ssdnm
środowiskowe
studia doktoranckie
z nauk matematycznych

Piotr Skowron

Uniwersytet Warszawski

Proportional Representation as Resource Allocation: Approximability Results

Praca semestralna nr 3

(semestr letni 2011/12)

Opiekun pracy: dr hab. Piotr Faliszewski

KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY

Proportional Representation as Resource Allocation: Approximability Results

Piotr Skowron
University of Warsaw
Warsaw, Poland

Piotr Faliszewski
AGH University
Krakow, Poland

Arkadii Slinko
University of Auckland
Auckland, New Zealand

August 3, 2012

Abstract

We model Monroe's and Chamberlin and Courant's multiwinner voting systems as a certain resource allocation problem. We show that for many restricted variants of this problem, under standard complexity-theoretic assumptions, there are no constant-factor approximation algorithms. Yet, we also show cases where good approximation algorithms exist (briefly put, these variants correspond to optimizing total voter satisfaction under Borda scores, within Monroe's and Chamberlin and Courant's voting systems).

1 Introduction

Resource allocation is one of the most important issues in multiagent systems, equally important both to human societies and to artificial software agents [29]. For example, if there is a set of items (or a set of bundles of items) to distribute among agents then we may use one of many auction mechanisms (see, e.g., [22,29] for an introduction and a review, and numerous recent papers on auction theory for current results). However, typically in auctions if an agent obtains an item (a resource) then this agent has exclusive access to it. In this paper we consider resource allocation for items that can be shared, and we are interested in computing (approximately) optimal assignments (in particular, for settings where resource allocation boils down to multiwinner voting). As opposed to a large body of research on auctions, resource allocation, and mechanism design, we do not make any strategic considerations.

Let us explain our resource allocation problem through an example. Consider a company that wants to provide free sport classes to its employees. We have a set $N = \{1, \dots, n\}$ of employees and a set $A = \{a_1, \dots, a_m\}$ of classes that are offered. Naturally, not every class is equally appealing to each employee and, thus, each employee orders the classes from the most desirable one to the least desirable one. For example, the first employee might have preference order $a_1 \succ a_3 \succ \cdots \succ a_m$, meaning that for him or her a_1 is the most attractive class, a_3 is second, and so on, until a_m, which is least appealing. Further, each class a_i has some maximum capacity cap_{a_i}, that is, a maximum number of people that can comfortably participate, and a cost, denoted c_{a_i}, of opening the class (independent of the number of participants). The company wants to assign the employees to the classes so that it does not exceed its sport-classes budget B and so that the employees' satisfaction is maximal (or, equivalently, their dissatisfaction is minimal).

There are many ways to measure (dis)satisfaction. For example, we may measure an employee's dissatisfaction as the position of the class to which he or she was assigned in his or her preference order (and satisfaction as m less the voter's dissatisfaction). Then, we could demand that, for example, the maximum dissatisfaction of an employee is as low as possible (minimal satisfaction is as high as possible; in economics this corresponds to egalitarian social welfare) or that the sum of dissatisfactions is minimal (the sum of satisfactions is maximal; this corresponds to the utilitarian approach in economics).

It turns out that our model generalizes two well-known multiwinner voting rules; namely, those of Monroe [20] and of Chamberlin and Courant [8]. Under both these rules voters from the set N submit preference orders regarding alternatives from the set A, and the goal is to select K candidates (the representatives) best representing the voters. For simplicity, let us assume that K divides $\|N\|$.[1] Under Monroe's rule we have to match each selected representative to $\frac{\|N\|}{K}$ voters so that each voter has a unique representative and so that the sum of voters' dissatisfactions is minimal (dissatisfaction is, again, measured by the position of the representative in the voter's preference order). Chamberlin and Courant's rule is similar except that there are no restrictions on the number of voters a given alternative represents (in this case it is better to think of the alternatives as political parties rather than particular politicians). It is easy to see that both methods are special cases of our setting: For example, for Monroe it suffices to set the "cost" of each alternative to be 1, to set the budget to be K, and to set the "capacity" of each alternative to be $\frac{\|N\|}{K}$. We can consider variants of these two systems using different measures of voter (dis)satisfaction, as indicated above (see also the works of Potthoff and Brams [24], Betzler et al. [4] and of Lu and Boutilier [19]).

Unfortunately, it is well-known that both Monroe's method and Chamberlin and Courant's method are NP-hard to compute in essentially all nontrivial settings [4,19,25]. This holds even if various natural parameters of the election are low [4]. Notable exceptions include, e.g., the case where K is bounded by a fixed constant and the case where voter preferences are single-peaked [4].

Nonetheless, Lu and Boutilier [19]—starting from a very different motivation and context—propose to rectify the high computational complexity of Chamberlin and Courant's system by designing approximation algorithms. In particular, they show that if one focuses on the sum of voters' satisfactions, then there is a polynomial-time approximation algorithm with approximation ratio $(1 - \frac{1}{e}) \approx 0.63$ (i.e., their algorithm outputs an assignment

[1] We stress that this assumption does not really affect our results. Our algorithms would maintain their quality without the assumption. On the other hand, without the assumption modeling Monroe's and Chamberlin and Courant's systems would be more tedious and some calculations would be a bit more involved.

that achieves no less than about 0.63 of optimal voter satisfaction). Unfortunately, total satisfaction is a tricky measure. For example, under standard Chamberlin and Courant's system, a $\frac{1}{2}$-approximation algorithm is allowed to match each voter to an alternative somewhere in the middle of this voter's preference order, even if there is a feasible solution that matches each voter to his or her most preferred candidate. On the other hand, it seems that a 2-approximation focusing on total dissatisfaction would give results of very high quality.

The goal of this paper is to provide an analysis of our resource allocation scenario, focusing on approximation algorithms for the special cases of Monroe's and Chamberlin and Courant's voting systems. We obtain the following results:

1. Monroe's and Chamberlin and Courant's systems are hard to approximate up to any constant factor for the case where we measure dissatisfaction, irrespective of whether we measure the total dissatisfaction (Theorems 1 and 3) or the dissatisfaction of the most dissatisfied voter (Theorems 2 and 4).
2. Monroe's and Chamberlin and Courant's systems are hard to approximate within any constant factor for the case where we measure satisfaction of the least satisfied voter (Theorems 5 and 7). However, there are good approximation algorithms for total satisfaction—for the Monroe's system we achieve approximation ratio arbitrarily close to 0.715 (and often a much better one; see Section 4.1). For Chamberlin and Courant's system we give a polynomial-time approximation scheme (that is, for each ϵ, $0 < \epsilon < 1$, we have a polynomial-time $(1-\epsilon)$-approximation algorithm; see Theorem 13).

Our work is similar to several lines of research on computational social choice and multiagent systems. In particular, there is a well-established line of work on the hardness and approximability of winner determination for single-winner voting rules, with results for, for example, Dodgson's rule [3,6,7,12,15], Kemeny's rule [1,3,10,16,17], Young's rule [6,27], and Ranked Pairs method [5]. Hardness of winner determination for multiwinner voting rules was studied by Procaccia, Rosenschein, and Zohar [25], by Lu and Boutilier [19], and by Betzler, Slinko and Uhlman [4]. Lu and Boutilier, while starting from a different context, initiated the study of approximation algorithms in this setting, which we continue and extend in this paper.

In the context of resource allocation, our model closely resembles multi-unit resource allocation with single-unit demand [29, Chapter 11] (see also the work of Chevaleyre et al. [9] for a survey of the most fundamental issues in the multiagent resource allocation theory). The problem of multi-unit resource allocation is mostly addressed in the context of auctions (and so it is referred in the literature as multi-unit auctions); in contrast, we consider the problem of finding a solution maximizing the social welfare given the agents' preferences. More generally, our model, is similar to resource allocation with sharable indivisible goods [2,9]. The most substantial difference is that we require each agent to be assigned to exactly one alternative. Also, in the context of resource allocation with sharable items, it is often assumed that the agents' satisfaction is affected by the number of agents using the alternatives (the congestion on the alternatives). This forms a class of problems that are closely related to congestion games [26]. Finally, it is worth mentioning that in the literature on resource allocation it is common to consider other criteria of optimality, such as envy-freeness [18], Pareto optimality, Nash equilibria [2], and others.

Finally, we should mention that our paper is very close in spirit (especially in terms of the motivation of the resource allocation problem) to the recent work of Darmann et al. [11].

2 Preliminaries

We first define basic notions such as, e.g., preference orders and positional scoring rules. Then we present our resource allocation problem in full generality and discuss restrictions modeling Monroe's and Chamberlin and Courant's voting systems. Finally, we briefly recall relevant notions regarding computational complexity theory.

Alternatives, Profiles, Positional Scoring Functions. For each $n \in \mathbb{N}$, we take $[n]$ to mean $\{1, \ldots, n\}$. We assume that there is a set $N = [n]$ of *agents* and a set $A = \{a_1, \ldots a_m\}$ of *alternatives*. Each agent i has *weight* $w_i \in \mathbb{N}$, and each alternative a has *capacity* $\mathrm{cap}_a \in \mathbb{N}$ and *cost* $c_a \in \mathbb{N}$. The weight of an agent corresponds to its size (measured in some abstract way). An alternative's capacity gives the total weight of the agents that can be assigned to it, and its cost gives the price of selecting the alternative (the price is the same irrespective of the weight of the agents assigned to the alternative). Further, each agent i has a *preference order* $\succ_i$ over A, i.e., a strict linear order of the form $a_{\pi(1)} \succ_i a_{\pi(2)} \succ_i \cdots \succ_i a_{\pi(m)}$ for some permutation π of $[m]$. For an alternative a, by $\mathrm{pos}_i(a)$ we mean the position of a in i'th agent's preference order. For example, if a is the most preferred alternative for i then $\mathrm{pos}_i(a) = 1$, and if a is the most despised one then $\mathrm{pos}_i(a) = m$. A collection $V = (\succ_1, \ldots, \succ_n)$ of agents' preference orders is called a *preference profile*. We write $\mathcal{L}(A)$ to denote the set of all possible preference orders over A. Thus, for preference profile V of n agents we have $V \in \mathcal{L}(A)^n$.

In our computational hardness proofs, we will often include subsets of alternatives in the descriptions of preference orders. For example, if A is the set of alternatives and B is some nonempty strict subset of A, then by saying that some agent has preference order of the form $B \succ A - B$, we mean that this agent ranks all the alternatives in B ahead of all the alternatives outside of B, and that the order in which this agent ranks alternatives within B and within $A - B$ is irrelevant (and, thus, one can assume any easily computable order).

A *positional scoring function* (PSF) is a function $\alpha^m : [m] \to \mathbb{N}$. A PSF α^m is an *increasing positional scoring function* (IPSF) if for each $i, j \in [m]$, if $i < j$ then $\alpha^m(i) < \alpha(j)$. Analogously, a PSF α^m is a *decreasing positional scoring function* (DPSF) if for each $i, j \in [m]$, if $i < j$ then $\alpha^m(i) > \alpha^m(j)$.

Intuitively, if β^m is an IPSF then $\beta^m(i)$ gives the *dissatisfaction* that an agent suffers from when assigned to an alternative that is ranked i'th on his or her preference order. Thus, we assume that for each IPSF β^m it holds that $\beta^m(1) = 0$ (an agent is not dissatisfied by his or her top alternative). Similarly, a DPSF γ^m measures an agent's satisfaction and we assume that for each DPSF γ^m it holds that $\gamma^m(m) = 0$.

We will often speak of families α of IPSFs (DPSFs) of the form $\{\alpha^m \mid m \in \mathbb{N}, \alpha^m \text{ is a PSF}\}$, where the following holds:

1. For IPSFs, for each $m \in \mathbb{N}$ it holds that $(\forall i \in [m])[\alpha^{m+1}(i) = \alpha^m(i)]$.
2. For DPSFs, for each $m \in \mathbb{N}$ it holds that $(\forall i \in [m])[\alpha^{m+1}(i+1) = \alpha^m(i)]$.

In other words, we build our families of IPSFs (DPSFs) by appending (prepending) values to functions with smaller domains. We assume that each function α^m from a family can be computed in polynomial time with respect to m. To simplify notation, we will refer to such families of IPSFs (DPSFs) as *normal* IPSFs (normal DPSFs).

We are particularly interested in normal IPSFs (normal DPSFs) corresponding to the Borda count method. That is, in the families of IPSFs $\alpha^m_{\mathrm{B,inc}}(i) = i - 1$ (in the families of DPSFs $\alpha^m_{\mathrm{B,dec}}(i) = m - i$).

Our Resource Allocation Problem. We consider a problem of finding function $\Phi : N \to A$ that assigns each agent to some alternative (we will call Φ an *assignment function*). We say that Φ is feasible if for each alternative a it holds that the total weight of the agents assigned to it does not exceed its capacity cap_a. Further, we define the cost of assignment Φ to be $\mathrm{cost}(\Phi) = \sum_{a:\Phi^{-1}(a)\neq\emptyset} c_a$.

Given an IPSF (DPSF) α^m, we consider two *dissatisfaction functions*, $\ell_1^\alpha(\Phi)$ and $\ell_\infty^\alpha(\Phi)$, (two *satisfaction functions*, $\ell_1^\alpha(\Phi)$ and $\min^\alpha(\Phi)$), measuring the quality of the assignment as follows:

1. $\ell_1^\alpha(\Phi) = \sum_{i=1}^n \alpha(\mathrm{pos}_i(\Phi(i)))$.
2. $\ell_\infty^\alpha(\Phi) = \max_{i=1}^n \alpha(\mathrm{pos}_i(\Phi(i)))$ (or, $\min^\alpha(\Phi) = \min_{i=1}^n \alpha(\mathrm{pos}_i(\Phi(i)))$).

The former one measures agents' total dissatisfaction (satisfaction), whereas the latter one considers the most dissatisfied (the least satisfied) agent only. In welfare economics and multiagent resource allocation theory the two metrics correspond to, respectively, utilitarian and egalitarian social welfare. We define our resource allocation problem as follows.

Definition 1. *Let α be a normal IPSF. An instance of* α-ASSIGNMENT-INC *problem consists of a set of agents $N = [n]$, a set of alternatives $A = \{a_1, \dots a_m\}$, a preference profile V of the agents, a sequence $(w_1, \dots, w_n)$ of agents' weights, sequences $(\mathrm{cap}_{a_1}, \dots, \mathrm{cap}_{a_m})$ and $(c_{a_1}, \dots, c_{a_m})$ of alternatives' capacities and costs, respectively, and budget $B \in \mathbb{N}$. We ask for the assignment function Φ such that:*

1. $\mathrm{cost}(\Phi) \leq B$,
2. $\forall_{a\in A} \sum_{i:\Phi(i)=a} w_i \leq cap_a$*, and*
3. $\ell_1^\alpha(\Phi)$ *is minimized.*

In other words, in α-ASSIGNMENT-INC we ask for a feasible assignment that minimizes the total dissatisfaction of the agents without exceeding the budget.

Problem α-ASSIGNMENT-DEC is defined identically except that α is a normal DPSF and in the third condition we seek to maximize $\ell_1^\alpha(\Phi)$ (that is, in α-ASSIGNMENT-DEC our goal is to maximize total satisfaction). If we replace ℓ_1^α with ℓ_∞^α in α-Assignment-Inc then we obtain problem α-Minmax-Assignment-Inc, where we seek to minimize the dissatisfaction of the most dissatisfied agent. If we replace ℓ_1^α with $\min^\alpha$ in α-Assignment-Dec then we obtain problem α-Minmax-Assignment-Dec, where we seek to maximize the satisfaction of the least satisfied agent.

As far as optimal solutions go, satisfaction and dissatisfaction formulations of our problems are equivalent. However, as we will see, there are striking differences in terms of their approximability.

Clearly, each of our four Assignment problems is NP-complete: Even without costs they reduce to the standard NP-complete Partition problem, where we ask if a set of integers (in our case these integers would be agents' weights) can be split evenly between two sets (in our case, two alternatives with the capacities equal to half of the total agent weight). However, in very many applications (for example, in the sport classes example from the introduction) it suffices to consider unit-weight agents. Thus, from now on we assume the agents have unit weights.

Our four problems can be viewed as generalizations of Monroe's [20] and Chamberlin and Courant's [8] multiwinner voting systems (see the introduction for their definitions). To model Monroe's system, it suffices to set the budget $B = K$, the cost of each alternative to be 1, and the capacity of each alternative to be $\frac{\|N\|}{K}$ (for simplicity, throughout the paper we assume that K divides $\|N\|$). We will refer to thus restricted variants of our problems as Monroe-Assignment variants. To represent Chamberlin and Courant's system within our framework, it suffices to take the same restrictions as for Monroe's system, except that each alternative has capacity equal to $\|N\|$. We will refer to thus restricted variants of our problems as CC-Assignment variants.

The eight above-defined special cases of our resource allocation problem were, in various forms and shapes, considered by Procaccia, Rosenschein, and Zohar [25], Lu and Boutilier [19], and Betzler, Slinko and Uhlmann [4].

Computational Complexity, Approximation Algorithms. For many normal IPSFs α (and, in particular, for Borda count), even the above-mentioned restricted versions of the original problem, namely, α-Monroe-Assignment-Inc, α-Minmax-Monroe-Assignment-Inc, α-CC-Assignment-Inc, and α-Minmax-CC-Assignment-Inc are NP-complete [4,25] (the same holds for normal DPSFs and Dec variants of the problems). Thus, we explore possibilities for approximate solutions.

Definition 2. *Let β be a real number such that $\beta \geq 1$ ($0 < \beta \leq 1$) and let α be a normal IPSF (a normal DPSF). An algorithm is a β-approximation algorithm for α-*Assignment-Inc *problem (for α-*Assignment-Dec *problem) if on each instance I it returns a feasible assignment Φ that meets the budget restriction and such that $\ell_1^\alpha(\Phi) \leq \beta \cdot \mathrm{OPT}$ (and such that $\ell_1^\alpha(\Phi) \geq \beta \cdot \mathrm{OPT}$), where* OPT *is the aggregated dissatisfaction (satisfaction) $\ell_1^\alpha(\Phi_{\mathrm{OPT}})$ of the optimal assignment Φ_{OPT}.*

We define β-approximation algorithms for the Minmax variants of our problems analogously. For example, Lu and Boutilier [19] present a $(1 - \frac{1}{e})$-approximation algorithm for the case of CC-Assignment-Dec.

Throughout this paper, we will consider each of the Monroe-Assignment and CC-Assignment variants of the problem and for each we will either prove inapproximability with respect to any constant β (under standard complexity-theoretic assumptions) or we will show an approximation algorithm. In our inapproximability proofs, we will use the following classic NP-complete problems [14].

Definition 3. *An instance I of* Set-Cover *consists of set $U = [n]$ (called the ground set), family $\mathcal{F} = \{F_1, F_2, \dots, F_m\}$ of subsets of U, and positive integer K. We ask if there exists a set $I \subseteq [m]$ such that $\|I\| \leq K$ and $\bigcup_{i \in I} F_i = U$.*

Definition 4. Vertex-Cover *is a special case of* Set-Cover*, where U and $\mathcal{F}$ are constructed from a given graph G. Specifically, U is the set of G's edges and $\mathcal{F} = \{F_1, \dots, F_n\}$ corresponds to G's vertices (for each vertex v of G, $\mathcal{F}$ has a corresponding set F, which contains the edges incident to v).*

Definition 5. X3C *is a special case of* Set-Cover *where $\|U\|$ is divisible by 3, each member of $\mathcal{F}$ has exactly three elements, and $K = \frac{n}{3}$.*

Note that X3C remains NP-complete even if we additionally assume that n is divisible by 2 and each member of U appears in at most 3 sets from $\mathcal{F}$ [14].

3 Hardness of Approximation

In this section we present our inapproximability results for Monroe-Assignment and CC-Assignment variants of the resource allocation problem. In particular, we show that if we focus on voter dissatisfaction (i.e., on the Inc variants) then for each $\beta > 1$, neither Monroe's nor Chamberlin and Courant's system has a polynomial-time β-approximation algorithm. Further, we show that analogous results hold if we focus on the satisfaction of the least satisfied voter.

Naturally, these inapproximability results carry over to more general settings. In particular, unless P = NP, there are no polynomial-time constant-factor approximation algorithms for the general resource allocation problem for the case where we focus on voter dissatisfaction. On the other hand, our results do not preclude good approximation algorithms for the case where we measure agents' total satisfaction. Indeed, in Section 4 we derive algorithms for several satisfaction-based special cases of the problem.

We start by showing that for each normal IPSF α there is no constant-factor polynomial-time approximation algorithm for α-Monroe-Assignment-Inc (and, thus, there is no such algorithm for general α-Assignment-Inc).

Theorem 1. *For each normal IPSF α and each constant factor β, $\beta > 1$, there is no polynomial-time β-approximation algorithm for α-*Monroe-Assignment-Inc *unless* P = NP.

Proof. Let us fix a normal IPSF α and let us assume, for the sake of contradiction, that there is some constant β, $\beta > 1$, and a polynomial-time β-approximation algorithm $\mathcal{A}$ for α-MONROE-ASSIGNMENT-INC.

Let I be an instance of X3C with ground set $U = [n]$ and family $\mathcal{F} = \{F_1, F_2, \ldots, F_m\}$ of 3-element subsets of U. W.l.o.g., we assume that n is divisible by both 2 and 3 and that each member of U appears in at most 3 sets from $\mathcal{F}$.

Given I, we build instance I_M of α-MONROE-ASSIGNMENT-INC as follows. We set $N = U$ (that is, the elements of the ground set are the agents) and we set $A = A_1 \cup A_2$, where $A_1 = \{a_1, \ldots, a_m\}$ is a set of alternatives corresponding to the sets from the family $\mathcal{F}$ and A_2, $\|A_2\| = \frac{n^2 \cdot \alpha(3) \cdot \beta}{2}$, is a set of dummy alternatives needed for our construction. We let $m' = \|A_2\|$ and we rename the alternatives in A_2 so that $A_2 = \{b_1, \ldots, b_{m'}\}$. We set $K = \frac{n}{3}$.

We build agents' preference orders using the following algorithm. For each $j \in N$, set $M_f(j) = \{a_i \mid j \in F_i\}$ and $M_l = \{a_i \mid j \notin F_i\}$. Set $m_f(j) = \|M_f(j)\|$ and $m_l(j) = \|M_l(j)\|$; as the frequency of the elements from U is bounded by 3, $m_f(j) \leq 3$. For each agent j we set his or her preference order to be of the form $M_f(j) \succ_j A_2 \succ_j M_l(j)$, where the alternatives in $M_f(j)$ and $M_l(j)$ are ranked in an arbitrary way and the alternatives from A_2 are placed at positions $m_f(j)+1, \ldots, m_f(j)+m'$ in the way described below (see Figure 1 for a high-level illustration of the construction).

We place the alternatives from A_2 in the preference orders of the agents in such a way that for each alternative $b_i \in A_2$ there are at most two agents that rank b_i among their $n \cdot \alpha(3) \cdot \beta$ top alternatives. The following construction achieves this effect. If $(i+j) \bmod n < 2$, then alternative b_i is placed at one of the positions $m_f(j)+1, \ldots, m_f(j) + n \cdot \alpha(3) \cdot \beta$ in j's preference order. Otherwise, b_i is placed at a position with index higher than $m_f(j) + n \cdot \alpha(3) \cdot \beta$ (and, thus, at a position higher than $n \cdot \alpha(3) \cdot \beta$). This construction can be implemented because for each agent j there are exactly $m' \cdot \frac{2}{n} = n \cdot \alpha(3) \cdot \beta$ alternatives $b_{i_1}, b_{i_2}, b_{i_{n\alpha(3)\beta}}$ such that $(i_k + j) \bmod n < 2$.

Let Φ be an assignment computed by $\mathcal{A}$ on I_M. We will show that $\ell_1^\alpha(\Phi) \leq n \cdot \alpha(3) \cdot \beta$ if and only if I is a *yes*-instance of X3C.

($\Leftarrow$) If there exists a solution for I (i.e., an exact cover of U with $\frac{n}{3}$ sets from $\mathcal{F}$), then we can easily show an assignment in which each agent j is assigned to an alternative from the top $m_f(j)$ positions of his or her preference order (namely, one that assigns each agent j to the alternative $a_i \in A_1$ that corresponds to the set F_i, from the exact cover of U, that contains j). Thus, for the optimal assignment Φ_{OPT} it holds that $\ell_1^\alpha(\Phi_{\mathrm{OPT}}) \leq \alpha(3) \cdot n$. In consequence, $\mathcal{A}$ must return an assignment with the total dissatisfaction at most $n \cdot \alpha(3) \cdot \beta$.

($\Rightarrow$) Let us now consider the opposite direction. We assume that $\mathcal{A}$ found an assignment Φ such that $\ell_1^\alpha(\Phi) \leq n \cdot \alpha(3) \cdot \beta$ and we will show that I is a *yes*-instance of X3C. Since we require each alternative to be assigned to either 0 or 3 agents, if some alternative b_i from A_2 were assigned to some 3 agents, at least one of them would rank him or her at a position worse than $n \cdot \alpha(3) \cdot \beta$. This would mean that $\ell_1^\alpha(\Phi) \geq n \cdot \alpha(3) \cdot \beta + 1$. Analogously, no agent j can be assigned to an alternative that is placed at one of the $m_l(j)$ bottom positions of j's preference order. Thus, only the alternatives in A_1 have agents assigned to them and,

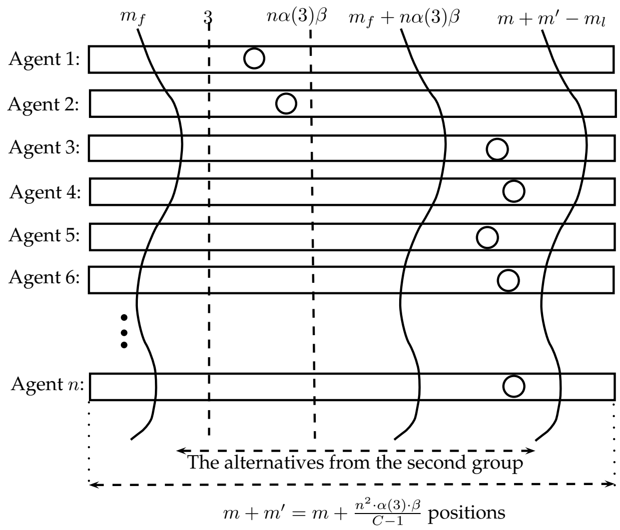

Figure 1: The alignment of the positions in the preference orders of the agents. The positions are numbered from the left to the right. The left wavy line shows the positions $m_f(\cdot)$, each no greater than 3. The right wavy line shows the positions $m_l(\cdot)$, each higher than $n \cdot \alpha(3) \cdot \beta$. The alternatives from A_2 (positions of one such an alternative is illustrated with the circle) are placed only between the peripheral wavy lines. Each alternative from A_2 is placed on the left from the middle wavy line exactly 2 times, thus each such alternative is placed on the left from the right dashed line no more than 2 times (exactly two times at the figure).

further, if agents x, y, z, are assigned to some $a_i \in A_1$, then it holds that $F_i = \{x, y, z\}$ (we will call each set F_i for which alternative a_i is assigned to some agents x, y, z *selected*). Since each agent is assigned to exactly one alternative, the selected sets are disjoint. Since the number of selected sets is $K = \frac{n}{3}$, it must be the case that the selected sets form an exact cover of U. Thus, I is a *yes*-instance of X3C. □

One may wonder if hardness of approximation for α-Monroe-Assignment-Inc is not an artifact of the strict requirements regarding the budget. It turns out that unless P = NP, there is no β-γ-approximation algorithm that finds an assignment with the following properties: (1) the aggregated dissatisfaction $\ell_1^{\alpha}(\Phi)$ is at most β times higher than the optimal one, (2) the number of alternatives to which agents are assigned is at most γK and (3) each selected alternative (the alternative that has agents assigned), is assigned to no more than $\gamma \lceil \frac{n}{K} \rceil$ and no less than $\frac{1}{\gamma} \lceil \frac{n}{K} \rceil$ agents. (The proof is similar to the one used for Theorem 1.) Thus, in our further study we do not consider relaxations of the budget constraints.

Theorem 2. *For each normal IPSF α and each constant β, $\beta > 1$, there is no polynomial-time β-approximation algorithm for α-*Minmax-Monroe-Assignment-Inc *unless* P = NP*.*

Proof. The proof of Theorem 1 applies to this case as well. (In fact, it even suffices to take $m' = \|A_2\| = \frac{n \cdot \alpha(3) \cdot \beta}{2}$). □

Results analogous to Theorems 1 and 2 hold for the CC-Assginment-Inc family of problems as well.

Theorem 3. *For each normal IPSF α and each constant factor β, $\beta > 1$, there is no polynomial-time β-approximation algorithm for α-*CC-Assignment-Inc *unless* P = NP.

Proof. Let us fix a normal IPSF α. For the sake of contradiction, let us assume that there is some constant β, $\beta > 1$, and a β-approximation algorithm $\mathcal{A}$ for α-CC-Assignment-Inc. We will show that it is possible to use $\mathcal{A}$ to solve the NP-complete Vertex-Cover problem.

Let $I = (U, \mathcal{F}, K)$ be an instance of Vertex-Cover, where $U = [n]$ is the ground set, $\mathcal{F} = \{F_1, \dots, F_m\}$ is a family of subsets of U (where each member of U belongs to exactly two sets in $\mathcal{F}$), and K is a positive integer.

Given I, we construct an instance I_{CC} of α-CC-Assignment-Inc in the following way: The set of agents is $N = U$ and the set of alternatives is $A = \bigcup_{j=1}^{m} A_j$, where each A_j contains exactly $\alpha(2) \cdot \beta \cdot n$ (unique) alternatives. Intuitively, for each j, $1 \leq j \leq m$, the alternatives in A_j correspond to the set F_j. For each A_j, $1 \leq j \leq m$, we pick one alternative, which we denote a_j. For each agent $i \in N$, we set i's preference order as follows: Let F_j and F_k, $j < k$, be the two sets that contain i. Agent i's preference order is of the form $a_j \succ_i a_k \succ_i A_k - \{a_k\} \succ_i A - (A_k \cup \{a_j, a_k\})$ (the particular order of alternatives in the sets $A_k - \{a_k\}$ and $A - (A_k \cup \{a_j, a_k\})$ is irrelevant for the construction). We ask for an assignment of the agents to at most K alternatives.

Let us consider a solution Φ returned by $\mathcal{A}$ on input I_{CC}. We claim that $\ell_1^{\alpha}(\Phi) \leq n \cdot \alpha(2) \cdot \beta$ if and only if I is a *yes*-instance of Vertex-Cover.

($\Leftarrow$) If I is a *yes*-instance then, clearly, each agent i can be assigned to one of the top two alternatives in his or her preference order (if there is a size-K cover, then this assignment selects at most K candidates). Thus the total dissatisfaction of an optimal assignment is at most $\alpha(2) \cdot n$. As a result, the solution Φ returned by $\mathcal{A}$ has total dissatisfaction at most $\alpha(2) \cdot \beta \cdot n$.

($\Rightarrow$) If $\mathcal{A}$ returns an assignment with total dissatisfaction no greater than $\alpha(2) \cdot \beta \cdot n$, then, by the construction of agents preference orders, we see that each agent i was assigned to an alternative from a set A_j such that $i \in F_j$. Since the assignment can use at most K alternatives, this directly implies that there is a size-K cover of U with sets from $\mathcal{F}$. □

Theorem 4. *For each normal IPSF α and each constant factor β, $\beta > 1$, there is no polynomial-time β-approximation algorithm for α-*Minmax-CC-Assignment-Inc *unless* P = NP.

Proof. The proof of Theorem 3 works correctly in this case as well. In fact, it even suffices to take the m groups of alternatives, $A_1, \dots, A_m$, to contain $\alpha(2) \cdot \beta$ alternatives each. □

The above results show that approximating the minimal dissatisfaction of agents is difficult. On the other hand, if we focus on agents' total satisfaction then constant-factor approximation exist in many cases (see, e.g., the work of Lu and Boutilier [19] and the next section). Yet, if we focus on the satisfaction of the least satisfied voter, there are no efficient constant-factor approximation algorithms for Monroe's and Courant and Chamberlin's systems. (However, note that our result for the Monroe setting is more general than the result for the Chamberlin-Courant setting; the latter is for the Borda DPSF only.)

Theorem 5. *For each normal DPSF α (where each entry is polynomially bounded in the number of alternatives) and each constant factor β, $0 < \beta \leq 1$, there is no β-approximation algorithm for α-*MINMAX-MONROE-ASSIGNMENT-DEC *unless* P = NP.

Proof. Let us fix a DPSF α where each entry is polynomially bounded in the number of alternatives. For the sake of contradiction, let us assume that for some β, $0 < \beta \leq 1$, there is a β-approximation algorithm $\mathcal{A}$ for α-MINMAX-MONROE-ASSIGNMENT-DEC. We will show that the existence of this algorithm implies that X3C is solvable in polynomial time.

Let I be an X3C instance with ground set $U = \{1, 2, \dots, n\}$ and a collection $\mathcal{F} = \{F_1, \dots, F_m\}$ of subsets of U. Each set in $\mathcal{F}$ has cardinality three. Further, w.l.o.g., we can assume that n is divisible by three and that each $i \in U$ appears in at most three sets from $\mathcal{F}$. Given I, we form an instance I_M of α-MINMAX-MONROE-ASSIGNMENT-DEC as follows. Let $n' = 3 \cdot (\alpha_{dec}^{m+1}(1) \cdot \lceil \frac{1-\beta}{\beta} \rceil + 3)$. The set N of agents is partitioned into two subsets, N_1 and N_2. N_1 contains n agents (intuitively, corresponding to the elements of the ground set U) and N_2 contains n' agents (used to enforce certain properties of the solution). The set of alternatives A is partitioned into two subsets, A_1 and A_2. We set $A_1 = \{a_1, \dots, a_m\}$ (members of A_1 correspond to the sets in $\mathcal{F}$), and we set $A_2 = \{b_1, \dots, b_{m'}\}$, where $m' = \frac{n'}{3}$.

For each j, $1 \leq j \leq n$, we set $M_f(j) = \{a_i \mid j \in F_i\}$. For each j, $1 \leq j \leq n$, we set the preference order of the j'th agent in N_1 to be of the form

$$M_f(j) \succ A_2 \succ A_1 - M_f(j).$$

Note that by our assumptions, $\|M_f(j)\| \leq 3$. For each j, $1 \leq j \leq n'$, we set the preference order of the j'th agent in N_2 to be of the form

$$b_{\lceil \frac{j}{3} \rceil} \succ A_2 - \{b_{\lceil \frac{j}{3} \rceil}\} \succ A_1.$$

Note that each agent in N_2 ranks the alternatives from A_1 in positions positions $m' + 1, \dots, m' + m$. Finally, we set the budget/number of candidates that can be selected, to be $K = \frac{n+n'}{3}$.

Now, consider the solution Φ returned by $\mathcal{A}$ on I_M. We will show that $\ell_{\infty}^{\alpha_{dec}^{m+m'}}(\Phi) \leq \beta\alpha_{dec}^{m+m'}(3)$ if and only if I is a *yes*-instance of X3C.

($\Leftarrow$) If there exists an exact set cover of U with sets from $\mathcal{F}$, then it is easy to construct a solution for I_M where the satisfaction of each agent is greater or equal to $\beta \cdot \alpha_{dec}^{m+m'}(3)$.

Let $I \subseteq \{1, \dots, m\}$ be a set such that $\bigcup_{i \in I} F_i = U$ and $\|I\| = \frac{n}{3}$. We assign each agent j from N_1 to the alternative a_i such that (a) $i \in I$ and (b) $j \in F_i$, and we assign each agent from N_2 to his or her most preferred alternative. Thus, Algorithm $\mathcal{A}$ has to return an assignment with the minimal satisfaction greater or equal to $\beta \cdot \alpha_{dec}^{m+m'}(3)$.

($\Rightarrow$) For the other direction, we first show that $\beta \cdot \alpha_{dec}^{m+m'}(3) \geq \alpha_{dec}^{m+m'}(m')$. Since DPSFs are strictly decreasing, it holds that:

$$\beta \cdot \alpha_{dec}^{m+m'}(3) \geq \beta \cdot (\alpha_{dec}^{m+m'}(m') + m' - 3). \tag{1}$$

Then, by the definition of DPSFs, it holds that:

$$\alpha_{dec}^{m+m'}(m') = \alpha_{dec}^{m+1}(1) \tag{2}$$

Using the fact that $m' = (\alpha_{dec}^{m+1}(1) \cdot \lceil \frac{1-\beta}{\beta} \rceil + 3)$ and using (2), we can transform inequality (1) to obtain the following:

$$\begin{aligned}
\beta \cdot \alpha_{dec}^{m+m'}(3) &\geq \beta \cdot (\alpha_{dec}^{m+m'}(m') + m' - 3) \\
&= \beta \cdot \left(\alpha_{dec}^{m+m'}(m') + (\alpha_{dec}^{m+1}(1) \cdot \left\lceil \frac{1-\beta}{\beta} \right\rceil + 3) - 3 \right) \\
&\geq \beta \cdot \alpha_{dec}^{m+m'}(m') + (1-\beta) \cdot \alpha_{dec}^{m+1}(1) \\
&= \beta \cdot \alpha_{dec}^{m+m'}(m') + (1-\beta) \cdot \alpha_{dec}^{m+m'}(m') = \alpha_{dec}^{m+m'}(m').
\end{aligned}$$

This means that if the minimal satisfaction of an agent is at least $\beta \cdot \alpha_{dec}^{m+m'}(3)$, then no agent was assigned to an alternative that he or she ranked beyond position m'. If some agent j from N_1 were assigned to an alternative from A_2, then, by the pigeonhole principle, some agent from N_2 were assigned to an alternative from A_1. However, each agent in N_2 ranks the alternatives from A_1 beyond position m' and thus such an assignment is impossible. In consequence, it must be that each agent in j was assigned to an alternative that corresponds to a set F_i in $\mathcal{F}$ that contains j. Such an assignment directly leads to a solution for I. ☐

Let us now move on to the case of Minmax-CC-Assignment-Dec family of problems. Unfortunately, in this case our inapproximability argument holds for the case of Borda DPSF only (though we believe that it can be adapted to other DPSFs as well). Further, in our previous theorems we were showing that existence of a respective constant-factor approximation algorithm implies that NP collapses to P. In the following theorem we will show a seemingly weaker collapse of W[2] to FPT.

Intuitively, FPT is a class of problems that can be solved in time $f(k)n^{O(1)}$, where n is the size of the input instance, k is a so-called parameter (some quantity, typically characterizing the difficulty of the instance), and f is some computable function. For example, for the Set-Cover and Vertex-Cover problems one often takes K as the value of the parameter. In the world of parametrized complexity, FPT is viewed as the class of *easy* problems (analogous to the class P), whereas classes $W[1] \subseteq W[2] \subseteq \cdots$ are believed to form a hierarchy of classes of *hard* problems (somewhat analogous to the class NP). It holds that FPT $\subseteq$ W[1], but it seems unlikely that FPT = W[1], let alone FPT = W[2]. We point the reader to the books of Niedermeier [21] and Flum and Grohe [13] for detailed overviews of parametrized complexity theory. Interestingly, while both Set-Cover and Vertex-Cover are NP-complete, the former is W[2]-complete and the latter belongs to FPT (see, e.g., the book of Niedermeier [21] for these now-standard results and their history).

To prove hardness of approximation for $\alpha_{\mathrm{B,dec}}$-Minmax-CC-Assignment-Dec, we first prove the following simple lemma.

Lemma 6. *Let K, p, l be three positive integers and let X be a set of cardinality lpK. There exists a family $\mathcal{S} = \{S_1, \dots, S_{\binom{lK}{K}}\}$ of pK-element subsets of X such that for each K-element subset B of X, there is a set $S_i \in \mathcal{S}$ such that $B \subseteq S_i$.*

Proof. Set $X' = [lK]$ and let Y' be a family of all K-element subsets of X'. Replace each element i of X' with p new elements (at the same time replacing i with the same p elements within each set in Y' that contains i). As a result we obtain two new sets, X and Y, that satisfy the statement of the theorem (up to the renaming of the elements). ☐

Theorem 7. *Let $\alpha^m_{\mathrm{B,dec}}$ be the Borda DPSF ($\alpha^m_{\mathrm{B,dec}}(i) = m - i$). For each constant factor β, $0 < \beta \leq 1$, there is no β-approximation algorithm for $\alpha^m_{\mathrm{B,dec}}$-Minmax-CC-Assignment-Dec unless* FPT = W[2].

Proof. For the sake of contradiction, let us assume that there is some constant β, $0 < \beta \leq 1$, and a polynomial-time β-approximation algorithm $\mathcal{A}$ for $\alpha^m_{\mathrm{B,dec}}$-Minmax-CC-Assignment-Dec. We will show that the existence of this algorithm implies that Set-Cover is fixed-parameter tractable for the parameter K (since Set-Cover is known to be W[2]-complete for this parameter, this will imply FPT = W[2]).

Let I be an instance of Set-Cover with ground set $U = [n]$ and family $\mathcal{F} = \{F_1, F_2, \dots, F_m\}$ of subsets of U. Given I, we build an instance I_{CC} of $\alpha^m_{\mathrm{B,dec}}$-Minmax-CC-Assignment-Dec as follows. The set of agents N consists of n subsets of agents, $N_1, \dots, N_n$, where each group N_i contains exactly $n' = \binom{\lceil \frac{2}{\beta} \rceil K}{K}$ agents. Intuitively, for each i, $1 \leq i \leq n$, the agents in the set N_i correspond to the element i in U. The set of alternatives A is partitioned into two subsets, A_1 and A_2, such that $A_1 = \{a_1, \dots, a_m\}$ is a set of alternatives corresponding to the sets from the family $\mathcal{F}$, and A_2, $\|A_2\| = \lceil \frac{2}{\beta} \rceil \lceil \frac{m(1+\beta)}{K} \rceil K$, is a set of dummy alternatives needed for our construction. We set $m' = \|A\| = m + \|A_2\|$.

Before we describe the preference orders of the agents in N, we form a family $R = \{r_1, \dots, r_{n'}\}$ of preference orders over A_2 that satisfies the following condition: For each K-element subset B of A_2, there exists r_j in R such that all members of B are ranked among the bottom $\lceil \frac{m(1+\beta)}{K} \rceil K$ positions in r_j. By Lemma 6, such a construction is possible (it suffices to take $l = \lceil \frac{2}{\beta} \rceil$ and $p = \lceil \frac{m(1+\beta)}{K} \rceil$); further, the proof of the lemma provides an algorithmic way to construct R.

We form the preference orders of the agents as follows. For each i, $1 \leq i \leq n$, set $M_f(i) = \{a_t \mid i \in F_t\}$. For each i, $1 \leq i \leq n$, and each j, $1 \leq j \leq n'$, the j'th agent from N_i

has preference order of the form:

$$M_f(i) \succ r_j \succ A_1 - M_f(i)$$

(we pick any arbitrary, polynomial-time computable order of candidates within $M_f(i)$ and $M_l(i)$).

Let Φ be an assignment computed by $\mathcal{A}$ on I_M. We will show that $\ell_{\infty}^{\alpha^{m'}_{\mathrm{B,dec}}}(\Phi) \geq \beta \cdot (m'-m)$ if and only if I is a *yes*-instance of Set-Cover.

($\Leftarrow$) If there exists a solution for I (i.e., a cover of U with K sets from $\mathcal{F}$), then we can easily show an assignment where each agent is assigned to an alternative that he or she ranks among the top m positions (namely, for each j, $1 \leq j \leq n$, we assign all the agents from the set N_j to the alternative $a_i \in A_1$ such that $j \in F_i$ and F_i belongs to the alleged K-element cover of U). Under this assignment, the least satisfied agent's satisfaction is at least $m' - m$ and, thus, $\mathcal{A}$ has to return an assignment Φ where $\ell_{\infty}^{\alpha^{m'}_{\mathrm{B,dec}}}(\Phi) \geq \beta \cdot (m' - m)$.

($\Rightarrow$) Let us now consider the opposite direction. We assume that $\mathcal{A}$ found an assignment Φ such that $\ell_{\infty}^{\alpha^{m}_{\mathrm{B,dec}}}(\Phi) \geq \beta \cdot (m' - m)$ and we will show that I is a *yes*-instance of Set-Cover. We claim that for each i, $1 \leq i \leq n$, at least one agent j in N_i were assigned to an alternative from A_1. If all the agents in N_i were assigned to alternatives from A_2, then, by the construction of R, at least one of them would have been assigned to an alternative that he or she ranks at a position greater than $\|A_2\| - \left\lceil \frac{m(1+\beta)}{K} \right\rceil K = \left\lceil \frac{2}{\beta} \right\rceil \left\lceil \frac{m(1+\beta)}{K} \right\rceil K - \left\lceil \frac{m(1+\beta)}{K} \right\rceil K$. Since for $x = \left\lceil \frac{m(1+\beta)}{K} \right\rceil K$ we have:

$$\left\lceil \frac{2}{\beta} \right\rceil x - x \geq m' - m'\beta + m\beta$$

(we skip the straightforward calculation). This means that this agent would have been assigned to an alternative that he or she ranks at a position greater than $m' - m'\beta + m\beta$. As a consequence, this agent's satisfaction would be lower than $(m' - m)\beta$. Similarly, no agent from N_i can be assigned to an alternative from $M_l(i)$. Thus, for each i, $1 \leq i \leq n$, there exists at least one agent $j \in N_i$ that is assigned to an alternative from $M_f(i)$. In consequence, the covering subfamily of $\mathcal{F}$ consists simply of those sets F_k, for which some agent is assigned to alternative $a_k \in A_1$.

The presented construction gives the exact algorithm for Set-Cover problem running in time $f(K)(n + m)^{O(1)}$, where $f(K)$ is polynomial in $\binom{\lceil \frac{2}{\beta} \rceil}{K}$. The existence of such an algorithm means that Set-Cover is in FPT. On the other hand, we know that Set-Cover is W[2]-complete, and thus if $\mathcal{A}$ existed then FPT = W[2] would hold. ◻

4 Approximation Algorithms

We now turn to approximation algorithms for Monroe's and Chamberlin and Courant's multiwinner voting rules (both of which are special cases of our resource allocation problem).

Indeed, if one focuses on agents' total satisfaction then it is possible to obtain high-quality approximation results. In particular, we show the first nontrivial (randomized) approximation algorithm for $\alpha_{\mathrm{B,dec}}$-Monroe-Assignment-Dec (for each $\epsilon > 0$, we can provide a randomized polynomial-time algorithm that achieves $0.715 - \epsilon$ approximation ratio), and the first polynomial-time approximation scheme (PTAS) for $\alpha_{\mathrm{B,dec}}$-CC-Assignment-Dec. These results stand in a sharp contrast to those from the previous section, where we have shown that approximation is hard for essentially all remaining variants of the problem.

The core difficulty in solving α-Monroe/CC-Assignment problems lays in selecting the alternatives that should be assigned to the agents. Given a preference profile and a set A' of K alternatives, using a standard network-flow argument, it is easy to match them optimally to the agents.

Proposition 8 (Implicit in the paper of Betzler et al. [4]). *Let α be a normal DPSF, N be a set of agents, A be a set of alternatives, V be a preference profile of N over A, and A' a K-element subset of A (where K divides $\|N\|$). There is a polynomial-time algorithm that computes an optimal assignment Φ of the alternatives from A' to the agents, both for ℓ_1^α and for $\min^\alpha$, both for the case where each alternative in A' should be assigned to the same number of agents (Monroe's case) and for the case without additional restrictions (Chamberlin and Courant's case).*

Thus, in the algorithms in this section, we will focus on the issue of selecting the alternatives and not on the issue of matching them to the agents.

4.1 Monroe's System

We first consider $\alpha_{\mathrm{B,dec}}$-Monroe-Assignment-Dec. Perhaps the most natural approach to solve this problem is to build a solution iteratively: In each step we pick some not-yet-assigned alternative a_i (using some criterion) and assign him or her to those $\lceil \frac{N}{K} \rceil$ agents that (a) are not assigned to any other alternative yet, and (b) whose satisfaction of being matched with a_i is maximal. It turns out that this idea, implemented formally in Algorithm 1, works very well in many cases. We provide a lower bound on the total satisfaction it guarantees in the next lemma. (For each positive integer k, we let $H_k = \sum_{i=1}^{k} \frac{1}{i}$ be the k'th harmonic number. Recall that $H_k = \Theta(\log k)$.)

Lemma 9. *Algorithm 1 is a polynomial-time $(1 - \frac{K-1}{2(m-1)} - \frac{H_K}{K})$-approximation algorithm for $\alpha_{\mathrm{B,dec}}$-Monroe-Assignment-Dec.*

Proof. Our algorithm computes an optimal solution for $K \leq 2$. Thus we assume $K \geq 3$. Let us consider the situation in the algorithm after the i'th iteration of the outer loop (we have $i = 0$ if no iteration has been executed yet). So far, the algorithm has picked i alternatives and assigned them to $i\frac{n}{K}$ agents (recall that for simplicity we assume that K divides n evenly). Hence, each agent has $\lceil \frac{m-i}{K-i} \rceil$ unassigned alternatives among his or her $i + \lceil \frac{m-i}{K-i} \rceil$ top-ranked alternatives. By pigeonhole principle, this means that there is an unassigned alternative a_ℓ who is ranked among top $i + \lceil \frac{m-i}{K-i} \rceil$ positions by at least $\frac{n}{K}$ agents. To see

Algorithm 1: The algorithm for MONROE-ASSIGNMENT.

Notation: $\Phi \leftarrow$ a map defining a partial assignment, iteratively built by the algorithm.
$\Phi^{\leftarrow} \leftarrow$ the set of agents for which the assignment is already defined.
$\Phi^{\rightarrow} \leftarrow$ the set of alternatives already used in the assignment.

```
if K ≤ 2 then
    compute the optimal solution using an algorithm of Betzler et al. [4] and return.
Φ = {}
for i ← 1 to K do
    score ← {}
    bests ← {}
    foreach a_i ∈ A \ Φ→ do
        agents ← sort N \ Φ← so that j ≺ k in agents ⟹
                pos_j(a_i) ≤ pos_k(a_i)
        bests[a_i] ← chose first ⌈N/K⌉ elements from agents
        score[a_i] ← Σ_{j∈bests} (m − pos_j(a_i))
    a_best ← argmax_{a∈A\Φ→} score[a]
    foreach j ∈ bests[a_best] do
        Φ[j] ← a_best
```

this, note that there are $(n - i\frac{n}{K})\lceil\frac{m-i}{K-i}\rceil$ slots for unassigned alternatives among the top $i + \lceil\frac{m-i}{K-i}\rceil$ positions in the preference orders of unassigned agents, and that there are $m - i$ unassigned alternatives. As a result, there must be an alternative a_ℓ for whom the number of agents that rank him or her among the top $i + \lceil\frac{m-i}{K-i}\rceil$ positions is at least:

$$\frac{1}{m-i}\left((n - i\frac{n}{K})\lceil\frac{m-i}{K-i}\rceil\right) \geq \frac{n}{m-i}\left(\frac{K-i}{K}\right)\left(\frac{m-i}{K-i}\right) = \frac{n}{K}.$$

In consequence, the $\lceil\frac{n}{K}\rceil$ agents assigned in the next step of the algorithm will have the total satisfaction at least $\lceil\frac{n}{K}\rceil \cdot (m - i - \lceil\frac{m-i}{K-i}\rceil)$. Thus, summing over the K iterations, the total satisfaction guaranteed by the assignment Φ computed by Algorithm 1 is at least the following value (see the comment below for the fourth inequality; for the last inequality we assume $K \geq 3$):

$$\begin{aligned}
\ell_1^{\alpha_b}(\Phi) &\geq \sum_{i=0}^{K-1} \frac{n}{K} \cdot \left(m - i - \lceil\frac{m-i}{K-i}\rceil\right) \\
&\geq \sum_{i=0}^{K-1} \frac{n}{K} \cdot \left(m - i - \frac{m-i}{K-i} - 1\right) \\
&= \sum_{i=1}^{K} \frac{n}{K} \cdot \left(m - i - \frac{m-1}{K-i+1} + \frac{i-2}{K-i+1}\right) \\
&= \frac{n}{K}\left(\frac{K(2m-K-1)}{2} - (m-1)H_K + K(H_K - 1) - H_K\right)
\end{aligned}$$

$$= (m-1)n\left(1 - \frac{K-1}{2(m-1)} - \frac{H_K}{K} + \frac{H_K - 1}{m-1} - \frac{H_K}{K(m-1)}\right)$$
$$> (m-1)n\left(1 - \frac{K-1}{2(m-1)} - \frac{H_K}{K}\right)$$

The fourth equality holds because:

$$K(H_K - 1) - H_K = \sum_{i=1}^{K}\left(\frac{K}{i} - 1\right) - H_K = \sum_{i=1}^{K}\left(\frac{K}{K-i+1} - 1\right) - H_K$$
$$= \sum_{i=1}^{K}\frac{i-1}{K-i+1} - H_K = \sum_{i=1}^{K}\frac{i-2}{K-i+1}.$$

If each agent were assigned to his or her top alternative, the total satisfaction would be equal to $(m-1)n$. Thus we get the following bound:

$$\frac{\ell_1^{\alpha_{\mathrm{B,dec}}}(\Phi)}{\mathrm{OPT}} \leq 1 - \frac{K-1}{2(m-1)} - \frac{H_K}{K}.$$

This completes the proof. □

Note that in the above proof we measure the quality of our assignment against a perhaps-impossible, perfect solution, where each agent is assigned to his or her top alternative. This means that for relatively large m and K, and small $\frac{K}{m}$ ratio, the algorithm can achieve a close-to-ideal solution irrespective of the voters' preference orders. We believe that this is an argument in favor of using Monroe's system in multiwinner elections. On the flip side, to obtain a better approximation ratio, we would have to use a more involved bound on the quality of the optimal solution. To see that this is the case, form an instance I of $\alpha_{\mathrm{B,dec}}$-MONROE-ASSIGNMENT-DEC with n agents and m alternatives, where all the agents have the same preference order, and where the budget is K (and where K divides n). It is easy to see that each solution that assigns the K universally top-ranked alternatives to the agents is optimal. Thus the total dissatisfaction of the agents in the optimal solution is:

$$\frac{n}{K}\left((m-1) + \cdots + (m-K)\right) = \frac{n}{K}\left(\frac{K(2m-K-1)}{2}\right) = n(m-1)\left(1 - \frac{K-1}{2(m-1)}\right).$$

By taking large enough m and K (even for a fixed value of $\frac{m}{K}$), the fraction $1 - \frac{K-1}{2(m-1)}$ can be arbitrarily close to the approximation ratio of our algorithm (the reasoning here is somewhat in spirit of the idea of identifying maximally robust elections, as studied by Shiryaev, Yu, and Elkind [28]).

Betzler et al. [4] showed that for each fixed constant K, $\alpha_{\mathrm{B,dec}}$-MONROE-ASSIGNMENT-DEC can be solved in polynomial time. Thus, for small values of K for which the fraction $\frac{H_K}{K}$ affects the approximation guarantees of Algorithm 1 too much, we can use this polynomial-time algorithm to find an optimal solution. This means that we can essentially disregard the

$\frac{H_K}{K}$ part of Algorithm 1's approximation ratio. In consequence, the quality of the solution produced by Algorithm 1 most strongly depends on the ratio $\frac{K-1}{m-1}$. In most cases we can expect it to be small (for example, in Polish parliamentary elections $K = 460$ and $m \approx 6000$; in this case the greedy algorithm's approximation ratio is about 0.96). For the remaining cases, for example, when $K > \frac{m}{2}$, we can use a simple sampling-based randomized algorithm described below.

The idea of this algorithm is to randomly pick K alternatives and match them optimally to the agents, using Proposition 8. Naturally, such an algorithm might be very unlucky and pick K alternatives that all of the agents rank low. Yet, if K is large relative to m then it is likely that such a random sample would include a large chunk of some optimal solution. In the lemma below, we asses the expected satisfaction obtained with a single sampling step (relative to the satisfaction given by the optimal solution) and the probability that a single sampling step gives satisfaction close to the expected one. Naturally, in practice one should try several sampling steps and pick the one with the highest satisfaction.

Lemma 10. *A single sampling step of the randomized algorithm for* $\alpha_{\mathrm{B,dec}}$-Monroe-Assignment-Dec *achieves expected approximation ratio of* $\frac{1}{2}(1+\frac{K}{m}-\frac{K^2}{m^2-m}+\frac{K^3}{m^3-m^2})$. *Let* p_ϵ *denote the probability that the relative deviation between the obtained total satisfaction and the expected total satisfaction is higher than* ϵ*; for* $K \geq 8$ *we have* $p_\epsilon \leq \exp\left(-\frac{K\epsilon^2}{128}\right)$.

Proof. Let $N = [n]$ be the set of agents, $A = \{a_1, \dots, a_m\}$ be the set of alternatives, and V be the preference profile of the agents. Let us fix some optimal solution Φ_{opt} and let A_{opt} be the set of alternatives assigned to the agents in this solution. For each $a_i \in A_{\mathrm{opt}}$, we write $\mathrm{sat}(a_i)$ to denote the total satisfaction of the agents assigned to a_i in Φ_{opt}. Naturally, we have $\sum_{a \in A_{\mathrm{opt}}} \mathrm{sat}(a) = \mathrm{OPT}$. In a single sampling step, we choose uniformly at random a K-element subset B of A. Then, we form a solution Φ_B by matching the alternatives in B optimally to the agents (via Proposition 8). We write K_{opt} to denote the random variable equal to $\|A_{\mathrm{opt}} \cap B\|$, the number of sampled alternatives that belong to A_{opt}. We define $p_i = \Pr(K_{\mathrm{opt}} = i)$. For each $j \in \{1, \dots, K\}$, we write X_j to denote the random variable equal to the total satisfaction of the agents assigned to the j'th alternative from the sample. We claim that for each i, $0 \leq i \leq K$, it holds that:

$$\mathbb{E}\left(\sum_{j=1}^{K} X_j | K_{\mathrm{opt}} = i\right) \geq \frac{i}{K}\mathrm{OPT} + \frac{m-i-1}{2} \cdot (n - i\frac{n}{K}).$$

Why is this so? Given sample B that contains i members of A_{opt}, our algorithm's solution is at least as good as a solution that matches the alternatives from $B \cap A_{\mathrm{opt}}$ in the same way as Φ_{opt}, and the alternatives from $B - A_{\mathrm{opt}}$ in a random manner. Since $K_{\mathrm{opt}} = i$ and each $a_j \in A_{\mathrm{opt}}$ has equal probability of being in the sample, it is easy to see that the expected value of $\sum_{a_j \in B \cap A_{\mathrm{opt}}} \mathrm{sat}(a_j)$ is $\frac{i}{K}\mathrm{OPT}$. After we allocate the agents from $B \cap A_{\mathrm{opt}}$, each of the remaining, unassigned agents has $m - i$ positions in his or her preference order where he ranks the agents from $A - A_{\mathrm{opt}}$. For each unassigned agents, the average score value associated with these positions is at least $\frac{m-i-1}{2}$ (this is so, because in the worst case the agent could rank the alternatives from $B \cap A_{\text{opt}}$ in the top i positions). There are $(n - i\frac{n}{K})$ such not yet assigned agents and so the expected total satisfaction from assigning them randomly to the alternatives is $\frac{m-i-1}{2} \cdot (n - i\frac{n}{K})$. This proves our bound on the expected satisfaction of a solution yielded by optimally matching a random sample of K alternatives.

Since OPT is upper bounded by $(m-1)n$ (consider a possibly-nonexistent solution where every agent is assigned to his or her top preference), we get that:

$$\mathbb{E}\left(\sum_{j=1}^{K} X_j | K_{\text{opt}} = i\right) \geq \frac{i}{K}\text{OPT} + \frac{m-i-1}{2(m-1)} \cdot (1 - \frac{i}{K})\text{OPT}.$$

We can compute the unconditional expected satisfaction of Φ_B as follows:

$$\begin{aligned}\mathbb{E}\left(\sum_{j=1}^{K} X_j\right) &= \sum_{i=0}^{K} p_i \, \mathbb{E}\left(\sum_{j=1}^{K} X_j | K_{\text{opt}} = i\right) \\ &\geq \sum_{i=0}^{K} p_i \left(\frac{i}{K}\text{OPT} + \frac{m-i-1}{2(m-1)} \cdot (1 - \frac{i}{K})\text{OPT}\right).\end{aligned}$$

Since $\sum_{i=1}^{K} p_i \cdot i$ is the expected number of the alternatives in A_{opt}, we have that $\sum_{i=1}^{K} p_i \cdot i = \frac{K^2}{m}$. (one can think of summing the expected values of K indicator random variables; one for each element of A_{opt}, taking the value 1 if a given alternative is selected and taking the value 0 otherwise). Further, from the generalized mean inequality we obtain $\sum_{i=1}^{K} p_i \cdot i^2 \geq \left(\frac{K^2}{m}\right)^2$. In consequence, through routine calculation, we get that:

$$\begin{aligned}\mathbb{E}\left(\sum_{j=1}^{K} X_j\right) &\geq \left(\frac{K}{m}\text{OPT} + \frac{m^2 - K^2 - m}{2m(m-1)} \cdot \left(1 - \frac{K}{m}\right)\text{OPT}\right) \\ &= \frac{\text{OPT}}{2}\left(1 + \frac{K}{m} - \frac{K^2}{m^2 - m} + \frac{K^3}{m^3 - m^2}\right).\end{aligned}$$

It remains to assess the probability that the total satisfaction obtained through Φ_B is close to its expected value. Since $X_j \in \langle 0, \frac{(m-1)n}{K} \rangle$, from Hoeffding's inequality we get:

$$\begin{aligned}p_\epsilon &= \Pr\left(\left|\sum_{j=1}^{K} X_j - \mathbb{E}(\sum_{j=1}^{K} X_j)\right| \geq \epsilon \, \mathbb{E}(\sum_{j=1}^{K} X_j)\right) \\ &\leq \exp\left(-\frac{2\epsilon^2(\mathbb{E}(\sum_{j=1}^{K} X_j))^2}{K(\frac{(m-1)n}{K})^2}\right) = \exp\left(-\frac{K\epsilon^2(\mathbb{E}(\sum_{j=1}^{K} X_j))^2}{((m-1)n)^2}\right)\end{aligned}$$

We note that since $\frac{K}{m} - \frac{K^2}{m^2-m} \geq 0$, our previous calculations show that $\mathbb{E}(\sum_{j=1}^{K} X_j) \geq \frac{\text{OPT}}{2}$. Further, for $K \geq 8$, Lemma 9 (and the fact that in its proof we upper-bound OPT to be $(m-1)n$) gives that $\text{OPT} \geq \frac{mn}{8}$. Thus $p_\epsilon \leq \exp\left(-\frac{K\epsilon^2}{128}\right)$. This completes the proof. □

The threshold for $\frac{K}{m}$, where the randomized algorithm is (in expectation) better than the greedy algorithm is about 0.57. Thus, by combining the two algorithms, we can guarantee an expected approximation ratio of $0.715 - \epsilon$, for each fixed constant ϵ. The pseudo-code of the combination of the two algorithms is presented in Algorithm 2.

Theorem 11. *For each fixed ϵ, Algorithm 2 provides a $(0.715-\epsilon)$-approximate solution for the problem $\alpha_{\mathrm{B,dec}}$-Monroe-Assignment-Dec with probability λ in time polynomial with respect to the input instance size and $-\log(1-\lambda)$.*

Proof. Let ϵ be a fixed constant. We are given an instance I of $\alpha_{\mathrm{B,dec}}$-Monroe-Assignment-Dec. If $m \leq 1 + \frac{2}{\epsilon}$, we solve I using a brute-force algorithm (note that in this case the number of alternatives is at most a fixed constant). Similarly, if $\frac{H_K}{K} \geq \frac{\epsilon}{2}$ then we use the exact algorithm of Betzler et al. [4] for a fixed value of K (note that in this case K is no greater than a certain fixed constant). We do the same if $K \leq 8$.

On the other hand, if neither of the above conditions hold, we try both Algorithm 1 and a number of runs of the sampling-based algorithm. It is easy to check through routine calculation that if $\frac{H_K}{K} \leq \frac{\epsilon}{2}$ and $m > 1 + \frac{2}{\epsilon}$ then Algorithm 1 achieves approximation ratio no worse than $(1 - \frac{K}{2m} - \epsilon)$. We run the sampling-based algorithm $\frac{-512\log(1-\lambda)}{K\epsilon^2}$ times. The probability that a single run fails to find a solution with approximation ratio at least $\frac{1}{2}(1 + \frac{K}{m} - \frac{K^2}{m^2-m} + \frac{K^3}{m^3-m^2}) - \frac{\epsilon}{2}$ is $p_{\frac{\epsilon}{2}} \leq \exp\left(-\frac{K\epsilon^2}{4\cdot 128}\right)$. Thus, the probability that at least one run will find a solution with at least this approximation ratio is at least:

$$1 - p_{\frac{\epsilon}{2}}^{\frac{-512\log(1-\lambda)}{K\epsilon^2}} = 1 - \exp\left(-\frac{K\epsilon^2}{4\cdot 128}\cdot\frac{-512\log(1-\lambda)}{K\epsilon^2}\right) = \lambda.$$

Since $m \leq 1 + \frac{2}{\epsilon}$, by routine calculation we see that the sampling-based algorithm with probability λ finds a solution with approximation ratio at least $\frac{1}{2}(1 + \frac{K}{m} - \frac{K^2}{m^2} + \frac{K^3}{m^3}) - \epsilon$. By solving the equality:

$$\frac{1}{2}\left(1 + \frac{K}{m} - \frac{K^2}{m^2} + \frac{K^3}{m^3}\right) = 1 - \frac{K}{2m}$$

we can find the value of $\frac{K}{m}$ for which the two algorithms give the same approximation ratio. By substituting $x = \frac{K}{m}$ we get equality $1 + x - x^2 + x^3 = 2 - x$. One can calculate that this equality has a single solution within $\langle 0, 1\rangle$ and that this solution is $x \approx 0.57$. For this x both algorithms guarantee approximation ratio of $0.715 - \epsilon$. For $x < 0.57$ the deterministic algorithm guarantees a better approximation ratio and for $x > 0.57$, the randomized algorithm does better. □

4.2 Chamberlin and Courant's System

Let us now move on to the Chamberlin and Courant's system. Since Chamberlin and Courant's system places fewer restrictions on the solution assignment than the Monroe's system, our algorithms for $\alpha_{\mathrm{B,dec}}$-Monroe-Assignement-Dec can also be used

Algorithm 2: The combination of two algorithms for MONROE-ASSIGNMENT.

Notation: We use the same notation as in Algorithm 1; $\mathrm{w}(\cdot)$ denotes the Lambert W-Function.

Parameters: $\lambda \leftarrow$ required probability of achieving the approximation ratio equal $0.715 - \epsilon$

if $\frac{H_K}{K} \geq \frac{\epsilon}{2}$ *or* $K \leq 8$ **then**

  compute the optimal solution using an algorithm of Betzler et al. [4] and return.

if $m \leq 1 + \frac{2}{\epsilon}$ **then**

  compute the optimal solution using a simple brute force algorithm and return.

$\Phi_1 \leftarrow$ solution returned by Algorithm 1

$\Phi_2 \leftarrow$ run the sampling-based algorithm $\frac{-512 \log(1-\lambda)}{K\epsilon^2}$ times; select the assignment of the best quality

return the better assignment among Φ_1 and Φ_2

Algorithm 3: The algorithm for CC-ASSIGNMENT.

Notation: We use the same notation as in Algorithm 2; $\mathrm{num\_pos}_x(a) \leftarrow \|\{i \in [n] \setminus \Phi^{\leftarrow} : pos_i(a) \leq x\}\|$ (the number of not-yet assigned agents that rank alternative a in one of their first x positions)

$\Phi = \{\}$

$x = \lceil \frac{m\mathrm{w}(K)}{K} \rceil$

for $i \leftarrow 1$ **to** K **do**

  $a_i \leftarrow \mathrm{argmax}_{a \in A \setminus \Phi^{\rightarrow}} \mathrm{num\_pos}_x(a)$

  foreach $j \in [n] \setminus \Phi^{\leftarrow}$ **do**

    if $pos_j(a_i) < x$ **then**

      $\Phi[j] \leftarrow a_i$

foreach $j \in A \setminus \Phi^{\leftarrow}$ **do**

  $a \leftarrow$ such server from $\Phi^{\rightarrow}$ that $\forall_{a' \in \Phi^{\rightarrow}} pos_j(a) \leq pos_j(a')$

  $\Phi[j] \leftarrow a$

for $\alpha_{\mathrm{B,dec}}$-CC-ASSIGNEMENT-DEC, improving upon the approximation algorithms of Lu and Boutilier [19]. However, it turns out that the additional freedom of Chamberlin and Courant's system allows us to go even further, and to design a polynomial-time approximation scheme for $\alpha_{\mathrm{B,dec}}$-CC-ASSIGNEMENT-DEC.

The idea of our algorithm (presented as Algorithm 3 below) is to compute a certain value x and to greedily compute an assignment that (approximately) maximizes the number of agents assigned to one of their top-x alternatives. If after this process some agent has no alternative assigned, we assign him or her to his or her most preferred alternative from those already picked. Somewhat surprisingly, it turns out that this greedy strategy achieves high-quality results. (Recall that for nonnegative real numbers, Lambert's W function, $\mathrm{w}(x)$, is defined to be the solution of the equality $x = \mathrm{w}(x)e^{\mathrm{w}(x)}$.)

Lemma 12. *Algorithm 3 is a polynomial-time* $(1 - \frac{2\mathrm{w}(K)}{K})$*-approximation algorithm for* $\alpha_{\mathrm{B,dec}}$*-CC-ASSIGNEMENT-DEC.*

Proof. Let $x = \frac{m\mathrm{w}(K)}{K}$. We will first give an inductive proof that, for each i, $0 \leq i \leq K$, after the i'th iteration of the outer loop at most $n(1-\frac{w(K)}{K})^i$ agents are unassigned. Based on this observation, we will derive the approximation ratio of our algorithm.

For $i = 0$, the inductive hypothesis holds because $n(1-\frac{\mathrm{w}(K)}{K})^0 = n$. For each i, let n_i denote the number of unassigned agents after the i'th iteration. Thus, after the i'th iteration there are n_i unassigned agents, each with x unassigned alternatives among his or her top-x ranked alternatives. As a result, at least one unassigned alternative is present in at least $\frac{n_i x}{m-i}$ of top-x positions of unassigned agents. This means that after the $(i+1)$'st iteration the number of unassigned agents is:

$$n_{i+1} \leq n_i - \frac{n_i x}{m-i} \leq n_i\left(1-\frac{x}{m}\right) = n_i\left(1-\frac{\mathrm{w}(K)}{K}\right).$$

If for a given i the inductive hypothesis holds, that is, if $n_i \leq n\left(1-\frac{\mathrm{w}(K)}{K}\right)^i$, then:

$$n_{i+1} \leq n(1-\frac{\mathrm{w}(K)}{K})^i(1-\frac{\mathrm{w}(K)}{K}) = n(1-\frac{\mathrm{w}(K)}{K})^{i+1}$$

Thus the hypothesis holds and, as a result, we have that:

$$n_k \leq n\left(1-\frac{\mathrm{w}(K)}{K}\right)^K \leq n\left(\frac{1}{e}\right)^{\mathrm{w}(K)} = \frac{n\mathrm{w}(K)}{K}.$$

Let Φ be the assignment computed by our algorithm. To compare it against the optimal solution, it suffices to observe that the optimal solution has satisfaction at most $\mathrm{OPT} \leq (m-1)n$, that each agent selected during the first K steps has satisfaction at least $m - x = m - \frac{m\mathrm{w}(K)}{K}$, and that the agents not assigned within the first K steps have satisfaction no worse than 0. Thus it holds that:

$$\begin{aligned}\frac{\ell_1^{\alpha_{\mathrm{B,dec}}}(\Phi)}{\mathrm{OPT}} &\geq \frac{(n-\frac{n\mathrm{w}(K)}{K})(m-\frac{m\mathrm{w}(K)}{K})}{(m-1)n} \\ &\geq (1-\frac{\mathrm{w}(K)}{K})(1-\frac{\mathrm{w}(K)}{K}) \geq 1-\frac{2\mathrm{w}(K)}{K}\end{aligned}$$

This completes the proof. □

Since for each $\epsilon > 0$ there is a value K_ϵ such that for each $K > K_\epsilon$ it holds that $\frac{2\mathrm{w}(K)}{K} < \epsilon$, and $\alpha_{\mathrm{B,dec}}$-CC-Assignment problem can be solved optimally in polynomial time for each fixed constant K(see the work of Betzler et al. [4]), there is a polynomial-time approximation scheme (PTAS) for $\alpha_{\mathrm{B,dec}}$-CC-Assignment (i.e., a family of algorithms such that for each fixed β, $0 < \beta < 1$, there is a polynomial-time β-approximation algorithm for $\alpha_{\mathrm{B,dec}}$-CC-Assignment in the family; note that in PTASes we measure the running time by considering β to be a fixed constant).

Theorem 13. *There is a PTAS for* $\alpha_{\mathrm{B,dec}}$-CC-Assignment.

	Monroe		Chamberlin and Courant		General model	
	dissat.	satisfaction	dissat.	satisfaction	dissat.	satisfaction
utilitarian	Inapprox. Theorem 1	Results for Borda only. randomized algorithm: (a) $0.715 - \epsilon$; (b) $\frac{1+\frac{K}{m}-\frac{K^2}{m^2-m}+\frac{K^3}{m^3-m^2}}{2} - \epsilon$ deterministic algorithm: $1 - \frac{K-1}{2(m-1)} - \epsilon$	Inapprox. Theorem 3	for Borda: PTAS (Sec. 4.2); for general PSF: $(\frac{e-1}{e})$ [19]	Inapprox. Theorem 1 Theorem 3	Open problem
egal.	Inapprox. Theorem 2	Inapprox. Theorem 5	Inapprox. Theorem 4	Inapprox. Theorem 7	Inapprox. Theorem 2 Theorem 4	Inapprox. Theorem 5 Theorem 7

Table 1: Summary of approximation results fo Monroe's and Chamberlin-Courant's multiwinner voting systems and for our general resource allocation problem.

The idea used in Algorithm 3 can also be used to address a generalized Minmax-CC-Assignment-Dec problem. We can consider the following relaxation of Minmax-CC-Assignment-Dec: Instead of requiring that each agent's satisfaction is lower-bounded by some value, we ask that the satisfactions of a significant majority of the agents are lower-bounded by a given value. More formally, for a given constant δ, we introduce an additional quality metric:

$$\overset{\alpha}{\underset{\delta}{\min}}(\Phi) = \max_{N' \subseteq N : \frac{||N|| - ||N'||}{||N||} \leq \delta} \min_{i \in N'} \alpha(pos_i(\Phi(i))).$$

For a given $0 < \delta < 1$, by putting $x = \frac{-m \ln(\delta)}{K}$, we get $(1 + \frac{\ln(\delta)}{K})$-approximation algorithm for the $\min_{\delta}^{\alpha}(\Phi)$ metric.

It would also be natural to try a sampling-based approach to solving $\alpha_{\mathrm{B,dec}}$-CC-Assignement-Dec, just as we did for the Monroe variant. Indeed, as recently (and independently) observed by Oren [23], this leads to a randomized algorithm with expected approximation ratio of $(1 - \frac{1}{K+1})(1 + \frac{1}{m})$.

5 Conclusions

We have defined a certain resource allocation problem and have shown that it generalizes multiwinner voting rules of Monroe and of Chamberlin and Courant. Since it is known that these voting rules are hard to compute [4,19,25], we focused on approximation properties of our problems. We have shown that if we focus on agents' dissatisfaction then our problems are hard to approximate up to any constant factor. The same holds for the case where we focus on least satisfied agent's satisfaction. However, for the case of optimizing total satisfaction, we have shown good approximate solutions for the special cases of the resource allocation problem pertaining to Monroe's and Chamberlin and Courant's voting systems. In particular, we have shown a randomized algorithm that for Monroe's system achieves approximation ratio arbitrarily close to 0.715, and for Chamberlin and Courant's system we have shown a polynomial-time approximation scheme (PTAS). Table 1 summarizes the presented results. The most pressing, natural open question is whether there is a PTAS for Monroe's system. We plan to address the problem of finding approximate solution in the general framework for the case of optimizing total satisfaction and to empirically verify the quality of our algorithms.

Acknowledgements The authors were supported in part by AGH University of Science and Technology grant 11.11.120.865, by the Foundation for Polish Science's Homing/Powroty program, and by EU's Human Capital Program "National PhD Programme in Mathematical Sciences" carried out at the University of Warsaw.

References

[1] N. Ailon, M. Charikar, and A. Newman. Aggregating inconsistent information: Ranking and clustering. *Journal of the ACM*, 55(5):Article 23, 2008.

[2] S. Airiau and U. Endriss. Multiagent resource allocation with sharable items: Simple protocols and nash equilibria. In *Proceedings of AAMAS-2010*, pages 167–174, May 2010.

[3] J. Bartholdi, III, C. Tovey, and M. Trick. Voting schemes for which it can be difficult to tell who won the election. *Social Choice and Welfare*, 6(2):157–165, 1989.

[4] N. Betzler, A. Slinko, and J. Uhlmann. On the computation of fully proportional representation. Technical Report SSRN: http://ssrn.com/abstract=1952497, U. of Auckland, November 2011.

[5] M. Brill and F. Fisher. The price of neutrality for the ranked pairs method. In *Proceedings of AAAI-2012*, 2012. To appear.

[6] I. Caragiannis, J. Covey, M. Feldman, C. Homan, C. Kaklamanis, N. Karanikolas, A. Procaccia, and J. Rosenschein. On the approximability of Dodgson and Young elections. In *Proceedings of SODA-2009*, pages 1058–1067, January 2009.

[7] I. Caragiannis, C. Kaklamanis, N. Karanikolas, and A. Procaccia. Socially desirable approximations for Dodgson's voting rule. In *Proceedings of EC-2010*, pages 253–262, June 2010.

[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] Y. Chevaleyre, P. Dunne, U. Endriss, J. Lang, M. Lematre, N. Maudet, J. Padget, S. Phelps, J. Rodríguez-Aguilar, and P. Sousa. Issues in multiagent resource allocation. *Informatica*, 30:3–31, 2006.

[10] D. Coppersmith, L. Fleisher, and A. Rurda. Ordering by weighted number of wins gives a good ranking for weighted tournaments. *ACM Transactions on Algorithms*, 6(3):Article 55, 2010.

[11] A. Darmann, E. Elkind, S. Kurz, J. Lang, J. Schauer, and G. Woeginger. Group activity selection problem. In *Workshop Notes of COMSOC-2012*, 2012. To appear.

[12] P. Faliszewski, E. Hemaspaandra, and L. Hemaspaandra. Multimode control attacks on elections. *Journal of Artificial Intelligence Research*, 40:305–351, 2011.

[13] J. Flum and M. Grohe. *Parameterized Complexity Theory*. Springer-Verlag, 2006.

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

[15] E. Hemaspaandra, L. Hemaspaandra, and J. Rothe. Exact analysis of Dodgson elections: Lewis Carroll's 1876 voting system is complete for parallel access to NP. *Journal of the ACM*, 44(6):806–825, 1997.

[16] E. Hemaspaandra, H. Spakowski, and J. Vogel. The complexity of Kemeny elections. *Theoretical Computer Science*, 349(3):382–391, 2005.

[17] C. Kenyon-Mathieu and W. Schudy. How to rank with few errors. In *Proceedings of STOC-2007*, pages 95–103, June 2007.

[18] R. Lipton, E. Markakis, E. Mossel, and A. Saberi. On approximately fair allocations of indivisible goods. In *Proceedings of ACM EC-2004*, pages 125–131, May 2004.

[19] T. Lu and C. Boutilier. Budgeted social choice: From consensus to personalized decision making. In *Proceedings of IJCAI-2011*, pages 280–286, 2011. See also a version that appeared in COMSOC-2010.

[20] B. Monroe. Fully proportional representation. *American Political Science Review*, 89(4):925–940, 1995.

[21] R. Niedermeier. *Invitation to Fixed-Parameter Algorithms*. Oxford University Press, 2006.

[22] N. Nisan, T. Roughgarden, É. Tardos, and V. Vazirani, editors. *Algorithmic Game Theory*. Cambridge University Press, 2007.

[23] J. Oren. Personal communication, 2012.

[24] R. Potthoff and S. Brams. Proportional representation: Broadening the options. *Journal of Theoretical Politics*, 10(2):147–178, 1998.

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

[26] R. Rosenthal. A class of games possessing pure-strategy Nash equilibria. *International Journal of Game Theory*, 2(1):65–67, 1973.

[27] J. Rothe, H. Spakowski, and J. Vogel. Exact complexity of the winner problem for Young elections. *ACM Transactions on Computer Systems*, 36(4):375–386, 2003.

[28] D. Shiryaev, L. Yu, and E. Elkind. On elections with robust winners. In *Workshop Notes of COMSOC-2012*, 2012. To appear.

[29] Y. Shoham and K. Leyton-Brown. *Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations*. Cambridge University Press, 2009.