli trials, we are interested in getting a value that is probably approximately correct (PAC).

If for some $\epsilon > 0$, we consider values that are within a distance of ϵ from the correct value, β_i, accurate enough, and are willing to accept a certain low probability δ of having our estimator $\hat{\beta}_i$ miss β_i by more than ϵ, we can formulate the problem as building a *confidence interval* for β_i, with an accuracy of ϵ and with confidence level of $1 - \delta$. The interval we build is:

$$[\hat{\beta}_i - \epsilon, \hat{\beta}_i + \epsilon]$$

The interval is centered at $\hat{\beta}$, has a width of $2 \cdot \epsilon > 0$, and contains the true β_i with a probability of at least $1 - \delta$.

We obtain an equation relating the required number of samples k, the confidence level δ, and the accuracy ϵ (width of the interval), by using Hoeffding's inequality [16].

Theorem 1 (Hoeffding's inequality) *Let $X_1, \ldots, X_n$ be independent random variables, where all X_i are bounded so that $X_i \in [a_i, b_i]$, and let $X = \sum_{i=1}^{n} X_i$. Then the following inequality holds.*

$$\Pr(|X - \mathrm{E}[X]| \geq n\epsilon) \leq 2 \exp\left(-\frac{2\, n^2\, \epsilon^2}{\sum_{i=1}^{n} (b_i - a_i)^2}\right)$$

We now use Hoeffding's inequality to build a confidence interval for the power index. Let $C_1, \ldots, C_k$ be a set of k randomly sampled coalitions, and $X_1, \ldots, X_k$ be the series of k

Bernoulli trials, as defined above. Again, let $X = \sum_{j=1}^{k} X_j$, and take $\hat{\beta}_i = \frac{X}{k}$ as an estimator for β_i. All X_i are either 0 or 1 (and are thus bounded between these values), and

$$\mathrm{E}[X] = k \cdot \beta_i$$

Thus, the following holds:

$$\Pr(|X - k\beta_i| \geq k\epsilon) \leq 2e^{-2k\epsilon^2}$$

Therefore the following also holds:

$$\Pr(|\hat{\beta}_i - \beta_i| \geq \epsilon) \leq 2e^{-2k\epsilon^2}$$

We now calculate the required number of samples in order to make sure that this probability is below some required confidence level δ.[4]

Our method generates a 'conservative confidence interval'. Intervals based on Hoeffding's bound are sometimes referred to as 'conservative confidence intervals', since they are based on *exact* bounds rather than on *approximated* bounds, such as the normal approximation for the binomial distribution.

An approach based on the normal approximation for the binomial distribution could be taken here to obtain slightly smaller confidence intervals. However, such a procedure only holds the required confidence level *approximately*. This slight error in the confidence level accumulates when repeating the process, and is thus inappropriate for the ranking procedure presented later in this paper.[5]

We now show how to build the conservative confidence interval, using the Hoeffding inequality.

Theorem 2 (Power Confidence Interval) *For any required accuracy $\epsilon > 0$ and required confidence level $1 - \delta$, we can construct a conservative confidence interval with width 2ϵ of the form:*

$$[\hat{\beta}_i - \epsilon, \hat{\beta}_i + \epsilon]$$

This interval holds the correct Banzhaf index β_i with probability $1 - \delta$. The required number of samples for this is

$$k \geq \frac{\ln \frac{2}{\delta}}{2\epsilon^2}$$

Similarly, given k samples and a required confidence of $1 - \delta$, the following is a conservative confidence interval for β_i, with the required confidence level of $1 - \delta$:

$$\left[\hat{\beta}_i - \sqrt{\frac{1}{2k} \ln \frac{2}{\delta}}, \hat{\beta}_i + \sqrt{\frac{1}{2k} \ln \frac{2}{\delta}}\right]$$

Proof We use the Hoeffding inequality to make sure the error does not exceed our target confidence level δ, and get:

$$Pr(|\hat{\beta}_i - \beta_i| \geq \epsilon) \leq 2\,e^{-2k\epsilon^2} \leq \delta$$

4 Such methods are widely used for the analysis of randomized algorithms in various areas. For example, PAC learning algorithms are typically based on such techniques.

5 Also, it is harder to analyze the asymptotic behavior of the running time of our procedure when using the normal approximation, since it has no direct formula.

We now extract the required ϵ and k:

$$-2\,k\,\epsilon^2 \leq \ln\frac{\delta}{2}$$

This can be restated as:

$$\epsilon^2 \geq -\frac{\ln\frac{\delta}{2}}{2\,k}$$

Finally we get the desired equations, connecting the accuracy, the confidence, and the number of samples:

$$\epsilon \geq \sqrt{\frac{1}{2k}\ \ln\frac{2}{\delta}}$$

$$k \geq \frac{\ln\frac{2}{\delta}}{2\,\epsilon^2}$$

□

We now present an algorithm for building a confidence interval for the power index in simple coalitional games. The algorithm gets the required confidence level δ and interval width w, and returns the desired confidence interval.

Algorithm 1 *ConfidenceBanzhaf(δ, w):*

1. $X = 0, k = 0, \epsilon = \frac{w}{2}$.
2. *Loop until* $k \geq \frac{\ln\frac{2}{\delta}}{2\,\epsilon^2}$.
   (a) *Randomly choose a coalition* C *such that* C *contains agent* i
   (b) $k = k + 1$
   (c) *If* i *is critical in* C *then* $X = X + 1$
3. $\hat{\beta}_i = \frac{X}{k}$
4. *Calculate the confidence interval ConfInterval using Theorem 2.*
5. *Return ConfInterval.*

In certain cases, we may want to compute the power indices of several or even all the agents in a certain game. For example, when ranking agents according to their power index, we need accurate estimates of the power index of all the ranked agents.

Consider a series of m runs of an algorithm for approximating power indices, each of a different agent. Such a series returns m intervals $c_i = [l_i, r_i]$, where each such interval contains β_i with probability $1 - \delta'$.[6]

In order to rank the agents according to their power indices, we can sort the agents according to the intervals' centers. If no two intervals $c_i \neq c_j$ intersect, and if each of them does contain the actual power index of that agent i so that $\beta_i \in c_i$, this results in the correct ranking.

Lemma 2 *Let* $c_1, \ldots, c_m$ *be a series of* m *pairwise non-intersecting confidence intervals for* β_i*'s, each with confidence level of* δ' *as defined above. The probability of having at least one interval that misses a power index is bounded from above by* $m \cdot \delta'$.

[6] We use δ' here rather than δ so as to reserve δ for the required confidence of the ranking procedure.

Proof Each such interval misses its power index with probability of at most δ', so for all i we have:

$$Pr(\beta_i \notin c_i) \leq \delta'$$

We use the union bound to bound the probability of having at least one interval that misses a power index:

$$Pr(\exists i | \beta_i \notin c_i) \leq \sum_{i=1}^{m} P(\beta_i \notin c_i) \leq m \cdot \delta'$$

Thus, if the intervals do not intersect, we have a wrong ranking with probability of at most $\gamma < m \cdot \delta'$. □

Given a target confidence level δ for the ranking procedure, we can now take $\delta' = \frac{\delta}{m}$ as the target confidence level for the single confidence interval. In this case, the error probability would be:

$$\gamma < m \cdot \delta' = \delta$$

Thus, we achieve the desired confidence level for the ranking.

When approximating the power indices of all the agents, we can use the following corollary.

Corollary 1 *When approximating the power index for all the agents in the game, $\beta = (\beta_1, \ldots, \beta_n)$, with a required probability δ of not missing any of the values by more than ϵ, we can approximate each value using Algorithm 1 with confidence level of $\frac{\delta}{n}$, the same ϵ, and with a total of $O(\frac{1}{\epsilon^2} n \ln \frac{n}{\delta})$ samples.*

Proof We simply apply Lemma 2 for the case where $m = n$, the number of all the agents. □

We now consider how to adapt the above procedure for the Shapley–Shubik index as well.

4.4 Adaptations for the Shapley–Shubik power index

The Shapley–Shubik power index can be approximated in a very similar way to that described above for the Banzhaf power index. Both indices measure the probability of an agent being critical, under different coalition formation scenarios. Our algorithm is based on randomly sampling coalitions, and it is possible to change the way these coalitions are sampled so that the result approximates the Shapley–Shubik power index, rather than the Banzhaf power index. In this section we discuss why, given a slightly modified sampling procedure, the exact same confidence intervals and required number of samples of Sect. 4, Theorem 2, and Corollary 1 hold for the Shapley–Shubik power index.

Rather than randomly choosing coalitions, we can randomly choose permutations. The Shapley–Shubik sampling procedure simply returns a random permutation of the agents. There are many known algorithms that produce random permutations, algorithms whose running times are linear in the size of the returned permutation.[7]

[7] See [8] for algorithms that generate permutations.

Similarly to the algorithm for the Banzhaf index, once the random sample is returned, we simply check if agent i is critical in that sample. Agent i is critical in the returned permutation π if:

$$v(S_\pi(i) \cup \{i\}) - v(S_\pi(i)) = 1$$

We denote this as $Critical(i, \pi)$.

After taking several such samples, we estimate $sh_i(v)$ by the proportion of the sampled permutations where agent i is critical. It is easy to see from the definition of the Shapley–Shubik index that the probability that agent i is critical in a random permutation is exactly its Shapley–Shubik power index, so:

$$Pr_{\pi \in \Pi}(Critical(i, \pi)) = sh_i(v)$$

We now define the random variable X_j by letting π_j be a random *permutation*, and X_j being 1 if i is critical in π_j, and being 0 if it is not. We can then continue the process exactly as in Sect. 4.2, and get a procedure for the Shapley–Shubik power index. Theorem 2 and Corollary 1 hold for the Shapley–Shubik power index as well, given this new sampling procedure.

The Shapley–Shubik power index is really the application of the Shapley value in a *simple* coalitional game. It is possible to adapt the procedure for the Shapley value as well, by defining a random variable X_j which is j's marginal addition to a random permutation, and taking proper bounds for the Hoeffding inequality. However, in this case, the bound would depend on the minimal and maximal such marginal contributions to any coalition.[8]

5 Lower bounds for calculating and approximating power indices

From Theorem 2 we see that we can build a confidence interval for the power index, with accuracy of $\epsilon = 1/p(n)$, where $p(n)$ is a polynomial, and with confidence of $1 - \delta$, by taking enough samples. The formula in Theorem 2 states that the number of required samples is

$$O(p(n) \ln(1/\delta))$$

The number of samples is thus polynomial even if δ is exponentially small.

A natural question now is whether we can have a deterministic algorithm that can achieve comparable accuracy with a polynomial number of queries. Another question is whether there is some other randomized algorithm that achieves superpolynomial accuracy, e.g., $\frac{1}{q(n)}$, where $q(n) = \Omega(2^n)$ or even $q(n) = 2^{n^\epsilon}$, for $\epsilon > 0$.

In the rest of this section, we give negative answers to the above questions.[9]

Our results hold for algorithms that have access to a game oracle that returns the game's value for a given coalition. Such an oracle can only answer queries of the form: return the value $v(S)$ of the coalition S. The results do not hold for special restricted cases, where there is a succinct representation of the game.

---

[8] For the Shapley value, an agent's marginal contribution to any coalition is bounded by 0 and the maximal possible value of a coalition. Thus, applying Hoeffding's inequality (Theorem 1) requires a different number of samples, and although the algorithm remains very similar, this can have a large effect on the running time.

[9] The results in the rest of this section are specific to the Banzhaf power index, for ease of analysis.

5.1 Lower bounds for deterministic approximation algorithms

We show that for deterministic algorithms we need an exponential number of queries to achieve polynomial accuracy, in contrast to the randomized case. In fact, this still holds even if we relax the accuracy to linear or even sublinear.

Theorem 3 *There is a constant* $c > 0$ *such that any deterministic algorithm that computes the Banzhaf index with accuracy better than* $c/\sqrt{n}$ *requires* $\Omega(2^n/\sqrt{n})$ *samples.*

Proof Consider a deterministic algorithm $\mathcal{A}$ that always uses less than the stated number of queries. We will use a certain family of instances to derive a contradiction.

Our instances have $n + 1$ players, where n is an even number, and we are interested in computing the index of the $(n + 1)$-st player. We let I_0 denote an instance where $\beta_{n+1} = 0$.

We also define the following family of instances $\mathcal{F}$: consider a coalition S of $\{1, \ldots, n\}$. An instance I belongs to $\mathcal{F}$ if for $|S| < n/2$, $v(S) = 0$ and $v(S \cup \{n+1\}) = 0$. For coalitions S with $|S| = n/2$, $v(S) = 0$ and for exactly half of them, adding $n + 1$ makes it a winning coalition and for the rest not. For $|S| > n/2$, the values are determined by monotonicity, i.e., the value is 1 only if S contains some winning coalition of lower cardinality, otherwise it is 0.

Hence we see that for $I \in \mathcal{F}$, the agent $n+1$ is critical in exactly $\binom{n}{n/2}/2$ coalitions. Thus:

$$\beta_{n+1} = \frac{\binom{n}{n/2}}{2 \cdot 2^n}$$

By using Stirling's approximation, we can see that for large enough n:

$$\binom{n}{n/2} = \Omega(2^n/\sqrt{n})$$

This implies that for all $I \in \mathcal{F}$ we have:

$$\beta_{n+1} = \Omega\left(\frac{1}{\sqrt{n}}\right)$$

Consider now the algorithm $\mathcal{A}$. If it achieves an accuracy better than β_{n+1}, then it should be able to distinguish between the instance I_0 and any instance $I \in \mathcal{F}$.

However there will always be an instance of $\mathcal{F}$ that will make $\mathcal{A}$ fail, if $\mathcal{A}$ asks less than $\binom{n}{n/2}/2$ queries. To see this, look at the queries asked by $\mathcal{A}$ (whether adaptive or not). We can always answer zero to all queries for coalitions of the form $S \cup \{n + 1\}$, with $|S| = n/2$. This can still correspond to a member, I^*, of $\mathcal{F}$, but the algorithm has no way of deducing that $\beta_{n+1} \neq 0$ and hence cannot distinguish between I_0 and I^* (the answers to queries with $|S| < n/2$ or $|S| > n/2$ cannot give any additional information on the index either). □

5.2 Lower bounds for randomized approximation algorithms

We now consider the existence of randomized algorithms that can achieve superpolynomial approximation, e.g., $1/2^{\sqrt{n}}$, or even better $O(1/2^n)$. Below, we answer this question negatively as well for functions that grow faster than polynomials.

This essentially implies that our algorithm of Sect. 4.3 is the best we can hope for in polynomial time.

Theorem 4 *Let δ be a constant less than $1/2$ and let $q(n)$ be any superpolynomial function with $q(n) < 2^n$, i.e., $q(n) = \omega(p(n))$ for any polynomial $p(n)$. Then for any $\epsilon > 0$, no polynomial-time randomized algorithm can build a confidence interval with accuracy better than $1/q(n)$ and confidence level of at least $1 - \delta$.*

Proof We use the standard Minimax principle of Yao [31]: To show a lower bound on a randomized algorithm, it suffices to define a distribution on some family of instances and show a lower bound for a deterministic algorithm on this distribution.

Fix $\epsilon > 0$. The distribution is as follows. As in Theorem 3, we have $n + 1$ players, with n even. Out of all coalitions S of $\{1, \ldots, n\}$ with $|S| = n/2$, we pick uniformly at random $2^n/q(n)$ of them and we make $n + 1$ critical in the corresponding coalitions $S \cup \{n + 1\}$.

This defines a distribution on a family of instances $\mathcal{F}'$. We will have this distribution on $\mathcal{F}'$ with probability 1/2. With the remaining probability of 1/2, we will have the instance I_0. For $I \in \mathcal{F}'$ the Banzhaf index is exactly:

$$\beta_{n+1} = \frac{2^n}{q(n)2^n} = \frac{1}{q(n)}$$

Consider any deterministic algorithm that asks at most $p(n)$ queries for some polynomial $p(n)$. If it achieves accuracy better than $1/q(n)$, it should be able to output a nonzero value for $I \in \mathcal{F}'$ with a constant probability. For this it needs to identify at least one coalition where $n + 1$ is critical.

However, no matter what the queries of the algorithm are, since the critical coalitions were chosen uniformly at random, each query succeeds with probability at most:

$$\frac{2^n}{q(n)\binom{n}{n/2}}$$

Thus, for large enough n, the overall probability of success is at most:

$$\frac{1}{2} + \frac{1}{2}\frac{p(n)2^n}{q(n)\binom{n}{n/2}} \leq \frac{1}{2} + O\left(\frac{p(n)\sqrt{n}}{q(n)}\right) < 1 - \delta$$

□

As an example, we see from the proof above that if we need accuracy better than, say, $1/2^{\sqrt{n}}$, any algorithm would require exponentially many queries. Or more generally:

Corollary 2 *Any randomized algorithm that succeeds with probability higher than $1/2$ needs an exponential number of queries to achieve accuracy better than $1/2^{n^\epsilon}$ for any $\epsilon > 0$.*

The results given in this section are lower bounds on the required amount of information to achieve a desired approximation (specifically, the required number of coalition value queries), and are independent of common complexity-theoretic assumptions, i.e., they hold even if $\mathbf{P} = \mathbf{NP}$.

6 Empirical analysis

We now give results regarding an empirical analysis of our suggested algorithm for approximating the Banzhaf and Shapley–Shubik power indices. Our results were obtained by running the algorithm on a set of simulated weighted voting games. For each such simulated game, we compared the approximation results to the actual power index values (computed using the power index definitions).

We generated random weighted voting games as follows. For each game, the weights of the players were selected randomly, using either the uniform distribution or the normal distribution (with certain parameters). The uniform distribution was over the range $[1, 20]$. The normal distributions used had a mean value of 10 and a standard deviation of 1, 2, or 3. The quota for each game was chosen uniformly from the range $[\frac{1}{4}S, \frac{3}{4}S]$ where S is the sum of all the players' weights (a quota which is too small or too large makes the game uninteresting). The number of players in each game ranged from 10 to 13.[10]

For approximation purposes, each game was assigned a desired accuracy level ϵ (0.025 or 0.05) and a desired confidence level δ (0.025 or 0.05), meaning that the probability that the approximation given by the algorithm will miss the actual value by more than ϵ should be at most δ.

The simulation performed for each game was as follows. First, the actual Banzhaf and Shapley–Shubik power indices of the game were computed exactly. Next, the approximation algorithm was run 10,000 times.[11] For each run, the approximation results were compared to the actual power index values. The *actual accuracy level* was computed as the average difference between the approximate value and the correct value over all players and all runs. We denote by ϵ_b and ϵ_s the actual accuracy levels for the Banzhaf and the Shapley–Shubik power index, respectively. The *actual confidence level* was computed as the fraction of approximate values (over all players and all runs) which differ from the correct value by more than ϵ. We denote by δ_b and δ_s the actual confidence levels for the Banzhaf and the Shapley–Shubik power index, respectively.

6.1 Empirical results

Some sample results can be seen in Table 1 (note that some numbers have been rounded). The actual confidence levels achieved by the approximation algorithm for the games tested were typically several orders of magnitude better than required (and sometimes 100% of the approximations were within the required interval). It is interesting to note that the Shapley–Shubik power index approximations were usually better than the Banzhaf power index approximations.

These results indicate that when using our approximation approach, and its stated number of random samples, the actual accuracy and confidence achieved are typically much better than required. The number of samples our algorithm uses is determined by applying Hoeffding *bounds*, which hold for any distribution. In practice, when the weight distributions are similar to those we simulated, we expect that the number of samples taken by our approximation method will yield much better confidence and accuracy than those requested. Alternatively, when the weight distributions are similar to those we simulated, the number

[10] The computational requirements for exactly computing power indices are enormous even for small games, as the complexity is exponential (or worse) in the number of players. For example, at the time of writing this paper, calculating the Shapley–Shubik power index for a game with 13 players requires about an hour on a standard PC.

[11] Each such run randomly chooses various player coalitions or permutations, as required for the given accuracy and confidence levels, and generates an approximation of the players' power indices based on those samples.

Table 1 Empirical results of the power index approximation algorithm for some sample games

Dist.	Players	ϵ	δ	ϵ_b	δ_b	ϵ_s	δ_s
$U(20)$	10	0.025	0.025	0.005	0.0004	0.004	0.00001
$U(20)$	11	0.025	0.05	0.005	0.002	0.004	0.00005
$U(20)$	12	0.05	0.05	0.008	0.00009	0.008	0.00007
$U(20)$	13	0.05	0.025	0.009	0.0004	0.007	0.000008
$N(10, 1)$	13	0.025	0.05	0.006	0.001	0.004	0
$N(10, 3)$	12	0.025	0.025	0.004	0.00003	0.004	0.00002
$N(10, 1)$	11	0.05	0.025	0.011	0.0005	0.008	0
$N(10, 3)$	10	0.05	0.05	0.011	0.0002	0.009	0.00002

Table 2 Empirical Banzhaf power index approximation results for some sample games using a fraction of the number of samples prescribed by the algorithm

Dist.	Players	ϵ	δ	f	ϵ_b	δ_b
$U(20)$	12	0.025	0.025	0.3125	0.008	0.018
$U(20)$	12	0.05	0.05	0.25	0.017	0.033
$N(10, 1)$	12	0.025	0.025	0.4375	0.008	0.017
$N(10, 3)$	12	0.025	0.025	0.25	0.008	0.012
$N(10, 1)$	12	0.05	0.05	0.25	0.01	0.044
$N(10, 3)$	12	0.05	0.05	0.25	0.01	0.045

of samples required to achieve the requested confidence and accuracy may in fact be much smaller than those taken by our approximation algorithm. We examine this perspective in the next section.

6.2 Empirical estimates of the number of samples

The previous section investigated the actual accuracy and confidence levels obtained when using our approximation approach, and using Theorem 2 to compute the number of required samples for building a conservative confidence interval. We now attempt to empirically determine the number of samples required for achieving given accuracy and confidence levels, rather than using the formula from Theorem 2. In order to do so, we repeated the above simulations using a factor $f < 1$, which indicates the fraction of random samples used out of the total number taken by Algorithm 1, as determined by Theorem 2. In other words, if, for a given accuracy level ϵ and a given confidence level δ, the algorithm normally uses k random samples, we attempt to achieve the required confidence interval using only fk random samples.[12]

Tables 2 and 3 show that confidence intervals with similar quality to those desired can be achieved using only a fraction (about $\frac{1}{4}$) of the number of samples used in our approximation method. These results indicate that when sampling a player coalition (or permutation) is costly, we can typically use many fewer samples, and still obtain good accuracy and confidence. Once again, we see that the algorithm performs slightly better for the Shapley–Shubik power index than for the Banzhaf power index. We believe that the fact that the Shapley–Shubik power index is based on the proportion of all *permutations* (rather than *coalitions*) where

[12] Of course, we have empirically tested only some specific game distributions, and other bounds may be applicable for other game families, whereas the number of samples taken by Algorithm 1 is enough to achieve the desired accuracy and confidence levels for *any* game. However, these results do indicate how “wasteful” our approximation algorithm may be.

Table 3 Empirical Shapley–Shubik power index approximation results for some sample games using a fraction of the number of samples prescribed by the algorithm

Dist.	Players	ϵ	δ	f	ϵ_s	δ_s
$U(20)$	12	0.025	0.025	0.25	0.007	0.011
$U(20)$	12	0.05	0.05	0.1875	0.018	0.041
$N(10, 1)$	12	0.025	0.025	0.1875	0.009	0.021
$N(10, 3)$	12	0.025	0.025	0.1875	0.009	0.022
$N(10, 1)$	12	0.05	0.05	0.1875	0.009	0.02
$N(10, 3)$	12	0.05	0.05	0.1875	0.008	0.024

an agent is critical makes it less "sensitive" to changes in the weights; we believe that this makes our suggested approximation method perform slightly better for this index in terms of the required number of samples.

7 Conclusions and future research

We have suggested algorithms for *approximately* calculating power indices, in any simple coalitional game. The method is suited for both the Banzhaf power index and for the Shapley–Shubik power index. We also believe the method is quite general and can be adapted for other power indices as well.

Our method is probably approximately correct: our procedure returns a *confidence interval* that contains the actual value with high probability. The running time of our method depends on both the accuracy (the desired width of the interval), and the confidence level (the maximal allowed probability of having the true power index outside the interval), and is polynomial in both.

The approximation algorithm we suggested is a randomized algorithm. Although the algorithm is simple, we showed that it performs very well in terms of running time, accuracy, and confidence. We also showed that no deterministic algorithm can achieve comparable accuracy to our algorithm with a polynomial number of queries, and that no randomized algorithm can achieve superpolynomial accuracy. Therefore, our algorithm is close to optimal in this sense. The lower bounds given in this paper are bounds on the required amount of information, and are thus independent of complexity theory assumptions.

The empirical evaluation we carried out to test our algorithm indicates that typically it achieves much better accuracy and confidence levels than those required, for both the Banzhaf and Shapley–Shubik power indices. We have also shown that we can typically achieve the required accuracy and confidence levels with only a fraction of the number of samples used by the algorithm.

There are several directions for future work. First, we note that for *restricted* domains, it might be possible to exactly calculate power indices, or find ways to obtain better approximations. For example, it may be possible to achieve good deterministic algorithms and approximation algorithms with better quality for calculating power indices in restricted weighted voting games or restricted network reliability domains. Second, although our approximation method can be used for both the Banzhaf index and the Shapley–Shubik index, it would be interesting to see if there are domains where one index can be polynomially computed, while the other is hard to compute.

Acknowledgments The second author would like to thank Krzysztof Apt for introducing him to the concepts of power indices and for providing useful pointers to the literature. We would also like to thank Elchanan Mossel for valuable discussions. This work was partially supported by Israel Science Foundation Grant #898/05.

References

1. Abramowitz, M., & Stegun, I. A. (1964). *Handbook of mathematical functions with formulas, graphs, and mathematical tables* (ninth dover printing, tenth gpo printing ed.). New York: Dover.
2. Bachrach, Y., Markakis, V., Procaccia, A. D., Rosenschein, J. S., & Saberi, A. (2008). Approximating power indices. In *The seventh international joint conference on autonomous agents and multiagent systems (AAMAS 2008)*, 12–16 May, Estoril, Portugal (pp. 943–950).
3. Bachrach, Y., & Rosenschein, J. S. (2007). Computing the Banzhaf power index in network flow games. In *The sixth international joint conference on autonomous agents and multiagent systems (AAMAS 2007)*, 14–18 May, Honolulu, Hawaii (pp. 323–329).
4. Banzhaf, J. F. (1965). Weighted voting doesn't work: A mathematical analysis. *Rutgers Law Review, 19*, 317–343.
5. Barua, R., Chakravarty, S. R., & Roy, S. (2007, May). *Measuring power in weighted majority games*. Staff general research papers, Department of Economics, Iowa State University.
6. Bilbao, J. M., Fernandez, J. R., Jimenez, N., & Lopez, J. J. (2002). Voting power in the European Union enlargement. *European Journal of Operational Research, 143*(1), 181–196.
7. Bilbao, J. M., Fernandez, J. R., Losada, A. J., & Lopez, J. J. (2000). Generating functions for computing power indices efficiently. *TOP, 8*(2), 191–213.
8. Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2001). *Introduction to algorithms*. Cambridge, MA: The MIT Press.
9. Deegan, J., & Packel, E. W. (1978). A new index of power for simple n-person games. *International Journal of Game Theory, 7*(2), 113–123.
10. Deng, X., & Papadimitriou, C. H. (1994). On the complexity of cooperative solution concepts. *Mathematics of Operations Research, 19*(2), 257–266.
11. Dubey, P., & Shapley, L. (1979). Mathematical properties of the Banzhaf power index. *Mathematics of Operations Research, 4*(2), 99–131.
12. Faliszewski, P., & Hemaspaandra, L. A. (2008). The complexity of power-index comparison. *CoRR*, abs/0801.4585.
13. Fatima, S. S., Wooldridge, M., & Jennings, N. R. (2007). A randomized method for the Shapley value for the voting game. In *The sixth international joint conference on autonomous agents and multiagent systems (AAMAS 2007)*, 14–18 May, Honolulu, Hawaii (pp. 955–962).
14. Fatima, S. S., Wooldridge, M., & Jennings, N. R. (2008). A linear approximation method for the shapley value. *Artificial Intelligence, 172*(14), 1673–1699.
15. Felsenthal, D., & Machover, M. (2000). The measurement of voting power: Theory and practice, problems and paradoxes. *Public Choice, 102*(3–4), 373–376.
16. Hoeffding, W. (1963). Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association, 58*(301), 13–30.
17. Laruelle, A. (1999). *On the choice of a power index*. Papers 99-10. Valencia: Instituto de Investigaciones Económicas.
18. Leech, D. (2002). An empirical comparison of the performance of classical power indices. *Political Studies, 50*(1), 1–22.
19. Leech, D. (2002). Voting power in the governance of the International Monetary Fund. *Annals of Operations Research, 109*(1–4), 375–397.
20. Leech, D. (2003). Computing power indices for large voting games. *Journal of Management Science, 49*(6), 831–837.
21. Machover, M., & Felsenthal, D. S. (2001). The treaty of Nice and qualified majority voting. *Social Choice and Welfare, 18*(3), 431–464.
22. Mann, I., & Shapley, L. S. (1960). Values of large games, IV: Evaluating the Electoral College by Monte-Carlo techniques. Technical report, The Rand Corporation, Santa Monica, CA.
23. Mann, I., & Shapley, L. S. (1962). Values of large games, VI: Evaluating the Electoral College exactly. Technical report, The Rand Corporation, Santa Monica, CA.
24. Matsui, Y., & Matsui, T. (2000). A survey of algorithms for calculating power indices of weighted majority games. *Journal of the Operations Research Society of Japan, 43*(1), 71–86.

25. Matsui, Y., & Matsui, T. (2001). NP-completeness for calculating power indices of weighted majority games. *Theoretical Computer Science, 263*(1–2), 305–310.
26. Owen, G. (1975). Multilinear extensions and the Banzhaf value. *Naval Research Logistics Quarterly, 22*(4), 741–750.
27. Prasad, K., & Kelly, J. (1990). NP-completeness of some problems concerning voting games. *GAMETHEORY: International Journal of Game Theory, 19*, 1–9.
28. Shapley, L. S. (1953). A value for n-person games. In H. W. Kuhn & A. W. Tucker (Eds.), *Contributions to the theory of games* (pp. 31–40). Princeton, NJ: Princeton University Press.
29. Shapley, L. S., & Shubik, M. (1954). A method for evaluating the distribution of power in a committee system. *American Political Science Review, 48*, 787–792.
30. Straffin, P. (1977). Homogeneity, independence and power indices. *Public Choice, 30*, 107–118.
31. Yao, A. C.-C. (1977). Probabilistic computations: Toward a unified measure of complexity. In *Proceedings of the 18th annual symposium on foundations of computer science (FOCS)*, Providence, RI (pp. 222–227).
32. Young, H. P. (1994). Cost allocation. In R. J. Aumann & S. Hart (Eds.), *The handbook of game theory* (Vol. II, pp. 1193–1235). Amsterdam: North-Holland.