Multiwinner Elections with Diversity Constraints

Robert Bredereck, Piotr Faliszewski,
Ayumi Igarashi, Martin Lackner, Piotr Skowron

Abstract

We develop a model of multiwinner elections that combines performance-based measures of the quality of the committee (such as, e.g., Borda scores of the committee members) with diversity constraints. Specifically, we assume that the candidates have certain attributes (such as being a male or a female, being junior or senior, etc.) and the goal is to elect a committee that, on the one hand, has as high a score regarding a given performance measure, but that, on the other hand, meets certain requirements (e.g., of the form "at least 30% of the committee members are junior candidates and at least 40% are females"). We analyze the computational complexity of computing winning committees in this model, obtaining polynomial-time algorithms (exact and approximate) and NP-hardness results. We focus on several natural classes of voting rules and diversity constraints.

1 Introduction

We study the problem of computing committees (i.e., sets of candidates) that, on the one hand, are of high quality (e.g., consist of high-performing individuals) and that, on the other hand, are diverse (as specified by a set of constraints). The following example shows our problem in more concrete terms.

Consider an organization that wants to hold a research meeting on some interdisciplinary topic such as, e.g., "AI and Economics." The meeting will take place in some secluded location and only a certain limited number of researchers can attend. How should the organizers choose the researchers to invite? If their main criterion were the number of highly influential AI/economics papers that each person published, then they would likely end up with a very homogeneous group of highly-respected AI professors. Thus, while this criterion definitely should be important, the organizers might put forward additional constraints. For example, they could require that at least 30% of the attendees are junior researchers, at least 40% are female, at least a few economists are invited (but only senior ones), the majority of attendees work on AI, and the attendees come from at least 3 continents and represent at least 10 different countries.[1] In other words, the organizers would still seek researchers with high numbers of strong publications, but they would give priority to making the seminar more diverse (indeed, junior researchers or representatives of different subareas of AI can provide new perspectives; it is also important to understand what people working in economics have to say, but the organizers would prefer to learn from established researchers and not from junior ones).

The above example shows a number of key features of our committee-selection model. First, we assume that there is some function that evaluates the committees (we refer to it as the *objective function*). In the example it was (implicitly) the number of high-quality papers that the members of the committee published. In other settings (e.g., if we were shortlisting job candidates) these could be aggregated opinions of a group of voters (the recruitment committee, in the shortlisting example).

[1]For example, the Leibniz-Zentrum für Informatik that runs Dagstuhl Seminars gives similar suggestions to event organizers.

Second, we assume that each prospective committee member (i.e., each researcher in our example) has a number of attributes, which we call labels. For example, a researcher can be *junior* or *senior*, a *male* or a *female*, can *work in AI* or in *economics* or in some other area, etc. Further, the way in which labels are assigned to the candidates may have a structure on its own. For example, each researcher is either male or female and either junior or senior, but otherwise these attributes are independent (i.e., any combination of gender and seniority level is possible). Other labels may be interdependent and may form hierarchical structures (e.g., every researcher based in Germany is also labeled as representing Europe). Yet other labels may be completely unstructured; e.g., researchers can specialize in many subareas of AI, irrespective how (un)related they seem.

Third, we assume that there is a formalism that specifies when a committee is *diverse*. In principle, this formalism could be any function that takes a committee and gives an *accept/reject* answer. However, in many typical settings it suffices to consider simple constraints that regard each label separately (e.g., "at least 30% of the researchers are junior" or "the number of male researchers is even"). We focus on such independent constraints, but studying more involved ones, that regard multiple labels (e.g., "all invited economists must be senior researchers") would also be interesting.

Our goal is to find a committee of a given size k that is diverse and has the highest possible score from the objective function. While similar problems have already been considered (see the Related Work section), we believe that our paper is the first to systematically study the problem of selecting a diverse committee, where diversity is evaluated with respect to candidate attributes. We provide the following main contributions:

1. We formally define the general problem of selecting a diverse committee and we provide its natural restrictions. Specifically, we focus on the case of submodular objective functions (with the special case of separable functions), candidate labels that are either layered or laminar,[2] and constraints that specify sets of acceptable cardinalities for each label independently (with the special case of specifying intervals of acceptable values).

2. We study the complexity of finding a diverse committee of a given size, depending on the type of the objective function, the type of the label structure, and the type of diversity constraints. While in most cases we find our problems to be NP-hard (even if we only want to check if a committee meeting diversity constraints exists; without optimizing the objective function), we also find practically relevant cases with polynomial-time algorithms (e.g., our algorithms would suffice for the research-meeting example restricted to the constraints regarding the seniority level and gender). We provide approximation algorithms for some of our NP-hard problems.

3. We study the complexity of recognizing various types of label structures. For example, given a set of labeled candidates, we ask if their labels have laminar or layered structure. It turns out that recognizing structures with three independent sets of labels is NP-hard, whereas recognizing up to two independent sets is polynomial-time computable.

4. Finally, we introduce the concept of *price of diversity*, which quantifies the "cost" of introducing diversity constraints subject to the assumed objective function.

Our main results are presented in Table 1.

[2] If we restricted our example to labels regarding gender and seniority level, we would have 2-layered labels (because there are two sets of labels, $\{male, female\}$ and $\{junior, senior\}$, and each candidate has one label from each set. On the other hand, hierarchical labels, such as those regarding countries and continents, are 1-laminar (see description of the model for more details).

2 The Model

For $i, j \in \mathbb{N}$, we write $[i, j]$ to denote the set $\{i, i+1, \ldots, j\}$. We write $[i]$ as an abbreviation for $[1, i]$. For a set X, we write 2^X to denote the family of all of its subsets. We first present our model in full generality and then describe the particular instantiations that we focus on in our analysis.

General Model Let $C = \{c_1, \ldots, c_m\}$ be a set of candidates and let L be a set of labels (such as *junior*, *senior*, etc.). Each candidate is associated with a subset of these labels through a labeling function $\lambda\colon C \to 2^L$. We say that a candidate c has label ℓ if $\ell \in \lambda(c)$, and we write C_ℓ to denote the set of all candidates that have label ℓ.

A diversity specification is a function that given a committee (i.e., a set of candidates), the set of labels, and the labeling function provides a *yes/no* answer specifying if the committee is *diverse*. If a committee is diverse with respect to diversity specification D, then we say that it is D-*diverse*.

An objective function $f\colon 2^C \to \mathbb{R}$ is a function that associates each committee with a score. We assume that $f(\emptyset) = 0$ and that the function is monotone (i.e., for each two committees A and B such that $A \subseteq B$, it holds that $f(A) \leq f(B)$). In other words, an empty committee has no value and extending a committee cannot hurt it.

Our goal is to find a committee of a given size k that meets the diversity specification and that has the highest possible score according to the objective function.

Definition 2.1 (Diverse Committee Winner Determination (DCWD))**.** Given a set of candidates C, a set of labels L, a labeling function λ, a diversity specification D, a desired committee size k, and an objective function f, find a committee $W \subseteq C$ with $|W| = k$ that achieves the maximum value $f(W)$ among all D-diverse size-k committees.

The model, as specified above, is far to general to obtain any sort of meaningful computational results. Below we specify its restrictions that we study.

Objective Functions An objective function is submodular if for each two committees S and S' such that $S \subseteq S' \subseteq C$ and each $c \in C \setminus S'$ it holds that $f(S \cup \{c\}) - f(S) \geq f(S' \cup \{c\}) - f(S')$. For two sets of candidates X and S, we write $f(X|S)$ to denote the marginal contribution of the candidates from X with respect to those in S. Formally, we have $f(X|S) = f(S \cup X) - f(S)$. Submodular functions are very common and suffice to express many natural problems. We assume all our objective functions to be submodular.

Example 2.2. Consider the following voting scenario. Let $C = \{c_1, \ldots, c_m\}$ be a set of candidates and $V = \{v_1, \ldots, v_n\}$ a set of voters, where each voter ranks all the candidates from best to worst. We write $\mathrm{pos}_{v_i}(c)$ to denote the position of candidate c in the ranking of voter v_i (the best candidate is ranked on position 1, the next one on position 2, and so on). The Borda score associated with position i (among m possible ones) is $\beta_m(i) = m - i$. Under the Chamberlin–Courant rule (CC), the score of a committee S is defined by objective function $f^{\mathrm{CC}}(S) = \sum_{i=1}^{n} \beta_m(\min\{\mathrm{pos}_{v_i}(c) \mid c \in S\})$. Intuitively, this function associates each voter with her representative (the member of the committee that the voter ranks highest) and defines the score of the committee as the sum of the Borda scores of the voters' representatives. It is well-known that this function is submodular [28]. The CC rule outputs those committees (of a given size k) for which the CC objective function gives the highest value (and, intuitively, where each voter is represented by a committee member that the voter ranks highly).

As a special case of submodular functions, we also consider *separable* functions. A function is separable if for every candidate $c \in C$ there is a weight w_c such that the value of a committee S is given as $f(S) = \sum_{c \in S} w_c$. While separable functions are very restrictive, they are also very natural.

Example 2.3. Consider the setting from Example 2.2, but with objective function $f^{\mathrm{kB}(W)} = \sum_{i=1}^{n} \left(\sum_{c \in W} \beta_m(\mathrm{pos}_{v_i}(c)\right)$. This function sums Borda scores of all the committee members from all the voters and models the k-Borda voting rule (the committee with the highest score is selected). The function is separable as for each candidate c it suffices to take $w_c = f^{\text{k-Borda}}(\{c\})$. It is often argued that k-Borda is a good rule when our goal is to shortlist a set of individually excellent candidates [14].

Together, Example 2.2 and Example 2.3 show that our model suffices to capture many well-known multiwinner voting scenarios. Many other voting rules, such as Proportional Approval Voting, or many committee scoring rules, can be expressed through submodular objective functions [36, 13].

Diversity Specifications We focus on diversity specifications that regard each label independently. In other words, the answer to the question if a given committee S is diverse or not depends only on the cardinalities of the sets $C_\ell \cap S$.

Definition 2.4. For a set of candidates C, a set of labels L, and a labeling function λ, we say that a diversity specification D is independent (consists of independent constraints) if and only if there is a function $b\colon L \to 2^{[|C|]}$ (referred to as the cardinality constraint function) such that a committee S is diverse exactly if for each label ℓ it holds that $|S \cap C_\ell| \in b(\ell)$.

If we have m candidates then specifying independent constraints requires providing at most $m+1$ numbers for each label. Thus independent constraints can easily be encoded in the inputs for our algorithms.

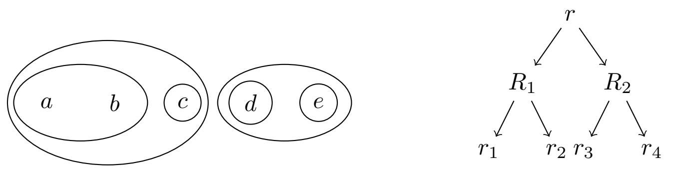
Independent constraints are quite expressive. For example, they are sufficient to express conditions such as "the committee must contain an even number of junior researchers" or, since our committees are of a given fixed size, conditions of the form "the committee must contain at least 40% females." Indeed, the conditions of the latter form are so important that we consider them separately.

Definition 2.5. For a set of candidates C, a set of labels L, and a labeling function λ, we say that a diversity specification D is interval-based (consists of interval constraints) if and only if there are functions $b_1, b_2\colon L \to 2^{[|C|]}$ (referred to as the lower and upper interval constraint functions) such that a committee S is diverse if and only if for each label ℓ it holds that $b_1(\ell) \leq |S \cap C_\ell| \leq b_2(\ell)$.

Label Structures In principle, our model allows each candidate to have an arbitrary set of labels. In practice, there usually are some dependencies between the labels and these dependencies can have strong impact in the complexity of our problem. We focus on labels that are arranged in independent, possibly hierarchically structured, layers.

Let C be a set of candidates, let L be a set of labels, and let λ be a labeling function. We say that λ has 1-*layered* structure (i.e., we have a 1-layered labeling) if for each two distinct labels x, y it holds that $C_x \cap C_y = \emptyset$ (i.e., each candidate has at most one of these labels). For example, if we restricted the example from the introduction to labels regarding the seniority level (junior or senior), then we would have a 1-layered labeling.

More generally, we say that a labeling is 1-*laminar* if for each two distinct labels x, y we have that either (a) $C_x \cap C_y = \emptyset$ or (b) $C_x \subseteq C_y$ or (c) $C_y \subseteq C_x$. In other words, 1-laminar labellings allow the labels to be arranged hierarchically.



(a) 1-laminar labeling (b) Tree representation

Figure 1: Illustration of a 1-laminar labeling structure.

Example 2.6. Consider a set $C = \{a, b, c, d, e\}$ of five candidates and labels that encode the countries and continents where the candidates come from. Specifically, there are four countries r_1, r_2, r_3, r_4, and two continents R_1 and R_2. The candidates are labeled as follows:

$$\lambda(a) = \{r_1, R_1\}, \quad \lambda(b) = \{r_1, R_1\}, \quad \lambda(c) = \{r_2, R_1\},$$
$$\lambda(d) = \{r_3, R_2\}, \quad \lambda(e) = \{r_4, R_2\}.$$

Figure 1a illustrates the 1-laminar inclusion-wise relations between the labels (there can be more levels of the hierarchy; for example, for each country there could be labels specifying local administrative division).

Every 1-laminar labeling, together with the set of candidates, can be represented as a rooted tree T in the following way: For a pair of distinct labels x, y we create an arc from x to y if $C_x \subsetneq C_y$ and there is no label z such that $C_x \subsetneq C_z \subsetneq C_y$. We add a *root label* r and we impose that each candidate has this label; we add an arc from r to each label without an incoming arc. The resulting digraph T is clearly a rooted tree. See Figure 1b for an illustration.

For each positive integer t, we say that a labeling is *t-layered* (respectively, *t-laminar*) if the set L of labels can be partitioned into sets $L_1, L_2, \ldots, L_t$ such that for each $i \in [t]$, the labeling restricted to the labels from L_i is 1-layered (respective, 1-laminar).

Example 2.7. In the example from the introduction, restricting our attention to candidates' gender and seniority levels, we get a 2-layered labeling structure. If we also consider labels regarding countries and continents, then we get a 3-laminar structure (however, only the geographic labels would be using the full power of laminar labellings).

We assume that when we are given a t-layered (t-laminar) labeling structure, we are also given the partition of the set of labels that defines this structure (in Section 5 we analyze the problem of recognizing such structures algorithmically).

Balanced Committee Model As a very natural special case of our model we considered the problem of computing balanced committees. In this case there are only two labels (e.g., *male* and *female*), each candidate has exactly one label, and the constraint specification is that we need to select exactly the same number of candidates with either label (thus, by definition, the committee must be of an even size).

Computing balanced committees is a very natural problem. For example, seeking gender balance is a common requirement in many settings. In this paper, we seek exact balance (that is, we seek exactly the same number of candidates with either label) but allowing any other proportion would lead to similar results.

3 Separable Objective Functions

Separable objective functions form a simple, but very important special case of our setting. Indeed, such functions are very natural in shortlisting examples, where diversity constraints are used to implement, e.g., affirmative actions or employment-equity laws. We organize our discussion with respect to the type of constraint specifications.

Independent Constraints It turns out that independent constraints are quite difficult to work with. If the labels are 1-laminar then polynomial-time algorithms exist (both for deciding if feasible committees exist and for computing optimal ones), but with 2-layered labellings our problems become NP-hard (recall that t-layered labellings are a special case of t-laminar ones). Our polynomial-time algorithms proceed via dynamic programming and hardness proofs use reductions from Exact 3 Set Cover (X3C).

Theorem 3.1. *Let D be a diversity specification of independent constraints. Suppose that λ is 1-laminar and f is separable. Then,* DCWD *can be solved in $O(|L|^2k^2 + |C|\log|C|))$ time. Moreover,* DCF *can be solved in $O(|L|^2k^2)$ time. If the λ function is 2-layered then both problems are* NP*-hard (even if each candidate has at most two labels, and each label is associated to at most three candidates).*

Given the above hardness results, it is immediate to ask about the parametrized complexity of our problems because in many settings the label structures are very limited (for example, the 2-layered gender/seniority labeling from the introduction contains only 4 labels and already is very relevant for practical applications). Unfortunately, for independent constraints our problems remain hard when parametrized by the number of labels.

Theorem 3.2. *Both* DCF *and* DCWD *problems are $W[1]$-hard with respect to the number of labels $|L|$, even if D is a diversity specification of independent constraints.*

However, not all is lost and sometimes brute-force algorithms are sufficiently effective. For example, if we have a t-layered labeling (where t is a small constant) then each candidate has at most t different labels and it suffices to consider each size-t labeling separately. A brute-force algorithm based on this idea suffices, e.g., for the example from the seniority/specialty labels from the introduction (it would have $O(|C|^4)$ running time, because there are 4 combinations of labels $\{junior, senior\}$ and $\{AI, economics\}$; the algorithm could also deal with non-independent constraints).

Interval constraints Interval constraints are more restrictive than general independent ones, but usually suffice for practical applications and are more tractable. For example, for the case of 1-laminar labellings we give a linear-time algorithm for recognizing if a feasible committee exists (for independent constraints, our best algorithm for this task is quadratic).

Theorem 3.3. *Let D be a diversity specification of interval constraints. If λ is 1-laminar, then* DCF *can be solved in $O(|C| + |L|)$ time.*

For the case of computing the winning committee we no longer obtain a significant speedup from focusing on interval constraints, but we do get a much better structural understanding of the problem. In particular, we can use a greedy algorithm instead of relying on dynamic programming. Briefly put, our algorithm (presented as Algorithm 1) starts with an empty committee and performs k iterations (k is the desired committee size), in each extending the committee with a candidate that increases the score maximally, while ensuring that the committee can still be extended to one that meets the diversity constraints. To show that this greedy algorithm is correct and that it can be implemented efficiently, we use some notions from the matroid theory.

Algorithm 1: Greedy Algorithm 1

notation: $\mathcal{K}_D \neq \emptyset$ is the set of D-diverse, size-k committees, $\overline{\mathcal{K}}_D$ is its lower extension.
input : $f \colon 2^C \to \mathbb{R}$: the objective function,
k: the size of the committee.
output : $W \in \mathcal{K}_D$

set $W = \emptyset$;
while $|W| < k$ **do**
  choose a candidate $y \in C \setminus W$ such that $W \cup \{y\} \in \overline{\mathcal{K}}_D$ with the maximum improvement $f(\{c\}|W)$;
  set $W \leftarrow W \cup \{y\}$;

Formally, a *matroid* is an ordered pair $(C, \mathcal{I})$, where C is some finite set and $\mathcal{I}$ is a family of its subsets (referred to as the *independent sets* of the matroid). We require that (I1) $\emptyset \in \mathcal{I}$, (I2) if $S \subseteq T \in \mathcal{I}$, then $S \in \mathcal{I}$, and (I3) if $S, T \in \mathcal{I}$ and $|S| > |T|$, then there exists $s \in S \setminus T$ such that $T \cup \{s\} \in \mathcal{I}$. The family of maximal (with respect to inclusion) independent sets of a matroid is called its *basis*. Many of our arguments use results from matroid theory, but often used in very different contexts than originally developed. In particular, the next theorem, in essence, translates the results of Yokoi [41] to our setting.

Theorem 3.4. *Let D be a diversity specification of interval constraints. Suppose that λ is a 1-laminar, and f is a separable function given by a weight vector $\boldsymbol{w} \colon C \to \mathbb{R}$. Then,* DCWD *can be solved in $O(k^2|C||L| + |C|\log|C|)$ time.*

Proof. Let $\mathcal{K}_D$ be the set of D-diverse committees of size k, and assume that $\mathcal{K}_D$ is nonempty. For a family of subsets $\mathcal{K}$ of a finite set C, we define its lower extension to be

$$\overline{\mathcal{K}} = \{\, T \mid \exists S \in \mathcal{K} : T \subseteq S \,\}.$$

It follows from the work of Yokoi [41] that if the constraints are given by intervals and $\mathcal{K}_D \neq \emptyset$, then the lower extension $\overline{\mathcal{K}}_D$ of $\mathcal{K}_D$ forms a family of independent sets of some matroid.[3] Thus Algorithm 1 finds an optimal solution $W \in \mathrm{argmax}_{W' \in \overline{\mathcal{K}}_D} f(W')$ (see, e.g., Chapter 13 of Korte and Vygen [22]).

Let W be the committee produced by Algorithm 1. Since W contains k elements, it must belong to $\mathcal{K}_D$ (because all size-k subsets of $\overline{\mathcal{K}}_D$ are elements of $\mathcal{K}_D$). For the same reason, since W maximizes the score among the sets from $\overline{\mathcal{K}}_D$, it must be the case that $W \in \mathrm{argmax}_{W' \in \mathcal{K}_D} f(W')$ and, so, W is a winning committee. Further, [41] showed that checking whether a set $W \cup \{y\}$ belongs to $\overline{\mathcal{K}}_D$ can be efficiently done by maintaining a set $B \in \mathcal{K}_D$ with $W \subseteq B$ and, so, the greedy algorithm runs in polynomial time. $\square$

Unfortunately, the greedy algorithm does not work for more involved labeling structures, but for 2-laminar labellings we can compute winning committees by reducing the problem to the matroid intersection problem [11]. For more involved labeling structures our problems become NP-hard.

Theorem 3.5. *Let D be a diversity specification of interval constraints. Suppose that λ is 2-laminar and f is separable. Then,* DCF *can be solved in $O(k^2|C|^3|L|)$ time, and* DCWD *can be solved in $O(k|C|^3 + k^3|C|^2|L|)$ time.*

[3]These independent sets form a relaxed version of our constraints. However, taking the lower extension does not necessarily ignore the lower bounds. For instance, consider a setting where we want to select a committee of size 5 such that there are exactly three female candidates and at most two male candidates; the corresponding lower extension $\overline{\mathcal{K}}_D$ only includes the sets of female candidates of size at most 3, whereas a male-only committee of size 2 satisfies the upper bounds on the respective number of female/male candidates.

The bound on the number of layers turns out to be necessary: the following theorem shows that finding a D-diverse committee is intractable even with 3-layers.

Theorem 3.6. DCF *is* NP*-hard even if D is a diversity specification of interval constraints and λ is 3-layered.*

Proof. We reduce from 3-DIMENSIONAL MATCHING (3-DM). Given three disjoint sets X, Y, Z of size n and a set $T \subseteq X \times Y \times Z$ of ordered triplets, 3-DM asks whether there is a set of n triplets in T such that each element is contained in exactly one triplet.

Given an instance $((X, Y, Z), T)$ of 3-DM, we create one candidate $t_i = (x_i, y_i, B_i)$ for each $t_i \in T$. The set of labels is given by $L = X \cup Y \cup Z$. Each candidate t_i has exactly three labels $\lambda(t_i) = \{x_i, y_i, B_i\}$. The lower bound $b_1(\ell)$ and the upper bound $b_2(\ell)$ of each label $\ell \in L$ are set to be 1. Lastly, we set $k = n$. It can be easily verified that $W \subseteq T$ is a desired solution for 3-DM if and only if W is a D-diverse committee of size k, namely, $|W| = k$, and

(i) $|C_x \cap W| = 1$ for each $x \in X$,

(ii) $|C_y \cap W| = 1$ for each $y \in Y$, and

(iii) $|C_z \cap W| = 1$ for each $z \in Z$.

□

Nevertheless, if the number of labels is small (i.e., is taken as the parameter from the point of view of parametrized complexity theory) we can compute optimal diverse committees efficiently. The next theorem expresses this formally (note that interval diversity specifications can be phrased as linear programs, but this language allows also some more involved constraints, such as, "at the research meeting the number of *senior* researchers should be larger than the number of *junior* ones, but without taking the *PhD students* into account").

Theorem 3.7. *Let f be separable objective function and let D be a diversity specification which can be expressed through a linear program* LP *with the set of variables $\{x_\ell : \ell \in L\}$ such that $d \in D$ if and only if* LP *instantiated with variables x_ℓ giving the numbers of committee members with labels ℓ is feasible. Then,* DCWD *is in* FPT *with respect to $|L|$.*

4 Submodular Objective Functions

The case of submodular objective functions is computationally far more difficult than that of separable ones. Indeed, even without diversity constraints computing a winning Chamberlin–Courant committee (specified through a submodular objective function) is NP-hard [28] and, in general, the best polynomial-time approximation algorithm for submodular functions is the classic greedy algorithm [29, 15], which achieves the $1 - \frac{1}{e} \approx 0.63$ approximation ratio. [4] Adding diversity constraints makes our problems even more difficult. Nonetheless, we provide a polynomial-time $\frac{1}{2}$-approximation algorithm for the case of interval constraints and 1-laminar labelings.

Theorem 4.1. *Let D be a diversity specification of interval constraints. If λ is* 1*-laminar and f is a monotone submodular function, then Algorithm 1 gives $\frac{1}{2}$-approximation algorithm for* DCWD*.*

[4] This algorithm starts with an empty committee and extends it with candidates one-by-one, always choosing the candidate that increases the objective function maximally.

Algorithm 2: Greedy Algorithm for BCWD

input : $f \colon 2^C \to \mathbb{R}$, $A \subseteq C$ and $B \subseteq C$ where $A \cap B = \emptyset$, $|A| \geq k'$ and $|B| \geq k'$
output: $W \subseteq C$ where $|W \cap A| = |W \cap B| = k'$

```
while |W| < 2k' do
    choose a pair e = {a, b} where a ∈ A \ W and b ∈ B \ W with maximum
      improvement f(e|W);
    set W ← W ∪ e;
```

Balanced Committees For the balanced committee model it is possible to achieve notably stronger results. Since the balanced case is practically relevant from practical standpoint, we provide its simpler definition, renaming it as BCWD.

Definition 4.2 (BCWD)**.** Given a set of candidates C, two subsets $A, B \subseteq C$ such that $A \cap B = \emptyset$ and $A \cup B = C$, a desired committee size $k = 2k'$, and an objective function f, find a committee $W \subseteq C$ that maximizes $f(W)$ and that satisfies $|W \cap A| = |W \cap B| = k'$.

For the case of BCWD, we provide a polynomial-time $1 - \frac{1}{e}$ approximation algorithm. Since this is the best possible approximation ratio for general submodular functions without diversity constraints, it is also the best one for the balanced setting (formally, the results without diversity constraints translate because we could assume that all the candidates with one of the labels have no influence on the objective value and use the remaining ones to model an unconstrained submodular optimization problem). Our algorithm (presented as Algorithm 2) is very similar to the classic greedy algorithm, but it considers candidates in pairs.

Theorem 4.3. *Let f be a monotone submodular function. Algorithm 2 gives $(1 - \frac{1}{e})$-approximation algorithm for* BCWD*.*

We note that Theorem 4.3 is a special case of a much more general result on approximating the Multidimensional Knapsack problem [24], which gives the same approximation ratio even for maximizing monotone submodular functions subject to interval constraints consisting only of upper bounds. Yet, our algorithm is simpler and faster than this general approach.

Theorem 4.3 applies to all submodular functions. However, for some special cases it is possible to achieve much stronger results. For example, for the Chamberlin–Courant function we find a polynomial-time approximation scheme (PTAS).

Theorem 4.4. *For each Chamberlin–Courant function there exists a PTAS for* BCWD*.*

The main idea behind the proof is to use the PTAS of Skowron et al. [38] to compute a committee of size k' and then to complement it so that it satisfies the diversity constraints. The specific nature of the algorithm of Skowron et al. makes it possible to do this efficiently and effectively.

5 Recognizing Structure of the Labels

In this section we ask how difficult it is to recognize a given labeling structure if it is not provided with the problem. While in most cases it is natural to assume that the structure would be provided (as it would be a common knowledge of the society for which we would want to compute the committee), it is interesting to be able to derive it automatically.

In the previous sections we have seen that there usually are polynomial-time algorithms for computing winning committees for 1-laminar labellings and, sometimes, there are such algorithms for 2-laminar ones. However, 3-laminar labellings always lead to NP-hardness results. The same holds for the label-structure recognition problem. There are algorithms that decide if given labellings are 1- or 2-laminar, but recognizing 3-layered ones is NP-hard. In the labeling-recognition problem we are given a set of candidates C, a set of labels L, and a labeling function λ. Our goal is to recognize if λ is t-laminar (or t-layered), for a given t.

Proposition 5.1. *For $t \in \{1, 2\}$ there exists a polynomial-time algorithm for deciding if a given labeling λ is t-laminar. The problem of deciding if a given labeling λ is* 3*-layered is* NP*-hard.*

6 Related Work

Our work touches upon many concepts and, thus, is related to many pieces of research. In this section we briefly mention some of the most relevant ones.

Lang and Skowron [25] considered a model of diversity requirements that closely resembles our interval constraints. There are two main differences between their work and ours:

(i) they do not consider objective functions and

(ii) their input consists of "ideal points" instead of intervals for each label; since there might not exist a committee satisfying such "exact" constraints, they focus on finding committees minimizing a certain distance to the ideal diversity distributions.

If the labels denote party affiliations of the candidates, the diversity constraints are one-layered and form instances for the apportionment problem, where seats in the parliament should be distributed among the parties (see the book of Balinski and Young [1] for an overview of the apportionment problems). Bi-apportionment [2, 3] can be viewed as an extension of the traditional apportionment to the case when the diversity constraints are two-layered. However, in all these settings there is no objective functions, and the goal is only to find a committee satisfying certain label-based constraints. For this reason our paper is even closer the work of Brams [5], who introduces a specific method based on approval voting that takes diversity constraints into account, which are expressed as quotas for each possible *tuple* of labels; Potthoff [31] and Straszak et al. [39] formulated an ILP for this method.

Optimization of a given objective functions due to constraints is a classic problem studied extensively in the literature. For a review of this literature we refer the reader to the book of Korte and Vygen [22]. More specifically, Krause and Golovin [23] provide a comprehensive survey for the case when the optimized function is submodular. For submodular functions different types of general constraints are considered, including matroid and knapsack constraints [9]. A particularly related case is when the constraints are given for the size of the committee (see e.g., the works of Qian et al. [34] and the references inside)—interestingly, this case can be represented in our model, when we assume that there is a single label assigned to each candidate, and the constraints are given for the number of occurrences of this label in the elected committee. Candidates having positive synergies may induce supermodular (instead of submodular) objective functions. We note that constrained maximization of a supermodular function is equivalent to constrained minimization of a submodular function, known to be NP-hard [21].

Our model is related to the Multidimensional Knapsack problem with submodular objective functions [19, 40, 26, 17, 33, 24], but differs in a few important aspects. The two biggest differences are: (i) Multidimensional Knapsack has constraints of the form "no more than value D on dimension i" (dimensions correspond to labels in our work), whereas our constraints can have more structure (specific quantities of a given label, or upper and lower

Rule	Label Structure	Interval Constraints	Independent Constraints
separable	1-laminar	P	P
	2-laminar	P	NP-hard
	3-layer	NP-hard	NP-hard
	few labels	FPT	W[1]-hard
submodular	1-laminar	0.5-approx.	?
	balanced	0.63-approx.	—
CC	balanced	PTAS	—

Table 1: The complexity of computing winning committees for rules of a given type, for the case of candidates with particular label structures, and particular diversity specifications. The complexity results for the problem of testing if a feasible committee exists are the same as those for computing winning committees. “Balanced” label structure refers to the problem of computing balanced committees (thus the case of independent constraints is not defined for this setting).

bounds), (ii) Multidimensional Knapsack has items that can contribute more than a unit weight to a particular dimension, whereas our candidates only have 0/1 contributions. Thus, our problem is more general regarding the constraint specification, but less general regarding the structure of the weights of items.

The complexity of selecting an optimal committee without constraints has been studied extensively. For a general overview of this literature, we point the reader to a chapter by Faliszewski et al. [14]. Perhaps the most attention was dedicated to the study of the Chamberlin–Courant rule [8]. For instance, it is known that this rule is NP-hard to compute [32]. The problem of finding a winning Chamberlin–Courant committee under restricted domains of voters' preferences was further studied by Betzler et al. [4], Yu et al. [42], Elkind and Lackner [12], Skowron et al. [37], and Peters and Lackner [30]. Parametrized complexity of the problem was studied by Betzler et al. [4] and its approximability by Lu and Boutilier [28], Skowron et al. [38] and Skowron and Faliszewski [35].

Finally, we note that Celis et al. [7] very recently and independently introduced a model for diversity constraints (in their paper refered to as *fairness constraints*) that is similar to our model. Their paper contains algorithmic results, which are also applicable in our setting.

7 Conclusion

We studied the problem of selecting a committee of a given size that, on the one hand, would be diverse (according to a given diversity specification) and, on the other hand, would obtain as high an objective value as possible. We present our results in Table 1. We find that in general our problem is computationally hard, but there are many tractable special cases, especially for separable objective functions (which are very useful for shortlisting tasks, where diversity constraints are particularly relevant) and for up to 2-laminar label structures (which means that dealing with two sets of independent, hierarchically arranged labels, is feasible). Our work leads to many open problems. In particular, we barely scratched the surface regarding approximation of our problems, or their parameterized complexity. Experimental studies would be very desirable as well.

References

[1] M. Balinski and H. P. Young. *Fair Representation: Meeting the Ideal of One Man, One Vote*. Yale University Press, 1982. (2nd Edition [with identical pagination], Brookings Institution Press, 2001).

[2] M. L. Balinski and G. Demange. Algorithm for Proportional Matrices in Reals and Integers. *Mathematical Programming, Series A*, 45(1-3):193–210, 1989.

[3] M. L. Balinski and G. Demange. An axiomatic approach to proportionality between matrices. *Mathematics of Operations Research*, 14:700–719, 1989.

[4] N. Betzler, A. Slinko, and J. Uhlmann. On the computation of fully proportional representation. *Journal of Artificial Intelligence Research*, 47:475–519, 2013.

[5] S. J. Brams. Constrained approval voting: A voting system to elect a governing board. *Interfaces*, 20(5):67–80, 1990.

[6] R. Bredereck, P. Faliszewski, R. Niedermeier, P. Skowron, and N. Talmon. Mixed integer programming with convex/concave constraints: Fixed-parameter tractability and applications to multicovering and voting. Technical Report arXiv:1709.02850 [cs.DS], arXiv.org, 2017.

[7] L Elisa Celis, Lingxiao Huang, and Nisheeth K Vishnoi. Group fairness in multiwinner voting. Technical Report arXiv:1710.10057 [cs.CY], arXiv.org, 2017.

[8] B. Chamberlin and P. Courant. Representative deliberations and representative decisions: Proportional representation and the Borda rule. *American Political Science Review*, 77(3):718–733, 1983.

[9] C. Chekuri, J. Vondrk, and R. Zenklusen. Submodular function maximization via the multilinear relaxation and contention resolution schemes. *SIAM Journal on Computing*, 43(6):1831–1879, 2014.

[10] Gérard Cornuéjols. General factors of graphs. *Journal of Combinatorial Theory, Series B*, 45(2):185–198, 1988.

[11] Jack Edmonds. Matroid intersection. *Annals of Discrete Mathematics*, 4:39–49, 1979. Discrete Optimization I.

[12] E. Elkind and M. Lackner. Structure in dichotomous preferences. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI-2015)*, pages 2019–2025, 2015.

[13] P. Faliszewski, P. Skowron, A. Slinko, and N. Talmon. Committee scoring rules: Axiomatic classification and hierarchy. In *Proceedings of the 25th International Joint Conference on Artificial Intelligence (IJCAI-2016)*, pages 250–256, 2016.

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

[15] U. Feige. A threshold of $\ln n$ for approximating set cover. *Journal of the ACM*, 45(4):634–652, 1998.

[16] M. L. Fisher, G. L. Nemhauser, and L. A. Wolsey. An analysis of approximations for maximizing submodular set functions—ii. In M. L. Balinski and A. J. Hoffman, editors, *Polyhedral Combinatorics: Dedicated to the memory of D.R. Fulkerson*, pages 73–87. Springer, Berlin, Heidelberg, 1978.

[17] K. Florios, G. Mavrotas, and D. Diakoulaki. Solving multiobjective, multiconstraint knapsack problems using mathematical programming and evolutionary algorithms. *European Journal of Operational Research*, 203(1):14–21, 2010.

[18] A. Frank. A weighted matroid intersection algorithm. *Journal of Algorithms*, 2(4):328–336, 1981.

[19] A. Fréville. The multidimensional 0–1 knapsack problem: An overview. *European Journal of Operational Research*, 155(1):1–21, 2004.

[20] M. Garey and D. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company, 1979.

[21] Satoru Iwata and Kiyohito Nagano. Submodular function minimization under covering constraints. In *Proceedings of the 50th Annual IEEE Symposium on Foundations of Computer Science (FOCS-2009)*, pages 671–680, 2009.

[22] B. Korte and J. Vygen. *Combinatorial Optimization: Polyhedra and Efficiency*. Algorithms and Combinatorics. Springer, 2006.

[23] A. Krause and D. Golovin. Submodular function maximization. Technical report, 2012.

[24] A. Kulik, H. Shachnai, and T. Tamir. Approximations for monotone and nonmonotone submodular maximization with knapsack constraints. *Mathematics of Operations Research*, 38(4):729–739, 2013.

[25] J. Lang and P. Skowron. Multi-attribute proportional representation. In *Proceedings of the 30th Conference on Artificial Intelligence (AAAI-2016)*, pages 530–536, 2016.

[26] J. Lee, V. Mirrokni, V. Nagarajan, and M. Sviridenko. Non-monotone submodular maximization under matroid and knapsack constraints. In *Proceedings of the 41st Symposium on Theory of Computing (STOC-2009)*, pages 323–332, 2009.

[27] Hendrik W. Lenstra. Integer programming with a fixed number of variables. *Math. Oper. Res.*, 8(4):538–548, 1983.

[28] T. Lu and C. Boutilier. Budgeted social choice: From consensus to personalized decision making. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI-2011)*, pages 280–286, 2011.

[29] G. Nemhauser, L. Wolsey, and M. Fisher. An analysis of approximations for maximizing submodular set functions. *Mathematical Programming*, 14(1):265–294, December 1978.

[30] D. Peters and M. Lackner. Preferences single-peaked on a circle. In *Proceedings of the 31st Conference on Artificial Intelligence (AAAI-2017)*, pages 649–655, 2017.

[31] R. Potthoff. Use of linear programming for constrained approval voting. *Interfaces*, 20(5):79–80, 1990.

[32] A. Procaccia, J. Rosenschein, and A. Zohar. On the complexity of achieving proportional representation. *Social Choice and Welfare*, 30(3):353–362, 2008.

[33] J. Puchinger, G. Raidl, and U. Pferschy. The multidimensional knapsack problem: Structure and algorithms. *INFORMS Journal on Computing*, 22(2):250–265, 2010.

[34] C. Qian, J.-C. Shi, Y. Yu, and K. Tang. On subset selection with general cost constraints. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI-2017)*, pages 2613–2619, 2017.

[35] P. Skowron and P. Faliszewski. Fully proportional representation with approval ballots: Approximating the MaxCover problem with bounded frequencies in FPT time. In *Proceedings of the 29th Conference on Artificial Intelligence (AAAI-2015)*, pages 2124–2130, 2015.

[36] P. Skowron, P. Faliszewski, and J. Lang. Finding a collective set of items: From proportional multirepresentation to group recommendation. *Artificial Intelligence*, 241:191–216, 2016.

[37] P. Skowron, L. Yu, P. Faliszewski, and E. Elkind. The complexity of fully proportional representation for single-crossing electorates. *Theoretical Computer Science*, 569:43–57, 2015.

[38] Piotr Skowron, Piotr Faliszewski, and Arkadii M. Slinko. Achieving fully proportional representation: Approximability results. *Artifcial Intelligence*, 222:67–103, 2015.

[39] A. Straszak, M. Libura, J. Sikorski, and D. Wagner. Computer-assisted constrained approval voting. *Group Decision and Negotiation*, 2(4):375–385, 1993.

[40] M. Sviridenko. A note on maximizing a submodular set function subject to a knapsack constraint. *Operations Research Letters*, 32(1):41–43, 2004.

[41] Y. Yokoi. A generalized polymatroid approach to stable matchings with lower quotas. *Mathematics of Operations Research*, 42(1):238–255, 2017.

[42] L. Yu, H. Chan, and E. Elkind. Multiwinner elections under preferences that are single-peaked on a tree. In *Proceedings of the 23rd International Joint Conference on Artificial Intelligence (IJCAI-2013)*, pages 425–431, 2013.

Robert Bredereck
TU Berlin, Berlin, Germany
Email: robert.bredereck@tu-berlin.de

Piotr Faliszewski
AGH University, Krakow, Poland
Email: faliszew@agh.edu.pl

Ayumi Igarashi
University of Oxford
Oxford, United Kingdom
Email: ayumi.igarashi@cs.ox.ac.uk

Martin Lackner
TU Wien, Vienna, Austria
Email: lackner@dbai.tuwien.ac.at

Piotr Skowron
University of Warsaw, Warsaw, Poland
Email: p.k.skowron@gmail.com

Supplementary Material: Omitted Proofs

Proof of Theorem 3.1

Proof. We first consider the case where λ is 1-laminar and we give a polynomial-time algorithm.

Let $b\colon L \to 2^{[|C|]}$ be the cardinality constraint function corresponding to diversity specification D of the input. Let T be a rooted tree representation for L; we denote by r the root label that corresponds to the size k constraint on the whole committee size, i.e., $C_r = C$ and $b(r) = \{k\}$. Additionally, we add for every non-leaf label q in the tree representation of the labeling structure (including the possibly newly created r) an artificial label q^* with $b(q^*) = [0, |C|]$ and add this label to every candidate that has label q but none of the (original) child-labels of q. This step clearly does no influence the solvability of our problem but ensures that every candidate has at least one label that is a leaf node in the tree representation of the labeling structure.

For each $\ell \in L$, we denote by $\mathrm{child}(\ell, i)$ the ith child of $\ell \in L$ in T, and by $\#\mathrm{children}(\ell)$ number of children of $\ell \in L$ in T. By $\mathrm{desc}(\ell)$ we denote the set of all descendants of ℓ (including ℓ itself, i.e., $\ell \in \mathrm{desc}(\ell)$). Furthermore, let $\mathrm{best}(\ell, j)$ be the candidate from C_ℓ with the jth largest value according to f. For technical reasons, we introduce the $\perp$ symbol as placeholder for a non-existing (sub)committee and define $X \cup \perp := \perp$ for any set X. We set $f(\emptyset) := 0$ and $f(\perp) := -\infty$.

We describe a dynamic programming algorithm that solves DCWD using the integer table Opt where $\mathsf{Opt}[\ell, w, i]$ contains a (sub)committee W with maximum total score $f(W)$ among all committees that consist of $|W| = w$ candidates with labels from $\{\mathrm{child}(\ell, 1), \dots, \mathrm{child}(\ell, i)\}$ such that $|W \cap C_{\ell'}| \in b(\ell')$ for all $\ell' \in \mathrm{desc}(\mathrm{child}(\ell, j))$ and $j = 1, 2, \dots, i$.

It is not hard to verify that the overall solution for the DCWD instance can be read from the table as $\mathsf{Opt}[r, k, \#\mathrm{children}(r)]$.

We will now show how to compute the table Opt in a bottom-up manner. For each leaf label-node ℓ we set $\mathsf{Opt}[\ell, w, 0] := \{\mathrm{best}(\ell, j) \mid j \leq w\}$ if $w \in b(\ell)$, that is, $\mathsf{Opt}[\ell, w, 0]$ is the set of the w "best" candidates with label ℓ and, otherwise, we set $\mathsf{Opt}[\ell, w, 0] := \perp$. For each inner label-node ℓ, we set $\mathsf{Opt}[\ell, w, 1]$ to $\mathsf{Opt}[\mathrm{child}(\ell, i), x^*, \#\mathrm{children}(\mathrm{child}(\ell, i))]$ where

$$x^* := \mathrm{argmax}_{x \in [w]} f(\mathsf{Opt}[\mathrm{child}(\ell, i), x, \#\mathrm{children}(\mathrm{child}(\ell, i))])$$

if $w \in b(\ell)$ and, otherwise, we set $\mathsf{Opt}[\ell, w, 1] := \perp$. Further, for each inner label-node ℓ and $i > 1$, we set $\mathsf{Opt}[\ell, w, i]$ to $\mathsf{Opt}[\ell, w - x^*, i - 1] \cup \mathsf{Opt}[\mathrm{child}(\ell, i), x^*, \#\mathrm{children}(\mathrm{child}(\ell, i))]$ where $x^* := \mathrm{argmax}_{x \in [w]} f(\mathsf{Opt}[\ell, w - x, i - 1] \cup \mathsf{Opt}[\mathrm{child}(\ell, i), x, \#\mathrm{children}(\mathrm{child}(\ell, i))])$ if $w \in b(\ell)$ and, otherwise, we set $\mathsf{Opt}[\ell, w, i] := \perp$.

As for the running time, sorting the candidates with respect to their value according to f takes $O(|C| \cdot \log |C|)$ time. The table is of size $O(|L|^2 k)$ and computing a single table entries takes at most $O(k)$ time. The overall running time is $O(|L|^2 k^2 + |C| \log |C|))$ which is polynomial since $k \leq |C|$.

For DCF, we can skip to sort candidates which leads to the improved running time $O(|L|^2 k^2)$.

Let us now consider the second part of the theorem. We use a reduction from the NP-hard Exact Cover by 3-Sets which, given a finite set X and a collection $\mathcal{S}$ of size-3 subsets of X, asks whether there is a subcollection $\mathcal{S}' \subseteq \mathcal{S}$ that partitions X, that is, each element of X is contained in exactly one subset from $\mathcal{S}'$. The reduction is similar to the reduction of the somewhat closely related General Factor problem [10] and works as follows: Create one *element label* x for each element $x \in X$ and one *set label* S for each subset $S \in \mathcal{S}$. We set $b(S) := \{0, 3\}$ for each $S \in \mathcal{S}$ and $b(x) := \{1\}$ for each $x \in X$. For each subset $S = \{x, x', x''\} \in \mathcal{S}$, create three candidates $c(S, x)$, $c(S, x')$, and $c(S, x'')$ labeled with $\{S, x\}$, $\{S, x'\}$, and $\{S, x''\}$, respectively. Finally, set the committee size $k := |X|$. This completes the construction which can clearly be performed in polynomial time. For the correctness, assume that there is a subcollection $\mathcal{S}' \subseteq \mathcal{S}$ that partitions X. It is easy to verify that $\{c(S, x^*) \mid S \in \mathcal{S}', x^* \in S\}$ is a D-diverse committee. Furthermore, let $C^* \subseteq C$ be an arbitrary D-diverse committee. Now, $\mathcal{S}' = \{S \in \mathcal{S} \mid c(S, x) \in C^* \text{ for some } x \in S\}$ partitions X: each element $x \in X$ is covered exactly once since $b(x) = \{1\}$ for all $x \in X$, and $\mathcal{S}'$ is pairwise disjoint since $b(S) = \{0, 3\}$ for all $S \in \mathcal{S}$. □

Proof of Theorem 3.2

Proof. We describe a parametrized reduction from the $W[1]$-hard MULTICOLORED CLIQUE (MCC) problem which, given an undirected graph $G = (V, E)$, a non-negative integer $h \in \mathbb{N}$, and a vertex coloring $\phi\colon V \to \{1, 2, \ldots, h\}$, asks whether graph G admits a colorful h-clique, that is, a size-h vertex subset $H \subseteq V$ such that the vertices in H are pairwise adjacent and have pairwise distinct colors. Without loss of generality, we assume that the number of vertices from each color class equals some integer $q \leq |V|$. Let $(G = (V, E), \phi)$ be an MCC instance. We denote the set of vertices of color i as $V(i) = \{v_1^i, \ldots, v_q^i\}$. We construct a DCF instance as follows.

Labels. For each color $i \in [h]$ we have a *lower vertex label* low_i and a higher *higher vertex label* high_i. For each (unordered) color pair $i, j \in [h], i \neq j$ we have an *edge label* $\mathrm{ed}_{i,j}$. (So that $|L| = 2h + h(h-1)/2$ is obviously upper-bounded by some function in h.)

Candidates and Labeling. For each color $i \in [h]$ and each vertex $v \in V(i)$ we introduce $q(q+1)$ *lower color-i-selection candidates* and q *higher color-i-selection candidates.* The labeling function λ is defined as follows. For each lower color-i-selection candidate c we have $\lambda(c) := \{\mathrm{low}_i, \mathrm{high}_i\} \cup \{\mathrm{ed}_{i,j} \mid j < i\}$. For each higher color-$i$-selection candidate c we have $\lambda(c) := \{\mathrm{high}_i\} \cup \{\mathrm{ed}_{i,j} \mid j > i\}$. Introduce further $h(q+2)^2$ *dummy candidates* u with $\lambda(u) := \emptyset$.

Diversity Constraints. We define the cardinality constraint function b as follows. For each color $i \in [h]$ we set $b(\mathrm{low}_i) := \{(q+1)x \mid 1 \leq x \leq q\}$ and set $b(\mathrm{high}_i) := \{y \mid 1 \leq y \leq q\}$. For each (unordered) color pair $i, j \in [h], i < j$ we set $b(\mathrm{ed}_{i,j}) := \{(q+1)x + y \mid \text{there is an edge between the } x\text{th vertex from } V(i) \text{ and the } y\text{th vertex from } V(j)\}$.

We finally set the committee size $k := h(q+2)^2$. This completes the reduction which clearly runs in polynomial time. It remains to show that the graph G has a colorful h-clique if and only if the constructed DCF instance admits a diverse committee.

Assume that G has a colorful h-clique H. Let $\mathrm{id}(H, i)$ denote the index of the color i vertex from H, that is, $\mathrm{id}(H, i) = x$ if and only if H contains the xth vertex of color i. It is not hard to verify that a diverse committee can be constructed as follows. Start with a committee that consists only of k dummy candidates. For each color $i \in [h]$ replace $(q+1)\mathrm{id}(H, i)$ dummy candidates by lower color i-selection candidates and replace $\mathrm{id}(H, i)$ dummy candidates by higher color i-selection candidates. The diversity constraints of the lower and higher vertex labels are clearly fulfilled by this construction. Now, consider some edge label $\mathrm{ed}_{i,j}, i < j$. Our construction ensures that there are exactly $\mathrm{id}(H, i)(q+1)$ lower color-i-selection candidates in the committee with label $\mathrm{ed}_{i,j}$ and further $\mathrm{id}(H, j)$ higher color-j-selection candidates with label $\mathrm{ed}_{i,j}$ (and no further candidates with label $\mathrm{ed}_{i,j}$). Since H is a clique, we know that the $\mathrm{id}(H, i)$th vertex of color i is adjacent to he $\mathrm{id}(H, j)$th vertex of color j and thus $\mathrm{id}(H, i)(q+1) + \mathrm{id}(H, j) \in b(\mathrm{ed}_{i,j})$. Thus, also the diversity constraints for the edge labels are fulfilled and the committee is indeed diverse.

Finally, assume that the constructed DCF instance admits some diverse committee. To fulfill the diversity constraints for the lower vertex labels for each color $i \in [h]$ there is some number $\mathrm{id}(i)$ such that there are exactly $(q+1)\mathrm{id}(i)$ lower color-i-selection candidates and further $\mathrm{id}(i)$ higher color-i-selection candidates in the committee. (The former is directly enforced by the diversity constraints for the lower vertex labels and the latter follows then immediately from the diversity constraints for the higher vertex labels.) We claim that $H = \{v_i^H \mid \text{vertex } v_i \text{ is the id}(i)\text{th vertex of color } i\}$ is an h-colored clique. It is clear from the definition of H that $|H| = h$ and that H is h-colored but it remains to show that H is indeed a clique. To show this, suppose towards a contradiction that there are two colors $i, j \in [h], i < j$ such that vertex v_i^H and vertex v_j^H are not adjacent. Now, there are exactly $\mathrm{id}(i)(q+1)$ lower color-i-selection candidates in the committee with label $\mathrm{ed}_{i,j}$ and further $\mathrm{id}(j)$ higher color-j-selection candidates with label $\mathrm{ed}_{i,j}$ (and no further candidates with label $\mathrm{ed}_{i,j}$). Furthermore, since the diversity constraint of label $\mathrm{ed}_{i,j}$ is fulfilled, it must hold that $\mathrm{id}(i)(q+1) + \mathrm{id}(j) \in b(\mathrm{ed}_{i,j})$ and so that vertex v_i^H and vertex v_j^H are adjacent—a contradiction. □

Proof of Theorem 3.3

Proof. Let b_1 and b_2 denote the lower and upper interval constraint functions, respectively. Let T be a rooted tree representation for L; we denote by r the root label that corresponds to the size k constraint on the whole committee size, i.e., $C_r = C$ and $b_1(r) = b_2(r) = k$. For each $\ell \in L$, we denote by $\mathrm{child}(\ell)$ the set of children of ℓ and by $\mathrm{desc}(\ell)$ the set of descendants of ℓ in T (including ℓ).

For every label $\ell \in L$, let $\mathcal{K}_\ell$ be the set of committees $W \subseteq C_\ell$ satisfying the constraints up until ℓ, namely, $b_1(x) \le |W \cap C_x| \le b_2(x)$ for each $x \in \mathrm{desc}(\ell)$; then, we define $A_1[\ell]$ (respectively, $A_2[\ell]$) to be the minimum (respectively, the maximum) value w for which there is a set $W \in \mathcal{K}_\ell$ with $|W| = w$. If there exists no committee satisfying the aforementioned constraints we set $A_1[\ell]$ and $A_2[\ell]$ to ∞ and $-\infty$, respectively. Clearly, there is a D-diverse committee if and only if $A_1[r] = A_2[r] = k$. The values $A_1[\ell]$ and $A_2[\ell]$ can be efficiently computed by a dynamic programming in a bottom-up manner as follows.

For each leaf $\ell \in L$ at T, we set $A_1[\ell] = b_1(\ell)$ and $A_2[\ell] = \min\{b_2(\ell), |C_\ell|\}$ if $b_1(\ell) \le \min\{b_2(\ell), |C_\ell|\}$; we set $A_1[\ell] = +\infty$ and $A_2[\ell] = -\infty$ otherwise. For each internal node $\ell \in L$, we set $A_1[\ell] = \max\{\sum_{x \in \mathrm{child}(\ell)} T_1(x), b_1(\ell)\}$ and $A_2[\ell] = \min\{\sum_{x \in \mathrm{child}(\ell)} A_2[x], b_2(\ell), |C_\ell|\}$ if

- $A_1[x] \le A_2[x]$ for all child $x \in \mathrm{child}(\ell)$; and
- there are enough candidates in C_ℓ to fill in the lower bound ($|C_\ell| \ge A_1[\ell]$) and the lower bound does not exceed the upper bound, i.e., $\max\{\sum_{x \in \mathrm{child}(\ell)} A_1[x], b_1(\ell)\} \le \min\{\sum_{x \in \mathrm{child}(\ell)} A_2[x], b_2(\ell), |C_\ell|\}$.

Otherwise, we set $A_1[\ell] = +\infty$ and $A_2[\ell] = -\infty$. This can be done in $O(|C| + |L|)$ time since each $|C_\ell|$ for $\ell \in L$ can be computed in $O(|C|)$ time and since the size of the dynamic programming table is at most $|L|$ and each entry can be filled in constant time. It can be shown that for each $\ell \in L$ and each $w \in \mathbb{N}$, there is a set $W \in \mathcal{K}_\ell$ of size w if and only if $A_1[\ell] \le w \le A_2[\ell]$.

Now we will show by induction that for each $\ell \in L$ and each $w \in \mathbb{N}$, there is a set $W \in \mathcal{K}_\ell$ of size w if and only if $A_1[\ell] \le w \le A_2[\ell]$. The claim is immediate when $\mathrm{child}(\ell) = \emptyset$. Now consider an internal node $\ell \in L$ and suppose that the claim holds for all $x \in \mathrm{child}(\ell)$.

Suppose first that $A_1[\ell] \le w \le A_2[\ell]$. By induction hypothesis, for each child $x \in \mathrm{child}(\ell)$, there is a committee $W_x \in \mathcal{K}_x$ where $|W_x| = w_x$ for any $w_x \in [A_1[x], A_2[x]]$. By combining all such committees, we have that for any $t \in [\sum_{x \in \mathrm{child}(\ell)} A_1[x], \sum_{x \in \mathrm{child}(\ell)} T_2(x)]$, there is a committee $W \subseteq C_\ell$ of size t such that $W \cap C_x \in \mathcal{K}_x$ for all $x \in \mathrm{child}(\ell)$. In particular, since $w \in [\sum_{x \in \text{child}(\ell)} A_1[x], \sum_{x \in \text{child}(\ell)} T_2(x)]$, there is a set $W \subseteq C_\ell$ of size w such that $W \cap C_x \in \mathcal{K}_x$ for all $x \in \text{child}(\ell)$. Since $b_1(\ell) \leq w \leq b_2(\ell)$, we have $W \in \mathcal{K}_\ell$.

Conversely, suppose that w does not belong to the interval $[A_1[\ell], A_2[\ell]]$. Suppose towards a contradiction that there is a set $W \in \mathcal{K}_\ell$ of size w. Notice that for each $x \in \text{child}(\ell)$, it holds that $W \cap C_x \in \mathcal{K}_x$ and hence $A_1[x] \leq |W \cap C_x| \leq A_2[x]$ by induction hypothesis. If $w < b_1(\ell)$ or $w > b_2(\ell)$, it is clear that $W \notin \mathcal{K}_\ell$, a contradiction. Further, if $w > |C_\ell|$, W cannot be a subset of C_ℓ, a contradiction. If $w < \sum_{x \in \text{child}(\ell)} A_1[\ell]$, then $|W \cap C_x| < A_1[x]$ for some label $x \in \text{child}(\ell)$; however, since $W \cap C_x \in \mathcal{K}_x$, we have $|W \cap C_x| \geq A_1[x]$ by induction hypothesis, a contradiction. A similar argument leads to a contradiction if we assume $w > \sum_{x \in \text{child}(\ell)} A_1[\ell]$. ☐

Full proof of Theorem 3.4

Proof. Let $\mathcal{K}_D$ be the set of D-diverse committees of size k and assume that $\mathcal{K}_D$ is nonempty. For a family of subsets $\mathcal{K}$ of a finite set C, we define its lower extension[5] by

$$\overline{\mathcal{K}} = \{ T \mid \exists S \in \mathcal{K} : T \subseteq S \}.$$

It is known that if our constraints are given by intervals, the lower extension $\overline{\mathcal{K}}_D$ of $\mathcal{K}_D$ comprises the independent sets of a matroid whenever $\mathcal{K}_D \neq \emptyset$ [41]. Thus, the greedy algorithm (Algorithm 1) finds an optimal solution $W \in \text{argmax}_{W' \in \overline{\mathcal{K}}_D} f(W')$ (see e.g. Chapter 13 in [22]). By construction, $|W| = k$ and hence W is a maximal element in $\overline{\mathcal{K}}_D$, which follows that $W \in \mathcal{K}_D$. Since $\mathcal{K}_D \subseteq \overline{\mathcal{K}}_D$, we have $W \in \text{argmax}_{W' \in \mathcal{K}_D} f(W')$. Further, [41] showed that checking whether a set $W \cup \{y\}$ belongs to $\overline{\mathcal{K}}_D$ can be efficiently done by maintaining a set $B \in \mathcal{K}_D$ with $W \subseteq B$; thus, the greedy algorithm runs in polynomial time

Now it remains to analyze the running time of the algorithm. Sorting the candidates from the best to the worst requires $O(|C| \log |C|)$ time, given a weight vector $\boldsymbol{w}$. In each step, we need to check whether a set $W \cup \{y\}$ belongs to $\overline{\mathcal{K}}_D$. [41] showed that this can be efficiently done by maintaining a set $B \in \mathcal{K}_D$ with $W \subseteq B$. Specifically, the following lemma holds.

Lemma 7.1 (Lemma 6 of [41])**.** *Let $(C, \mathcal{I})$ be a matroid. Let W be an independent set of the matroid, B be a basis with $W \subseteq B$, and $y \in C \setminus W$. Then, $W \cup \{y\}$ is independent if and only if $y \in B$ or $(B \cup \{y\}) \setminus \{x\}$ is a basis for some $x \in B \setminus W$.*

The lemma implies that provided a set $B \in \mathcal{K}_D$ with $W \subseteq B$, deciding $W \cup \{y\} \in \overline{\mathcal{K}}_D$ can be verified by checking whether $y \in B$ or $(B \cup \{y\}) \setminus \{x\} \in \mathcal{K}_D$ for some $x \in B \setminus W$; this can be done in $O(k^2|L|)$ time. One can maintain such a superset $B \in \overline{\mathcal{K}}_D$ of W by first computing a set $B \in \mathcal{K}_D$ in $O(|C| + |L|)$ time as we have proved in Theorem 3.3, and updating the set B in each step as follows: If $y \notin B$, then find a candidate $x \in B \setminus W$ such that $(B \cup \{y\}) \setminus \{x\} \in \mathcal{K}_D$, and set $B = (B \cup \{y\}) \setminus \{x\}$; otherwise, we do not change the set B. Since there are at most $|C|$ iterations, the greedy algorithm runs in $O(k^2|C||L|)$ time. ☐

Proof of Theorem 3.5

In the subsequent proof, we will use the following notions and results in matroid theory: Given a matroid $(C, \mathcal{I})$, the sets in $2^C \setminus \mathcal{I}$ are called *dependent*, and a minimal dependent set of a matroid is called *circuit*. Crucial properties of circuits are the following.

[5] Note that the lower extension does not necessarily ignore the lower bounds. For instance, consider when we want to select a committee of size 3 such that there are exactly three female candidates and at most two male candidates; the corresponding lower extension $\overline{\mathcal{K}}_D$ only includes the sets of female candidates of size at most 3, whereas a male-only committee of size 2 satisfies the upper bounds.

Lemma 7.2. *Let $(C,\mathcal{I})$ be a matroid, $W \in \mathcal{I}$, and $y \in C \setminus W$ such that $W \cup \{y\} \notin \mathcal{I}$. Then the set $W \cup \{y\}$ contains a unique circuit.*

We write $C(W, y)$ for the unique circuit in $W \cup \{y\}$. The set $C(W, y)$ can be characterized by the elements that can replace y, i.e., for each independent set W of a matroid $(C,\mathcal{I})$ and $y \in C \setminus W$ with $W \cup \{y\} \notin \mathcal{I}$,

$$C(W, y) = \{\, x \in W \cup \{y\} \mid W \cup \{y\} \setminus \{x\} \in \mathcal{I} \,\}.$$

The following lemma by Frank [18] serves as a fundamental property for proving the matroid intersection theorem.

Lemma 7.3 (Frank [18])**.** *Let $(C,\mathcal{I})$ be a matroid and $W \in \mathcal{I}$. Let $x_1, x_2, \ldots, x_s \in W$ and $y_1, y_2, \ldots, y_s \notin W$ where $W \cup \{y_j\} \notin \mathcal{I}$ for $j \in [s]$. Suppose that*

(i) *$x_j \in C(W, y_j)$ for $j \in [s]$ and*

(ii) *$x_j \notin C(W, y_t)$ for $1 \le j < t < s$.*

Then, $(W \setminus \{x_1, x_2, \ldots, x_s\}) \cup \{y_1, y_2, \ldots, y_s\} \in \mathcal{I}$.

Now we are ready to prove Theorem 3.5.

Proof. Let b_1 and b_2 denote the lower and upper interval constraint functions, respectively. Let $L = L_1 \cup L_2, L_1 \cap L_2 = \emptyset$ be a partition of L such that for each $i = 1, 2$, the labeling restricted to the labels from L_i is 1-laminar. For $i = 1, 2$, we denote by $\mathcal{K}_i$ the set of committees of size k satisfying the constrains in L_i, i.e., $\mathcal{K}_i = \{\, S \subseteq C \mid |S| = k \text{ and } b_1(\ell) \le |S \cap C_\ell| \le b_2(\ell) \text{ for all } \ell \in L_i \,\}$. If at least one of them is empty, then there is no D-diverse committee; thus we assume otherwise. We have argued that the lower extension $\overline{\mathcal{K}}_i$ for each $i = 1, 2$ forms the independent sets of a matroid when $\lambda|_{L_i}$ is 1-laminar. Thus, our problem can be reduced to finding a maximum common independent set over the two matroids. That is, we will try to compute the following value:

$$\max\{\, |W| \mid W \in \overline{\mathcal{K}}_1 \cap \overline{\mathcal{K}}_2 \,\}.$$

Clearly, there is a D-diverse committee of size k if and only the maximum value equals k. It is well-known that this problem can be solved by Edmond's matroid intersection algorithm [11], given a membership oracle for each $\overline{\mathcal{K}}_i$. The idea is that starting with the empty set, we repeatedly find 'alternating paths' and augment W by one element in each iteration while keeping the property $W \in \overline{\mathcal{K}}_1 \cap \overline{K}_2$. Specifically, we apply the notion $C(W, y)$ to $(C, \overline{\mathcal{K}}_i)$ and write $C_i(W, y)$ for each $i = 1, 2$. For $W \in \overline{\mathcal{K}}_1 \cap \overline{\mathcal{K}}_2$, we define an auxiliary graph $G_W = (C, A_W^{(1)} \cup A_W^{(2)})$ where the set of arcs is given by

$$A_W^{(1)} = \{\, (x, y) \mid W \cup \{y\} \notin \overline{\mathcal{K}}_1 \wedge x \in C_1(W, y) \,\},$$
$$A_W^{(2)} = \{\, (y, x) \mid W \cup \{y\} \notin \overline{\mathcal{K}}_2 \wedge x \in C_2(W, y) \,\},$$

for $i = 1, 2$. We then look for a shortest path from $S_W^{(1)}$ to $S_W^{(2)}$, where

$$S_W^{(i)} = \{\, y \in C \setminus W \mid W \cup \{y\} \in \overline{\mathcal{K}}_i \,\},$$

for $i = 1, 2$. We increase the size of W by taking the symmetric difference with the path. It was shown that this procedure computes the desired value. We provide a formal description of the algorithm below (Algorithm 3).

Similarly to Yokoi [41], we can efficiently construct an auxiliary graph in each step by maintaining a set B_i such that $W \subseteq B_i$ and $B_i \in \mathcal{K}_i$ for each $i = 1, 2$: First, as we have seen in Lemma 7.1, we can determine the membership of a given set in $\overline{\mathcal{K}}_i$ in polynomial time. Moreover, it can be easily verified that the unique circuit $C_i(W, y)$ coincides with $C_i(B_i, y)$ when $W \cup \{y\} \notin \overline{\mathcal{K}}_i$.

Lemma 7.4. *Let $(C, \mathcal{I})$ be a matroid. Let W be an independent set of the matroid, B be a basis with $W \subseteq B$, and $y \in C \setminus W$ with $W \cup \{y\}$ being dependent. Then, $C(W, y) = C(B, y)$.*

Proof. Notice that $B \cup \{y\}$ is dependent: thus it contains a unique circuit $C(B, y)$. Then, $C(B, y) = C(W, y)$ clearly holds since $C(W, y) \subseteq B \cup \{y\}$. □

Thus, we will show how to maintain such a set $B_i \in \mathcal{K}_i$ with $W \subseteq B_i$ for each $i = 1, 2$. We will first compute a set $B_i \in \mathcal{K}_i$ for each $i = 1, 2$ in $O(|C||L|)$ time. Now suppose that $W \in \overline{\mathcal{K}}_1 \cap \overline{\mathcal{K}}_2$. Let $B_i \in \mathcal{K}_i$ where $W \subseteq B_i$ for $i = 1, 2$, and $P = (y_0, x_1, y_1, \dots, x_s, y_s)$ be a shortest path in G_W with $y_0 \in S_W^{(1)}$ and $y_s \in S_W^{(2)}$. Notice that $y_j \notin B_1$ for $j = 1, 2, \dots, s$ since otherwise B_1 contains a dependent set $W \cup \{y_j\}$ of a matroid $(C, \mathcal{K}_1)$, contradicting (I2); similarly, $y_j \notin B_2$ for $j = s, s-1, \dots, 1$ since otherwise B_2 contains a dependent set $W \cup \{y_j\}$ of a matroid $(C, \mathcal{K}_2)$, a contradiction. If $y_0 \notin B_1$, then there is a candidate $x \in B_1 \setminus W$ such that $(B_1 \cup \{y_0\}) \setminus \{x\} \in \mathcal{K}_1$ by Lemma 7.1, and we set B_1 to be $(B_1 \cup \{y_0\}) \setminus \{x\}$. Similarly, if $y_s \notin B_2$, then there is a candidate $x \in B_2 \setminus W$ such that $(B_2 \cup \{y_s\}) \setminus \{x\} \in \mathcal{K}_2$ by Lemma 7.1, and we set B_2 to be $(B_2 \cup \{y_0\}) \setminus \{x\}$. We then update each B_i as follows.

$$B_i' = (B_i \cup \{y_0, y_1, \dots, y_s\}) \setminus \{x_1, x_2, \dots, x_s\}.$$

Clearly, $W \cup \{y_0, y_1, \dots, y_s\}) \setminus \{x_1, x_2, \dots, x_s\} \subseteq B_i'$ for each $i = 1, 2$ since $y_0, y_1, \dots, y_s \notin W$ and $x_1, x_2, \dots, x_s \in W$; further we have the following.

Claim 7.5. *For $i = 1, 2$, $B_i' \in \mathcal{K}_i$.*

Proof. First, we show that $B_1 \cup \{y_0\}$, $y_1, y_2, \dots, y_s$, and $x_1, x_2, \dots, x_s$ satisfy the requirements of Lemma 7.3. Since $y_0 \in B_1$, we know that $B_1 = B_1 \cup \{y_0\} \in \mathcal{K}_1$. The condition (i) is satisfied because $(x_j, y_j) \in A_W^{(1)}$ and $C_1(W, y_j) = C_1(B_1, y_j)$ for all $j = 1, 2, \dots, s$. The condition (ii) is satisfied because otherwise $x_j \notin C(B, y_t) = C(W, y_t)$ for some $j < t$ and hence the path could be shortcut. Thus, $B_1' \in \overline{\mathcal{K}}_1$. Moreover, since $|B_1| = |B_1'| = k$, B_1' is a maximal set in $\overline{\mathcal{K}}_1$ and hence $B_1' \in \mathcal{K}_1$. Similarly, one can show that $B_2 \cup \{y_s\}$, $y_{s-1}, y_{s-2}, \dots, y_0$, and $x_s, x_{s-1}, \dots, x_1$ satisfy the requirements of Lemma 7.3, and therefore $B_2' \in \mathcal{K}_2$. □

It remains to analyze the running time of the algorithm. By Lemma 7.1 and Lemma 7.4, constructing an auxiliary graph G_W can be done in $O(k^2|C|^2|L|)$ time, given that we have a superset B_i of W where $B_i \in \mathcal{K}_i$, and finding a shortest path can be done in $O(|C|)$ time by breadth first search. Since there are at most $|C|$ augmentations, the overall running time is $O(k^2|C|^3|L|)$ time. □

Theorem 7.6. *Let D be a diversity specification of interval constraints. Suppose that λ is 2-laminar and f is a separable function given by a weight vector $\boldsymbol{w}\colon C \to \mathbb{R}$. Then,* DCWD *can be solved in $O(k|C|^3 + k^3|C|^2|L|)$ time.*

Proof. Again, for $i = 1, 2$, we denote by $\mathcal{K}_i$ the set of committees of size k satisfying the constrains in L_i, i.e., $\mathcal{K}_i = \{\, S \subseteq C \mid |S| = k \text{ and } b_1(\ell) \le |S \cap C_\ell| \le b_2(\ell) \text{ for all } \ell \in L_i \,\}$. We assume that $\mathcal{K}_1 \cap \mathcal{K}_2$ is nonempty. Frank [18] has shown that for each $k' \in [k]$, one can calculate $W_{k'} \in \overline{\mathcal{K}}_1 \cap \overline{\mathcal{K}}_2$ where

$$f(W_{k'}) = \max\{\, f(W) \mid W \in \overline{\mathcal{K}}_1 \cap \overline{\mathcal{K}}_2 \wedge |W| = k' \,\}$$

in $O(|C|^3 + \gamma)$, where γ is the time required for constructing an auxiliary graph G_W for each $W \in \overline{\mathcal{K}}_1 \cap \overline{\mathcal{K}}_2$. This completes the proof. □

Algorithm 3: Matroid intersection

input : $\mathcal{K}_i \neq \emptyset$ for $i = 1, 2$.
output : $W \in \overline{\mathcal{K}}_1 \cap \overline{\mathcal{K}}_2$ of maximum cardinality

set $W = \emptyset$;
compute $B_i \in \mathcal{K}_i$ for each $i = 1, 2$;
while *there is a shortest path* $P = (y_0, x_1, y_1, \ldots, x_s, y_s)$ *from* $S_W^{(1)}$ *to* $S_W^{(2)}$ *in* G_W **do**
  if $y_0 \notin B_1$ **then**
    find $x \in B_1 \setminus W$ such that $(B_1 \setminus \{x\}) \cup \{y_0\} \in \mathcal{K}_1$;
    set $B_1 = (B_1 \setminus \{x\}) \cup \{y_0\}$;
  if $y_s \notin B_2$ **then**
    find $x \in B_2 \setminus W$ such that $(B_2 \setminus \{x\}) \cup \{y_s\} \in \mathcal{K}_2$;
    set $B_2 = (B_2 \setminus \{x\}) \cup \{y_s\}$;
  set $W = W \cup \{y_0, y_1, \ldots, y_s\}) \setminus \{x_1, x_2, \ldots, x_s\}$ and
  $B_i = B_i \cup \{y_0, y_1, \ldots, y_s\}) \setminus \{x_1, x_2, \ldots, x_s\}$ for each $i = 1, 2$;

Proof of Theorem 3.7

Proof. To prove the theorem we will use the classic result of [27] which states that the problem of solving a mixed integer program is in FPT for the parameter being the number of integer variables. We create the following mixed integer linear program. For each combination of labels $Y \in 2^L$ we create an integer variable z_Y. Each z_Y denotes the number of committee members which have exactly the labels from Y in an optimal committee. Further, for each $Y \in 2^L$ we construct a function $g_Y : \mathbb{N} \to \mathbb{R}$ in the following way: $g_Y(x) = \max_{S\colon |S|=x} f(S)$. In words, $g_Y(x)$ gives the value of the best, according to the objective function f and ignoring the distribution constraints, x-element committee which consists only of candidates who have sets of labels equal to Y. Clearly, since f is separable, the functions g_Y are piecewise-linear and concave. We will construct a mixed ILP with the following non-linear objective function:

$$\text{minimize} \sum_{Y \in 2^L} g_Y(z_Y).$$

We can use piecewise linear concave functions in the minimized objective functions by the result of [6] (Theorem 2 in their work)—such programs can still be solved in FPT time with respect to the number of integer variables. The set of constraints is defined by taking the feasibility linear program `LP` and for each $\ell \in L$ setting $x_\ell = \sum_{Y\colon \ell \in Y} z_Y$. □

Proof of Theorem 4.1

Proof. Fisher et al. [16] showed that if $\overline{\mathcal{K}}_D$ is the set of independent sets of a matroid, Algorithm 1 produces a solution W such that $f(W) \geq \frac{1}{2} f(O'')$ for any optimal solution $O'' \in \operatorname{argmax}_{W' \in \overline{\mathcal{K}}_D} f(W')$. Now let $O' \in \operatorname{argmax}_{W' \in \mathcal{K}_D} f(W')$. Since $\mathcal{K}_D \subseteq \overline{\mathcal{K}}_D$, we have $f(W) \geq \frac{1}{2} f(O')$. Clearly, by construction, $W \in \mathcal{K}(D)$. □

Proof of Theorem 4.3

Proof. We first observe the following lemma.

Lemma 7.7. *Let f be a submodular function. Then, for any $S \subseteq T$ and any subset $X \subseteq C \setminus T$ of candidates,*

$$f(X|T) \leq f(X|S).$$

Proof. Let $P = S \cup X$ and $Q = T$. Then $P \cap Q = S$ and $P \cup Q = T \cup X$. Hence,

$$\begin{aligned}
&f(P \cup Q) + f(P \cap Q) \leq f(P) + f(Q) \\
&\Leftrightarrow f(T \cup X) + f(S) \leq f(S \cup X) + f(T) \\
&\Leftrightarrow f(T \cup X) - f(T) \leq f(S \cup X) - f(S) \\
&\Leftrightarrow f(X|T) \leq f(X|S).
\end{aligned}$$

□

Let W_i be our greedy solution after i iterations of the algorithm and $e_i = \{a_i, b_i\}$ be the pair chosen in the iteration where $a_i \in A$ and $b_i \in B$. Let O' be an optimal solution where $|O' \cap A| = |O' \cap B| = k'$. We first prove the following lemma.

Lemma 7.8. $f(W_i) \geq \frac{1}{k'} f(O') + (1 - \frac{1}{k'}) f(W_i)$

Proof. Let $O' \setminus W_i = \{a'_1, \ldots, a'_p\} \cup \{b'_1, \ldots, b'_q\}$ where each $a'_s \in A$ and $b'_t \in B$ for $s \in [p]$ and $t \in [q]$. Without loss of generality, suppose that $p \geq q$. Clearly, we have $p \leq k'$. Also, $f(O') \leq f(O' \cup W_i)$ by the monotonicity of f. Now, we let $e'_j = \{a'_j, b'_j\}$ for $j \in [q]$, and observe that

$$\begin{aligned}
&f(O' \cup W_i) = f(W_i) + \sum_{j=1}^{q} f(e'_j | W_i \cup e'_1 \cup \ldots \cup e'_{j-1}) \\
&+ \sum_{j=1}^{p-q} f(a'_{q+j} | W_i \cup e'_1 \cup \ldots \cup e'_q \cup \{a'_{q+1}, \ldots, a'_{q+j-1}\}) \\
&\leq f(W_i) + \sum_{j=1}^{q} f(e'_j | W_i) + \sum_{j=1}^{p-q} f(a'_{q+j} | W_i) \\
&\leq f(W_i) + \sum_{j=1}^{q} f(e'_j | W_i) + \sum_{j=1}^{p-q} f(\{a'_{q+j}, a_{i+1}\} | W_i) \\
&\leq f(W_i) + q f(e_{i+1} | W_i) + (p - q) f(e_{i+1} | W_i).
\end{aligned}$$

Here the first inequality follows from Lemma 7.7, the second inequality follows from the monotonicity of f, and the third inequality follows from the choice of e_{i+1}. Now, since $p \leq k'$, we have $f(O' \cup W_i) \leq f(W_i) + k' f(e_{i+1} | W_i)$. Thus, $f(O') \leq f(W_i) + k' f(e_{i+1} | W_i)$. □

Now by a usual calculation, we have

$$\begin{aligned}
f(W_{k'}) &\geq \frac{1}{k'} f(O') + (1 - \frac{1}{k'}) f(W_{k'-1}) \\
&\geq \cdots \\
&\geq f(O') \cdot [1 - (1 - \frac{1}{k'})^{k'}] \\
&\geq f(O')(1 - \frac{1}{e}).
\end{aligned}$$

□

Proof of Theorem 4.4

Proof. Let V be the set of n voters used to represent the Chamberlin–Courant objective function f. We first make a simple observation that for each $S \subseteq C$ we have $f(S) = \sum_{v_i \in V} \max_{c \in S}(m - \mathrm{pos}_i(c)) \leq \sum_{v_i \in V} m = nm$. We will use as a black box the result of[38][Theorem 11] who have shown an algorithm for selecting a committee S of k candidates such that $f(S) \geq nm\left(1 - \frac{2w(k)}{k}\right)$, where w is the Lambert's W-function (in particular $w(k) = o(\log(k))$). Now, for each $\epsilon > 0$ we can construct a polynomial-time $(1-\epsilon)$-approximation algorithm for BCWD as follows. We first define a threshold value $k_t \in \mathbb{N}$, as a smallest integer such that $\frac{2w(k_t)}{k_t} \leq \epsilon$. Now, if $k < k_t$, then we run a brute-force algorithm trying each k-element subset of the set of candidates and, this way, we can find an exactly optimal committee. If $k > k_t$, we run the algorithm of Skowron et al. for the instance of our problem without constraints and for the size of the committee equal to k; here we have $\epsilon \geq \frac{2w(k_t)}{k_t} \geq \frac{2w(k)}{k}$. Next, we complement the committee returned by this algorithm with some k candidates selected in an arbitrary way so that the diversity constraints are satisfied. Since, adding the candidates can only increase the value of the Chamberlin–Courant objective function, the value of the objective function for such constructed committee is at least equal to $nm(1 - \frac{2w(k)}{k}) \geq nm(1-\epsilon)$. Thus, this is a $(1-\epsilon)$-approximation committee. □

Proof of Theorem 5.1

Proof. For $t = 1$ we can simply check if for each two labels $\ell_1, \ell_2 \in L$ it holds that $C_{\ell_1} \setminus C_{\ell_2} = \emptyset$ or $C_{\ell_2} \setminus C_{\ell_1} = \emptyset$. For $t = 2$ we reduce the problem to 2-satisfiability (2SAT), which can be solved in polynomial time. Let us recall that a labeling λ is 2-laminar if L can be represented as a disjoint union of L_1 and L_2 such that $\lambda|_{L_i}$ is 1-laminar for each $i = 1, 2$.

For each label $\ell \in L$ we create one Boolean variable x_ℓ. Intuitively, if $x_\ell =$ True then $\ell \in L_1$, and if $x_\ell =$ False, then $\ell \in L_2$. For each two labels $\ell_1, \ell_2 \in L$ such that $C_{\ell_1} \setminus C_{\ell_2} \neq \emptyset$ and $C_{\ell_2} \setminus C_{\ell_1} \neq \emptyset$ we include two clauses: $(x_{\ell_1} \vee x_{\ell_2})$ and $(\neg x_{\ell_1} \vee \neg x_{\ell_2})$—these clauses ensure that $x_{\ell_1} \neq x_{\ell_2}$.

It is apparent that such constructed instance of 2SAT is satisfiable if and only if λ is 2-laminar.

For $t = 3$ we give a reduction from the partition into 3 cliques problem, which is NP-hard [20]. In this problem we are given a graph $G = (V, E)$ and we ask if it is possible to partition the set of vertices V into three sets such that the graphs induced by them are cliques.

For each vertex $x \in V$ we introduce a vertex label x, and for each non-adjacent pair of vertices $\{x, y\} \notin E$ we introduce a candidate $c_{\{x,y\}}$ with labels corresponding to x and y. First, we will show that if it is possible to partition the so-constructed graph into three cliques, V_1, V_2 and V_3, then the labeling is 3-layered with the layers corresponding to V_1, V_2 and V_3. For that we need to show that the constructed labeling is 1-layered when restricted to V_i for $i \in [3]$. Towards a contradiction, assume this is not the case, i.e., that there exist two labels $x, y \in V_i$ such that $C_x \cap C_y \neq \emptyset$. Let $c \in C_x \cap C_y$. Given that we constructed c, we infer that x and y were not adjacent in G, a contradiction.

On the other hand, suppose that the labeling is 3-layered, with layers V_1, V_2, V_3. Then V_i for $i \in [3]$ is a clique. Indeed, if there is a pair of labels $x, y \in V_i$ with $\{x, y\} \notin E$, then there is a candidate $c_{\{x,y\}} \in C_x \cap C_y$ and hence $C_x \cap C_y \neq \emptyset$, contradicting the fact that V_i is 1-layered. This completes the proof. □