

Department of Mathematics

Optimal proportional representation

Kaj Holmberg

LiTH-MAT-R--2020/03--SE

Linköping University
Department of Mathematics
SE-581 83 Linköping

Optimal proportional representation

Kaj Holmberg

kaj.holmberg@liu.se

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

March 6, 2020

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 the feasible solution that has the minimal disproportionality (with the measure chosen), even in the presence of a parliament threshold, which is not always the case for the practical 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 modified Sainte-Laguë method, 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. The numbers of mandates for the constituencies are based on the number of votes given, not on estimated numbers of inhabitants entitled to vote. This removes the need for compensatory mandates, and makes the question about sizes of the constituencies less important.

Key words: *OR in government; democracy; proportional representation; Gallagher index; Sainte-Laguë index.*

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 (seats 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 an integer, and considerably less than the number of votes, so perfect proportionality is not possible.

We formulate an optimization model for finding the feasible integer solution that is as proportional as possible. The objective function is based on a measure of disproportionality, and different such measures are discussed. We then show how one can solve the problem optimally. By computational tests, both on small artificial instances, and real life data, we verify that our method gives better solutions than other known methods (as measured by the least square difference).

We have not found our optimization model in the literature, nor our solution method. However, the general approach, separable programming, is clearly not new. We do not claim that our method gives different results from all other methods in all cases. Actually we will show that without a parliament threshold it gives the same result as Hamilton's method. However, with a threshold, which is the case we consider, there is a difference. Furthermore we have not found any other method in the literature where the objective function can be replaced as easily as in ours.

We also wish to point out that the word "optimal" in the title means that we will actually obtain an *optimal* solution to our model, not only a near-optimal one.

In principle, all (democratic) countries have different procedures for mandate allocation. In Sweden today, 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, in principle more than a hundred years old. See section 6 for details. Our work is aimed at Swedish circumstances, and will therefore make assumptions based on that. However, many countries use similar procedures, so hopefully this work may be of interest in many countries.

In this paper we first ignore constituencies, i.e. see all of the country as one constituency. In Sweden there are compensatory mandates and procedures for allocating them introduced to eliminate the difference that arises because of the constituencies. The goal is to get a representation that is as proportional as possible for the whole nation.

However, sometimes the number of compensatory 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 compensatory mandates and the procedure for allocating them.

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 entitled to vote. There can therefore not be any misrepresentation due to outdated data. This removes the need for fixed and compensatory mandates, and also makes the question about sizes of the constituencies less important.

We should mention that this paper proposes a procedure, based on sound mathematics, that we find natural and appealing. There are of course political and social aspects that are not dealt with in this paper, that could make our approach less appealing, and may motivate changes in the procedure, maybe especially concerning constituencies. This paper contains two proposals, the first is to use our method for the whole nation, and the second is to use our approach for the constituencies. The first part could easily be introduced into the current framework, without the second part. In any case, we do feel that this paper fills a gap in the literature, as it clearly formulates an optimization model and shows how to get the optimal solution to it. Hopefully this may lead to a better procedure for allocating mandates, without the need for mending and patching.

This paper is an updated, much extended and slightly corrected version of Holmberg (2019), where the ideas were first presented.

2 Previous work

The idea of democracy and proportional representation is obviously very old. However, we are mainly interested in work based on mathematics with some contents of optimization. A complete survey of the area is well beyond the scope of this paper. For those that read Swedish, we recommend the very extensive work Janson (2014), which covers most mathematical aspects of proportional election systems, and has a very comprehensive reference list. Other general works in this area are Gallagher and Mitchell (2005), Maunula (2008), Brandt, Conitzer, Endriss, Lang, and Procaccia (2016), Pukelsheim (2017) and Ricca, Scozzari, and Serafini (2017).

In this paper we will focus on the specific situation that is relevant for us, which is described in section 3. The main point is that each voter casts a vote for *one* party, and the allocation of mandates for parliament should be as proportional as possible for the parties. We will not go into systems where votes are cast on persons rather than parties, such as the system called Single Transferable Vote, or so called Multiwinner and Committee Scoring systems, where each voter is either giving an ordered preference list of the candidates, or a yes-no answer for the approval of each candidate. There is more work done for these situations, and a few references are Potthoff and Brams (1998) (which actually uses integer programming), Betzler, Slinko, and Uhlmann (2013), Brill, Laslier, and Skowron (2016), Faliszewski, Skowron, Slinko, and Talmon (2018a) and Faliszewski, Slinko, Stahl, and Talmon (2018b).

More interesting references for us are Anthonisse (1984), which uses a network flow formulation with piecewise linearization, solved as minimum cost flow problem, Ricca et al. (2017), which also use a piecewise linearization, Gaffke and Pukelsheim (2008), which describe algorithms based on rounding in a vector and matrix problem setting, and Benoit (2000), Aleskerov and Platonov (2004) and Karpov (2008), where several

measures of disproportionality are investigated.

We might also mention some investigations in specific countries and elections, such as Laakso and Taagepera (1981), discussing whether or not to introduce a threshold in Finland, Hylland (2010), about election results in Norway, and Linusson and Ryd (2012), discussing improvements of the Swedish system with fixed and compensatory mandates.

3 The framework

We consider an election with n parties. Each voter votes for one party, and the mandates in the parliament should be 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. In the election 2018 in Sweden, $p = 6476725$ and $m = 349$, which gives $d = 0.0000538$ and $1/d = 18557$.

Let r_j specify how many votes party j received in the election (so $p = \sum_j r_j$). We will work with the variables x_j as the number of mandates party j will obtain.

3.1 Measures

A solution with perfect proportionality would be $x_j = dr_j$ for all j . Then the proportions of the mandates would coincide exactly with the proportions of the votes. However, dr are seldom integral. Then we need to find an integer solution x that lies “as close as possible” to dr . In order to choose among possible such solutions, we need a measure to compare the solutions with. One often chooses to minimize a measure of “disproportionality”, which is based on the difference between x and dr .

In Gallagher (1991), Benoit (2000), Aleskerov and Platonov (2004) and Karpov (2008) several such measures are compared. One measure is the Loosemore-Hanby index, Loosemore and Hanby (1971), where the target function is $f(x) = \sum_{j=1}^n |x_j - dr_j|$ (divided by 2). (We will use the notation LH-index.) It is simply the sum of deviations, regardless of sign. In Gallagher (1991) some disadvantages of this measure are described, based on certain paradoxes that might appear.

Another measure, in Gallagher (1991) called the Least Square index, but later called the Gallagher index, is $\sqrt{v/2}$ where $v = \sum_{j=1}^n |x_j - dr_j|^2$. (We will use the notation G-index.) It is the sum of squared deviations of the absolute values. Compared to the LH-index, the G-index punishes larger deviations more.

In a third measure, the Sainte-Laguë index, the terms are weighted with the proportion of votes, $f(x) = \sum_{j=1}^n |x_j - dr_j|^2/r_j$. (We will use the notation SL-index.) It means that we compare relative deviations, so one unit’s error is worse for a party with fewer votes. It can equivalently be expressed as $f(x) = \sum_{j=1}^n r_j |x_j/r_j - d|^2$, so here the

relative proportions of mandates to votes for each party should be close to the overall relative proportion d . In other words, the number of mandates per vote should be as similar as possible.

A generalization of the first two measures is to use the p -norm, $f(x) = \left(\sum_{j=1}^n |x_j - dr_j|^p \right)^{1/p}$. Setting $p = 1$ yields the LH-index while $p = 2$ yields the G-index. Letting $p \rightarrow \infty$, we get the max-norm, $f(x) = \max_j |x_j - dr_j|$.

One might actually use any measure with the following desirable properties (satisfied by the functions above):

1. If dr is integer, the solution $x = dr$ should give perfect proportionality, i.e. index value zero.
2. The function should be separable in j . Moving a mandate between two parties should not affect a third party.
3. The function should be convex. Increasing the deviation from dr_j should always give a marginal increase of the cost.

This allows us to write $f(x) = \sum_j f_j(x_j)$, where each $f_j(x_j)$ is a convex function. The magnitudes of the indices are different, so one should not compare different measures, only the same measure for different solutions.

Which measure is best? Here different opinions are given in the literature. Let us now give our reasoning and our conclusion. Let us first recall that the purpose of the function is to evaluate and compare deviations from dr_j over j .

1. We prefer a strictly convex function, since it has better controllability. A function where many different solutions have the same value is not good. This is satisfied by all measures except the LH-index and the max-norm. Our computational tests with the LH-index will illustrate this.
2. A certain deviation should have the same measure regardless if it is a vote for a small party or a large party. This is satisfied by all measures except the SL-index.

Let us discuss point 2 more. The argument that the number of mandates per vote should be as similar as possible has been put forward. However, this only means that x_j/r_j should be close to d , which is something all the indices aim at.

The question is what happens when there is a deviation. Let us assume that $x_j - dr_j = 0.3$, and look at the influence on the index value from party j only. The LH-index then is 0.3 and the G-index is 0.387, but the SL-index depends on the size of the party. If $r_j = 10$, then the SL-index is 0.009, but if $r_j = 10000$, then SL-index is 0.000009. We see that this deviation gives much larger effect for a small party than a large party. It means that a certain deviation for one vote for a smaller party may give the same value as the same deviation for many votes for a larger party.

One may argue that 0.3 votes is 3% for the small party, but only 0.003% for the large party, but we think that this is irrelevant. It is the absolute numbers of votes/mandates that matters, not relative numbers. Furthermore, the system should not primarily be designed with the goal to treat all parties equally (by relative measures), but to treat

all votes equally.

We thus believe that absolute deviations should be compared, not relative deviations. It is 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. If a small and a large party are collaborating, and we consider the sum of votes/mandates for the two parties, one mandate more or less for the large party has the same practical effect as one mandate more or less for the small party, and should therefore have the same measure.

In section 10.5, we give a numerical example where a certain effect (moving a mandate from one party to another) requires a larger change in votes for a large party than for a small party. This is not a good property.

Our conclusion is therefore that the G-index is the most relevant measure of the deviations. An added plus is the well-known interpretation as a distance, which one often has in mind when saying the something should be “as close as possible” to something. The LH- and SL-indices have certain disadvantages that don’t make them our first choice. However, we wish to point out that any of these measures can be used in our model and in our method. An important aspect of this paper is that it gives a framework for finding optimal proportional representations, for any separable, convex function.

4 An optimization model

The integer solution that lies as close as possible to perfect proportionality in a least square meaning 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.

Since the objective function is based on the G-index, we might use the notation P1-G for this problem. In a similar manner, we can formulate P1-SL, using the SL-index in the objective function, and P1-LH, using the LH-index.

It is important to understand that it is the integrality requirements in the model that makes it difficult. The continuous relaxation of P1, denoted by CP1, is obtained by relaxing the integer requirements. It is a convex and easy problem, and is discussed in section 5.1. Integrality is not a linear, monotone or even continuous mapping of indata. For a continuous problem, small differences in indata often give small changes in the

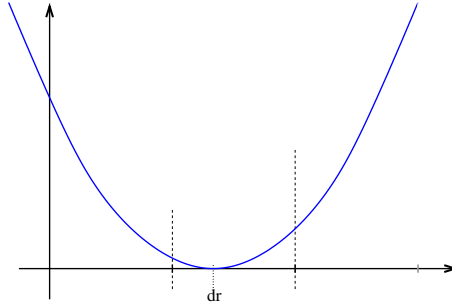


Figure 1: The objective function of P1-G.

solution. That is not the case for integer problems.

In figure 1 we give an example of how $f_j(x_j) = (x_j - dr_j)^2$ may look. The best value, dr_j , is not integral, so the solution must be one of the integer points, indicated by vertical dashed lines.

5 A threshold

Often there is an explicit parliament threshold, 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.

Can we 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 seats, so the smallest party in parliament will have more than 12 seats.

Convergence proofs for older methods generally do not include thresholds.

5.1 The continuous solution

It is clear that $x_j = dr_j$ is the optimal solution of the continuous relaxation of P1, CP1, if there is no lower threshold.

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 threshold. We then get $x_j = 0$ for $j \notin J$. Let $\pi = \sum_{j \in J} r_j$, i.e. π is the total number of votes given to parties that do not fall below the threshold. Then $w = p - \pi$ votes were given to parties that fall below the threshold and will not get any mandates. One might call them “wasted” votes, but they are not meaningless, as we will show. (Is there a difference between voting for a small party and not voting at all?)

Under the assumption that all parties having more than lp votes will receive at least one mandate, 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. $d\pi - qu/2 = m$, where $q = |J|$, i.e. the number of parties that do not fall below the threshold. This gives $u = 2m(\pi - p)/pq = -2mw/pq$. Letting $\delta = -mw/pq$, we get $x_j = dr_j - u/2 = dr_j + \delta$. The conclusion is that all values for the remaining variables will be increased by the same amount, δ , so that the sum becomes equal to m . This way CP1 can be explicitly and exactly solved.

Let us now do the same for CP1-SL, i.e. use the SL-index. Then the KKT conditions give $x_j = dr_j - ur_j/2$ for all $j \in J$. Now $\sum_{j \in J} x_j = m$ yields $\sum_{j \in J} (dr_j - ur_j/2) = m$, i.e. $d\pi - u\pi/2 = m$. This gives $u = 2m(\pi - p)/p\pi = -2mw/p\pi$. Letting $\sigma = p/\pi$, we get $x_j = dr_j - ur_j/2 = \sigma dr_j$. The conclusion now is that all values for the remaining variables will be multiplied by the same amount, σ , so that the sum becomes equal to m .

Thus the continuous relaxations of P1-G and P1-SL can be explicitly solved. (We cannot use the KKT conditions for CP1-LH, since the objective function is not differentiable.) We note the significant difference that using the G-index, there is a common additive adjustment of the values, while using the SL-index there is a common multiplicative adjustment. In other words, the G-index leads to changes in absolute values while the SL-index leads to changes in relative values.

Unfortunately, the integer problem P1 is more complicated when there is a lower threshold. Especially we cannot adjust the values with the same amount for all parties in J .

5.2 Wasted votes?

An interesting question is how a threshold influences disproportionality. Convergence proofs based on arithmetic series rarely hold when there is threshold. Sometimes one starts by removing all the parties that fall below the threshold, and their votes, from consideration. If that is done, there is no difference between voting for a very small party and not voting at all.

We believe that all votes should be considered when calculating the disproportionality measure. Let us use a small, unrealistic example for motivation. Suppose that there are two parties, two mandates and three voters, and the threshold is 40%. Party A get two votes and party B one, i.e. $r = (2, 1)$. Since party B gets less than 40% of the votes, it gets no mandate. Party A gets both. If the vote for party B is removed, then party A gets all votes, and all mandates, which gives index value zero, i.e. perfect proportionality. If the vote is not removed, we get $d = 2/3$, and $dr = (4/3, 2/3)$, while $x = (2, 0)$. The LH-index then becomes $2/3$, the G-index $2/3$, and the SL-index $6/9$. The voter who voted for party B is likely to think that these measures reflect the situation better than stating perfect proportionality.

Clearly a threshold will increase disproportionality, but this is a fact, and should not be hidden.

Suppose that we have w “wasted” votes, i.e. votes that does not give any mandates. (Note that even though the threshold is lp for single parties, the sum might exceed lp .) It is clear that these small parties will get no mandates, but can the allocation of mandates to the other parties be affected by these votes? For the continuous solutions, the answer is no. After removing these parties and votes, no remaining party will fall below the threshold, and the situation will be as if there was no threshold. One should however note that the parameters p and d will change.

For the integer solution, it is more complicated. There will be a difference between the best continuous solution and the integer solution at the optimum, and the question is how this difference affects the objective function value.

Let us compare the following two cases:

1. The parties and votes are kept.
2. The parties and votes are removed.

Removing the parties and votes from consideration will decrease n and p , which will increase d .

We will look at the slope of $f_j(x_j)$ for $j \in J$, which is obtained by the derivative of $f_j(x_j)$, as $f'_j(x_j) = 2(x_j - dr_j) = 2x_j - 2dr_j$.

Let us first study a very small example: $n = 3$, $m = 3$, $r = (5, 4, 1)$ and $l = 0.2$. (In words, party A got 5 votes, party B got 4 and party C got one.) This yields $p = 10$ and $d = 0.3$ if we keep all votes. We get $dr = (1.5, 1.2, 0.3)$. The optimal solution of P1-G is $x = (2, 1, 0)$. (P1-SL and P1-LH give the same solution.)

This means that $x - dr = (0.5, -0.2, -0.3)$. The slopes are $f'_1(2) = 1$, $f'_2(1) = -0.4$ and $f'_3(0) = -0.6$.

If we remove party C and its vote, we get $p = 9$ and $d = 0.333$, and $dr = (1.667, 1.333)$. The optimal solution is (still) $x = (2, 1)$. This means that $x - dr = (0.333, -0.333)$. The slopes of the objective function at these points are $f'_1(2) = 0.667$ and $f'_2(1) = -0.667$.

We see that the slopes have changed. The slope for party A is less positive, while the slope for party B is more negative, and the changes have different size. Since the slopes change, it is possible that the optimal integer solution will change. In section 10.4, we give a numerical example where the optimal solution actually changes.

If we instead use the SL-index, with derivative $f'_j(x_j) = 2(x_j - dr_j)/r_j = 2x_j/r_j - 2d$, we get the following slopes. With all votes left, we get $f'_1(2) = 0.2$, $f'_2(1) = -0.1$ and $f'_3(0) = -0.6$. If we remove party C and its vote, we get $f'_1(2) = 0.1333$ and $f'_2(1) = -0.1667$. We see that both slopes are decreased by 0.0667, which means that the difference is constant, and the solution will not change.

Let us now look at the slopes in general in the two cases. If we keep all votes, we get p votes and $d^1 = m/p$. If we remove the votes, we get π votes and $d^2 = m/\pi = d^1 p/\pi$. For the G-index, we get $f_j^1(x_j) = 2x_j - 2d^1 r_j$ in the first case, and $f_j^2(x_j) = 2x_j - 2d^2 r_j = 2x_j - 2d^1 r_j p/\pi = (2x_j - 2d^1 r_j) + 2d^1 r_j - 2d^1 r_j p/\pi = f_j^1(x_j) + 2d^1 r_j(1 - p/\pi) = f_j^1(x_j) - 2d^1 r_j w/\pi$, in the second case, so a factor, $2d^1 w/\pi$, times r_j is subtracted.

For the SL-index, we get $f_j^1(x_j) = 2(x_j - d^1 r_j)/r_j = 2x_j/r_j - 2d^1$ in the first case, and $f_j^2(x_j) = 2(x_j - d^2 r_j)/r_j = 2x_j/r_j - 2d^2 = 2x_j/r_j - 2d^1 p/\pi = (2x_j/r_j - 2d^1) + 2d^1 - 2pd^1/\pi = f_j^1(x_j) + 2d^1(1 - p/\pi) = f_j^1(x_j) - 2d^1 w/\pi$, in the second case, so a common factor, $2d^1 w/\pi$, is subtracted.

We find that the slopes for the SL-function are changed by the same amount, so the order of two slopes will never change. However, for the G-function, this is not the case. The conclusion is that the number of mandates for the other parties might be affected by removing the votes for parties that falls below the threshold, if we use the G-index, but not if we use the SL-index.

If we prefer the G-index, the conclusion is that we should not remove such votes. On the other hand there is no gain in removing them, the optimization problem is not notably harder to solve if we keep them.

6 The present method in Sweden

In Sweden, the mandates are distributed according to “the adjusted odd number method”, (a straightforward translation of the Swedish name “jämkkade uddatalsmetoden”). It is specified in legal text, namely chapter 14, paragraph 3, in Sweden’s election law. It can also be called the modified Sainte-Laguë method or the modified Webster/Sainte-Laguë method. The (unmodified) Sainte-Laguë method was proposed 1910 by Sainte-Laguë. In 1832 Webster proposed a method that yields the same result, although it was described differently.

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 values are initially equal to r , the number of votes the party got.

The adjustment/modification is to divide the initial values of v by a certain coefficient. In the election in Sweden 2018, the coefficient was 1.2, but before that 1.4. The effect of the adjustment is that the first mandate for each party is somewhat delayed, so it gives a disadvantage for smaller parties.

Algorithm 1 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. It aims at minimizing the Sainte-Laguë index. It has been motivated the following way. The method

chooses the maximal $r_t/(2s_t + 1)$ in each step, i.e. the minimal $(2s_t + 1)/r_t$. The objective function can be written

$$f(x) = \sum_{j=1}^n r_j \left(\frac{x_j}{r_j} - d \right)^2 = \sum_{j=1}^n \frac{x_j^2}{r_j} - md = \sum_{j=1}^n \sum_{k=1}^{x_j} \frac{2k-1}{r_j} - md, \text{ since } x_j^2 = \sum_{k=1}^{x_j} (2k-1).$$

We note that k starts at 1, while s_t starts at 0. Therefore we see that the method chooses the best m of these terms.

However, this proof does not include the adjusted factor 1.2, or a threshold. A threshold may remove certain of the best m terms from the sum $\sum_{k=1}^{x_j} (2k-1)$, in which case it will not be equal to x_j^2 . In any case, the existence of this method might have increased interest for the SL-index, at the expense of the G-index. We also note that parties below the threshold do not enter the calculations at all, so the question of whether or not to keep those votes is not very interesting if this method is used.

If the initial factor 1.2 is replaced by one, the method is simply called the Sainte-Laguë method or the “odd number method” (“uddatalsmetoden” in Swedish). A threshold, 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’Hondt’s 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.

A technique occasionally used to avoid the disadvantages for smaller parties 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 threshold, and also to avoid effects of the initial factor 1.2. Furthermore, it changes the sizes of parties, so that a small party becomes (a part of) a large party. This has effect on the SL-index, since a large and a small party are treated differently, as we have shown. For the G-index, there is no such difference (although integrality may give small random effects).

7 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 thresholds at this stage. We will consider thresholds later.)

7.1 Piecewise linearization

To solve the integer problem P1, we first note that the objective function is convex and 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. In figure 2 we show the piecewise linearization.

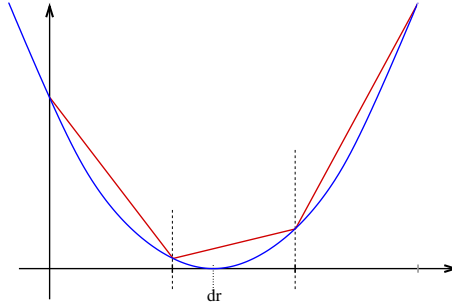


Figure 2: Piecewise linearization of the objective function of P1-G.

We can calculate coefficients representing the slope of the objective function between two adjacent integer points by the following expression.

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

We might use the notation c_{jk}^G , since we used the G-index. These coefficients can be calculated from any separable, convex function, and we will later give the expressions when using the SL-index and the LH-index. We will use the notation c^G , c^{SL} and c^{LH} when the difference between them is important. However most results hold for any separable, convex function, and then we will not use any superscript on c .

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 optimal 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, since the coefficients in constraint (1) are all one, and m is integer, which means that solving P3 produces an integer solution 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, since m is integer and we in each iteration subtract one from it.

We have integer variables represented by several binary variables, $x_j = \sum_k x_{jk}$ with costs c_{jk} and wish to increase the best x_j . Let the current value of x be denoted by \hat{x} . Since the cost function is convex, this means that $x_{jk} = 1$ for all $k \leq \hat{x}_j$ and $x_{jk} = 0$ for all $k > \hat{x}_j$. Now the question is if we should increase x_j to $\hat{x}_j + 1$ or not, i.e. if we should set $x_{jk'} = 1$ for $k' = \hat{x}_j + 1$. For this reason we compare $c_{jk'}$ over all j .

Let us now specify the algorithm. We denote the number of allocated mandates 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 $v_j = c_{jk_j} = 2(k_j - dr_j) - 1$. (We will later use the notation v^G , v^{SL} and v^{LH} , when it is important which measure we are using.) This yields $v_j = 2(\hat{x}_j + 1 - dr_j) - 1 = 2(\hat{x}_j - dr_j) + 1$. 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 threshold l is simply taken care of by not allowing any mandates to parties with less votes than this. In the model P3, some variables are simply set equal to zero. This does not affect the validity of the method. It is however necessary, as noted earlier, to include the votes for such parties in the calculation of the coefficients.

7.2 The new 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 threshold.

Note that this actually gives the optimal solution, i.e. the feasible solution that has the

Algorithm 2 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 $v_t = 2(s_t - dr_t) + 1$.
-

minimal disproportionality, in P3, P2 and P1, for all the measures described.

In general it is not necessary to understand an algorithm in order to use it, it is sufficient if it is correct. However, this algorithm is simple enough to allow some intuitive interpretation. In each iteration, one new mandate is handed out, and the question is which party shall get it. For each party we calculate the change in disproportionality (the cost function) if it was given the mandate. As long as s_j is less than dr_j (see step 5), this cost is actually a gain, i.e. a negative coefficient, and in each iteration the mandate is given to the party that gives the maximal decrease of the disproportionality.

7.3 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 will calculate the corresponding dual solution, and show that both solutions are optimal. Let us first restate P3 with an explicit threshold. We start by calculating $J = \{j : r_j \geq lp\}$, which is the set of parties that will get votes.

$$\begin{aligned} \min \quad z &= \sum_{j=1}^n \sum_{k=1}^m c_{jk} x_{jk} \\ \text{s.t.} \quad & \sum_{j=1}^n \sum_{k=1}^m x_{jk} = m & (1) \\ & x_{jk} \leq 1, \text{ for all } j, k & (2) \\ & x_{jk} \leq 0, \text{ for } j \notin J, k & (3) \\ & x_{jk} \geq 0, \text{ for all } j, k & (4) \end{aligned} \tag{P4}$$

The algorithm yields a 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 , and also $\sum_{j,k} x_{jk} = m$. 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), β for constraints (2), and γ for constraints (3) is as follows.

$$\begin{aligned}
\max \quad v = & \quad m\alpha - \sum_{j=1}^n \sum_{k=1}^m \beta_{jk} \\
\text{s.t.} \quad & \quad \alpha - \beta_{jk} \leq c_{jk}, \text{ for all } j, k \\
\text{s.t.} \quad & \quad \alpha - \beta_{jk} - \gamma_{jk} \leq c_{jk}, \text{ for } j \notin J, k \\
& \quad \beta_{jk} \geq 0, \text{ for all } j, k \\
& \quad \gamma_{jk} \geq 0, \text{ for all } j \notin J, k
\end{aligned} \tag{D4}$$

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

For $j \notin J$, we have $x_{ij} = 0$, so complementary slackness yields $\beta_{jk} = 0$. For $j \in J$ and $k > s_j$, we also have $x_{jk} = 0$, and subsequently $\beta_{jk} = 0$. This means that the dual constraints reduce to $\alpha \leq c_{jk}$ for $j \in J$ and all $k > s_j$, and to $\alpha - \gamma_{jk} \leq c_{jk}$ for $j \notin J$ and all k . For each $j \notin J$ γ_{jk} can be increased without affecting the objective function or any other constraint, so the constraints containing γ can always be satisfied.

For $j \in J$ and $k \leq s_j$, we have $x_{ij} = 1$, so complementary slackness yields $\alpha - \beta_{jk} = c_{jk}$. This can be written as $\beta_{jk} = \alpha - c_{jk}$, and since $\beta_{jk} \geq 0$, we must have $\alpha - c_{jk} \geq 0$, i.e. $\alpha \geq c_{jk}$.

We now have the following bounds on α : $\alpha \leq c_{jk}$ for $j \in J, k > s_j$, and $\alpha \geq c_{jk}$ for $j \in J, k \leq s_j$. Since c is increasing in k , this simplifies to $\alpha \leq c_{js_j+1}$ and $\alpha \geq c_{js_j}$ for $j \in J$. An optimal value of α is thus any value satisfying $\max_{j \in J} c_{js_j} \leq \alpha \leq \min_{j \in J} c_{js_j+1}$.

In words we can say that the lower bound is the highest cost for an x -variable that is used, while the upper bound is the lowest cost for an x -variable that is not used.

Given the value of α , we calculate $\beta_{jk} = \alpha - c_{jk}$ for $j \in J$ and $k \leq s_j$. Now we have a complete dual solution that is feasible in D4 and satisfies complementary slackness. (As mention above, the value of γ is uninteresting.) The dual 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, so they are optimal solutions. One may note that this proof holds for any c that is convex in k , and that a threshold is included in the proof.

An advantage of having an LP-problem is that LP-theory applies, with local optimality criteria (the reduced costs) and possibilities for sensitivity analysis. One can for example start from a given solution and check if it is optimal. If it is not optimal, one can find out which party should get more mandates.

The reduced costs are $\hat{c}_{jk} = c_{jk} - \alpha + \beta_{jk}$ for $j \in J$. If $\hat{c}_{jk} \geq 0$ for all k and all $j \in J$, the solution is optimal. Otherwise, $\hat{c}_{jk} < 0$ indicates that x_{jk} should be increased.

7.4 Lagrangean duality

We can apply Lagrangean duality to P3. The Lagrange dual is $\max_u \varphi(u)$, where

$$\varphi(u) = \min_{0 \leq x \leq 1} \sum_{j=1}^n \sum_{k=1}^m c_{jk} x_{jk} + u \left(\sum_{j=1}^n \sum_{k=1}^m x_{jk} - m \right) \min_{0 \leq x \leq 1} \sum_{j=1}^n \sum_{k=1}^m (c_{jk} + u) x_{jk} - um$$

The optimal solution is to set $x_{jk} = 1$ if $c_{jk} + u < 0$ and $x_{jk} = 0$ if $c_{jk} + u > 0$, and either value if $c_{jk} + u = 0$. A dual ascent method for solving this could be as follows.

Algorithm 3 Lagrangean dual ascent

- 1: Set $u = 0$.
 - 2: **repeat**
 - 3: Set $\bar{c}_{jk} = c_{jk} + u$ for all j, k .
 - 4: **for** $j \in J$ and all k **do**
 - 5: **if** $\bar{c}_{jk} \leq 0$ **then**
 - 6: set $x_{jk} = 1$.
 - 7: **if** $\bar{c}_{jk} > 0$ **then**
 - 8: set $x_{jk} = 0$.
 - 9: Find c^n : maximal negative element in \bar{c} .
 - 10: Find c^p : minimal positive element in \bar{c} .
 - 11: Set $\xi = \sum_{j,k} x_{jk} - m$.
 - 12: **if** $\xi = 0$ **then**
 - 13: stop.
 - 14: **if** $\xi > 0$ **then**
 - 15: set $u = u - c^n$.
 - 16: **if** $\xi < 0$ **then**
 - 17: set $u = u - c^p$.
 - 18: **until** $\xi = 0$.
-

Comments: $c^n < 0$, so u is increasing in step 15. $c^p > 0$, so u is decreasing in step 17. $\xi = 0$ means that the correct number of mandates have been allocated. This method will find the dual maximum, and since the problem is convex, there will be no duality gap. Furthermore there will be no lack of controllability.

Without giving any proof, we claim that this method gives the same solution as algorithm 2. Both solves P3 to optimality. Actually one could use any correct method for solving P3, such as the simplex method.

The point of this observation is that one actually has a wider choice of method, if one should wish. Furthermore, all known facts about such methods and solutions apply. It also makes it easier to “repair” optimality, i.e. start with a near-optimal solution and find the optimal solution. It might also be easier to add certain complications and still get the optimal solution.

On the other hand, algorithm 2 is probably the easiest and quickest way of finding the optimal solution, so we suggest to use that in practice, even if some properties of the optimal solution coming from other possible methods are used.

7.5 Relation to previous methods

Traditional methods for allocating mandates can be classified as “largest average methods” (“divisor methods”) or “largest remainder methods”, where the Sainte-Laguë method and d’Hondt’s method are divisor methods. In for example Janson (2014), many properties of these classes of methods are given.

An example of a largest remainder method is Hamilton’s or Hare’s method (1860). The initial allocation of mandates is simply $s_j = \lfloor dr_j \rfloor$ (i.e. the continuous solution rounded down) and then the additional mandates are allocated one by one to the party with largest fraction, $s_j - \lfloor dr_j \rfloor$, until m mandates have been allocated.

In algorithm 2, mandates are allocated based on $\min 2(s_j - dr_j) - 1$, which gives the same result as $\min s_j - dr_j$, or $\max dr_j - s_j$. This can be seen as choosing the m largest terms of $dr_j - k$ over all j and k . (Here the convexity is important, making sure that $(j, k - 1)$ is always chosen before (j, k) .)

While $s_j < dr_j$, the algorithm will decrease the maximal distance from s to dr , until $s_j = \lfloor dr_j \rfloor$ (or in other words $dr_j - s_j < 1$). If $\hat{m} < m$, more mandates need to be allocated. Then $\max dr_j - s_j$ means the maximal fraction, so our method is in principle a largest remainder method. If there is no threshold, it gives the same solution as Hamilton’s method.

However, if there is a threshold, the solution might not be the m largest terms of $dr_j - k$ over all j and k . In that case our method and Hamilton’s method may give different solutions, and we will give examples of big such differences in the computational tests.

7.6 Other measures

An advantage of the approach of first formulating an optimization problem and then solve it, is that we can easily change the objective function. We can use the G-index, as we have done, or the SL-index, or even the LH-index, as objective function. The effect is only that the coefficients c and values v are calculated differently. It is also possible to include other measures, if one should wish. We note that it is not necessary for our method that the objective function is differentiable.

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.

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

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

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

The cost curve of c^{LH} is thus convex, but not strictly convex. It is linear in all but two points, and this is, as we shall see, an undesirable property.

Using the general p -norm, we get $c_{jk}^N = (k - dr_j)^p - (k - 1 - dr_j)^p$.

For max-norm, we get $c_{jk}^M = \max(k - dr_j, k - 1 - dr_j)$.

If we believe that several of the measures have merit, we can actually combine them. We can for example use the objective function $f(x) = w_1(\sum_{j=1}^n (x_j - dr_j)^2) + w_2(\sum_{j=1}^n (x_j - dr_j)^2/r_j) + w_3(\sum_{j=1}^n |x_j - dr_j|) = \sum_{j=1}^n (w_1(x_j - dr_j)^2 + w_2(x_j - dr_j)^2/r_j + w_3|x_j - dr_j|)$, where w_1 , w_2 and w_3 are the (nonnegative) weights we wish to put on the different measures, and are chosen in advance. One should however have in mind the different magnitudes of the measures, and compensate for this with the values of w . We will give an example of this in the section of computational results. In model P3, the objective function coefficients are simply calculated as $w_1c^G + w_2c^{SL} + w_3c^{LH}$, and the values as $w_1v^G + w_2v^{SL} + w_3v^{LH}$. We note that $v_j^{SL} = v_j^G/r_j$.

7.7 Nonuniqueness

A question raised by mathematicians is what to do if the minimizer in an iteration in our method is not unique. In practice, we expect that this issue will never arise, due to the large number of votes, but let us in any case look at it.

First of all we note that the verification of optimality in section 7.3 does not require uniqueness.

If two parties get exactly the same number of votes, there is no way to distinguish between them. In that case they should preferably get the same number of mandates. If that is not possible, there is no other option but to use a random allocation of the final mandate.

However, it is possible that the minimal value v_j in an iteration is not unique, even though the number of votes are different, so let us look at this case.

In the method presently used in Sweden, it is stated that randomness should be used in each case the choice is not unique. There is however another possibility, with a better motivation.

We consider a tie where the minimal value of v is obtained by two parties, j_1 and j_2 , i.e. $v_{j_1} = v_{j_2}$, but the number of votes are not equal, i.e. $r_{j_1} \neq r_{j_2}$. To be able to handle such a case, we suggest, in principle, to use an objective function based on the G-index plus ε times the SL-index, where ε is a very small positive number. This means that $v_j = v_j^G + \varepsilon v_j^{SL}$.

If $v_{j_1}^G \neq v_{j_2}^G$, we assume that ε is so small that the SL-index will have no influence, i.e. will not change which value v is the smallest. In practice this means that we don't use the SL-index at all in this case.

However, if $v_{j_1}^G = v_{j_2}^G$, then the difference between the values will be $v_{j_1} - v_{j_2} = (v_{j_1}^G + \varepsilon v_{j_1}^{SL}) - (v_{j_2}^G + \varepsilon v_{j_2}^{SL}) = \varepsilon(v_{j_1}^{SL} - v_{j_2}^{SL}) = \varepsilon(v_{j_1}^G/r_{j_1} - v_{j_2}^G/r_{j_2}) = \varepsilon v_{j_1}^G(1/r_{j_1} - 1/r_{j_2})$.

If this is negative, $v_{j_1} < v_{j_2}$, party j_1 should get the mandate, and this happens if $1/r_{j_1} < 1/r_{j_2}$, i.e. if $r_{j_1} > r_{j_2}$. If the difference is positive, $v_{j_1} > v_{j_2}$, party j_2 should get the mandate, and this happens if $1/r_{j_1} > 1/r_{j_2}$, i.e. if $r_{j_1} < r_{j_2}$. Since we are assuming that $r_{j_1} \neq r_{j_2}$, this will always identify a unique party. We can say that this induces a kind of lexicographic ordering.

Summing up, we find that the value of ε does not matter. The method is simply as follows. Use Algorithm 2 with c^G as long as the minimizer is unique. If the minimizer is not unique, give the mandate to the party that has the highest number of votes (of those that gave the minimal v_j^G).

Giving the mandate to the party with highest number of votes seems like a very natural thing to do. One might even come up with this idea without any mathematical considerations. This idea (but together with another basic algorithm) has been used in Norway, Belgium and Luxembourg, according to Janson (2014). What is interesting here is that we have a good mathematical motivation for doing this.

8 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 entitled to vote 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. Compensatory 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 compensatory mandates and the number of compensatory mandates have been changed.

8.1 Allocation of mandates for constituencies

We here suggest a different approach. First of all, we say that the allocation for the nation as a whole (preferably found with Algorithm 2) 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 .

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

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 2 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 do not fall below a threshold, i.e. those constituencies where party

j may get votes. (The thresholds here may be different from the national level.)

Algorithm 4 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 $v_{\tau j} = 2(y_{\tau j} - dt_{\tau j}) + 1$.
-

This algorithm solves P4 exactly (just as Algorithm 2 solves P1).

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 that are entitled to vote. 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 compensatory 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.

There could be many other political and social aspects concerning constituencies, and our goal is not to try to settle that question here. We just want to point out a new possibility.

8.2 Lower bounds for constituencies

In the procedure used today, there are certain fixed numbers of mandates for the constituencies, which serves as lower bounds on the number of mandates for each constituency. In the previous section we did not use that, but let the number of mandates depend only on the number of votes. In other words, we assume that the allocation with respect to parties is more important than the allocation with respect to constituencies.

In section 10.10, we discuss a numerical example where the number of mandates allocated to a constituency is less than the number of fixed mandates. This happens for a very small constituency, when the votes are spread out over the parties, so that each party gets a very small amount, dt_{ij} . Then it is optimal to set most of the y_{ij} to zero, even though this makes $\sum_j y_{ij}$ less than $\sum_j dt_{ij}$. From a national proportional point of view this might still be the best solution.

However, there might be regional differences, making the numbers of mandates for the constituencies very important. In that case our optimization model is not perfect, as it does not include this aspect.

Let us first consider lower bounds on each party in each constituency, saying that party j must not get less than l_{ij} mandates in constituency i . Then we simply add the constraints $y_{ij} \geq l_{ij}$ to model P4. In order to efficiently solve the resulting problem, we do the substitution $y'_{ij} = y_{ij} - l_{ij}$ for all i and j . Then we have $y'_{ij} \geq 0$. We also get $\sum_i y_{ij} = \sum_i (y'_{ij} + l_{ij}) = \sum_i y'_{ij} + \sum_i l_{ij}$, so constraint (1) becomes $\sum_i y'_{ij} = s_j - \sum_i l_{ij}$. In the objective function, we get the terms $(y'_{ij} + l_{ij} - dt_{ij})^2$, and the coefficients in the piecewise linearization becomes

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

With these changes of indata, the problem can be solved with Algorithm 4. Afterwards, we get the solution as $y_{ij} = y'_{ij} + l_{ij}$. Thus individual lower bounds can be handled by our method without complications.

Let us now consider lower bounds, l_i , on the total number of mandates in each constituency. Then we need to add the constraints $\sum_j y_{ij} \geq l_i$ to our optimization model. Previously our optimization problem, P4, could be solved separately for each party, in order to find the allocation of that party's mandates over the constituencies. Now we introduce a constraint over the constituencies, so the problem does not separate in j as P4 did. We then have to solve the following problem, containing all constituencies and all parties.

$$\begin{aligned} \min \quad & \sum_{i=1}^q \sum_{j=1}^n (y_{ij} - dt_{ij})^2 \\ \text{s.t.} \quad & \sum_{i=1}^q y_{ij} = s_j \quad j = 1, \dots, n \quad (1) \\ & \sum_{j=1}^n y_{ij} \geq l_i \quad i = 1, \dots, q \quad (2) \\ & y_{ij} \geq 0, \text{ integer} \quad \forall i, j \end{aligned} \quad (\text{P6})$$

We can still do the same exact piecewise linearization, since the objective function still is additively separable in j , and obtain the following problem.

$$\begin{aligned} \min \quad & \sum_{i=1}^q \sum_{j=1}^n \sum_{k=1}^m a_{ijk} y_{ijk} \\ \text{s.t.} \quad & \sum_{i=1}^q \sum_{k=1}^m y_{ijk} = s_j \quad j = 1, \dots, n \quad (1) \\ & \sum_{j=1}^n \sum_{k=1}^m y_{ijk} \geq l_i \quad i = 1, \dots, q \quad (2) \\ & 0 \leq y_{ijk} \leq 1 \quad \forall i, j, k \end{aligned} \quad (\text{P7})$$

Note that, unlike the present procedure, the national numbers of mandates for the parties are not changed at all. No compensation mandates are needed.

We now have qnm variables, which for Sweden is slightly more than 100 000. It is not possible to solve P7 with a simple greedy method, as for P5. However, it can be solved with standard optimization methods for linear programming problems, of which there are plenty implementations. Computational tests with this problem is reported in section 10.11.

9 Implementation

The algorithms are implemented in Python. We have implemented our method, algorithm 2, denoted by KH (for lack of a better name), the modified Sainte-Laguë method, algorithm 1, denoted by mSL, (used for example in Sweden and Norway), d’Hondt’s method, denoted by dHo, (used for example in Finland, Belgium, Iceland, Israel, Netherlands, Spain and Austria), Hamilton’s method, denoted by Ha (used for example in Italy, Greece, Latvia, Denmark, Bulgaria and Cyprus), and Droop’s method without rounding, denoted by Dr, which is similar to Hamilton’s method, but with $d = (m + 1)/p$ instead of m/p , (used for example in Slovakia). In the tables we have sometimes multiplied the SL-index with a factor (for example 1000) just for readability.

The code is run in a terminal with the following command.

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

The first argument, votes-2018.txt, is which input file to use. The second argument, 0.04, is the threshold 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 thresholds and different numbers of mandates. The code is obtainable from the author on request.

It is also possible to give a fourth argument as the name of a file containing a suggested solution, i.e. mandates for all parties. In that case this solution is evaluated and the values of the indices reported.

The code can also calculate the dual solution as described in section 7.3. This can be used to check if a given solution is optimal, and indicate which party should get more mandates, if the solution is not optimal.

For solving P7, we used the free code GLPK. Our Python code reads indata, constructs and writes an input file for GLPK, solves the problem, and reads the output.

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. The votes for many insignificant parties are in this data reported under one name, “Others”. Usually the proportion of votes are below the threshold l , so no mandates are allocated. However, it is not impossible that the sum of votes for all the small parties exceeds the threshold. We must therefore deal with this item by not include it in J in Algorithm 2.

Party	r_j	dr_j	r_j/p	KH	mSL	dHo	Ha	Dr
The Administrators (A)	320	32.0	0.32	34	34	34	38	35
The Bureaucrats (B)	280	28.0	0.28	30	30	30	28	29
The Commoners (C)	260	26.0	0.26	27	28	28	26	27
The Different (D)	80	8.0	0.08	9	8	8	8	9
The Xenophobes (X)	30	3.0	0.03	0	0	0	0	0
The Yetis (Y)	20	2.0	0.02	0	0	0	0	0
The Zeptoparty (Z)	10	1.0	0.01	0	0	0	0	0
G-index				3.464	3.606	3.606	5.000	3.606
SL-index				0.643	0.642	0.642	0.713	0.648
LH-index				6.000	6.000	6.000	6.000	6.000

Table 1: Parties, votes and mandates for the first instance.

10 Computational results

10.1 One artificial instance

First we solve an artificial instance, initially used for debugging. We have 7 parties, 1000 votes and 100 mandates. The numbers 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 threshold, all algorithms give the solution $x_j = dr_j$, with the indices equal to zero. Also for $l = 0.03$, the algorithms KH and mSL give the same solution. In table 1, the results for $l = 0.04$ are given. There we find that our method gives a lower G-index than the others. The SL-index for KH and mSL are almost the same. The LH-index is exactly the same for all solutions, which indicates that it is not a good measure. It also turns out that the minimal index in step 3 in Algorithm 2 was not unique 67 times.

10.2 Additional small artificial instances

We have also solved some small test instances described in Linusson (2008), there used to illustrate the modified Sainte-Laguë method, algorithm 1. Here there are three parties, 700 votes and 7 mandates, and no threshold, i.e. $l = 0$. This gives 0.01 mandate per vote or 100 votes per mandate. In table 2, 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 Örketräsk 4 and Örketräsk 5. The last five columns give the number of mandates allocated by the algorithms, and the indices for the solutions.

For the first instance, KH and Ha give the solution with lowest G-index. For the second, all methods except mSL give the lowest G-index. On the other hand, mSL gives a lower SL-index.

Örkelträsk 4

Party	r_j	dr_j	r_j/p	KH	mSL	dHo	Ha	Dr
A	333	3.33	0.475	3	4	4	3	4
B	237	2.37	0.338	3	2	2	3	2
C	130	1.3	0.185	1	1	1	1	1
G-index				0.545	0.581	0.581	0.545	0.581
SL-index				0.269	0.262	0.262	0.269	0.262
LH-index				0.630	0.670	0.670	0.630	0.670

Örkelträsk 5

Party	r_j	dr_j	r_j/p	KH	mSL	dHo	Ha	Dr
A	367	3.67	0.524	4	3	4	4	4
B	267	2.67	0.381	3	3	3	3	3
C	66	0.66	0.094	0	1	0	0	0
G-index				0.571	0.580	0.571	0.571	0.571
SL-index				0.730	0.338	0.730	0.730	0.730
LH-index				0.660	0.670	0.660	0.660	0.660

Table 2: Votes and mandates for the second set of instances.

Party	r_j	dr_j	r_j/p	KH	mSL	dHo	Ha	Dr
A	2371	5.179	0.575	5	6	6	5	6
B	1274	2.783	0.309	3	3	3	3	3
C	245	0.535	0.059	1	0	0	1	0
D	230	0.502	0.056	0	0	0	0	0
G-index				0.523	0.794	0.794	0.523	0.794
SL-index				0.203	0.259	0.259	0.203	0.259
LH-index				0.682	1.038	1.038	0.682	1.038

Table 3: Mandates for the second Hungarian instance.

10.3 Other real life instances

In Benoit (2000) numerical examples from two Hungarian cities are described. In the first $r = [397, 394, 285, 224, 209, 172, 136]$ and $m = 7$. Here KH, mSL, Ha and Dr all give the solution $[2, 1, 1, 1, 1, 1, 0]$ with G-index 0.687, while dHo gives a solution with G-index 0.785. In the paper Benoit (2000) the Sainte-Laguë method is said to give the solution of one mandate to each party, which is said to be a better solution (i.e. less least squares disproportionality). However, when we test that solution, we get G-index 0.691, which is worse than the above. On the other hand, the SL-index is lower for that solution than for our solution.

Results for the second instance are reported in table 3. Here we find that KH and Ha give the best solutions (least G-index, 0.523), while the other methods give worse (G-index 0.794). An interesting fact here is that mSL gives a worse SL-index than KH, which might seem contradictory. It turns out that in this case it is the modification of the SL-method that is bad. Without the modification, (m)SL gives the same solution as KH.

A general conclusion in Benoit (2000) is that the choice of measure is very important, which we agree with. However, it is also stated that the Sainte-Laguë method is the most proportional method, which is different from our conclusions.

10.4 Removing small party votes

Here we give a small example to show the difference between keeping votes for small parties and removing them before allocation. We start by keeping all votes. We have $n = 3$, $m = 4$, $r = (70, 40, 19)$ and $l = 0.2$. This yields $p = 129$ and $d = 0.031$. We get $dr = (2.17, 1.24, 0.59)$. The optimal solution of P1-G is $x = (2, 2, 0)$. If we remove party C and its votes, we get $p = 110$ and $d = 0.036$, and $dr = (2.54, 1.45)$. The optimal solution is now $x = (3, 1)$. With all votes left, the solution $x = (2, 2, 0)$ has G-index 0.69 and SL-index 0.033, while $x = (3, 1, 0)$ has G-index 0.74 and SL-index 0.029. This shows that removing small parties and their votes can affect the allocation of mandates to the remaining parties.

If we instead solve P1-SL, we get the optimal solutions $x = (3, 1, 0)$ and $x = (3, 1)$. We see that the solution of P1-SL is not affected by removing small parties and their votes.

Let us give another small example, with $n = 11$, $l = 0.1$, $m = 10$. Table 4 gives two sets of numbers of votes for the parties and the results for the methods KH, mSL and Ha. In the first count one vote for party A was overlooked, but that was corrected in the second count. In both cases approximately 115 votes are needed for a mandate. We find that only party A and B are above the threshold, so the other 9 parties get no mandates. Keeping all votes yields the following. KH and mSL give the same solution, but we note that Hamilton's method is much affected by the threshold. The additional vote for party A does not make any difference for KH and Ha, but for mSL one mandate is moved from B to A. Since a small G-index is our goal, the solution of KH is the one we want.

If we remove all votes for small parties, there are only two parties left. Now only 26 votes are needed for a mandate. Here we see that mSL is unaffected by this, while the solution Ha is much improved. The three methods give the same solutions, but the added vote changes the solution. The value of the G-index for a certain solution is decreased very much by the removal of the small party votes.

The conclusion is that the result of KH and Ha can change if small party votes are removed, Ha gives a better solution, while KH gives a worse solution. Therefore we should keep all votes when using KH.

10.5 An example regarding the SL-index

The main difference between the G-index and the SL-index is that the latter compares relative changes. Let us now give a small example without a threshold that illustrates this.

Party	First count					Second count				
	r_j	dr_j	KH	mSL	Ha	r_j	dr_j	KH	mSL	Ha
A	146	1.26	5	5	8	147	1.27	5	6	8
B	120	1.04	5	5	2	120	1.04	5	4	2
C	99	0.85	0	0	0	99	0.85	0	0	0
D	99	0.85	0	0	0	99	0.85	0	0	0
E	99	0.85	0	0	0	99	0.85	0	0	0
F	99	0.85	0	0	0	99	0.85	0	0	0
G	99	0.85	0	0	0	99	0.85	0	0	0
H	99	0.85	0	0	0	99	0.85	0	0	0
I	99	0.85	0	0	0	99	0.85	0	0	0
J	99	0.85	0	0	0	99	0.85	0	0	0
K	99	0.85	0	0	0	99	0.85	0	0	0
G-index			4.258	4.258	5.144			4.255	4.344	5.138
SL-index			0.293	0.293	0.385			0.292	0.292	0.382
A	146	5.49	5	5	5	147	5.51	6	6	6
B	120	4.51	5	5	5	120	4.49	4	4	4
G-index			0.489	0.489	0.489			0.494	0.494	0.494
SL-index			0.363	0.363	0.363			0.370	0.370	0.370

Table 4: Illustration of wasted votes.

Consider three parties, A, B and C, and $m = 3$. In our base case $r = (100, 25, 20)$. Both KH and mSL gives the solution $x = (2, 1, 0)$. If we remove a vote for party B, i.e. have $r = (100, 24, 20)$, mSL gives solution $x = (3, 0, 0)$, i.e. moves one mandate from B to A.

If we instead increase the number of votes for party A, the question is how many votes need to be added, in order to get the same effect, namely move a mandate from A to B. Tests reveal that we need to add 5 votes to party A, i.e. have $r = (105, 25, 20)$, in order to make mSL give $x = (3, 0, 0)$.

The conclusion is that we need to add 5 votes to the large party A to get the same effect as removing one vote from the small party B. It seems that votes for small parties and large parties are treated differently by mSL.

Could this be an effect of the modification of the SL-method (the initial division by 1.2)? Since very few mandates are allocated, the modification might have a large effect. To check this, we did the same test with initial factor 1.0, i.e. the unmodified SL-method.

We started with the same base case as above, and got the same solution. In order to move one mandate from B to A, we needed to remove 5 votes from party B. If we instead add votes to party A, the solution was not changed until we had added 25 votes to party A. Again we see that votes for small parties matter more than votes for large parties, when using the SL-index.

Party	r_j	dr_j	KH	mSL	dHo	Ha	Dr	Res
M	1791766	104.91	106	106	107	107	107	107
C	390804	22.88	23	23	23	23	23	23
FP	420524	24.62	25	25	25	25	25	24
KD	33369	19.54	20	20	19	20	20	19
S	1827497	107.00	108	109	109	108	108	112
V	334053	19.56	20	20	20	20	20	19
MP	437435	25.61	26	26	26	26	26	25
SD	339610	19.88	21	20	20	20	20	20
Others	85023	4.98	0	0	0	0	0	
G-index			3.802	3.916	4.118	3.928	3.928	5.267
SL-index			0.298	0.296	0.298	0.296	0.296	0.311
LH-index			4.978	4.978	5.517	4.978	4.978	7.313

Table 5: Votes and mandates for the 2010 election in Sweden.

Party	r_j	dr_j	KH	mSL	dHo	Ha	Dr	Res
M	1453517	81.40	83	85	85	93	92	84
C	380937	21.33	23	22	22	22	22	22
FP	337773	18.92	21	20	19	19	19	19
KD	284806	15.95	18	17	16	16	16	16
S	1932711	108.24	110	112	114	109	109	113
V	356331	19.96	22	21	21	20	21	21
MP	429275	24.04	26	25	25	25	25	25
SD	801178	44.87	46	47	47	45	45	49
FI	194719	10.90	0	0	0	0	0	0
Others	60326	3.38	0	0	0	0	0	0
G-index			8.848	9.128	9.576	11.549	11.083	9.466
SL-index			0.860	0.835	0.838	0.896	0.884	0.844
LH-index			14.284	14.284	14.284	14.284	14.284	14.284

Table 6: Votes and mandates for the 2014 election in Sweden.

10.6 Sweden

In our main computational tests we have used the data from the three last elections in Sweden, see tables 5, 6 and 7. In these runs we used $l = 0.04$ and $m = 349$, as is the case in reality. (We have not included the Swedish party names, but only the abbreviations.) The last row, named “Others” is the sum of all minor parties.

The results from Algorithm 1 are not identical to the final real life allocations, due to some deficiencies in the allocation of compensatory mandates. Our main goal is to compare the algorithms, but we also give the actual mandate allocation in the last column, under Res.

In all these elections, KH gives the lowest G-index. For 2010 and 2014 all other methods are worse, and for 2018, Ha and Dr are equally good. Our method is better than mSL (which is used in practice) in all cases. When it comes to the SL-index, mSL and KH give very similar results for 2010 and 2018, while for 2014 mSL yields a lower SL-index.

Party	r_j	dr_j	KH	mSL	dHo	Ha	Dr	Res
M	1284698	69.22	70	70	71	70	70	70
C	557500	30.04	31	31	30	31	31	31
L	355546	19.16	20	20	19	20	20	20
KD	409478	22.06	23	22	22	23	23	22
S	1830386	98.63	99	100	101	99	99	100
V	518454	27.94	28	28	28	28	28	28
MP	285899	15.40	16	16	15	16	16	16
SD	1135627	61.19	62	62	63	62	62	62
FI	29665	1.60	0	0	0	0	0	0
Others	69472	3.74	0	0	0	0	0	0
G-index			3.226	3.292	3.794	3.226	3.226	3.292
SL-index			0.296	0.295	0.297	0.296	0.296	0.295
LH-index			5.342	5.407	6.012	5.342	5.342	5.407

Table 7: Votes and mandates for the 2018 election in Sweden.

We conclude that based on the least square distance, KH would have given a more proportional mandate allocation in all the elections. We find that the minimal index in step 3 in Algorithm 2 is always unique for these instances.

For 2018 the solution of mSL and the actual allocation are identical, which shows that the procedure with compensatory mandates worked rather well that year (if the goal is a minimal SL-index). For 2014, the actual result is slightly worse, and for 2010, there is a large difference, as KH gives G-index 3.802 and mSL 3.916, while the actual result gives 5.267. There were articles in newspapers claiming that the election result was flawed, and we 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 mSL 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 “adjustment” part of mSL does not make a big difference in practice.

We earlier mentioned electoral cooperation, where two parties are treated as one. In section 10.1 this was used for small parties to get above the threshold. Let us also consider the effect it may have for parties over the threshold. We have used the data from 2018 with three (fictitious) cooperations, and the results are given in table 8 for KH and mSL, where small parties have been removed from the table.

Due to the integrality of the mandates, it is inevitable that electoral cooperation may have some effects, but it is not desired, and the effects should be small and random, and not possible to plan in advance and use tactically. The G-index for KH is 3.225 without cooperation, and 3.236, 3.237 and 3.229 for the three cooperations. For mSL the corresponding numbers are 3.292, 3.408, 3.414 and 3.538. Clearly the disproportionality increases more for mSL than for KH. This is to be expected, since the size of the party matters for the SL-index, so if some small parties are treated as one large party, deviations will be evaluated differently. The changes in KH are small and random, so we conclude that our method is better at handling such changes.

Cooperation1			Cooperation2			Cooperation3		
Party	KH	mSL	Party	KH	mSL	Party	KH	mSL
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		
	KH	mSL	Res	KH	mSL	Res	KH	mSL	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.

10.7 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 mSL gives the left block one more mandate than the right block, which is equal to the actual result. However, KH 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 made when it came to forming a government, which was unusually difficult (for Sweden) and took more than 130 days, and seems to have changed the political blocks.

The government that was formed in 2019 was S + MP, 116 mandates together, with the support of C + L, 51 mandates, which sums up to 167. This a minority government, since majority would need 175 mandates. An assumption, however, was that M + KD + SD, 154 mandates, would not cooperate with V, 28 mandates. This has turned out to be false.

Using our method, the numbers would have been very similar: S + MP, 115 mandates, C + L, 51 mandates, which sums up to 166, M + KD + SD, 155 mandates, and V, 28 mandates. However, L has a new party leader, and might in the future change its support for the government. Then S + MP + C with V would have 175 mandates, which is exactly what is needed for majority. However with our method, S + MP + C

Party	r_j	d^1r_j	d^2r_j	2019	KH	mSL	dHo	Ha	Dr
M	698770	3.36	3.53	4	4	4	4	4	4
C	447641	2.15	2.26	2	2	2	2	2	2
L	171419	0.82	0.86	1	1	1	1	1	1
KD	357856	1.72	1.81	2	2	2	2	2	2
S	974589	4.69	4.93	5	5	5	6	5	5
V	282300	1.36	1.43	1	2	1	1	2	1
MP	478258	2.30	2.42	2	2	3	2	2	3
SD	636877	3.06	3.22	3	3	3	3	3	3
FI	321436	0.15	0.16	0	0	0	0	0	0
PP	26526	0.12	0.13	0	0	0	0	0	0
Others	45091	0.21	0.22	0	0	0	0	0	0
G-index				0.685	0.705	0.711	0.997	0.705	0.711

Table 10: Votes and mandates for the 2019 EU election in Sweden.

with V would have 174, while M + KD + L + SD would have 175.

The point here is not to dwell on Swedish party politics, but to show that even the small differences between mSL and KH might make a important difference for the government. Our conclusion is that it is not sufficient to allocate mandates in a “rather good” way. If it is possible to do it the best way (which it is), one should do that.

10.8 The EU-election 2019

In the election for the parliament of EU, the different countries use their own systems. When the election took place 2019, Sweden had 20 seats. When we ran our code on the election results, all the methods KH, mSL, dHo, Ha and Dr gave the same result, which also was the result that was decided.

However, when the United Kingdom left the EU, Sweden got one more seat. It was decided to give this mandate to MP. There was an article in Svenska Dagbladet 2020-02-16 claiming that this was a result of a “manipulated system”, and that the mandate should have gone to V. The article argues that the Swedish election system should be changed in order for the Swedish population not to loose confidence in the system.

Running the code with $m = 21$ gave the results in table 10. In the table we give the party, the number of votes, the optimal continuous solution, first for 20 seats, d^1r_j , and then for 21 seats, d^2r_j (as changing m changes d). After that we give the number of mandates at the time of the election, 2019, (which all methods agree on), and then the number of mandates for the different methods for $m = 21$.

In the table we see that dr increases for all parties, for V from 1.36 to 1.43 and for MP from 2.30 to 2.42. We also find that mSL indeed gives the additional mandate to MP, while KH gives the additional mandate to V. Using the SL-metric it seems that 2.42 is closer to 3 than 1.43 is to 2, while using an ordinary distance, as KH does, it would be the other way around.

In other words we agree with the newspaper article. Why settle for a solution with G-index 0.711, when there is one with G-index 0.705? The article also shows that one doesn't have to be a mathematician to find flaws in the present system.

10.9 Tests with other measures

We have also considered using the other measures, c^{SL} and c^{LH} , in our algorithm. The first conclusion was that the LH-index has a lack of controllability, i.e. many coefficients are equal (-1 or 1), so the choice of minimizer is often random. Practical tests confirmed that this measure is inferior. The same seems to apply to the max-norm.

Another observation was that using c^{SL} as objective function, the two algorithms gave the same result for the Sweden instances. So it seems that KH with c^{SL} as objective function is just as good as mSL for minimizing the SL-index.

Let us finally mention the possibility of combining the different measures. For the election 2018 in Sweden, KH gives G-index 3.225, SL-index 0.000295 and LH-index 5.342. A possibility would then be to use $w_1 = 1$, $w_2 = 10000$ and $w_3 = 0.1$, which in principle gives 3.22, 2.95, 0.53, i.e. G- and SL-index almost equal in weight, and LH-index less.

We solved the three Sweden instances with this objective function, and got the same solutions as with $w_1 = 1$, $w_2 = 0$ and $w_3 = 0$ (i.e. our original run with KH). This is not surprising, since the three indices correlate rather strongly, i.e. they point in approximately the same direction. In any case we note that it is perfectly possible to use a combination of the measures.

We also recall the discussion in section 7.7, how using $w_1 = 1$ and $w_2 = \varepsilon$ (and $w_3 = 0$) can be a way of dealing with the situation when the minimizer in an iteration is not unique. A practical suggestion could be to use $w_1 = 1$, $w_2 = 1$ and $w_3 = 0$, since the SL-index usually is much smaller than the G-index. This illustrates the flexibility of our approach.

10.10 Constituency allocation

We have also made tests with the method described in section 8. Here we take the result of national allocation of mandates as input, and allocate 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 entitled to vote, or if the number of votes are not proportional to the number of inhabitants entitled to vote.

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

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.

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 mSL to 3.801 for KH, 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 2028. That can be compared with the 310 fixed mandates and 39 compensatory mandates that has been used. We conclude that

the mandate allocations are rather similar, and that our method would not mean a drastic change.

Looking at the differences, 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 previously 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 in this case the number of mandates allocated by the old procedure changes rather slowly in the direction indicated by our method. A similar pattern can be seen for Göteborgs kommun. It is possible that this indicates that the numbers of mandates used in the present procedure are not really updated. 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 compensatory mandates are added. However, in a few cases the fixed numbers of mandates lie above what our method would give.

Let us take a closer look at this question for 2018. In table 12, we give the number of mandates for each constituency our method gave, and in table 13, we give the corresponding continuous solution, dt .

The sums of the continuous solution for the constituencies, $\sum_j dt_{ij}$, are [5.63 10.13 2.22 9.99 18.84 11.69 4.59 12.37 8.67 6.71 10.42 8.98 10.91 13.62 10.32 32.93 43.92 9.89 13.01 9.91 9.64 8.78 9.31 9.41 7.73 13.11 9.54 10.52 16.21]. Looking at for example Gotland, the third row, the sum is 2.22, which indicates that it should get two mandates. However, the number of mandates in the continuous solution for the different parties is [0.37 0.38 0.08 0.09 0.66 0.20 0.11 0.28 0.02 0.02], i.e. all but one are below 0.4. Therefore it is not surprising that our method gives the solution [0 0 0 0 1 0 0 0 0 0] (which actually is the continuous solution rounded). This only gives one mandate to Gotland. We get $\sum_j y_{ij} = 1$, while $\sum_j dt_{ij} = 2.22$.

The reason is that the votes are spread out over the parties, so that each party gets a very small part, in this case well below 1. The integral solution is usually not far away from the rounded continuous solution, and it is unusual that almost all changes from the continuous solution are in the same direction, in this case downwards. However, it may happen that $\sum_j y_{ij}$ is more than one unit away from $\sum_j dt_{ij}$.

This is probably mostly seen as a problem when $\sum_j y_{ij} \leq \lfloor \sum_j dt_{ij} \rfloor - 1$, especially for a constituency that is very small. One could say that integrality is more restrictive for small constituencies. Still, this is the optimal solution to the optimization problem we stated. Our conclusion is that, in view of the integrality, which we cannot escape, it is not good to have very small constituencies.

Constituency	M	C	L	KD	S	V	MP	SD	FI	O
Blekinge län	1	0	0	0	2	0	0	2	0	0
Dalarnas län	2	1	0	1	3	1	0	2	0	0
Gotlands län	0	0	0	0	1	0	0	0	0	0
Gävleborgs län	2	1	0	0	3	1	0	2	0	0
Göteborgs kommun	4	1	1	1	4	3	1	3	0	0
Hallands län	3	1	1	1	3	1	1	2	0	0
Jämtlands län	1	1	0	0	2	0	0	1	0	0
Jönköpings län	2	1	1	1	3	1	1	2	0	0
Kalmar län	2	1	0	1	3	0	0	2	0	0
Kronobergs län	1	1	0	0	2	0	0	1	0	0
Malmö kommun	2	0	1	0	3	1	1	2	0	0
Norrbottnens län	1	1	0	0	4	1	0	1	0	0
Skåne läns norra och östra	2	1	1	1	3	0	0	3	0	0
Skåne läns södra	3	1	1	1	3	1	1	3	0	0
Skåne läns västra	2	1	1	1	3	0	1	3	0	0
Stockholms kommun	7	3	3	2	8	4	3	3	0	0
Stockholms län	12	4	3	3	10	3	2	7	0	0
Södermanlands län	2	1	0	1	3	1	0	2	0	0
Uppsala län	2	1	1	1	4	1	1	2	0	0
Värmlands län	2	1	1	1	3	1	0	2	0	0
Västerbottens län	1	1	0	0	4	1	1	1	0	0
Västernorrlands län	1	1	0	0	3	1	0	1	0	0
Västmanlands län	2	1	1	1	3	1	0	2	0	0
Västra Götalands läns norra	2	1	1	1	3	1	0	2	0	0
Västra Götalands läns södra	1	1	0	1	2	0	0	2	0	0
Västra Götalands läns västra	3	1	1	1	3	1	1	2	0	0
Västra Götalands läns östra	2	1	0	1	3	1	0	2	0	0
Örebro län	2	1	1	1	3	1	1	2	0	0
Östergötlands län	3	1	1	1	5	1	1	3	0	0

Table 12: Mandates for constituencies 2018.

Constituency	M	C	L	KD	S	V	MP	SD	FI	O
Blekinge län	0.97	0.38	0.23	0.32	1.77	0.31	0.16	1.42	0.02	0.05
Dalarnas län	1.69	1.01	0.37	0.62	3.11	0.74	0.34	2.10	0.04	0.11
Gotlands län	0.37	0.38	0.08	0.09	0.66	0.20	0.11	0.28	0.02	0.02
Gävleborgs län	1.53	0.91	0.40	0.53	3.41	0.83	0.31	1.96	0.03	0.09
Göteborgs kommun	3.74	1.34	1.37	1.04	4.48	2.64	1.31	2.53	0.16	0.24
Hallands län	2.67	1.18	0.67	0.83	3.02	0.57	0.41	2.18	0.04	0.12
Jämtlands län	0.66	0.71	0.13	0.22	1.54	0.39	0.16	0.71	0.02	0.05
Jönköpings län	2.18	1.24	0.49	1.48	3.44	0.61	0.40	2.38	0.03	0.12
Kalmar län	1.49	0.85	0.33	0.62	2.70	0.53	0.25	1.78	0.03	0.08
Kronobergs län	1.29	0.63	0.24	0.51	1.98	0.41	0.22	1.36	0.02	0.06
Malmö kommun	2.03	0.59	0.58	0.40	3.03	1.24	0.58	1.75	0.08	0.13
Norrbottnens län	1.15	0.63	0.29	0.42	3.74	0.96	0.25	1.42	0.04	0.09
Skåne läns norra och östra	2.20	0.75	0.48	0.71	2.73	0.45	0.31	3.14	0.03	0.10
Skåne läns södra	3.27	1.08	0.96	0.73	2.97	0.72	0.67	3.01	0.07	0.15
Skåne läns västra	2.17	0.66	0.54	0.55	2.70	0.49	0.37	2.70	0.04	0.10
Stockholms kommun	7.21	2.99	2.59	1.60	7.83	4.32	2.53	3.24	0.23	0.39
Stockholms län	11.42	3.76	3.05	3.07	10.13	3.04	2.10	6.69	0.17	0.49
Södermanlands län	2.02	0.73	0.43	0.54	3.10	0.66	0.36	1.91	0.03	0.10
Uppsala län	2.47	1.20	0.81	0.90	3.51	1.14	0.72	2.01	0.08	0.18
Värmlands län	1.63	0.92	0.44	0.63	3.36	0.69	0.33	1.78	0.04	0.09
Västerbottens län	1.30	0.97	0.38	0.51	3.67	1.22	0.37	1.05	0.06	0.09
Västernorrlands län	1.20	0.87	0.30	0.48	3.50	0.72	0.24	1.37	0.04	0.09
Västmanlands län	1.78	0.62	0.51	0.54	2.93	0.66	0.28	1.87	0.03	0.09
Västra Götalands läns norra	1.66	0.74	0.44	0.61	2.87	0.59	0.34	2.01	0.04	0.09
Västra Götalands läns södra	1.47	0.74	0.41	0.56	2.17	0.50	0.26	1.52	0.02	0.08
Västra Götalands läns västra	2.85	1.07	0.90	1.01	3.18	0.89	0.66	2.36	0.05	0.13
Västra Götalands läns östra	1.75	0.89	0.41	0.75	2.90	0.54	0.29	1.86	0.03	0.10
Örebro län	1.75	0.82	0.48	0.73	3.48	0.80	0.39	1.91	0.04	0.12
Östergötlands län	3.31	1.36	0.87	1.08	4.71	1.10	0.68	2.86	0.07	0.19

Table 13: Continuous solution dt for constituencies 2018.

Constituency	M	C	L	KD	S	V	MP	SD	FI	O
Blekinge län	1	0	0	0	2	0	0	2	0	0
Dalarnas län	2	1	0	1	3	1	0	2	0	0
Gotlands län	1	0	0	0	1	0	0	0	0	0
Gävleborgs län	2	1	0	0	3	1	0	2	0	0
Göteborgs kommun	4	1	1	1	4	3	1	3	0	0
Hallands län	3	1	1	1	3	1	1	2	0	0
Jämtlands län	1	1	0	0	2	0	0	1	0	0
Jönköpings län	2	1	1	1	3	1	1	2	0	0
Kalmar län	2	1	0	1	3	0	0	2	0	0
Kronobergs län	1	1	0	1	2	0	0	1	0	0
Malmö kommun	2	0	1	0	3	1	1	2	0	0
Norrbottnens län	1	1	0	0	4	1	0	1	0	0
Skåne läns norra och östra	2	1	1	1	3	0	0	3	0	0
Skåne läns södra	3	1	1	1	3	1	1	3	0	0
Skåne läns västra	2	1	1	1	3	0	1	3	0	0
Stockholms kommun	7	3	3	2	8	4	3	3	0	0
Stockholms län	11	4	3	3	10	3	2	7	0	0
Södermanlands län	2	1	0	1	3	1	0	2	0	0
Uppsala län	2	1	1	1	3	1	1	2	0	0
Värmlands län	2	1	1	1	3	1	0	2	0	0
Västerbottens län	1	1	0	0	4	1	1	1	0	0
Västernorrlands län	1	1	0	0	4	1	0	1	0	0
Västmanlands län	2	1	1	0	3	1	0	2	0	0
Västra Götalands läns norra	2	1	1	1	3	1	0	2	0	0
Västra Götalands läns södra	1	1	0	1	2	0	0	2	0	0
Västra Götalands läns västra	3	1	1	1	3	1	1	2	0	0
Västra Götalands läns östra	2	1	0	1	3	1	0	2	0	0
Örebro län	2	1	1	1	3	1	1	2	0	0
Östergötlands län	3	1	1	1	5	1	1	3	0	0

Table 14: Mandates for constituencies 2018 with lower bounds.

10.11 Constituency allocation with lower bounds

As discussed in section 8.2, there could possibly be regional aspects which lead to requirements that the number of mandates for certain constituencies may not be less than certain prescribed numbers. It is possible, as we did in section 8.2, to formulate an optimization problem to handle this case also. We have made a computational test with the data for Sweden 2018, and used the fixed mandates used in practice as lower bounds.

The LP-problem has 101 500 variables, 330 constraints and 405 130 non-zero elements in the constraint matrix. It was solved by GLPK in 0.5 seconds, after 1003 simplex iterations. In table 14, we give the results.

In the previous run, without lower bounds, 3 constituencies, Gotland, Kronobergs län and Västernorrlands län, got one less than their fixed numbers of mandates. Here, they

get one more mandate, and end up at the lower bound, Gotland at 2, Kronobergs län at 6 and Västernorrlands län at 8. The three mandates were taken from Stockholms län, ending up at 43, Uppsala län, ending up at 12, and Västmanlands län, ending up at 10. In the optimal solution with lower bounds, only these three mandates were moved, compared to our solution without lower bounds.

Comparing the optimal solution with lower bounds to the actual result used in practice reveals larger differences. In total 20 mandates were moved, of which 10 gave net differences between constituencies, and the other half of the moved mandates were between different parties in the same constituencies.

Our conclusion here is clear. If one insists on using lower bounds for the constituencies, one should use the optimal solution of P7. The solution will still be better than the procedure used today.

Another question crops up, in view of this result. Would the authorities require that an algorithm is specified in the legal text, as it is now, or would it be sufficient to state that one should use the optimal solution to a well specified optimization problem? In the first case, it is in practice only possible to use fairly simple algorithms, probably of greedy type (since people might react negatively to mandates allocated in a certain stage in the algorithm being removed in a later stage).

We were lucky that our (simple) method actually finds the optimal solution (without lower bounds). With lower bounds, there is no such simple greedy method that can find the optimal solution. It seems unreasonable that the legal text should contain an algorithm for solving a general LP-problem.

Under such limitations, it seems that the optimal solution with lower bounds for constituencies is out of reach. Any algorithm that is simple enough to be specified in the legal paragraphs would be a heuristic, hopefully giving rather good results, but not guaranteeing optimality. (This is actually what we have today.)

On the other hand, people are today putting their faith in computers and mobile phones, while having no insight in how things are calculated. Therefore there might be a hope for legal text that does not specify an algorithm, but only that the optimal solution to a certain problem should be used.

But let us just recall that this is an issue only if one insists on using lower bounds for constituencies that make the proportionality worse. In our test, there was only a few lower bounds that were that high. Lower bounds that lie below what would be nationwide optimal is not a problem.

11 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 ones presently used, measure by least square distance. The method 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, and it includes a mathematically appealing way of handling ties. The method can optimally handle a parliament threshold, and allows different objective functions. We believe that the Gallagher index is the most relevant measure, but the method could also be used with the Sainte-Laguë index, or a combination of them. Previously different measures required different methods, and for many variations of the methods, it is unclear which optimization problem the method actually solves, if any. With our method, it is possible to choose measure more freely.

We also suggest a new way of allocating mandates to constituencies, which eliminates the need for compensatory 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 also present an optimization model for allocating mandates to constituencies when there is a lower bound on the number of mandates for each constituency. That problem is a general LP-problem, and not solvable by a greedy algorithm. However, it is easily solvable by an LP-code.

In principle, 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 believe that our method does the best job in this aspect. The code can be obtained by contacting the author at kaj.holmberg@liu.se.

Acknowledgment: We thank Svante Janson for useful comments.

References

- Aleskerov, F., and Platonov, V. (2004), “Disproportionality indices for pr systems”, in: *Reasoned Choices*, The Finnish Political Science Association, Turku.
- Anthonisse, J. (1984), “Proportional representation in a regional council”, Technical Paper 5, Centre for Mathematics and Computer Science, Amsterdam, NL.
- Benoit, K. (2000), “Which electoral formula is the most proportional? A new look with new evidence”, *Political Analysis* 8/4, 381–388.
- Betzler, N., Slinko, A., and Uhlmann, J. (2013), “On the computation of fully proportional representation”, *Journal of Artificial Intelligence Research* 47, 475–519.
- Brandt, F., Conitzer, V., Endriss, U., Lang, J., and Procaccia, A. D. (eds.). (2016), *Handbook of Computational Social Choice*, Cambridge University Press.
- Brill, M., Laslier, J.-F., and Skowron, P. (2016). “Multiwinner approval rules as apportionment methods”.

- Faliszewski, P., Skowron, P., Slinko, A., and Talmon, N. (2018a). “Committee scoring rules: Axiomatic characterization and hierarchy”,.
- Faliszewski, P., Slinko, A., Stahl, K., and Talmon, N. (2018b), “Achieving fully proportional representation by clustering voters”, *Journal of Heuristics* 24, 725–756.
- Gaffke, N., and Pukelsheim, F. (2008), “Divisor methods for proportional representation systems: An optimization approach to vector and matrix apportionment problems”, *Mathematical Social Sciences* 56, 166–184.
- Gallagher, M. (1991), “Proportionality, disproportionality and electoral systems”, *Electoral Studies* 10/1, 33 – 51.
- Gallagher, M., and Mitchell, P. (eds.). (2005), *The Politics of Electoral Systems*, Oxford University Press.
- Holmberg, K. (2019), “A new method for optimal proportional representation”, Research report LiTH-MAT-R-2019/01-SE, Department of Mathematics, Linköping University, Sweden.
- Hylland, A. (2010), “Elections in Norway: Notes on the electoral system”,.
- Janson, S. (2014), *Proportionella valmetoder*. Department of Mathematics, Uppsala University.
- Janson, S., and Linusson, S. (2014), “Föreslagna ändringar i Sveriges valsystem”,.
- Karpov, A. (2008), “Measurement of disproportionality in proportional representation systems”, *Mathematical and Computer Modelling* 48, 1421–1438.
- Laakso, M., and Taagepera, R. (1981), “Proportional representation and effective number of parties in Finland”, in: Holler, M. (ed.), *Power, Voting, and Voting Power*. Physica, Heidelberg.
- Linusson, S. (2008), *Matematik och människor*, chap. Uddatalsmetoden och valsystem, Nationellt Centrum för Matematikutbildning.
- Linusson, S., and Ryd, G. (2012), “Dynamic adjustment: An electoral method for relaxed double proportionality”, KTH.
- Loosemore, J., and Hanby, V. J. (1971), “The theoretical limits of maximum distortion: some analytic expressions for electoral systems”, *British Journal of Political Science* 1/4, 467–477.
- Maunula, M. (2008), *Proportionality and party success in Europe*. Ph.D. thesis, Department of Political Science, Faculty of Social Sciences, University of Turku, Finland.
- Potthoff, R. F., and Brams, S. J. (1998), “Proportional representation: Broadening the options”, *Journal of Theoretical Politics* 10/2, 147–178.
- Pukelsheim, F. (2017), *Proportional Representation, Apportionment Methods and Their Applications* (2 edition), Springer.

Ricca, F., Scozzari, A., and Serafini, P. (2017), “A guided tour of the mathematics of seat allocation and political districting”, in: Endriss, U. (ed.), *Trends in computational social choice*, AI Access.