Technische Universität Berlin
Electrical Engineering and Computer Science
Institute of Software Engineering and Theoretical Computer Science
Algorithmics and Computational Complexity (AKT)

Algorithmic Complexity of Bi-Criteria Multilevel Committee Election

Eva Deltl

Thesis submitted in fulfillment of the requirements for the degree
“Bachelor of Science” (B. Sc.) in the field of Computer Science

May 2022

Supervisor and first reviewer:	Dr. André Nichterlein
Second reviewer:	Prof. Dr. Markus Brill
Co-Supervisor:	Dr. Till Fluschnik

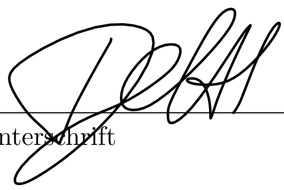
Your work must contain an affirmation in lieu of oath. Please check your study conditions for this. An example for affirmation in lieu of oath is the following.

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig und eigenhändig sowie ohne unerlaubte fremde Hilfe und ausschließlich unter Verwendung der aufgeführten Quellen und Hilfsmittel angefertigt habe.

Die selbstständige und eigenhändige Ausfertigung versichert an Eides statt

Berlin, den 04.05.2022

Datum



Unterschrift

Zusammenfassung

Wir führen das Problem **Bi-Criteria Multilevel Committee Election** (BICMCE) ein. Dabei ist das Ziel eine Folge von Komitees zu finden, die sowohl die Komiteebedingung in jeder Stufe, als auch die Agentinnenbedingung für jede der Wählerinnen erfüllt. Wir untersuchen die algorithmische Komplexität dieses Problems mit verschiedene Wahlmechanismen, hauptsächlich mit $\leq$ 1-Approval Voting ($\leq$ 1-Zustimmungswahl). Bei dieser Variante kann jede Agentin für maximal eine Kandidatin in jeder Stufe abstimmen. Das Ziel ist es, dass jede Agentin y ihrer gewählten Kandidatinnen in der Folge hat und dass jedes Komitee höchstens Größe k hat und mindestens x Agentinnen zufriedenstellt. Wir zeigen, dass $(\leq, \leq)$-BICMCE mit $\leq$ 1-Approval Voting NP-schwer ist und analysieren zudem dessen parametrisierte Komplexität. Unsere Analyse zeigt, dass es para-NP-schwer für die Mehrheit der von uns untersuchten Parameter ist. Zusätzlich zeigen wir, dass es einen polynomiellen Problemkern in der Anzahl der Stufen und der Agentinnen hat. Wir beweisen außerdem, dass unser Hauptproblem auch bei Veränderung einer Ungleichung NP-schwer bleibt, jedoch trivial wird wenn wir beide ändern. Abschließend zeigen wir, dass unser Problem auch NP-schwer für andere Abstimmungsregeln ist, sogar ohne den Aspekt mehrerer Stufen unter Nutzung von allgemeiner Zustimmungswahl oder der Borda Regel.

Abstract

We introduce and model the problem **Bi-Criteria Multilevel Committee Election** (BICMCE). Herein, the goal is to find a committee sequence over a sequence of voting profiles while satisfying the given constraints. We study the algorithmic complexity of BICMCE for different voting rules, with a focus on $\leq$ 1-Approval Voting. In this variant, every agent can approve of a maximum of one candidate in every level. While every agent should have at least y approved candidates in the sequence, every committee should have at most size k and also be approved by at least x agents. We prove $(\geq, \geq)$ BICMCE with $\leq$ 1-Approval Voting to be NP-hard and further analyze its parameterized complexity. Our analysis concludes that it is intractable for most of our parameters, with a few tractable cases, for example when parameterized by the number of candidates plus the number of levels. Additionally, we prove that our problem admits a problem kernel of size polynomial in the number of agents plus the number of levels. Further, we find out that our problem stays NP-hard when changing one inequality, while it becomes trivial when changing both. Finally, we show that our problem is NP-hard for different variants of this problem and voting rules and even intractable without the aspect of multiple levels when using (general) approval voting or Borda rule.

Contents

Chapter 1

Introduction

In the classical committee election, the goal is to elect a group of candidates considering each agent's voting profile. How these candidates are chosen is not always evident, due to the fact that how we choose a "good" committee is dependent on which voting and scoring rules we use and which factors we consider. It is really important to consider different opinions (based on ethnicities, religious beliefs, social classes, etc.) so the goal is to not only please the majority, but make sure everybody gets represented. In this paper we broaden this topic by adding a time dimension, which is an important factor in real-life scenarios, but oftentimes not acknowledged in the classical committee election.

Electing diverse committees, that are approved by the majority, is a difficult task. The main idea of our model is to reflect different opinions by making sure that each agent has a minimum of approved candidates in the elected sequence. In real-life scenarios this can improve overall satisfaction of a community, by preventing that the same committee is reelected repeatedly, even if it only satisfies a small majority. Our model can also be used in other scenarios, for example in resource allocation or scheduling.

We are given a set of agents that can vote for their candidates in τ many levels, building our τ voting profiles. In each of these levels (also understood as time steps or stages), our goal is to elect a small-size committee whose score is constrained by x. We also want every agent's score over the committee sequence to be constrained by y. The relation between x and the committee's score is denoted by $\sim_x$, while $\sim_y$ denotes the relation between the agent's score and y.

Our general model is the following.

Problem 1: $(\sim_x, \sim_y)$ Bi-Criteria Multilevel Committee Election with R

Input: A set $A = \{a_1, \dots, a_n\}$ of agents, a set C of candidates, agents preferences u_t over C following voting rule R for every $t \in \{1, \dots, \tau\}$, $\tau \in \mathbb{N}$, and integers $x, y, k \in \mathbb{N}_0$.

Question: Is there a sequence $(C_1, \dots, C_\tau)$ of candidate subsets $C_t \subseteq C$, with $|C_t| \leq k$, for all $t \in \{1, \dots, \tau\}$ such that:

$$\begin{aligned} \text{for all levels } t \in \{1, \dots, \tau\}: \quad & g_X(u_t(a_1), \dots, u_t(a_n)) \sim_x x \\ \text{and for all agents } a_i \in A: \quad & g_Y(u_1(a_i), \dots, u_\tau(a_i)) \sim_y y \end{aligned}$$

where g_X and g_Y denoting the chosen scoring functions.

We will also refer to Problem 1 in the following by $(\sim_x, \sim_y)$ BICMCE with R.

(a)

	1	2
a_1	D	R
a_2	S	M
a_3	D	H
a_4	S	T
a_5	M	M

(b)

	1	2
a_1	D	R
a_2	S	M
a_3	D	H
a_4	S	T
a_5	M	M

Figure 1.1: Example of $(\geq, \geq)$ BICMCE WITH $\leq$ 1-APPROVAL VOTING with 5 agents and 2 voting profiles.

We study the classical and parameterized complexity of $(\sim_x, \sim_y)$ BICMCE with R for different voting rules R, mainly focusing on $(\geq, \geq)$ BICMCE WITH $\leq$ 1-APPROVAL VOTING. In this variant, our goal is to elect a small-size committee in each level satisfying at least x agents. While the elected committee should be "good" in each level, we also want that the sequence of elected committees over all levels is "good" for each agent. This means that each agent has at least y of their approved candidates in the τ levels. We use plurality scoring for our x and y-scores, which corresponds to the sum of satisfied agents (for a formal description see Problem 2).

Example: Five agents want to plan a weekend trip. Each one has different plans for what they want to do on this trip. They will only join if they are happy with at least one of the chosen activities. Possible activities are: sightseeing (S), dancing (D), museum (M), theater (T), restaurant (R), and hiking (H). Their preferences change from day to day and are given in Figure 1.1 (a). We use our model under $\leq$ 1-Approval and plurality scoring. The group wants to choose two activities per day with a total score of at least 3. Possible pairs of activities are {(S), (D)} or {(M), (D)} for Day 1 and {(M), (T)} for Day 2. They have to go dancing on the first day, due to the fact that agents a_1 and a_3 have different preferences for Day 2, going to a restaurant (R) and hiking (H), which would not meet the criteria of satisfying at least three members of the group. A possible solution is shown in Figure 1.1 (b).

1.1 Motivation

Our model also finds importance outside of classical election problems, a possible application being overfishing regulation. It can be used to organize rights-based fisheries management, which after Barner et al. [Bar+15] and Grafton et al. [Gra+06] could help regulate fisheries and prevent overfishing. In this scenario, the candidates would be the fish grounds, the fishers the agents and the voting profiles would represent to which fish ground each fisher wants access to at the corresponding point in time. The y-score would help to assure fairness, by making sure that every fisher gets a minimum amount of fishing time. The x-score on the other hand allows the assignments to cover the majority of fishers in every time step, which we need to ensure in order for the model to be accepted in real-life scenarios. To regulate fisheries, the number of accessible fish grounds should be limited, which is modelled by our committee size k.

1.2 Related work

Committee or multiwinner election is not a new concept. There has been a lot of related work reviewing this problem and the difficulties surrounding determining a winning committee, for example in the work Faliszewski et al. [Fal+17]. We should also mention Aziz et al. [Azi+15], who focused their work on the computational complexity of multiwinner election and the difficulty of ensuring fairness. Participatory budgeting processes, as described by Wampler [Wam00], also center on what we want to achieve. They allow each citizen to contribute to the decisions being made, whilst simultaneously getting a majority to agree on the term.

Furthermore we are interested in the multilevel aspect of committee election. Multilayer or multistage problems have been studied in the past by Gupta, Talwar, and Wieder [GTW14], who showed the importance of adding this new dimension to problems. Specifically for multiwinner election this has been studied in the work of Bredereck, Kaczmarczyk, and Niedermeier [BKN20]. In this paper different multilayer committee election situations are studied, already showing how different voting rules may influence the complexity, which will be interesting for us later on when comparing different voting and scoring rules. Multistage committee election, as described by Bredereck, Fluschnik and Kaczmarczyk [BFK20], where multistage plurality voting is analyzed in terms of computational complexity, is very closely related to our problem. In the multistage setting, we try to find a sequence of solutions such that consecutive solutions relate in some way to each other. In our case, the order in which these committees are elected is not relevant, which is why we use the term 'level' instead of 'stage'. This difference has been addressed in the work of Boehmer and Niedermeier [BN21].

1.3 Our Contributions

We give an overview of our results in Figure 1.2 for $(\geq,\geq)$ BICMCE WITH ≤ 1-APPROVAL VOTING and $(\geq,\geq)$ BICMCE WITH $= 1$-APPROVAL VOTING. We show that both problems are NP-hard even with a constant number of levels, candidates and committee size. While $(\geq,\geq)$ BICMCE WITH ≤ 1-APPROVAL VOTING is polynomial-time solvable if $y \in \{0,\tau\}$, it becomes NP-hard if $y \in \{1,\dots,\tau-1\}$ (Theorem 3.5). For the committee score x our problem turns out to be NP-hard even if $x = 0$. Most of the results for $(\geq,\geq)$ BICMCE WITH $= 1$-APPROVAL VOTING follow from our results for $(\geq,\geq)$ BICMCE WITH ≤ 1-APPROVAL VOTING via a reduction given in Theorem 4.1. We show that $(\geq,\geq)$ BICMCE WITH ≤ 1-APPROVAL VOTING admits a problem kernel of size polynomial in $n+\tau$ and is fixed-parameter tractable when parameterized by $m+\tau$ without admitting a polynomial kernel, unless NP $\subseteq$ coNP/poly. Additionally, we show that our problem admits no $(2-\varepsilon)^\tau \cdot \text{poly}(\tau+n+m)$-time algorithm unless the SETH fails, even if there are only two candidates, and no $2^{o(n+m)} \cdot \text{poly}(\tau + n + m)$-time algorithm unless the ETH fails, even if there are only three levels (Theorems 3.7 and 3.9).

We discuss further variants of this model, by studying the problem with changed inequality which proves to be polynomial for $(\leq, \leq)$. As soon as one inequality is 'greater than' or 'equal' the problem becomes NP-hard (Theorems 3.27 and 3.28).

We also analyze our problem with general approval voting, which we find out to be NP-hard with $\tau = 1$ and W[2]-hard when parameterized by $\tau + k$ (Proposition 4.2). We approach ordinal voting by studying our problem with Borda rule, which also turns out to be NP-hard even without the aspect of multiple layers (Theorem 4.8).

1.4 Organization of this work

Chapter 2 gives an introduction and definition to the terminology used in this thesis. In Chapter 3 we give our results for $\leq$ 1-Approval. In Chapter 4, we investigate our problem using other voting rules, mainly showing that our problem is NP-hard for most voting rules, even if we only have one level. We conclude this thesis in Chapter 5 by discussing our results and bring up ideas of possible following research.

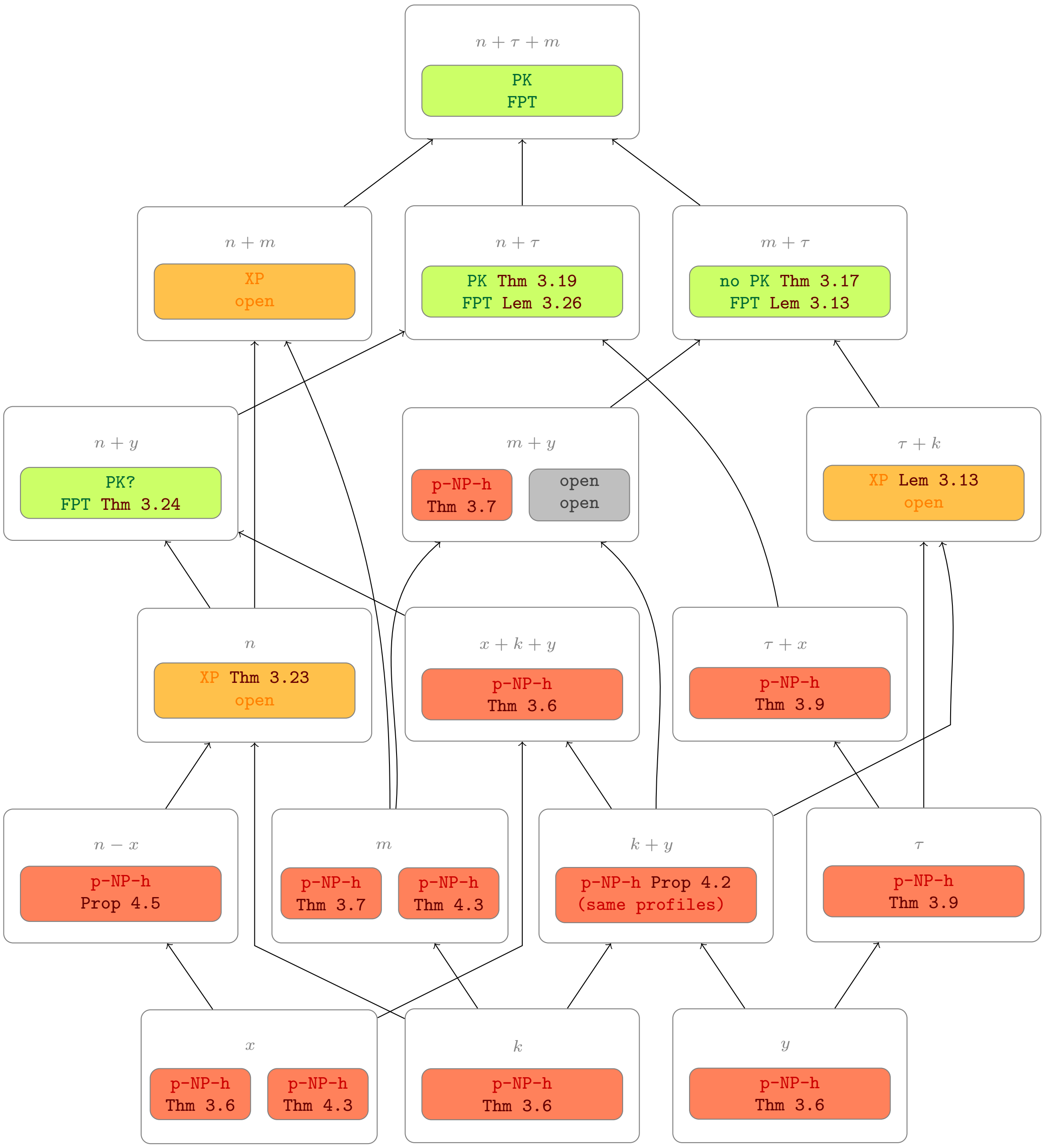


Figure 1.2: Overview of our results: PK, noPK, p-NP-h, and W[1]-h respectively abbreviate "polynomial kernel", "no polynomial kernel unless NP $\subseteq$ coNP / poly", para-NP-hard, and W[1]-hard. An arrow from one parameter p to another parameter p' indicates that p can be upper bounded by some function in p'. If their complexity differ given our results, then the result for $(\geq, \geq)$ BICMCE with $\leq$ 1-Approval Voting is depicted on the left and for $(\geq, \geq)$ BICMCE with $=$ 1-Approval Voting on the right.

Chapter 2

Preliminaries

Graph theory. Let $G = (V, E)$ denote an undirected graph, where V denotes the set of vertices and $E \subseteq \{\{v, w\} \mid v, w \in V,\ v \neq w\}$ denotes the set of edges. For a graph G, we also write $V(G)$ and $E(G)$ to denote the set of vertices and the set of edges of G, respectively.
We denote by

$V(G)$	the *vertex set* of G.
$E(G)$	the *edge set* of G with $E(G) \subseteq \binom{V(G)}{2}$; for an edge $e = \{u, v\} \in E(G)$ the two vertices u and v are called *endpoints* of e. We say that an edge is *covered* by a set X, if X contains at least one of the *endpoints* of e.
$G - V'$	the graph obtained from G by deleting the vertices $V' \subseteq V(G)$, formally, $G - V' := (V(G) \setminus V', \{\{u, v\} \in E(G) \mid u, v \in V(G) \setminus V'\})$.
$G = (V_1 \cup V_2, E)$	a *bipartite* graph: the vertices are decomposed into two disjoint sets such that no two graph vertices within the same sets are connected by an edge.

Logic. Let X denote a set of variables. A literal is a variable that is either positive or negated (we denote the negation of x by $\neg x$). A clause is a disjunction over literals. A formula φ is in conjunctive normal form (CNF) if it is of the form $\bigwedge_i K_i$, where K_i is a clause. φ is in d-CNF if every clause contains at most d literals. We say that a *truth assignment* $f : X \to \{\bot, \top\}$ satisfies φ if each clause is satisfied, which is the case if at least one literal in the clause is evaluated to true. Here $\top$ symbolizes true and $\bot$ symbolizes false.

Parameterized complexity. We use basic notations from parameterized complexity and algorithmics as described by Cygan et al. [Cyg+15], Fellows et al. [Fel+09], Niedermeier [Nie06] and Flum and Grohe [FG04].

A *parameterized* problem is a subset $L \subseteq \Sigma^* \times \mathbb{N}$, where Σ is a finite alphabet. L is *fixed-parameter tractable* if for any given instance (x, k), k denoting the parameter, it can be decided whether $(x, k) \in L$ (yes-instance) or $(x, k) \notin L$ (no-instance) in $f(k) \cdot |x|^{\mathcal{O}(1)}$ time, where f is some computable function. The complexity class of *fixed-parameter tractable* problems is called FPT. The class XP contains the parameterized problems that can be solved in $|x|^{f(k)}$ time for any given instance (x, k), where f is some computable function. A parameterized problem is *para-NP-hard* with respect to parameter k if it is NP-hard even for some constant value of k.

Further we use reduction to a *problem kernel*. Here, given any problem instance I with parameter k, one transforms it in polynomial time into a new instance I' with parameter k' such that $|I'|, k' \leq g(k)$ for some computable function g, and (I, k) is a yes-instance if and only if (I', k') is a yes-instance. If g is polynomial, we say that L admits a *polynomial kernel*.

Social Choice Theory. We adapt basic notations from computational social choice and multiwinner voting to the multistage setting. Let $A = \{a_1, \ldots, a_n\}$ denote a set of agents (also called voters) and $C = \{c_1, \ldots, c_m\}$ denote a set of candidates.

Approval Voting: An *approval ballot* $u_t(a_i)$ of an agent a_i at level t is a subset of the candidates in C. We say that agent a_i approves of candidate c at level t if $c \in u_t(a_i)$. By $u_t^{-1}(c)$ we denote the set of agents approving of candidate c in level t. A voting profile of approval ballots for each agent $a_i \in A$ at level t is denoted by $u_t : A \to P(C)$.

Ordinal Voting: A *ranked ballot* $\succ_{t,i}$ is a strict linear order of the candidates in C of an agent a_i at the level t. The voting profile of ranked ballots for each agent $a_i \in A$ at level t are denoted by $p_t = (\succ_{t,1}, \ldots, \succ_{t,n})$.

Chapter 3

Plurality Voting

To represent the political opinion of agents proportionally, most parliament elections rely on plurality voting. Although there have been a lot of discussions concerning its fairness, it is still one of the most common voting rules to this day. The idea is that every agent can elect a candidate and the elected committee contains the candidates with the most approvals. As mentioned before the idea here is to take more factors into consideration, like time and overall representation. We model this with the following:

Problem 2: $(\geq, \geq)$ BICMCE WITH ≤ 1-APPROVAL VOTING
Input: A set $A = \{a_1, \ldots, a_n\}$ of agents, a set C of candidates, voting profiles $u_t : A \to C \cup \{\emptyset\}$ for $t \in \{1, \ldots, \tau\}$, $\tau \in \mathbb{N}$, and integers $x, y, k \in \mathbb{N}_0$.
Question: Is there a sequence $(C_1, \ldots, C_\tau)$ of candidate subsets $C_t \subseteq C$, with $|C_t| \leq k$, for all $t \in \{1, \ldots, \tau\}$ such that:

$$\text{for all layers } t \in \{1, \ldots, \tau\} : \qquad \sum_{c \in C_t} |{u_t}^{-1}(c)| \geq x$$

$$\text{and for all agents } a_i \in A : \qquad \sum_{t=1}^{\tau} |u_t(a_i) \cap C_t| \geq y$$

In this chapter, we want to analyze this problem in terms of classical computational complexity, starting with basic cases and showing its NP-hardness even with constant variables. Following this, we will focus on the parameterized complexity of $(\geq, \geq)$ BICMCE WITH ≤ 1-APPROVAL VOTING and finish of with alternative models.

3.1 Basic Observations

Observation 3.1. *Given a set C of candidates and $n \cdot \tau$ voting profiles over these candidates, one can check whether there exists a solution in every level that satisfies the x-score in $\mathcal{O}((n+m) \cdot \tau)$ time.*

Proof. Find the set C' of the k most approved candidates which can be done in $\mathcal{O}(n+m)$. If $\sum_{c \in C'} |u_t^{-1}(c)| \geq x$, then C' is a possible committee with $score(C') \geq x$. If this is not the case we cannot find a committee sequence guaranteeing an x-score in every level. Repeat this in every level. Thus the solution can be computed in $\mathcal{O}((n+m) \cdot \tau)$ □

From Observation 3.1 we can follow that

Corollary 3.2. *Given an instance $I = (A, C, u, k, x, y)$ with n agents, m candidates, and $y = 0$, a solution to I can be computed in $\mathcal{O}((n + m) \cdot \tau)$ time.*

Observation 3.3. *Given an instance $I = (A, C, u, k, x, y)$ with n agents, m candidates, and $y = \tau$, a solution to I can be computed in $\mathcal{O}(n \cdot \tau)$ time.*

Proof. If there is a level $i \in \{1, \ldots, \tau\}$ with $|\bigcup_{a \in A} u_i(a)| > k$, then I is a `no`-instance, since we are not able to elect a candidate for each agent. This can be checked in $\mathcal{O}(n)$ time. After checking this in every level, which can be done in $\mathcal{O}(n \cdot \tau)$ time, we can conclude that there exists a solution to I. □

Observation 3.4. *Given an instance $I = (A, C, u, k, x, y)$ with n agents, m candidates, and $\tau = 1$, a solution to I can be computed in $\mathcal{O}(n + m)$ time.*

Proof. Since y is upper bounded by τ, when $\tau = 1$, there are two possibilities. If $y = 0$, we can compute a solution in $\mathcal{O}(n + m)$ time, as seen in Corollary 3.2. On the other hand if $y = 1$, meaning $y = \tau$, using Observation 3.3 we can find a solution in $\mathcal{O}(n)$ time. It follows that $(\geq, \geq)$ BICMCE WITH $\leq$ 1-APPROVAL VOTING with $\tau = 1$ can be solved in polynomial time. □

3.2 NP-hardness

Theorem 3.5. *$(\geq, \geq)$* BICMCE WITH $\leq$ 1-APPROVAL VOTING *is NP-hard even if $x = 0$, $y = 1$, and $\tau = 2$.*

In the following proof we give a polynomial-time many-one reduction from the following NP-hard [FN01] problem:

Problem 3: CONSTRAINT BIPARTITE VERTEX COVER
Input: A bipartite graph $G = (V = A \cup B, E)$ with n vertices and two positive integers k_A, k_B.
Question: Does there exist a set $X \subseteq V$ such that $|X \cap A| \leq k_A$, $|X \cap B| \leq k_B$ and $G - X$ contains no edge?

Proof. Let $I = (G = (V = (A \cup B), E), k_A, k_B)$ be an instance of CONSTRAINT BIPARTITE VERTEX COVER. We construct an instance $I' := (A, C, u, k, x, y)$ of $(\geq, \geq)$ BICMCE WITH $\leq$ 1-APPROVAL VOTING in polynomial time as follows.
Construction: Construct the set C of candidates by creating a candidate for every vertex in G. For the set of agents, add one agent for every edge $e \in E$. Assume without loss of generality that $k_A \leq k_B$ and let $p := k_B - k_A$. Add p agents $\{a_1^*, \ldots, a_p^*\}$ and p candidates $\{c_1^*, \ldots, c_p^*\}$. Next we build the voting profiles. For every edge $\{u, v\} \in E$ with $u \in A$ and $v \in B$, agent $a_{u,v}$ approves of candidate c_u in level 1 and c_v in level 2. For the extra agent a_j^*, $j \in \{1, \ldots, p\}$, we elect candidate c_j^* in level 1. These agents do not approve of any candidate in level 2. Finally, let $x := 0$, $y := 1$ and $k := k_B$.
Running time: Constructing our candidate and agent sets can be done in $\mathcal{O}(|V| + k + |E|)$ time. The voting profiles can be constructed in $\mathcal{O}(|E| + k)$ time. Hence, our

construction runs in polynomial time.
Correctness: We claim that I is a yes-instance if and only if I' is a yes-instance.

($\Rightarrow$) Let $X \subseteq V$ with $|X \cap A| \leq k_A$ and $|X \cap B| \leq k_B$ be a solution to I, that is, X forms a vertex cover of G. We claim that $\mathcal{C} = (C_1, C_2)$ with $C_1 = \{u \mid u \in X \cap A\} \cup \{c_j^* \mid j \in \{1, \dots, p\}\}$ and $C_2 = \{v \mid v \in X \cap B\}$ is a solution for I'. Assume towards a contradiction that $\mathcal{C}$ is not a solution to I'. This could mean that there exists an agent that does not have an approved candidate in the committee sequence. After our construction, this agent is an edge agent, which means the corresponding edge is not covered by X, contradicting X being a vertex cover of G. On the other hand, this could also mean that $|C_1| > k$. Since every filler agent has an approved candidate elected in C_1, the k candidates in the chosen committee will contain more then $k - p = k_A$ vertex candidates from A, which contradicts the fact that $|X \cap A| \leq k_A$. It follows that $\mathcal{C}$ is a solution to I'.

($\Leftarrow$) Let $\mathcal{C} = (C_1, C_2)$ be a solution to I'. We show that $X = \{v \in V \mid c_v \in (C_1 \cup C_2)\}$ is a solution to I. Assume towards a contradiction that this is not true. Then there exists an edge in E that is not covered by X or the size constraints are not satisfied. In the first case, the corresponding edge agent does not have his approved candidates in $\mathcal{C}$, contradicting $\mathcal{C}$ being a solution to I'. Otherwise, $|X \cup A| > k_A$ or $|X \cup B| > k_B$. Since the approved candidates in level 1 correspond to vertices in A and p filler candidates and the approved candidates in level 2 correspond to vertices in B, this would mean that $|C_1| > k_A + p = k$ or $|C_2| > k_B = k$, contradicting that every committee contains at most k candidates. It follows that X is a solution to I. □

We proved $(\geq, \geq)$ BICMCE WITH $\leq$ 1-APPROVAL VOTING to be NP-hard even with two levels. In this proof our committee size is dependent on the input. We now show that our problem remains NP-hard when we have a constant committee size.

Theorem 3.6. *$(\geq, \geq)$* BICMCE WITH $\leq$ 1-APPROVAL VOTING *is NP-hard even if $x = 0$, $k = 1$, and $y = 1$.*

In the following proof we give a polynomial-time many-one reduction from the following NP-hard [Fel+09] problem:

Problem 4: MULTICOLORED CLIQUE
Input: An undirected graph $G = (V, E)$ and a coloring $f : V \to \{1, \dots, k\}$ of the vertices in V.
Question: Does G have a clique of size exactly k where each vertex has a different color?

Proof. Let $I = (G = (V, E), k)$ be an instance of MULTICOLORED CLIQUE, with a coloring $f : V \to \{1, \dots, k\}$. We construct an instance $I' := (A, C, u, k', x, y)$ of $(\geq, \geq)$ BICMCE WITH $\leq$ 1-APPROVAL VOTING in polynomial time as follows.
Construction: Construct the set C of candidates by creating one candidate c_v for every vertex $v \in V$ and one candidate c_e for every edge $e \in E$. To build the set A of agents, for every edge $e = \{u, v\} \in E$ we create two agents a_u^e and a_v^e. Let $\tau := k + (|E| - \binom{k}{2})$. Next construct the voting profiles as follows. In levels 1 to k, every agent a_v^e approves of the corresponding vertex candidate c_v in level $f(v)$, else they do not approve of

	1 ⋯	i ⋯	j ⋯	l ⋯	k	$k+1$		⋯		τ
⋯			⋯					⋯		
$a_u^{u,v}$	⋯	c_u	/	/	⋯	⋯	$c_{u,v}$	$c_{u,v}$	$c_{u,v}$	⋯
$a_v^{u,v}$	⋯	/	c_v	/	⋯	⋯	$c_{u,v}$	$c_{u,v}$	$c_{u,v}$	⋯
$a_v^{v,w}$	⋯	/	c_v	/	⋯	⋯	$c_{v,w}$	$c_{v,w}$	$c_{v,w}$	⋯
$a_w^{v,w}$	⋯	/	/	c_w	⋯	⋯	$c_{v,w}$	$c_{v,w}$	$c_{v,w}$	⋯
$a_u^{u,w}$	⋯	c_u	/	/	⋯	⋯	$c_{u,w}$	$c_{u,w}$	$c_{u,w}$	⋯
$a_w^{u,w}$	⋯	/	/	c_w	⋯	⋯	$c_{u,w}$	$c_{u,w}$	$c_{u,w}$	⋯
$a_u^{u,z}$	⋯	c_u	/	/	⋯	⋯	$c_{u,z}$	$c_{u,z}$	$c_{u,z}$	⋯
$a_z^{u,z}$	⋯	/	/	c_z	⋯	⋯	$c_{u,z}$	$c_{u,z}$	$c_{u,z}$	⋯
⋯			⋯					⋯		

Figure 3.1: This is an excerpt of the voting profiles for the proof of Theorem 3.6.

any candidate. In levels $k+1$ to τ every agent a_v^e approves of it's corresponding edge candidate c_e. Finally, let $x := 0$, $k := 1$, and $y := 1$. This finishes the construction.
Running time: Constructing our candidate and agent sets can be done in $\mathcal{O}(|V|+3|E|)$ time and the voting profiles can be constructed in $\mathcal{O}((k+k^2+|E|)\cdot 2|E|)$ time. Hence, our construction runs in polynomial time.
Correctness: We claim that I is a yes-instance if and only if I' is a yes-instance.

($\Rightarrow$) Let $V' \subseteq V$ with $|V'| \geq k$ be a solution to I such that $V' = \{v_1, \ldots, v_k\}$ forms a multicolored clique of G. We want to show that $\mathcal{C} = (C_1, \ldots, C_\tau)$ is a solution to I', where $C_i = \{c_{v_i}\}$ for every $i \in \{1, \ldots, k\}$, and for $j \in \{k+1, \ldots, \tau\}$ we elect one candidate for every edge $e_j \in E \setminus \{\{v, u\} \mid v, u \in V'\}$, so $C_j = \{c_{e_j}\}$. The corresponding vertices to our elected candidates in the first k levels build a multicoloured clique, which means that at least $2 \cdot \binom{k}{2}$ agents have an approved candidate in these levels. Now there are at most $|E| - \binom{k}{2}$ edges whose corresponding agents do not have an approved candidate in the first levels. For each of these edges e we approve of candidate c_e in one of the levels $k+1$ to $k + (|E| - \binom{k}{2})$, as illustrated in Figure 3.1. Since every edge is covered, every agent will have at least one candidate she approves. Since $x = 0$, $score(C_i) \geq x$. Thus $\mathcal{C}$ is a solution to I'.

($\Leftarrow$) Let $\mathcal{C} = (C_1, C_2, \ldots, C_\tau)$ be a solution of I'. We claim that $V' = \{v_i \mid c_{v_i} \in \bigcup_{i=1}^k C_i\}$ is a multicolored clique of G. Assume there exists an empty committee, then we elect $|E| - \binom{k}{2} + k - 1$ candidates to satisfy $2 \cdot |E|$ agents. Due to our construction we can elect at most $(|E| - \binom{k}{2})$ edge candidates, so $2 \cdot (|E| - \binom{k}{2})$ agents have their approved candidate in the last levels, the other $\binom{k}{2}$ edge agent pairs have to be satisfied in the first $k-1$ levels. Due to the fact that these edges contain k distinct vertices, the k corresponding candidates will have to be elected, so that every agent has an approved candidate in $\mathcal{C}$. This can not be achieved in $k-1$ levels, since every committee can have at most one candidate, so every committee has to have a candidate. We know that these k chosen candidates represent k nodes of different colors, since a vertex is only approved in level $\ell \in \{1, \ldots, k\}$ if its color corresponds to ℓ. Assume these k vertices do not form a clique. This means there exists vertices v_i and v_j in V' that are not connected by an edge. Due to our construction, at most $2 \cdot (|E| - \binom{k}{2})$ agents have their approved candidate in the last levels, the other $2 \cdot \binom{k}{2}$ agents have to be satisfied in the first k levels. This means we have to elect both endpoint candidates of $\binom{k}{2}$ edges with k nodes. If $\{v_i, v_j\} \notin E$ this is not possible, thus $\mathcal{C}$ would not be a solution to I'. It follows that V' is a multicoloured clique and I is a yes-instance. □

Theorem 3.7. *($\geq, \geq$)* BICMCE WITH $\leq$ 1-APPROVAL VOTING *is NP-hard, even with two candidates and, unless the SETH breaks, admits no* $(2-\varepsilon)^{\tau} \cdot \text{poly}(\tau+n)$ *- algorithm.*

In the following proof we give a polynomial-time many-one reduction from the following NP-hard [Coo71] Problem:

Problem 5: SAT
Input: A set X of variables and a CNF formula φ over X.
Question: Is there a satisfying truth assignment $f : X \to \{\bot, \top\}$ for φ?

The Strong Exponential Time Hypothesis (SETH) was introduced in 2001 by Impagliazzo and Paturi [IP01]. In a simplified form, it claims that for deciding CNF-SAT, one cannot find an algorithm which outperforms the trivial brute-force algorithm in the worst case. A consequence of the SETH as stated by Komusiewicz et al. [Kom+19] is:

Hypothesis 3.8 (STRONG EXPONENTIAL TIME HYPOTHESIS [IP01])**.** SAT *does not admit a* $(2-\varepsilon)^N \cdot (N+M)^{\mathcal{O}(1)}$ *algorithm for any* $\varepsilon > 0$.

Proof of Theorem 3.7. Let $I = (X, \varphi)$ be an instance of SAT with variable set $X = \{x_1, \ldots, x_N\}$ and formula $\varphi = \bigwedge_{i=1}^{M} K_i$. We construct an instance $I' = (A, C, u, k, x, y)$ of ($\geq, \geq$) BICMCE WITH $\leq$ 1-APPROVAL VOTING in polynomial time as follows.
Construction: We assume that no clause contains a variable both negated and non-negated, else the clause would always be satisfied and can be ignored. Next we create the set C of candidates by creating 2 candidates, $c_\top$ and $c_\bot$. Let $\tau = N$ and create an agent a_K for every clause K in φ. For every variable x_i that appears as a literal in K, in level i, a_K approves $c_\top$ in if x_i appears as a positive literal or $c_\bot$ if it appears as a negative literal (see Figure 3.2). In every other layer a_K does not approve of any candidate. Finally, let $x := 0$, $k := 1$, and $y := 1$. This finishes the construction.
Running time: Adapting the formula can be done in $\mathcal{O}(M \cdot N)$ time, since we examine every clause once and a clause with more then N variables, will always be ignored. We create the voting profiles in $\mathcal{O}(M \cdot N)$ time. The rest of our construction can be done in constant time, which makes the complete construction polynomial.
Correctness: We claim that I is a yes-instance if and only if I' is a yes-instance.

(⇒) Let f be a satisfying truth assignment to φ. We claim that $\mathcal{C} = (C_1, \ldots, C_N)$ is a solution to I', where the candidate chosen in C_i is $c_\top$ if $f(x_i) = \top$ and $c_\bot$ if $f(x_i) = \bot$. We now have to prove that every agent has at least one approved candidate in this committee sequence. Since f is a satisfying truth assignment, every clause in φ contains at least one corresponding literal. Hence, the corresponding agent will have this candidate approved in our committee sequence. It follows that $\mathcal{C}$ is a solution to I'.

(⇐) Let $\mathcal{C} = (C_1, C_2, \ldots, C_N)$ be a solution of I'. We claim that f is solution to I, where $f : X \to \{\bot, \top\}$ and $f(x_i) := \top$ if the candidate chosen in level i is $c_\top$ or no

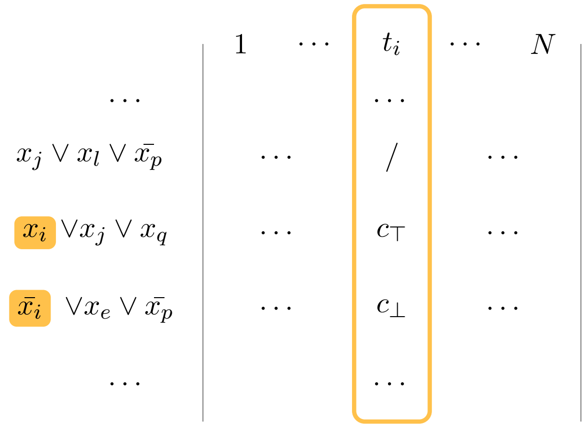


Figure 3.2: This excerpt of an example illustrates how the voting profiles correlate with the clauses in φ for the proof of Theorem 3.7.

candidate is chosen and $f(x_i) = \bot$ if the candidate chosen in level i is $c_\bot$. Assume this truth assignment does not satisfy φ. Then there exists a clause K that is not satisfied by f. Due to our construction the corresponding agent a_K will therefore not have an approved candidate in $(C_1, C_2, \ldots, C_N)$. This is a contraction to $\mathcal{C}$ being a solution to I'. Furthermore every clause that is not contained in our adapted formula will be satisfied no matter the truth assignment. It follows that I is a `yes`-instance for SAT.

From our reduction we have $\tau \in \mathcal{O}(N)$ and $n \in \mathcal{O}(M)$. We can therefore deduce that $(\geq, \geq)$ BICMCE with $\leq$ 1-Approval Voting has no $(2-\varepsilon)^\tau \cdot (\tau + n)^{\mathcal{O}(1)}$ algorithm (Hypothesis 3.8). ☐

Theorem 3.9. *$(\geq, \geq)$ BICMCE with $\leq$ 1-Approval Voting is NP-hard, even if there are three levels, and, unless the ETH breaks, admits no $2^{o(n+m)} \cdot (n+m)^{\mathcal{O}(1)}$-time algorithm.*

In the following proof for Theorem 3.9 we give a polynomial-time many-one reduction from the following NP-hard [GJ82] Problem:

Problem 6: 3-SAT
Input: A set X of variables and a 3-CNF formula φ over X.
Question: Is there a satisfying truth assignment $f : X \to \{\bot, \top\}$ for φ?

The Exponential Time Hypothesis (ETH) was introduced by Impagliazzo and Paturi [IP01] in 2001. In a simplified form, it states:

Hypothesis 3.10 (Exponential time hypothesis [IP01])**.** 3-SAT *does not admit a* $2^{o(N)} \cdot (N+M)^{\mathcal{O}(1)}$ *algorithm.*

Proof of Theorem 3.9. Let $I = (X, \varphi)$ be an instance of SAT, where $X = \{x_1, \ldots, x_N\}$ is a set of N variables and $\varphi = \bigwedge_{i=1}^{M} K_i$ a 3-CNF formula, where $K_i = (\ell_1 \vee \ell_2 \vee \ell_3)$. We construct an instance $I' = (A, C, u, k, x, y)$ of $(\geq, \geq)$ BICMCE with $\leq$ 1-Approval Voting in polynomial time as follows.
Construction: We start by creating our set of agents, by creating one agent a_K for

every clause $K \in \varphi$ and 6 agents $a_{x_i,j}$ for every variable $x_i \in X$ and $j \in \{1,\dots,6\}$. Next we create $2 \cdot N$ candidates, representing our possible literals. Let $\tau := 3$. For every clause $K_i = (\ell_1 \vee \ell_2 \vee \ell_3)$, $i \in \{1,\dots,M\}$, let $u_j(a_i) = z_j$, if ℓ_j is a positive literal and $u_j(a_i) = y_j$ if it is negative for $j \in \{1,2,3\}$. For our variable agents we set the voting profiles as follows:

$$
\begin{array}{llllll}
u_1(a_{x_i,1}) = y_i & u_1(a_{x_i,2}) = z_i & u_1(a_{x_i,3}) = y_i & u_1(a_{x_i,4}) = z_i & u_1(a_{x_i,5}) = / & u_1(a_{x_i,6}) = / \\
u_2(a_{x_i,1}) = z_i & u_2(a_{x_i,2}) = y_i & u_2(a_{x_i,3}) = / & u_2(a_{x_i,4}) = / & u_2(a_{x_i,5}) = y_i & u_2(a_{x_i,6}) = z_i \\
u_3(a_{x_i,1}) = / & u_3(a_{x_i,2}) = / & u_3(a_{x_i,3}) = z_i & u_3(a_{x_i,4}) = y_i & u_3(a_{x_i,5}) = z_i & u_3(a_{x_i,6}) = y_i
\end{array}
$$

Finally, let $x := 0$, $k := N$ and $y := 1$. This finishes the construction.

Running time: Our sets of agents and candidates can be created in $\mathcal{O}(8 \cdot N + M) \subseteq \mathcal{O}(N + M)$. For our agents we create $(6 \cdot N + M) \cdot 3$ voting profiles, which can also be done in $\mathcal{O}(N + M)$. The rest of our construction can be done in constant time, which makes the complete construction polynomial.

Correctness: We claim that I is a `yes`-instance if and only if I' is a `yes`-instance.

($\Rightarrow$) Let f be a satisfying truth assignment to I. We claim that (C_1, C_2, C_3) is a solution of I', where for all $j \in \{1,\dots,N\}$ the chosen candidate in C_i, $i \in \{1,2,3\}$, is z_j if $f(x_j) = \top$ and y_j if $f(x_j) = \bot$. We now have to prove that every agent has at least one approved candidate in this committee sequence. Since f is a satisfying truth assignment, every clause in φ contains at least one corresponding literal. Hence, the corresponding agent will have this candidate approved in our committee sequence. For our variable agents, we see that if the same variable candidate is chosen in the 3 committees, all 6 agents will have voted for this candidate at one point, thus will have at least one approved candidate. It follows that (C_1, C_2, C_3) is a solution to I'.

($\Leftarrow$) Let $\mathcal{C} = (C_1, C_2, C_3)$ be a solution of I'. Assume that there exists a committee which contains z_i and y_i. This will satisfy 4 of our 6 corresponding variable agents $a_{x_i,j}$, the other 2 will have to have an approved candidate in another committee. Due to our construction these two agents will not have the same voting profiles, thus we would have to elect 4 candidates to cover all of the agents (see Figure 3.3). Now since we have N candidates in one committee, the number of possible elected candidates in our sequence is $3 \cdot N$. To cover all of our variable agents we will need at least $\frac{6}{2}$ elected candidates. So 3 for every variable $j \in \{1,\dots,N\}$. It follows that, if we elect 4 candidates for one variable, another variable agent will not have their approved candidate in the committee sequence, contradicting the fact that $\mathcal{C}$ is a solution to I. So every variable has at most one assignment, making it a correct truth assignment. We claim that f is solution to I, where $f : X \to \{\bot, \top\}$ and $f(x_i) = \top$ if the candidate chosen in level 1 is z_i and $f(x_i) = \bot$ if the candidate chosen in level 1 is y_i. Assume this truth assignment does not satisfy φ. So there exists a clause ℓ in φ that is not satisfied by our assignment. In this case the corresponding clause agent a_ℓ will not have any approved candidate in our sequence, which means $\mathcal{C}$ would not be a solution to I' - a contradiction. It follows that I is a `yes`-instance for 3-SAT. □

From our reduction we have $n + m \in \mathcal{O}(N + M)$. We can therefore deduce from Hypothesis 3.10 that $(\geq, \geq)$ BICMCE WITH $\leq$ 1-APPROVAL VOTING has no $2^{o(n+m)} \cdot (n+m)^{\mathcal{O}(1)}$-time algorithm.

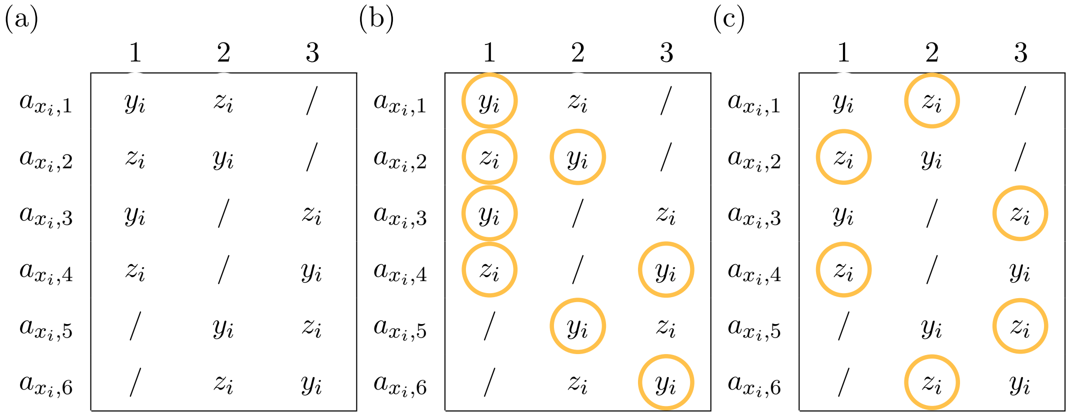


Figure 3.3: Excerpt (a) of our voting profiles for Theorem 3.9, illustrating why when we vote for 2 complementary candidates (b) we need to elect 4 candidates to satisfy our agents, but if we elect the same candidate in all three levels (c) 3 candidates are sufficient.

3.3 Parameterized Complexity

In the last chapter we have seen that $(\geq, \geq)$ BICMCE WITH ≤ 1-APPROVAL VOTING is NP-hard even with constant variables, allowing us to deduct para-NP-hardness for τ, m, k, x, and y. In this chapter we want to focus on the parameterized complexity of combinations of our variables and of n.

3.3.1 Number of candidates m and levels τ

Theorem 3.11. *$(\geq, \geq)$ BICMCE WITH ≤ 1-APPROVAL VOTING parameterized by $\tau + k$ is contained in* XP.

Theorem 3.12. *$(\geq, \geq)$ BICMCE WITH ≤ 1-APPROVAL VOTING is in* FPT *when parameterized by $m + \tau$.*

We prove Theorems 3.11 and 3.12 via the following lemma:

Lemma 3.13. *Given an instance $I = (A, C, u, k, x, y)$ with n agents, m candidates and τ voting profiles, a solution to I can be computed in $\mathcal{O}(k \cdot n \cdot \tau \cdot m^{k \cdot \tau})$ time.*

Proof. Let m be the number of candidates, τ the number of levels, and k the maximum committee size. There exist $\binom{m}{k}^{\tau}$ possible committee sequences. These can be computed in $\mathcal{O}((m^k)^{\tau}) = \mathcal{O}(m^{k \cdot \tau})$ time. For each committee sequence we can now check if our constraints are covered, which can be done in $\mathcal{O}(k \cdot n \cdot \tau)$ time. If one sequence meets all the requirements we have a solution to I. If we cannot find such a solution after checking all possible committees, we can conclude that there is no committee sequence satisfying our requirements. It follows that deciding $(\geq, \geq)$ BICMCE WITH ≤ 1-APPROVAL VOTING can be done in $\mathcal{O}(k \cdot n \cdot \tau \cdot m^{k \cdot \tau}) \subseteq \mathcal{O}(n \cdot \tau \cdot m^{m \cdot \tau})$ since $k \leq m$ and $k \leq n$. □

Proposition 3.14. *Solving an instance $I = (A, C, u, k, x, y)$ with n agents, m candidates and τ voting profiles is equivalent to solving an instance $I' = (A', C, u, k, x, y)$, with n' agents, where $n' \geq n$, when $x = 0$.*

Proof. We build the set of agents in I' as follows: $A' = A \cup \{a_i \mid i \in \{1, \dots, n'-n\}$. The added $n'-n$ agents copying the exact voting profile of one agent $a \in A$.

C is a solution to I.
$\Leftrightarrow$ Every agent in A has y of his approved candidates in the committee sequence.
Which means a has y approved candidates in C.
$\Leftrightarrow$ Every agent in A' has y of his approved candidates in the committee sequence.
$\Leftrightarrow$ C is a solution to I'.

□

Lemma 3.15. *Solving an instance $I = (A, C, u, k, x, y)$ with n agents, m candidates and τ voting profiles is equivalent to solving an instance $I' = (A, C', u, k, x, y)$, with m' candidates, where $m' \geq m$.*

Proof. Due to the fact that unapproved candidates do not influence our voting profiles and accordingly our x and y scores, we can add $m' - m$ candidates to our instance, without changing our set of possible solutions. □

Lemma 3.16. *Solving an instance $I = (A, C, u, k, x, y)$ with n agents, m candidates and τ voting profiles is equivalent to solving an instance $I' = (A, C, u', k, x, y)$, with τ' levels, where $\tau' \geq \tau$, when $x = 0$.*

Proof. By adding $\tau' - \tau$ levels to our instance in which every agent in A does not approve of a candidate, the y-score can not be influenced. Therefore, since $score(C) \geq 0$ for every committee C, a solution to I is equivalent to a solution to I'. □

Theorem 3.17. *$(\geq, \geq)$* BICMCE WITH $\leq$ 1-APPROVAL VOTING *admits no problem kernel of size polynomial in $m + \tau$ unless NP $\subseteq$ coNP $\setminus$ poly.*

To prove Theorem 3.17, we are going to prove that $(\geq, \geq)$ BICMCE WITH $\leq$ 1-APPROVAL VOTING, when parameterized by $m+\tau$, is OR-compositional [BJK14]. That is, there is an algorithm taking p instance $I_1, \dots, I_p$, each with the same number m of candidates (Lemma 3.15), n of agents (Proposition 3.14) and τ of levels (Lemma 3.16). It then constructs in time polynomial in $\sum_{i=1}^{p} |I_i|$ an instance I such that the number of levels and candidates is in $(m + n + \log(p))^{O(1)}$, and that I is a `yes`-instance if and only if at least one of $I_1, \dots, I_p$ is a `yes`-instance.

Construction 1. Consider $2^{\log(p)}$ instances $I_1, \dots, I_p$ of $(\geq, \geq)$ BICMCE WITH $\leq$ 1-APPROVAL VOTING with $k = 1$, $x = 0$, and $y = 1$, each with n agents $A_i = \{a_{i,1}, \dots, a_{i,n}\}$, m candidates $C_i = \{c_{i,1}, \dots, c_{i,m}\}$, and τ voting profiles. We construct the agent set $A = \bigcup_{i=1}^{p} A_i$ and the candidate set $C = \{c_1, \dots, c_m\} \cup \{c_\top, c_\bot\}$ and let $\tau' = \tau + \log(p)$. Arrange the p voting profiles in the first τ levels, where we identify each candidate $c_{i,j}$ with c_j for all agent sets A_i, $i \in \{1, \dots, p\}$, and $j \in \{1, \dots, m\}$. Next, in levels $\tau + 1$ to $\tau + \log(p)$, for every $q \in \{1, \dots, p\}$, if the binary encoding of $q - 1$ at position $j \in \{1, \dots, \log(p)\}$ is 1, then the agents in A_q approve of candidate $c_\top$. If the binary encoding is 0 they approve of candidate $c_\bot$ in the $(\tau+j)$-th level (see Figure 3.4). Finally let $x' := 0$, $k' := 1$, and $y' := 1$.

		1	$\cdots$	j	$\cdots$	τ	$\tau+1$	$\cdots$	$\tau+\log(p)$
A_1	$a_{1,1}$	$u_1^1(a_1)$	$\cdots$	$u_j^1(a_1)$	$\cdots$	$u_\tau^1(a_1)$	$c_\bot$	$\cdots$	$c_\bot$
	$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\cdots$	$\vdots$	$\vdots$	$\cdots$	$\vdots$
	$a_{1,n}$	$u_1^1(a_n)$	$\cdots$	$u_j^1(a_n)$	$\cdots$	$u_\tau^1(a_n)$	$c_\bot$	$\cdots$	$c_\bot$
A_2	$a_{2,1}$	$u_1^2(a_1)$	$\cdots$	$u_j^2(a_1)$	$\cdots$	$u_\tau^2(a_1)$	$c_\bot$	$\cdots$	$c_\top$
	$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\cdots$	$\vdots$	$\vdots$	$\cdots$	$\vdots$
	$a_{2,n}$	$u_1^2(a_n)$	$\cdots$	$u_j^2(a_n)$	$\cdots$	$u_\tau^2(a_n)$	$c_\bot$	$\cdots$	$c_\top$
	$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\cdots$	$\vdots$	$\vdots$	$\cdots$	$\vdots$
A_p	$a_{p,1}$	$u_1^p(a_1)$	$\cdots$	$u_j^p(a_1)$	$\cdots$	$u_\tau^p(a_1)$	$c_\top$	$\cdots$	$c_\top$
	$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\cdots$	$\vdots$	$\vdots$	$\cdots$	$\vdots$
	$a_{p,n}$	$u_1^p(a_n)$	$\cdots$	$u_j^p(a_n)$	$\cdots$	$u_\tau^p(a_n)$	$c_\top$	$\cdots$	$c_\top$

Figure 3.4: This excerpt of our voting profiles for the proof of Theorem 3.17

By construction, we have the following.

Lemma 3.18. *Let $I' = (A, C, u, k', x', y')$ be the instance obtained from Construction 1 given $I_1, \dots, I_p$. Then, I' is a* `yes`*-instance if and only if one of the instances $I_1, \dots, I_p$ is a* `yes`*-instance.*

Proof. (⇒) Assume that no instance of $I_1, \dots, I_p$ is a `yes`-instance, meaning at least one agent of every subset has to have his approved candidate in level $\tau + 1$ to $\tau + \log(p)$. We can elect at most $\log(p)$ candidates in these levels. Since we have p distinct voting profiles that correspond to the binary encoding of the agent set index, no matter which candidates we elect in these levels, the agent set encoded by the complement of these candidates will not have an approved candidate in these committees. Making I' a `no`-instance for $(\geq, \geq)$ BICMCE WITH $\leq$ 1-APPROVAL VOTING.

(⇐) Let one instance I_q, $q \in \{1, \dots, p\}$, be a `yes`-instance of $(\geq, \geq)$ BICMCE WITH $\leq$ 1-APPROVAL VOTING, and let $\mathcal{C}_q = (C_{q,1}, \dots, C_{q,\tau})$ be a solution to I_q. We prove that $\mathcal{C} = (C_1, \dots, C_{\tau+\log(p)})$ is a solution to I', where $C_i = C_{q,i}$ for every $i \in \{1, \dots, \tau\}$ and in each level $\tau + t$, $t \in \{1, \dots, \log(p)\}$, we elect candidate $c_\bot$ if the binary encoding of $q - 1$ is 1 at position t and candidate $c_\top$ if its binary encoding is 0. Assume there is an agent a that does not have their approved candidate in $\mathcal{C}$. We know that the voting profiles at level $\tau + t$, $t \in \{1, \dots, \log(p)\}$ correspond to the binary encoding of the corresponding agent set. This means that agent a is part of the set encoded by the opposite of the chosen candidates, which by construction is q. Therefore there exists an agent in A_q that is not satisfied in the first τ levels, contradicting that $\mathcal{C}_q$ is a solution to I_q. It follows that $\mathcal{C}$ is a solution to I'. □

3.3.2 Number of agents n and levels τ

Theorem 3.19. *$(\geq, \geq)$* BICMCE WITH $\leq$ 1-APPROVAL VOTING *admits a problem kernel of size polynomial in $n + \tau$.*

Observation 3.20. *The maximum number of candidates being approved in one level can be upper bounded by the number of agents, since each one has exactly one vote. There can be at most $n \cdot \tau$ approved candidates over all levels.*

Observation 3.20 gives that if $m > n \cdot \tau$, then there are at least $m - n \cdot \tau$ candidates that are never approved. This will allow us to reduce any instance of $(\geq, \geq)$ BICMCE WITH $\leq$ 1-APPROVAL VOTING to an equivalent instance with $m \leq n \cdot \tau$.

Proposition 3.21. *There is an algorithm that, given any instance $I = (A, C, u, k, x, y)$ of $(\geq, \geq)$* BICMCE WITH $\leq$ 1-APPROVAL VOTING, *computes an equivalent instance $I' = (A, C', u, k, x, y)$ with $|C'| \leq |A| \cdot \tau$ in $\mathcal{O}(m \cdot n \cdot \tau)$ time.*

To obtain Proposition 3.21 we employ the following reduction rule.

Reduction Rule 3.3.1. *If $m > n \cdot \tau$, then delete all candidates which are never approved.*

Proof. Applying Reduction Rule 3.3.1 to an instance I creates a new instance I' where we deleted a candidate $z \in C$ which is never approved. We show that there is a solution to I' if and only if there is a solution to I.

($\Rightarrow$) Let $(C_1, \ldots, C_\tau)$ be a solution to I containing z. We prove that $(C'_1, \ldots, C'_\tau)$ is a solution to I', where $C'_i := C_i \setminus \{z\}$. Since z is not approved by any agent in A deleting it from the chosen committees would not break any of our constraints. Thus, $(C'_1, \ldots, C'_\tau)$ is a solution to I'.

($\Leftarrow$) Now assume that there exists a solution $C = (C_1, \ldots, C_\tau)$ to I'. In this case C is also a solution to I, since the set of agents and their utilities are equivalent and every candidate in I' also exists in I. This proves our rule to be correct. ☐

By applying Reduction Rule 3.3.1 we get Proposition 3.21.

Corollary 3.22. *Given an instance $I = (A, C, u, k, x, y)$ with n agents, m candidates, and τ levels, a solution to I can be computed in $\mathcal{O}((n \cdot \tau)^{(n \cdot \tau) \cdot \tau})$ time.*

Proof. We start by constructing an equivalent instance I' in $\mathcal{O}(m \cdot n \cdot \tau)$ where $m \leq n \cdot \tau$, using Proposition 3.21. Due to Theorem 3.12, we can solve $(\geq, \geq)$ BICMCE WITH $\leq$ 1-APPROVAL VOTING in $\mathcal{O}(m^{m \cdot \tau}) \subseteq \mathcal{O}((n \cdot \tau)^{(n \cdot \tau) \cdot \tau})$ since $m \leq n \cdot \tau$. ☐

3.3.3 Number of agents n

Theorem 3.23. *$(\geq, \geq)$* BICMCE WITH $\leq$ 1-APPROVAL VOTING *parameterized by n is contained in* XP.

Theorem 3.24. *$(\geq, \geq)$* BICMCE WITH $\leq$ 1-APPROVAL VOTING *is in* FPT *when parameterized by $n + y$.*

First, we define the boolean dynamic programming table needed for our algorithm.

Definition 3.25. The dynamic programming table of an instance $I = (A, C, u, k, x, y)$ with $n = |A|$ and $C = \{c_1, \ldots, c_m\}$ of $(\geq, \geq)$ BICMCE WITH $\leq$ 1-APPROVAL VOTING is a boolean table D with entries

$$D[t, \mathbf{y}] \in \{\top, \bot\}$$

where $t \in \{0 \ldots, \tau\}$ and $\mathbf{y} \in \{0, \ldots, y\}^n$, with $\mathbf{y} = (y_1, \ldots, y_n)$. We interpret the table as follows. An entry $D[t, \mathbf{y}]$ in the table is true if and only if there are committees $C_1, \ldots, C_t$ each with committee size at most k and a score of at least x such that the score of each agent a_i at time t sums up to at least y_i.

Lemma 3.26. *Given an instance $I = (A, C, u, k, x, y)$ with n agents, m candidates, and τ levels, the dynamic programming table of I can be computed in $\mathcal{O}(\tau \cdot (y+1)^n \cdot 2^n)$ time.*

Proof. We note that the table is of size $\tau \cdot (y + 1)^n$. Next, we show how to fill the table and compute the entries efficiently.
Construction: First, we set

$$D[0, \mathbf{y}] = \begin{cases} \top, & \text{if } \mathbf{y} = (0, \ldots, 0) \\ \bot, & \text{otherwise} \end{cases}$$

For $t > 0$, we compute all entries $D[t, \cdot]$ as follows, assuming all entries $D[t-1, \cdot]$ have already been computed. Set $D[t, \mathbf{y}]$ to true, if there exists a $D[t-1, \mathbf{y}']$ set to true

- with $\mathbf{y} \leq \mathbf{y}'$, and there exists a committee C' of size at most k at time t and score of at least x.
- and a committee C' of size at most k at time t and score of at least x with approval vector $\vec{c}$ where

$$c_i = \begin{cases} 1, & \text{if } |u_t(a_i) \cap C'| \geq 1, \\ 0, & \text{otherwise,} \end{cases}$$

  such that $\mathbf{y}' + \vec{c} = \mathbf{y}$. Otherwise set it to false.

Running time: In every level there can be at most n approved candidates, which gives us a maximum of 2^n possible committees. To find all valid committees, containing at most k candidates and having a score of x we can sort out all possible committees that do not meet the constraints. This tells us that all acceptable committees can be computed in $\mathcal{O}(n \cdot 2^n)$. Therefore checking if there exists a $D[t-1, \mathbf{y}']$ set to true and a correct committee whose approval vector adds up to $\mathbf{y}$ can be done in $\mathcal{O}((y+1)^n \cdot 2^n \cdot n)$. Since we do this for every level, filling the whole table can be done in $\mathcal{O}((y+1)^n \cdot 2^n \cdot n \cdot \tau)$.
Correctness: We claim that each entry $D[t, \mathbf{y}]$ is set to true if and only if there are committees $C_1, \ldots, C_t$ such that the score of agent a_i at level t sums up to at least y_i. We prove our claim via induction. Initially note that for $t = 0$ all entries are correct, since before electing any committee every agent has a score of 0. So only $D[0, \vec{0}]$ can be achieved. Now assume that there exists a $t \in \{0, \ldots, \tau\}$ such that $D[t, \mathbf{y}]$ is set to true if and only if there are committees $C_1, \ldots, C_t$ such that the score of agent a_i at level t sums up to at least y_i. We want to show that this also applies to level $t + 1$.

($\Rightarrow$) Let $D[t+1, \mathbf{y}]$ be set to true. Due to our induction assumption this could mean $D[t, \mathbf{y}]$ is set to true, in which case no matter what valid committee we add to it the score of every agent a_i will be at least y_i. The other possibility would be that there exists a table entry $D[t, \mathbf{z}]$ set to true and a valid committee C' at level $t+1$, with approval vector $\mathbf{c}'$, such that $\mathbf{z} + \mathbf{c}' = \mathbf{y}$. Since there is a committee sequence $(C_1, \ldots, C_t)$ where the score of agent a_i at level t sums up to at least z_i, the sequence $(C_1, \ldots C_t, C')$ has a score of at least $z_i + c'_i = y_i$ for every agent a_i. In both cases, there exists a committee sequence $(C_1, \ldots, C_{t+1})$ such that the score of agent a_i at level $t+1$ sums up to at least y_i.

($\Leftarrow$) Let $(C_1, \ldots, C_{t+1})$ be a committee sequence such that the score of candidate i at level $t+1$ sums up to y_i. Let $\mathbf{c}$ be the vector of approval for committee C_{t+1}. We know that in $(C_1, \ldots, C_t)$ the score of every agent a_i sums up to at least $y_i - c_i$. Which from our induction assumption would mean that our algorithm would have set $D[t, y_i - c_i]$ to true and we would find committee C_{t+1}, meaning $D[t+1, \mathbf{y}']$ would be set to true. □

Given Lemma 3.26, we can prove Theorems 3.23 and 3.24 as follows:

Proof (Theorems 3.23 and 3.24). Let $I = (A, C, u, k, x, y)$ be an instance of $(\geq, \geq)$ BICMCE WITH $\leq$ 1-APPROVAL VOTING with n agents and m candidates. We construct the table D in $\mathcal{O}((y+1)^n \cdot 2^n \cdot \tau)$ and then check if $D[\tau, (y, \ldots, y)]$ is set to true. If so then I is a `yes`-instance, else I is a `no`-instance. □

3.4 Changing Inequalities

We have seen that $(\geq, \geq)$ BICMCE WITH $\leq$ 1-APPROVAL VOTING is NP-hard even for $x = 0$ and $y = 1$. In the following we want to analyze the computational complexity of the problem, after changing the inequalities of our problem. We can begin by stating that $(\leq, \leq)$ BICMCE WITH $\leq$ 1-APPROVAL VOTING is solvable in polynomial time, due to the fact that not electing any candidate is always an apparent solution. If we only change one inequality the problem stays NP-hard, which we prove in the upcoming subsection.

Problem 7: $(\geq, \leq)$ BICMCE WITH $\leq$ 1-APPROVAL VOTING

Input: A set $A = \{a_1, \ldots, a_n\}$ of agents, a set C of candidates, voting profiles $u_t : A \to C \cup \{\emptyset\}$ for $t \in \{1, \ldots, \tau\}$, $\tau \in \mathbb{N}$, and integers $x, y, k \in \mathbb{N}_0$.

Question: Is there a sequence $(C_1, \ldots, C_\tau)$ of candidate subsets $C_t \subseteq C$, with $|C_t| \leq k$, for all $t \in \{1, \ldots, \tau\}$ such that:

$$\text{for all layers } t \in \{1, \ldots, \tau\} : \qquad \sum_{c \in C_t} |{u_t}^{-1}(c)| \geq x$$

$$\text{and for all agents } a_i \in A : \qquad \sum_{t=1}^{\tau} |u_t(a_i) \cap C_t| \leq y$$

Theorem 3.27. $(\geq, \leq)$ BICMCE WITH $\leq$ 1-APPROVAL VOTING *is NP-hard and W[1]-hard when parameterized by τ, even for constant x, y, and k.*

In the following proof we give a polynomial-time many-one reduction from the following NP-hard and W[1]-hard [Fel+09] Problem:

Problem 8: MULTICOLORED INDEPEDENT SET
Input: An undirected graph $G = (V, E)$ and a coloring $f : V \to \{1, \dots, r\}$ of the vertices in V.
Question: Does G have an independent set of size r where each vertex has a different color?

Proof. Let $I = (G = (V, E), r)$ be an instance of MULTICOLORED INDEPEDENT SET, let G be an undirected graph where every vertex has a coloring $f : V \to \{1, \dots, r\}$. We construct an instance $I' := (A, C, u, k, x, y)$ of $(\geq, \leq)$ BICMCE WITH $\leq$ 1-APPROVAL VOTING in polynomial time as follows.
Construction: Create the set C of candidates by creating one candidate c_v for every vertex $v \in V$. To build the set A of agents, we create one agent a_e for every edge $e \in E$ and one agent a_v for every vertex $v \in V$. Let $\tau := r$. Next construct the voting profiles as follows. Every agent a_e approves of the corresponding vertex candidate to their endpoints if their color corresponds to the level number, else they do not approve of any candidate. Our vertex agents also approve of the corresponding vertex candidate if their color corresponds to the level number. Finally let $x := 1$, $k := 1$ and $y := 1$. This finishes the construction.
Running time: Creating our candidate and agent sets can be done in $\mathcal{O}(2|V| + |E|)$ time and the voting profiles can be constructed in $\mathcal{O}(r \cdot (|E| + |V|))$ time. Hence our construction runs in polynomial time.
Correctness: We claim that I is a `yes`-instance if and only if I' is a `yes`-instance.

($\Rightarrow$) Let $V' \subseteq V$ with $|V'| = r$ be a solution to I such that $V' = \{v_1, \dots, v_r\}$ builds a multicolored independent set of G. We now have to show that $\mathcal{C} = (C_1, \dots, C_r)$ is a solution to I' where $C_i = \{c_{v_i}\}$. Since V' is a independent set, no edge will have both its endpoints in the set. Which means no edge will have both its approved candidates in the committee, so the agent score never exceeds y. Since every committee contains exactly one candidate and there is at least the corresponding node agent which approves of this candidate, $score(C_i) \geq 1$. Thus $\mathcal{C}$ is a solution to I'.
($\Leftarrow$) Let $(C_1, C_2, \dots, C_r)$ be a solution of I'. We claim that $V' = \{v_i \mid c_{v_i} \in \bigcup_{i=1}^{k} C_i\}$ is a multicolored independent set of G. First of all due to our construction, we know that the r chosen candidates represent r vertices of different colors, since a vertex is only approved in level ℓ if its color corresponds to $\ell \in \{1, \dots, r\}$. Now we have to prove that these vertices are not connected by an edge. We know that if both end points of an edge were in our independent set, then there would be an edge agent with two approved candidates (as pictured in Figure 3.5). Due to our constraint $y \leq 1$ this cannot be the case. It follows that no two vertices in V' are connected by an edge. Therefore V' is a multicolored independent set and I is a `yes`-instance. □

It follows that $(\geq, \leq)$ BICMCE WITH $\leq$ 1-APPROVAL VOTING is NP-hard.

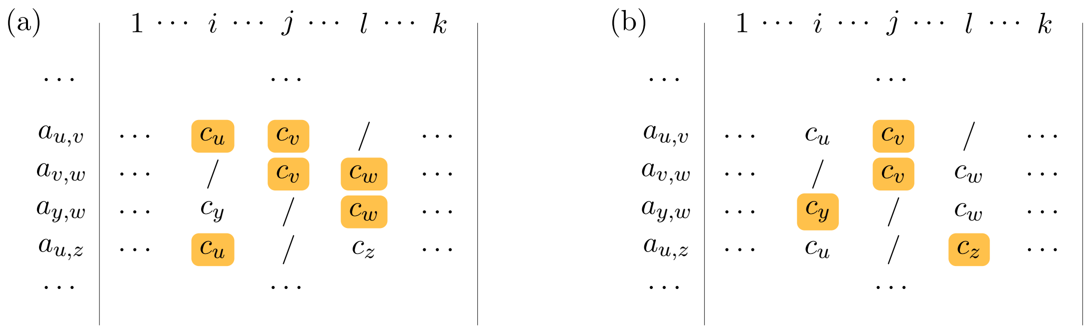


Figure 3.5: This is an excerpt of our voting profiles for the proof of Theorem 3.27 illustrating that the chosen candidates can not be connected by an edge or one agent will have a score of ≥ 1.

Problem 9: $(\leq, \geq)$ BICMCE WITH PLURALITY VOTING

Input: A set A $= \{a_1, \ldots, a_n\}$ of agents, a set C of candidates, voting profiles $u_t : A \to C \cup \{\emptyset\}$ for $t \in \{1, \ldots, \tau\}$, $\tau \in \mathbb{N}$, and integers $x, y, k \in \mathbb{N}_0$.

Question: Is there a sequence $(C_1, \ldots, C_\tau)$ of candidate subsets $C_t \subseteq C$, with $|C_t| \leq k$, for all $t \in \{1, \ldots, \tau\}$ such that:

$$\text{for all layers } t \in \{1, \ldots, \tau\} : \qquad \sum_{c \in C_t} |{u_t}^{-1}(c)| \leq x$$

$$\text{and for all agents } a_i \in A : \qquad \sum_{t=1}^{\tau} |u_t(a_i) \cap C_t| \geq y$$

Theorem 3.28. $(\leq, \geq)$ BICMCE WITH PLURALITY VOTING *is NP-hard.*

In the following proof we give a polynomial-time many-one reduction from the following NP-hard problem $(\geq, \geq)$ BICMCE WITH $\leq$ 1-APPROVAL VOTING with $x = 0$ (Theorem 3.6) Problem:

Proof. Let $I = (A, C, u, k, x, y)$ be an instance of $(\geq, \geq)$ BICMCE WITH $\leq$ 1-APPROVAL VOTING, where $|C| = m$, $|A| = n$ and $x = 0$. We construct an instance $I' := (A, C, u, k, x', y)$ of $(\leq, \geq)$ BICMCE WITH PLURALITY VOTING in polynomial time by setting $x' = n$.

Correctness: We claim that I is a yes-instance if and only if I' is a yes-instance.

($\Rightarrow$) Let $\mathcal{C}$ be a solution of I. We claim that $\mathcal{C}$ is also a solution to I'. Since every agent approves of at most one candidate, $score(C_i) = \sum_{c \in C_i} |u_i^{-1}(c)| \leq n = x'$ for every committee $C_i \in \mathcal{C}$. Due to the fact that every other variable stays the same, it follows that $\mathcal{C}$ is a solution to I'.

($\Leftarrow$) Let $\mathcal{C}'$ be a solution of I'. We want to prove that $\mathcal{C}'$ is a solution to I. Since $x = 0$, $x \leq score(C'_i) \leq x'$ for every committee $C'_i \in \mathcal{C}'$. Since all our other constraints are satisfied for I', it follows that I is a yes-instance for $(\geq, \geq)$ BICMCE WITH $\leq$ 1-APPROVAL VOTING. □

Chapter 4

Other Variants

After having analyzed the algorithmic complexity of $(\geq, \geq)$ BICMCE WITH $\leq$ 1-APPROVAL VOTING in detail, in this chapter we want to touch on the $(\geq, \geq)$ BICMCE problem with different voting rules. We start by looking at different approval voting methods and finish with ordinal voting, specifically with the borda rule.

4.1 Strict Plurality Voting

We take a look at strict plurality voting, which is closest to our main problem, the only difference being that every agent has to approve of a candidate. We model it with the following:

Problem 10: $(\geq, \geq)$ BICMCE WITH $=$ 1-APPROVAL VOTING

Input: A set $A = \{a_1, \ldots, a_n\}$ of agents, a set C of candidates, voting profiles $u_t : A \to C$ for $t \in \{1, \ldots, \tau\}$, $\tau \in \mathbb{N}$, and integers $x, y, k \in \mathbb{N}_0$.

Question: Is there a sequence $(C_1, \ldots, C_\tau)$ of candidate subsets $C_t \subseteq C$, with $|C_t| \leq k$, for all $t \in \{1, \ldots, \tau\}$ such that:

$$\text{for all layers } t \in \{1, \ldots, \tau\}: \qquad \sum_{c \in C_t} |{u_t}^{-1}(c)| \geq x$$

$$\text{and for all agents } a_i \in A: \qquad \sum_{t=1}^{\tau} |u_t(a_i) \cap C_t| \geq y$$

Theorem 4.1. *There is an algorithm that, given any instance $I = (A, C, u, k, x, y)$ of $(\geq, \geq)$ BICMCE WITH $\leq$ 1-APPROVAL VOTING, computes an equivalent instance $I' = (A', C', u', k, x', y)$ of $(\geq, \geq)$ BICMCE WITH $=$ 1-APPROVAL VOTING in polynomial time.*

Proof. Let $I = (A, C, u, k, x, y)$ be an instance of $(\geq, \geq)$ BICMCE WITH $\leq$ 1-APPROVAL VOTING, where $|C| = m$ and $|A| = n$. We construct an instance $I' := (A', C', u', k, x', y)$ of $(\geq, \geq)$ BICMCE WITH $=$ 1-APPROVAL VOTING in polynomial time as follows.
Construction: If $x = 0$ and there is a level in which every agent does not approve of a candidate, we delete this level. Next the set C' of candidates can be constructed by adding the candidates $c_1, \ldots, c_m$ from C and $n \cdot (n + 1) \cdot \tau$ filler candidates

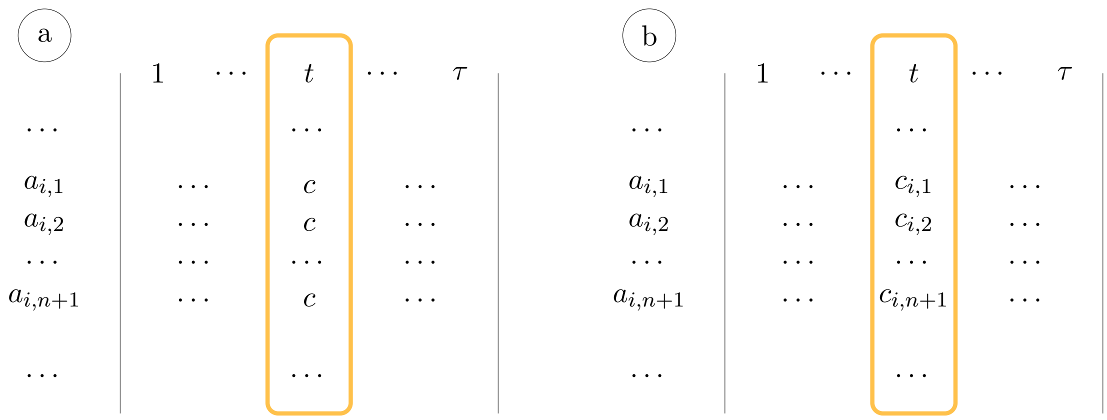


Figure 4.1: Illustration of the voting profiles in the construction behind the proof of Theorem 4.1, in the case that (a) $u_t(a_i) = c$. (b) $u_t(a_i) = /$.

$c_{1,1}, \ldots, c_{n,(n+1)\cdot\tau}$. For the set of agents A' we create $n \cdot (n+1) \cdot \tau$ agents, where agents $a_{i,1}, \ldots, a_{i,(n+1)\cdot\tau}$ correspond to agent a_i in A. We continue by constructing the voting profiles (illustrated in Figure 4.1). If agent a_i approves of candidate c in level t then the agents $a_{i,j}$, $j \in \{1, \ldots, (n+1) \cdot \tau\}$, also approve of this agent c. On the other hand, if agent a_i does not approve of a candidate then agent $a_{i,j}$ approves of candidate $c_{i,j}$ for all $j \in \{1, \ldots, (n+1) \cdot \tau\}$. Finally, let $x' := \max((n+1) \cdot \tau, x \cdot (n+1) \cdot \tau)$.

Running time: Deleting levels with no approvals can be done in $\mathcal{O}(\tau \cdot n)$. Creating our candidate and agent sets can be done in $\mathcal{O}((n+1)^2 \cdot \tau + m)$ time and the voting profiles can be constructed in $\mathcal{O}((n+1)^2 \cdot \tau)$ time. Hence, our construction runs in polynomial time.

Correctness: We claim that I is a `yes`-instance if and only if I' is a `yes`-instance.

($\Rightarrow$) Let $\mathcal{C} = (C_1, C_2, \ldots, C_\tau)$ be a solution of I. We claim that $\mathcal{C}' = (C_1', C_2', \ldots, C_\tau')$ is a solution to I', where if $C_i = \emptyset$ we elect a random candidate from C approved by an agent in level i (which always exists due to our construction) and $C_i' = C_i$ for all the other committees. We have τ committees of size k and every agent in C also exists in C', so what we have left to prove is that every agent in A' has y approved candidates in our sequence and that every committee has a score of at least x'. Assume this is not the case. That means there exists an agent with less then y approved candidates in the sequence. This agent is either contained in C, or has an agent in C with the same approved candidates, meaning $\mathcal{C}$ would not be a solution to I. For the second constraint, assume there exists a committee C_i' that has a score lower then $x' = \max((n+1)\cdot\tau, x\cdot(n+1)\cdot\tau)$. If $x > 0$ that means that we do not have x agents from A who approve of the committee, because their $n+1\cdot\tau$ copies approve of the same candidates in I', leading to a score of at least $x \cdot (n+1) \cdot \tau$, thus $\mathcal{C}$ would not be a solution to I. If $x = 0$ and $C_i' = \emptyset$ we elect an approved candidate from C, which due to our construction is approved by at least $n+1\cdot\tau$ agents. It follows that $\mathcal{C}'$ is a solution to I'.

($\Leftarrow$) Let $\mathcal{C}' = (C_1', C_2', \ldots, C_\tau')$ be a solution of I'. We want to prove that $\mathcal{C}$ is a solution to I, where $\mathcal{C} = ((C_1 = C \cap C_1'), (C_2 = C \cap C_2'), \ldots, (C_\tau = C \cap C_\tau'))$. Assume that there exists a committee C_i in $\mathcal{C}$, where $\mathit{score}(C_i) \leq x - 1$ in I. In that case $\mathit{score}(C_i) \leq (x-1) \cdot (n+1) \cdot \tau$ in I' and $\mathit{score}(C_i' \setminus C_i) \geq (n+1) \cdot \tau$. Since every candidate in $C'_i \setminus C_i$ is approved by at most one agent, $|C'_i| \geq (n+1) > k$, in every level, which contradicts our committee constraint. This shows that $score(C_i) \geq x$ for each committee $C_i \in \mathcal{C}$. Next assume there exists an agent a_i with less then y approved candidates in the sequence. This means that for every agent $a_{i,j}$, $j \in \{1, \ldots, (n+1) \cdot \tau\}$, one of his approved candidates has to be in $C'_i \setminus C_i$. Due to our construction we know that each agent approves of its individual candidate, thus we need to elect $(n+1) \cdot \tau$ different candidates. This leads to a contradiction, since we elect τ committees of size k and k is upperbounded by n, allowing us to elect at most $\tau \cdot n$ candidates. It follows that every agent a_i has y approved candidates. Since all our constraints have to be satisfied, it follows that I is a yes-instance for $(\geq, \geq)$ BICMCE WITH $\leq$ 1-APPROVAL VOTING. □

Proposition 4.2. *$(\geq, \geq)$ BICMCE WITH $=$ 1-APPROVAL VOTING is NP-hard even for constant y and k and each agent approves the same candidate over all levels.*

In the following proof we give a polynomial-time many-one reduction from the following strongly NP-hard [GJ82] Problem:

Problem 11: 3-PARTITION
Input: A multiset $S = \{s_1, \ldots, s_n\}$ of $n = 3 \cdot m$ positive integers. The sum of all integers is $m \cdot T$.
Question: Is there a partition of S into m triplets $S_1, S_2, \ldots, S_m$ such that every subset sums up to T?

Proof. Let $I = S$ be an instance of 3-PARTITION, and let S be a multiset of $n = 3m$ positive integers. We construct an instance $I' = (A, C, u, k, x, y)$ of $(\geq, \geq)$ BICMCE WITH $=$ 1-APPROVAL VOTING in polynomial time as follows.
Construction: Create the set C of candidates by creating $3m$ candidates c_i, each representing one integer in S. For each candidate i we create a set A_i of voters, containing s_i voters a_i^ℓ, $\ell \in \{0, \ldots, s_i - 1\}$. Build the set $A \coloneqq A_1 \cup ... \cup A_{3m}$ of voters. Next, construct the voting profiles as follows. In each time step candidate c_i is approved by every agent in A_i. So if c_i is elected, $|A_i| = s_i$ will be added to the score. Finally, set $x \coloneqq T$, $k \coloneqq 3$ and $y \coloneqq 1$. This finishes the construction.
Running time: Since our number of candidates is equal to the size of our integer set, creating the set can be done in polynomial time. For our agents we create s_i agents for the candidate representing integer s_i. Denoting s' to be the biggest integer in S, this can be constructed in $\mathcal{O}(s' \cdot 3m)$ which since $s' < T$ makes it polynomial.
Correctness: We claim that I is a yes-instance if and only if I' is a yes-instance.

($\Rightarrow$) Let $S_1, S_2, \ldots, S_m$ be a solution to I such that the sum of the numbers in each triplet is equal to T and that they must form a partition of S. That means each subset contains three integers and each triplet sums up to T. We now have to show that $(C_1, \ldots, C_m)$ is a solution of I', each candidate in C_i representing one integer in S_i. We know that we have at most three candidates in each committee and the score for each committee is equal to $T = x$, since every candidate is approved by the same number of agents to the integer it is representing. Now we still have to make sure every agent has at least one candidate he favours in one of these committees. Due to the fact that $S_1, S_2, \ldots, S_m$ is a partition of S, we know each candidate is in exactly one committee.

So every agent will find its candidate in exactly one committee. Hence, $(C_1, \dots, C_m)$ is a solution to I'.

($\Leftarrow$) Let $(C_1, C_2, \dots, C_m)$ be a solution of I'. We claim that $(S_1, S_2, \dots, S_m)$ is solution of I. S_i contains the integers each candidate in C_i is representing. We know that every committee contains at most 3 candidates, each representing a positive integer, with a score of at least x. Since the score of a committee is the number of agents approving this committee and each candidate c_i is approved by x agents, each set S_i sums up to at least T. Now what we have to prove is that those sets are disjoint and cover S. Due to our construction we can see that we have $3m$ sets of agents, each set approving a different candidate. We know that since we have m committees of size ≤ 3, we have a total of $3m$ candidates over time. Following that up, we now have $3m$ different candidates each preferred by a number of different agents and at most $3m$ slots that can be chosen. Due to the constraint $y = 1$, we know that each agent has to have a candidate he likes during this time period. Which means every candidate will be in exactly one committee and every committee contains 3 candidates. So this is indeed a partition of our candidates in sets of exactly 3, since every candidate represents one integer in S. Furthermore this tells us that every set S_i sums up to exactly T. It follows that I is a `yes`-instance for 3-Partition. □

Theorem 4.3. *$(\geq, \geq)$* BICMCE with $=$ 1-Approval Voting *is NP-hard, even with three candidates, $k = 2$, and $x = 5$.*

In the following proof we give a polynomial-time many-one reduction from 3-SAT (Problem 6) where every variable occurs at most 4 times, which is still NP-hard [Dub90]:

Proof. Let $I = (X, \varphi)$ be an instance of 3-SAT where every variable occurs at most 4 times with variable set $X = \{x_1, \dots, x_N\}$ and formula $\varphi = \bigwedge_{i=1}^{M} \ell_i$, where $K_i = (\ell_1 \vee \ell_2 \vee \ell_3)$. We construct an instance $I' = (A, C, u, k, x, y)$ of $(\geq, \geq)$ BICMCE with $=$ 1-Approval Voting in polynomial time as follows.
Construction: Assume $M \geq 9$ (since the problem can be solved in polynomial time when M is constant). We start by creating the set C of candidates by creating 3 candidates, $c_\top$, $c_\bot$ and c^*. Let $\tau = N$ and for every clause $K_i = (\ell_1 \vee \ell_2 \vee \ell_3)$ in φ' we create an agent a_i that approves $c_\top$ in level j, $j \in \{1, 2, 3\}$, if ℓ_j is a positive literal and $c_\bot$ if it is a negative literal. If a clause contains the positive as well as the negative literal, then a_i approves of c^*. In every other layer a_i approves of candidate c^*. Finally, let $x := 5$, $k := 2$ and $y := N - 3$. This finishes the construction.
Running time: We create the voting profiles in $\mathcal{O}(M \cdot N)$ time. The rest of our construction can be done in constant time, which makes the complete construction polynomial.
Correctness: We claim that I is a `yes`-instance if and only if I' is a `yes`-instance.

($\Rightarrow$) Let f be a satisfying truth assignment to I. We claim that $(C_1, \dots, C_N)$ is a solution of I', where the candidates chosen in C_i are c^* and $c_\top$ if $f(x_i) = \top$ or c^* and $c_\bot$ if $f(x_i) = \bot$. We now have to prove that for every agent a_i, $\sum_{t=1}^{\tau} |u_t(a_i) \cap C_t| \geq N - 2$. Assume this is not the case, we know that a_i approves of c^* in $N-3$ levels, which would lead to a score of at least $N-3$. Due to our construction that would mean that K_i is not satisfied by f, but since f is a satisfying truth assignment, the corresponding agent will have at least one other candidate approved in our committee sequence. We also know that for every committee C_i, $score(C_i) \geq 5$, since $\max(M-4, 5)$ agents will have voted for c^*. It follows that $(C_1, \ldots, C_N)$ is a solution to I'.

($\Leftarrow$) Let $C = (C_1, C_2, \ldots, C_N)$ be a solution of I'. We claim that f is solution to I, where $f : X \to \{\bot, \top\}$ and $f(x_i) := \top$ if the candidate chosen in level i is $c_\top$, or no candidate is chosen and $f(x_i) = \bot$ if the candidate chosen in level i is $c_\bot$. Assume that both candidates are chosen in one level, this would lead to a score of 4, not satisfying the constraint of $score(C_i) \geq 5$, which would mean that C is not a solution to I'. Next assume this truth assignment does not satisfy φ. Then there exists a clause K that is not satisfied by f. Due to our construction the corresponding agent a_K will therefore only have $N-3$ approved candidates in $(C_1, C_2, \ldots, C_N)$, which as a consequence would not be a solution to I'. Furthermore every clause that has both a positive and negative literal of a variable will be satisfied no matter the truth assignment. It follows that I is a `yes`-instance for 3-SAT.

It follows that $(\geq, \geq)$ BICMCE WITH $=1$-APPROVAL VOTING is NP-hard even for a constant number of candidates. □

Observation 4.4. *The reduction also works for $x = M - 4$.*

From Observation 4.4 it follows that:

Proposition 4.5. *$(\geq, \geq)$* BICMCE WITH $=1$-APPROVAL VOTING *is NP-hard, even if $n - x$ is constant.*

4.2 Approval

Problem 12: $(\geq, \geq)$ BICMCE WITH APPROVAL VOTING

Input: A set $A = \{a_1, \ldots, a_n\}$ of agents, a set C of candidates, voting profiles $u_t : A \to \mathcal{P}(C)$ for $t \in \{1, \ldots, \tau\}$, $\tau \in \mathbb{N}$, and integers $x, y, k \in \mathbb{N}_0$.

Question: Is there a sequence $(C_1, \ldots, C_\tau)$ of candidate subsets $C_t \subseteq C$, with $|C_t| \leq k$, for all $t \in \{1, \ldots, \tau\}$ such that:

$$\text{for all layers } t \in \{1, \ldots, \tau\} : \quad \sum_{a_i \in A} \min(|u_t(a_i) \cap C_t|, 1) \geq x$$

$$\text{and for all agents } a_i \in A : \quad \sum_{t=1}^{\tau} \min(|u_t(a_i) \cap C_t|, 1) \geq y$$

Theorem 4.6. $(\geq, \geq)$ BICMCE WITH APPROVAL VOTING *is NP-hard, even for constant x, y, and τ.*

Proposition 4.7. $(\geq, \geq)$ BICMCE WITH APPROVAL VOTING *is W[2]-hard when parameterized by $\tau + k$.*

In the following proof of Theorem 4.6 and Proposition 4.7 we give a polynomial-time many-one reduction from the following NP-hard [Cyg+15] and W[2]-hard [DF13] Problem:

Problem 13: HITTING SET
Input: A collection $\mathcal{H}$ of subsets of a finite set S and a positive integer r.
Question: Is there a subset $S' \subseteq S$ with $|S'| \leq r$ such that S' contains at least one element from each subset in $\mathcal{H}$?

Proof. Let $I = (\mathcal{H}, S, r)$ be an instance of HITTING SET. We construct an instance $I' = (A, C, u, k, x, y)$ of $(\geq, \geq)$ BICMCE WITH APPROVAL VOTING in polynomial time as follows.
Construction: Set the set C of candidates equal to S and create $|\mathcal{H}|$ different agents to build the set A. Next, construct $|\mathcal{H}|$ voting profiles as follows. For each set agent a_H with $H = \{s_1, \ldots, s_d\}$ set $u_t(a_H) = \{c_{s_1}, \ldots, c_{s_d}\}$. Finally, let $k := r$, $\tau := 1$, $y := 1$ and $x = 1$. This finishes the construction.
Running time: Construction is possible in $d \cdot |\mathcal{H}| + |S|$ so $\mathcal{O}(|\mathcal{H}| + |S|)$ time, which is polynomial.
Correctness: We claim that I is a `yes`-instance if and only if I' is a `yes`-instance.

($\Rightarrow$) Let S' be a solution to I, that is $S' \subseteq S$ is of size at most r such that S' contains at least one element from each subset in $\mathcal{H}$. We now have to show that (C_1) is a solution of I', where C_1 contains all corresponding candidates to S'. So we have a committee of size r. Now we have to make sure that every agent has at least one approved candidate in this sequence. Since we know that S' contains at least one element of each set, it follows that our committee contains at least one of the approved candidates. Hence, (C_1) is a solution to I'.

($\Leftarrow$) Let (C_1) be a solution to I'. We claim that $S' = \{S_i \mid c_{S_i} \in C_1\}$ is a hitting set $\mathcal{H}$. We know that the committee contains at most r candidates, so S' contains at most r elements. Assume S' does not cover every set in $\mathcal{H}$, then there exists a set H that is not covered by S'. The corresponding set agent a_H will therefore not have an approved candidate in C_1, meaning it is not a solution to I'. It follows that I is a `yes`-instance of HITTING SET. □

4.3 Borda Rule

Problem 14: $(\geq, \geq)$ BICMCE WITH BORDA RULE
Input: A set A $= \{a_1, \ldots, a_n\}$ of agents, a set C of candidates, ranking profiles $p_t = \{\succ_{t,1}, \ldots, \succ_{t,n}\}$ for $t \in \{1, \ldots, \tau\}$, $\tau \in \mathbb{N}$, and integers $x, y, k \in \mathbb{N}_0$.
Question: Is there a sequence $(C_1, \ldots, C_\tau)$ of candidate subsets $C_t \subseteq C$, with $|C_t| \leq k$, for all $t \in \{1, \ldots, \tau\}$ such that:

$$\text{for all layers } t \in \{1, \ldots, \tau\}: \quad \sum_{c \in C_t} \sum_{a_i \in A} |\{c' \in C \mid c \succ_{t,i} c'\}| \geq x$$

$$\text{and for all agents } a_i \in A: \quad \sum_{t=1}^{\tau} \max_{c \in C_t} |\{c' \in C \mid c \succ_{t,i} c'\}| \geq y$$

Theorem 4.8. $(\geq, \geq)$ BICMCE WITH BORDA RULE *is NP-hard.*

In the following proof we give a polynomial-time many-one reduction from the following NP-hard [GJ82] Problem:

Problem 15: 3-Hitting Set
Input: A collection $\mathcal{H}$ of subsets of size three of a finite set S and a positive integer r.
Question: Is there a subset $S' \subseteq S$ with $|S'| \leq r$ that allows S' to contain at least one element from each subset in $\mathcal{H}$?

Proof. Let $I = (\mathcal{H}, S, r)$ be an instance of 3-Hitting Set. We construct an instance $I' = (A, C, u, k, x, y)$ of $(\geq, \geq)$ BICMCE with Borda Rule in polynomial time as follows.
Construction: Set the set C of candidates equal to S and create $|\mathcal{H}|$ different agents to build the set A. Next, construct the ranking profiles as follows. Agent i will rank the candidates contained in H_i the highest ($H_i^1 \succ_{1,i} H_i^2 \succ_{1,i} H_i^3 \succ_{1,i} \dots$), followed by the rest of the candidates in arbitrary order. Finally, set $k = r$, $\tau = 1$, $y = |S| - 2$ and $x = 0$. This finishes the construction.
Complexity: Constructing the agent and candidate sets is possible in $O(2 \cdot |S| + |\mathcal{H}|)$ time and constructing the voting profiles can be done in $O(3 \cdot |\mathcal{H}|^2)$ time. Therefore our construction is polynomial.
Correctness: We claim that I is a `yes`-instance if and only if I' is a `yes`-instance.
($\Rightarrow$) Let S' be a solution to I, that is, $S' \subseteq S$ is of size at most r and S' contains at least one element from each set in $\mathcal{H}$. We now have to show that (C_1) is a solution of I', where C_1 contains the corresponding candidates to S'. So we have a Committee of size r. Now we have to make sure that every agent has a score of at least $|S| - 2$. Since we know that S' contains one element of every subset in $\mathcal{H}$, every agent will get one of his 3 top choices. In the worst case this would mean he gets a score of $|S| - 2$. Hence, (C_1) is a solution to I'.
($\Leftarrow$) Let (C_1) be a solution to I'. We claim that $S' = C_1$ is a 3-Hitting Set of $\mathcal{H}$. We know that our committee contains at most r candidates, so $|S'| \leq r$. Furthermore we know that each agent has a score of at least $|S| - 2$ which means one of their three top choices has to be in the chosen committee. Due to our construction we know that these top choices represent the three elements in one of the sets of $\mathcal{H}$. So there has to be at least one representing candidate of an element of each subset in our set. From this we can say that one element of each subset has to be in our committee sequence, meaning also in S'. It follows that I is a `yes`-instance for 3-Hitting Set.

It follows that $(\geq, \geq)$ BICMCE with Borda Rule is NP-hard. □

Chapter 5

Conclusion

In this thesis, we analyzed the algorithmic complexity of $(\sim_x, \sim_y)$ BICMCE with R with different voting rules, focusing on $(\geq, \geq)$ BICMCE with $\leq$ 1-Approval Voting. Although our problem with plurality voting is polynomial-time solvable when $\tau = 1$, we have seen that as soon as we add only 1 level to it, it becomes NP-hard. For general approval voting and ordinal voting, we have seen that our problem is NP-hard even without the aspect of multiple levels. It would be interesting to further analyze this by studying $(\sim_x, \sim_y)$ BICMCE with other voting rules, for example with score voting, where the agents give each candidate a score and the candidates with the highest scores are elected. In particular expanding our work on ordinal voting could be of further interest due to the fact that how we evaluate the y-score is unclear and there are a lot of elections relying on ordinal voting methods.

For $(\geq, \geq)$ BICMCE with $\leq$ 1-Approval Voting we have seen that even in an election where every agent can choose between only two candidates our problem is NP-hard. This is interesting for real-life scenarios, since situations like this can be found frequently in political elections.

Our results also show that if we only have a small number of agents, then computing a solution might not take as long, which implies that the number of agents plays a significant role in our problems complexity. It would be interesting to build up on this, by resolving our open questions, for example if $(\geq, \geq)$ BICMCE with $\leq$ 1-Approval Voting parameterized by n is in FPT or W[1]-hard. It would also be of interest to see how this might differ for $(\geq, \geq)$ BICMCE with $=$ 1-Approval Voting, since our current results show that although an agent can not abstain of the vote, the complexity stays almost the same for our different parameters.

Another interesting insight from our result is that our agent score constraint plays a more important role in the complexity of our problem than our committee score constraint: Our problem stays NP-hard even if $x = 0$ but becomes polynomial time solvable if $y = 0$. This indicates that providing fairness is a lot more challenging then pleasing a certain number of agents. If y is a lot smaller then τ, some agents might have a great advantage in the overall sequence. Therefore another aspect could be to study how our complexity changes if our score constraints would be set to exactly x or exactly y. Note that for our score constraint set to exactly y we proved $(\geq, \geq)$ BICMCE with $\leq$ 1-Approval Voting to remain NP-hard (see Proposition 4.2). This might also be interesting in the real world because our discussed model can be very disproportionate, leading to unfair results.

Literature

[Azi+15] Haris Aziz, Serge Gaspers, Joachim Gudmundsson, Simon Mackenzie, Nicholas Mattei, and Toby Walsh. *Computational Aspects of Multi-Winner Approval Voting*. In: *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems (AAMAS '15)*. International Foundation for Autonomous Agents and Multiagent Systems, 2015, pp. 107–115 (cit. on p. 11).

[Bar+15] Allison K. Barner, Jane Lubchenco, Christopher Costello, Steven D. Gaines, Amanda Leland, Brett Jenks, Steven Murawski, Eric Schwaab, and Margaret Spring. *Solutions for Recovering and Sustaining the Bounty of the Ocean: Combining Fishery Reforms, Rights-Based Fisheries Management, and Marine Reserves*. In: *Oceanography* 28.2 (2015), pp. 252–263 (cit. on p. 10).

[BFK20] Robert Bredereck, Till Fluschnik, and Andrzej Kaczmarczyk. *Multistage Committee Election*. 2020 (cit. on p. 11).

[BJK14] Hans L. Bodlaender, Bart M. P. Jansen, and Stefan Kratsch. *Kernelization Lower Bounds by Cross-Composition*. In: *SIAM Journal on Discrete Mathematics* 28.1 (2014), pp. 277–305 (cit. on p. 25).

[BKN20] Robert Bredereck, Andrzej Kaczmarczyk, and Rolf Niedermeier. *Electing Successive Committees: Complexity and Algorithms*. In: *Proceedings of the 34st AAAI Conference on Artificial Intelligence (AAAI'20)*. Association for the Advancement of Artificial Intelligence, 2020, pp. 1846–1853 (cit. on p. 11).

[BN21] Niclas Boehmer and Rolf Niedermeier. *Broadening the research agenda for computational social choice: Multiple preference profiles and multiple solutions*. In: *Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS '21)*. 2021, pp. 1–5 (cit. on p. 11).

[Coo71] Stephen A. Cook. *The Complexity of Theorem-Proving Procedures*. In: *Proceedings of the Third Annual ACM Symposium on Theory of Computing (STOC'71)*. Association for Computing Machinery, 1971, pp. 151–158 (cit. on p. 21).

[Cyg+15] Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, and Saket Saurabh. *Parameterized Algorithms*. Springer, 2015 (cit. on pp. 15, 37).

[DF13] Rodney G. Downey and Michael R. Fellows. *Fundamentals of Parameterized Complexity*. Springer, 2013 (cit. on p. 37).

[Dub90] Olivier Dubois. *On the r,s-SAT satisfiability problem and a conjecture of Tovey*. In: *Discrete Applied Mathematics* 26.1 (1990), pp. 51–60 (cit. on p. 36).

[Fal+17] Piotr Faliszewski, Piotr Skowron, Arkadii Slinko, and Nimrod Talmon. *Multiwinner voting: A new challenge for social choice theory*. In: *Trends in computational social choice* 74.2017 (2017), pp. 27–47 (cit. on p. 11).

[Fel+09] Michael R. Fellows, Danny Hermelin, Frances Rosamond, and Stéphane Vialette. *On the parameterized complexity of multiple-interval graph problems*. In: *Theoretical Computer Science* 410.1 (2009), pp. 53–61 (cit. on pp. 15, 19, 30).

[FG04] Jörg Flum and Martin Grohe. *The Parameterized Complexity of Counting Problems*. In: *SIAM Journal on Computing* 33.4 (2004), pp. 892–922 (cit. on p. 15).

[FN01] Henning Fernau and Rolf Niedermeier. *An Efficient Exact Algorithm for Constraint Bipartite Vertex Cover*. In: *Journal of Algorithms* 38.2 (2001), pp. 374–410 (cit. on p. 18).

[GJ82] Michael R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. In: *SIAM Review* 24.1 (1982), pp. 90–2 (cit. on pp. 22, 35, 39).

[Gra+06] R. Quentin Grafton, Ragnar Arnason, Trond Bjørndal, David Campbell, Harry F. Campbell, Colin W. Clark, Robin Connor, Diane P. Dupont, Rögnvaldur Hannesson, Ray Hilborn, James E. Kirkley, Tom Kompas, Daniel E. Lane, Gordon R. Munro, Sean Pascoe, Dale Squires, Stein Ivar Steinshamn, Bruce R. Turris, and Quinn Weninger. *Incentive-based approaches to sustainable fisheries*. In: *Canadian Journal of Fisheries and Aquatic Sciences* 63.3 (2006), pp. 699–710 (cit. on p. 10).

[GTW14] Anupam Gupta, Kunal Talwar, and Udi Wieder. *Changing Bases: Multistage Optimization for Matroids and Matchings*. In: *Automata, Languages, and Programming (ICALP'14)*. Springer Berlin Heidelberg, 2014, pp. 563–575 (cit. on p. 11).

[IP01] Russell Impagliazzo and Ramamohan Paturi. *On the Complexity of k-SAT*. In: *Journal of Computer and System Sciences* 62.2 (2001), pp. 367–375 (cit. on pp. 21, 22).

[Kom+19] Christian Komusiewicz, André Nichterlein, Rolf Niedermeier, and Marten Picker. *Exact algorithms for finding well-connected 2-clubs in sparse real-world graphs: Theory and experiments*. In: *European Journal of Operational Research* 275.3 (2019), pp. 846–864 (cit. on p. 21).

[Nie06] Rolf Niedermeier. *Invitation to Fixed-Parameter Algorithms*. Oxford University Press, 2006 (cit. on p. 15).

[Wam00] Brian Wampler. *A guide to participatory budgeting*. International Budget Partnership, 2000 (cit. on p. 11).