A Lower Bound for Local Search Proportional Approval Voting

Sonja Kraiczy
Department of Computer Science, University of Oxford, UK

Edith Elkind
Department of Computer Science, University of Oxford, UK
Alan Turing Institute, London, UK

Abstract

Selecting k out of m items based on the preferences of n heterogeneous agents is a widely studied problem in algorithmic game theory. If agents have approval preferences over individual items and harmonic utility functions over bundles – an agent receives $\sum_{j=1}^{t} \frac{1}{j}$ utility if t of her approved items are selected – then welfare optimisation is captured by a voting rule known as Proportional Approval Voting (PAV). PAV also satisfies demanding fairness axioms. However, finding a winning set of items under PAV is NP-hard. In search of a tractable method with strong fairness guarantees, a bounded local search version of PAV was proposed [2]. It proceeds by starting with an arbitrary size-k set W and, at each step, checking if there is a pair of candidates $a \in W$, $b \notin W$ such that swapping a and b increases the total welfare by at least ε; if yes, it performs the swap. Aziz et al. show that setting $\varepsilon = \frac{n}{k^2}$ ensures both the desired fairness guarantees and polynomial running time. However, they leave it open whether the algorithm converges in polynomial time if ε is very small (in particular, if we do not stop until there are no welfare-improving swaps). We resolve this open question, by showing that if ε can be arbitrarily small, the running time of this algorithm may be super-polynomial. Specifically, we prove a lower bound of $\Omega(k^{\log k})$ if improvements are chosen lexicographically. To complement our lower bound, we provide an empirical comparison of two variants of local search – better-response and best-response – on several real-life data sets and a variety of synthetic data sets. Our experiments indicate that, empirically, better response exhibits faster running time than best response.

2012 ACM Subject Classification Theory of computation → Exact and approximate computation of equilibria

Keywords and phrases Computational Social Choice, Committee Elections, Local Search, Fairness

Digital Object Identifier 10.4230/LIPIcs.ESA.2024.82

Related Version *Full Version*: `https://arxiv.org/abs/2408.02300`

Funding *Sonja Kraiczy*: The author is grateful to be supported by an EPSRC studentship.
Edith Elkind: This work is supported by AI Programme of The Alan Turing Institute and EPSRC grant EP/X038548/1.

1 Introduction

We study the collective decision problem where the goal is to select k out of m items (candidates) based on the preferences of n agents (voters). This problem (or, more precisely, its generalisation to the setting where different items may have different costs and there is a budget constraint) is known as the *combinatorial public project* problem in the algorithmic mechanism design literature, where the focus in on optimisation of the utilitarian welfare subject to strategyproofness constraints [7, 8]. In the computational social choice literature, it is known as the the *multiwinner voting* problem, and an important concern is proportionality, i.e., ensuring that large groups of voters with similar preferences are fairly represented by the elected candidates [10].

32nd Annual European Symposium on Algorithms (ESA 2024).
Editors: Timothy Chan, Johannes Fischer, John Iacono, and Grzegorz Herman; Article No. 82; pp. 82:1–82:14
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Proportional Approval Voting (PAV) [9], which belongs to the class of Thiele's methods [1], is a multiwinner voting rule that can be viewed both from the welfare maximisation perspective and from the proportionality perspective. Given an election with a set of voters N, a set of candidates C, a target committee size k, and the voters' approval ballots $(A_v)_{v\in N}$, where $A_v \subseteq C$ for all $v \in N$, it computes the *PAV score* of a committee $W \subseteq C$ as $\textsc{pavsc}(W) = \sum_{v\in N}\sum_{j=1}^{|A_v\cap W|}\frac{1}{j}$, and outputs all size-$k$ committees with the maximum PAV score. This rule admits a utilitarian interpretation: agent v with ballot A_v is assumed to compute the utility of bundle W as $\sum_{j=1}^{|A_v\cap W|}\frac{1}{j}$, and the rule maximises the utilitarian welfare. Another interpretation makes no assumptions about agents' values for bundles, but aims to achieve proportionality. Intuitively, proportionality means that an α-proportion of the population that jointly approves at least αk items should be represented by an α-fraction of the k selected items. There are several ways to formalise this intuition, including justified representation axioms, such as PJR and EJR [1, 14], or the notion of proportionality degree [15], and PAV is among the very few voting rules that satisfy EJR and have optimal proportionality degree. Unfortunately, however, PAV is known to be NP-hard to compute [16, 3].

Algorithm 1 ε-local search PAV (ε-ls-PAV).

Data: $\varepsilon > 0$, arbitrary committee W_{init} of size k, voters' approval ballots $(A_v)_{v\in N}$
Result: W of size k
$W \leftarrow W_{\text{init}}$;
while $\exists b \notin W, a \in W$ *such that* $\Delta(W, a, b) \geq \varepsilon$ **do**
  $W \leftarrow (W \cup \{b\}) \setminus \{a\}$
end
return W

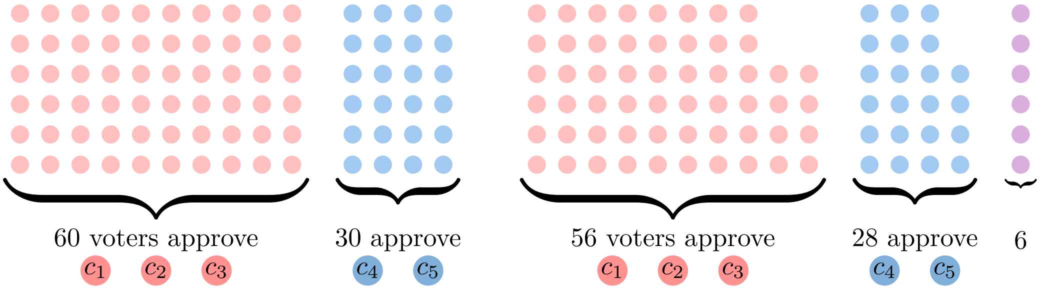
Since PAV has excellent proportionality properties, there has been a lot of interest in identifying tractable variants of this rule. Two natural approaches to explore in this context are greedy sequential optimisation and local search. The former is a special case of the greedy algorithm for submodular function maximisation, and approximates the social welfare by a factor of $(1 - \frac{1}{e})$ [11]. Unfortunately, high PAV score does not imply good proportionality guarantees [1], so approximation algorithms do not appear to be very useful from the fairness perspective.

In contrast, the local search approach turns out to be well-suited for identifying fair outcomes. The reason for this is that proofs of proportionality guarantees for PAV use local swap arguments: they show that if a committee W is proportional, there is no pair of candidates $a \in W, b \notin W$ such that swapping them increases the PAV score, i.e., the quantity

$$\Delta(W, a, b) = \textsc{pavsc}((W \setminus \{a\}) \cup \{b\}) - \textsc{pavsc}(W)$$

is positive. The local search algorithm that starts with an arbitrary committee and performs PAV score-improving swaps is therefore a natural heuristic for PAV. It was first introduced and studied by Aziz et al. [2]. However, Aziz et al. were unable to show that local search converges after polynomially many iterations, as a single iteration may only increase the PAV score by a tiny amount. To overcome this issue, they considered a parameterised version of local search, which only performs swaps if they improve the PAV score by at least ε (see Algorithm 1 for the pseudocode). They observed that if ε is sufficiently small, the algorithm, which we will call ε-ls-PAV, preserves the proportionality guarantees of PAV, and if ε is sufficiently large, it converges in polynomial time; setting $\varepsilon = \frac{n}{k^2}$ achieves both of these objectives. However, Aziz et al. left it as an open problem if ε-ls-PAV converges in polynomial time for *all* values of $\varepsilon > 0$. The following conjecture is implicit in their work:

▶ **Conjecture 1** (left open by [2]). *For small ε, ε-ls-PAV may make a super-polynomial number of swaps.*



(a) 60 voters approve c_1, c_2, c_3 and 30 voters approve c_4, c_5

(b) 6 voters leave their groups and vote for themselves. 56 voters approve c_1, c_2, c_3 and 28 voters approve c_4, c_5

Figure 1 For $k = 3$ and initial committee $W = \{c_1, c_2, c_3\}$, $\frac{n}{k^2}$-ls-PAV would replace a member of W with one of c_4, c_5 in the instance in Figure 1a, but not in the instance in Figure 1b.

Now, while setting $\varepsilon = \frac{n}{k^2}$ preserves the worst-case proportionality guarantees, the faster running time comes at a cost: for large values of n and small values of k the algorithm may get "lazy" and choose not to perform a swap even if it would result in a much fairer outcome. Indeed, consider an instance with 90 voters, 60 of which vote for $\{c_1, c_2, c_3\}$ and 30 vote for $\{c_4, c_5\}$ (see Figure 1a). For a committee size of $k = 3$, a fair outcome gives two representatives to the group of 60 and one to the group of 30. Indeed, one can check that the majoritarian committee $W = \{c_1, c_2, c_3\}$ is not locally optimal, in the sense that $\frac{n}{k^2}$-ls-PAV will replace one of the candidates in W with c_4 or c_5. However, if the voters' preferences are a bit more diffuse, this is no longer the case. To see this, consider Figure 1b, where the groups of size 60 and 30 lose four and two members, respectively, and the six breakaway voters become candidates and vote for themselves. For this instance, $W = \{c_1, c_2, c_3\}$ is an output of 10-ls-PAV, because the PAV scores of W and $(W \setminus \{c_3\}) \cup \{c_4\}$ are, respectively, $56 \cdot (1 + 1/2 + 1/3)$ and $56 \cdot (1 + 1/2) + 28$, and $28 - \frac{56}{3} < \frac{n}{k^2} = 10$. On the other hand, for the "vanilla" version of ls-PAV, which does not stop until there are no PAV-score improving swaps, the "unfair" committee W is not among the outputs of local search on the modified instance.

There are other reasons to favour "vanilla" ls-PAV over $\frac{n}{k^2}$-ls-PAV. For instance, the former rule is easier to explain to voters, who may be disappointed that the latter rule stops despite the availability of score-improving swaps. Also, "vanilla" ls-PAV is more decisive – it is easy to see that for $\varepsilon < \varepsilon'$ each output of ε-ls-PAV is an output of ε'-ls-PAV, but the converse is not necessarily true – and decisiveness is viewed as a desirable property in the social choice literature. Thus, it is important to understand whether the conjecture of [2] is true, i.e., whether ε-ls-PAV converges in polynomial time even for small values of ε.

Motivated by these considerations, in this work we investigate Conjecture 1 and resolve it in the positive. Specifically, for adversarially chosen swaps we show a lower bound of $\Omega(k^{\log k})$ via a subtle combinatorial construction. We then extend our result to a natural pivoting rule, which chooses swaps lexicographically. While our result does not rule out the possibility that the outputs of ε-ls-PAV can be computed by a different polynomial-time algorithm (the complexity of this problem remains an interesting open question), it justifies the choice of $\varepsilon = \frac{n}{k^2}$ in the work of Aziz et al. [2].

Importantly, while our lower bound holds for better response with the lexicographic pivoting rule, it does not extend to another natural pivoting rule: choosing the swap that offers the maximum increase in the PAV score, i.e., best-response dynamics. Indeed, it remains an open problem to determine if the best-response variant of ls-PAV may take superpolynomially many steps. Motivated by this question, in the extended version we provide an empirical comparison between lexicographic better response and best response, using several real-life and synthetic datasets. We measure the performance of each algorithm on a given instance as the number of candidate swaps it needs to consider before termination (this is a useful proxy for running time as long as we do not have access to parallel processing hardware). Interestingly, on the datasets we investigate, better response considers fewer swaps than best response. Hence, while our theoretical worst-case results seem to favour best response over better response, the empirical results paint the opposite picture.

1.1 Roadmap of our Approach

We study the runtime of ε-ls PAV with the threshold ε set to a very small positive value (which we will denote by 0^+). Specifically, for each $k \geq 0$ we construct a multiwinner election E^k with target committee size k and the following properties: (1) the number of voters in E^k is polynomial in k; (2) on E^k, 0^+-ls-PAV may require $\Omega(k^{\log k})$ steps until convergence. Our argument is closely connected to the following simple-to-formulate number-theoretic question:

$$\text{minimise} \sum_{i=1}^{k} \frac{w_i}{i} \qquad \text{subject to} \tag{1}$$

$$\sum_{i=1}^{k} \frac{w_i}{i} > 0, \qquad \text{where } w_i \in \mathbb{Z} \text{ and } w_i = \text{poly}(k) \tag{2}$$

If w_i are not required to be polynomial in k, it is not difficult to see that the minimum of the sum in (1) is the inverse of the least common multiple of $\{1, 2, \ldots, k\}$. However, in our construction the number $\sum_{i=1}^{k} |w_i|$ corresponds to the size of the voter set. Hence, if we want the size of E^k to be polynomial in k, the w_i need to be polynomial in k as well. At the heart of our construction is an instance of size polynomial in k that corresponds to a value of $\frac{1}{k^{\log k}}$ for the above objective. We use it to build an election with a carefully crafted combinatorial structure on which 0^+-ls-PAV is forced to repeatedly undo previously achieved progress. We note that if $w_i \in \{-1, 1\}$, then the best known upper bound is due to [4] and matches our construction in Lemma 4.3.

As a warm-up, in Section 3, we first prove a simpler result, which illustrates one of the main ideas behind the more complex construction. We study $\frac{n}{k^2}$-ls-PAV where swaps may be chosen adversarially, and show that its worst-case runtime is $\widetilde{\Theta}(k^2)$. In Section 4.2 we first construct an election together with a committee where a swap increases the PAV score just by $\Theta(\frac{1}{k^{\log k}})$. We use this as a building block to construct two further levels of complexity, which give us our desired instance (Sections 4.3 and 4.4). This instance is used to show our main result for the adversarial setting (Section 4.5): In Theorem 3, we exhibit a sequence of swaps of length super-polynomial in k that may be performed by 0^+-ls-PAV. Finally, in Section 4.6 we show that lexicographic better response executes a subsequence of our sequence from Theorem 3, and the length of this subsequence is still superpolynomial in k.

1.2 Related Work

Initially, proportionality in multiwinner committee voting was considered from an axiomatic perspective: there is a spectrum of justified representation axioms, ranging from Justified Representation (JR) (which is rather mild and easy to satisfy) to more demanding axioms such as Proportional, Extended and Full Justified Representation (PJR, EJR and FJR) [1, 14, 12]; as well as EJR+ [6]; $\frac{n}{k^2}$-ls-PAV is among the very few polynomial-time computable voting rules that satisfy EJR. Subsequently, Skowron [15] pursued a qualitative approach, and put forward the notion of the proportionality degree, which formalises to what degree β it holds that an α proportion of the population proposing at least αk candidates will be represented (so, ideally $\beta \sim \alpha$ and the group is represented by roughly βk candidates on average). PAV exhibits excellent performance according to this measure: for PAV the ratio $\frac{\beta}{\alpha}$ approaches 1. For $k \leq 200$ Skowron shows that Sequential PAV has a proportionality degree of at least 0.7, but for larger k the proportionality of Sequential PAV remains open. Other voting rules with good proportionality guarantees are Sequential Phragmén's rule (which satisfies PJR) [9, 5] and the Method of Equal Shares (which satisfies the stronger EJR axiom) [13]. Unlike PAV, both of these rules are formulated in terms of voters sharing the "load" incurred by the candidates in the committee, and have a proportionality degree of 0.5.

2 Preliminaries

For $n \in \mathbb{N}$, we write $[n] = \{1, \dots, n\}$. An *approval election* is a 4-tuple $E = (N, C, (A_v)_{v\in N}, k)$, where $N = [n]$ is a set of *voters*, C is a set of *candidates*, $|C| = m$, $A_v \subseteq C$ is the *ballot* of voter $v \in N$, and $k \in [m]$ is the *target committee size*. Subsets of C (not necessarily of size k) are called *committees*.

We define the *PAV satisfaction* of a set of voters V from a committee W as $\textsc{pavsc}_V(W) = \sum_{v\in V} \sum_{j=1}^{|A_v\cap W|} \frac{1}{j}$; if V is a singleton, i.e., $V = \{v\}$, we omit the braces and write $\textsc{pavsc}_v$ instead of $\textsc{pavsc}_{\{v\}}$. Given a committee W, a pair of candidates $b \notin W$, $a \in W$, and a set of voters V, we denote by $\Delta_V(W, a, b)$ the change in the PAV-satisfaction of V that is caused by swapping a with b:

$$\Delta_V(W, a, b) = \textsc{pavsc}_V(W \cup \{b\} \setminus \{a\}) - \textsc{pavsc}_V(W).$$

For readability, we omit V from the notation when $V = N$, i.e., we write $\Delta(W, a, b) := \Delta_N(W, a, b)$.

The "vanilla" local search algorithm, which swaps $a \in W$ with $b \notin W$ as long as $\Delta(W, a, b) > 0$, can be described as ε-ls-PAV for $\varepsilon \leq \min\{\Delta(W, a, b) : W \subseteq C, |W| = k, b \notin W, a \in W, \Delta(W, a, b) > 0\}$. It can be shown that this condition can be satisfied by setting $\varepsilon = \frac{1}{\mathrm{lcm}([k])}$, where for each $S \subset \mathbb{N}$ we denote by $\mathrm{lcm}(S)$ the least common multiple of the integers in S. In what follows, we denote this value of ε by 0^+.

Given a committee W, we say that a sequence of swaps

$$\mathbf{X} = (a_1, b_1), (a_2, b_2), \dots, (a_s, b_s)$$

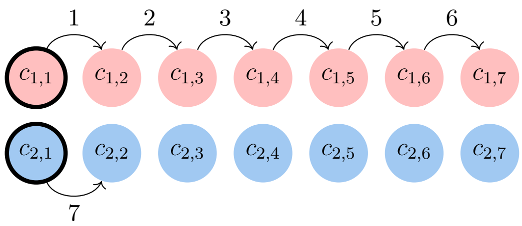
is *valid* if for each $t \in [s]$ the committee $W_t = (W_{t-1} \cup \{b_t\}) \setminus \{a_t\}$ (where $W_0 := W$) satisfies $a_t \in W_{t-1}$, $b_t \notin W_{t-1}$. The *length* of a sequence $\mathbf{X}$, denoted by $|\mathbf{X}|$, is the number of pairs in $\mathbf{X}$. We define the *inverse* (sequence) of $\mathbf{X}$ as $\mathbf{X}^{-1} = (b_s, a_s), (b_{s-1}, a_{s-1}), \dots, (b_1, a_1)$. Given two finite sequences of swaps $\mathbf{X}$ and $\mathbf{Y}$, we define their *concatenation* $\mathbf{X} \oplus \mathbf{Y}$ as the sequence with prefix $\mathbf{X}$ followed by suffix $\mathbf{Y}$. For our proofs, it will be useful to have an arbitrarily large pool of "dummy" candidates. We therefore define $D_k = \{d_1, \dots, d_k\}$ so that $D_{k+1} = D_k \cup \{d_{k+1}\}$, $D_0 = \varnothing$ and $D = \cup_{k=1}^{\infty}\{D_k\}$.

We omit some proofs due to space constraints; the respective claims are marked with ★. All missing proofs and the simulation results appear in the extended version of the paper.

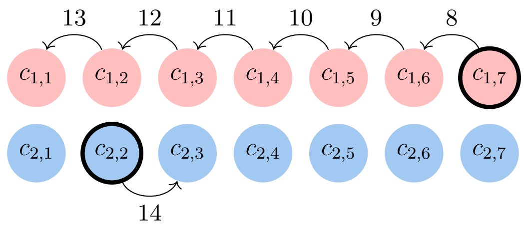
3 Warm-up: Lower Bound for $\frac{n}{k^2}$-ls-PAV

To showcase the ideas behind our main lower bound (Theorem 9), we start by presenting a lower bound on $\frac{n}{k^2}$-ls-PAV in the adversarial setting. Specifically, for each n-voter election and committee size k, we consider a directed graph whose vertices are committees, and whose edges are pairs (W, W') such that W' can be obtained from W via a swap and the PAV score of W' is at least $\frac{n}{k^2}$ higher than that of W. We exhibit an election for which this graph contains a path of length $\Omega(k^2)$. We call this setting adversarial, because these swaps may be suggested by an adversary whose aim is to maximise the number of iterations.

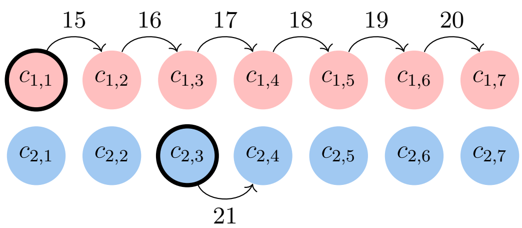
This result establishes that the upper bound on the number of iterations of the algorithm of [2] is tight up to a $\log k$ factor. Indeed, since the maximum PAV score of a size-k committee is $O(n \log k)$, it follows that $\frac{n}{k^2}$-ls-PAV converges in at most $O(k^2 \log k)$ iterations.



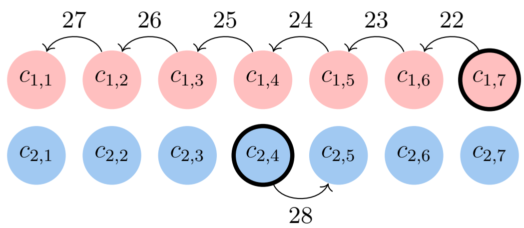
(a) Illustration of swaps 1–7.



(b) Illustration of swaps 8–14.



(c) Illustration of swaps 15–21.



(d) Illustration of swaps 22–28.

Figure 2 Highlighted candidates are in the committee; arrows from a to b labelled with t indicate that a is replaced by b in iteration t. Observe that pink candidates are only replaced by pink candidates; similarly, blue candidates are only replaced by blue candidates. This illustrates a swap sequence similar to that in the proof of Theorem 2, with the exception that, to make the figure visually appealing, we display an equal number of blue and pink candidates.

▶ **Theorem 2.** *For every $k \geq 1$ there exists a committee election with* poly(k) *voters, a committee W_0, $|W_0| = k$, and a sequence of $\Omega(k^2)$ swaps starting from W_0 such that each swap in this sequence strictly increases the PAV score by at least $\frac{n}{k^2}$.*

Proof. Let $k \geq 4$ and $t = \lfloor \frac{k}{4} \rfloor$. We define the election $E = (N, C, (A_v)_{v \in N}, k)$ as follows.

$$C = C_1 \cup C_2 \cup D_{k-2}, \quad C_1 = \{c_{1,1}, \ldots, c_{1,t+1}\}, \quad C_2 = \{c_{2,1}, \ldots, c_{2,k}\};$$

$$N = V_1 \cup V_2 \cup \bigcup_{j \in [k]} S_j \cup U, \text{ where } V_\ell = \{v_{\ell,1}, \ldots, v_{\ell,t}\}, \ell \in [2], \quad |U| = \left\lfloor \frac{k^2}{4} - \frac{k}{2} \right\rfloor,$$

$$|S_j| = t, j \in [k].$$

The approval sets of voters $v_{1,i}$ and $v_{2,i}$, $i \in [t]$, are given by

$A_{v_{1,i}} = \{c_{1,i+1}, \dots, c_{1,t+1}\} \cup \{c_{2,j} \in C_2 \mid j \text{ is even}\}$ and
$A_{v_{2,i}} = \{c_{1,1}, \dots, c_{1,i}\} \cup \{c_{2,j} \in C_2 \mid j \text{ is odd}\}$,

For each $j \in [k]$ the approval set of each voter $s \in S_j$ is $A_s = \{c_{2,j}, \dots, c_{2,k}\}$. Each $u \in U$ has approval set $A_u = D_{k-2}$. Intuitively, voters in U are dummy voters and candidates in D_{k-2} are dummy candidates. The sequence of swaps we will exhibit affects voters in $N \setminus U$ and candidates in $C_1 \cup C_2$, but no other voters or candidates.

Set-up. Consider $\frac{n}{k^2}$-ls-PAV on the above instance with an initial committee

$$W_0 = D_{k-2} \cup \{c_{1,1}, c_{2,1}\}.$$

We first show that $\frac{n}{k^2} \leq \frac{1}{2}$. Indeed, we have

$$n = |V_1| + |V_2| + \sum_{j=1}^{k} |S_j| + |U| = \left\lfloor \frac{k}{4} \right\rfloor + \left\lfloor \frac{k}{4} \right\rfloor + k \times \left\lfloor \frac{k}{4} \right\rfloor + \left\lfloor \frac{k^2}{4} - \frac{k}{2} \right\rfloor \leq \frac{k^2}{2}.$$

In what follows, we will say that a swap (a, b) is a *good swap for W* if $\Delta(W, a, b) \geq \frac{n}{k^2}$; we will say that (a, b) is a *good swap* if W is clear from the context. As we have argued that $\frac{n}{k^2} \leq \frac{1}{2}$, every valid swap that increases the PAV score by at least $\frac{1}{2}$ is a good swap.

Sequence of Swaps. Define $\mathbf{Y} = \oplus_{i=1}^{t}(c_{1,i}, c_{1,i+1})$ and let $\mathbf{Z}_j = \mathbf{Y}$ if j is odd and $\mathbf{Z}_j = \mathbf{Y}^{-1}$ if j is even. Let $\mathbf{X} = \oplus_{j=1}^{k-1} (\mathbf{Z}_j \oplus (c_{2,j}, c_{2,j+1}))$, and note that $|\mathbf{Y}| = |\mathbf{Y}^{-1}| = t$, so $|\mathbf{X}| = (k-1) \cdot (t+1) = \Omega(k^2)$. We will argue that all swaps in $\mathbf{X}$ are good.

To this end, we split up the analysis into the following four claims.

(i) For committee $W = D_{k-2} \cup \{c_{1,i}, c_{2,j}\}$, if j is odd and $i \leq t$, $(c_{1,i}, c_{1,i+1})$ is a good swap.
(ii) For committee $W = D_{k-2} \cup \{c_{1,i}, c_{2,j}\}$, if j is even and $i > 1$, $(c_{1,i}, c_{1,i-1})$ is a good swap.
(iii) For committee $W = D_{k-2} \cup \{c_{1,t+1}, c_{2,j}\}$, if $j < k$ is odd, $(c_{2,j}, c_{2,j+1})$ is a good swap.
(iv) For committee $W = D_{k-2} \cup \{c_{1,1}, c_{2,j}\}$, if $j < k$ is even, $(c_{2,j}, c_{2,j+1})$ is a good swap.

Together, these four claims imply that $\mathbf{X}$ is a sequence of good swaps. Indeed, suppose that j is odd, and consider the committee $W = D_{k-2} \cup \{c_{1,1}, c_{2,j}\}$. By Claim (i), $\mathbf{Z}_j = \mathbf{Y}$ is a sequence of good swaps. After executing $\mathbf{Y}$ we obtain a committee $W = D_{k-2} \cup \{c_{1,t+1}, c_{2,j}\}$ satisfying the condition in Claim (iii). Hence, if $j < k$, then $(c_{2,j}, c_{2,j+1})$ is a good swap. This swap results in a committee $W = D_{k-2} \cup \{c_{1,t+1}, c_{2,j+1}\}$ satisfying the condition in Claim (ii). This implies that $\mathbf{Z}_{j+1} = \mathbf{Y}^{-1}$ is a sequence of good swaps, resulting in a committee $W = D_{k-2} \cup \{c_{1,1}, c_{2,j+1}\}$. This committee, in turn, is as described in (iv), so if $j + 1 < k$, then $(c_{2,j+1}, c_{2,j+2})$ a good swap. This results in $W = D_{k-2} \cup \{c_{1,1}, c_{2,j+2}\}$, which again satisfies the condition in (i). As this reasoning applies to all odd values of j, including $j = 1$ (which corresponds to our starting point W_0), we can conclude that the sequence $\mathbf{X}$ is a sequence of good swaps. It remains to prove Claims (i)–(iv).

Claim (i). Suppose $W = D_{k-2} \cup \{c_{1,i}, c_{2,j}\}$ where $j \in [k]$ is odd and $i \in [t]$. Then $|A_{v_{2,i}} \cap W| = 2$ and $|A_{v_{1,i}} \cap W| = 0$. Moreover, $v_{2,i}$ approves $c_{1,i}$ and not $c_{1,i+1}$, and conversely for $v_{1,i}$, while every other voter approves either both or neither of $c_{1,i}$ and $c_{1,i+1}$. We conclude that $(c_{1,i}, c_{1,i+1})$ is a good swap, because $\Delta(W, c_{1,i}, c_{1,i+1}) = +1 - \frac{1}{2} = \frac{1}{2}$.

Claim (ii). Suppose $W = D_{k-2} \cup \{c_{1,i}, c_{2,j}\}$, where $1 < i \leq t + 1$ and $j \in [k]$ is even. Then $|A_{v_{1,i-1}} \cap W| = 2$, $|A_{v_{2,i-1}} \cap W| = 0$, and every other voter approves either both of $c_{1,i-1}$ and $c_{1,i}$ or neither of them. Thus, $\Delta(W, c_{1,i}, c_{1,i-1}) = +1 - \frac{1}{2} = \frac{1}{2}$.

Claim (iii). Suppose $W = D_{k-2} \cup \{c_{1,t+1}, c_{2,j}\}$, where $j < k$ is odd. Each voter in S_{j+1} approves $c_{2,j+1}$ and not $c_{2,j}$. Every voter in V_2 approves $c_{2,j}$, but not $c_{2,j+1}$, while every voter in V_1 approves $c_{2,j+1}$ and not $c_{2,j}$. By construction, the remaining voters (i.e., the voters in S_ℓ, $\ell \neq j+1$, and the voters in U) approve either both of $c_{2,j}$ and $c_{2,j+1}$ or neither. For $s \in S_{j+1}$, their satisfaction is $|A_s \cap W| = 0$. For every $i \in [t]$ we have $|A_{v_{1,i}} \cap W| = 1$ as $v_{1,i}$ only approves $c_{1,t+1}$ in W, and $|A_{v_{2,i}} \cap W| = 1$, because $v_{2,i}$ only approves $c_{2,j}$ as j is odd. Thus, $\Delta(W, c_{2,j}, c_{2,j+1}) = +|S_{j+1}| + \frac{1}{2} \cdot |V_1| - |V_2| = t + \frac{t}{2} - t = \frac{t}{2} \geq \frac{1}{2}$, provided $k \geq 4$. This shows that $(c_{2,j}, c_{2,j+1})$ is a good swap.

Claim (iv). Suppose $W = D_{k-2} \cup \{c_{1,1}, c_{2,j}\}$, where $j < k$ is even. Similarly to the proof of Claim (iii), we can show that $\Delta(W, c_{2,j}, c_{2,j+1}) \geq \frac{1}{2}$. This concludes the proof. ◀

4 Main Result

We are now ready to present our main result.

▶ **Theorem 3 (★).** *For every $k \geq 1$ there exists a committee election with* poly(k) *voters, a committee W_0, $|W_0| = k$, and a sequence of $\Omega(k^{\log k})$ swaps starting from W_0 such that each swap in this sequence strictly increases the PAV score.*

The proof of our main lower bound is in many ways similar to the proof of Theorem 2. We start by giving a high-level overview of the proof, and then describe our construction; the proof of correctness is mostly relegated to the extended version of the paper. We first consider the adversarial setting; in Section 4.6 we will show how to extend our proof to lexicographic better response.

4.1 Proof Overview

We construct a committee W of size k and a sequence of swaps of length $\Omega(k^{\log k})$ such that 0^+-ls-PAV may execute this sequence when initialised on W. Just as in the proof of Theorem 2, this sequence of swaps leaves most members of the initial committee untouched. Most of the action takes place in the first $k_1 = O(\log k)$ committee spots; the candidates in the remaining spots stay in place throughout the entire sequence of swaps. For each $i = 1, \ldots, k_1$, the committee spot i is assigned its own set of candidates C_i, so that in the sequence we construct, a candidate in C_i can only be replaced by another candidate from C_i.

Furthermore, each of the first k_1 committee positions has its corresponding set of voters, and between different committee positions, initially the voters' satisfaction is very unequal: voters corresponding to the first position are most happy, and voters corresponding to later committee positions are increasingly unhappy. The sequence of swaps will again start by making the most happy voters happier and then move on to less happy voters, making them better off. *Thereby it will undo all the work it has done so far and will have to repeat it.*

Consider Figure 3, where each board represents a committee. The figure should be read left to right and top to bottom, where the next board/committee results from the previous one if certain swaps are made. More precisely, within a board each square corresponds to a candidate, so that C_i is the set of candidates in the i-th column. The coloured squares mark the candidates that are in the committee, and arrows between squares indicate swaps between candidates. The result of the swap(s) can be seen in the next board.

The pattern in Figure 3 may not be immediately evident; the reader may want to revisit it after having read the proof of Theorem 3. For now, we use Figure 3 to illustrate how we build up our instance in several steps, creating increasingly larger building blocks. Observe

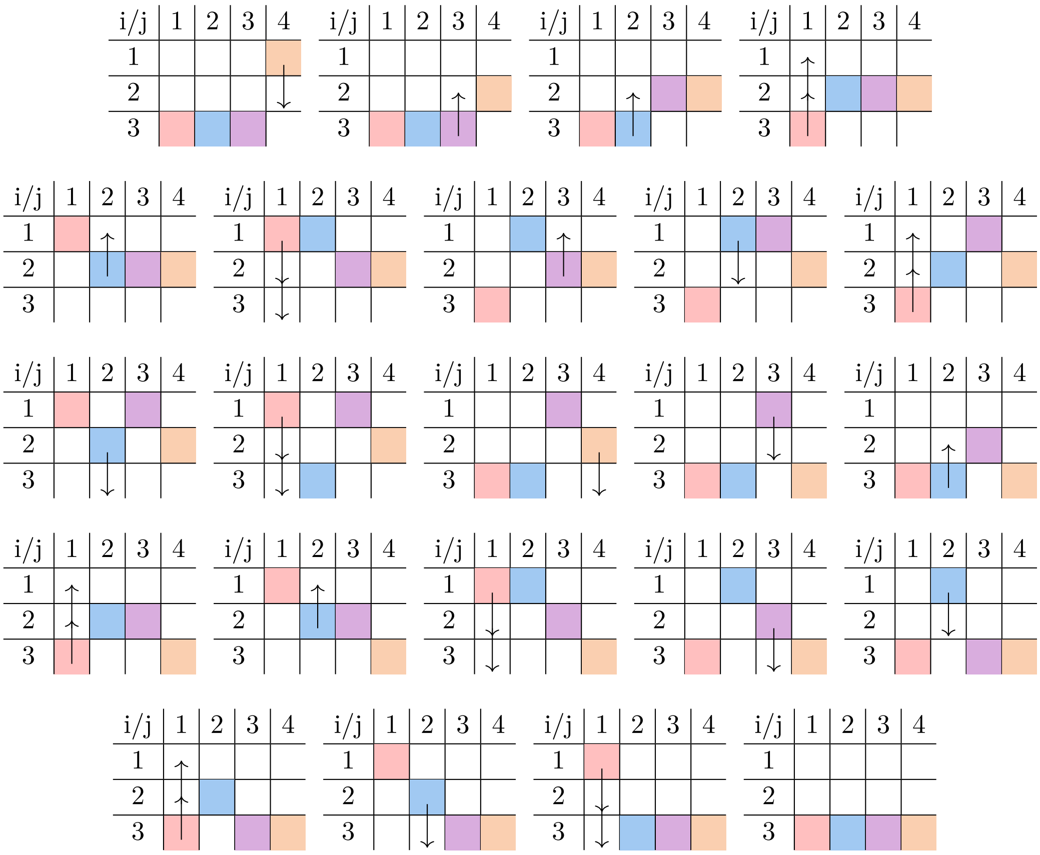


Figure 3 We illustrate the sequence of swaps in Theorem 3 by a small example. Each one of the 23 3×4 grids shows 12 candidates, one for each empty square, the 4 coloured ones indicating candidates currently in the committee. The empty squares in column i are the candidates in C_i; they are ordered as $c_{i,1}, c_{i,2}, c_{i,3}$ from top to bottom. We omit the dummy candidates from this picture, and let $k_1 = 4$ and $t + 1 = 3$, as larger t is only necessary for the sequence length in the proof. An arrow indicates the swap that will transform the current committee into the next committee. The top left initial committee is $\{c_{1,3}, c_{2,3}, c_{3,3}, c_{4,1}\}$ and the bottom right final committee is $\{c_{1,3}, c_{2,3}, c_{3,3}, c_{4,3}\}$.

that in Figure 3 swaps only occur along a column (corresponding to candidates C_i for some i): indeed, as we mentioned earlier, in our construction swaps can only replace candidates in C_i with other candidates in C_i.

1. Zooming in on a single swap, the voters responsible for this swap form an atomic building block of our construction. This building block, Election $E(j, k)$, is given in Section 4.2. Here, j is carefully picked to depend on the column i (i.e., the i-th spot on the committee). In Lemma 5 we show that the corresponding swap increases the PAV score by exactly

$$\delta(j, k) = \frac{j!}{\prod_{j'=0}^{j}(k - j')}.$$

2. Zooming out to just the i-th column, the election responsible for the dynamics along the i-th column is $E^t(j, k)$, where $t = |C_i| - 1$. We discuss how to construct $E^t(j, k)$ from $E(j, k)$ in Section 4.3.
3. Finally, the entire board roughly corresponds to election E, constructed out of the building blocks $E^t(j, k)$ (Section 4.4).

With the constructed election E in hand, in Section 4.5 we exhibit an initial committee and a sequence of good swaps of length superpolynomial in k, such that, starting from this committee, 0^+-ls-PAV under adversarial better response executes this sequence of swaps. Finally, in Section 4.6 we show how to modify this instance to establish that, even when the improving swaps are selected according to a fixed pivoting rule (rather than adversarially), 0^+-ls-PAV may still make super-polynomially many swaps. To this end, we prove that a long subsequence of the swap sequence from Theorem 3 is preserved by the pivoting rule.

4.2 First Steps: Election $E(j,k)$

We now introduce a family of elections that form the smallest building blocks of our instance. We will frequently say that an election E has structure X if it is isomorphic to election X. As in the proof of Theorem 2, we write $D_\ell = \{d_1, \dots, d_\ell\}$ to denote a set of ℓ dummy candidates. For committee size $k \in \mathbb{N}$, we will use induction on j to construct elections $F(j,k) = (N, C, (A_v)_{v\in N}, k)$ with $|N| = 2^j$ and $C = D_{k-1} \cup \{a, b\}$ for each $1 \le j < k$.

Construction 1. [Election $F(j,k)$] Fix $j, k \in \mathbb{N}$ with $1 \le j < k$. For $j = 1$, let $F(j,k) = (\{1,2\}, C, (A_v)_{v=1,2}, k)$, where $C = D_{k-1} \cup \{a, b\}$, $A_1 = D_{k-1} \cup \{a\}$ and $A_2 = D_{k-2} \cup \{b\}$.

For $j > 1$ (and $k \ge j+1$), we construct $F(j,k)$ as follows. Consider elections $F(j-1,k) = (N_1, C_1, (A_v)_{v\in N_1}, k)$ and $F(j-1,k-1) = (N_2, C_2, (A_v)_{v\in N_2}, k-1)$, where $N_1 \cap N_2 = \varnothing$ (we relabel the voters to ensure they are distinct), and $C_1 = D_{k-1} \cup \{a_1, b_1\}$, $C_2 = D_{k-2} \cup \{a_2, b_2\}$. We set $C = D_{k-1} \cup \{a, b\}$, $N = N_1 \sqcup N_2$, and modify the voters' ballots as follows: for each $v \in N_1$ we replace each occurrence of a_1 and b_1 in A_v with b and a, respectively, and for each $v \in N_2$ we replace each occurrence of a_2 and b_2 in A_v with a and b, respectively. We then define $F(j,k) = (N, C, (A_v)_{v\in N}, k)$.

Observe that for each $k \ge 1$ the number of voters in $F(j,k)$ is exactly 2^j; this follows easily by induction since there are two voters in election $F(1,k)$, and for $j > 1$, elections $F(j-1,k)$ and $F(j-1,k-1)$ have disjoint sets of voters of size 2^{j-1} each.

Construction 2. [Election $E(j,k)$] Election $E(j,k)$ is built similarly to $F(j,k)$. We start with two copies of $F(j-1,k-1)$, merge them as in the construction for $F(j,k)$, introduce two new candidates x and y, and make all voters from the first copy approve x and all voters from the second copy approve y.

Formally, consider two disjoint copies of $F(j-1,k-1)$ given by $(N_1, C_1, (A_v)_{v\in N_1}, k)$ and $(N_2, C_2, (A_v)_{v\in N_2}, k)$, where $C_1 = D_{k-2} \cup \{a_1, b_1\}$, $C_2 = D_{k-2} \cup \{a_2, b_2\}$, and define a new election $(N, C, (A'_v)_{v\in N_1\cup N_2}, k)$ as follows. Set $N = N_1 \sqcup N_2$ and $C = D_{k-2} \cup \{x, y\} \cup \{a, b\}$. Again, modify the voters' ballots accordingly: for each $v \in N_1$ replace each occurrence of a_1 and b_1 in A_v with b and a, respectively, and for each $v \in N_2$ replace each occurrence of a_2 and b_2 in A_v with a and b, respectively. Then, set $A'_v = A_v \cup \{x\}$ if $v \in N_1$ and $A'_v = A_v \cup \{y\}$ if $v \in N_2$. Let $s : N_1 \mapsto N_2$ be the natural bijection between the sets of voters in the two isomorphic elections that $E(j,k)$ is built from; we will say that $s(v)$ is the counterpart of v, and v is the counterpart of $s(v)$. We collect a few simple properties of $E(j,k)$ in the following observation.

▶ **Proposition 4.** *For all $1 \le j < k$, the election $E(j,k)$ has the following properties:*

1. *$A_v \cap D = A_{s(v)} \cap D$,*
2. *v approves a (resp., b) if and only if $s(v)$ approves b (resp., a), and*
3. *v approves x (resp., y) if and only if $s(v)$ approves y (resp., x).*

Consider the committee $W = D_{k-2} \cup \{x, a\}$. The election $E(j,k)$ satisfies two important properties with respect to W, which will be needed in the proof of Theorem 3. We state them in the following lemma. To make the lemma easier to use, we define $\delta : \mathbb{N} \times \mathbb{N} \mapsto \mathbb{R}$ as

$$\delta(j,k) = \frac{j!}{\prod_{j'=0}^{j}(k-j')}.$$

▶ **Lemma 5** (★). *For all $1 \le j < k$, the election $E(j,k)$ and the committee $W = D_{k-2} \cup \{x, a\}$ have the following properties:*

1. *$\Delta(W, a, b) = \delta(j,k)$, and*
2. *for every voter $v \in N$ we have $|A_v \cap W| \ge k - (j+1)$.*

Proposition 4 together with Lemma 5 imply the following corollary.

▶ **Corollary 6.** *Consider election $E(j,k)$ with $1 \le j < k$ and committees $W = D_{k-2} \cup \{x, a\}$ and $W' = D_{k-2} \cup \{y, b\}$. It holds that $\Delta(W, a, b) = \delta(j,k)$, and hence (a, b) is a good swap. Moreover, $\Delta(W', b, a) = \delta(j,k)$, and hence (b, a) is a good swap.*

4.3 Level up: Election $E^t(j,k)$

We now combine t copies of the election $E(j,k)$ into a single election, which we will call $E^t(j,k)$. This election is the building block in our final construction that is responsible for the up-and-down movement within columns, as shown in Figure 3.

Construction 3. [Election $E^t(j,k)$]. Let $t \in \mathbb{N}$. For each $i \in [t]$, we consider an election $E_i = (N_i, C_i, (A_v)_{v \in N_i}, k)$, where E_i has structure $E(j,k)$ with $C_i = D_{k-2} \cup \{a_i, b_i\} \cup \{x_i, y_i\}$. For each $i = 1, \dots, t-1$, we identify b_i with a_{i+1}, and relabel $a_1, \dots, a_t, b_t$ as $c_1, c_2, \dots, c_{t+1}$. Furthermore, we identify x_i with y_{i+1} as well as y_i with x_{i+1} for $i = 1, \dots t-1$, and we write $x = x_1 = y_2 = x_3 \dots$ and $y = y_1 = x_2 = y_3 \dots$. We then set $C = \{c_1, c_2, \dots, c_{t+1}, x, y\}$. Let $C_i^- = \{c_1, \dots, c_i\}$ and $C_i^+ = \{c_{i+1}, \dots, c_{t+1}\}$. For each $v \in N_i$, if $A_v \cap C_i^- \neq \varnothing$, we add all candidates in C_i^- to v's ballot, and if $A_v \cap C_i^+ \neq \varnothing$, we add all candidates in C_i^+ to v's ballot; that is, we set

$$A'_v = A_v \cup \bigcup_{\substack{X \in \{C_i^-, C_i^+\}: \\ X \cap A_v \neq \varnothing}} X.$$

Then, each voter $v \in N_i$ views the candidates $c_1, \dots c_i$ as *clones*: she either approves all or none of them. Similarly, she views $c_{i+1}, \dots, c_{t+1}$ as clones, too. Finally, let $N = \sqcup_{i=1}^{t} N_i$, and define $E^t(j,k) = (N, D_{k-2} \cup C, (A'_v)_{v \in N}, k)$. Since each $E(j,k)$ is an election with 2^j voters, we can easily calculate the number of voters in $E^t(j,k)$.

▶ **Proposition 7.** *The number of voters in $E^t(j,k)$ is $t2^j$.*

Consider an election E with structure $E^t(j,k)$ and voters $\sqcup_{i=1}^{t} N_i$, as in the above construction. Since in election E_i with structure $E(j,k)$ each voter v in N_i approves exactly one candidate from $\{a_i, b_i\}$, in $E^t(j,k)$ voter v also approves exactly one of c_i and c_{i+1}. So, by the above construction, in $E^t(j,k)$ for each $v \in N_i$ we have either $A_v \cap C = \{c_1, \dots, c_i\}$ or $A_v \cap C = \{c_{i+1}, \dots, c_{t+1}\}$. Consequently, in election $E^t(j,k)$ each of the swaps (c_i, c_{i+1}) and (c_{i+1}, c_i) can only change the satisfaction of voters in $N_i \subset N$.

We can therefore make some useful observations regarding good swaps in election $E^t(j,k)$ with respect to the committee $W = \{d_1, \dots, d_{k-2}, z, c\}$, where $z \in \{x, y\}$ and $c \in \{c_1, c_{t+1}\}$.

▶ **Proposition 8** (★).

(1) *If* $W_1 = D_{k-2} \cup \{x, c_1\}$*, then* $(c_1, c_2), (c_2, c_3), \ldots, (c_t, c_{t+1})$ *is a sequence of* t *good swaps starting from* W_1*, increasing the PAV score by* $t\delta(j, k)$*.*

(2) *If* $W_2 = D_{k-2} \cup \{y, c_{t+1}\}$*, then* $(c_{t+1}, c_t), (c_t, c_{t-1}), \ldots, (c_2, c_1)$ *is a sequence of* t *good swaps starting from* W_2*, increasing the PAV score by* $t\delta(j, k)$*.*

(3) *If* $W_3 = D_{k-2} \cup \{x, c_{t+1}\}$ *then* $\Delta(W_3, x, y) = -t\delta(j, k)$*.*

(4) *If* $W_4 = D_{k-2} \cup \{y, c_1\}$*, then* $\Delta(W_4, y, x) = -t\delta(j, k)$*.*

4.4 Final Election Instance E

Let k be the desired committee size, and let $k_1 = \lceil \log k \rceil$, $k_2 = k - k_1$. We will construct an election $E = (N, C, (A_v)_{v \in N}, k)$ together with an initial committee W_0 so that there exists a sequence of swaps of super-polynomial length starting from W_0. Briefly, E is obtained by combining k_1 elections of the form $E^t(2j, k_2 + 1)$ for $j = 1, \ldots, k_1$, with some modifications of the ballots.

Constructing the Instance Let $t = 2 \cdot \lceil \frac{k}{2} \rceil$, so that $t \in \{k, k+1\}$ and t is even. For each $i \in [k_1]$, let $C_i = \{c_{i,1}, \ldots, c_{i,t+1}, x_i, y_i\}$ and

$$E_i = (N_i, C_i \cup D_{k_2 - 1}, (A_v)_{v \in N_i}, k_2 + 1) \text{ with structure } E^t(2k_1 - 2(i-1), k_2 + 1). \tag{3}$$

The committee size of $k_2 + 1$ in this construction is chosen so that $k_2 - 1$ spots are reserved for dummy candidates D_{k_2}, one spot is reserved for one of x_i and y_i, and the last spot is reserved for one of the candidates $c_{i,j}, 1 \leq j \leq t+1$. We define E by merging these elections in the natural way, but we additionally modify some approvals. Informally, for all $i \in [k_1]$, we remove all candidates x_i, y_i and let the candidates in $C_{i+1} \setminus \{x_{i+1}, y_{i+1}\}$ take on the roles of x_i and y_i for voters N_i. That is, in election E we modify the ballots so that for each $i < k_1$ it holds that all voters in N_i who approve x_i in E_i instead approve of all $c_{i+1,j}$ with j odd, and all voters in N_i who approve y_i in E_i instead approve of all $c_{i+1,j}$ with j even. Furthermore, we add an additional dummy voter d_{k_2} to the election and identify d_{k_2} with x_{k_1}, so that every voter in N_{k_1} who previously approved x_{k_1} now approves d_{k_2} instead. More formally, let $E = (N, C, (A'_v)_{v \in N}, k_1 + k_2)$, where

$$N = \bigcup_{i=1}^{k_1} N_i, \qquad C = D_{k_2} \cup \bigcup_{i=1}^{k_1} (C_i \setminus \{y_i, x_i\}), \text{ and}$$

$$A'_v = \begin{cases} A_v \cup \{c_{i+1,j} \mid j \text{ is odd, } x_i \in A_v \text{ or } j \text{ is even, } y_i \in A_v\} \setminus \{y_i, x_i\} & \text{if } v \in N_i, i < k_1 \\ A_v \cup \{d_{k_2} \mid x_{k_1} \in A_v\} \setminus \{x_{k_1}, y_{k_1}\} & \text{if } v \in N_{k_1} \end{cases}$$

To see why the size of E is polynomial in k, note that $|C| = k_2 + k_1 \cdot (t+1) \leq 2k \log k$, and by Proposition 7 we have $|N_i| \leq t 2^{2(\lceil \log k \rceil) - 2(i-1)} \leq t \cdot 2^{2\lceil \log k \rceil} \leq 4k^2(k+1)$ and so $|N| = \sum_{i=1}^{k_1} |N_i| = \text{poly}(k)$.

4.5 Adversarial Better Response

We are now ready to prove our lower bound of $\Omega(k^{\log k})$. We refer to our sequence of swaps as adversarial better response, because these are the swaps that an agent that points out improvements of the existing state, but acts adversarially, might choose to show us.

Let the initial committee be

$$W_0 = (c_{1,t+1}, c_{2,t+1}, \dots, c_{k_1-1,t+1}, c_{k_1,1}, d_1, \dots d_{k_2}),$$

so $|W_0| = k_1 + k_2 = k$. We will exhibit a sequence of good swaps that results in the final committee

$$(c_{1,t+1}, c_{2,t+1}, \dots, c_{k_1,t+1}, d_1, \dots, d_{k_2}).$$

Our basic building block is the sequence $\mathbf{Y}_1 = \oplus_{j=1}^{t}(c_{1,j}, c_{1,j+1})$. Let $\mathbf{X}_1^1 = \mathbf{Y}_1$ and $\mathbf{X}_1^0 = \mathbf{Y}_1^{-1}$. Further, for $i > 1$ define

$$\mathbf{X}_i^0 = \oplus_{j=1}^{t}\left((c_{i,t-j+2}, c_{i,t-j+1}) \oplus \mathbf{X}_{i-1}^{j-1 \bmod 2}\right), \quad \mathbf{X}_i^1 = \oplus_{j=1}^{t}\left((c_{i,j}, c_{i,j+1}) \oplus \mathbf{X}_{i-1}^{j-1 \bmod 2}\right).$$

Our proof shows that 0^+-ls-PAV will perform the sequence of swaps $\mathbf{X}_{k_1}^1$ when run on election E (constructed in Section 4.4) with initial committee W_0. That is, $\mathbf{X}_{k_1}^1$ is a sequence of good swaps. Further, since $|\mathbf{X}_i^0| = |\mathbf{X}_i^1|$, the length of $\mathbf{X}_i^\ell$, $\ell \in \{0,1\}$, is $t(|\mathbf{X}_{i-1}^\ell| + 1)$ and $|\mathbf{X}_1^\ell| = t \geq k$. Hence, $\mathbf{X}_{k_1}^1$ has length $\Omega(t^{k_1}) = \Omega(k^{\log k})$, i.e., it is a sequence of good swaps with super-polynomial length.

4.6 Extension to a Fixed Pivoting Rule

We adapt the proof of Theorem 3 to a natural non-adversarial setting. An intuitive method to select swaps is to consider a fixed ordering on the candidates $C = \{c_1, \dots, c_m\}$, for example $c_1 < \dots < c_m$. To find a good swap $(c', c) \in W \times (C \setminus W)$, we go over the candidates in $C \setminus W$, in the order suggested by $<$; for each $c \in C \setminus W$, we go over candidates in W, in the order suggested by $<$, to find c' such that (c', c) is a good swap. That is, we consider a lexicographic ordering on pairs (c, c') (where c is to be added and c' is to be removed from the committee) induced by the order $<$. In light of this, we call the corresponding pivoting rule *lexicographic better response.*

▶ **Theorem 9 (★).** *For any $k \geq 1$ there exists a committee election with $\mathrm{poly}(k)$ voters and a committee W_0 such that executing 0^+-ls-PAV with lexicographic better response from W_0 results in $\Omega(k^{\log k})$ swaps.*

Due to space constraints, we relegate the proof of Theorem 9 to the extended version of the paper.

5 Discussion

We have shown that if ε can be arbitrarily small, the running time of ε-ls-PAV with lexicographic better response may be super-polynomial, resolving the open question of Aziz et al. [2]. Thus, while using very small values of ε would be attractive both in terms of obtaining a more decisive rule and in terms of providing fairness guarantees to small minorities of voters, this would come at a cost of superpolynomial execution time in the worst case. While a similar result for best response remains elusive, our simulations (see the extended version of the paper) shows that, at least empirically, better response is preferable to best response on both synthetic and real-world datasets. We note that our lower bound does not preclude the possibility that an outcome of 0^+-ls-PAV can be found in polynomial time by other means; it is an interesting open question whether this is indeed possible.

References

1 Haris Aziz, Markus Brill, Vincent Conitzer, Edith Elkind, Rupert Freeman, and Toby Walsh. Justified representation in approval-based committee voting. *Social Choice and Welfare*, 48(2):461–485, 2017.

2 Haris Aziz, Edith Elkind, Shenwei Huang, Martin Lackner, Luis Sánchez-Fernández, and Piotr Skowron. On the complexity of extended and proportional justified representation. In *Proceedings of AAAI'18*, pages 902–909, 2018.

3 Haris Aziz, Serge Gaspers, Joachim Gudmundsson, Simon Mackenzie, Nicholas Mattei, and Toby Walsh. Computational aspects of multi-winner approval voting. In *Proceedings of AAMAS'15*, pages 107–115, 2015.

4 Sandro Bettin, Giuseppe Molteni, and Carlo Sanna. Small values of signed harmonic sums. *Comptes Rendus Mathematique*, 356(11-12):1062–1074, 2018.

5 Markus Brill, Rupert Freeman, Svante Janson, and Martin Lackner. Phragmén's voting methods and justified representation. *Mathematical Programming*, 203(1):47–76, 2024.

6 Markus Brill and Jannik Peters. Robust and verifiable proportionality axioms for multiwinner voting. In *Proceedings of ACM EC'23*, 2023.

7 Dave Buchfuhrer, Michael Schapira, and Yaron Singer. Computation and incentives in combinatorial public projects. In *Proceedings of ACM EC'10*, pages 33–42, 2010.

8 Shaddin Dughmi. A truthful randomized mechanism for combinatorial public projects via convex optimization. In *Proceedings of ACM EC'11*, pages 263–272, 2011.

9 Svante Janson. Phragmén's and Thiele's election methods. *arXiv preprint arXiv:1611.08826*, 2016.

10 Martin Lackner and Piotr Skowron. *Multi-winner voting with approval preferences*. Springer Nature, 2023.

11 George L Nemhauser, Laurence A Wolsey, and Marshall L Fisher. An analysis of approximations for maximizing submodular set functions—I. *Mathematical Programming*, 14:265–294, 1978.

12 Dominik Peters, Grzegorz Pierczyński, Nisarg Shah, and Piotr Skowron. Market-based explanations of collective decisions. In *Proceedings of AAAI'21*, pages 5656–5663, 2021.

13 Dominik Peters and Piotr Skowron. Proportionality and the limits of welfarism. In *Proceedings of ACM EC'20*, pages 793–794, 2020.

14 Luis Sánchez-Fernández, Edith Elkind, Martin Lackner, Norberto Fernández, Jesús Fisteus, Pablo Basanta Val, and Piotr Skowron. Proportional justified representation. In *Proceedings of AAAI'17*, pages 670–676, 2017.

15 Piotr Skowron. Proportionality degree of multiwinner rules. In *Proceedings of ACM EC'21*, pages 820–840, 2021.

16 Piotr Skowron, Piotr Faliszewski, and Jérôme Lang. Finding a collective set of items: From proportional multirepresentation to group recommendation. *Artificial Intelligence*, 241:191–216, 2016.