# Algorithmic Analysis of Elections: Exercise Set

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

## Rules

Essentially all the tasks from this problem have solutions somewhere in the literature. Of course, your goal is to solve them without referring to works on computational social choice and you should rely only on the lecture slides. For each group of problems you can skip one subproblem.

## Preliminaries

An election is a pair $E = (C, V)$, where $E = \{c_1, \dots, c_n\}$ is a set of candidates and $V = (v_1, \dots, v_n)$ is a collection of voters, where each voter $v_i$ is associated with a linear order $\succ_{v_i}$ over the set of candidates (referred to as the preference order of voter $v_i$). For a voter $v$ and candidate $c$, we write $\mathrm{pos}_v(c)$ to denote the position of $c$ in $v$'s preference order (the top-ranked candidate has position 1, the next one position 2, and so on).

A single-winner election rule is a function $\mathcal{R}$ that given an election $E$ outputs a set of a candidates tied as winners (that is, we assume the non-unique winner model).

For a given integer $m$, we write $[m]$ to denote the set $\{1, \dots, m\}$. An $m$-candidate single-winner scoring function $f_m \colon [m] \to \mathbb{R}$ is a function that associates every position in a preference order (over $m$ candidates) with a score. Given an election $E = (C, V)$ and scoring function $f_{|C|}$, we define the $f_{|C|}$-score of candidate $c \in C$ as:

$$f_{|C|}\text{-score}(c) = \sum_{v \in V} f_{|C|}(\mathrm{pos}_v(c)).$$

Typical examples of scoring functions include the $k$-Approval function (note that the function is the same for all numbers of candidates and the $k$ index denotes a parameter of the function):

$$\alpha_k(i) = \begin{cases} 1 & \text{if } i \leq k \\ 0 & \text{otherwise} \end{cases}$$

and the Borda function:

$$\beta_m(i) = m - i.$$

Every family of scoring functions defines a voting rule in a natural way.

**Definition 2.** *Let $f = (f_{m,k})_{k \leq m}$ be a sequence of scoring functions (one for each number of candidates). Scoring rule $\mathcal{R}_f$ is an election rule that given an election $E = (C, V)$ outputs exactly the set of candidates with the highest $f_{|C|}$-scores.*

**Definition 1.** *Let $f = (f_i)_{i \in \mathbb{N}}$ be a sequence of committee scoring functions (one for each number of candidates and committee size). Committee scoring rule $\mathcal{R}_f$ is an election rule that given an election $E = (C, V)$ and committee size $k$ outputs exactly the set of size-$k$ committees with the highest $f_{|C|,k}$-scores.*

For example, the Plurality rule is defined by the $\alpha_1$ scoring function and the Borda rule is defined by the Borda scoring function.

A multiwinner election rule is a function $\mathcal{R}$ that given an election $E = (C, V)$ and an integer $k$ outputs a family of size-$k$ subsets of $C$ (the tied winning committees).

In particular, we are interested in committee scoring rules. Consider a vote $v$ over candidate set $C$ and a size-$k$ committee $S$. The position of $S$ in $v$, denoted $\mathrm{pos}_v(S)$, is the increasing sequence resulting from sorting the set $\{\mathrm{pos}_v(s) \mid s \in S\}$. For example, for vote $v \colon a > b > c > d > e$, the position of committee $S = \{a, c, d\}$ is $(1, 3, 4)$. For integers $m$ and $k$, we write $[m]_k$ to denote the set of length-$k$ increasing sequences of numbers from $[m]$ (in other words, $[m]_k$ is the set of possible positions of committees of size $k$ in votes over $m$ candidates). We say that a committee position $I = (i_1, \dots, i_k)$ weakly dominates committee position $J = (j_1, \dots, j_k)$ (denoted $I \succeq J$) if for each $t \in [k]$ we have that $i_t \leq j_t$.

A function $f_{m,k} \colon [m]_k \to \mathbb{R}$ is a committee scoring function for $m$ candidates and committees of size $k$ if for each two committee positions $I, J \in [m]_k$ such that $I \succeq J$ it holds that $f_{m,k}(I) \geq f_{m,k}(J)$. Analogously to the single-winner setting, given an election $E = (C, V)$, a committee size $k$, and a committee scoring function $f_{|C|,k}$, we define the $f_{|C|,k}$-score of a size-$k$ committee $S$ as:

$$f_{|C|,k}\text{-score}(S) = \sum_{v \in V} f_{|C|}(\mathrm{pos}_v(S)).$$

Example of committee scoring functions and rules that they define include:

**SNTV.** Single Non-Transferable Rule is defined through functions of the form $f^{\mathrm{SNTV}}_{m,k}(i_1,\dots,i_k) = \alpha_1(i_1) + \cdots + \alpha_1(i_k) = \alpha_1(i_1)$. In other words, under SNTV a committee gets a point from a voter whenever this voter ranks some member of this committee on the top position.

**Bloc.** The Bloc rule uses functions of the form $f^{\mathrm{Bloc}}_{m,k}(i_1,\dots,i_k) = \alpha_k(i_1) + \cdots + \alpha_k(i_k)$. That is, under Bloc each voter lists his or her favorite committee members and those $k$ that are mentioned most frequently are selected.

**$k$-Borda.** Under the $k$-Borda rule, we use the committee scoring functions $f^{k\text{-Borda}}_{m,k}(i_1,\dots,i_k) = \beta_m(i_1)+\cdots+\beta_m(i_k)$. That is, we select $k$ candidates with the highest individual Borda scores.

**$\beta$-CC.** Under the Borda-based Chamberlin–Courant rule, we assume that for a given committee $S$, a voter $v$ is represented by this member of $S$ that $v$ ranks highest. The score of a committee is the sum of the Borda scores of the voters' representatives. That is, $\beta$-CC uses committee scoring functions $f^{\beta\text{-CC}}_{m,k}(i_1,\dots,i_k) = \beta_m(i_1)$.

**$\alpha_k$-PAV** The $k$-Approval variant of the Proportional Approval Voting rule ($\alpha_k$-PAV) uses scoring functions of the form $f^{\alpha_k\text{-PAV}}_{m,k}(i_1,\dots,i_k) = \alpha_k(i_1) + \frac{1}{2}\alpha_k(i_2) + \frac{1}{3}\alpha_k(i_3) + \cdots + \frac{1}{k}\alpha_k(i_k)$.

## Problem 1: Control in Plurality Elections

We consider the problem of control by adding candidates under the Plurality rule.

**Definition 3.** *In the constructive control by adding candidates problem for the Plurality rule (Plurality-CCAC), we are given an election $(C \cup A, V)$, a candidate $p \in C$, and an integer $k$. We ask if there exists a subset $A'$ of at most $k$ candidates from $A$, such that $p$ is a Plurality winner in election $(C \cup A', V)$.*

We refer to the candidates in $C$ as the registered candidates and to those in $A$ as to unregistered ones.

**Problem 1.1.** Show that Plurality-CCAC is NP-complete.

We define the destructive version of the control by adding candidates problem (denoted Plurality-DCAC) analogously, except that we require that $p$ is not a winner in election $(C \cup A', V)$. After proving that Plurality-CCAC is NP-complete, noting that Plurality-DCAC is NP-complete as well is easy (no need to provide formal proof). However, in the destructive setting it is possible to obtain an FPT algorithm (in this case, for the parametrization by the number of voters).

**Problem 1.2.** Show that Plurality-CCDC is in FPT when parametrized by the number of voters (note that it is crucial here that we consider the destructive variant of the problem; it is possible to achieve $O^*(2^n)$ running time, where $n$ is the number of voters; running time $O^*(2^{2^n})$ is a easier to show).

## Problem 2: Committee Scoring Rules

Consider an election $E = (C, V)$, where $V = (v_1,\dots,v_n)$. We say that $E$ is single-crossing (with respect to the given ordering of the voters) if for each two candidates $a, b \in C$ it holds that there is an integer $t \in [n]$ such that the set $\{i \mid v_i : a \succ b\}$ of voters that prefer $a$ to $b$ is either of the form $\{1,2,\dots,t\}$ or $\{t,\dots,n\}$. In other words, an election is single-crossing (with respect to the given ordering of the voters) if for each two candidates $a$ and $b$, if $v_1$ prefers $a$ to $b$, then as we move from $v_1$ towards $v_n$, the relative ranking of $a$ and $b$ changes at most once. An election is single-crossing if there exists an ordering of the voters with respect to which it is single-crossing.

**Problem 2.1.** Show a polynomial-time algorithm for deciding if a given election is single-crossing.

**Problem 2.2.** Show a polynomial-time algorithm that given a single-crossing election $E = (C, V)$ and committee size $k$ computes the highest Chamberlin–Courant score of a size-$k$ committee in this election. (While in general this problem is NP-complete, for single-crossing elections it is possible to solve it in polynomial time.)

Recall Algorithm P and how it can be used to define a PTAS for the Borda-based Chamberlin–Courant problem.

**Problem 2.3.** Using Algorithm P as a building bloc, show that there is a PTAS for the committee scoring rule defined by the family of committee scoring functions:

$$f_{m,k}(i_1,\dots,i_k) = \beta_m(i_1) + \frac{1}{2}\beta_m(i_2) + \frac{1}{4}\beta_m(i_3) + \cdots + \frac{1}{2^k}\beta_m(i_k).$$

(By a PTAS, we mean an algorithm that for a given number $\varepsilon$ can compute a committee that achieves at least a $1-\varepsilon$ fraction of the score of the winning one; the algorithm should run in polynomial time with respect to the input size, but can depend exponentially on $\frac{1}{\varepsilon}$.)

**Research Problem** Is there a constant-factor approximation algorithm for a committee scoring rule defined by the function:

$$f_{m,k}(i_1,\dots,i_k) = \beta_m(i_1) + \frac{k-1}{k}\beta_m(i_2) + \frac{k-2}{k}\beta_m(i_3) + \cdots + \frac{1}{k}\beta_m(i_k).$$

## Problem 3: Justified Representation

In this problem we switch to the approval model of elections. In this model, given an election $E = (C, V)$, with $V = (v_1, \ldots, v_n)$, each voter is associated with the set $A_i \subseteq C$ of the candidates that he or she approves. The notion of a multiwinner voting rule in the approval setting is defined in the same way as in the preliminaries.

Consider an election $E = (C, V)$, where $V = (v_1, \ldots, v_n)$ and each voter $v_i$ approves candidates from the set $A_i$. The committee size is $k$. We are interested in the following rules:

**GreedyApprovalCC.** The GreedyApprovalCC rule starts with empty committee $W_0$ and performs $k$ rounds as follows: In the beginning of the $i$-th round, it chooses a candidate $w$ ($w \not{i}n W_{i-1}$) that is approved by the largest number of still-remaining votes, sets $W_i = W_{i-1} \cup \{w\}$, and removes all the voters that approve $w$ from the consideration. It outputs committee $W_k$.

**PAV.** The PAV score of a size-$k$ committee $S$ is defined as:

$$\mathrm{score}(S) = \sum_{i=1}^{n} \left( \sum_{t=1}^{|A_i|} \frac{1}{t} \right).$$

The PAV rule outputs those size-$k$ committees that have the highest PAV score. (Intuitively, under the PAV rule, a voter receives 1 point for "the first" candidate from the committee that he or she approves, $\frac{1}{2}$ point for "the second" one, $\frac{1}{3}$ for the third one, and so on.)

The goal in this problem set is to show that committees that win under these rules provide justified representation.

**Definition 4.** *Consider an approval election $E = (C, V)$. A size-$k$ committee $S$* fails *to provide justified representation if there exists a group of at least $\frac{n}{k}$ voters such that: (a) there is at least one candidate that all these voters approve, and (b) none of the voters in this group approves any of the members of $S$. A committee $S$ provides justified representation if it does not fail to provide it.*

**Problem 3.1.** Show a polynomial-time algorithm that given an approval election $E = (C, V)$ and a committee $S$ decides if $S$ provides justified representation or not.

**Problem 3.2.** Show that the winning committee under GreedyApprovalCC always provides justified representation for the input election (note that this implies that for every election and committee size $k$, there is a size-$k$ committee that provides justified representation).

**Problem 3.3.** Show that every PAV winning committee provides justified representation for the input election.