When Votes Change and Committees Should (Not)

Robert Bredereck[*1,2] , **Till Fluschnik**[3] and **Andrzej Kaczmarczyk**[3,4]
[1] Institut für Informatik, TU Clausthal, Germany
[2] Humboldt-Universität zu Berlin, Berlin, Germany
[3] Technische Universität Berlin, Faculty IV, Algorithmics and Computational Complexity, Germany
[4] AGH University, Kraków, Poland
robert.bredereck@tu-clausthal.de, till.fluschnik@tu-berlin.de, andrzej.kaczmarczyk@agh.edu.pl

Abstract

Electing a single committee of a small size is a classical and well-understood voting situation. Being interested in a sequence of committees, we introduce two time-dependent multistage models based on simple scoring-based voting. Therein, we are given a sequence of voting profiles (stages) over the same set of agents and candidates, and our task is to find a small committee for each stage of high score. In the *conservative model* we additionally require that any two consecutive committees have a small symmetric difference. Analogously, in the *revolutionary model* we require large symmetric differences. We prove both models to be NP-hard even for a constant number of agents, and, based on this, initiate a parameterized complexity analysis for the most natural parameters and combinations thereof. Among other results, we prove both models to be in XP yet W[1]-hard regarding the number of stages, and that being revolutionary seems to be "easier" than being conservative.

1 Introduction

In well-studied classical committee election scenarios, given a set of candidates, we aim at selecting a small committee that is, in a certain sense, most suitable for a given collection of preferences over the candidates [Brandt *et al.*, 2016; Faliszewski *et al.*, 2017; Rothe, 2016]. However, typically these scenarios concentrate solely on electing a committee in a single election to the neglect of a time dimension. This neglect results in serious limitations of the model. For instance, it is not possible to ensure a relationship (e.g., a small number of changes) between any two consecutive committees. We tackle this issue by introducing a *multistage* [Gupta *et al.*, 2014] variant of the problem. In this variant, a *sequence* of voting profiles *is given*, and we seek a *sequence* of small committees, each collecting a reasonable number of approvals, such that the *difference between consecutive committees* is upper-bounded.

For instance, assume a research community to seek organizers of a series of events (say those scheduled for next year). The organizers must be fixed in advance to allow preparation time. Since events may differ in location and type, not every candidate fits equally well to every event. Thus, each member (agent) of the community is asked to name one suitable organizer for each event and, based on this, the goal is to determine a sequence of organizing committees (one for each event). Naturally, there are three constraints: (i) each committee is bounded in size, (ii) each committee has enough support from the agents, and (iii) at least a certain number of candidates in consecutive committees overlap to avoid a lack of knowledge transfer jeopardizing effectiveness.

Initiating a study of so far overlooked multistage variant of multiwinner elections, we aim at understanding the computational complexity of the related computational problems. In particular we want to detect computationally tractable cases. Notably, our multistage setting introduces two new dimensions to the standard model of multiwinner elections: the relation between consecutive committees and the time. Thus, our second goal is to observe how these two dimensions affect the computational complexity of the introduced model.

1.1 Model and Examples

We denote by $\mathbb{N}$ and $\mathbb{N}_0$ the natural numbers excluding and including zero, respectively. For a function $f\colon A \to B$, let $f^{-1}(B') = \{a \in A \mid f(a) \in B'\}$ for every $B' \subseteq B$. We use basic notation from graph theory [Diestel, 2010] and parameterized algorithmics [Cygan *et al.*, 2015]. The main problem of this work is as follows.

MULTISTAGE SNTV (MSNTV)
Input: A set of agents $A = \{a_1, \ldots, a_n\}$, a set of candidates $C = \{c_1, \ldots, c_m\}$, a sequence $U = (u_1, \ldots, u_\tau)$ of τ voting profiles with $u_t\colon A \to C \cup \{\emptyset\}$, $t \in \{1, \ldots, \tau\}$, and three integers $k \in \mathbb{N}$, $\ell \in \mathbb{N}_0$, and $x \in \mathbb{N}$.
Question: Is there a sequence $(C_1, \ldots, C_\tau)$ of committees $C_t \subseteq C$ such that for all $t \in \{1, \ldots, \tau\}$ it holds true that $|C_t| \leq k$ and $\text{score}_t(C_t) := |u_t^{-1}(C_t)| \geq x$, and

$$|C_t \triangle C_{t+1}| \leq \ell \qquad (1)$$

holds true for all $t \in \{1, \ldots, \tau - 1\}$?

One may wonder why we chose an upper bound on k in the problem definition instead of specifying an exact constant committee size. While most natural instances will have solutions with committees of size exactly k, requiring them rules

[*]Contact Author

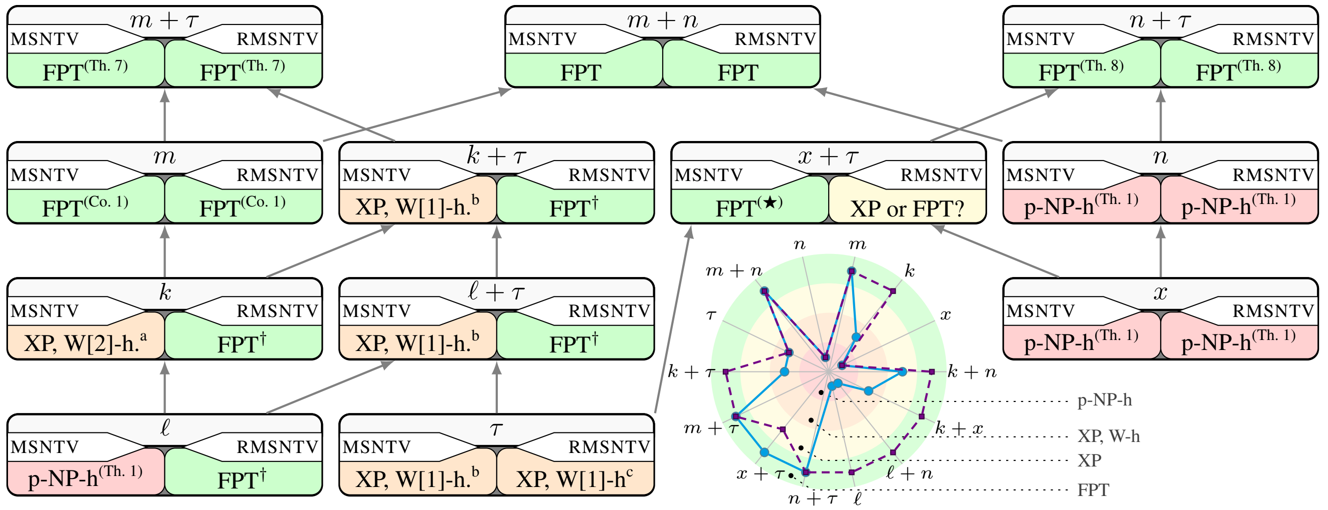

Figure 1: Overview of results for MSNTV and RMSNTV. Abbreviations p-NP-h and W[1]-h stand for, respectively, para-NP-hard and W[1]-hard. An arrow from one parameter p to another parameter p' indicates that p can be upper bounded by some function in p' (e.g., $\ell \leq 2k$, $k \leq m$, or $x \leq n$). The spiderweb diagram depicts further results being not displayed for readability (solid: conservative; dashed: revolutionary). [a] (Th. 2 & 4) [b] (Th. 2 & 5) [c] (Th. 3 & 5) [†] [Kellerhals *et al.*, 2021]

out odd symmetric differences, e.g., $\ell = 1$. All but one of our results easily translate to the setting with size k committees in each stage so that this decision is mainly a technical simplification. See a full version of the paper [Bredereck *et al.*, 2020a] for a detailed discussion.

SNTV comes from "single non-transferable vote", to which our model boils down for a single stage. While most of our results transfer to the setting of general approval profiles (see Section 5), we use "Plurality" profiles, where each agent approves exactly one candidate, for four reasons. (I) Aiming for positive algorithmic results and for recognizing the influence of the basic model properties, it is most natural to start with the simplest relevant scenario. Even though SNTV is simple, (II) it is the only committee scoring rule serving finding representation- *and* excellence-focused committees [Faliszewski *et al.*, 2019]. Hence, it forms a good basis for a further exploration of our model for another rules reaching these two goals. (III) Plurality profiles are widely accepted in practice and form complex voting procedures (e.g., STV, the two-vote or two-stage voting systems used for the German or French parliament). (IV) Selecting a single candidate is only a weak (cognitive) barrier for human agents increasing the applicability of the model. In fact, our definition can be easily extended to more expressive scoring-based voting profiles.

Motivated by respective requirements in many applications, Boehmer and Niedermeier [2021] propose a systematic study of (multiwinner) voting models that handle multiple preference profiles at once (e.g., when incorporating changes over time). Following their work, our paper provides one of the first models opening the field for new potential applications. Indeed, MSNTV models various possible practical scenarios, two of which we briefly sketch below.

Buffet Selection. Suppose we are asked to organize the venue's breakfast buffet of a multiday event (like a workshop seminar). We offer different disjoint food bundles (candidates) for breakfast and ask the participants of the event to share their preferences of which bundle is their favorite for which day. Due to limited space, we can offer only at most some number of bundles in the buffet (committee) simultaneously. Moreover, to stay at low cost and to avoid food waste, we want that few bundles change from one day to the next. Clearly, given these constraints and the collected preferences (voting profiles), we want at all days to have a high number of participants whose voted bundle made it into the buffet.

Exhibition Composition. When planning a multiday exhibition of sculptures (candidates) in a lobby of a hotel where we are enabled neither to show at once all the sculptures that we want to exhibit, nor to exchange arbitrarily many sculptures between consecutive exhibitions days (due to, e.g., limited capacity of transporting sculptures between some depot and the hotel). To nevertheless offer an enjoyable experience for numerous visitors, we ask the visitors to vote for each day they plan to visit for their favorite sculpture to be exhibited (to keep the poll simple and robust).

Note that if we drop condition (1) (or, equivalently, set $\ell = 2k$), then on the one hand, we have no control over changes between consecutive committees, yet on the other hand, we obtain a linear-time solvable problem. Thus, control comes with a computational cost, bound in the value of ℓ. To proceed with our second goal—in particular, better understanding of condition (1), we additionally study a problem variant of MSNTV. We obtain this variant, referred to as REVOLUTIONARY MULTISTAGE SNTV (RMSNTV), by replacing (1) by $|C_t \triangle C_{t+1}| \geq \ell$. In words, in RMSNTV we request a change of size at least ℓ between consecutive committees. By this, while we complemented the meaning of ℓ, it still expresses a control over the changes, and hence comes possibly again with a computational cost. We investigate whether the "*conservative*" (MSNTV) and the revolutionary (RMSNTV) variant differ, and if so, then how and why.

1.2 State of the Art and Our Contributions

Our model follows the recently proposed *multistage* model [Eisenstat *et al.*, 2014; Gupta *et al.*, 2014], that led to several multistage problems [Bampis *et al.*, 2018a; Bampis *et al.*, 2019b; Fluschnik *et al.*, 2019; Fluschnik *et al.*, 2020; Chimani *et al.*, 2021; Heeger *et al.*, 2021; Bampis *et al.*, 2019a; Bampis *et al.*, 2018b; Kellerhals *et al.*, 2021; Fluschnik, 2021] next to ours. Graph problems considered in the multistage model often study classic problems on temporal graphs (a sequence of graphs over the same vertices). While all the multistage problems known from literature cover the variant we call "conservative," our revolutionary variant forms a novel submodel herein.

Although, to the best of our knowledge our model is novel, other aspects of selecting multiple (sub)committees have been studied in (computational) social choice theory. The closest is a recent work of Bredereck *et al.* [2020b], who augment classic multiwinner elections with a time dimension. Accordingly, they consider selecting a sequence of committees. However, the major differences with our work are, first, that they do not allow agents (voters) to change their ballots over time and, second, that there are no explicit constraints on the differences between two successive committees. Freeman *et al.* [2017], Lackner [2020], and Parkes and Procaccia [2013] allow this but they consider an online scenario (in contrary to our problem that is offline). Finally, Aziz and Lee [2018] study a so-called subcommittee voting, where a final committee is a collection of several subcommittees. Their model, however, does not take time into account and requires that all subcommittees are mutually disjoint.

Our Contributions. We present the first work in the multistage model that studies a problem from computational social choice and that compares the two cases that we call conservative and revolutionary. We prove MSNTV and RMSNTV to be NP-complete, even for two agents. We present a full parameterized complexity analysis of the two problems (see Figure 1 for an overview of our results; refer to a full version [Bredereck *et al.*, 2020a] for a more detailed overview). Herein, the central tractability concept is fixed-parameter tractability: Given some parameter ρ, a problem is called fixed-parameter tractable (FPT) when it can be solved in $f(\rho)|I|^c$ time, where f is a computable function only depending on ρ, $|I|$ is the instance size, and c is some constant. Less positively, we can sometimes only show XP-membership parameterized by ρ, which means that the problem can be solved in polynomial time when ρ is constant (but the degree of the polynomial may depend on ρ). Fixed-parameter tractability can often be excluded (under standard parameterized complexity assumptions) showing W[1]- or W[2]-hardness (using parameterized reductions which are similar to standard polynomial-time many-one reductions). MSNTV and RMSNTV are almost indistinguishable regarding their parameterized complexity, but when parameterized by ℓ, MSNTV is NP-hard and RMSNTV is contained in XP. Moreover, both problems are contained in XP and W[1]-hard regarding the parameter number τ of stages; Note that for many natural multistage problems (and even temporal graph problems), such a classification is unknown—τ usually leads to para-NP-hardness. Our results further indicate that efficient data reductions (see Section 4 for the definition) in terms of polynomial-size problem kernelizations require a combination with τ: While combining the number of agents with the number of candidates allows for no polynomial-size problem kernel (unless $\mathrm{NP} \subseteq \mathrm{coNP}/\mathrm{poly}$), combining any of the two with τ yields kernels of polynomial size.

Due to the space constraints, many details, marked by ★, can be found in a full version [Bredereck *et al.*, 2020a]

2 Limits of Efficient Computation

To prepare the ground for our algorithmic investigations of the introduced problems, we first settle the computational complexity lower bounds of MSNTV and RMSNTV. We begin with NP-hardness for quite restricted cases.

Theorem 1 (★)**.** *(i)* MSNTV *is* NP*-hard even for two agents, $\ell = 0$, $x = 1$, and $k = |C|/2$. (ii)* RMSNTV *is* NP*-hard even for two agents, $\ell = 2k$, $x = 1$, and $k = |C|/2$.*

Herein, Theorem 1(ii) follows from Theorem 1(i) due to the following result (which we also use later in this section).

Lemma 1 (★)**.** *There is an algorithm that, on every instance (A, C, U, k, ℓ, x) with $\ell = 0$ and $k = |C|/2$ of* MSNTV*, computes an equivalent instance $(A, C', U', k', \ell', x)$ of* RMSNTV *with $k' = |C'|/2$, $\ell' = 2k'$, and $|U'| = 2|U| + 1$ in polynomial time.*

We point out that $\ell = 0$ (MSNTV) and $\ell = 2k$ (RMSNTV) are not the only intractable cases (★).

Theorem 1 shows that MSNTV and RMSNTV remain NP-hard even for very specific scenarios. However, the restrictions in Theorem 1 do not deal with the size k of committee and the number τ of stages. This gives hope that instances in which these numbers are small could be solved more effectively. However, as we show in the remainder of this section, regarding parameters k and τ (and their combination) for MSNTV and parameter τ for RMSNTV we (presumably) cannot obtain running times for which the exponential blow-up is only depending on values of the parameters.

Theorem 2 (★)**.** MSNTV *is (i)* W[1]*-hard when parameterized by $k + \tau$, even if $\ell = 0$; (ii)* W[2]*-hard when parameterized by k, even if $x = 1$ and $\ell = 0$.*

For Theorem 2(ii), we reduce from the W[2]-complete DOMINATING SET problem (★).In the reduction behind the proof of Theorem 2(i), we employ *Sidon sets* defined subsequently. A Sidon set is a set $S = \{s_1, s_2, \ldots, s_b\}$ of b natural numbers such that every pairwise sum of the elements in S is different. Sidon sets can be computed efficiently.

Lemma 2. *A Sidon set of size b can be computed in $\mathcal{O}(b)$ time if b is encoded in unary.*

Proof. Suppose we aim at obtaining a Sidon set $S = \{s_1, \ldots, s_b\}$. For every $i \in \{1, \ldots, b\}$, we compute $s_i := 2\hat{b}i + (i^2 \bmod \hat{b})$, where $\hat{b}$ is the smallest prime number greater than b [Erdös and Turán, 1941]. Thus, given $\hat{b}$, one can compute S in linear time.

It remains to show how to find $\hat{b}$ in linear time. Due to the Bertrand-Chebyshev [Tchebichef, 1852] theorem, we have that $\hat{b} < 2b$. Searching all prime numbers smaller than $2b$ is doable in $\mathcal{O}(b)$ time (see, for example, an intuitive algorithm by Gries and Misra [1978]). □

Proof of Theorem 2(i). We reduce from MULTICOLORED CLIQUE that is W[1]-complete when parameterized by the solution size [Pietrzak, 2003; Fellows *et al.*, 2009]. An instance $\hat{I}$ of MULTICOLORED CLIQUE consists of a q-partite graph $G = (V_1 \uplus V_2 \uplus \cdots \uplus V_q, E)$ and the task is to decide whether there is a set K of q pairwise connected vertices, each from a distinct part. For brevity, for some $i, j \in \{1, \dots, q\}$, $i < j$, let E_i^j be a set of edges connecting vertices from parts V_i and V_j; thus, $E = \bigcup_{i,j \in \{1,\dots,q\}, i<j} E_i^j$.

Construction. In the corresponding instance I of MSNTV, we let all vertices and edges in G be candidates. Then, we define three gadgets (see Figure 2 for an illustration): the *vertex selection* gadget, the *edge selection* gadget, and the *coherence* gadget. Further, we show how to use the gadgets to construct I. Instance I will be constructed in a way that its solution is a single committee of size exactly $q + \binom{q}{2}$ corresponding to vertices and edges of a clique witnessing a `yes`-instance of $\hat{I}$ (if one exists). To define the gadgets, we use a value x that we explicitly define at the end of the construction.

Vertex selection gadget. Fix some part V_i, $i \in \{1, \dots, q\}$. The vertex selection gadget for i ensures that exactly one vertex from V_i is selected. We construct the gadget by forming a preference profile $p(V_i)$ consisting of $x \cdot |V_i|$ agents such that each vertex $v \in V_i$ is approved by exactly x agents.

Edge selection gadget. For each two parts V_i and V_j such that $i < j$, we construct the edge selection gadget that allows to select exactly one edge from E_i^j. Accordingly, we build a preference profile $p(E_i^j)$ consisting of $x \cdot |E_i^j|$ agents. Again, each edge in E_i^j is approved by exactly x agents.

Coherence gadget. For the construction of the coherence gadget, let $h := |\bigcup_{i \in \{1,\dots,q\}} V_i|$ and let $S = \{s_1, \dots, s_h\}$ be a Sidon set computed according to Lemma 2. We define a bijective function $\mathrm{id}\colon \bigcup_{i \in \{1,\dots,q\}} V_i \to S$ associating each vertex of G with its (unique) *id*. Now, the construction of the coherence gadget for some pair $\{V_i, V_j\}$ of parts such that $i < j$ goes as follows. We introduce two preference profiles $p((i,j))$ and $p'((i,j))$. In preference profile $p((i,j))$, (i) each candidate $v \in V_i \cup V_j$ is approved by exactly $\mathrm{id}(v)$ agents and (ii) each edge $e = \{v, v'\} \in E_i^j$ is approved by exactly $(x - \mathrm{id}(v) - \mathrm{id}(v'))$ agents. In preference profile $p'((i,j))$, (i) each candidate $v \in V_i \cup V_j$ is approved by exactly $\frac{x}{2} - \mathrm{id}(v)$ agents and (ii) each edge $e = \{v, v'\} \in E_i^j$ is approved by exactly $(\mathrm{id}(v) + \mathrm{id}(v'))$ agents.

Having all the gadgets defined it remains to use them to form the agents and the preference profiles of instance I; and to define x, ℓ, and k. Since we want to have a committee consisting of q vertices and $\binom{q}{2}$ edges, we let $k := q + \binom{q}{2}$. We aim at a single committee, thus we set $\ell = 0$, which enforces that the committee must stay the same over time. Further, we set $x = 2s_h$. Finally, to form the preference profiles of I we put together, in any order, vertex selection gadgets for every part V_i, $i \in \{1, \dots, q\}$ as well as edge selection gadgets and coherence gadgets for every pair $\{V_i, V_j\}$ of parts such that $i < j$. As for the agents of I, with each gadget $\mathcal{G}$ we add a separate set of agents needed to implement $\mathcal{G}$ making sure that all other agents introduced by all other gadgets are approving no candidate in their voting profiles occurring in $\mathcal{G}$.

The running time analysis and correctness proof can be found in a full version (★). □

As for RMSNTV and the parameters k and τ, the situation differs from that for MSNTV. Namely, for the former we can only show the (parameterized) intractability with respect to τ.

Theorem 3 (★). RMSNTV *parameterized by* τ *is* W[1]-*hard.*

Altogether, Theorems 1 to 3 mark clear borders of computational tractability, allowing us to refine our search for efficient (parameterized) algorithms for the problems we introduced.

3 Polynomial for Constant Parameter Values

We start a series of algorithms in this section, with the one offering a polynomial-time running time in the case of a small target committee size k. The proof of Theorem 4 is based on computing in XP-running time an auxiliary directed graph in which we then check for the existence of an s-t path witnessing a `yes`-instance.

Theorem 4 (★). MSNTV *and* RMSNTV *both admit an* $\mathcal{O}(\tau \cdot m^{2k+1} \cdot n)$*-time algorithm and hence are contained in* XP *when parameterized by* k.

Since the committee size k must be at most m, we obtain the following fixed-parameter tractability result regarding m.

Corollary 1. MSNTV *and* RMSNTV *both are solvable in time* $2^{\mathcal{O}(m \log(m))} \cdot \tau n$.

The above two results are quite meaningful for elections with few candidates, even more if the committee to be chosen is very small. In fact, small-scale elections seem quite common in practice, which can be observed in preflib [Mattei and Walsh, 2017], an open-access collection of real-world election data. It turns out that 37% and 48% of instances stored therein feature, respectively, at most 10 and 30 candidates. Thus, we believe the above algorithm is promising in the light of real-world applications, especially as our theoretical bounds only regard the worst-case complexity.

We move on to the next result, providing an algorithm exploiting a small number τ of stages.

Theorem 5 (★). *When parameterized by* τ*,* MSNTV *and* RMSNTV *are contained in* XP.

The XP containment shown in Theorem 5 is surprising because known results for multistage or temporal (graph) problems show either NP-hardness for constant lifetime or trivial fixed-parameter tractability for this parameter. In practice, the algorithm from Theorem 5 could prove useful for short-term planning, which is inevitable for successful planning.

The last result features the difference between RMSNTV and MSNTV manifesting in the impact on the computational complexity of the difference ℓ of consecutive committees.

Theorem 6 (★). *Every instance* $I = (A, C, U, k, \ell, x)$ *of* RMSNTV *with* n *agents,* m *candidates, and* τ *voting profiles can be decided in* $\mathcal{O}(\tau \cdot m^{4\ell+1} \cdot n)$ *time.*

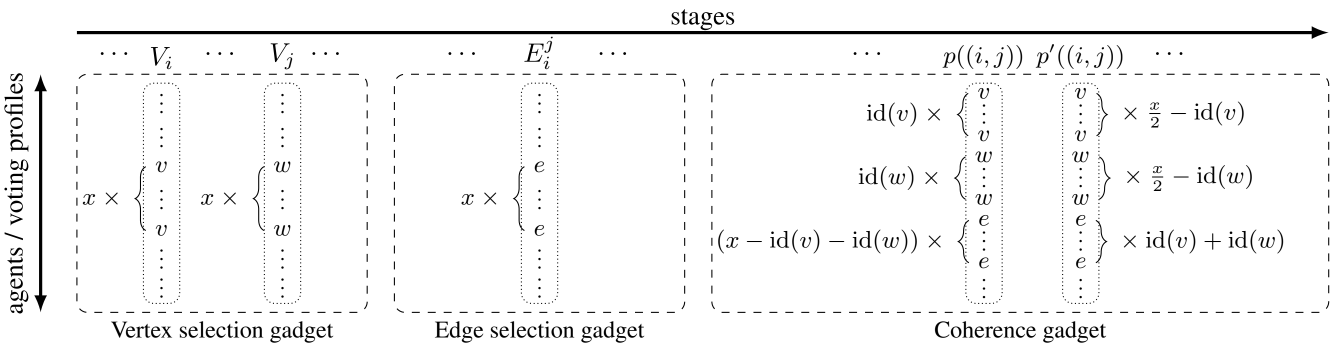

Figure 2: Illustration of the construction in the proof of Theorem 2, exemplified with edge $e = \{v, w\} \in E_i^j$ with $v \in V_i$ and $w \in V_j$. A column represents a stage (which in turn represents an vertex or edge selection gadget for some vertex or edge set, respectively, or a coherence gadget of a pair of colors) and a row represents an agent (approving either a vertex or an edge).

Inspired by a first draft of our paper, recently Kellerhals *et al.* [2021] proved that RMSNTV is solvable in $2^{\mathcal{O}(\ell)} \cdot (n + m + \tau)^{\mathcal{O}(1)}$ time—it is in FPT when parameterized only by ℓ.

4 Provably Effective Efficient Data Reduction

In this section we discuss efficient procedures that can be used to preprocess an instance in order to simplify its further processing. Preprocessing can be quite effective even for NP-hard problems [Weihe, 1998]. In terms of parameterized complexity, preprocessing with guarantee is called *problem kernelization*: For a parameterized problem L, it is a polynomial-time algorithm that maps any instance $(x, p) \in \Sigma^* \times \mathbb{N}_0$ of L to an equivalent instance (x', p') of L (a *problem kernel*) such that $|x'| + p' \leq f(p)$ for some function f only depending on the parameter p.

Preferably, we want f to be some polynomial, in which case we call the problem kernelization *polynomial*. Polynomial problem kernelizations serve as efficient and provably effective data reductions, which intuitively “cut off” “obvious” parts of an instance. After such preprocessing, a final algorithm fed with the obtained kernel can perform significantly better compared to the original instance.

We first consider a kernelization regarding the numbers m of candidates and τ of stages.

Theorem 7 (★). MSNTV *and* RMSNTV *admit problem kernels of size polynomial in* $m + \tau$.

The proof of Theorem 7 uses weighted versions of our problems (called W-MSNTV). Each weighted version takes, for each stage, a vector of size m in which each entry $i \in \{1, \ldots, m\}$ corresponds to the number of approvals that candidate i gets in a given stage (so, the number of agents upper-bounds the sum of all entries of the vector). Roughly put, in the proof of Theorem 7 one takes the original instance of MSNTV, translates it into an instance of W-MSNTV (by simply stage-wise computing the approvals of each candidate), compresses its weights (via a result by Frank and Tardos [1987]), and finally translates it back into a new instance of MSNTV, which then forms the desired kernel.

Theorem 7 has also an appealing intuitive interpretation. Namely, when there are few candidates and few stages, then the instance cannot be too large. In fact, a complementary intuition is also true: if there are few agents and few stages, then there cannot be too many meaningful candidates. Formally:

Theorem 8 (★). MSNTV *and* RMSNTV *admit problem kernels of size polynomial in* $n + \tau$.

Theorem 8 makes use of two data reduction rules which explicitly show how to prune the unnecessary candidates. To this end, recall that there are at most $n \cdot \tau$ approvals in any instance. Hence, we have the following.

Observation 1. *There are at least* $\max\{0, m - n \cdot \tau\}$ *candidates which are never approved.*

Upon Observation 1, we will next discuss deleting candidates which are never approved, in order to upper-bound the number m of candidates by some polynomial in $n + \tau$. Then, we can apply Theorem 7 to obtain the polynomial-size problem kernels. We treat MSNTV and RMSNTV separately.

For MSNTV, Observation 1 allows us to reduce any instance to an equivalent instance with $m \leq n \cdot \tau$.

Reduction Rule 1. *If* $m > n\tau$, *delete a candidate which is never approved.*

Intuitively, Reduction Rule 1 is correct because selecting a candidate which is never approved into a committee at some stage is not beneficial: it only increases the symmetric difference at the respective stage but not the committee's score.

For RMSNTV, a similar approach applies.

Reduction Rule 2. *If* $m > \max\{n, k\} \cdot \tau$, *then delete a candidate that is never approved.*

It is not so clear that deleting an unapproved candidate is safe. Indeed, this can decrease the symmetric difference between some consecutive committees in a potential solution below ℓ. Still, assuming that $m > n\tau$ (otherwise, we simply output the original instance as the kernel), one can show that there must exist a candidate y that can replace the deleted candidate $z \neq y$ in every committee that z was previously a member of. In the case of $k \leq n$, applying Reduction Rule 2 exhaustively already yields the result from Theorem 8. Yet, the case of $k > n$ and other details of the reduction rules require a more complex analysis (★).

Theorems 7 & 8 emphasize the role of a short time-horizon in our problems as they both deal with instances with few stages. In passing, we point out that the above theorems also imply that the corresponding parameterized problems are in FPT. This is due to a well-known fact that a kernelization implies fixed-parameter tractability for the same parameter.

5 Conclusion and Discussion

While our multivariate analysis revealed several intractability results, we emphasize that we also identified quite practically relevant tractable cases. In natural applications, such as electing committees serving for only few days/events, the number τ of stages is usually small. Even more, since planning too far in advance usually increases uncertainty. Furthermore, usually either the number n of voters or the number m of candidates is expected to be small (as suggested by quite a significant number of small-sized election in the preflib database). Additionally, in many elections the committee size k is not very large. Hence, our results for τ and for k (polynomial-time solvability for constant values) as well as for $\tau+n$, for $\tau+m$, or for m alone (fixed-parameter tractability) form a very positive message.

Our results also underline the importance of the time aspect for preprocessing. For both MSNTV and RMSNTV, we show that efficient data reduction to size polynomial in the number of agents and the number of candidates is unlikely [Bredereck *et al.*, 2020a]. Yet, combining any of the two parameters with the number of stages (see Theorems 7 & 8) allows for efficient (polynomial) data reduction.

Moreover, the tractability result for τ seems generally insightful for the multistage community where problems usually remain computationally hard even a for constant number of stages. Additionally, the revolutionary multistage model may be relevant on its own and may pave the way for studying more new models where consecutive changes are both lower- and upper-bounded. This is already underlined by the study of Kellerhals *et al.* [2021] on several further problems like matching or s-t path taking up our revolutionary setup.

Representation. Although it is naturally justified to start with the simplest meaningful model variant, our focus on SNTV might look restrictive. We stress that most results transfer easily to general Approval profiles (see W-MSNTV): Replace each voter approving multiple candidates by multiple voters, each approving one candidate. Also basic scoring rules can be modeled: Create for each candidate c that receives score $s(c)$ exactly $s(c)$ votes for c. The main drawback is that now the parameter number n of voters corresponds to the total number of approvals or the total score sum, respectively; yet, most positive results still hold. It remains open whether a direct modeling of more complex preferences instead of blowing up in the number of voters can avoid blowing up the running time as well. Recently, it was shown that many of our results transfer to more complex voting rules.[1]

Deeper comparison of our models. As opposed to the single-stage case, both conservative and revolutionary committee election over multiple stages are NP-complete, even for a constant number of agents. From a parameterized algorithmic point of view, computing a revolutionary committee is easier than a conservative one: When asking for committees to change for all but constantly many candidates, RMSNTV is polynomial time solvable, yet when asking for committees to change for only constant many candidates, MSNTV remains NP-hard. Finally, we wonder if RMSNTV parameterized by $k + \tau$ or $\ell + \tau$ admits a polynomial problem kernel, and whether RMSNTV parameterized by $x + \tau$ is in FPT.

Future work. Further future work may include studying approximate or randomized algorithms for MSNTV and RMSNTV as well as experimentally testing our results. Moreover, further concepts and problems (e.g., bribery and manipulation) from computational social choice may be studied in the (conservative and revolutionary) multistage model. Note that, for instance, 2-Approval (each agent approves up to two candidates) in the conservative multistage setup is already NP-hard for one agent (similar to the proof of Theorem 1(i)). As a concrete future work, we want to prove that MSNTV is W[1]-hard when parameterized by $k+n$ (inspired by a proof of Fluschnik *et al.* [2019]).

Offline versus online. Importantly, our model is applicable for offline scenarios, in which preferences are collected in advance (e.g., by social media polls, Internet profiling, customer targeting). However, online scenarios also offer an interesting research direction, yet requiring significant changes in our original models. To observe this, consider an online scenario of selecting two committees such that in the first profile all committees are scoring equally high and in the second profile there is exactly one such a committee. In the worst case, every algorithm returns a solution requiring exchanging all candidates between the two selected committees; thus, no reasonable guarantee concerning the number of changes is achievable. To avoid such trivial cases, when studying the online setting, one needs to carry out significant model modifications. One way to proceed (following the multistage literature [Bampis *et al.*, 2019a; Bampis *et al.*, 2018b; Gupta *et al.*, 2014]) is to introduce a goal function and consider the quality of the selected committees and the symmetric difference as soft constraints. Another way is to restrict the differences of consecutive profiles (analogously to Parkes and Procaccia [2013]). Such a correlation between consecutive profiles (greatly restricting the model) provides enough information to achieve some guarantees on the solution.

Acknowledgements

We thank the IJCAI'22 reviewers for their helpful comments. This work was started when all authors were with TU Berlin. TF was supported by the DFG, projects TORE (NI 369/18) and MATE (NI 369/17). AK was supported by the DFG, project AFFA (BR 5207/1 and NI 369/15), and by the European Research Council (ERC). This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No 101002854).

[1] Burak Arinalp. Multistage committee elections: Beyond plurality voting, March, 2021. Bachelor thesis. TU Berlin.

References

[Aziz and Lee, 2018] H. Aziz and B. E. Lee. Sub-committee approval voting and generalized justified representation axioms. In *Proc. of 1st AIES*, pages 3–9, 2018.

[Bampis *et al.*, 2018a] E. Bampis, B. Escoffier, M. Lampis, and V. Th. Paschos. Multistage matchings. In *Proc. of 16th SWAT*, pages 7:1–7:13, 2018.

[Bampis *et al.*, 2018b] E. Bampis, B. Escoffier, and S. Mladenovic. Fair resource allocation over time. In *Proc. of 17th AAMAS*, pages 766–773, 2018.

[Bampis *et al.*, 2019a] E. Bampis, B. Escoffier, K. Schewior, and A. Teiller. Online multistage subset maximization problems. In *Proc. of 27th ESA*, pages 11:1–11:14, 2019.

[Bampis *et al.*, 2019b] E. Bampis, B. Escoffier, and A. Teiller. Multistage knapsack. In *Proc. of 44th MFCS*, pages 22:1–22:14, 2019.

[Boehmer and Niedermeier, 2021] N. Boehmer and R. Niedermeier. Broadening the research agenda for computational social choice: Multiple preference profiles and multiple solutions. In *Proc. of 20th AAMAS*, pages 1–5, 2021.

[Brandt *et al.*, 2016] F. Brandt, V. Conitzer, U. Endriss, J. Lang, and A. D. Procaccia, editors. *Handbook of Computational Social Choice*. Cambridge University Press, 2016.

[Bredereck *et al.*, 2020a] R. Bredereck, T. Fluschnik, and A. Kaczmarczyk. Multistage committee election. *CoRR*, abs/2111.09049, 2020.

[Bredereck *et al.*, 2020b] R. Bredereck, A. Kaczmarczyk, and R. Niedermeier. Electing successive committees: Complexity and algorithms. In *Proc. of 34th AAAI*, pages 1846–1853, 2020.

[Chimani *et al.*, 2021] M. Chimani, N. Troost, and T. Wiedera. Approximating multistage matching problems. In *Proc. of 32nd IWOCA*, pages 558–570, 2021.

[Cygan *et al.*, 2015] M. Cygan, F. V Fomin, Ł. Kowalik, D. Lokshtanov, D. Marx, M. Pilipczuk, M. Pilipczuk, and S. Saurabh. *Parameterized Algorithms*. Springer, 2015.

[Diestel, 2010] R. Diestel. *Graph Theory*, volume 173 of *Graduate Texts in Mathematics*. Springer, 2010.

[Eisenstat *et al.*, 2014] D. Eisenstat, C. Mathieu, and N. Schabanel. Facility location in evolving metrics. In *Proc. of 41st ICALP*, pages 459–470, 2014.

[Erdös and Turán, 1941] P. Erdös and P. Turán. On a problem of Sidon in additive number theory, and on some related problems. *J. London Math. Soc.*, s1–16(4):212–215, 1941.

[Faliszewski *et al.*, 2017] P. Faliszewski, P. Skowron, A. Slinko, and N. Talmon. Multiwinner voting: A new challenge for social choice theory. In U. Endriss, editor, *Trends in Computational Social Choice*. AI Access, 2017.

[Faliszewski *et al.*, 2019] P. Faliszewski, P. Skowron, A. Slinko, and N. Talmon. Committee scoring rules: Axiomatic characterization and hierarchy. *ACM Trans. Econ. Comput.*, 7(1):Article 3, 1–39, 2019.

[Fellows *et al.*, 2009] M. R. Fellows, D. Hermelin, F. Rosamond, and Stéphane Vialette. On the parameterized complexity of multiple-interval graph problems. *Theor. Comput. Sci.*, 410(1):53–61, 2009.

[Fluschnik *et al.*, 2019] T. Fluschnik, R. Niedermeier, V. Rohm, and P. Zschoche. Multistage vertex cover. In *Proc. of 14th IPEC*, pages 14:1–14:14, 2019.

[Fluschnik *et al.*, 2020] T. Fluschnik, R. Niedermeier, C. Schubert, and P. Zschoche. Multistage s-t path: Confronting similarity with dissimilarity in temporal graphs. In *Proc. of 31st ISAAC*, pages 43:1–43:16, 2020.

[Fluschnik, 2021] T. Fluschnik. A multistage view on 2-satisfiability. In *Proc. of 12th CIAC*, pages 231–244, 2021.

[Frank and Tardos, 1987] A. Frank and É. Tardos. An application of simultaneous diophantine approximation in combinatorial optimization. *Combinatorica*, 7(1):49–65, 1987.

[Freeman *et al.*, 2017] R. Freeman, S. M. Zahedi, and V. Conitzer. Fair and efficient social choice in dynamic settings. In *Proc. of 26th IJCAI*, pages 4580–4587, 2017.

[Gries and Misra, 1978] D. Gries and J. Misra. A linear sieve algorithm for finding prime numbers. *Commun. ACM*, 21(12):999–1003, 1978.

[Gupta *et al.*, 2014] A. Gupta, K. Talwar, and U. Wieder. Changing bases: Multistage optimization for matroids and matchings. In *Proc. of 41st ICALP*, pages 563–575, 2014.

[Heeger *et al.*, 2021] K. Heeger, A. Himmel, F. Kammer, R. Niedermeier, M. Renken, and A. Sajenko. Multistage graph problems on a global budget. *Theor. Comput. Sci.*, 868:46–64, 2021.

[Kellerhals *et al.*, 2021] L. Kellerhals, M. Renken, and P. Zschoche. Parameterized algorithms for diverse multistage problems. In *Proc. of 29th ESA*, pages 55:1–55:17, 2021.

[Lackner, 2020] M. Lackner. Perpetual voting: Fairness in long-term decision making. In *Proc. of 34th AAAI*, pages 2103–2110, 2020.

[Mattei and Walsh, 2017] N. Mattei and T. Walsh. A preflib.org retrospective: Lessons learned and new directions. In Ulle Endriss, editor, *Trends in Computational Social Choice*. AI Access, 2017.

[Parkes and Procaccia, 2013] D. C. Parkes and A. D. Procaccia. Dynamic social choice with evolving preferences. In *Proc. of 27th AAAI*, page 767–773, 2013.

[Pietrzak, 2003] K. Pietrzak. On the parameterized complexity of the fixed alphabet shortest common supersequence and longest common subsequence problems. *J. Comput. Syst. Sci.*, 67(4):757–771, 2003.

[Rothe, 2016] J. Rothe, editor. *Economics and Computation*. Springer, 2016.

[Tchebichef, 1852] P. Tchebichef. Mémoire sur les nombres premiers. *J. Math. Pures. Appl.*, 1(17):366–390, 1852.

[Weihe, 1998] K. Weihe. Covering trains by stations or the power of data reduction. *Proc. of 1st ALEX*, pages 1–8, 1998.