

# On the Computation of Fully Proportional Representation

Nadja Betzler\*

Institut für Softwaretechnik und  
Theoretische Informatik  
TU Berlin  
Franklinstr. 28/29  
10587 Berlin  
nadja.betzler@campus.tu-berlin.de

Arkadii Slinko

Department of Mathematics  
University of Auckland  
slinko@math.auckland.ac.nz

Johannes Uhlmann†

Institut für Softwaretechnik und  
Theoretische Informatik  
TU Berlin  
Franklinstr. 28/29  
10587 Berlin  
johannes.uhlmann@campus.tu-berlin.de

September 27, 2011

## Abstract

We investigate two systems of fully proportional representation suggested by Chamberlin & Courant (1983) and Monroe (1995). Both systems assign a representative to each voter so that the “sum of misrepresentations” is minimized. The winner determination problem for both systems is known to be NP-hard, hence this work aims at investigating whether there are variants of the proposed rules and/or specific electorates for which these problems can be solved efficiently. As a variation of these rules, instead of minimizing the sum of misrepresentations, we considered minimizing the maximal misrepresentation introducing effectively two new rules. In the general case these “minimax” versions of classical rules appeared to be still NP-hard.

We investigated the parameterized complexity of winner determination of the two classical and two new rules with respect to several parameters. Here we have a mixture of positive and negative results: e.g. we proved fixed-parameter tractability for the parameter the number of candidates but fixed-parameter intractability for the number of winners.

For single-peaked electorates our results are overwhelmingly positive: we provide polynomial-time algorithms for most of the considered problems. The only rule that remains NP-hard for single-peaked electorates is the classical Monroe rule.

**Key words:** multi-winner elections, single-peaked electorate, parameterized complexity, NP-hardness, dynamic programming.

---

\*Main work was done while the author stayed for a research visit at the University of Auckland. The visit was funded by a fellowship by the Deutscher Akademischer Austausch Dienst (DAAD).

†Partially supported by the DFG, project PAWS NI-931/10.

# 1 Introduction

There is a significant difference in the purpose of single-winner and multi-winner elections. Single-winner social choice rules are used to make final decisions, e.g., to elect a president or to choose a certain course of action. The multi-winner election rules are used to elect an assembly whose members will be authorized to take final decisions on behalf of the society. As a result the main property that multi-winner rules have to satisfy is that the elected assembly represents the society adequately. This, in particular, means that when a final decision is taken all opinions existing in the society are heard and taken into consideration. As Black powerfully expressed it:

A scheme of proportional representation attempts to secure an assembly whose membership will, so far as possible, be proportionate to the volume of the different shades of political opinion held throughout the country; the microcosm is to be a true reflexion of the macrocosm. [6, p. 75].

And although any single-winner social choice rule can be easily extended to select an assembly—e.g., by taking candidates with best scores or applying the rule repeatedly until the required quantity of representatives is elected—this is a wrong approach to the problem [9] (see also [40] for some experimental evidence). The reason is that the majoritarian logic which dominates the design of single-winner social choice rules cannot provide for a balanced assembly membership.

The standard solution to the problem of electing an assembly has been the division of the election into single-member districts with approximately equal population. Each district elects one member of the assembly using a single-winner rule, normally the plurality. And although one might question whether districting should be instead based on the total adult population or on the number of registered voters, the current practice is well established and entrenched by law in many countries, including the United States [8]. However the main problem with this approach is not the districting but the fact that it also fails to give a representation to minorities; the minority can comprise 49% of the population and be not represented in the assembly.

What can be an alternative? An important idea was suggested by Charles Dodgson (Lewis Carroll) [16] and considered in a different form by Black [6]. Then the idea was further developed by Chamberlin & Courant [12] and later by Monroe [43] (relative advantages of both methods from political science point of view have been extensively discussed by Brams [8]). Dodgson asserted that “a representation system should find the coalitions in the election that would have formed if the voters had the necessary time and information” and allow each of the coalitions to elect their representative. If this is adopted, then the minority can form a coalition and be represented.

The realization of this idea required a new concept which is the *misrepresentation*. It is assumed that voters form individual preferences over the candidates based on their political ideology and “their judgement about the abilities of candidates to participate in deliberations and decision making consistent with how the individuals would wish to act were they present” [12]. This was in a way a revolution.

Indeed, in the “single-winner” literature on voting rules it is widely accepted that voter’s political preferences are more complex than their first choices alone. However in “multi-winner” voting literature fixation on first preferences led researchers to think about proportional representation exclusively in terms of first preferences. In list systems of proportional representation, parties are assigned a number of seats in parliament that is proportional to the number of votes (first preferences) they received. The systems like Single Non-Transferable Vote, Block Voting and Cumulative Voting all also do not take second preferences in account [39]. Only the Single Transferable Vote

(STV) is a system of proportional representation (in fact a family of voting methods according to [52]) that allows voters to express the order of preference of candidates [39]. Voters rank the candidates in order of preference; first preference votes are the first to be looked at, and the votes are then transferred, if necessary, from candidates who have either been comfortably elected or who have done so badly that they have been eliminated from the election.<sup>1</sup>

So, if a voter is represented by a candidate who is her first preference it is reasonable to say that he is represented optimally or that the misrepresentation in this case is zero. In general, if a voter is represented by a candidate who is her  $i$ th preference we may assume that she is misrepresented to the degree  $s_i$ . Of course, it is reasonable to assume that  $0 = s_1 \leq s_2 \leq \dots \leq s_m$ . So in this case the rule for measuring the total misrepresentation is fully defined by the vector  $\mathbf{s} = (s_1, \dots, s_m)$ , where  $m$  is the number of candidates. Using the analogy with positional scoring rules for single-winner elections we may say that this misrepresentation function is *positional*. In general, the problem of choosing a proper misrepresentation function is far from being trivial. Levin and Nalebuff [39] illustrate the difficulty vividly: “if the electorate is uniformly distributed on the segment between 0 and 1, and we are to choose three representatives, should they be equally spaced [0.25, 0.5, 0.75], or should they be selected so as to minimize the average distance traveled to the nearest legislator [0.16, 0.5, 0.83]?” In the broadest possible framework the misrepresentation function may be even voter-dependent.

Staying with classical positional misrepresentation functions for the time being suppose now that every voter is assigned to a representative in some way. Measuring the total misrepresentation for the whole society we may adopt either the Harsanyi approach [37] or the Rawlsian one (assuming that the utility of being represented by your  $i$ th best candidate is  $-s_i$ ). By Harsanyi we will have to measure the total misrepresentation as

$$M_H = \sum_{i=1}^m n_i s_i,$$

where  $n_i$  is the number of voters represented by their  $i$ th most preferred candidate. According to Rawls [50] “welfare is maximized when the utility of those society members that have the least is the greatest.” This leads us to the total misrepresentation function

$$M_R = \max_{i=1}^m s_i.$$

Both Chamberlin & Courant [12], and Monroe [43] consider that the best set of representatives must minimize the total misrepresentation which they both calculate using the Borda vector of scores, that is,  $(0, 1, 2, \dots, m - 1)$  and Harsanyi misrepresentation function. Their methods are however different and the difference is very important. Chamberlin & Courant [12] did not impose any restriction on the function that assigns candidates to voters. This may potentially lead to different number of voters whom each candidate represents. To remedy this Chamberlin & Courant suggested to use weighted voting in the assembly where each elected candidate has the weight equal to the number of voters she represents. Monroe rejected this approach and insisted on the principle ‘one member of assembly one vote’. For this reason he insisted that the difference between the number of voters assigned to any two representatives is at most one.

---

<sup>1</sup>In Northern Ireland this is the voting system used for elections to local councils, the Assembly, and the European Parliament. It is used for all elections in the Irish Republic, Malta, and Australia (although single-member constituencies are prevalent in Australia, apart from state level elections in Tasmania and the ACT). Several other countries have recently debated adopting it.

The computational problems corresponding to the Harsanyi approach are known to be NP-hard [40, 49] for classical misrepresentation functions such as Borda and approval. In this paper we ask whether or not the problem of finding an optimal fully proportional representation becomes easier for this classical misrepresentation functions if we adopt the Rawlsian approach for measurement of total misrepresentation. The second goal is to find the parameterized complexity of the considered problems for some natural choices of parameters. The third is to develop efficient algorithms for achieving an optimal fully proportional representation in single-peaked elections. In the remainder of this section, we formally introduce the considered computational problems, summarize previous results, and describe our approaches and results.

## 1.1 Computational problems considered

An *election* is a pair  $E = (C, V)$  where  $C$  is the set of *candidates* (or *alternatives*) and  $V$  is an ordered list of *voters*. Each voter is represented by her *vote*, i.e., a strict, linear order over the set of candidates (also called this voter's *preference order*). We will refer to the list  $V$  as a *preference profile*, and we denote the number of voters in  $V$  by  $n$ . The number of alternatives will be denoted by  $m$ . If the order of voters is not important (the election is anonymous), then  $V$  can be considered as a multiset<sup>2</sup> of votes. In this paper we will consider only anonymous elections.

By  $\text{pos}_v(c)$  we will denote the position of the alternative  $c$  in the ranking of voter  $v$ ; the top-ranked alternative has position 1, the second best has position 2, etc.

**Definition 1.** Given a profile  $V$  over  $C$ , a mapping  $r: V \times C \rightarrow \mathbb{Q}_0^+$  will be called a *misrepresentation function* if for any  $v \in V$  and any two candidates  $c, c' \in C$  the condition  $\text{pos}_v(c) < \text{pos}_v(c')$  implies  $r(v, c) \leq r(v, c')$ .

That is to say that if  $c$  is preferred to  $c'$  in  $v$ 's ranking, then the misrepresentation of  $v$ , when she is represented by  $c'$  will be at least as large as her misrepresentation, when she is represented by  $c$ . In the classical framework the positional misrepresentation function given by  $s = (s_1, \dots, s_m)$  the misrepresentation function is given by the formula

$$r(v, c) = s_{\text{pos}_v(c)}.$$

In the general framework the misrepresentation function may be arbitrary. By  $w: V \rightarrow C$  we denote the function that assigns voters to representatives (or the other way around), i.e., under this assignment voter  $v$  is represented by candidate  $w(v)$ . The total misrepresentation of the election under  $w$  is then given by

$$\sum_{v \in V} r(v, w(v)) \quad \text{or} \quad \max_{v \in V} r(v, w(v))$$

in Harsanyi's classical and Rawls minimax versions, respectively. We say that a mapping  $w$  respects the *M-criterion* if  $|w(V)| = k$  and  $w$  assigns at least  $\lfloor n/k \rfloor$  and at most  $\lceil n/k \rceil$  voters to every candidate from  $w(V)$ , where  $k$  denotes the number of representatives.

Based on the previous discussion, in this work we investigate the computational complexity of the following four combinatorial problems. The two classical ones described above are named after Chamberlin and Courant (CC), for the case when a winner candidate can represent an arbitrary number of voters (and this number of voters will be his weight in the elected assembly), and Monroe (M), for the case when every candidate represents roughly the same number of voters (and

---

<sup>2</sup>This is not a set since two different voters may have the same preference order.

each representative has one vote in the assembly). The two previously unstudied versions which adopt the Rawlsian approach for measuring the total misrepresentation are called the *minimax* versions of the classical ones.

#### CC-MULTIWINNER (CC-MW)

**Given:** A set  $C$  of candidates, a multiset  $V$  of voters, a misrepresentation function  $r$ , a misrepresentation bound  $R \in \mathbb{Q}_0^+$  and a positive integer  $k$ .

**Task:** Find a subset  $C' \subseteq C$  of size  $k$  and an assignment of voters  $w$  such that  $w(V) = C'$  and  $\sum_{v \in V} r(v, w(v)) \leq R$ .

#### MINIMAX CC-MULTIWINNER (MINIMAX CC-MW)

**Given:** A set  $C$  of candidates, a multiset  $V$  of voters, a misrepresentation function  $r$ , a misrepresentation bound  $R \in \mathbb{Q}_0^+$  and a positive integer  $k$ .

**Task:** Find a subset  $C' \subseteq C$  of size  $k$  and an assignment of voters  $w$  such that  $w(V) = C'$  and for every  $v \in V$  one has  $r(v, w(v)) \leq R$ .

#### M-MULTIWINNER (M-MW)

**Given:** A set  $C$  of candidates, a multiset  $V$  of voters, a misrepresentation function  $r$ , a misrepresentation bound  $R \in \mathbb{Q}_0^+$  and a positive integer  $k$ .

**Task:** Find a subset  $C' \subseteq C$  of size  $k$  and an assignment of voters  $w$ , which respects the *M*-criterion,  $w(V) = C'$  and such that  $\sum_{v \in V} r(v, w(v)) \leq R$ .

#### MINIMAX M-MULTIWINNER (MINIMAX M-MW)

**Given:** A set  $C$  of candidates, a multiset  $V$  of voters, a misrepresentation function  $r$ , a misrepresentation bound  $R \in \mathbb{Q}_0^+$  and a positive integer  $k$ .

**Task:** Find a subset  $C' \subseteq C$  of size  $k$  and an assignment of voters  $w$ , which respects the *M*-criterion,  $w(V) = C'$  and such that  $r(v, w(v)) \leq R$ .

We assume that  $k < m$  and  $k < n$  in what follows.

The four problems above are stated for general misrepresentation functions (since some of our algorithmic results hold even for this case) but the main focus of this work is on the following two important misrepresentation functions. We denote the positional misrepresentation function defined by the vector  $(0, 1, \dots, m-1)$  as *Borda misrepresentation function*. In Approval Voting framework, if a voter is represented by her approved alternative, then her misrepresentation is considered to be zero, alternatively it is equal to one. This function is called the *approval misrepresentation function*. Note that some misrepresentation functions, like Borda, can be derived from the preference lists of the voters. In contrast, an approval misrepresentation function cannot be obtained from a preference list without further information about the threshold that separates approved candidates from disapproved ones. Another meaningful positional misrepresentation function would be the  $k$ -Approval one, defined by the vector  $s = (1, \dots, 1, 0, \dots, 0)$  ( $k$  ones), but we do not consider it here.

## 1.2 Previous computational complexity results

The study of the computational complexity of problems in the context of voting was initiated by Bartholdi et al. [2] about 20 years ago but has became an active area of research only recently [14,

27, 24, 26]. While there is a large number of papers dealing with single-winner elections or multiwinner elections whose final goal is still to choose a single winner after a tiebreaking, only few articles [47, 49, 40] deal with the computational complexity of multiwinner elections aimed at achieving a proportional representation. In particular, these works contain NP-hardness proofs for CC-MULTIWINNER and M-MULTIWINNER for approval misrepresentation function [49] and for CC-MULTIWINNER for Borda misrepresentation function [40]. Algorithmic approaches comprise Integer Linear Programming [47, 8] for CC-MW and M-MW, approximation algorithms based on greedy strategies [40] for CC-MW, and polynomial-time algorithms for CC-MW and M-MW for instances where the number of candidates is constant [49]. In contrast, to the best of our knowledge, the computational complexity of the minimax versions of the problems remained unstudied.

We are only aware of one further work explicitly studying computational complexity issues in the context of multiwinner elections. Meir et al. [41] investigate the computational complexity of strategic voting for several multiwinner elections for which a winner can be determined in polynomial time. The systems considered in [41] do not lead to any kind of proportional representation.

### 1.3 Our approach and results for general elections

The first result of this work is that the minimax versions of the classical Chamberlin-Courant and Monroe problems are also NP-complete. In other words, adopting Rawlsian approach does not make computation of the problems easier in general (but we will see that the situation changes completely for single-peaked elections where the minimax version becomes indeed easier). Based on these negative results, this work aims at extending the previous algorithmic approaches described above by an analysis whether or not there are settings in which the problems becomes tractable. To this end, parameterized algorithmics is an appropriate tool as it aims at identifying tractable special cases of NP-hard problems. The cornerstone of this approach is the idea that the complexity of a problem is not only measured in the total size of an input instance  $I$  but additionally in a parameter  $p$ , usually a nonnegative integer (but it can be a pair of integers or virtually anything). A problem is called *fixed-parameter tractable* if there is an algorithm solving any instance of it in  $f(p) \cdot \text{poly}(|I|)$  time, where  $f$  denotes a computable function [19, 30, 46]. For small values of  $p$  an algorithm with such running time might represent an efficient algorithm for the NP-hard problem under consideration. Parameterized complexity also provides a tool of “parameterized reductions” by which one can show that a problem is presumably not fixed-parameter tractable. One of the most important parameterized complexity classes for this purpose is  $W[2]$  (see Section 2 for more details). We remark in passing that a parameterized complexity analysis has been employed for several other voting problems, e.g., [11, 4, 5, 13, 18, 21, 25] (see [3, Chapter 2.3] for an overview).

In the context of multiwinner elections, a parameter that immediately attracts attention is the number  $k$  of winners, which in many settings might be much smaller than the number of candidates or the number of voters. Another reasonable parameter is the misrepresentation bound  $R$  since in an ideal (or fully personalizable as in [40]) situation  $R$  is equal to zero, that is, every voter is represented by one of her most preferred candidates. We provide a parameterized complexity analysis of all four considered problems for the Borda and approval misrepresentation functions with respect to the parameters  $k$  and  $R$ . In addition, we also investigate the combined parameter  $(k, R)$  consisting of the number of winners and the misrepresentation bound.

An overview of the results is provided in Table 1. While, when the number of winners  $k$  is a parameter, all considered problems turn out to be  $W[2]$ -hard, for the parameterization by the total misrepresentation bound  $R$  the results are more varied. For the case  $R = 0$ , for the approval

Table 1: Parameterized complexity of the considered multiwinner problems for instances where the misrepresentation function  $r$  is either approval (A), Borda (B) or unrestricted (U). The entry ‘‘FPT’’ for  $R = 1$ ’’ in the row for the parameter  $(R, k)$  stands for fixed-parameter tractability with respect to  $k$  when  $R = 1$ .

Parameter	$r$	CC-MW	MINIMAX CC-MW	M-MW	MINIMAX M-MW
# winner $k$	A	W[2]-hard	W[2]-hard	W[2]-hard	W[2]-hard
# winner $k$	B	W[2]-hard	W[2]-hard	W[2]-hard	W[2]-hard
misr. $R$	A	NP-h for $R = 0$	NP-h for $R = 0$	NP-h for $R = 0$	NP-h for $R = 0$
misr. $R$	B	XP	NP-h for $R \geq 1$ P for $R = 0$	XP	NP-h for $R \geq 1$ P for $R = 0$
$(R, k)$	A	W[2]-hard	W[2]-hard	W[2]-hard	W[2]-hard
$(R, k)$	B	FPT	FPT	FPT	FPT for $R = 1$
# cand.	U	FPT	FPT	FPT	FPT
# voters	U	FPT	FPT	FPT	FPT

misrepresentation function all four problems are NP-hard while they are solvable in polynomial time for the Borda misrepresentation function. However, MINIMAX CC-MW and MINIMAX M-MW become NP-hard for every  $R \geq 1$ . In contrast, the sum-minimization variants CC-MW and M-MW for the Borda misrepresentation function are solvable in polynomial time for constant  $R$  (the corresponding parameterized complexity class is called XP). Note that the provided algorithm shows the containment in XP with respect to  $R$  but not fixed-parameter tractability, this problem remains open. This inspired our analysis of the combined parameter  $(R, k)$ , covering scenarios in which there is a small set of winners that can represent all voters with a small total misrepresentation. While for the approval misrepresentation function, this still leads to parameterized intractability, for the Borda misrepresentation function, we show fixed-parameter tractability for all problems except MINIMAX M-MW. For this problem, we obtain fixed-parameter tractability with respect to  $k$  when  $R = 1$  and discuss why the analogous approach does not lead to fixed-parameter tractability with respect to  $k$  for  $R = 2$ .

## 1.4 Results for single-peaked elections

Single-peakedness is one of the central notions in social choice and political science alike [6, 44, 51]. The preferences of voters are single-peaked when a single issue dominates their formation. This could be their ideological position on the Left-Right or Liberal-Conservative spectra, level of taxation, immigration quota, etc. Tideman [51] compares single-peakedness with convexity of preferences and discusses when it is reasonable to assume this. He has access to a data collection containing 87 ranked-ballot real-life elections and claims that most of them are single-peaked [51].

This dominating single issue are normally represented by an axis and each voter is characterized by a single point on this axis (see an example on Figure 1, page 21. The misrepresentation function for a fixed voter is then a function of single variable defined on that axis. The single-peakedness of preferences implies that this function has exactly one local minimum. We refer to Section 5 for a formal proof of this statement.

We note that for votes in form of approval ballots as well as linear orders, single-peakedness of

Table 2: Overview of the computational complexity for singled-peaked elections. In case of polynomial-time solvability, the table provides the running times depending on the number  $n$  of voters, the number  $m$  of candidates, and the number  $k$  of winners. If not stated otherwise, the result holds for an arbitrary misrepresentation function.

CC-MW	MINIMAX CC-MW	M-MW	MINIMAX M-MW
$O(nm^3)$	$O(nm)$	$O(n^5mk^3)$ for approval NP-hard for integer mis. func.	$O(n^2m^2(n+m))$

the profile can be checked in linear time [7, 22] with the reconstruction of the order of the candidates on the axis.

The study of the computational complexity of voting rules with NP-hard winner-determination problem shows that for all Condorcet-consistent ones—and these include Dodgson, Kemeny, and Young rules—the winner-determination problem becomes polynomial-time solvable if we restrict ourselves with single-peaked profiles [10]. The obvious reason for this is that single-peakedness eliminates the possibility of Condorcet cycles in the election.

It is not that obvious that single-peakedness must also simplify the winner-determination problem for methods of proportional representation. However, it seems natural to investigate this possibility. Our results show that, indeed, in many instances the winner-determination problem for methods of proportional representation does become easier too.

Our results are summarized in Table 2. For CC-MW and MINIMAX CC-MW the problems are solvable in polynomial time for an arbitrary (rational-valued) misrepresentation function. More specifically, for CC-MW we provide a dynamic programming algorithm running in  $O(nm^3)$  time for  $n$  voters and  $m$  candidates, and MINIMAX CC-MW can be solved in  $O(nm)$  time by a greedy algorithm. For the Monroe system and its variants, the results become more diverse. While MINIMAX M-MW for the general misrepresentation function is still solvable in polynomial time, M-MW is NP-hard. However, on the positive side, we can still show polynomial-time solvability for M-MW for the approval misrepresentation function. Basically, the results are obtained as follows. For MINIMAX M-MW we establish a close connection to a “one-dimensional rectangle stabbing” problem with capacities which allows to directly employ a known polynomial-time algorithm [23]. Moreover, for M-MW for the approval misrepresentation function, we show how to extend this dynamic programming algorithm. The NP-hardness of M-MW is established by a many-one reduction from a restricted version of the EXACT 3-COVER problem. The NP-hardness holds for an integer-valued misrepresentation function for which the maximum misrepresentation value is still polynomial in the number of candidates. However, we allow that a voter may assign the same misrepresentation value to several candidates. Hence, it is not obvious how to transfer the corresponding many-one reduction to M-MW for the Borda misrepresentation function. For this problem the computational complexity thus is left open.

## 1.5 Organization of the paper

The paper is organized as follows. In Section 2 we introduce the main concepts of parameterized complexity and some graph algorithms. Section 3 contains basic observations about the relations

of the four problems under consideration and some fixed-parameter tractability results with respect to the number of voters and the number of candidates. The two main contributions are proved in Section 4, where most important parameterized complexity results as well as the NP-hardness results for the minimax versions are proved, and in Section 5, where the special case of single-peaked elections is handled. Finally, in Section 6 we conclude with a discussion of the relevance of our results and some related problems and settings.

## 2 Preliminaries

We briefly introduce the framework of parameterized complexity followed by some basic graph algorithms that are used in this paper. For basic notions regarding classical complexity theory we refer to [32].

### 2.1 Parameterized Complexity

The concept of parameterized complexity was pioneered by Downey and Fellows [19] (see also [30, 46] for text books). The fundamental goal is to find out whether the seemingly unavoidable combinatorial explosion occurring in algorithms to decide NP-hard problems can be confined to certain problem-specific parameters. The idea is that when such a parameter assumes only small values in applications, then an algorithm with a running time that is exponential exclusively with respect to the parameter may be efficient and practical. We now provide the formal definitions.

**Definition 2.** A parameterized problem is a language  $L \subseteq \Sigma^* \times \Sigma^*$ , where  $\Sigma$  is a finite alphabet. The second component is called the parameter of the problem.

We consider parameters which are positive integers or “combined” parameters which are tuples of positive integers.

**Definition 3.** A parameterized problem  $L$  is fixed-parameter tractable if there is an algorithm that decides in  $f(p) \cdot |x|^{O(1)}$  time whether  $(x, p) \in L$ , where  $f$  is an arbitrary computable function depending only on  $p$ . The complexity class of all fixed-parameter tractable problems is called FPT.

We stress that the concept of fixed-parameter tractability is different from the notion of “polynomial-time solvability for constant  $p$ ” since an algorithm running in  $O(|x|^p)$  time does not show fixed-parameter tractability. All problems that can be solved in the running time  $O(|x|^{f(p)})$  for a computable function  $f$  form the complexity class XP. Unfortunately, not all parameterized problems are fixed-parameter tractable. To this end, Downey and Fellows [19] developed a theory of parameterized intractability by means of a completeness program with complexity classes. More specifically, the so-called W-hierarchy is defined by using Boolean circuits and consists of the following classes and interrelations:

$$\text{FPT} \subseteq \text{W}[1] \subseteq \text{W}[2] \subseteq \dots \subseteq \text{W}[\text{Sat}] \subseteq \text{W}[\text{P}] \subseteq \text{XP}.$$

In this work, we only provide results regarding the second level of (presumable) parameterized intractability captured by the complexity class W[2]. The containment  $\text{W}[1] \subseteq \text{FPT}$  would not imply  $\text{P} = \text{NP}$  but the failure of the Exponential Time Hypothesis [38]. Hence, it is commonly believed that W[1]-hard problems are not fixed-parameter tractable. To show W[t]-hardness for any positive integer  $t$ , the following reduction concept was introduced.

**Definition 4.** Let  $L, L' \subseteq \Sigma^* \times \Sigma^*$  be two parameterized problems. We say that  $L$  reduces to  $L'$  by a parameterized reduction if there are two computable functions  $h_1: \Sigma^* \rightarrow \Sigma^*$  and  $h_2: \Sigma^* \rightarrow \Sigma^*$  and a function  $f: \Sigma^* \times \Sigma^* \rightarrow \Sigma^* \times \Sigma^*$  such that for each  $(x, p) \in \Sigma^* \times \Sigma^*$

1.  $(x, p) \in L \iff f(x, p) \in L'$  and  $f$  is computable in time  $|x|^{O(1)} \cdot h_2(|p|)$  and
2.  $p' = h_1(p)$  for  $(x', p') := f(x, p)$ .

Analogously to the case of NP-hardness, for any positive integer  $t$ , it suffices to give a parameterized reduction from one  $\text{W}[t]$ -hard parameterized problem  $X$  to a parameterized problem  $Y$  to show the  $\text{W}[t]$ -hardness of  $Y$ . For more details about parameterized complexity theory we refer to the textbooks [19, 30, 46].

Several parameterized reductions in this work are from the  $\text{W}[2]$ -complete HITTING SET (HS) problem: Given a set family  $\mathcal{F} = \{F_1, \dots, F_n\}$  over a universe  $U = \{u_1, \dots, u_m\}$  and an integer  $k \geq 0$ , decide whether there is a *hitting set*  $U' \subseteq U$  of size at most  $k$  by which we understand a set  $U'$  such that  $F_i \cap U' \neq \emptyset$  for every  $1 \leq i \leq m$ . HS is NP-hard [32] and  $\text{W}[2]$ -hard with respect to parameter  $k$  [19].

## 2.2 Graph algorithms

Some of our algorithmic results employ basic graph algorithms. An *undirected graph* is a pair  $G = (U, E)$ , consisting of the set  $U$  of vertices and the set  $E$  of edges, where an edge is an unordered pair (size-two set) of vertices. Two vertices  $u, v \in U$  are called *adjacent* if  $\{u, v\} \in E$ . For an undirected graph  $G = (U, E)$  and a vertex  $u \in U$ , the *neighborhood*  $N(u)$  of  $u$  is the set of vertices adjacent to  $u$ .

An undirected graph  $G = (U, E)$  is called *bipartite* if the vertex set  $U$  can be partitioned into two subsets  $U_1$  and  $U_2$  ( $U = U_1 \cup U_2$ ,  $U_1 \cap U_2 = \emptyset$ ) such that for every edge  $\{u, v\} \in E$  it holds that  $U_1 \cap \{u, v\} \neq \emptyset$  and  $U_2 \cap \{u, v\} \neq \emptyset$ . A *matching* is an edge set  $E' \subseteq E$  such that  $e \cap e' = \emptyset$  for every two  $e, e' \in E'$ ,  $e \neq e'$ . A *maximum matching* is a matching with maximum cardinality. A *maximum-weight matching* in an undirected graph where each edge  $\{u, v\}$  is associated with a weight  $w(\{u, v\})$  is a matching  $E'$  such that  $\sum_{\{u, v\} \in E'} w(\{u, v\})$  is maximal.

A directed graph or directed network is a pair  $G = (U, A)$ , consisting of the set  $U$  of vertices and the set  $A \subseteq U \times U$  of directed edges (or arcs). A *flow network* is a directed network  $G = (U, A)$  with two distinguished vertices  $s \in U$  (the source) and  $t \in U$  (the sink or target) where each arc  $(u, v) \in A$  is associated with a nonnegative capacity  $c(u, v)$ . Roughly speaking, a flow is a function  $f$  that assigns a real value  $f(u, v)$  with  $0 \leq f(u, v) \leq c(u, v)$  to every arc  $(u, v) \in A$  and fulfills the constraints that for every vertex  $v$  except for the source and the sink the total flow into  $v$  equals the total flow out of  $v$ . See [15] for details. A *maximum flow* is a flow such that the total flow into the sink is maximal.

## 3 Basic results and observations

In this section, we shed light on the combinatorial relations between the problems and investigate the parameterized complexity of the considered problems with respect to the parameters “number of voters” and “number of candidates”. These results will also be useful for the following sections. In particular, some of the fixed-parameter tractability results will be used as “subroutines” in Section 4 to obtain fixed-parameter tractability for other parameterizations.

### 3.1 Relations between the problems

Although all four problems come with different properties in general, for some special cases, some of them coincide. One such example is the so-called *fully personalizable setting* [40], that is, the case that the misrepresentation bound  $R = 0$  and hence every voter is represented optimally. Clearly, asking for a set of winners with where the sum of misrepresentations is zero is equivalent to asking for a set of winners with maximum misrepresentation value zero. This leads to the following observation.

**Observation 1.** *For  $R = 0$ , MINIMAX M-MULTIWINNER coincides with M-MULTIWINNER and MINIMAX CC-MULTIWINNER coincides with CC-MULTIWINNER.*

Moreover, for the two minimax versions of the problems, it only matters whether a particular misrepresentation value exceeds the threshold  $R$  or not. Hence, an instance with an arbitrary misrepresentation function  $r$  can be reduced to an equivalent instance with the approval misrepresentation function  $r'$ ; for every voter  $v$  and every candidate  $c$ , set  $r'(v, c) = 1$  if  $r(v, c) > R$ , and  $r'(v, c) = 0$  if  $r(v, c) \leq R$  and, finally, set  $R' := 0$ .

**Observation 2.** *For a MINIMAX M-/CC-MULTIWINNER instance  $(C, V, r, R, k)$  with misrepresentation function  $r$ , there is an instance  $(C, V, r', 0, k)$  with the new misrepresentation function  $r'$  taking values 0 and 1 such that the new instance is a yes-instance if and only if the original instance is a yes-instance.*

As a direct consequence, for the minimax versions every algorithm for the approval misrepresentation function also applies to instances with the general misrepresentation function. Moreover, hardness results for an arbitrary misrepresentation function transfer to the approval misrepresentation function. Combining Observations 1 and 2, conclude that an algorithm for M-MW (CC-MW) for instances with  $R = 0$  also solves the corresponding minimax version for the general misrepresentation function.

Finally, observe that for the minimax versions in a similar style hardness results given for the approval misrepresentation function directly transfer to misrepresentation functions in which a voter is allowed to give an arbitrary number of candidates a misrepresentation value at most  $R$ . Note that this does not hold for the Borda misrepresentation function where every voter  $v$  must specify exactly  $R + 1$  candidates that can represent  $v$  with misrepresentation at most  $R$ .

### 3.2 Number of voters and candidates as parameter

We argue that all four considered problems are fixed-parameter tractable with respect to the number of candidates as well as the number of voters. Our algorithms are based on brute-force search combined with maximum flow and matching techniques. First, we consider the parameterization by the number of voters. Then, we focus on the parameterization by the number of candidates.

#### 3.2.1 Parameter number of voters

We show that all considered multiwinner problems are fixed-parameter tractable when parameterized by the number  $n$  of voters. The basic idea to establish fixed-parameter tractability is to try all  $O(n^k) \subseteq O(n^n)$  partitions of the voters into  $k$  sets in order to find a partition such that for each subset the voters are represented by the same candidate in a solution. Then, a set of candidates can be found by the computation of a matching in a bipartite auxiliary graph.

**Proposition 1.** (MINIMAX) CC-MULTIWINNER and (MINIMAX) M-MULTIWINNER can be solved in  $n^n \cdot \text{poly}(n, m)$  time for an instance with  $n$  voters and  $m$  candidates.

*Proof.* First, we present a solving strategy for MINIMAX CC-MULTIWINNER. To find a set of  $k$  winners for MINIMAX CC-MULTIWINNER, try all  $O(n^k)$  partitions of the voters into  $k$  sets. Since all partitions are tested, there must be one such partition  $V_1, \dots, V_k$  such that all voters from  $V_i$  are assigned to the same candidate  $c$  of an optimal set of  $k$  winners and no other voter is assigned to  $c$ . Hence, for every partition, it remains to select  $k$  candidates, one candidate  $c_i$  for every subset  $V_i$ , such that by assigning the voters in  $V_i$  to  $c_i$  the total misrepresentation of a voter is at most  $R$ . For MINIMAX CC-MULTIWINNER the set of candidates can be determined by computing a maximum-cardinality matching in the following bipartite graph. One partition of the graph represents the candidates and the other partition the sets  $V_1, \dots, V_k$ . Moreover, there is an edge between a vertex representing a candidate  $c$  and a vertex representing a set  $V_i$  if and only if  $r(v, c) \leq R$  for all  $v \in V_i$ . It is straightforward to verify that all voters can be represented with maximal misrepresentation bound  $R$  if and only if there is a maximum-cardinality matching of size  $k$  (all vertices representing the subsets are “matched”) in the constructed graph.

Next, we focus on CC-MULTIWINNER. Again, we try all partitions of the voters into  $k$  sets. For every such partition, we compute a maximum-weight matching in the following edge-weighted bipartite graph. Again, one partition consists of one vertex  $w_c$  for every candidate  $c$  and the other partition consists of a vertex  $w_{V_i}$  for every set of a partition  $V_1, \dots, V_k$  of the multiset of voters. Moreover, there is an edge between every vertex  $w_c$  and every vertex  $w_{V_i}$  with weight  $T - \sum_{v \in V_i} r(v, c)$ , where  $T$  is a positive integer that is large enough to ensure that all weights are positive. The crucial observation is that in a maximum-weight matching all vertices representing a subset are matched since the edge weights are positive (here we assume that  $k \leq m$ ). With this observation it is easy to verify that the computation of a maximum-weight matching yields a set of  $k$  candidates “representing” the subsets of voters as good as possible. More specifically, let  $W$  denote the weight of the maximum-weight matching. Then,  $kT - W$  is the total misrepresentation of the corresponding assignment.

Finally, observe that for the two problems where the assignment of the voters to the winners must fulfill the  $M$ -criterion we can proceed in the same way with the single exception that we need only to consider partitions such that every subset contains at most  $\lceil n/k \rceil$  voters.

Regarding the running time, the computation of a maximum-weight matching in a bipartite graph with  $n_v$  vertices and  $n_e$  edges can be accomplished in  $O(n_v(n_e + n_v \cdot \log n_v))$  time [31]. Since the number of edges and vertices in the constructed bipartite graph are polynomial in the number of candidates and  $k \leq n$ , the claimed running time follows.  $\square$

### 3.2.2 Parameter number of candidates

We show that for a fixed number of candidates all four considered multiwinner problems can be solved efficiently. For (MINIMAX) CC-MULTIWINNER parameterized by the number  $m$  of candidates fixed-parameter tractability is trivial: We can test all  $\binom{m}{k} \leq 2^m$  subsets of candidates and report a set of candidates with minimum total misrepresentation. To this end, one assigns every voter  $v$  to the candidate of the considered subset with represents  $v$  best and then directly obtains the sum or maximum misrepresentation.

Clearly, such an assignment of the voters does not have to fulfill the  $M$ -criterion. However, for (MINIMAX) M-MULTIWINNER, one can apply network flow algorithms for an optimal assignment of the voters to a size- $k$  set  $C'$  of candidates (see the Preliminaries in Subsection 2.2 for basic definitions

regarding network flows). For MINIMAX M-MULTIWINNER, construct a directed network with a vertex for every candidate from  $C'$ , one vertex for every voter, a source, and a sink vertex. There is an arc with capacity  $\lceil n/k \rceil$  from the source to every “candidate-vertex” and a capacity-one arc from a “candidate-vertex” to a “voter-vertex” if and only if the corresponding candidate can represent the corresponding voter with misrepresentation at most  $R$ . Finally, there is an arc with capacity one from every “voter-vertex” to the sink vertex. It is straightforward to verify that there is a network flow of size  $n$  if and only there is an assignment from the voters to  $C'$  that fulfills the  $M$ -criterion and every voter is represented with misrepresentation at most  $R$ .

For M-MULTIWINNER, the construction given for the minimax version can be further extended. More specifically, it follows from [49, Theorem 2], that finding an  $M$ -criterion fulfilling assignment from  $V$  to  $C'$  with minimum total misrepresentation can be accomplished in polynomial time by the computation of a transportation problem or, equivalently, by the computation of a minimum-weight maximum flow.

**Proposition 2.** (MINIMAX) CC-MULTIWINNER and (MINIMAX) M-MULTIWINNER can be solved in  $O(2^m \cdot nm)$  and  $O(2^m \cdot \text{poly}(n, m))$  time, respectively, for instances with  $m$  candidates.

## 4 Parameters number of winners and misrepresentation bound

In this section, we show that all four problems for approval as well as for the Borda misrepresentation functions are W[2]-hard with respect to the number of winners. For both misrepresentation functions we provide one parameterized reduction that works for all four problems. Moreover, we investigate the parameter misrepresentation bound  $R$ . While for the approval misrepresentation function NP-hardness for  $R = 0$  directly follows from the parameterized reduction with respect to the number of winners, for the Borda misrepresentation function, this parameter needs to be investigated separately. In particular, we show that CC-MW and M-MW are in XP with respect to  $R$ , that is, they are solvable in polynomial time for constant  $R$ . For CC-MW and M-MW with the Borda misrepresentation function, the question of fixed-parameter tractability with respect to the single parameter  $R$  is left open but we present some fixed-parameter tractability results with respect to the combined parameter  $(R, k)$  at the end of this section. An overview of the results can be found in Table 1.

### 4.1 The approval misrepresentation function

We provide a reduction from the W[2]-complete HITTING SET problem to establish W[2]-hardness for all four problems. First, we discuss related results. In the conference version [48], Procaccia et al. stated that the NP-hardness for CC-MULTIWINNER and M-MULTIWINNER follows from a reduction from MAX  $k$ -COVER (omitting the problem definition and the construction) but the reduction from the corresponding journal version [49] is given from EXACT 3-COVER. Although this is sufficient to show NP-hardness, a reduction from EXACT 3-COVER does not imply W[2]-hardness. The reduction given here is conceptually similar but requires some additional voters to deal with the fact that the sets of a HITTING SET instance might come with different/unbounded size.

**Theorem 1.** For the approval misrepresentation function, (MINIMAX) CC-MULTIWINNER and (MINIMAX) M-MULTIWINNER are W[2]-hard with respect to the number  $k$  of winners even if  $R = 0$ . MINIMAX CC-MULTIWINNER and MINIMAX M-MULTIWINNER are NP-complete.

*Proof.* First, we show W[2]-hardness for M-MULTIWINNER. Then, we argue that the presented reduction works for the other three problems as well.

Given an HS-instance  $(\mathcal{F} = \{F_1, \dots, F_n\}, U = \{u_1, \dots, u_m\}, k)$ , build an instance of M-MULTIWINNER with set  $C$  of candidates as follows. There is a candidate  $c_i \in C$  for every element  $u_i \in U$ . The multiset of voters is  $V_{\mathcal{F}} \uplus D$ , where  $V_{\mathcal{F}} := \{v_F \mid F \in \mathcal{F}\}$  and  $|D| = n(k - 1)$ . Furthermore, for every  $F \in \mathcal{F}$  and every  $u_i \in U$ , let  $r(v_F, c_i) := 0$  if  $u_i \in F$  and  $r(v_F, c_i) := 1$ , otherwise. Finally, for each  $d \in D$  and  $u_i \in U$ , set  $r(d, c_i) := 0$ . This completes the construction. For the correctness we show the following.

*Claim.* There is a hitting set of size  $k$  for  $\mathcal{F}$  if and only if there is a winner set of size  $k$  for M-MULTIWINNER that represents all voters with total misrepresentation  $R = 0$ .

“ $\Rightarrow$ ”: Let  $U'$  denote a size- $k$  hitting set for  $\mathcal{F}$  and  $C' := \{c_i \mid u_i \in U'\}$ . We show that one can build a mapping  $w : V \rightarrow C'$  that respects the  $M$ -criterion and with total misrepresentation zero. First, for every  $F \in \mathcal{F}$ , set  $w(v_F) := c_i$  for an arbitrary chosen element  $c_i \in F \cap U'$ . Clearly,  $r(v_F, c_i) = 0$ . So far, the  $n$  voters from  $V_{\mathcal{F}}$  are assigned to the candidates from  $C'$  and it remains to assign the  $n(k - 1)$  voters from  $D$ . Since each candidate in  $C'$  can represent each dummy voter in  $D$  with misrepresentation zero, we can easily extend this assignment such that each  $c' \in C'$  is assigned to exactly  $n$  voters.

“ $\Leftarrow$ ”: Let  $C' \subseteq C$  denote a size- $k$  winner set and let  $w$  be a mapping from  $V$  to  $C'$  such that  $\sum_{v \in V} r(v, w(v)) = 0$ . Since a voter  $v_F \in V_{\mathcal{F}}$  can only be represented with cost zero by a candidate  $c_i$  if  $u_i \in F$ , the set  $U' := \{u_i \mid c_i \in C'\}$  is a size- $k$  hitting set for  $\mathcal{F}$ .

This completes the proof for M-MULTIWINNER. It is straightforward to verify that the same construction yields a parameterized reduction for CC-MULTIWINNER. Finally, the W[2]-hardness for the minimax versions follows directly by Observation 1 since the reduction works for  $R = 0$ . Moreover, NP-hardness directly follows since the reduction can clearly be carried out in polynomial time and containment in NP is obvious.  $\square$

## 4.2 The Borda misrepresentation function

We refine the reduction from the previous subsection to show that also for the Borda misrepresentation function the considered problems are W[2]-hard with respect to the parameter number  $k$  of winners. However, in contrast to the case of the approval misrepresentation function the reduction does not hold for the case that  $R = 0$ . Hence, we investigate the parameter total misrepresentation  $R$  as well as the combined parameter  $(R, k)$  subsequently.

### 4.2.1 Parameter Number of Winners

For the Borda misrepresentation function, we also provide a many-one reduction from HITTING SET for M-MULTIWINNER and then argue that the presented reduction works for the other three problems as well. Note that for CC-MULTIWINNER W[2]-hardness also directly follows from the NP-hardness reduction (also from HITTING SET) provided by Lu and Boutilier [40]. In this sense, the following reduction can also be considered as to extend this reduction to work for the other problem variants; in particular, using some padding to deal with the  $M$ -criterion.

**Theorem 2.** (MINIMAX) CC-MULTIWINNER and (MINIMAX) M-MULTIWINNER are W[2]-hard with respect to the number  $k$  of winners for the Borda misrepresentation function. MINIMAX CC-MULTIWINNER and MINIMAX M-MULTIWINNER are NP-complete.

*Proof.* First, we show W[2]-hardness for M-MULTIWINNER by a parameterized reduction from HITTING SET. Given an HS-instance  $(\mathcal{F} = \{F_1, \dots, F_n\}, U = \{u_1, \dots, u_m\}, k)$  build an instance of M-MULTIWINNER as follows. Let  $z := nmk$ . The set  $C$  of candidates is  $C_U \cup B$ , where  $C_U := \{c_u \mid u \in U\}$  and  $B := \{b_i^1, \dots, b_i^z \mid 1 \leq i \leq nk\}$ . Moreover, the multiset of voters is  $V_{\mathcal{F}} \cup D$ , where  $V_{\mathcal{F}} := \{v_i \mid F_i \in \mathcal{F}\}$  and  $D := \{d_1, \dots, d_{n \cdot (k-1)}\}$ . For each voter his misrepresentation function is given by his preference list.

Each of the  $n$  “set voters”  $v_i \in V_{\mathcal{F}}$  has the following preference list:

$$\{c_u \mid u \in F_i\} > b_i^1 \dots > b_i^z > \{c_u \mid u \in U \setminus F_i\} > \{b_j^1, \dots, b_j^z \mid 1 \leq j \leq nk, j \neq i\}.$$

Finally, for each  $i \in \{1, \dots, n \cdot (k-1)\}$ , the voter  $d_i$  from  $D$  has the following preference list:

$$c_1 > c_2 \dots > c_m > b_{n+i}^1 \dots > b_{n+i}^z > \{b_j^1, \dots, b_j^z \mid 1 \leq j \leq nk, j \neq n+i\}.$$

This completes the construction. For the correctness we show the following.

*Claim.* There is a size- $k$  hitting set for  $\mathcal{F}$  if and only if there is a size- $k$  winner set for M-MULTIWINNER that represents all voters with total misrepresentation at most  $z = nmk$ .

“ $\Rightarrow$ ”: Let  $U'$  denote a size- $k$  hitting set for  $\mathcal{F}$  and  $C' := \{c_u \mid u \in U'\}$ . We build a mapping  $w : V \rightarrow C'$  as follows. First, for every  $F \in \mathcal{F}$  set  $w(v_i) := c_u$  for an arbitrarily chosen element  $c_u \in F \cap U'$ . Clearly,  $r(v_i, c_u) \leq m$  since the elements in  $F_i$  take the first positions in the preference list of  $v_i$  and  $|F_i| \leq m$ . So far, the  $n$  voters from  $V_{\mathcal{F}}$  are assigned to the candidates from  $C'$ . Since each candidate in  $C'$  can represent each dummy voter in  $D$  with misrepresentation at most  $m$ , one can extend the mapping such that exactly  $n$  voters are assigned to each  $c_u \in C'$  with misrepresentation at most  $m$  for each voter. Thus, the total misrepresentation of this assignment is at most  $nmk$ .

“ $\Leftarrow$ ”: Let  $C' \subseteq C$  denote a size- $k$  winner set and  $w$  be a mapping from  $V$  to  $C'$  such that  $\sum_{v \in V} r(v, w(v)) \leq nmk$ . First, we show that  $C'$  cannot contain a candidate  $b_i^j$  from  $B$ . Every candidate from  $B$  can represent at most one voter with better misrepresentation value than  $z$ . More specifically, if  $1 \leq i \leq n$ ,  $b_i^j$  can at most represent the voter  $v_i$  and if  $n < i \leq nk$ ,  $b_i^j$  can at most represent the voter  $d_i$  with misrepresentation at most  $z$ . Since every candidate from  $C'$  must be assigned to exactly  $n$  voters and the misrepresentation bound is  $z$ ,  $C' \cap B = \emptyset$ .

We argue that  $U' := \{u \in U \mid c_u \in C'\}$  is a hitting set for  $\mathcal{F}$ . To this end, a voter  $v_{F_i} \in V_{\mathcal{F}}$  can only be represented by a candidate  $c_u \in C_U$  with misrepresentation cost at most  $z$  if  $u \in F_i$  since all candidates  $c_{u'}$  with  $u' \in U \setminus F_i$  occur in the preference list of  $v_{F_i}$  after the candidates  $b_i^1, \dots, b_i^z$ . Thus,  $U'$  is a hitting set for  $\mathcal{F}$  of size at most  $k$ .

This completes the proof for M-MULTIWINNER. The same construction yields a parameterized reduction for CC-MULTIWINNER based on the same claim. The direction from left to right follows in complete analogy. For the other direction, the only difference is that here a solution set  $C'$  of the CC-MULTIWINNER instance might contain a candidate from  $B$ . However, if there is such a candidate  $b_i^j$ , then it can represent at most one voter (that is,  $v_i$ ) in the required misrepresentation bound and hence can be replaced by a candidate  $c_u \in F_i$  that represents the corresponding voter even better.

For the proof of MINIMAX M-MULTIWINNER and MINIMAX CC-MULTIWINNER, it follows directly from the arguments above that there is a size- $k$  hitting set for  $\mathcal{F}$  if and only if there is a winner set for MINIMAX M/CC-MULTIWINNER consisting of  $k$  candidates that represent all voters with maximum misrepresentation at most  $R := m$ . Hence, W[2]-hardness as well as NP-hardness follow.  $\square$

#### 4.2.2 Parameter Misrepresentation Bound

Recall that for the approval misrepresentation function, all four problems are NP-hard even in the fully personalized setting, that is,  $R := 0$ . In contrast, for CC-MW and M-MW for the Borda misrepresentation function, we provide polynomial-time algorithms for every constant  $R$  while the minimax versions are NP-hard for  $R \geq 1$ . First, by a simple exhaustive search strategy, one obtains the following.

**Theorem 3.** *For the Borda misrepresentation function, CC-MULTIWINNER and M-MULTIWINNER are solvable in polynomial time when the misrepresentation bound  $R$  is constant.*

*Proof.* In every solution, at most  $R$  voters can be represented with misrepresentation greater than 0. Thus, for constant values of  $R$ , one can try all  $O(|V|^R)$  subsets of at most  $R$  voters to find a set  $V' \subseteq V$  of voters that are not represented with misrepresentation value zero by an optimal winner set. For each such subset  $V'$ , for each voter of  $v \in V'$ , one further tries all possible misrepresentation values from 1 to  $R$ , that is, one tries  $O(R^R)$  possibilities for each  $V'$ . For each such possibility, the “misrepresentation value” of each voter is determined and since for Borda there is exactly one candidate that can represent a voter with a specific value, this also implies a corresponding mapping to a set of candidates. For the case of CC-MULTIWINNER, it remains to test whether the set of corresponding candidates represents all voters with total misrepresentation at most  $R$  and has size at most  $k$ . In the case of M-MULTIWINNER one additionally needs to check whether the corresponding assignment fulfills the  $M$ -criterion. It follows that in both cases an optimal set of  $k$  winners can be computed in  $O((|V| \cdot R)^R \cdot |V||C|)$  time.  $\square$

Note that Theorem 3 does not imply fixed-parameter tractability with respect to  $R$ , which remains open in this work. However, we provide fixed-parameter tractability results with respect to the combined parameter  $(R, k)$  at the end of this section. Now, we contrast the results for CC-MW and M-MW by showing that the minimax versions become provably more difficult. More specifically, we show the following.

**Theorem 4.** *For the Borda misrepresentation function, MINIMAX CC-MULTIWINNER and MINIMAX M-MULTIWINNER are solvable in polynomial time if the total misrepresentation bound  $R = 0$  and are NP-hard for every  $R \geq 1$ .*

*Proof.* For  $R = 0$  polynomial-time solvability follows directly from the fact that every voter  $v$  must be assigned to a candidate  $c$  with  $r(v, c) = 0$  and for the Borda misrepresentation function there is only one such candidate. Hence, one only needs to check if there are less than  $k$  such candidates and, for MINIMAX M-MULTIWINNER whether the corresponding assignment fulfills the  $M$ -criterion.

Now, we show NP-hardness for  $R = 1$  by a reduction from a special case of HITTING SET. More specifically, HITTING SET is even NP-hard if every set consists of two elements and every element appears in at most three sets [33, Theorem 2.4]<sup>3</sup>.

Given such a restricted HS-instance  $(\mathcal{F} = \{F_1, \dots, F_n\}, U = \{u_1, \dots, u_m\}, k)$ , build an instance of M-MULTIWINNER as follows. Identify every set from  $\mathcal{F}$  with a voter and identify every element from  $U$  with a candidate. For each  $F = \{u, v\} \in \mathcal{F}$ , let the misrepresentation of voter  $F$  be zero for the candidate  $u$  and one for the candidate  $v$ , and the remaining misrepresentation values are assigned arbitrarily to the remaining candidates. Then, the following claim is easy to see.

---

<sup>3</sup>The problem equals VERTEX COVER on cubic graphs.

```

Branch ( $V', R', C'$ ) :
1 Consider an arbitrary  $v \in V'$ ;
2  $V' := V' \setminus \{v\}$ ;
3 for each  $c \in \{c \in C \mid r(v, c) \leq R'\}$  do
4    $R' := R' - r(v, c)$  ;
5   if  $c \notin C'$  then
6      $C' := C' \cup \{c\}$ ;
7      $V' := V' \setminus \{w \in V' \mid r(w, c) = 0\}$ ;
8   if  $\sum_{w \in V'} (\min_{c' \in C'} r(w, c')) \leq R'$  then
9     return "yes";
10  if  $V' \neq \emptyset$  and  $R' \geq 0$  and  $|C'| < k$  then
11    Branch ( $V', R', C'$ );
12  else
13    return "no";
14 end

```

**Algorithm 1:** Branching strategy for CC-MULTIWINNER showing fixed-parameter tractability with respect to the combined parameter  $(R, k)$ .

*Claim:* There is a hitting set of size  $k$  if and only if there is a set of  $k$  winners such that the maximum misrepresentation per voter is at most 1.

This shows the theorem for MINIMAX CC-MW and  $R = 1$ . For MINIMAX M-MW, one can use the following observation showing NP-hardness for an even more restricted setting. It follows directly from the HITTING SET instances constructed in the NP-hardness proof [33, Theorem 2.4] that for a yes-instance there is always a hitting set such that every element “hits” either two or three sets from  $\mathcal{F}$ . More specifically, in case of a yes-instance there is a hitting set  $U' \subseteq U$  such that every  $u' \in U'$  can be assigned either to two or three sets from  $\mathcal{F}$ . Such a hitting set then one-to-corresponds to a winner set fulfilling the M-criterion and hence the theorem also follows for MINIMAX M-MW and  $R = 1$ . For every  $R > 1$ , similar arguments show NP-hardness. More specifically, here one needs to go from sets of size two over to sets of size  $R - 1$  and then can argue analogously.  $\square$

#### 4.2.3 Combined Parameter Number of Winners and Misrepresentation Bound

In this paragraph, we focus on the scenario that one has a small set of winners that can represent all the voters with small total misrepresentation modelled by a combined parameter.

**Theorem 5.** *For the Borda misrepresentation function, CC-MULTIWINNER and MINIMAX CC-MULTIWINNER are fixed-parameter tractable with respect to the combined parameter  $(R, k)$  where  $k$  denotes the number of winners and  $R$  the misrepresentation bound.*

*Proof.* We first provide a branching strategy for MINIMAX CC-MW. Start with an empty solution set  $C'$ . Step I: For an arbitrary voter  $v \in V$ , branch according to all candidates  $c$  with  $r(v, c) \leq R$ . For each possibility, add the candidate  $c$  to  $C'$  and delete each voter  $w$  with  $r(w, c) \leq R$  from  $V$ . While  $|C'| < k$  and  $V \neq \emptyset$ , go back to Step I. If  $V$  is empty, then return “yes”, else return “no”.

The correctness of the algorithm is obvious since it tries all possibilities for a winner set. Regarding the running time, one branches into  $R + 1$  possibilities for every considered voter and considers at most  $k$  voters since  $|C'|$  is increased by one in every step.

In what follows, we show how to extend this branching strategy to work for CC-MW. The branching recursion is displayed in Algorithm 1 and is invoked with the arguments  $(V, R, \emptyset)$ . Moreover,  $C$  and  $k$  are provided as global variables. Regarding the correctness, for a considered voter (Line 1), we try all possible ways of representation (Line 3) and decrease  $R'$  by the value needed for the representation of  $v$  by the corresponding candidate (Line 4). If this possibility implies that a new candidate is added to the current solution (Line 6), then we can clearly assign all voters that are optimally represented by this candidate to it and hence delete the corresponding voters (Line 7). If for any of the considered possibilities, all voters can be assigned to the current candidate set  $C'$  such that the total misrepresentation is at most  $R'$ , then the algorithm returns “yes” (Line 9).

Regarding the running time, we show that in each recursive call (Line 10) the algorithm decreases  $R'$  or increases  $|C'|$  (or both). In the initial call, clearly  $|C'|$  is increased by one. For every further call, the only case in which  $|C'|$  is not increased is that the considered candidate  $c$  is already in the current solution set. In this case, one cannot have  $r(v, c) = 0$  since then  $v$  would have been deleted from  $V'$  at the point when  $c$  has been added to  $C'$ . Hence,  $r(v, c) > 0$  and  $R'$  is decreased (Line 4). Since the recursion ends when  $R' < 0$  or  $|C'| > k - 1$ , it follows that one can have a recursion depth of at most  $R + k$ . Moreover, in each recursive call, one branches according to  $R + 1$  possible candidates (Line 3). Hence, the algorithm can be executed in  $(R + 1)^{R+k} \cdot \text{poly}(n, m)$  time.  $\square$

The branching strategy for CC-MW from the previous theorem cannot be directly transferred to M-MW since due to the M-criterion one cannot just assign a voter to a candidate even in case of an optimal representation with misrepresentation value zero. Hence, this would lead to branching possibilities in which the parameter cannot be reduced and does not result in a “search tree” of bounded size. However, we apply a structural observation based on the M-criterion which allows us to obtain fixed-parameter tractability.

**Theorem 6.** *For the Borda misrepresentation function, M-MULTIWINNER is fixed-parameter tractable with respect to the combined parameter  $(R, k)$  where  $k$  denotes the number of winners and  $R$  the misrepresentation bound.*

*Proof.* Let a zero-candidate be a candidate  $c$  with  $r(v, c) = 0$  for at least one voter  $v \in V$ . We say that such a voter  $v$  corresponds to the zero-candidate  $c$ .

**Observation 3.** *In a yes-instance, there can be at most  $R + k$  zero-candidates.*

To see the correctness, we apply a proof by contradiction. Assume that there are more than  $R + k$  zero-candidates and there is a size- $k$  winner set  $C' \subset C$  representing all voters with misrepresentation at most  $R$ . Out of the more than  $R + k$  voters that correspond to the zero-candidates at most  $k$  can be represented with misrepresentation value zero by a size- $k$  winner set. Hence, there remain more than  $R$  voters that are represented with misrepresentation at most one, a contradiction.

To make use of the bounded number of zero-candidates, we provide another observation that exploits the M-criterion of a solution.

**Observation 4.** *If the number  $n$  of voters is greater than  $(R+1)k$ , then every size- $k$  set of winners consists of zero-candidates.*

To see the correctness, we apply a proof by contradiction. Assume there are more than  $(R+1)k$  voters and a candidate  $c$  in the solution set does not represent any of the voters with misrepresentation value zero. Due to the M-criterion and since there are more than  $(R+1)k$  voters,  $c$  must represent at least  $((R+1)k)/k = R+1$  voters with misrepresentation value at least one, respectively. Since the total bound for the misrepresentation is  $R$ ,  $c$  cannot be part of a solution. This finishes the proof of Observation 4.

Now, the algorithm can be described by distinguishing two cases.

- If  $n \leq (R+1)k$ , then fixed-parameter tractability follows from Proposition 1 (showing fixed-parameter tractability w.r.t. the number of voters),
- else, there are at most  $R+k$  zero-candidates (see Observation 3) and the solution must consist of  $k$  zero-candidates (see Observation 4). After removing all but the zero-candidates, fixed-parameter tractability follows from Proposition 2 (showing fixed-parameter tractability w.r.t. the number of candidates).

Regarding the running time, the first case leads to a running time of  $((R+1)k)^{(R+1)k} \cdot \text{poly}(n, m)$  while the second case can be accomplished in  $2^{R+k} \cdot \text{poly}(n, m)$  time. Hence, the theorem follows.  $\square$

Finally, for MINIMAX M-MW with the Borda misrepresentation function, we show fixed-parameter tractability with respect to  $k$  if  $R = 1$ . The fixed-parameter tractability with respect to the combined parameter  $(R, k)$  is left open for this case. The provided algorithm relies on a reduction to the fixed-parameter tractable CAPACITATED VERTEX COVER (CVC) problem defined as follows. Given an undirected graph  $G = (U, E)$  a capacity function  $U \rightarrow \{1, \dots, |E|\}$ , and an integer  $k > 0$ , it asks whether there is a size- $k$  subset  $U' \subseteq U$  and a mapping from  $E$  to  $U'$  that assigns every edge to one vertex and the number of edges assigned to a vertex is at most its capacity.

The parameterized complexity of CVC was an open question in [35] and the first fixed-parameter tractability result has been provided by Guo et al. [36]. The currently fastest algorithm with running time  $k^{3k} \cdot n^{O(1)}$  is due to Dom et al. [17]. A simple reduction shows the following.

**Theorem 7.** *For the Borda misrepresentation function, MINIMAX M-MULTIWINNER is fixed-parameter tractable with respect to the number of winners when the misrepresentation bound  $R$  is one.*

*Proof.* Every instance of MINIMAX M-MULTIWINNER with  $R = 1$  can be transformed into a CVC instance by identifying the candidates with the vertices and adding edges as follows. For every voter, one adds one edge that is adjacent to the two candidate vertices that represent this voter with misrepresentation one and zero, respectively. Moreover, the capacity of every vertex is set to  $\lceil n/k \rceil$ . It is straightforward to see that the graph has a capacitated vertex cover of size  $k$  if and only if the MINIMAX M-MULTIWINNER instance is a yes-instance.  $\square$

Note that for  $R > 1$ , the analogous approach would be to reduce MINIMAX M-MW to CAPACITATED  $(R+1)$ -HITTING SET. However, it is not hard to see that this problem is W[1]-hard with respect to  $k$  even for  $R = 2$ .<sup>4</sup> Hence, such a reduction does not lead to fixed-parameter tractabil-

---

<sup>4</sup>A corresponding reduction from the W[1]-hard PARTIAL VERTEX COVER (PVC) [36] is sketched in the following. Given an undirected graph  $G = (U, E)$ , and integers  $k, d > 0$ , PVC asks whether there is a size  $k$ -subset  $U' \subseteq U$  such that at least  $d$  edges are incident to vertices from  $U'$ . Now, construct a CAPACITATED 3-HITTING SET instance as follows. Identify the elements with the vertices and additionally add one new element  $x$ . For every edge  $\{u, v\} \in E$  there is the set  $\{u, v, x\}$ . Finally, set the capacity of  $x$  to  $|E| - d$  and to the degree of the corresponding vertex for every other element. It is not hard to verify that there is a capacitated hitting set of size  $k$  if and only if there is a size- $k$  partial vertex cover.

ity. On the other side, it seems not obvious how to reduce CAPACITATED  $(R + 1)$ -HITTING SET to MINIMAX M-MW since the capacities of different elements might differ while the M-criterion requires to have the roughly the same “capacity” for every candidate.

## 5 Single-peaked elections

As discussed in the introduction (Subsection 1.4), single-peakedness is a central notion in political sciences where a single issue dominates the preferences of all voters. Let us now formally define single-peakedness.

**Definition 5.** Let  $V$  be a profile over a candidate set  $C$ , and let  $\sqsupseteq$  be a linear order over  $C$  (the societal axis). We say that an order  $v$  over  $C$  is *compatible* with  $\sqsupseteq$  if for all  $c, d, e \in C$  such that either  $c \sqsupseteq d \sqsupseteq e$  or  $e \sqsupseteq d \sqsupseteq c$  it holds that

$$\text{pos}_v(c) < \text{pos}_v(d) \implies \text{pos}_v(d) < \text{pos}_v(e). \quad (1)$$

We say that  $V$  is *single-peaked with respect to  $\sqsupseteq$*  if for each  $i = 1, \dots, n$  the order  $v_i$  is compatible with  $\sqsupseteq$ . A profile  $V$  is called *single-peaked* if there exists a linear order  $\sqsupseteq$  over  $C$  such that  $V$  is single-peaked with respect to  $\sqsupseteq$ ; we will say that  $\sqsupseteq$  *witnesses* the single-peakedness of  $V$  and refer to  $\sqsupseteq$  as *societal order*.

**Proposition 3.** Let  $V$  be a single-peaked profile over a set of candidates  $C$  witnessed by the societal order  $\sqsupseteq$ . Let  $r$  be a misrepresentation function for  $V$ . Then for every triple  $\{c_i, c_j, c_k\} \subseteq C$  with  $c_i \sqsupseteq c_j \sqsupseteq c_k$  or  $c_k \sqsupseteq c_j \sqsupseteq c_i$

$$r(v, c_i) < r(v, c_j) \implies r(v, c_j) \leq r(v, c_k).$$

*Proof.* By the definition of misrepresentation function (see Definition 1)  $r(v, c_i) < r(v, c_j)$  implies  $\text{pos}_v(c_i) < \text{pos}_v(c_j)$ . Now the result follows by (1) and again Definition 1.  $\square$

In this section, we investigate the computational complexity of computing proportional representation using Chamberlin & Courant and Monroe methods together with their variants, when the input profile is single-peaked. As discussed in the introduction, when the input profile is single-peaked all voters can be viewed as located on a certain axis where their location is their bliss point. Their most preferred candidate will be either the one closest on the right or the one closest on the left. Without loss of generality we may assume that for each voter her bliss point is the location of one of the candidates (who is then most preferred by her). An alternative description of a single-peaked profile would be to consider the order of candidates on the axis forgetting about the axis itself. More specifically, this order is the societal order that witnesses the single-peakedness of the profile.

It is clear that the misrepresentation function  $r(v, c)$  for a single-peaked profile must satisfy the following. If we fix the voter  $v$  and change  $c$  from one direction of the societal axis to the other the value  $r(v, c)$  should decrease monotonically to  $v$ 's most preferred candidate at the bliss point and then increase again monotonically for all candidates beyond the bliss point. That is, for each voter the function expressing the voter's misrepresentation by candidates is single-peaked (that is, has exactly one local minimum) with respect to the order that witnesses single-peakedness of the profile.

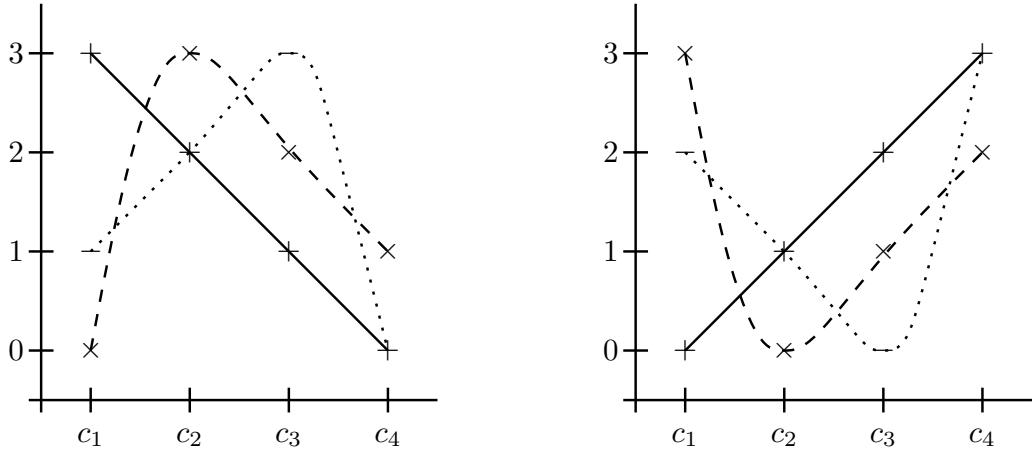


Figure 1: An election consists of three voters with the following preferences:  $c_1 > c_2 > c_3 > c_4$ ,  $c_2 > c_3 > c_4 > c_1$ , and  $c_3 > c_2 > c_1 > c_4$ . It is single-peaked witnessed by the societal order  $c_1 \sqsupset c_2 \sqsupset c_3 \sqsupset c_4$ . The diagram on the left-hand side shows, for every voter, the Borda score that each alternative gets from this voter marked by the solid line, dashed line and dotted line, respectively. Note that every “preference order” has one local maximum. If the voters express their Borda misrepresentations values instead, then one obtains the diagram on the right. Here, the misrepresentation function for an arbitrary fixed voter has one local minimum.

Before describing our results, we briefly outline a typical shape of some most prominent misrepresentation functions in single-peaked settings. The Borda misrepresentation function is strictly ascending when moving away from the local minimum in both directions. Moreover, if from the candidate preceding the candidate at the local minimum the misrepresentation function drops by  $d > 0$  points, then, for the next  $d - 1$  candidates on the other side of the local minimum, the misrepresentation function must ascend in size-one steps.

In contrast, for the approval misrepresentation function, there is exactly one interval of consecutive candidates on the societal axis for whom the misrepresentation is zero while for all remaining candidates outside the interval the misrepresentation is one. For the minimax variants one obtains a similar structure, in the sense that there can be only one interval in which a particular voter can be represented without exceeding the given misrepresentation bound. Note that there is some remote similarity here between the last two cases and preferences over intervals in the aggregating range values model introduced by Farfel and Conitzer [28].

In the remainder of this section, we provide the following results (see also Table 2, Page 8). We show that CC-MULTIWINNER, MINIMAX CC-MULTIWINNER, and MINIMAX M-MULTIWINNER for single-peaked elections can be solved in polynomial time for an arbitrary misrepresentation function (Theorem 8, Theorem 9, and Theorem 10, respectively). In contrast to the three aforementioned problems, we present a reduction from an NP-hard version of the EXACT 3-COVER problem which shows that M-MULTIWINNER is NP-hard even when restricted to single-peaked profiles (Theorem 13). However, for the approval misrepresentation function, we still obtain polynomial-time solvability for M-MULTIWINNER and single-peaked input profiles (Theorem 12). We leave open the computational complexity for M-MULTIWINNER for the Borda misrepresentation function in the single-peaked case.

**Function** SinglePeaked-CC-MW( $V, C, r, k$ )  
**Input:** A multiset of voters  $V := \{v_1, \dots, v_n\}$ , a set of candidates  $C := \{c_1, \dots, c_m\}$ , a misrepresentation function  $r$ , and a positive integer  $k$ . The voters have single-peaked preferences according to the societal order  $\sqsupseteq$ , where  $c_1 \sqsupseteq c_2 \sqsupseteq \dots \sqsupseteq c_m$ .  
**Output:** The minimum total misrepresentation for  $k$  winners.

```

1 begin
2   for  $i = 1, \dots, m$  do
3      $z(i, 1) := \sum_{v \in V} r(v, c_i);$ 
4   end
5   for  $i = 2, \dots, m$  do
6     for  $j = 2, \dots, \min(k, i)$  do
7        $z(i, j) := \min_{p \in \{j-1, \dots, i-1\}} (z(p, j-1) - \sum_{v \in V} \max\{0, r(v, c_p) - r(v, c_i)\});$ 
8     end
9   end
10  return  $\min_{i \in \{k, \dots, m\}} (z(i, k));$ 
11 end
```

**Algorithm 2:** Dynamic programming algorithm for CC-MULTIWINNER for single-peaked input profiles.

## 5.1 (Minimax) CC-Multiwinner

We show that on single-peaked input profiles CC-MULTIWINNER and MINIMAX CC-MULTIWINNER are polynomial-time solvable for an arbitrary misrepresentation function. We first provide a dynamic programming algorithm for the case of CC-MULTIWINNER. Second, we show that MINIMAX CC-MULTIWINNER can be solved optimally by a greedy algorithm when the input profile is single-peaked.

### 5.1.1 A dynamic programming procedure for CC-Multiwinner

For CC-MULTIWINNER the polynomial-time solvability is established by presenting a dynamic programming algorithm leading to the following.

**Theorem 8.** *For a single-peaked input profile and an arbitrary misrepresentation function CC-MULTIWINNER can be solved in  $O(nm^3)$  time.*

*Proof.* For a set  $C' \subseteq C$ , the *minimum total misrepresentation* is defined as

$$s(C') = \sum_{v \in V} \min_{c' \in C'} \{r(v, c')\}.$$

We define a dynamic programming table  $z$ , containing an entry  $z(i, j)$  for all  $1 \leq i \leq m$  and all  $1 \leq j \leq \min\{i, k\}$ . Informally speaking, the entry  $z(i, j)$  gives the minimum total misrepresentation for a set of  $j$  winners from  $\{c_1, \dots, c_i\}$  including  $c_i$ .

The dynamic programming procedure **SinglePeaked-CC-MW** is provided in Algorithm 2. We show that it correctly solves CC-MULTIWINNER in the claimed running time. Regarding the correctness, we will show that after the execution of **SinglePeaked-CC-MW** the following equation is

satisfied

$$z(i, j) = \min\{s(C') \mid C' \subseteq \{c_1, \dots, c_i\} \wedge |C'| = j \wedge c_i \in C'\}. \quad (2)$$

Then, the minimum total misrepresentation of an optimal size- $k$  winner set is clearly given by  $\min_{i \in \{k, \dots, m\}} z(i, k)$  (see Line 10).

The proof of Equation 2 follows by induction on  $j$ . First, we argue that the entries  $z(i, 1)$  fulfill Equation (2), yielding the induction base. To this end, observe that if there is only one candidate  $c_i$  in the winner set, then all voters must be assigned to  $c_i$ , yielding the misrepresentation sum  $s(\{c_i\}) = \sum_{v \in V} r(v, c_i)$ , see Line 3.

Next, we show that an entry  $z(i, j)$  with  $j > 1$  (as computed in Line 7) complies with Equation (2) provided that  $z(p, j-1)$  does for all  $1 \leq p < i$ . Consider a set  $C^* \subseteq \{c_1, \dots, c_i\}$  with  $c_i \in C^*$  of  $j$  candidates such that  $s(C^*)$  is minimum among all such sets. We argue that  $z(i, j) = s(C^*)$ . Let  $p < i$  such that  $c_p \in C^*$  and  $c_\ell \notin C^*$  for all  $p < \ell < i$ . This implies that  $p \geq j-1$ . The crucial observation is as follows. If for a voter  $v$  it holds that  $r(v, c_i) \leq r(v, c_p)$ , then the single-peakedness implies that  $r(v, c_q) \geq r(v, c_p) \geq r(v, c_i)$  for all  $q < p$ . Hence, if we consider a set  $C''$  of  $j-1$  candidates from  $\{c_1, \dots, c_p\}$  with  $c_p \in C''$ , then we can assume that the value  $r(v, c_p)$  is the contribution of voter  $v$  to the total misrepresentation  $s(C'')$ . Hence, by adding  $c_i$  to  $C''$  there is an improvement of  $r(v, c_p) - r(v, c_i)$  for each voter  $v$  with  $r(v, c_p) \geq r(v, c_i)$ . For every remaining voter  $v$ , it holds that  $r(v, c_i) \geq r(v, c_p)$  and hence one cannot improve the representation by assigning it to  $c_i$ . It follows that  $s(C^*) = s(C'') - \sum_{v \in V} \max\{0, r(v, c_p) - r(v, c_i)\}$  and by the induction assumption we have  $z(j-1, p) = s(C'')$ . Finally, since the algorithm tries all possible choices of  $p$  (see Line 7) we have that  $z(i, j) = s(C^*)$ .

It is straightforward to verify that the overall running time is  $O(m^3n)$  (see Algorithm 2).  $\square$

### 5.1.2 A greedy algorithm for Minimax CC-Multiwinner

For single-peaked input profiles, the minimax version of CC-MULTIWINNER can be solved by a greedy algorithm presented in Algorithm 3. Basically, the algorithm iterates over the candidates according to the societal order and puts into the solution the first candidate for whom there is a voter that cannot be represented by a later candidate.

**Theorem 9.** *For a single-peaked input profile and an arbitrary misrepresentation function MINIMAX-CC-MULTIWINNER can be solved in  $O(nm)$  time.*

*Proof.* We show that the procedure **SP-MiniMax-CC-MW** (Algorithm 3) decides MINIMAX-CC-MULTIWINNER correctly (and if it outputs “yes”, the set  $C'$  is a valid winner set). To this end, we argue that at every step of the algorithm, the following two points hold:

1. at least one candidate with index at most  $p$  must be in the solution,
2. if for a voter  $v \in V'$ ,  $r(v, c_i) \leq R$  for an  $i \in \{1, \dots, p-1\}$ , then  $r(v, c_p) \leq R$ .

The first point is a direct consequence of the single-peakedness, implying that when a voter gets over the misrepresentation threshold  $R$ , then he cannot be represented with misrepresentation threshold at most  $R$  by any of the following candidates. Hence, at least one candidate with index at most  $p$  must be in the solution to represent a voter  $v \in V$  with  $p(v) = p$ .

The second point can be seen by a simple proof by contradiction: Assume that there is a voter  $v \in V'$  such that  $r(v, c_i) \leq R$  for an  $1 \leq i \leq p-1$  and  $r(v, c_p) > R$ . In such a case single-peakedness of the profile implies that  $r(v, c_j) > R$  for every  $m \geq j > p$ . Then, however, in the corresponding

**Function** SP-MiniMax-CC-MW( $V, C, r, R, k$ )  
**Input:** A multiset of voters  $V := \{v_1, \dots, v_n\}$ , a set of candidates  $C := \{c_1, \dots, c_m\}$ , a misrepresentation function  $r$ , a maximum misrepresentation bound  $R$ , and a positive integer  $k$ . The voters have single-peaked preferences according to the societal order  $\sqsupseteq$ , where  $c_1 \sqsupseteq c_2 \sqsupseteq \dots \sqsupseteq c_m$ .  
**Output:** “Yes”, if there are  $k$  winners with total misrepresentation  $R$ ; otherwise, “No”

```

1 begin
2   for  $v \in V$  do  $p(v) := \max\{i \in \{1, \dots, m\} \mid r(v, c_i) \leq R\}$  (rightmost candidate that can
      represent  $v$ );
3   if  $p(v) = \emptyset$  for some  $v \in V$  then
4     return “No”;
5    $V' := V$  (the multiset of voters that are not represented so far);
6    $C' := \emptyset$  (the set of winners);
7   while  $|C'| < k$  and  $V' \neq \emptyset$  do
8      $p := \min\{p(v) \mid v \in V'\}$ ;
9      $C' := C' \cup \{c_p\}$ ;
10    Remove all voters  $v \in V'$  with  $r(v, c_p) \leq R$  from  $V'$ ;
11  end
12  if  $V' = \emptyset$  then
13    return “Yes”;
14  else
15    return “No”;
16 end
```

**Algorithm 3:** Greedy algorithm for MINIMAX-CC-MULTIWINNER for single-peaked input profiles.

step of the algorithm not  $c_p$  but a candidate with a smaller index would have been selected (Line 8) since  $p(v)$  would be smaller than  $p$ .

Regarding the running time, the sets  $p(v)$  can be computed and stored for all voters in  $O(nm)$  time. The search for a minimum  $p$  (Line 8) can be accomplished in  $O(n)$  time and the set of voters assigned to  $c_p$  (Line 10) can be found in  $O(n)$  time as well. Finally, the while loop (Line 7) is executed at most  $k \leq m$  times.  $\square$

## 5.2 (Minimax) M-Multiwinner

We focus on the case that the assignment of the candidates to the winner set satisfies the M-criterion, that is, it is required that each winner represents about the same number of candidates. This additional constraint makes the winner determination more involved. Indeed, we can show that for an integer-valued misrepresentation function M-MULTIWINNER is NP-hard even if the input profile is single-peaked. On the positive side, we show that M-MULTIWINNER for the approval misrepresentation function and MINIMAX M-MULTIWINNER for arbitrary misrepresentation functions are polynomial-time solvable for single-peaked input profiles. However, the solving strategies (that are also based on dynamic programming) are more intricate than for (MINIMAX) CC-MULTIWINNER. For proving polynomial-time solvability we establish a close relationship to the so-called 1-DIMENSIONAL RECTANGLE STABBING. We start with the polynomial-time algorithms

followed by the NP-hardness proof. The computational complexity for M-MULTIWINNER for the Borda misrepresentation function for single-peaked input profiles is left open.

### 5.2.1 M-Multiwinner for Approval and Minimax M-Multiwinner

The results of this subsection rely on a close connection between the considered problems and a rectangle stabbing problem as described below. More specifically, for single-peaked input profiles, the polynomial-time solvability for MINIMAX M-MULTIWINNER and the special case of M-MULTIWINNER where the misrepresentation bound is zero can be directly obtained by a polynomial-time algorithm for the one-dimensional rectangle stabbing problem by Even et al. [23]. For M-MULTIWINNER for the approval misrepresentation function Even et al.’s algorithm cannot be applied directly but the polynomial-time solvability when having single-peaked input profiles can be extended to this case as well. We start with introducing the rectangle stabbing problem and explain its relation to (MINIMAX) M-MULTIWINNER.

Even et al. [23] introduce the following capacitated version of 1-DIMENSIONAL RECTANGLE STABBING (in what follows, we use the same notation as Even et al. [23] whenever possible). The input consists of a set  $\mathcal{U}$  of horizontal intervals and a set  $\mathcal{S}$  of vertical lines with capacity  $c(S) \in \{0, \dots, |\mathcal{U}|\}$  for every line  $S \in \mathcal{S}$ . Informally, the task is to cover (or stab) all intervals by a minimum number of vertical lines from  $\mathcal{S}$ , where each line  $S$  covers at most  $c(S)$  intervals (a vertical line can cover a horizontal interval if they intersect). Since a line  $S \in \mathcal{S}$  can cover at most  $c(S)$  intervals, one has to specify which interval is assigned to which line in the solution. Let  $\mathcal{U}(S)$  denote the set of intervals from  $\mathcal{U}$  intersecting with  $S \in \mathcal{S}$ . An assignment is a function  $A: \mathcal{S} \rightarrow 2^{\mathcal{U}}$ , where  $A(S) \subseteq \mathcal{U}(S)$ . A set  $\mathcal{S}' \subseteq \mathcal{S}$  is a *cover* if there is an assignment  $A$  with  $|A(S)| \leq c(S)$  for all  $S \in \mathcal{S}$  and  $\bigcup_{S \in \mathcal{S}'} A(S) = \mathcal{U}$ .

ONE-DIMENSIONAL RECTANGLE STABBING WITH HARD CONSTRAINTS (HARD-1-RS):

**Input:** As set  $\mathcal{U}$  of horizontal intervals and as set  $\mathcal{S}$  of vertical lines with capacities  $c(S) \in \{0, \dots, |\mathcal{U}|\}$  for every line  $S \in \mathcal{S}$ .

**Task:** Find a minimum-cardinality cover  $\mathcal{S}' \subseteq \mathcal{S}$  (and a corresponding assignment).

An instance of M-MULTIWINNER with  $R = 0$  and single-peaked input profile can be reduced to HARD-1-RS as follows. For every candidate there is a vertical line according to its position in the societal order. Since each voter  $v$  must be represented by a candidate  $c$  with  $r(v, c) = 0$  and all the candidates with  $r(v, c) = 0$  are ordered consecutively in the societal order, we can represent each voter by a horizontal interval reaching from the leftmost candidate  $c$  with  $r(v, c) = 0$  to the rightmost such candidate. Finally, each vertical line is associated with a capacity of  $\lceil n/k \rceil$ . Clearly, there is a solution for the M-MULTIWINNER instance with  $R = 0$  if and only if there is a size- $k$  cover of the constructed instance of HARD-1-RS.

Even et al. [23] presented a dynamic programming for HARD-1-RS with running time  $O(|\mathcal{U}|^2 \cdot |\mathcal{S}|^2 \cdot (|\mathcal{U}| + |\mathcal{S}|))$ . Since the transformation described above can be easily accomplished in linear time, one directly obtains the following.

**Corollary 1.** M-MULTIWINNER for instances with single-peaked profile and  $R = 0$  (and an arbitrary misrepresentation function) can be solved in  $O(n^2 \cdot m^2 \cdot (n + m))$  time.

Recall that an instance of MINIMAX M-MULTIWINNER with  $R > 0$  can be reduced to an instance of M-MULTIWINNER with  $R' = 0$  by setting for each voter  $v$  and each candidate  $c$  the misrepresentation value to zero if  $r(v, c) \leq R$  and to one otherwise (see Observation 2). Moreover, by

Observation 1 MINIMAX M-MULTIWINNER and M-MULTIWINNER coincide for  $R = 0$ . Altogether, we arrive at the following.

**Theorem 10.** MINIMAX M-MULTIWINNER for single-peaked input profiles (and for an arbitrary misrepresentation function) can be solved in  $O(n^2 \cdot m^2 \cdot (n + m))$  time.

Finally, we show that for single-peaked input profiles, M-MULTIWINNER for the approval misrepresentation function (and arbitrary misrepresentation bound  $R$ ) can be solved in polynomial time. To this end, we show that these instances can be reduced to a version of capacitated one-dimensional rectangle stabbing where the goal is to stab a maximum number of horizontal intervals with  $k$  vertical lines. More specifically, we introduce the following problem which to the best of our knowledge has not been studied before.

MAXIMUM ONE-DIMENSIONAL RECTANGLE STABBING WITH HARD CONSTRAINTS (MAX-HARD-1-RS):

**Input:** A set  $\mathcal{U} = \{u_1, \dots, u_n\}$  of horizontal intervals and a set  $\mathcal{S} = \{S_1, \dots, S_m\}$  of vertical lines with capacity  $c(S) \in \{1, \dots, n\}$  for every line  $S \in \mathcal{S}$ , and a positive integer  $k$ .

**Task:** Find a size- $k$  set  $\mathcal{S}' \subseteq \mathcal{S}$  and an assignment  $A$  with  $|A(S)| \leq c(S)$  for each  $S \in \mathcal{S}'$  such that  $|\bigcup_{S \in \mathcal{S}'} A(S)|$  is maximal.

The main difference between MAX-HARD-1-RS and HARD-1-RS is that in the case of HARD-1-RS *all* intervals must be covered by a minimum number of lines whereas in case of MAX-HARD-1-RS the goal is to cover a *maximum* number of intervals with  $k$  lines. Note that a polynomial-time algorithm for MAX-HARD-1-RS would imply a polynomial-time algorithm for HARD-1-RS. Indeed, we could apply an algorithm for MAX-HARD-1-RS for increasing values of  $k$  (starting with  $k = 1$ ) to find the minimum number  $k$  such that all intervals are covered. On the other hand, it is not obvious how a polynomial-time algorithm for HARD-1-RS can be used for solving MAX-HARD-1-RS. However, we show that the dynamic programming algorithm of Even et al. [23] for HARD-1-RS can be adapted to work for MAX-HARD-1-RS. To this end, we employ the same decomposition property (stated in Observation 5 below) as Even et al. but the dynamic programming table and the algorithm are different.

We introduce the following notation needed for dynamic programming. Following Even et al., for an interval  $u \in \mathcal{U}$ , let  $r(u)$  denote the right endpoint of  $u$  and  $l(u)$  denote the left endpoint of  $u$  (that is,  $l(u) \leq r(u)$ ). We assume that all vertical lines are at integral coordinates, all end- and start-points of the horizontal intervals are integers and that at each coordinate there is at most one interval that starts or ends at this coordinate. As also mentioned by Even et al. [23] this does not constitute any restriction since every instance of MAX-HARD-1-RS can be easily transformed into an equivalent instance with at most  $2n + m$  coordinates and at each coordinate at most one interval starts or ends at it. Moreover, we assume that the intervals  $u_1, \dots, u_n$  are ordered so that  $l(u_i) < l(u_j)$  for all  $i < j$ . Let  $x(S)$  denote the coordinate of line  $S \in \mathcal{S}$ . We assume that  $l(u_1) = 1$ . Let  $N := r(u_n)$  and  $S(x)$  denote the vertical line associated with  $x \in [1, N]$ . Note that  $S(x)$  is not necessarily contained in  $\mathcal{S}$ . For two integers  $x_1 \leq x_2$  we use  $\mathcal{S}(x_1, x_2)$  to denote the lines from  $\mathcal{S}$  with coordinates from  $[x_1, x_2]$ , that is,  $\mathcal{S}(x_1, x_2) = \{S(x) \mid x \in [x_1, x_2] \wedge S(x) \in \mathcal{S}\}$ .

The algorithm makes use of the fact that there is always an optimal solution that satisfies the following *leftmost interval first* property [23]. Let  $\mathcal{S}'$  denote a size- $k$  set of lines and let  $A$  denote an assignment. We say that  $(\mathcal{S}', A)$  has the leftmost interval first property if the following holds.

Let  $S \in \mathcal{S}'$  and let  $u_i \in A(S)$ . Every  $S' \in \mathcal{S}'$  with  $l(u_i) \leq x(S') < x(S)$  (if exists) is assigned to  $\min(|U(S')|, c(S'))$  intervals and for every  $u_j$  with  $u_j \in A(S')$ , either  $j < i$  or  $r(u_j) < x(S)$ .

The dynamic programming is based on the following decomposition property which follows directly from the leftmost interval first property.

**Observation 5** ([23]). *Let  $(\mathcal{S}', A)$  be an optimal assignment that satisfies the leftmost interval first property. For any range  $[x_1, x_2] \subseteq [1, N]$ , let  $u$  be the interval with the minimum  $l(u)$  among the intervals covered by a line from  $\mathcal{S}(x_1, x_2)$ . If  $u$  is covered by line  $S$ , then the right endpoint of all intervals covered by lines in the range  $[x_1, x(S) - 1]$  are to the left of  $x(S)$ .*

Basically, Observation 5 is used in the algorithm in the following way. Consider the range  $[x_1, x_2]$  with  $x_2 = x_n$  and assume that  $S(x_1)$  is the first (leftmost) line of the considered solution. Moreover, assume that  $u$  is the leftmost interval that is covered by a line  $S(x)$  from  $\mathcal{S}(x_1, x_2)$ . Then, every interval  $v$  with  $l(v) < l(u)$  will not be covered by the solution, every interval  $u_l$  with  $l(u) < l(u_l) < x$  can only be covered by lines from  $S(x_1, x-1)$ , and every interval  $u_r$  with  $x < l(u_r)$  can only be covered by lines from  $S(x, x_2)$ . This implies a decomposition of the instance into two subinstances. A “left” instance consisting of the intervals  $u_l$  with  $l(u) < l(u_l) < x$  and a “right” instance consisting of the intervals  $u_r$  with  $x < l(u_r)$ . Basically, in any dynamic programming procedure the algorithm tries to combine all solutions for the left subinstance with the solutions for the right subinstance to find a solution for the whole instance.

**Theorem 11.** MAXIMUM ONE-DIMENSIONAL RECTANGLE STABBING WITH HARD CONSTRAINTS can be solved in  $O(n^5 mk^3)$  time.

*Proof.* We use the following definitions to state the algorithm. For  $u_i \in \mathcal{U}$  and for any two coordinates  $x_1 \leq x_2$  such that  $r(u_i) \in [x_1, x_2]$ , let

$$\mathcal{U}(u_i, x_1, x_2) := \{u_j \in \mathcal{U} \mid j \geq i \wedge r(u_j) \in [x_1, x_2]\}.$$

Note that for  $x_1 = x_2 = r(u_i)$ , it holds that  $\mathcal{U}(u_i, x_1, x_2) = \{u_i\}$  (because of the assumption that at each position there is at most one interval that ends or starts at this position).

The dynamic programming table  $\Pi(u_i, x_1, x_2, k', b)$  is defined for every  $u_i \in \mathcal{U}$ , for any two coordinates  $x_1 \leq x_2$  such that  $r(u_i) \in [x_1, x_2]$  and  $S(x_1) \in \mathcal{S}$ , for each  $1 \leq k' \leq k$ , and for each  $0 \leq b \leq c(S(x_1))$ . Informally, the table entry contains the maximum number of intervals from  $\mathcal{U}(u_i, x_1, x_2)$  that can be covered by  $k'$  lines in  $\mathcal{S}(x_1, x_2)$  under the assumption that  $S(x_1)$  is contained in the solution,  $u_i$  is covered by a line from  $S(x_1, x_2)$ , and that the capacity of  $S(x_1)$  is  $b$ .

For  $u_i \in \mathcal{U}$ , two coordinates  $x_1 \leq x_2$  such that  $r(u_i) \in [x_1, x_2]$ , and  $x \in \mathcal{S}(x_1, x_2)$  let

$$\mathcal{U}_l(u_i, x, x_1, x_2) := \{u_j \in \mathcal{U} \mid j > i \wedge r(u_j) \in [x_1, x-1]\}, \text{ and}$$

$$\mathcal{U}_r(u_i, x, x_1, x_2) := \mathcal{U}(u_i, x_1, x_2) \setminus (\mathcal{U}_l(u_i, x, x_1, x_2) \cup \{u_i\})$$

Finally, we can describe the algorithm. The initialization is as follows. For each  $u_i \in \mathcal{U}$  and for every two coordinates  $x_1 \leq x_2$  such that  $r(u_i) \in [x_1, x_2]$  and  $S(x_1) \in \mathcal{S}$ , and for every integer  $b \in [0, c(S(x_1))]$ , let

$$\Pi(u_i, x_1, x_2, 1, b) := \min(b, |\{u \in \mathcal{U}(u_i, x_1, x_2) : x_1 \in u\}|).$$

The update of a table entry  $\Pi(u_i, x_1, x_2, k', b)$  is provided by Algorithm 4 and the iteration loops by Algorithm 5. Finally, the algorithm returns the value of

$$\max_{u_i \in U, b, x_1} \Pi(u_1, x_1, N, k, b).$$

```

Function Update $\Pi(u_i, x_1, x_2, k', b)$  :
1 begin
2    $m = 0;$ 
3   for every  $x = \max(x_1, l(u_i))$  to  $r(u_i)$  with  $S(x) \in \mathcal{S}$  do
4     for all  $k_l \geq 1$  and  $k_r \geq 1$  with  $k_l + k_r = k'$  do
5        $m_l := 0;$ 
6       for  $u'_j \in \mathcal{U}_l(u_i, x_1, x_2, x)$  (compute minimum solution for “left” subinstance) do
7          $l_l := \Pi(u'_j, x_1, x - 1, k_l, b);$ 
8          $m_l := \max(l_l, m_l);$ 
9       end
10       $m_r := 0;$ 
11      for  $u''_j \in \mathcal{U}_r(u_i, x_1, x_2, x)$  (compute minimum solution for “right” subinstance) do
12        if  $x = x_1$  then
13           $l_r = \Pi(u''_j, x_1, x_2, k', b - 1)$  ( $x_1$  is used to cover  $u_i$ );
14        else
15           $l_r := \Pi(u''_j, x, x_2, k_r, c(S(x)) - 1)$  ( $x \neq x_1$  is used to cover  $u_i$ );
16           $m_r := \max(l_r, m_r);$ 
17        end
18         $m := \max(m, m_l + m_r);$ 
19      end
20    end
21     $\Pi(u_i, x_1, x_2, k', b) := m + 1;$ 
22 end

```

**Algorithm 4:** Update step employed by the dynamic programming algorithm for MAX-HARD-1-RS presented in the proof of Theorem 11.

Regarding the correctness, we discuss that in every stage of the dynamic programming an entry  $\Pi(u_i, x_1, x_2, k', b)$  contains the maximum number of intervals from  $\mathcal{U}(u_i, x_1, x_2)$  that can be covered by  $k'$  lines in  $\mathcal{S}(x_1, x_2)$  if

- $S(x_1)$  is part of the solution,
- $u_i$  is covered by a line from  $S(x_1, x_2)$ , and
- the capacity of  $S(x_1)$  is  $b$  (while the other capacities are as specified in the input).

For the initialization step, this is clearly fulfilled since it allows only one line to be part of the solution and since this line must be  $S(x_1)$ , it can cover at most  $\min(b, |\{u \in \mathcal{U}(u_i, x_1, x_2) : x_1 \in u\}|)$  intervals. Regarding the update step (see Algorithm 4),  $u_i$  must be covered by one of the lines in the considered range and all such possibilities are considered by the variable  $x$  (Line 3). Then, according to the Observation 5, the instance can be divided into two subinstances. All combinations of sizes of the subsolutions are tested by iterating over  $k_l$  and  $k_r$  (Line 4). Lines 6–8 compute then an optimal subsolution for the “left” instance (which is zero for  $x = x_1$ ) and Lines 11–16 compute a solution for the right subinstance obtained after assigning  $u_i$  to  $S(x_1)$ .

By iterating over the table entries as described in Algorithm 5, one ensures that the algorithm

```

Main :
1 for  $k' = 2, \dots, k$  do
2   for  $d = 1, 2, \dots, N - 1$  do
3     for every  $x_1$  with  $S(x_1) \in \mathcal{S}$  do
4       if  $x_1 + d \leq N$  and  $|S(x_1, x_1 + d)| \geq k'$  then
5          $x_2 := x_1 + d$ ;
6         for  $b = 1, \dots, c(S(x_1))$  do
7           for  $u_i \in U(u_i, x_1, x_2, k', b)$  do
8              $\Pi(u_i, x_1, x_2, k', b) := \text{Update}\Pi(u_i, x_1, x_2, k', b)$ ;
9         end
10        end
11      end
12    end
13  end
14 end

```

**Algorithm 5:** Main loop after the initialization.

only access other entries that have been computed before. Hence, the overall correctness follows.

Regarding the running time, the update can be accomplished in  $O(kN^3m)$  time (see Algorithm 4) and the overall loop gives an additional factor of  $O(N^2k^2)$  (see Algorithm 5). Hence the claimed running time follows.  $\square$

Since M-MULTIWINNER for the approval misrepresentation function for instances with a single-peaked input profile can be reduced to MAX-HARD-1-RS in linear time in a straightforward way, one arrives at the following.

**Theorem 12.** M-MULTIWINNER for the approval misrepresentation function and single-peaked input profiles can be decided in  $O(n^5mk^3)$  time.

### 5.2.2 NP-hardness of M-Multiwinner for a single-peaked election

Contrasting the polynomial-time solvability results for the other three considered problems, we show that there is an integer-valued misrepresentation function such that M-MULTIWINNER is NP-complete even restricted to instances coming with a single-peaked input profile. More specifically, we show that M-MULTIWINNER is NP-hard even for single-peaked input profiles and integer-valued misrepresentation functions such that the maximum misrepresentation value of a voter is bounded from above by a polynomial in the number of candidates. Note that for establishing the NP-hardness we have to allow that a voter can assign the same misrepresentation value to several candidates.

The NP-hardness follows by a reduction from a restricted variant of EXACT 3-COVER.

#### RESTRICTED EXACT 3-COVER (RX3C)

**Input:** A family  $\mathcal{S} := \{S_1, \dots, S_m\}$  of sets over elements  $E := \{e_1, \dots, e_n\}$  such that every set from  $\mathcal{S}$  has size three and every element occurs in exactly three sets.

**Question:** Is there a subset  $\mathcal{S}' \subseteq \mathcal{S}$  such that every element occurs in exactly one set of  $\mathcal{S}'$  and  $\bigcup_{S \in \mathcal{S}'} S = E$ ?

Such a set  $\mathcal{S}'$  is called an *exact 3-cover* of  $E$ . Since in every yes-instances  $n$  is a multiple of three, in what follows we assume that  $n$  is divisible by three. The NP-hardness of RX3C follows from an NP-hardness reduction for the case that every element occurs in at most three subsets [32] and a construction to extend this NP-hardness to the case that every element occurs in exactly three subsets [34].

**Theorem 13.** M-MULTIWINNER is NP-hard for single-peaked input profiles and an integer-valued misrepresentation function even if the maximum misrepresentation value of every voter is polynomial in the number of candidates (and every winner represents exactly three voters).

*Proof.* We use the following notation. Consider an RX3C instance  $(\mathcal{S}, E)$ . For an element  $e \in E$  that occurs in the three subsets  $S_i, S_j$ , and  $S_k$  with  $i < j < k$ , we say that the first occurrence of  $e$  is in  $S_i$ , the second occurrence is in  $S_j$ , and the third occurrence is in  $S_k$ .

For an RX3C instance  $(\mathcal{S}, E)$ , define an M-MW instance as follows. The set of candidates is

$$C := E \cup \{s_j \mid S_j \in \mathcal{S}\}$$

and the multiset of voters is

$$V := \{v_i^x \mid e_i \in E \text{ and } x \in \{1, 2, 3\}\} \cup \{f_i \mid e_i \in E\}.$$

That is, there is a candidate for each element and each subset and there are four voters for each element. Next, we specify the misrepresentation functions of the voters:

for $i \in \{1, \dots, n\}$	$r(f_i, e_i) := 0$
for $i \in \{1, \dots, n\}, c \in C \setminus \{e_i\}$	$r(f_i, c) := 2n^2 + 1$
for $i \in \{1, \dots, n\}, x \in \{1, 2, 3\}, 1 \leq z \leq i$	$r(v_i^x, e_z) := i + z - 1$
for $i \in \{1, \dots, n\}, x \in \{1, 2, 3\}, z > i$	$r(v_i^x, e_z) := 2n^2 + 1$
for $1 \leq j \leq m, x \in \{1, 2, 3\}$ , if the $x$ th occurrence of $e_i$ is in $S_j$	$r(v_i^x, s_j) := 0$
else	$r(v_i^x, s_j) := 1$

Finally, set the misrepresentation bound to  $R := 2n^2$  and let the number of winners be  $k := n/3 + n$ . Before showing the correctness of the reduction, we discuss three crucial properties of the construction.

First, we verify that the profile is single-peaked witnessed by the societal order

$$s_1 \sqsupset \cdots \sqsupset s_m \sqsupset e_1 \sqsupset \cdots \sqsupset e_n.$$

For every voter  $f_i$  single-peakedness is obvious since his misrepresentation is 0 for one candidate and  $2n^2 + 1$  for every other candidate. For every  $v_i^x$ , within the candidate set  $E$ , the misrepresentation function decreases monotonously when we move from  $e_n$  to  $e_1$  along the societal axis: For  $z > i$ , this is obvious since the misrepresentation remains at the value  $2n^2 + 1$  and for  $z \leq i$  the misrepresentation value is  $i + z - 1$  and hence the function clearly assumes smaller values for decreasing values of  $z$ . This settles the single-peakedness for the “range” from  $e_1$  to  $e_n$ . To see the overall single-peakedness, first note that for  $e_1$  the misrepresentation of every  $v_i^x$  is at least one. Then, since the misrepresentation is one for all but one of the candidates from  $\{s_1, \dots, s_m\}$  and zero for the remaining candidate, single-peakedness for every  $v_i^x$  follows.

Second, since there are  $4n$  voters and  $k = (4n)/3$ , exactly three voters have to be assigned to every winner candidate of a solution.

Third, we show that the four voters that can be strictly represented best by candidate  $e_i$  are  $f_i, v_i^1, v_i^2, v_i^3$ . More specifically, for every  $e_i$  and  $x \in \{1, 2, 3\}$ ,  $r(v_i^x, e_i) < r(y, e_i)$  for every  $y \in V \setminus (\{f_i\} \cup \{v_i^x\})$  (**Observation 1**). To see the correctness, observe that for every  $1 \leq z < i$  one has  $r(v_z^x, e_i) = 2n^2 + 1$ . Moreover, for every  $i < z \leq n$ ,  $r(v_z^x, e_i) = i + z - 1 > 2i - 1 = r(v_i^x, e_i)$ . Finally, for every  $f_j, j \neq i$  one also has misrepresentation value  $r(f_j, e_i) = 2n^2 + 1$ .

Now, we show the following.

*Claim:* There is an exact 3-cover for  $(\mathcal{S}, E)$  if and only if there is a set of  $k = n + n/3$  candidates that can represent all voters with total misrepresentation  $R = 2n^2$  such that exactly three voters are assigned to one candidate.

“ $\Leftarrow$ ” Given an exact 3-cover  $\mathcal{S}' \subseteq \mathcal{S}$ , we show that the candidate set  $\{s_j \mid S_j \in \mathcal{S}'\} \cup E$  is a winner set as required by the claim. The corresponding mapping is as follows.

- For every  $1 \leq i \leq n$ , the voter  $f_i$  is assigned to the candidate  $e_i$ .
- For  $1 \leq i \leq n$  and  $x \in \{1, 2, 3\}$ , if  $e_i$  occurs for the  $x$ th time in  $S_j$  for a  $S_j \in \mathcal{S}'$ , then assign  $v_i^x$  to  $s_j$ , else assign  $v_i^x$  to  $e_i$ .

Since in the exact 3-cover every element is covered exactly ones, it follows that every voter is assigned to exactly one candidate and every winner candidate “represents” three voters. More specifically, for the three voters  $v_i^1, v_i^2$ , and  $v_i^3$  corresponding to the three occurrences of the element  $e_i$ , one of them is represented by the candidate corresponding to the solution set in which  $e_i$  occurs and the two other voters by the candidate  $e_i$  (the third candidate represented by  $e_i$  is  $f_i$ ). It remains to compute the total misrepresentation of this solution. Due to definition, every candidate  $s_j$  represents all three voters with misrepresentation zero. Moreover, every candidate  $e_i$  represents  $f_i$  with misrepresentation zero and two voters from  $\{v_i^1, v_i^2, v_i^3\}$  with misrepresentation  $r(v_i^x, e_i) = i + i - 1 = 2i - 1$  for  $x \in \{1, 2, 3\}$ . Hence, the total misrepresentation is

$$\sum_{i=1}^n 2(2i - 1) = 2n(n + 1) - 2n = 2n^2. \quad (3)$$

“ $\Rightarrow$ ” Consider a size- $k$  set  $C' \subseteq C$  of winners that represent all voters with total misrepresentation  $R = 2n^2$ . Since for every voter  $f_i$ , the only candidate that can represent  $f_i$  with misrepresentation at most  $R$  is  $e_i$ , it follows that  $E \subseteq C'$ . Recall that due to the M-criterion, every candidate must represent exactly three voters. Thus, every candidate  $e_i \in E$  must represent two further voters (besides  $f_i$ ). Clearly, a lower bound for the total misrepresentation is hence given by assigning to every  $e_i \in E$  two further voters which can be represented as least as good as all other voters by  $e_i$ . Due to Observation 1, these two voters must be from  $\{v_i^1, v_i^2, v_i^3\}$ . Moreover, according to Equation 3 the corresponding lower bound for the total misrepresentation matches the total misrepresentation  $R = 2n^2$ . This implies that  $e_i$  is assigned to exactly two voters from  $\{v_i^1, v_i^2, v_i^3\}$ . Finally, for every  $1 \leq i \leq n$ , there remains one voter  $v_i^x$  that must be represented by a candidate from  $C' \setminus E$ . Since  $|C' \setminus E| = n/3$  and a candidate  $s_j$  can only represent a voter  $v_i^x$  by misrepresentation zero if the element  $e_i$  occurs in  $S_j$ , the sets corresponding to the candidates in  $C' \setminus E$  must form an exact 3-cover.  $\square$

## 6 Conclusion and Outlook

We start with summarizing the relevance of the results of this work. This will be followed by a discussion of closely related problems and models that might be investigated in future research. We conclude with several questions that directly follow from our results.

### 6.1 Relevance of results

The computation of a set of candidates that “fully proportionally” represent the society has applications in many relevant settings. The main problem with the suggested approaches in the extant literature is that the corresponding combinatorial problems are NP-hard, that is, they cannot be solved efficiently in general. This raises the question whether these approaches despite the theoretically proven advantages (see, e.g., a detailed discussion of those in [8]) are useless in practice.

One approach is of course to try to escape high complexity by modifying the concept while keeping it still meaningful. In this regard we tried to change the way the total misrepresentation is calculated taking the minimax (or Rawlsian) approach. This appeared not to help in the general case—all problems remain computationally hard—however, it partially helped for single-peaked elections: while the classical Monroe scheme remains NP-hard, its minimax version can be solved in polynomial time.

In general, there are several ways to deal with NP-hard problems. For example, NP-hardness is based on the worst-case analysis and hence one might be able to develop algorithms that work efficiently for most instances. However, although unlikely, it still might happen that the outcome of an election leads to a hard instance. Then, this would lead to the situation of political impasse with unpredictable consequences.

Another common approach to tackle NP-hard problems is to invoke approximation algorithms. While for some scenarios a nearly optimal solution might be sufficient, for other scenarios, like political elections, it seems unlikely that the voters will accept such a solution.

Based on the previous discussion, it seems clearly desirable to identify well-specified settings for which an optimal solution can be computed efficiently. This will extend the applicability of the fully proportional representation approach to such settings. In this regard, we conducted an investigation in two different directions. The first is the parameterized complexity analysis, and, the second, is the special case of single-peaked elections.

Regarding the parameterized complexity of the four studied problems, most of our results are negative (see Table 1, page 7). In particular, for the natural and well-motivated parameter number of winners, the corresponding problems turned out to be W[2]-complete. However, if in addition, there is a winner set that can represent all voters with a small total misrepresentation, three of the problems become tractable for the Borda misrepresentation function. Moreover, the fixed-parameter tractability results with respect to the number of voters and candidates, respectively, are useful for such restricted settings.

Regarding single-peaked elections, almost all of our results are positive and come with polynomial-time algorithms (see Table 2, page 8). A possible critique of this approach is to claim that single-peakedness is in a way an idealized model which is not robust enough. A smallest honest mistake of a voter in filling her ballot may result in election becoming not single-peaked. Also there may be a secondary issue in the election that is also important for some voters which may lead to the election being “almost” single-peaked but not exactly single-peaked. In this regard it would be interesting to investigate how difficult is to find a single-peaked profile “closest” to the given one. For this one

might employ techniques of the so-called distance rationalizability approach [1, 42, 20, 21]. Since our algorithms show polynomial-time solvability for the important basic case of single-peakedness, they might be a basis for developing efficient algorithms for such extended settings.

Summarizing, our work contributes to the important topic of making fully proportional representation ideas practical and complements the analysis of this method by Brams [8], who mainly concentrated on integer linear programming approach, as well as former computational complexity results for the considered problems [49, 40].

## 6.2 Related problems and scenarios

Before ending the work with some concrete open questions, we, first, describe some relations of the considered problems to facility location problems and, second, describe a reasonable alternative multi-winner model. Both topics might also lead to interesting questions for future research.

**Relations to facility location.** A basic scenario for this problem is that a company needs to choose a set of facility locations to serve a set of customers with as little cost to them as possible. Fellows and Fernau [29] investigated the parameterized complexity of a variant of this problem that is closely related to CC-MULTIWINNER. Basically, the facility locations can be considered as the set of candidates, the customers as the multiset of voters and the goal is to find a set of facility locations to serve the customers. The only difference is the cost function: In addition to a term that resembles the misrepresentation for every voter (customer), every facility location comes with a certain cost that is required to install the facility.

Similar to our study, Fellows and Fernau [29] studied the parameter number  $k$  of winners/selected facilities locations and the total cost. For the parameter  $k$ , W[2]-hardness for CC-MULTIWINNER follows from the reduction given for the facility location problem. Regarding the parameter “total cost”, the results of the two papers are not directly comparable. This is due to the fact that the facility location problem in [29] comes with a minimum cost of one for serving a customer even at the “best” facility location (which would be an analogue of the condition  $r(v, c) \geq 1$  for the misrepresentation function  $r$ ). In this case, the considered problem is fixed-parameter tractable with respect to the total cost. This might not come as a surprise since here the total cost/misrepresentation is at least the number of voters and fixed-parameter tractability with respect this parameter holds for all four considered voting problem (Proposition 2). In contrast, using the condition  $r(v, c) \geq 0$  all considered problems are at least W[2]-hard with respect to the total misrepresentation/cost (see Table 1).

The close connection between facility location and multi-winner problems clearly seems to deserve more attention in future work. We remark that the considered variants of this work might also make sense in the context of the facility location problem. For example, the Monroe model might apply for sets of facilities such that every facility can serve about the same number of customers. Moreover, the single-peaked scenario translates, for example, to the setting that all potential facility locations are along one main street and each resident ranks the cost of using the facility according to the distance from that facility to the place of his residence.

**Multiset of candidates model.** There may be a compromise solution between the two systems of Chamberlin & Courant and Monroe. We may still divide voters into equal or almost equal groups but we may assign the same representative to more than one group of voters. Say, if there are  $n$  voters and  $k$  representatives are to be elected we may split voters into groups of sizes  $\lfloor n/k \rfloor$  and

$\lfloor n/k \rfloor + 1$  but allow the same candidate to represent more than one group. Mathematically this would result in selecting not a set of representatives of cardinality  $k$  but a multiset of the same cardinality. The classic Monroe example [43] which considers subscription of newspapers for the common room is in fact a better fit for the multiset model since if demand, say for Financial Times, is strong several copies of this newspaper can be subscribed. We will still need to use weighted voting in the assembly but in this case all weights will be integers.

To illustrate the difference let us consider six people electing a representative assembly of three. Suppose our candidates must come from the set  $A = \{a, b, c, d\}$  and the preferences of voters are as follows:

4	2
a	c
b	b
c	a
d	d

A set variant of Monroe scheme will give us the set of representatives  $\{a, b, c\}$  while from the multiset point of view it is more natural to have a multiset  $\{a^2, c\}$  as the answer which could be interpreted to mean that two votes given to  $a$  and one to  $c$ . Multiset point of view seems more natural here, indeed,  $b$  does not seem to represent anybody nicely. So the misrepresentation will be nonzero in the set version and zero in the multiset one.

As far as we know the computational complexity for the computation of a winner in the multiset model is unstudied so far. On a first glance, it seems conceivable that the computational complexity for the multiset model lies between the complexity for CC-MULTIWINNER and M-MULTIWINNER. This leads to interesting questions such as whether a set of winners according to the multiset model can be computed in polynomial time when the electorate is single peaked.

### 6.3 Open questions

Several concrete questions arise from this work.

- For CC- and M-MULTIWINNER for the Borda misrepresentation function we provided algorithms showing polynomial-time solvability for a constant misrepresentation bound  $R$ . Are these problems fixed-parameter tractable with respect to  $R$ ?
- Is MINIMAX M-MULTIWINNER for the Borda misrepresentation function fixed-parameter tractable with respect to the composite parameter  $(R, k)$ ?
- For M-MULTIWINNER for single-peaked elections we have shown NP-hardness for integer-valued misrepresentation functions. Is the problem fixed-parameter tractable with respect to the number of winners  $k$  or/and with respect to the misrepresentation bound  $R$ ?
- Is M-MULTIWINNER for the Borda misrepresentation function polynomial-time solvable for single-peaked instances?
- Can the results for single-peaked elections be extended to generalized single-peakedness (e.g., as defined by Nehring and Puppe [45]) or to “almost” single-peaked profiles (in some sense)? This might be of particular interest if the problem of finding the “closest” single-peaked profile to a given one would turn out to be polynomial-time solvable (for some distance on the set of profiles).

## References

- [1] N. Baigent. Metric rationalisation of social choice functions according to principles of social choice. *Mathematical Social Sciences*, 13(1):59–65, 1987. 33
- [2] J. J. Bartholdi III, C. A. Tovey, and M. A. Trick. The computational difficulty of manipulating an election. *Social Choice and Welfare*, 6:227–241, 1989. 5
- [3] N. Betzler. *A Multivariate Complexity Analysis of Voting Problems*. PhD thesis, Friedrich-Schiller Universität Jena, 2010. 6
- [4] N. Betzler, J. Guo, and R. Niedermeier. Parameterized computational complexity of Dodgson and Young elections. *Information and Computation*, 208(2):165–177, 2010. 6
- [5] N. Betzler, S. Hemmann, and R. Niedermeier. A multivariate complexity analysis of determining possible winners given incomplete votes. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence (IJCAI)*, pages 53–58, 2009. 6
- [6] D. Black. *The Theory of Committees and Elections*. Cambridge University Press, 1986. 2, 7
- [7] E. Booth and G. Lueker. Testing for the consecutive ones property, interval graphs, and graph planarity using PQ-trees algorithms. *Journal of Computer and System Sciences*, 13:335–379, 1976. 7
- [8] S. Brams. *Mathematics and Democracy*. Princeton University Press, 2008. 2, 6, 32, 33
- [9] S. Brams and P. Fishburn. Voting procedures. In K. Arrow, A. K. Sen, and K. Suzumura, editors, *Handbook of Social Choice and Welfare*, volume 1, pages 173–236. Elsevier, 2002. 2
- [10] F. Brandt, M. Brill, E. Hemaspaandra, and L. A. Hemaspaandra. Bypassing combinatorial protections: Polynomial-time algorithms for single-peaked electorates. In *Proceedings of the 24th AAAI Conference on Artificial Intelligence (AAAI)*, pages 715–722, 2010. 8
- [11] F. Brandt, M. Brill, and H. G. Seedig. On the fixed-parameter tractability of composition-consistent tournament solutions. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI)*, pages 85–90. AAAI Press, 2011. 6
- [12] J. R. Chamberlin and P. N. Courant. Representative deliberations and representative decisions: Proportional representation and the Borda rule. *American Political Science Review*, 77(3):718–733, 1983. 2, 3
- [13] R. Christian, M. R. Fellows, F. A. Rosamond, and A. Slinko. On complexity of lobbying in multiple referenda. *Review of Economic Design*, 11(3):217–224, 2007. 6
- [14] V. Conitzer. Making decisions based on the preferences of multiple agents. *Communications of the ACM*, 53(3):84–94, 2010. 5
- [15] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms*. MIT Press, 2nd edition, 2001. 10
- [16] C. Dodgson. *The Principles of Parliamentary Representation*. Harrison, London, 1884. 2

- [17] M. Dom, D. Lokshtanov, S. Saurabh, and Y. Villanger. Capacitated domination and covering: A parameterized perspective. In *Proceedings of the 3rd International Workshop on Parameterized and Exact Computation (IWPEC)*, volume 5018 of *LNCS*, pages 78–90. Springer, 2008. 19
- [18] B. Dorn and I. Schlotter. Multivariate complexity analysis of swap bribery. In *Proceedings of the 5th International Symposium on Parameterized and Exact Computation (IPEC)*, volume 6478 of *LNCS*, pages 107–122. Springer, 2010. 6
- [19] R. G. Downey and M. R. Fellows. *Parameterized Complexity*. Springer, 1999. 6, 9, 10
- [20] E. Elkind, P. Faliszewski, and A. Slinko. Cloning in elections. In *Proceedings of the 24th AAAI Conference on Artificial Intelligence (AAAI)*, pages 768–773, 2010. 33
- [21] E. Elkind, P. Faliszewski, and A. Slinko. On the role of distances in defining voting rules. In *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 375–382, 2010. 6, 33
- [22] B. Escoffier, J. Lang, and M. Öztürk. Single-peaked consistency and its complexity. In *Proceedings of the 18th European Conference on Artificial Intelligence (ECAI)*, pages 366–370. IOS Press, 2008. 7
- [23] G. Even, R. Levi, D. Rawitz, B. Schieber, S. Shahar, and M. Sviridenko. Algorithms for capacitated rectangle stabbing and lot sizing with joint set-up costs. *ACM Transactions on Algorithms*, 4(3), 2008. 8, 25, 26, 27
- [24] P. Faliszewski, E. Hemaspaandra, and L. A. Hemaspaandra. Using complexity to protest elections. *Communications of the ACM*, 53(1):74–82, 2010. 5
- [25] P. Faliszewski, E. Hemaspaandra, L. A. Hemaspaandra, and J. Rothe. Llull and Copeland voting computationally resist bribery and constructive control. *Journal of Artificial Intelligence Research*, 35:275–341, 2009. 6
- [26] P. Faliszewski, E. Hemaspaandra, L. A. Hemaspaandra, and J. Rothe. A richer understanding of the complexity of election systems. In S. Ravi and S. Shukla, editors, *Fundamental Problems in Computing: Essays in Honor of Professor Daniel J. Rosenkrantz*, chapter 14, pages 375–406. Springer, 2009. 5
- [27] P. Faliszewski and A. Procaccia. AI’s war on manipulation: Are we winning? *AI Magazine*, 31(4):53–64, 2010. 5
- [28] J. Farfel and V. Conitzer. Aggregating value ranges: Preference elicitation and truthfulness. *Journal of Autonomous Agents and Multi-Agent Systems*, 22(1):127–150, 2011. 21
- [29] M. R. Fellows and H. Fernau. Facility location problems: A parameterized view. *Discrete Applied Mathematics*, 159(11):1118–1130, 2011. 33
- [30] J. Flum and M. Grohe. *Parameterized Complexity Theory*. Springer, 2006. 6, 9, 10
- [31] M. L. Fredman and R. E. Tarjan. Fibonacci heaps and their uses in improved network optimization algorithms. *Journal of the ACM*, 34(3):596–615, 1987. 12

- [32] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, 1979. 9, 10, 30
- [33] M. R. Garey, D. S. Johnson, and L. Stockmeyer. Some simplified NP-complete problems. In *Proceedings of the sixth annual ACM symposium on Theory of computing (STOC)*, pages 47–63. ACM, 1974. 16, 17
- [34] T. F. Gonzalez. Clustering to minimize the maximum intercluster distance. *Theoretical Computer Science*, 38:293–306, 1985. 30
- [35] S. Guha, R. Hassin, S. Khuller, and E. Or. Capacitated vertex covering. *Journal of Algorithms*, 48(1):257–270, 2003. 19
- [36] J. Guo, R. Niedermeier, and S. Wernicke. Parameterized complexity of Vertex Cover variants. *Theory of Computing Systems*, 41:501–520, 2007. 19
- [37] J. Harsanyi. Can the maximin principle serve as a basis for morality? a critique of John Rawls’s theory. *American Political Science Review*, 69(2):594–606, 1995. 3
- [38] R. Impagliazzo, R. Paturi, and F. Zane. Which problems have strongly exponential complexity? *Journal of Computer and System Sciences*, 63(4):512–530, 2001. 9
- [39] J. Levin and B. Nalebuff. An introduction to vote-counting schemes. *Journal of Economic Perspectives*, 9(1):3–26, 1995. 2, 3
- [40] T. Lu and C. Boutilier. Budgeted social choice: From consensus to personalized decision making. In *Proceedings of the Twenty-second International Joint Conference on Artificial Intelligence (IJCAI-11)*, 2011. To appear. Also presented at COMSOC-10. 2, 4, 6, 11, 14, 33
- [41] R. Meir, A. D. Procaccia, J. S. Rosenschein, and A. Zohar. The complexity of strategic behavior in multi-winner elections. *Journal of Artificial Intelligence Research*, 33:149–178, 2008. 6
- [42] T. Meskanen and H. Nurmi. Closeness counts in social choice. In M. Braham and F. Steffen, editors, *Power, Freedom, and Voting*, pages 289–306. Springer, 2008. 33
- [43] B. L. Monroe. Fully proportional representation. *American Political Science Review*, 89(4):925–940, 1995. 2, 3, 34
- [44] H. Moulin. *Axioms of Cooperative Decision Making*. Cambridge University Press, 1991. 7
- [45] K. Nehring and C. Puppe. The structure of strategy-proof social choice – Part I: General characterization and possibility results on median spaces. *Journal of Economic Theory*, 135(1):269–305, 2007. 34
- [46] R. Niedermeier. *Invitation to Fixed-Parameter Algorithms*. Oxford University Press, 2006. 6, 9, 10
- [47] R. F. Potthof and S. J. Brams. Proportional representation: Broadening the options. *Journal of Theoretical Politics*, 10(2):147–178, 1998. 6
- [48] A. D. Procaccia, J. S. Rosenschein, and A. Zohar. Multi-winner elections: Complexity of manipulation, control and winner-determination. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1476–1481, 2007. 13

- [49] A. D. Procaccia, J. S. Rosenschein, and A. Zohar. On the complexity of achieving proportional representation. *Social Choice and Welfare*, 30:353–362, 2008. 4, 6, 13, 33
- [50] J. Rawls. *A Theory of Justice*. Cambridge: Harvard University Press, 1971, Revised Edition (Cambridge: Harvard University Press), 1999. 3
- [51] N. Tideman. *Collective Decisions and Voting*. Ashgate Publishing Ltd., 2006. 7
- [52] N. Tideman and R. Richardson. Better voting methods through technology: The refinement-manageability trade-off in the single transferable vote. *Public Choice*, 103(1-2):13–34, 2000. 3