Computation of voting power indices using polynomial rings and ideals

Alasdair McAndrew
Alasdair.McAndrew@vu.edu.au
College of Engineering and Science
Victoria University
PO Box 14421, Melbourne, Victoria 8001
Australia

Abstract

In many jurisdictions globally, voting is done not by individuals, but by blocks of voters. Examples are the American Electoral College, the International Monetary Fund, the European Parliament, and the houses of parliament or of congress in many national legislatures. Voting is thus done by blocks: state, country, or political party; each member of which casts the same vote. It is tempting to assume that a block with the most votes is the most powerful, and has the greatest chance of influencing the outcome of the final ballot. As with so much else in voting theory, this is quite incorrect, and it is often the case that a minor block can have an influence entirely out of keeping with its size. There are various different methods of determining the relative power of each block, and allocating a numerical measure of power; these values are called "power indices". Some of these methods can be computationally intensive, and it took several years after the initial definition to compute power indices for the American Electoral College, with its 50 states. In this paper we explore a unified method for computing different power indices using the theory of polynomial rings. This allows not only a relatively simple computation, but one which can be adjusted to consider "coalitions" (two or more blocks which band together to increase their power), and "quarreling parties", where two blocks refuse to agree on any vote.

1 Introduction

Although power indices had first been defined and discussed in the mid 1940's [8], it was really a paper in 1965 which started to bring the notion into prominence. The lawyer John Banzhaf [1] was asked to mediate in a discussion of the fairness of voting allocation in Nassau County, New York state, which had been made as shown in table 1.

There were thus 30 votes in total, of which 16 (a majority) were required for any motion to pass. To determine if power was correlated with votes, Banzhaf made a list of all possible combinations which could win; that is, sum to 16 or greater. In each such combination, he then determined if a particular block was necessary for that win.

Municipality	Number of votes
Hempstead (No. 1):	9
Hempstead (No. 2):	9
North Hempstead:	7
Oyster Bay:	3
Glen Cove:	1
Long Beach:	1

Table 1: Nassau County voting allocations

For example, one winning combination would be Hempstead (No. 1), Hempstead (No. 2), Oyster Bay, and Glen Cove, for a total of 22 votes. However, both the Hempsteads are necessary for this combination, because if either withdrew their support, the votes would drop from 22 to 13. And neither Oyster Bay or Glen Cove are necessary, since if either or both withdrew their support, the total votes would drop to no lower than 18, which is still enough for a win. A necessary voting block in a winning combination is also called a "swinging" block.

Banzhaf identified 32 different winning combinations, and with them 48 different possible swings, with numbers as shown in table 2.

Municipality	Number of swings
Hempstead (No. 1):	16
Hempstead (No. 2):	16
North Hempstead:	16
Oyster Bay:	0
Glen Cove:	0
Long Beach:	0

Table 2: Nassau County necessary votes

It can be seen then that in spite of their different voting numbers, Hempstead 1, Hempstead 2 and North Hempstead are all exactly equal in their abilities to influence a vote; conversely Oyster Bay, Glen Cove and Long Beach have *no power at all*: it doesn't matter how they vote, because in any combination which reaches a winning value, none of their votes are actually necessary.

The conclusion was that the original voting allocation was manifestly unfair, and some other allocation was required so that the smaller blocks had non-zero power. As Banzhaf said in the article: "It is hard to conceive of any theory of representative government which could justify a system under which the representatives of three of the six municipalities "represented" are allowed to attend meetings and cast votes, but are unable to have any effect on legislative decisions. Yet this is exactly what occurs now in Nassau County."

Although Banzhaf had in a sense reinvented a method original proposed by Lionel Penrose (and that had gone completely unnoticed), it was again reinvented in 1971 by James Coleman [3]. These indices are thus known as the Banzhaf, or the Banzhaf-Coleman, or the Penrose-Banzhaf-Coleman, power indices. In this article the term *Banzhaf power index* will be used.

Preceding Banzhaf by over 10 years, in 1954 the economists Lloyd Shapley (who would win the Nobel Prize in 2012) and Martin Shubik developed a power index now known by their names. To compute the power index, every possible permutation of the voting blocks is listed, and reading from left to right, a block is said to be *pivotal* if it changes a hitherto losing combination to a winning one. For instance, with the Nassau County municipalities, one of the $6! = 720$ permutations is given in table 3.

Municipality	Votes	Cumulative Sum
Glen Cove:	1	1
North Hempstead:	7	8
Hempstead (No. 2):	9	17
Oyster Bay:	3	20
Long Beach:	1	21
Hempstead (No. 1):	9	30

Table 3: A permutation with its pivotal block

We see that in this permutation, Hempstead 2 is pivotal. The Shapley-Shubik index counts the number of times each block has been pivotal. (There are better ways of computing this than enumerating all permutations.) For this example they can be computed to be $(240, 240, 240, 0, 0, 0)$; so as for the Banzhaf indices: the top three municipalities have equal power, and the lower three have no power. We note that by definition the sum of these indices must be $n!$, where n is the number of voting blocks.

It is more convenient to scale indices so that their sum is one; in this instance then both the Banzhaf and Shapley-Shubik indices return $(1/3, 1/3, 1/3, 0, 0, 0)$. Although the two indices can agree, they are generally different.

2 Formal definitions

A *weighted voting game* consists of a sequence of weights w_i for $1 \leq i \leq n$ and a *quota* q. The quota represents the number of votes needed for a win, hence it must be less than or equal to the sum of all the weights. Such a game is notated as

$$[q; w_1, w_2, \ldots, w_n].$$

The Nassau County issue discussed in the introduction would thus be notated as

$$[16; 9, 9, 7, 3, 1, 1].$$

A *coalition* is any non-empty set of weights, and a *winning coalition* is a coalition whose weights sum to q or more. In a winning coalition, voter i is *pivotal* or *necessary* if the removal of i from the coalition reduces the sum to below the quota.

2.1 Computation of power indices

We have seen in the introduction that the Banzhaf power indices require consideration of all subsets of voters, and the Shapley-Shubik power indices require consideration of all permutations of voters. In either case, the computation is exponential in terms of the number of voters. Efficient computation is therefore a major consideration. For example, if we consider the American Electoral College as a weighted voting game with 51 voters (50 states plus DC) where the weights are the number of electors in each state[1], we would require $2^{51} = 2,251,799,813,685,248$ computations, and for the Shapley-Shubik indices we would need $51! \approx 1.55 \times 10^{66}$ computations.

Banzhaf power indices

We can of course compute these by simply enumerating each subset $S \subseteq \{1, 2, 3, \ldots, n\}$ and if it is a winning coalition, determining which voter is necessary. But a neater polynomial method was developed by Brams and Affuso in 1976 [2], and for demonstration we shall consider the voting game

$$[39; 34, 33, 7, 1, 1].$$

This games represents the Australian Federal Senate, or upper house, in 1985. The parties are Labor (34), Liberal/National (33), Democrats (7), Nuclear Disarmament (1), Independent (1).

To determine the power of voter k, first create the formal polynomial

$$p_k(x) = \prod_{\substack{i=1 \\ i \neq k}}^{n} (1 + x^{w_i})$$

so that for the first voter above,

$$p_1(x) = (1 + x^{33})(1 + x^7)(1 + x)(1 + x).$$

Then the coefficient of x^j in this polynomial is the number of ways all the other voters can combine to form a coalition with j votes. Expanding the polynomial:

$$p_1(x) = x^{42} + 2x^{41} + x^{40} + x^{35} + 2x^{34} + x^{33} + x^9 + 2x^8 + x^7 + x^2 + 2x + 1$$

shows that there are two ways, for example, of obtaining a total of 41 votes (voters 2, 3, and 4, or voters 2, 3, and 5). For the votes already above the quotient of 39, the addition of the new voter won't make any difference; the voter cannot be necessary. Voter k becomes necessary only for those votes with a sum less than the quotient q, but not less than $q - w_k$. The latter restriction is necessary because if a sum is less than $q - w_k$, then the addition of the weight w_k cannot produce a sum equal to or greater than the quotient. This means that the number we want is the sum of coefficients of all powers x^m for which $q - w_k \leq m < q$, so that the k-th Banzhaf index, b_k can be computed as

$$b_k = \sum_{j=q-w_k}^{q-1} c_j$$

where c_j is the coefficient of x_j in the polynomial.

In our example, given $p_1(x)$ above, we look at coefficients of powers x^m for which $39 - 34 \leq m < 39$:

$$x^{35} + 2x^{34} + x^{33} + x^9 + 2x^8 + x^7$$

and there are six such terms with coefficients summing to 8. Thus the Banzhaf power index for the first voter is 8.

This can be easily implemented in Python using the SymPy library as shown in Listing 1.

[1]The situation is slightly muddled in actuality in that two states: Maine and Nebraska, allocate their electoral votes at least partially according to the popular vote. In all other states it's "winner take all".

```
> import sympy as sy
> x = sy.Symbols('x')
> def banzhaf(q,w):
      for k in range(n):
          qe = sy.prod([(1+x**w[i]) for i in range(n) if i != k]).expand()
          b[k] = sum(qe.coeff(x,m) for m in range(q-w[k],q))
      return(b)

> banzhaf(39,[34,33,7,1,1])
  [8,8,8,0,0]
```

Listing 1: Computing the Banzhaf power indices using polynomials

The output, showing the Banzhaf indices, indicate that the three largest parties in the Senate have equivalent power, in spite of one being very much smaller than the other[2], and the two small parties have no power at all. As mentioned above, the "raw" indices can be normalized to sum to one.

Shapley-Shubik indices

The use of polynomials here predates the work of Brams and Affuso; in fact Shapley himself, working with Irwin Mann, produced a polynomial computation in 1962 [7]. However, before introducing the polynomials, we shall see how the Shapley-Shubik indices can be determined by a method very similar to the Banzhaf indices. Suppose that voter k is necessary in a coalition S, being a subset of $V = \{1, 2, 3, \ldots, n\}$. Consider the two sets $S_k = S - \{k\}$ and $V - S$. Then k is pivotal in any permutation of the form:

$$[\text{any permutation of the elements of } S_k], k, [\text{any permutation of } V - S].$$

If S has r members, then the number of such permutations is $(r-1)!(n-r)!$. This means that the Shapley-Shubik index for a voter k can be calculated as

$$\sum_S (|S| - 1)!(n - |S|)!$$

where the sum is taken over all coalitions in which k is necessary. For the Australian Senate example, the first voter is necessary to the coalitions

$$\{1,2\}, \quad \{1,3\}, \quad \{1,2,4\}, \quad \{1,2,5\}, \quad \{1,2,4,5\}, \quad \{1,3,4\}, \quad \{1,3,5\}, \quad \{1,3,4,5\}$$

and hence its power index will be

$$1!3! + 1!3! + 2!2! + 2!2! + 3!1! + 2!2! + 2!2! + 3!1! = 40.$$

[2]This smaller party, the "Australian Democrats" was created precisely for this purpose, to maintain a "balance of power in the Senate" and in the words of its founder: "to keep the bastards honest". In spite of small numbers, the Democrats were a significant player in Australian politics for over a decade.

To use polynomials here, we need—as well as knowing when a voter k is necessary for a coalition—to know the size of the coalition. This is easily done by introducing a new variable y which will act as a counter:

$$p_k(x) = \prod_{\substack{i=1 \\ i \neq k}}^{n} (1 + x^{w_i} y).$$

For the first voter in the Senate example,

$$\begin{aligned} p_1(x, y) &= (1 + x^{33}y)(1 + x^7 y)(1 + xy)(1 + xy) \\ &= x^{42}y^4 + 2x^{41}y^3 + x^{40}y^2 + x^{35}y^3 + 2x^{34}y^2 + x^{33}y + x^9 y^3 + 2x^8 y^2 \\ &\quad + x^7 y + x^2 y^2 + 2xy + 1. \end{aligned}$$

This shows, for example, that a coalition with combined weight 41 can be obtained with 3 voters in 2 ways. For the Banzhaf example, we knew that coalitions of weight 41 could be obtained in two ways, but the polynomial did not include the information about the numbers of voters.

As with the Banzhaf indices, to obtain a coalition in which k is necessary, we add the coefficients of $x^i y^j$ for which $q - w_k \leq i < q$. For $p_1(x, y)$ above, this produces

$$x^{35}y^3 + 2x^{34}y^2 + x^{33}y + x^9 y^3 + 2x^8 y^2 + x^7 y$$

and adding the coefficients of the powers of x produces

$$2y^3 + 4y^2 + 2y.$$

In this last polynomial in y, the powers are the sizes of the winning coalitions.

This means that the Banzhaf procedure given in Listing 1 can be used as the basis for a very similar procedure for computing the Shapley-Shubik indices; this is given in Listing 2.

This procedure can be easily adjusted to return the normalized values of $[1/3, 1/3, 1/3, 0, 0]$.

2.2 Deegan-Packel and Holler indices

More recently, some other power indices have been proposed. In 1978, Deegan and Packel [4] proposed an index based on *minimal winning coalitions*, abbreviated as MWCs, which are coalitions in which every party is critical. For example, with $w = [15, 12, 7, 4, 3, 2]$ and $q = 22$, then $(15, 4, 3)$ is a minimal winning coalition, as is $(12, 7, 3)$. However, $(15, 12, 4)$ is a winning coalition, but it is not minimal, since 4 is not needed. Let W be the set of all such minimal winning coalitions, and let $W_i \subset W$ be those that contain voter i. Then the Deegan-Packel power index is defined as

$$d_i = \sum_{S \in W_i} \frac{1}{|S|}.$$

For example, with the Nassau County example, the MWCs are

$$(9_1, 9_2), \quad (9_1, 7), \quad (9_2, 7).$$

```
>  import sympy as sy
>  x = sy.Symbols('x')
>  def shapley_shubik(q,w):
     n = len(w)
     ss = [0]*n
     for k in range(n):
         pe = sy.prod([(1+x**w[i]*y) for i in range(n) if i != k]).expand()
         py = sum(pe.coeff(x,m) for m in range(q-w[k],q))
         ss[k] = sum(sy.factorial(m)*sy.factorial(n-1-m)*py.coeff(y,m)\
                  for m in range(n))
     return(ss)

>  ss(39,[34,33,7,1,1])
   [40,40,40,0,0]
```

Listing 2: Computing the Shapley-Shubik power indices using polynomials

where we have distinguished the two largest voters with subscripts. Since each voter is a member of exactly two of these three coalitions, each one has the index $1/2 + 1/2 = 1$. As with the Banzhaf and Shapley-Shubik indices, the top three voters have equal power; the lower three none at all.

Using MWCs only accords with *Riker's size principle*, that "parties seek to increase votes only up to the size of a minimum coalition" [6]. This makes political sense, and hence winning coalitions that include non-critical parties may be considered as irrelevant to voting power (if the size principle is assumed).

Holler's public good index [6] is obtained by normalizing the values of $|W_i|$ for each voter. Since these values are $(2, 2, 2, 0, 0, 0)$, then the public good indices are $(1/3, 1/3, 1/3, 0, 0, 0)$ as they are for all the other indices.

Although the different indices return the same values for the Nassau County example, this is not normally the case. For example, with $w = [28, 16, 5, 4, 3, 3]$ and $q = 30$, the various normalized indices are:

$$\begin{aligned}
\text{Banzhaf} : \left[\frac{3}{4}, \frac{1}{20}, \frac{1}{20}, \frac{1}{20}, \frac{1}{20}, \frac{1}{20}\right] &\approx [0.75, 0.05, 0.05, 0.05, 0.05] \\
\text{Shapley-Shubik} : \left[\frac{2}{3}, \frac{1}{15}, \frac{1}{15}, \frac{1}{15}, \frac{1}{15}, \frac{1}{15}\right] &\approx [0.67, 0.067, 0.067, 0.067, 0.067, 0.067] \\
\text{Deegan-Packel} : \left[\frac{5}{12}, \frac{7}{60}, \frac{7}{60}, \frac{7}{60}, \frac{7}{60}, \frac{7}{60}\right] &\approx [0.42, 0.117, 0.117, 0.117, 0.117, 0.117] \\
\text{Holler} : \left[\frac{1}{3}, \frac{2}{15}, \frac{2}{15}, \frac{2}{15}, \frac{2}{15}, \frac{2}{15}\right] &\approx [0.33, 0.133, 0.133, 0.133, 0.133, 0.133]
\end{aligned}$$

Although all indices give the same smaller value to the five smaller parties, the relative weightings are different.

```
> q = 30
> w = [28,16,5,4,3,3]
> n = len(w)
> xs = sy.var('x0:%d'%n)
> pr = sy.prod([1+xs[i]**w[i] for i in range(n)]).expand()
> pa = pr.args
> pw = [x for x in pa[1:] if sum(sy.degree_list(x)) >= q]
```

Listing 3: Using multivariate polynomials

The Deegan-Packel indices are *non-monotonic*; in that it is possible for a smaller party to have a greater index. For example (the values are approximate only):

$$[51; 35, 20, 15, 15, 15] \Rightarrow (1.5, 0.75, 0.92, 0.92, 0.92)$$

and we see that a party of size 15 is assigned a *greater* voting power than a party of size 20. Holler's public good index conforms with Riker's size principle, but is also non-monotonic. The difference between the Deegan-Packel and Holler indices is that Deegan-Packel gives higher weight to coalitions with smaller numbers: a party that can influence an MWC of two members only is seen to have greater power than one which requires three or more members before it can be critical; Holler simply counts the number of MWCs for which a party is critical.

3 Working with multivariate polynomials

In order to deal with "coalitions" (when two or more parties join together), or with "quarreling" parties (who never agree), it will be convenient to use a different symbol for each party. Listing 3 shows how this can be done using SymPy.

In this script, the fourth line creates a list (named "`xs`") of the variables

$$x_0,\ x_1,\ \dots, x_{n-1}$$

and the next line creates the polynomial

$$\mathtt{pr} = \prod_{i=1}^{n-1}(1 + x_i^{w_i}).$$

The next two lines first break the polynomial up into its monomials, and then selects those for which the degree sum is not less than q. In this case there are 28 of them. These monomials correspond to all the winning coalitions. We shall call this polynomial the *encoding polynomial* of the winning coalitions.

To obtain the *minimal* winning coalitions, what we need to do now is to sieve out all those monomials which are multiples of another one, and we can write a recursive function to do this.

The idea is that at the beginning of the sieving routine, t is an empty list, and p consists of all the monomials. As the function works through its iterations, t picks up the "lowest" values from p, while p is reduced by multiples of the elements of t. For example:

```
def sieve(t,p):
    if len(p)==0:
        return(t)
    else:
        for x in p[1:]:
            if sy.rem(x,p[0]) == 0:
                p.remove(x)
        return(sieve(t+[p[0]],p[1:]))
```

Listing 4: A sieve function to remove polynomial multiples

```
>  pw2 = pw.copy()
>  pws = sieve([],pw2)
>  pws
```

$$[x_0^{28}x_1^{16},\ x_0^{28}x_2^5,\ x_0^{28}x_3^4,\ x_0^{28}x_4^3,\ x_0^{28}x_5^3,\ x_1^{16}x_2^5x_3^4x_4^3x_5^3]$$

The sum of these monomials will be a polynomial which, as before, can be referred to as the encoding polynomial of the MWCs. To obtain (for example) the Deegan-Packel power indices, we can work with in dictionary whose keys are the variables, and whose values will be increased by $1/k$ where k is the number of variables in each monomial:

```
>  dp = {xs[i]:0 for i in range(n)}
>  for p in pws:
       pv = p.free_symbols
       pm = len(pv)
       for x in pv:
           dp[x] += sy.Rational(1,pm)
>  dp
```

$$\left\{x_0 : \frac{5}{2},\ x_1 : \frac{7}{10},\ x_2 : \frac{7}{10},\ x_3 : \frac{7}{10},\ x_4 : \frac{7}{10},\ x_5 : \frac{7}{10}\right\}$$

and this can be normalized to add to one:

```
>  dpn = {xs[i]:0 for i in range(n)}
>  s = sum(dp.values())
>  for z in dpn.keys():
       dpn[z] = dp[z]/s
>  dpn
```

$$\left\{x_0 : \frac{5}{12},\ x_1 : \frac{7}{60},\ x_2 : \frac{7}{60},\ x_3 : \frac{7}{60},\ x_4 : \frac{7}{60},\ x_5 : \frac{7}{60}\right\}$$

As a proof-of-concept of the use of such polynomials, we'll consider the Banzhaf power indices for $[16; 10, 9, 6, 5]$. We start as above by creating four variables x_i and computing the product

```
>  q, w = 16, [10,9,6,5]
>  p = \prod_{i=0}^3(1+x_i^{w_i}).
```

This is done exactly as in Listing 3 and the polynomial returned is

$$x_0^{10}x_1^9x_2^6x_3^5 + x_0^{10}x_1^9x_2^6 + x_0^{10}x_1^9x_3^5 + x_0^{10}x_1^9 + x_0^{10}x_2^6x_3^5 + x_0^{10}x_2^6 + x_0^{10}x_3^5 + x_0^{10} + x_1^9x_2^6x_3^5 + x_1^9x_2^6 + x_1^9x_3^5 + x_1^9 + x_2^6x_3^5 + x_2^6 + x_3^5 + 1$$

To find the i-th power index, we reduce the polynomial modulo $x_i^{w_i}$, and add up the coefficients of those monomials whose degree sum is between $q - w_i$ and q. The reduction can be done either with a polynomial division or using Groebner bases; for this simple operation a division is adequate:

```
>  i = 0
>  quo, rem = sy.div(p,xs[i]**w[i])
```

and the remainder (which is what we want) is

$$x_1^9x_2^6x_3^5 + x_1^9x_2^6 + x_1^9x_3^5 + x_1^9 + x_2^6x_3^5 + x_2^6 + x_3^5 + 1.$$

We can find the coefficients and degree sums, and add the appropriate coefficients:

```
>  m = list(rem.args)
>  cs = [Poly(rem).coeff_monomial(x) for x in m]
>  ds = [1] + [sym(sy.degree_list*x( for x in m[1:]]
>  sum(x for x,y in zip(cs,ds) if (y >= q-w[i]) and (y < q))
```

These last scripts can be placed in a loop to run through all values of i.

Clearly this method is overkill for the computation of the Banzhaf indices: as we have seen earlier working with a univariate polynomial is quite sufficient. However, that polynomial cannot indicate which parties have met to form each coalition.

With the voting game $[16; 10, 9, 6, 5]$ the products of all of $(1 + x^{w_i})$ except for each term in turn are:

$$\begin{aligned}
&x^{20} + x^{15} + x^{14} + x^{11} + x^9 + x^6 + x^5 + 1\\
&x^{21} + x^{16} + x^{15} + x^{11} + x^{10} + x^6 + x^5 + 1\\
&x^{24} + x^{19} + x^{15} + x^{14} + x^{10} + x^9 + x^5 + 1\\
&x^{25} + x^{19} + x^{16} + x^{15} + x^{10} + x^9 + x^6 + 1
\end{aligned}$$

In the i-th polynomial, we then have to see which of the terms when multiplied by x^{w_i} will push the degree from under to at or over the quota q.

Using different variables, we can first find the product of $(1 + x_i^{w_i})$ and eliminate all terms with a degree sum less than q. This leads to the encoding polynomial:

$$x_0^{10}x_1^9x_2^6x_3^5 + x_0^{10}x_1^9x_2^6 + x_0^{10}x_1^9x_3^5 + x_0^{10}x_1^9 + x_0^{10}x_2^6x_3^5 + x_0^{10}x_2^6 + x_1^9x_2^6x_3^5$$

and we can see immediately which parties have met to form which coalition. The encoding polynomial thus does encode all the individual winning coalitions.

An alternative approach is to start with the encoding polynomial, and for the i-th party, take the quotient when divided by $x_i^{w_i}$. The number of monomials in the quotient with degree sum less than q will be the i-th Banzhaf power index.

4 Quarreling parties

These are parties who—either temporarily or permanently—refuse to vote together. This might be a matter of principle, or of irreconcilable political differences. If the j-th and k-parties are quarreling, then the polynomial $\prod(1+x_i^{w_i})$ must be reduced modulo x_jx_k. In other words, we remove all monomials which include both x_j and x_k.

As an example, we consider the 2021 composition of the Australian Federal Senate:

	Party	Support & Ideology	Numbers
Coalition	Liberal	Business & economy, right wing	31
	National	Primary Producers, right wing	5
Opposition	Labor	Workers, centre left	26
Cross-bench	Greens	Environment & sustainability, left	9
	One Nation	Anti-immigration, far right	2
	Centre Alliance	Centrist	1
	Lambie Network	Populist	1
	Patrick Team	Regional	1

Table 4: Australian Federal Senate, 2021

Confusingly, the Liberal[3] and National parties are separate political entities, but for the purposes of obtaining the numbers needed to form a government, have an alliance known as "The Coalition". And "cross-bench" simply means that the senators in those parties are not bound by coalition or opposition party lines, but can vote according to their consciences. As of late 2021, the Coalition are the governing body in Australia.

As a majority is required to pass any motion, the voting game is thus

$$[39; 31, 5, 26, 9, 2, 1, 1, 1]$$

but given the Liberal-National coalition, this is better expressed as

$$[39; 36, 26, 9, 2, 1, 1, 1]$$

The Banzhaf, Shapley-Shubik, Deegan-Packel indices with their normalizations, and the Holler index, are respectively:

$$[52, 12, 12, 10, 4, 4, 4] \Rightarrow [0.531, 0.122, 0.122, 0.102, 0.041, 0.041, 0.041]$$

$$[2616, 684, 684, 516, 180, 180, 180] \Rightarrow [0.519, 0.136, 0.136, 0.102, 0.036, 0.036, 0.036]$$

$$\left[\frac{9}{4}, \frac{11}{10}, \frac{11}{10}, \frac{8}{5}, \frac{59}{60}, \frac{59}{60}, \frac{59}{60}\right] \Rightarrow [0.25, 0.122, 0.122, 0.178, 0.109, 0.109, 0.109]$$

$$\frac{1}{16}[3, 2, 2, 3, 2, 2, 2] = [0.1875, 0.125, 0.125, 0.1875, 0.125, 0.125, 0.125]$$

Suppose that Labor and the Coalition are quarrelling; and will not vote together. We can encode this by determining the encoding polynomial modulo x_0x_1. We can start by creating the polynomial $\prod(1+x_i^{w_i})$ as shown in Listing 3, and from that obtain the encoding polynomial:

[3]Note that "Liberal" here has a very different meaning to that in America; Australia's Liberal Party is a right-wing conservative party.

```
bsq = [0]*n
for i in range(n):
    Gi = sy.groebner([xs[i]**ws[i]],xs)
    quo_i, rem_i = Gi.reduce(rem)
    mn = Poly(quo_i[0],xs).monoms()
    bsq[i] = len([x for x in qi if sum(x) < qs])
```

Listing 5: Computing the Banzhaf indices in the case of a quarrel

```
> pmn = Poly(pr,xs).monoms()
> pmnw = [x for x in pmn if sum(x) >= qs]
> p_enc = sum([sy.prod(x**y for x,y in zip(xs,p)) for p in pmnw])
```

and then reduce it:

```
> Gs = sy.groebner([xs[0]*xs[1]],xs1)
> quo,rem = Gs.reduce(p_enc)
> display(rem)
```

$$x_0^{36}x_2^9x_3^2x_4x_5x_6 + x_0^{36}x_2^9x_3^2x_4x_5 + \boxed{\text{24 terms omitted}} + x_1^{26}x_2^9x_3^2x_4x_6 + x_1^{26}x_2^9x_3^2x_5x_6$$

We can now compute the Banzhaf power indices by seeing to which winning coalitions party i is necessary. We can do this by finding the quotient modulo $x_i^{w_i}$ and determining the number of monomials whose degree sum is less than q. This is shown in Listing 5 and the index list, with its normalization, is:

$$[24, 4, 12, 10, 4, 4, 4] \Rightarrow [0.387, 0.065, 0.194, 0.161, 0.065, 0.065, 0.065]$$

Comparing these values with the previous (non-quarreling results) we see the Liberal/Nationals (the first term) have decreased in power slightly, Labor (the second term) has decreased in power drastically, and the loss is distributed among the other parties. The lesson here is that it would be very unwise for Labor to adopt a permanent quarrelling stance with the Coalition.

Suppose that the Greens and One Nation are quarrelling (which is a very reasonable assumption, given their respective ideologies); this requires reducing modulo x_2x_3. The resulting encoding polynomial has 40 terms, and the Banzhaf indices can be found to be

$$[40, 8, 7, 6, 2, 2, 2] \Rightarrow [0.597, 0.119, 0.104, 0.09, 0.03, 0.03, 0.03]$$

and so this particular quarrel has the effect of increasing the power of the Coalition.

A diagram of the quarrels and their effects on the normalized Banzhaf indices is given in Figure 1. One counter-intuitive result is that a quarrel may increase the power of the non-quarreling parties.

Clearly the method outlined can be used for multiple simultaneous quarrels.

5 Conclusions

Assessing the power of a voting body in weighted voting is a fundamental aspect of modern decision making; we expect and assume that voting power will be roughly proportional to the

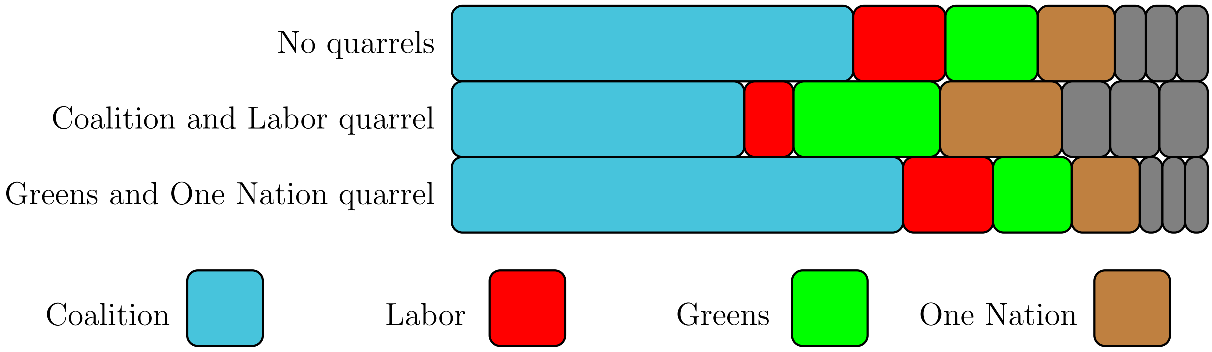

Figure 1: Effects of quarreling

voter's weight. However, this is rarely the case. Over 60 years of investigation into power indices have resulted in many different methods of assigning power, each with particular strengths and weaknesses. And there is now a sizable literature into axiomatic power indexing: what are the basic axioms we would expect a power index to satisfy, and can they be made consistent? And if not, what properties are we prepared to forsake? As we have seen, monotonicity may be a desirable property, which is not satisfied by either the Deegan-Packel or Holler indices. Power is thus a vital topic in decision theory, and such relatively recent upheavals as Brexit may require voting weights to be reassigned so that no country increases power at the cost of another [5].

This means that the computation of voting power indices is also an important topic, and polynomials and various combinatorial algorithms are still popular. We have shown how standard polynomial methods can be enlarged and deepened, using the theory of ideals, can provide a highly general approach that can also deal with quarrels.

Acknowledgements

The author gratefully acknowledges the close reading of the reviewers, who pointed out a number of errors: factual, typographical, and mathematical, (hopefully all of) which have been amended.

References

[1] John Banzhaf. "Weighted voting doesn't work: a mathematical analysis". In: *Rutgers Law Review* 19.2 (1964), pp. 317–343.

[2] Steven J. Brams and Paul J. Affuso. "Power and size: a new paradox". In: *Theory and Decision* 7 (1976), pp. 29–56.

[3] James S. Coleman. "Control of Collectives and the Power of a Collectivity to Act". In: *Social Choice*. Ed. by Bernhardt Liebermann. Gordon and Breach, 1971, pp. 192–225.

[4] John Deegan and Edward W. Packel. "A new index of power for simple n-person games". In: *International Journal of Game Theory* 7.2 (1978), pp. 113–123.

[5] Philip Grech. "Power in the Council of the EU: organizing theory, a new index, and Brexit". In: *Social Choice and Welfare* 56.2 (2021), pp. 223–258.

[6] Manfred J. Holler. “Forming coalitions and measuring voting power”. In: *Political studies* 30.2 (1982), pp. 262–271.

[7] Irwin Mann and Lloyd S. Shapley. *Values of Large Games, VI: Evaluating the Electoral College Exactly*. Tech. rep. United States Air Force Project Rand Memorandum PM-3158-PR, May 1962.

[8] Lionel Penrose. “The Elementary Statistics of Majority Voting”. In: *Journal of the Royal Statistical Society* 109.1 (1946), pp. 53–57.