

Department of Mathematics

A new method for optimal proportional representation

Kaj Holmberg

LiTH-MAT-R--2019/01--SE

A new method for optimal proportional representation

Kaj Holmberg

Department of Mathematics
Linköping Institute of Technology
SE-581 83 Linköping, Sweden

February 6, 2019

Abstract

In a democratic proportional election system, it is vital that the mandates in the parliament are allocated as proportionally as possible to the number of votes the parties got in the election. We formulate an optimization model for allocation of seats in a parliament so as to minimize the disproportionality. By applying separable programming techniques, we obtain an easily solvable problem, and present a method for solving it optimally. The obtained solution is thus the feasible solution that has the minimal disproportionality (with the measure chosen), in contrast to the heuristic procedures used in many countries. We apply the approach to real life data from the last three elections in Sweden, and show that the result is better, i.e. more proportional, than what was obtained with the “adjusted odd number rule”, which is presently used. A natural suggestion would be to use our method instead.

We also consider the issue about constituencies, and suggest a procedure, based on the same kind of optimization problem, for allocating mandates in the constituencies, without changing the overall allocation with respect to parties. In our approach, the numbers of mandates for the constituencies are based on the number of votes given, not on estimated numbers of inhabitants. This removes the need for fixed and equalization mandates, and also makes the question about sizes of the constituencies less important.

Key words: *Democracy, proportional representation, the adjusted odd number rule.*

1 Introduction

In a representative democracy, the rulers of a country are appointed with general elections. In a proportional electoral system, each party shall receive a number of mandates (places in the parliament) that is proportional to the number of votes the party got in the election. However, the number of seats in the parliament is integer, and considerably less than the number of votes, so perfect proportionality is not possible.

In Sweden, the mandates are distributed according to “the adjusted odd number method”, (in Swedish: “den jämkade uddatalsmetoden”), sometimes also called the modified Webster/Sainte-Laguë method, a sequential heuristic that is actually specified in legal text.

In this paper we first ignore constituencies, i.e. see all of the country as one constituency. The difference to reality is usually small, because one has introduced equalization mandates to eliminate the difference that arises because of the constituencies. We assume that the main common goal is to get a representation that is as proportional as possible for the whole nation.

However, sometimes the number of equalization mandates are not enough, or the allocation of them does not work as wished, so the final result is changed because of the constituencies. This is not desired, see for example Janson and Linusson (2014), and can give rise to questions about the formation and size of the constituencies, as well as about the number of equalization mandates.

Therefore we also consider the issue about constituencies, and suggest a procedure, based on the same kind of optimization problem as for the whole nation, for allocating mandates in the constituencies, without changing the overall allocation with respect to parties. In our approach, the numbers of mandates for the constituencies are based on the number of votes given, not on estimated numbers of inhabitants. This removes the need for fixed and equalization mandates, and also makes the question about sizes of the constituencies less important.

2 An optimization model

We consider an election with n parties. Each voter votes for one party, and the mandates in the parliament are allocated proportionally to the votes.

Let us denote the total number of votes with p and the total number of mandates to be allocated with m . We can calculate $d = m/p$, which in principle indicates the number of mandates per vote (significantly less than 1). Conversely, $1/d$ in principle indicates the number of votes per mandate. Let r_j specify how many votes party j received in the election (so $p = \sum_j r_j$).

Let us introduce the variables x_j as the number of mandates party j will get.

The value $v = \sum_{j=1}^n (x_j - dr_j)^2$ is often used for measuring how disproportionate the

distribution of mandates is, and $\sqrt{v/2}$ is called the Gallagher index. It has also been called the least square index. (We will later use the notation G-index.)

If we ignore that x should be integers, the solution $x_j = dr_j$ for all j gives perfect proportionality, i.e. Gallagher index 0.

In another measure, the Sainte-Laguë index, the terms are weighted with the proportion of votes, $f(x) = \sum_{j=1}^n (x_j - dr_j)^2 / r_j$. It means that we compare the relative deviation, so one unit's error is worse for a party with fewer votes. (We will later use the notation SL-index.)

A third measure is the Loosemore-Hanby index, where the target function is $f(x) = \sum_{j=1}^n |x_j - dr_j|$ (divided by 2). (We will later use the notation LH-index.)

We will start by using the Gallagher index, mainly because we believe that it is the best measure, and consider the other measures later.

The integer solution that lies as close as possible to perfect proportionality is the solution of the following optimization problem.

$$\begin{aligned} \min \quad & f(x) = \sum_{j=1}^n (x_j - dr_j)^2 \\ \text{s.t.} \quad & \sum_{j=1}^n x_j = m \\ & x_j \geq 0, \text{ integer, for all } j \end{aligned} \tag{P1}$$

This is a small nonlinear integer problem. The number of variables is equal to the number of parties.

3 A lower limit

Usually there is a parliament barrier, i.e. a lower proportional limit, l , for a party to get any mandates at all. If a party gets less than lp votes, the party gets no mandates. In Sweden, $l = 0.04$, i.e. if a party gets less than 4% of the votes, the party gets no mandates. In practice, parties that lie far below the limit are often combined into one party, "Others", under the condition that the sum of votes still lies below the limit.

Let CP1 denote the continuous relaxation of P1, obtained by relaxing the integer requirements. It is clear that $x_j = dr_j$ is the optimal solution of CP1 if there is no lower limit.

With the help of the KKT conditions one can show the following for the optimal solution of CP1. Let $J = \{j : r_j \geq lp\}$, i.e. J is the set of parties that do not fall below the limit. We then get $x_j = 0$ for $j \notin J$. Let $s = \sum_{j \in J} r_j$, i.e. s is the total number of votes given to parties that do not fall below the barrier. Then $p - s$ votes were given to parties that fall below the limit and will not get any mandates. One might call them "wasted" votes, but they are not meaningless, since they affect d , i.e. affect the number

of votes needed to get a mandate.

Let us now assume that all parties having more than lp votes will receive at least one mandate. This is certainly true if $m \geq 1/l$, so for $l = 0.04$, the parliament must have at least 25 seats. In Sweden there are 349.

Under these assumptions, the KKT conditions give $x_j = dr_j - u/2$ for all $j \in J$, where u is the multiplier of constraint (1) in P1. Since we must have $\sum_{j \in J} x_j = m$, we get $\sum_{j \in J} (dr_j - u/2) = m$, i.e. $ds - qu/2 = m$, where $q = |J|$, i.e. the number of parties that do not fall below the barrier. This gives $u = 2m(s/p - 1)/q$. (s/p is somewhat smaller than 1, so u will be negative.) In words $1 - s/p = (p - s)/s$ is the proportion of votes on parties that fall below the limit and will not get any mandates. This is divided by the number of remaining parties and multiplied with $2m$ to ensure that constraint (1) will be satisfied. We now get $x_j = dr_j - u/2 = dr_j + m(1 - s/p)/q$. The conclusion is that all values for the remaining variables will be increased by the same amount, u , so that the sum becomes equal to m . This way CP1 can be explicitly and exactly solved.

Unfortunately, the integer problem P1 is more complicated when there is a lower limit. Especially we cannot add the same amount to all parties in J . As mentioned above, we have $p - s$ “wasted” votes, i.e. votes that does not give any mandates. (Note that even though the limit is lp for single parties, the sum might exceed lp .)

Can we simply remove all parties that fall below the barrier? Let us compare two cases, one where a party with $\delta < lp$ votes is removed from the list before hand, together with its votes, and one where it is left in the list. In both cases, that party will get zero mandates. Can this make a difference for the remaining parties?

Removing the party from consideration will decrease n by one and decrease p by δ . Since $d = m/p$, this means that d will increase to $m/(p - \delta)$. This change will be the same for all parties, but since the objective function is not linear, the effect on the terms $(x_j - dr_j)^2$ will vary between parties. Above we traced the total effect on the continuous solution, namely that all x_j ’s were increased by the same amount. Here integrality will make the change different for different parties.

The shape of the objective function is the same (quadratic) for all parties, but is centered around dr_j , which is different for different parties. We note that the function is steeper when the distance between x_j and dr_j is larger. Different parties will lie at different distances from dr_j , so the change in slope will be different. The conclusion is that the number of mandates for the other parties might be affected by removing the votes for a party that falls below the limit.

Is this a good or bad property? Removing a party completely from consideration changes d , which means that the number of votes needed for a mandate will change, so this is unavoidable. The conclusion is that we should not remove such votes. (Later we will see that the adjusted odd number method does not use d , and would be unaffected by removing small parties.)

4 Solving the integer problem

Let us now address the integer problem P1, and let us temporarily assume that there are no parties with less than lp votes. (This is just a notational simplification, in order to avoid the discussion of limits at this stage. We will consider barriers later.)

To solve the integer problem P1, we first note that the objective function is additively separable in j . Therefore we can introduce a piecewise linearization of the non-linear objective function $f_j(x_j) = (x_j - dr_j)^2$. Since the variables must take integer values, this linearization becomes *exact* if it has the correct values in all integer points. We can calculate coefficients representing the slope of the objective function between two adjacent integer points by the following expression, denoted by C1.

$$c_{jk} = f_j(k) - f_j(k-1) = (k - dr_j)^2 - (k-1 - dr_j)^2 = 2(k - dr_j) - 1 \text{ for } k = 1, \dots, m.$$

Since $f_j(x_j)$ is a convex function, we have $c_{jk} \geq c_{j,k-1}$.

Now we replace x_j by $\sum_k x_{jk}$, where the binary variable x_{jk} is the part of x_j that lies in the interval $[k-1, k]$. We get the following optimization problem, which gives the same solution as P1 (with $x_j = \sum_k x_{jk}$).

$$\begin{aligned} \min \quad & z = \sum_{j=1}^n \sum_{k=1}^m c_{jk} x_{jk} \\ \text{s.t.} \quad & \sum_{j=1}^n \sum_{k=1}^m x_{jk} = m \quad (1) \\ & x_{jk} \in \{0, 1\} \text{ for all } j, k \end{aligned} \tag{P2}$$

P2 is a linear integer problem, and $f(x^*) = z^* + \sum_{j=1}^n (dr_j)^2$. The number of variables is mn . Since $c_{jk} \leq c_{j,k+1}$, $x_{j,k+1}$ may be equal to one only if $x_{jk} = 1$, $x_{j,k-1} = 1$, etc. Thus $x_{jk} = 1$ if $x_j \geq k$.

We may also consider the problem P3, which is P2 without integrality requirements.

$$\begin{aligned} \min \quad & z = \sum_{j=1}^n \sum_{k=1}^m c_{jk} x_{jk} \\ \text{s.t.} \quad & \sum_{j=1}^n \sum_{k=1}^m x_{jk} = m \quad (1) \\ & 0 \leq x_{jk} \leq 1, \text{ for all } j, k \end{aligned} \tag{P3}$$

One can show that all the extreme points of the feasible set of P3 are integer, which means that solving P3 produces an integer solutions which also is optimal in P2. Our optimization problem can thus be solved as an LP-problem.

There exists a well known greedy algorithm that optimally solves continuous knapsack problems, and P3 is a simpler type of such a problem, since all the coefficients in the knapsack constraint are equal to one. The general method is as follows: Find the best unused variable, increase it, repeat until the knapsack is full. For an ordinary continuous

knapsack problem, the last variable increased may get a non-integral value, in order to fill the knapsack exactly, but here that will not happen.

Let us temporarily ignore the index j . We then have an integer variable $x = \sum_k x_k$ with costs c_k and wish to increase it as long it is beneficial. Let the current value of x be denoted by \hat{x} . Since the cost function is convex, this means that $x_k = 1$ for all $k \leq \hat{x}$ and $x_k = 0$ for all $k > \hat{x}$. Now the question is if we should increase x to $\hat{x} + 1$ or not, i.e. if we should set $x_{k'} = 1$ for $k' = \hat{x} + 1$. For this reason we look at $c_{k'}$.

Let us now specify the algorithm (with j). We denote the number of mandates allocated by \hat{m} . We start by all variables equal to zero, $\hat{x}_j = 0$, i.e. no mandates allocated, $\hat{m} = 0$. In each iteration, we set $k_j = \hat{x}_j + 1$, and calculate $v_j = c_{jk_j} = 2(k_j - dr_j) - 1$ for all j . This yields $v_j = 2(\hat{x}_j + 1 - dr_j) - 1 = 2(\hat{x}_j - dr_j) + 1$ for all j . In the first iteration, $k_j = 1$, and we get $v_j = 2(1 - dr_j) - 1 = 1 - 2dr_j$.

The values v now give the cost for increasing each variable to the next integer value (which is k_j), i.e. allocating one more mandate. We choose the best of the possibilities, by finding $\min_j c_{jk_j}$, and the corresponding index by $\hat{j} = \arg \min_j c_{jk_j}$. Then \hat{j} identifies the best variable to increase, and we set $\hat{x}_{\hat{j}} = \hat{x}_{\hat{j}} + 1$, i.e. allocate one more mandate to party \hat{j} . This yields $\hat{m} = \hat{m} + 1$. If $\hat{m} = m$, we are ready. Otherwise this is repeated.

In each iteration, one more mandate is allocated, so there will be exactly m iterations. In each iteration, only one value v_j needs to be calculated, since v_j is unchanged for all $j \neq \hat{j}$. In other words, the value only needs to be recalculated for the party that got the mandate. Therefore, this method is very quick.

A lower limit l is simply taken care of by not allowing any mandates to parties with less votes than this. This does not change the method. It is however necessary, as noted above, to include the votes for such parties in the calculation of the coefficients.

4.1 The algorithm

Let us now give the algorithm for solving P3, with the simplified notation $s_j = \hat{x}_j$ and $t = \hat{j}$. Let $J = \{j : r_j \geq lp\}$, i.e. the set of parties that do not fall below the limit.

Algorithm 1 Exact mandate allocation

- 1: Set $\hat{m} = 0$, $s_j = 0$ for all j , and calculate $v_j = 1 - 2dr_j$ for all j .
 - 2: **while** $\hat{m} < m$ **do**
 - 3: Find $t = \arg \min_{j \in J} v_j$.
 - 4: Set $s_t = s_t + 1$, and $\hat{m} = \hat{m} + 1$.
 - 5: Calculate $v_t = 2(s_t - dr_t) + 1$.
-

Note that this actually gives the optimal solution, i.e. the feasible solution that has the minimal disproportionality, in P3, P2 and P1.

4.2 Verification of optimality

Let us now verify that the solution obtained by the algorithm is optimal, with the help of LP-duality. Given the primal solution $s = \hat{x}$, we calculate the corresponding dual solution, and show that both solutions are optimal.

The algorithm yields the solution s that satisfies the primal constraints. We have $x_{jk} = 1$ for all $k \leq s_j$ and $x_{jk} = 0$ for all $k > s_j$, for each $j \in J$, and $x_{jk} = 0$ for all $j \notin J$ and all k . The objective function value clearly is equal to $\sum_{j \in J} \sum_{k=1}^{s_j} c_{jk}$.

The LP-dual of P3, with dual variables α for constraint 1 and β for the constraints $x \leq 1$ is given below.

$$\begin{aligned} \max \quad v = \quad & m\alpha - \sum_{j=1}^n \sum_{k=1}^m \beta_{jk} \\ \text{s.t.} \quad & \alpha - \beta_{jk} \leq c_{jk}, \text{ for all } j, k \\ & \beta_{jk} \geq 0, \text{ for all } j, k \end{aligned} \tag{D3}$$

The complementary slackness conditions are $(\alpha - \beta_{jk} - c_{jk})x_{jk} = 0$ for all j, k , and $\beta_{jk}(x_{jk} - 1) = 0$ for all j, k .

The dual constraints can be written as $\beta_{jk} \geq \max(0, \alpha - c_{jk})$. For $k > s_j$, we have $x_{jk} = 0$, so complementary slackness yields $\beta_{jk} = 0$, and then the dual constraints reduce to $\alpha \leq c_{jk}$. This is also true for $j \notin J$, so we have $\alpha \leq c_{jk}$ for all $j \notin J$ and for all $k > s_j$ for $j \in J$. Since the objective function strives to maximize α , we set

$$\alpha = \min\left(\min_{k, j \notin J} c_{jk}, \min_{k > s_j, j \in J} c_{jk}\right).$$

For $k \leq \hat{x}_j$, we have $x_{jk} = 1$, so complementary slackness yields $\beta_{jk} = \alpha - c_{jk}$.

These values of α and β ensure that the dual solution is feasible in D3. The objective function value now becomes

$$m\alpha - \sum_{j=1}^n \sum_{k=1}^m \beta_{jk} = m\alpha - \sum_{j \in J} \sum_{k=1}^{s_j} (\alpha - c_{jk}) = (m - \sum_{j \in J} s_j)\alpha + \sum_{j \in J} \sum_{k=1}^{s_j} c_{jk} = \sum_{j \in J} \sum_{k=1}^{s_j} c_{jk}$$

where we have used primal constraint (1). We find that the dual objective function value is equal to the primal objective function value. We thus have a primal feasible solution and a dual feasible solution and they have the same objective function value. Therefore they are optimal solutions.

5 Other measures

First we note that the other measures suggested also are separable in j and convex. Actually we believe that all reasonable measures should be separable and convex. Separability is needed, so that moving a mandate between two parties should not affect a

third party. The only connection between parties should be constraint 1.

For a convex function, a larger deviation from dr_j gives a larger “cost”, i.e. will be less desirable. A non-convex function that gives a smaller cost for a larger deviation would not be acceptable. This reasoning also applies to marginal effects, so a convex function is desirable.

Using the Sainte-Laguë index, the terms are weighted with the proportion of votes, $f(x) = \sum_{j=1}^n (x_j - dr_j)^2 / r_j$. It is easy to see that this only means that the coefficients c_{jk} are all divided by r_j , and that the same algorithm can be used. Let C2 denote the expression $c_{jk} = (2(k - dr_j) - 1) / r_j$ for $k = 1, \dots, m$.

Using the Loosemore-Hanby index, $f(x) = \sum_{j=1}^n |x_j - dr_j|$, we get $c_{jk} = f_j(k) - f_j(k - 1) = |k - dr_j| - |k - 1 - dr_j|$ for $k = 1, \dots, m$.

If $k \leq dr_j$, $|k - dr_j| \leq 0$, so $|k - dr_j| = -k + dr_j$. Since $k - 1 < k$, we likewise get $|k - 1 - dr_j| = -k + 1 + dr_j$. Then $|k - dr_j| - |k - 1 - dr_j| = -k + dr_j - (-k + 1 + dr_j) = -1$.

If $k - 1 \geq dr_j$, $|k - 1 - dr_j| \geq 0$, so $|k - 1 - dr_j| = k - 1 - dr_j$. Since $k > k - 1$, we also get $|k - dr_j| = k - dr_j$. Then $|k - dr_j| - |k - 1 - dr_j| = k - dr_j - (k - 1 - dr_j) = 1$.

Finally, if $k - 1 < dr_j < k$, $|k - dr_j| = k - dr_j$, and $|k - 1 - dr_j| = -k + 1 + dr_j$. Then $|k - dr_j| - |k - 1 - dr_j| = k - dr_j - (-k + 1 + dr_j) = 2k - 2dr_j - 1$.

Summing up, we have the following expression, denoted by C3.

$$c_{jk} = \begin{cases} -1 & \text{if } k \leq dr_j \\ 2k - 2dr_j - 1 & \text{if } k - 1 < dr_j < k \\ 1 & \text{if } k - 1 \geq dr_j \end{cases}$$

The cost curve of C3 is thus convex, but not strictly convex. Mostly it is linear. This is, as we shall see, an undesirable property.

6 The adjusted odd number rule

In Sweden, the mandates are distributed according to “the adjusted odd number rule”, (“jämkade uddatalsmetoden” in Swedish), also called the modified Sainte-Laguë method or the modified Webster/Sainte-Laguë method, a sequential heuristic that is specified in legal text. The method has also been used in Denmark, Norway, Bosnia-Herzegovina, Iraq, Kosovo, Latvia, New Zealand and Nepal. It can be described as follows.

One works with values, v_j , for each party, and allocates mandates one at a time to the party that has the highest value. Then one divides the party’s value with the next odd number, and repeats this. The designation “adjusted” means that the initial values of v are the number of votes divided by 1.2. This makes the first mandate somewhat delayed and thus gives a disadvantage for smaller parties. In Sweden, before 2018, 1.4 was used instead of 1.2.

Algorithm 2 The adjusted odd number method

- 1: Set $\hat{m} = 0$, $s_j = 0$ for all j , and calculate $v_j = r_j/1.2$ for all j .
 - 2: **while** $\hat{m} < m$ **do**
 - 3: Find $t = \arg \max_{j \in J} v_j$.
 - 4: Set $s_t = s_t + 1$, and $\hat{m} = \hat{m} + 1$.
 - 5: Calculate $v_t = r_t/(2s_t + 1)$.
-

Empirically, this method appears to give quite good proportionality. Judging from the name, it seems to aim at minimizing the Sainte-Laguë index. However, we have not seen any derivation motivating this, and have found that it does not always produce the best solution, even with that objective function.

In Linusson (2008) it is mathematically shown that this method does very well when comparing two parties, and this is taken as motivation for claiming that the method is very good. However, for more than two parties, nothing is theoretically shown.

If the initial factor 1.2 is replaced by one, the method is simply called the “odd number method” (“uddtalsmetoden” in Swedish), or the Sainte-Laguë method. A lower limit, l , (like 0.04 used in Sweden) often removes the effect of the adjustment, as it is a stronger disadvantage for small parties.

In another method, called d’Hondts method, step 4 is replaced by $v_t = r_t/(s_t + 1)$, i.e. division is made with the next integer, not the next odd integer. In Sweden this method is used in elections conducted by a city council, municipal council or municipal board of directors. The method is said to favor large parties.

Since these methods are said to favor large parties, a technique occasionally used is electoral cooperation, where two different parties sum up their votes and are counted as one. Obviously this may make it possible to avoid the barrier, and also to avoid effects of the initial factor 1.2. Otherwise the effect is unclear.

7 Constituencies

In this paper, we have up to now ignored constituencies. However, in reality they are there. The procedure used (in Sweden) is that each constituency has a fixed number of mandates (based on the number of inhabitants in the area), and the allocation is made separately for each constituency. After this, the result is summed up, and compared to the result for the whole nation. Equalization mandates are then allocated, based on certain procedures, in order to eliminate the differences that appear because of the constituencies.

Sometimes, for example 2010 in Sweden, this procedure does not succeed to eliminate the differences, and the final result is not what it would have been for the whole nation as one constituency. This has raised debates and protests, and the rules for allocating equalization mandates and the number of equalization mandates have been changed.

We here suggest a different approach. First of all, we say that the allocation for the nation as a whole (preferably found with Algorithm 1) must be kept, i.e. not changed at all. Given the number of mandates for each party, s , the question is how to divide those between constituencies.

Assume that we have q constituencies, and let t_{ij} be the number of votes for party j in constituency i . (We have $r_j = \sum_i t_{ij}$.)

Now let y_{ij} be the number of mandates for party j in constituency i . We get the constraints $\sum_{i=1}^q y_{ij} = s_j$ for each j , stating that the votes from the constituencies should sum up to the national values for each party.

Then we suggest to use the same approach as used for the whole nation. The scaled proportions of votes, $y_{ij} = dt_{ij}$, would be the correct solution if non-integral values were allowed, since we have $\sum_{i=1}^q y_{ij} = \sum_{i=1}^q dt_{ij} = d \sum_{i=1}^q t_{ij} = dr_j$.

Since y has to be integer, we can formulate an optimization problem similar to P1 for each j . We use the same least square deviation as objective function.

$$\begin{aligned} \min \quad & \sum_{i=1}^q (y_{ij} - dt_{ij})^2 \\ \text{s.t.} \quad & \sum_{i=1}^q y_{ij} = s_j \\ & y_{ij} \geq 0, \text{ integer, for all } i \end{aligned} \tag{P4}$$

P4 has the same properties as P1, and can be solved in the same way. Note however that the summations are over i , the constituencies, and not over j , the parties, as in P1. We again make an exact linearization in integer points $k = y_{ij}$.

$$a_{ijk} = (k - dt_{ij})^2 - (k - 1 - dt_{ij})^2 = 2(k - dt_{ij}) - 1 \text{ for } k = 1, \dots, m.$$

Now we replace y_{ij} by $\sum_k y_{ijk}$, where the binary variable y_{ijk} is the part of y_{ij} that lies in the interval $[k - 1, k]$. We get the following optimization problem for each j .

$$\begin{aligned} \min \quad & z = \sum_{i=1}^q \sum_{k=1}^m a_{ijk} y_{ijk} \\ \text{s.t.} \quad & \sum_{i=1}^q \sum_{k=1}^m y_{ijk} = s_j \\ & 0 \leq y_{ijk} \leq 1, \text{ integer, for all } i, k \end{aligned} \tag{P5}$$

P5 is a linear problem with integer extreme points. We get $y_{ijk} = 1$ if $y_{ij} \geq k$. Now the problem can be solved with the following algorithm, which uses the output, s , of Algorithm 1 as input.

We let \hat{m}_j be the number of mandates allocated to party j . Also let I_j be the set of constituencies that does not fall below a lower limit, i.e. those constituencies where party j may get votes. (The lower limits here may be different from the national level.)

This algorithm solves P4 exactly (just as Algorithm 1 solves P1).

Algorithm 3 Constituency mandate allocation

```
1: for  $j = 1, \dots, n$  do  
2:   Set  $\hat{m}_j = 0$ ,  $y_{ij} = 0$ , and calculate  $v_{ij} = 1 - 2dt_{ij}$ , for all  $i$ .  
3:   while  $\hat{m}_j < s_j$  do  
4:     Find  $\tau = \arg \min_{i \in I_j} v_{ij}$ .  
5:     Set  $y_{\tau j} = y_{\tau j} + 1$ , and  $\hat{m}_j = \hat{m}_j + 1$ .  
6:     Calculate  $v_{\tau j} = 2(y_{\tau j} - dt_{\tau j}) + 1$ .
```

A big difference compared to the presently used procedure is that the number of mandates for a constituency depends on the number of *votes* from the area, not on the number of inhabitants. If few of the inhabitants vote, the constituency gets few mandates. On the other hand, if few inhabitants vote in the current system, those who vote have a larger impact.

An advantage is that the number of votes obviously is very current, while the number of inhabitants may be an outdated number. Another advantage is that it encourages voting.

The need for fixed mandates and equalization mandates and procedures for allocating them is completely removed. Furthermore the sizes of the constituencies is not as important as it is in the present procedure. Changing the sizes of constituencies might still have some effect, due to the integrality of the mandates, but those effects are small and random, and we believe that it is not possible to use such a change in order to achieve certain party-political goals.

8 Implementation

The methods are implemented in Python, and the code also includes an implementation of the adjusted odd number method, which is presently used in Sweden. The code is run in a terminal as follows.

```
python elect.py votes-2018.txt 0.04 349
```

The first argument, votes-2018.txt, is which input file to be used. The second argument, 0.04, is the limit l . The third argument, 349, is the number of mandates to be distributed. This means that one can easily run the program with different input files, various parliamentary barriers and different number of mandates. The code is obtainable from the author on request.

8.1 Indata

An input file has one row per party, and each row contains the name of the party and the number of votes the party received. The data is retrieved directly from the Election Authority's website ("Valmyndigheten" in Swedish) for the whole nation. There are three files, votes-2018.txt, votes-2014.txt and votes-2010.txt, with results from three

Party	r_j	dr_j	r_j/p
The Administrators (A)	320	32.0	0.32
The Bureaucrats (B)	280	28.0	0.28
The Commoners (C)	260	26.0	0.26
The Different (D)	80	8.0	0.08
The Xenophobes (X)	30	3.0	0.03
The Yetis (Y)	20	2.0	0.02
The Zeptoparty (Z)	10	1.0	0.01

Table 1: Parties and votes for the first instance.

elections.

The votes for many insignificant parties are in this data reported under one name, “Others”. Usually the proportion of votes are below the limit l , so no mandates are allocated. However, it is not impossible that the sum of votes for all the small parties exceeds the limit. Since they are not one party, but many, they should not get any mandates anyway. If the method treats it as one party, it might get mandates. We must therefore deal with this item in a special way, i.e. not include it in J in Algorithm 1. Another possibility is to extract the largest parties in “Others” and give them their own lines, until the remaining combined party gets less than lp votes.

We have also used a small artificial instance for initial testing, and a few small instances given in Linusson (2008).

9 Computational results

9.1 One artificial instance

First we solve an artificial instance, initially used for debugging. We have 7 parties, see table 1, 1000 votes and 100 mandates. The number are chosen in order to yield integer proportions. 100 mandates for 1000 votes gives 0.1 mandate per vote or 10 votes per mandate. In table 1, we give the name of the party, the number of votes it got, r_j , the continuous solution, dr_j , and the proportion of the votes it got, r_j/p .

With $l = 0$, i.e. no lower limit, both Algorithm 1 and 2 give the solution $x_j = dr_j$, with G-, SL- and LH-index all equal to zero. In table 2, the results for $l = 0.02$, $l = 0.03$ and $l = 0.04$ are given.

In table 2, we find that for $l \leq 0.02$, the two methods are equally good. However, for $l = 0.03$ and 0.04 , Algorithm 1 gives better solutions, i.e. lower G-index. Also the SL-index is lower, while there is no difference for the LH-index. We also find that larger l gives worse proportionality.

If the X, Y and Z parties form an electoral cooperation for $l = 0.04$, the result will be the same as for $l = 0$, since the combined party, XYZ, then has more than 4% of the votes.

Party	$l = 0.02$		$l = 0.03$		$l = 0.04$	
	Alg 1	Alg 2	Alg 1	Alg 2	Alg 1	Alg 2
A	33	33	33	35	34	38
B	28	28	29	28	30	28
C	26	26	27	26	27	26
D	8	8	8	8	9	8
X	3	3	3	3	0	0
Y	2	2	0	0	0	0
Z	0	0	0	0	0	0
G-index	1.000	1.000	2.000	2.645	3.464	5.000
SL-index	0.103	0.103	0.310	0.328	0.643	0.712
LH-index	1.000	1.000	3.000	3.000	6.000	6.000

Table 2: Mandates for different lower limits for the first instance.

Örkelträsk 4					
Party	r_j	dr_j	r_j/p	Alg 1	Alg 2
A	333	3.33	0.475	3	4
B	237	2.37	0.338	3	2
C	130	1.3	0.185	1	1
G-index				0.545	0.581

Örkelträsk 5					
Party	r_j	dr_j	r_j/p	Alg 1	Alg 2
A	367	3.67	0.524	4	3
B	267	2.67	0.381	3	3
C	66	0.66	0.094	0	1
G-index				0.571	0.580

Table 3: Votes and mandates for the second set of instances.

9.2 Small artificial instances

We have also solved some small test instances described in Linusson (2008), there used to illustrate the adjusted odd number method, Algorithm 2. Here there are three parties, 700 votes and 7 mandates, and no lower limit, i.e. $l = 0$. This gives 0.01 mandate per vote or 100 votes per mandate. In table 3, we give the name of the party, the number of votes it got, r_j , the continuous solution, dr_j , and the proportions of the votes it got, r_j/p , for the two instances Örkelträsk 4 and Örkelträsk 5. The last two columns give the number of mandates allocated by the two algorithms, and the Gallagher index for the solutions.

For both these instances, Algorithm 1 gives better solutions.

9.3 Sweden

Finally we have used the data from the three last elections in Sweden, see tables 4, 5 and 6. In these runs we used $l = 0.04$ and $m = 349$, as is the case in reality. (We have

Party	r_j	dr_j	r_j/p	Alg 1	Alg 2	Res
Moderaterna (M)	1791766	104.913	0.300	106	106	107
Centerpartiet (C)	390804	22.882	0.065	23	23	23
Folkpartiet (FP)	420524	24.622	0.0705	25	25	24
Kristdemokraterna (KD)	333696	19.538	0.055	20	20	19
Socialdemokraterna (S)	1827497	107.005	0.306	108	109	112
Vänsterpartiet (V)	334053	19.559	0.056	20	20	19
Miljöpartiet (MP)	437435	25.613	0.073	26	26	25
Sverigedemokraterna (SD)	339610	19.885	0.056	21	20	20
Övriga	85023	4.978	0.014	0	0	0
G-index				3.801	3.915	5.266

Table 4: Votes and mandates for the 2010 election in Sweden.

Party	r_j	dr_j	r_j/p	Alg 1	Alg 2	Res
Moderaterna (M)	1453517	81.404	0.233	83	85	84
Centerpartiet (C)	380937	21.334	0.061	23	22	22
Folkpartiet (FP)	337773	18.917	0.054	21	20	19
Kristdemokraterna (KD)	284806	15.950	0.045	18	17	16
Socialdemokraterna (S)	1932711	108.241	0.310	110	112	113
Vänsterpartiet (V)	356331	19.956	0.057	22	21	21
Miljöpartiet (MP)	429275	24.041	0.068	26	25	25
Sverigedemokraterna (SD)	801178	44.870	0.128	46	47	49
Feministiskt initiativ (FI)	194719	10.905	0.031	0	0	0
Övriga	60326	3.378	0.009	0	0	0
G-index				8.848	9.128	9.466

Table 5: Votes and mandates for the 2014 election in Sweden.

not translated the Swedish party names.) The last row, named “Övriga” is the sum of all minor parties.

In the election 2010, there was 5 960 408 votes, which gives 0.0000585 mandates per vote or 17078 votes per mandate. In the election 2014, there was 6 231 573 votes, which gives 0.0000560 mandates per vote or 17855 votes per mandate. In the election 2018, there was 6 476 725 votes, which gives 0.0000538 mandates per vote or 18557 votes per mandate. The results from Algorithm 2 are not identical to the final real life results, due to some deficiencies in the allocation of equalization mandates. Our main goal is to compare the two algorithms, but we also give the actual result in the last column.

In all these elections, Algorithm 1 gives better solutions than Algorithm 2. In other words, the new algorithm presented in this paper would give a more proportional mandate allocation.

We also find that for 2018, Algorithm 2 and the actual result are identical, which shows that the procedure with equalization mandates worked rather well that year. For 2014, the actual result is slightly worse, and for 2010, there is a large difference, as Algorithm 1 gives Gallagher index 3.801 and Algorithm 2 3.915, while the actual result gives 5.266. There were articles in newspapers claiming that the election result was flawed, and we

Party	r_j	dr_j	r_j/p	Alg 1	Alg 2	Res
Moderaterna (M)	1284698	69.22	0.198	70	70	70
Centerpartiet (C)	557500	30.041	0.086	31	31	31
Liberalerna (L)	355546	19.158	0.054	20	20	20
Kristdemokraterna (KD)	409478	22.064	0.063	23	22	22
Socialdemokraterna (S)	1830386	98.630	0.282	99	100	100
Vänsterpartiet (V)	518454	27.937	0.080	28	28	28
Miljöpartiet (MP)	285899	15.405	0.044	16	16	16
Sverigedemokraterna (SD)	1135627	61.193	0.175	62	62	62
Feministiskt initiativ (FI)	29665	1.598	0.004	0	0	0
Övriga	69472	3.743	0.010	0	0	0
G-index				3.225	3.292	3.292

Table 6: Votes and mandates for the 2018 election in Sweden.

Party	Alg 1	Alg 2
M	72	74
C	33	32
L	22	20
KD	25	23
S	102	105
V	31	30
MP	0	0
SD	64	65
FI	0	0
Övriga	0	0
G-index	12.562	13.065

Table 7: Votes and mandates for the 2018 election in Sweden with $l = 0.05$.

can only agree. However, since then either the flaws have been corrected, or we were just luckier with the numbers.

Tests with initial factor 1.4 in Algorithm 2, as was used in the elections in Sweden before 2018, gave identical results. The same is the case for initial factor 1.0. Our conclusion here is that the “adjusted” part of Algorithm 2 does not make a big difference.

On the other hand, raising the lower limit l to 0.05 gives the result for 2018 given in table 7, which shows important differences. Also it is clear that the Gallagher index increases when l increases.

We earlier mentioned electoral cooperation, where two parties are treated as one. In section 9.1 this was used for small parties to get above the lower limit. Let us also consider the effect it may have for parties over the limit. We have used the data from 2018 with three (fictitious) cooperations, and the results are given in table 8, where small parties have been removed.

Due to the integrality of the mandates, it is inevitable that electoral cooperation may have some effects, but it is not desired, and the effect should be small and random, and

Cooperation1			Cooperation2			Cooperation3		
Party	Alg 1	Alg 2	Party	Alg 1	Alg 2	Party	Alg 1	Alg 2
M	70	71	M+KD	92	93	M	70	70
C+L	50	50	C	31	31	C	31	31
KD	23	22	L	20	19	L	20	19
S	99	100	S	99	100	KD	23	22
V	29	28	V	29	28	S+V	127	129
MP	16	16	MP	16	16	MP	16	16
SD	62	62	SD	62	62	SD	62	62
G-index	3.236	3.408		3.237	3.414		3.229	3.538

Table 8: Votes and mandates for the 2018 election in Sweden with electoral cooperations between parties.

Block	2010			2014			2018		
	Alg 1	Alg 2	Res	Alg 1	Alg 2	Res	Alg 1	Alg 2	Res
RB	174	174	173	145	144	141	144	143	143
LB	154	155	156	158	158	159	143	144	144
SD	21	20	20	46	47	49	62	62	62

Table 9: Mandates for the blocks.

not possible to plan in advance and use tactically. The Gallagher indices for Algorithm 1 are 3.225 without cooperation, and 3.236, 3.237 and 3.229 for the three cooperations. For Algorithm 2, the corresponding numbers are 3.292, 3.408, 3.414 and 3.538. Clearly the disproportionality increases more for Algorithm 2 than Algorithm 1. The changes in our method are small and random, so we conclude that our method is better at handling such changes.

9.4 Political blocks

The issue about political blocks is discussed much in Sweden. Traditionally there has been a left block, LB, of S and V, where MP in later years has been included. The traditional right block, RB, has been M, KD, L and C. (L was called FP before 2018.) Where to put SD has been unclear, and it is sometimes called a third block. In table 9 we give the results for these blocks for the two algorithms, and the actual result.

There are some small differences between the algorithms, and one is especially interesting. For 2018 Algorithm 2 gives the left block one more mandate than the right block, which is equal to the actual result. However, Algorithm 1 would give the right block one more mandate than the left block. That difference would probably have been psychologically important, even though it is hard to say what difference it would have been when it came to forming a government, which was unusually difficult (for Sweden) and took more than 130 days.

Since we believe that Algorithm 1 gives the best result, we note that for 2010 and 2014, the actual results were even worse than Algorithm 2.

Party	$l = 0.03$		$l = 0.04$	
	Alg 1	Alg 2	Alg 1	Alg 2
A	33	35	34	38
B	29	28	30	28
C	27	26	28	26
D	8	8	8	8
X	3	3	0	0
Y	0	0	0	0
Z	0	0	0	0
G-index	2.0	2.645	3.605	5.0.00
SL-index	0.310	0.328	0.642	0.712

Table 10: Mandates for objective function C2 for the first instance.

Studying the three electoral cooperations we tested does not show any significant differences for the blocks. Especially it does not seem to help the cooperating parties, unless it is used to get above the lower limit.

9.5 Tests with other measures

We have also considered using the other measures, C2 and C3, in our algorithm. The first conclusion was that C3 has a lack of controllability, i.e. many coefficients are equal (-1 or 1), so the choice of minimizer is quite random. Practical tests confirmed that this measure is inferior.

Using C2 instead of C1, we found the following. While using C1 in the objective function often produces good results for the other indices too, using C2 as objective function produced worse results for C1.

The other observation was that using C2 as objective function, the two algorithms gave the same result for the Sweden instances. So it seems that Algorithm 2 gives good solutions for the problem with C2 as objective function.

To further investigate this, we solved the instance in section 9.1 with C2 as objective function. With $l \leq 0.02$, Algorithm 1 and 2 give the same solutions. In table 10, the results for $l = 0.03$ and $l = 0.04$ are given. (We also did runs with initial factor 1.0 instead of 1.2. i.e. with the unmodified Sainte-Laguë method, but the results were identical.)

Here we see that Algorithm 1 gives better solutions, i.e. lower SL-index. (Also the G-index is lower.) This means that the (modified) Sainte-Laguë method does not always give the lowest Sainte-Laguë index. Algorithm 1, however, does.

Therefore a reason to use Algorithm 1 instead of Algorithm 2 is that Algorithm 1 with C2 always produces the optimal solution, while this is not the case using Algorithm 2. (And it is not more difficult to use Algorithm 1 than Algorithm 2.)

Another reason is that we believe that C1 is the correct measure. It is not motivated

to put much higher cost on a certain deviation if the party in question has very few mandates than if the party has more mandates. From an emotional point of view from the smaller party, this might seem motivated, but from a more neutral point of view, there is no reason to favor votes for small parties. It would be unfortunate if votes for smaller parties always were given higher importance than votes for larger parties.

Furthermore, when it comes to forming government, several parties usually need to collaborate, and one mandate more or less for a smaller party is exactly as important as one mandate more or less for a large party if the two parties are collaborating.

9.6 Constituency allocation

We have also made tests with the method described in section 7. Here we take the result of national allocation of mandates as input, and allocated these mandates to the constituencies in the best possible way, using the same kind of quadratic error measure.

Here the total number of mandates for each party will not change. Instead the number of mandates for each constituency might change, if the estimated number of mandates is based on incorrect numbers of inhabitants, or if the number of votes are not proportional to the number of inhabitants.

It does not seem controversial to say that non-voters should not affect the election result. Especially, a high proportion of non-voters in a small constituency should not make each vote more important than in other constituencies.

We have made three runs, for the three last elections in Sweden. Since the allocation to parties is not changed, only the allocation to constituencies is interesting.

We begin by recalling that for 2010, the disproportionality decreased from 3.915 for the old method to 3.801 for our method, for 2014 from 9.128 to 8.848, and for 2018 from 3.292 to 3.225. The effect on the constituencies are summed up in table 11, where all constituencies are listed (the Swedish names are not translated) with their mandates, first the actual result of the election, Old, and then the results our method would give, New, and the difference, δ . We also, for comparison, give the number of fixed mandates, F, for 2018.

We find that using the new method would move in total 26 mandates between constituencies in 2010, 28 in 2014 and 14 in 2018. That can be compared with the 310 fixed mandates and 39 equalization mandates that has been used.

As a general trend, our method seems to move mandates to highly populated regions from sparsely populated. Göteborg and Stockholm get more, while the small Gotland loses one of its two. This might indicate that the numbers of mandates have been decided in a slightly conservative way, possibly wishing to avoid extremes.

Looking for example at Stockholms län, in 2010 it got 38 mandates, while our method would give 41, 3 more. In 2014, it got 39, while our method said 43, 4 more. In 2018 it got 43, and our method said 44, only one more. So it seems that the number of mandates

Constituency	2010			2014			2018			
	Old	New	δ	Old	New	δ	Old	New	δ	F
Blekinge län	6	5	-1	5	4	-1	5	5	0	5
Dalarnas län	11	11	0	11	11	0	10	10	0	9
Gotlands län	2	1	-1	2	1	-1	2	1	-1	2
Gävleborgs län	12	11	-1	11	11	0	9	9	0	9
Göteborgs kommun	18	20	+2	17	20	+3	19	18	-1	17
Hallands län	12	12	0	12	12	0	13	13	0	10
Jämtlands län	4	4	0	4	4	0	5	5	0	4
Jönköpings län	13	14	+1	13	15	+2	13	12	-1	11
Kalmar län	9	7	-2	8	7	-1	8	9	+1	8
Kronobergs län	6	5	-1	6	6	0	6	5	-1	6
Malmö kommun	10	10	0	11	9	-2	11	10	-1	10
Norrbottnens län	9	8	-1	8	8	0	8	8	0	8
Skåne läns norra och östra	12	12	0	13	13	0	11	11	0	10
Skåne läns södra	13	13	0	13	13	0	14	14	0	12
Skåne läns västra	10	9	-1	11	10	-1	11	12	+1	9
Stockholms kommun	29	32	+3	32	32	0	32	33	+1	29
Stockholms län	38	41	+3	39	43	+4	43	44	+1	39
Södermanlands län	11	11	0	11	12	+1	10	10	0	9
Uppsala län	13	14	+1	12	13	+1	13	13	0	11
Värmlands län	12	12	0	11	10	-1	11	11	0	9
Västerbottens län	11	11	0	10	10	0	9	9	0	9
Västernorrlands län	9	8	-1	10	8	-2	8	7	-1	8
Västmanlands län	11	9	-2	10	8	-2	9	11	+2	8
Västra Götalands läns norra	12	12	0	13	12	-1	10	11	+1	9
Västra Götalands läns södra	6	4	-2	6	5	-1	8	7	-1	7
Västra Götalands läns västra	13	14	+1	13	14	+1	13	13	0	11
Västra Götalands läns östra	10	10	0	10	9	-1	10	10	0	9
Örebro län	12	12	0	12	13	+1	12	12	0	9
Östergötlands län	15	17	+2	15	16	+1	16	16	0	14
Difference	26			28			14			

Table 11: Mandates for constituencies.

allocated by the old procedure changes rather slowly in the direction indicated (faster) by our method. A similar pattern can be seen for Göteborgs kommun. In general, however, there is no obvious pattern in the reallocation obtained by our method.

Comparing to the fixed mandates for 2018, it would be natural for the numbers of fixed mandates to lie slightly below the final result, since equalization mandates are added. However, in a few cases the fixed numbers of mandates lie above what our method would give. This occurs for small constituencies, and indicates, we believe, an unwillingness to accept that those constituencies really are that small.

Seen from another point of view, it might be reassuring that the mandate allocations really are rather similar. One can not say that our method would mean a drastic change. In any case, we believe that our method is much better motivated, both with theoretical and practical arguments.

10 Conclusion

We propose a new method for allocating mandates after an election. Tests show that our new method often produces solutions with better proportionality than the one presently used, and never worse. Furthermore it is not more complicated or time consuming to use. There is a sound theoretical base for the method, in that it correctly solves a relevant optimization problem. This can not be said for the present method, which is quite difficult to analyze.

We also suggest a new way of allocating mandates to constituencies, which eliminates the need for equalization mandates. We believe that it also eliminates many possible sources of errors, and that using this method is better than adjusting and amending the old rules when errors appear.

We believe that it is much better to discuss which optimization model to solve, i.e. what measure to use as objective function, than doing changes of parameters in an algorithm, when the effect of the changes is unclear.

After observing the recent difficulties in Sweden of forming a government, it seems even more vital that the allocation of mandates is done in the best possible way, meaning that it should be as proportional as possible, i.e. reflect the peoples votes as closely as possible. We claim that our method does the best job in this aspect. The code can be obtained by contacting the author at kaj.holmberg@liu.se.

References

- Janson, S., and Linusson, S. (2014), “Föreslagna ändringar i sveriges valsistem”, *SMS-bulletinen*. (In Swedish).
- Linusson, S. (2008), *Matematik och människor*, chap. Uddatalsmetoden och valsistem, Nationellt Centrum för Matematikutbildning. (In Swedish).

Linköping University
Department of Mathematics
SE-581 83 Linköping