Social Choice and Welfare (2025) 64:221–261
https://doi.org/10.1007/s00355-023-01498-8

ORIGINAL PAPER

Dynamic proportional rankings

Jonas Israel[1] · Markus Brill[1,2]

Received: 13 May 2022 / Accepted: 7 November 2023 / Published online: 20 March 2024

Abstract
Proportional ranking rules aggregate approval-style preferences of agents into a collective ranking such that groups of agents with similar preferences are adequately represented. Motivated by the application of live Q&A platforms, where submitted questions need to be ranked based on the interests of the audience, we study a dynamic extension of the proportional rankings setting. In our setting, the goal is to maintain the proportionality of a ranking when alternatives (i.e., questions)—not necessarily from the top of the ranking—get selected sequentially. We propose generalizations of well-known ranking rules to this setting and study their monotonicity and proportionality properties. We also evaluate the performance of these rules experimentally, using realistic probabilistic assumptions on the selection procedure.

1 Introduction

From "ask-me-anything" sessions to panel discussions and town hall meetings, an increasing number of both virtual and in-person discussion formats are enhanced by digital tools that aim to make the event more interactive and responsive to the audience. Using live Q&A platforms such as *slido* (https://www.sli.do), *Mentimeter* (https://www.mentimeter.com) or *Pigeonhole Live* (https://pigeonholelive.com), participants in the audience can submit questions and upvote questions submitted by others; a moderator then selects the most popular questions for the discussion. By reducing barriers to participation (e.g., by allowing anonymous submissions), these tools aim to better represent the diversity in the audience.[1]

[1] For instance, a blog post on the *slido* website from March 2021 states that running Q&A sessions can "support diversity of thought and opinion in the workplace" (https://blog.slido.com/fix-your-qa-session/).

✉ Jonas Israel
j.israel@tu-berlin.de

Markus Brill
markus.brill@warwick.ac.uk

1 Research Group Efficient Algorithms, TU Berlin, Berlin, Germany

2 Department of Computer Science, University of Warwick, Coventry, UK

The moderator of the discussion is presented with an aggregated list, in which audience questions are ranked by popularity (i.e., number of upvotes). Based on this ranking, the moderator then picks the next question. When selecting a question, it is usually not required to follow the ranking strictly; rather, the choice is at the moderator's discretion, allowing him or her to take into account other factors such as discussion flow. That being said, it is generally expected that questions at the top of the ranking are more likely to be selected than questions further down the list. After a question has been selected, it is removed from the ranking.

Ranking questions solely by popularity, though intuitively appealing, has a major downside: minority opinions might go completely unrepresented, even when the minority makes up a substantial proportion of the audience. To illustrate this phenomenon, which is often referred to as the "tyranny of the majority," consider a situation in which the audience is composed of two groups. One group makes up 60% of the entire audience and is only interested in questions related to topic A; the remaining 40% of participants are only interested in questions on a different topic B. Now, assuming that sufficiently many questions on topic A have been submitted, and that participants only upvote questions related to their own interest, questions on topic B are unlikely to appear anywhere near the top of the ranking, which is populated exclusively by questions on topic A. As a consequence, questions on topic B are very unlikely to be selected, despite the fact that these questions are supported by 40% of the audience.

In this paper, we propose an approach to avoid the problem of underrepresenting minority opinions. Specifically, we model the scenario described above as a *proportional representation* problem and employ ranking algorithms based on (approval-based) proportional voting rules (Aziz et al. 2017; Lackner and Skowron 2022). The algorithms we consider aggregate the upvotes of the participants into a *proportional* ranking over questions, such that each minority (i.e., group of participants with similar preferences) is represented in the ranking to an extent that is proportional to the group's size. Whenever a question is selected by the moderator, our methods dynamically recompute the ranking, pushing questions supported by underrepresented groups closer to the top.

Proportional representation in this setting can be understood in two distinct ways. Firstly, audience members can be represented via the selection of questions they upvoted. In the above example, fair representation of the two groups would mean that 60% of the questions selected by the moderator are on topic A and 40% are on topic B. This form of representation cannot be guaranteed without additional assumptions on the behavior of the moderator. Alternatively, audience members can be represented in the ranking if questions they have upvoted are placed near the top of the ranking, ensuring visibility of these questions to both the moderator and the rest of the audience. In the example, this would mean that around 60% of any prefix of the ranking (e.g., 3 out of the top-5 questions and 6 out of the top-10 questions) should consist of questions on topic A. We consider both versions of representation in this paper.

On a technical level, our point of departure is the theory of proportional rankings (Skowron et al. 2017), which studies how a collective ranking over a set of alternatives can be constructed in such a way that majority and minority opinions are represented adequately. The question we are interested in is how proportional ranking algorithms can be adapted to the dynamic setting. More specifically, we ask:

> *How can the proportionality of a collective ranking be maintained in a dynamic setting, where alternatives get selected sequentially?*

To answer this question, we consider two well-known aggregation rules dating back to the nineteenth century: *sequential Phragmén* (defined by Phragmén 1894) and *sequential PAV* (defined by Thiele 1895). These two rules, together with a few variants of the latter, performed best in the analysis conducted by Skowron et al. (2017). For both rules, we propose two distinct generalizations to our setting: a *dynamic* variant and a *myopic* variant (see Sect. 3 for details). As a benchmark, we also consider the rule that simply orders questions by the number of received upvotes.

Our Contribution. In this paper, we formalize the setting of dynamic ranking rules and generalize the rules of Phragmén and Thiele to this setting (Sect. 3); we define a notion of satisfaction monotonicity and analyze to what extent the considered rules satisfy it (Sect. 4); we provide theoretical bounds regarding two different proportionality notions (Sect. 5); and we experimentally evaluate our dynamic ranking rules (Sect. 6). Omitted proofs and further details can be found in the Appendix.

Related work. Proportional representation is a fundamental desideratum in multiwinner elections (Monroe 1995; Faliszewski et al. 2017; Lackner and Skowron 2022). For approval preferences in particular, a wide variety of proportionality axioms have been studied (Aziz et al. 2017; Sánchez-Fernández et al. 2017; Janson 2018; Peters and Skowron 2020; Brill et al. 2024a; Brill and Peters 2023). Proportionality in the context of rankings has been considered in the aforementioned paper by Skowron et al. (2017) and (for linear preferences) by Schulze (2011).

Notions of fairness over multiple elections among a fixed set of voters have received considerable attention in previous years. This line of work includes, e.g., the study of long-term fairness over different decisions (Freeman et al. 2017; Lackner 2020), single decisions under changing preferences (Tennenholtz 2004; Boutilier and Procaccia 2012; Parkes and Procaccia 2013; Oren and Lucier 2014; Hemaspaandra et al. 2017), and storable votes (Casella 2012).

In a practical attempt to avoid the underrepresentation of minorities, the live Q&A app *SpeakUp* (https://speakup.digital/) allows audience members to add *attributes* (relating to, e.g., gender or education) to submitted questions. The moderator can then manually filter questions with attributes that have been underrepresented in the discussion. Requiring organizers to identify relevant attributes poses the risk of overlooking important subgroups or introducing unwanted biases; it also presumes the willingness of participants to reveal potentially sensitive information. In contrast, the ranking algorithms considered in this paper do not require attributes in order to ensure the protection of minorities. Rather than using attributes to define and represent *minority groups* (an approach known as *descriptive representation* (Mansbridge 1999)), the ranking algorithms in this paper—not having access to attributes but only to preferences—aim to proportionally represent *minority opinions*.

2 Preliminaries

We briefly introduce some basic concepts from the theory of approval-based preference aggregation; for details, see the survey by Lackner and Skowron (2022). Let C be a finite set of candidates and $N = \{1, \dots, n\}$ a finite set of voters. An *(approval) profile* $A = (A_1, \dots, A_n)$ is a list that contains, for each $i \in N$, the approval set $A_i \subseteq C$ of voter i. Given an approval profile A and a candidate $c \in C$, we let $N_c = \{i \in N : c \in A_i\}$ denote the *supporters* of c. The *approval score* of c is given by $|N_c|$. In the motivating application, C consists of all submitted questions and N consists of all participants who have upvoted at least one question. Therefore, for each voter $i \in N$, the set $A_i \subseteq C$ of upvoted questions is nonempty by assumption.[2]

To measure the satisfaction of a group of voters $V \subseteq N$ with a set $S \subseteq C$ of candidates, we often use the *average satisfaction* of V with S, i.e.,

$$\text{avg}_V(S) = \frac{1}{|V|} \cdot \sum_{i \in V} |A_i \cap S|.$$

For a finite set S, we let $\mathcal{L}(S)$ denote the set of all linear orders, or *rankings*, over S. We often write a ranking $r \in \mathcal{L}(S)$ as a sequence $r = (r_1, r_2, \dots, r_{|S|})$, and for $j \leq |S|$, we let $r_{\leq j}$ denote the set $\{r_1, r_2, \dots, r_j\} \subseteq S$ of the first j elements in r. Accordingly we define $r_{\leq 0} = \emptyset$.

An approval-based *ranking rule* maps an approval profile A to a ranking $r \in \mathcal{L}(C)$ of all candidates. Note that all rules may encounter ties; we assume that a priority ordering over candidates is used as a tiebreaker. In the motivating example, the submission time of a question yields a natural priority ordering. We will make use of the following three (non-dynamic) ranking rules.

- **Approval Voting (AV)** ranks the candidates according to their approval score. This rule is not proportional and we use it mainly as a benchmark.
- **Sequential PAV (seqPAV)** (Thiele 1895; Janson 2016; Aziz et al. 2017) ranks candidates iteratively, in each iteration choosing an unranked candidate maximizing the marginal contribution in terms of weighted voter satisfaction. Formally, for a subset $S \subseteq C$ of candidates, define its score by

  $$\text{sc}(S) = \sum_{i \in N} \sum_{j=1}^{|A_i \cap S|} \frac{1}{j}.$$

  If k candidates have already been ranked, the marginal contribution of an unranked candidate c is given by $\text{mc}(c) = \text{sc}(r_{\leq k} \cup \{c\}) - \text{sc}(r_{\leq k})$. SeqPAV then selects an unranked candidate maximizing the marginal contribution and ranks it in the $(k + 1)$-st position.

[2] The rationale for not considering “inactive” participants as voters is that their preferences are unknown and can therefore not be considered in the ranking of questions. Moreover, this modeling choice makes the proportionality guarantees in Sect. 5 stronger, as the size of voter groups is measured as a fraction of “active” participants only.

- **Sequential Phragmén** (Phragmén 1894; Janson 2016; Brill et al. 2024b) can be described in terms of voters buying candidates with credits they earn over time.[3] Assume that every candidate costs 1 credit and all voters start with 0 credits. All voters earn credits continuously over time at a constant and identical rate, e.g. by a rate of 1 credit per time unit. As soon as a group of voters who all approve the same candidate c together own 1 credit, they immediately buy that candidate. That means that each voter approving c will spend their total current budget on that candidate (as it is the first point in time where the supporters of c collectively own 1 credit). At this point, their balance is thus reset to 0 and candidate c is added in the next position of the ranking. The voters then continue to earn credits until the next candidate is bought, and so on. This is done until all candidates are ranked.

3 Dynamic ranking rules

In this section, we formally introduce the setting of dynamic ranking rules and we adapt existing (non-dynamic) ranking rules to this setting.

The input of a dynamic ranking rule consists of two parts: an approval profile and a (potentially empty) sequence of candidates that have already been "implemented" or "executed"; the output is a ranking of all not-yet-implemented candidates. To formalize this notion, we let $X = (x_1, x_2, \dots, x_j)$ denote the sequence of implemented candidates (where $j \in \{0, \dots, |C|\}$); whenever the order of elements in X does not matter, we slightly abuse notation and treat X as the set $X = \{x_1, x_2, \dots, x_j\}$.

Definition 1 An *(approval-based) dynamic ranking rule* $\mathcal{R}$ maps a profile A and a sequence $X = (x_1, x_2, \dots, x_j)$ of candidates to a ranking $\mathcal{R}(A, X) \in \mathcal{L}(C \setminus X)$.

Applying a dynamic ranking rule to a sequential selection process (as outlined in the introduction) is now straightforward: At the beginning, when no candidate has been implemented yet, $X = ()$ and the ranking $\mathcal{R}(A, ())$ ranks all candidates in C. Given this ranking, a *decision maker (DM)* selects a candidate $x_1 \in C$ to be implemented. The updated ranking of the remaining candidates is then given by $\mathcal{R}(A, (x_1))$, and the process is repeated. At iteration $t \in \mathbb{N}$, when $t-1$ candidates have been implemented and thus $X = (x_1, x_2, \dots, x_{t-1})$, we let r^t denote the ranking $\mathcal{R}(A, X) \in \mathcal{L}(C \setminus X)$ from which the DM can make a choice.

We will sometimes make the assumption that the DM only ever implements candidates that appear near the top of the ranking. In this *depth-restricted setting*, we are given a natural number h and we assume that $x_t \in r^t_{\leq h}$ for all time steps t. This setting models situations in which the DM does not have the resources (or the ability) to consider the whole ranking. When using the depth-restricted setting, we usually assume $h \geq 2$. This is because in the case of $h = 1$, the DM has no choice at all

[3] An equivalent formulation of this method is in terms of a load balancing procedure (Janson 2016; Brill et al. 2024b).

and always picks the top element in the ranking. It is straightforward to verify that in this situation, all the dynamic ranking rules we consider degenerate to their non-dynamic counterparts studied in the context of proportional rankings Skowron et al. (2017). We make this statement precise in Proposition 4 at the end of this section.

The straightforward ranking rule AV trivially translates to the dynamic setting: When a candidate is implemented, it is simply removed from the ranking; the order between the remaining candidates does not change. To the best of our knowledge, AV is used in all of the live Q&A platforms mentioned in the introduction.

To address the issues of AV as discussed in Sect. 1, we now propose dynamic variants of *proportional* ranking rules. For a more detailed description of these rules, including pseudocode formulations, we refer to Appendix A.

- **Dynamic seqPAV** is a straightforward dynamization of seqPAV. It ranks candidates iteratively, in each iteration choosing an unranked candidate maximizing the marginal contribution in terms of weighted voter satisfaction. The score $\mathrm{sc}(S)$ of a subset $S \subseteq C$ of candidates is defined exactly as under seqPAV. However, we modify the notion of marginal contribution to also take into account the satisfaction derived from previously implemented candidates. If k candidates have already been ranked and a set X of already implemented candidates is given, the marginal contribution of an unranked candidate $c \in C \setminus (r_{\leq k} \cup X)$ is given by

  $$\mathrm{mc}_{\mathrm{dyn}}(c) = \mathrm{sc}(X \cup r_{\leq k} \cup \{c\}) - \mathrm{sc}(X \cup r_{\leq k}).$$

  Dynamic seqPAV adds in each round a candidate $c \in C \setminus (r_{\leq k} \cup X)$ maximizing $\mathrm{mc}_{\mathrm{dyn}}(c)$. X is treated as a set here, as the order of elements in X does not matter.

- **Dynamic Phragmén** proceeds in two phases. As before, voters buy candidates and every candidate has a cost of 1 credit. Voters do not start with 0 credits, however; they may have an initial *debt* due to previously implemented candidates they approve. The debts of voters are determined in the *first phase*, which iterates through the sequence X (starting with x_1) and, for each implemented candidate $x_j \in X$, divides the cost of 1 among the voters in N_{x_j}. More precisely, this assignment of debts is done in such a way that, in each iteration j, the maximum total debt across all voters in N_{x_j} is as small as possible. (The assignment of debts, therefore, mimics the assignment of loads in the load balancing formulation of sequential Phragmén.) We let $d_i \geq 0$ denote the total debt of voter $i \in N$ resulting from this first phase. In the *second phase*, we run sequential Phragmén to obtain the desired ranking of candidates in $C \setminus X$. At the beginning of this phase, each voter i has a credit balance of $-d_i \leq 0$. As in sequential Phragmén, voters continuously earn credits, and voters starting with debts can only participate in the purchase of a candidate once they have a positive balance.

These dynamic rules rank candidates in the same fashion as their non-dynamic counterparts, while taking the sequence X of previously implemented candidates into account. Note that the implementation *order* matters for dynamic Phragmén, but not for dynamic seqPAV. In particular, both dynamic rules coincide with their

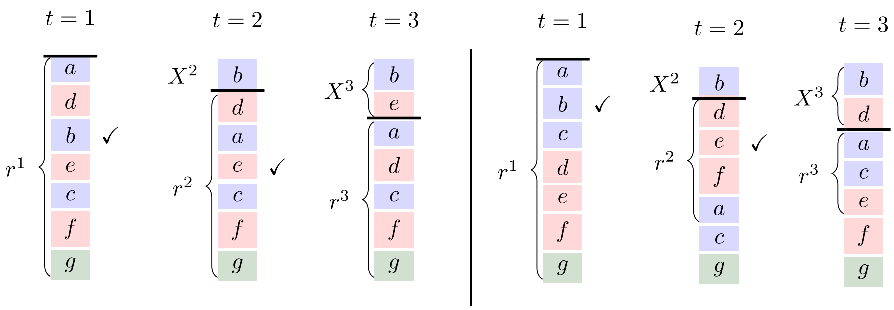

Fig. 1 Rankings discussed in Example 1. Candidates approved by the three voter groups V_{blue}, V_{red}, and V_{green} are highlighted in blue, red, and green, respectively. The rankings produced by the dynamic ranking rules are depicted on the left, the ones produced by the myopic rules on the right. The candidate that is chosen by the DM is marked with "✓" and appears in the sequence of implemented candidates X in the next iteration. In each iteration t, the horizontal bar separates the sequence X^t of implemented candidates (above the bar) from the ranking r^t over the remaining candidates in $C \backslash X^t$ (below the bar) (color figure online)

non-dynamic counterpart when $X = ()$. Moreover, the ranking among the remaining candidates does not change whenever the top-ranked candidate is implemented: if $r^t = (r_1, r_2, r_3, \dots)$ and $x_t = r_1$, then $r^{t+1} = (r_2, r_3, \dots)$.

We also consider two "myopic" dynamic ranking rules.

- **Myopic seqPAV:** In this myopic dynamization of seqPAV, we compute the marginal contribution of each candidate $c \in C \backslash X$ only with respect to the set X of previously implemented candidates, i.e., $\text{mc}_{\text{myopic}}(c) = \text{sc}(X \cup \{c\}) - \text{sc}(X)$. Then, we simply rank those candidates according to decreasing $\text{mc}_{\text{myopic}}(c)$-value.
- **Myopic Phragmén:** In this myopic dynamization of sequential Phragmén, we first run the first phase of dynamic Phragmén in order to determine the debts $\{d_i\}_{i \in N}$ of voters. Then, for each candidate $c \in C \backslash X$, we compute the voter debts that would result from adding candidate c to X (and running the first phase for one more iteration). Let the debts induced by candidate c be $\{d_i^c\}_{i \in N}$. Myopic Phragmén ranks the candidates in $C \backslash X$ according to increasing $\max_{i \in N_c} d_i^c$, breaking ties according to the second highest debt and so on. In other words, candidates whose addition would result in a low maximal debt are ranked higher than candidates for which the incurred maximal debt is larger.

Intuitively, myopic seqPAV and myopic Phragmén rank candidates according to their suitability of being the next implemented candidate. In contrast to dynamic seqPAV and dynamic Phragmén, this way of comparing candidates does not lead to rankings that are representative by themselves. In particular, both myopic rules coincide with AV when $X = ()$.

We illustrate these rules with a simple example. The rankings discussed in this example are depicted in Fig. 1.

***Example* 1** Let $C = \{a, b, c, d, e, f, g\}$ and assume alphabetic tiebreaking. Consider a set of $n = 9$ voters with the following approval sets:

$$5 \times \{a, b, c\}, \quad 3 \times \{d, e, f\}, \quad 1 \times \{g\}.$$

Let V_{blue} denote the group consisting of the five $\{a, b, c\}$-voters, V_{red} the group consisting of the three $\{d, e, f\}$-voters, and V_{green} the group consisting of the $\{g\}$-voter.

First, consider dynamic seqPAV and dynamic Phragmén. In the first iteration, both rules output $r^1 = (a, d, b, e, c, f, g)$, effectively alternating between candidates supported by voter groups V_{blue} and V_{red}. Let us assume that the DM first implements candidate $x_1 = b$, i.e., $X^2 = (b)$. Then, the two rules output $r^2 = (d, a, e, c, f, g)$. If the DM implements candidate $x_2 = d$ next (and thus $X^3 = (b, d)$), both rules output $r^3 = (a, e, c, f, g)$.

Next, consider myopic seqPAV and myopic Phragmén. In the first iteration, both rules (and AV) rank the candidates according to their approval scores: $r^1 = (a, b, c, d, e, f, g)$. After the implementation of b, both rules output $r^2 = (d, e, f, a, c, g)$, which differs from the AV ranking $r^2 = (a, c, d, e, f, g)$. If the DM then implements candidate $x_2 = d$, the two rules output $r^3 = (a, c, e, f, g)$.

In this example, all of our ranking rules demote candidate a in r^2 because voter group V_{blue} is already (partially) satisfied with $X^2 = (b)$. The myopic rules even rank *all* candidates supported by V_{red} higher than a in r^2, since implementing any one of them would yield a more proportional sequence X than implementing a would. This example effectively highlights the different philosophies of dynamic and myopic rules: When constructing the ranking for a given iteration t, the dynamic rules take into account not only the $t - 1$ candidates that have been implemented in previous iterations, but also the candidates that have already been ranked in r^t. The myopic rules, on the other hand, only consider the $t - 1$ already implemented candidates when constructing r^t. On a conceptual level, the dynamic rules provide a ranking that is proportional itself (with respect to the preferences of the voters and the already implemented candidates). In Example 1, this results in rankings that alternate between candidates supported by V_{blue} and V_{red}. Meanwhile, the myopic rules rank candidates greedily by their suitability to be selected next. In Example 1, this results in all candidates supported by one voter group being ranked consecutively.

All presented ranking rules can be computed in polynomial time. In order to give precise bounds on the running time, let $n = |N|$ denote the number of voters and $m = |C|$ the number of candidates.

Proposition 2 *Given an approval profile and a sequence of implemented candidates, the output of dynamic seqPAV can be computed in time $\mathcal{O}(m^3 n)$ and the output of myopic seqPAV can be computed in time $\mathcal{O}(m^2 n)$.*

Proposition 3 *Given an approval profile and a sequence of implemented candidates, the output of dynamic Phragmén can be computed in time $\mathcal{O}(m^2n^2)$ and the output of myopic Phragmén can be computed in time $\mathcal{O}(mn^2)$.*

Thus, the myopic versions of both sequential PAV and sequential Phragmén are asymptotically faster to compute (by a factor of m) compared to their dynamic counterparts. The proofs of Propositions 2 and 3 can be found in Appendix A, where we also provide pseudocode for computing the respective rules.

An important assumption in our model is that the DM has some flexibility as to which candidates to implement. We conclude this section by showing that, in the absence of this flexibility, the rules we consider degenerate to the respective rules in setting of (non-dynamic) proportional rankings Skowron et al. (2017).

Proposition 4 *Consider the depth-restricted setting with $h = 1$ and fix a profile A. Let r^{PAV} (respectively, r^{Phr}) be the ranking returned by the (non-dynamic) ranking rule sequential PAV (respectively, sequential Phragmén). The following statements hold for each iteration $t \in \mathbb{N}$.*

(i) *For dynamic seqPAV and dynamic Phragmén, the concatenation (X^t, r^t) of the sequence X^t of previously implemented candidates and the ranking r^t is identical to the ranking returned by the respective non-dynamic ranking rule: for dynamic seqPAV, $(X^t, r^t) = r^{\mathrm{PAV}}$; for dynamic Phragmén, $(X^t, r^t) = r^{\mathrm{Phr}}$.*

(ii) *For dynamic and myopic seqPAV and dynamic and myopic Phragmén, the sequence X^t of implemented candidates is identical to the $(t-1)$-prefix of the ranking returned by the respective non-dynamic ranking rule: for dynamic and myopic seqPAV, $X^t = r^{\mathrm{PAV}}_{\leq t-1}$; for dynamic and myopic Phragmén, $X^t = r^{\mathrm{Phr}}_{\leq t-1}$.*

Proof We first prove statements (i) and (ii) for dynamic seqPAV and dynamic Phragmén. For dynamic seqPAV, the statements hold for $t = 1$ because $X^1 = ()$ and $r^1 = r^{\mathrm{PAV}}$. At each iteration $t \geq 1$, the top-ranked element x_t of ranking r^t is implemented, so that $X^{t+1} = X^t \cup \{x_t\}$. In the ranking $r^{t+1} \in \mathcal{L}(C \backslash X^{t+1})$, the order among the candidates is exactly the same as in r^{PAV}: For each $k \geq 0$, we have

$$X^{t+1} \cup r^{t+1}_{\leq k} = X^t \cup \{x_t\} \cup r^{t+1}_{\leq k} = X^t \cup r^t_{\leq k+1} = r^{\mathrm{PAV}}_{\leq t+k}.$$

Therefore, the marginal contribution $\mathrm{mc}_{\mathrm{dyn}}(c)$ of a candidate $c \in C \backslash X^{t+1}$ with respect to $X^{t+1} \cup r^{t+1}_{\leq k}$ (as computed by dynamic seqPAV) is identical to the marginal contribution $\mathrm{mc}(c)$ of c with respect to $r^{\mathrm{PAV}}_{\leq t+k}$ (as computed by non-dynamic sequential PAV). It follows that $X^t = r^{\mathrm{PAV}}_{\leq t-1}$ and $(X^t, r^t) = r^{\mathrm{PAV}}$ for all $t \in \mathbb{N}$.

For dynamic Phragmén, a similar argument holds: Candidates are implemented exactly in the order in which they are ranked in r^{Phr}. To see this, consider an iteration $t \in \mathbb{N}$. At the end of the first phase of dynamic Phragmén, the debt d_i of voter i corresponds exactly to the amount of money that voter i has spent on candidates in X^t under (non-dynamic) sequential Phragmén. Therefore, in the second phase of dynamic Phragmén, the candidates in $C \backslash X^t$ are ranked exactly as they are ranked under sequential Phragmén.

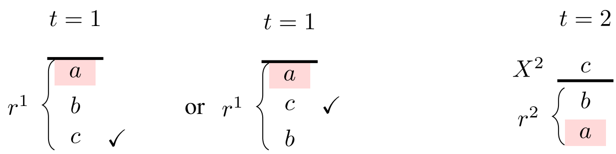

Fig. 2 Rankings discussed in Example 2. All rules considered here either output $r^1 = (a, b, c)$ or $r^1 = (a, c, b)$. If the DM chooses to implement candidate $x_1 = c$, all of the rules—except AV—output $r^2 = (b, a)$ in the second iteration. This violates an intuitive understanding of *monotonicity* for the voter with ballot $\{a\}$

For myopic seqPAV and myopic Phragmén, statement *(ii)* holds because, for each iteration $t \geq 1$, the top-ranked candidate in r^t is, by definition, exactly the candidate that will be ranked next by the respective non-dynamic ranking rule. □

4 Monotonicity of voter satisfaction

We start our analysis of dynamic ranking rules by considering the satisfaction of voters during the sequential selection process. In doing so, we assume that voters derive satisfaction not only from implemented candidates they approve, but also—possibly to a lesser extent—from approved candidates appearing near the top of the ranking: high positions in the ranking come with increased attention (and, presumably, high selection probabilities in future iterations) for the respective candidates. In particular, improved ranking positions of supported candidates can be viewed as a kind of compensation for (groups of) voters who are not (yet) well-represented by the implemented candidates. To make this concrete, consider an iteration t, where the DM is confronted with ranking r^t and chooses to implement candidate x_t. Following the logic outlined above, it might be natural to expect that voters *not* approving x_t (or, more precisely, the candidates approved by these voters) should get a "boost" in the ranking. At the very least, it seems reasonable to expect that the satisfaction of such voters with the new ranking r^{t+1} is at least as high as with the old ranking r^t. AV trivially satisfies this property, which we informally refer to as *satisfaction monotonicity*. Somewhat surprisingly, however, the following simple example demonstrates that this intuitive monotonicity notion is not achievable for dynamic ranking rules that satisfy a minimal degree of representativeness. The rankings discussed in this example are depicted in Fig. 2.

***Example* 2** Consider the following profile with 7 voters:

$$1 \times \{a\}, \quad 3 \times \{b\}, \quad 3 \times \{a, c\}.$$

All rules considered in this paper rank the approval winner a first in r^1. If the DM chooses to implement candidate $x_1 = c$, all of our rules—except AV—output $r^2 = (b, a)$ in the second iteration. Intuitively, the rules give more voting power to the 3 supporters of b (all of which are unrepresented by c) than to the 4 supporters of a (3 of which are already partially represented). Observe that the satisfaction of the voter only approving a decreases when going from r^1 to r^2, despite the fact that this voter does not approve the candidate being implemented.

Example 2 can be turned into an impossibility result: Every dynamic ranking rule that (i) ranks the approval winner at the top in the first iteration and (ii) gives priority to less satisfied voter groups fails satisfaction monotonicity.

4.1 Bounds for satisfaction monotonicity

The following definition is motivated by the question whether monotonicity failures can be prevented by moving to the depth-restricted setting and putting lower bounds on the size of voter groups for which monotonicity should hold.

Definition 5 For $h \geq 1$ and $\alpha \in (0, 1]$, a dynamic ranking rule satisfies (h, α)-*monotonicity* if, for all profiles and all groups of voters $V \subseteq N$ of size $|V| \geq \alpha \cdot |N|$, the following holds for every iteration t:

$$\text{If } x_t \notin \bigcup_{i \in V} A_i, \text{ then } \operatorname{avg}_V(r^{t+1}_{\leq h}) \geq \operatorname{avg}_V(r^t_{\leq h}).$$

That is, (h, α)-monotonicity requires that satisfaction monotonicity holds for groups that make up at least an α-fraction of the electorate, and when measuring satisfaction with respect to the first h positions in a ranking.

AV trivially satisfies (h, α)-monotonicity for all h and all α. On the other hand, all other considered rules violate this notion unless we consider rather large groups of voters.

Proposition 6 *Consider the depth-restricted setting for some $h \geq 3$. Then, dynamic seqPAV and dynamic Phragmén fail to satisfy (h, α)-monotonicity for all $\alpha < \frac{6}{2h+5}$. Furthermore, myopic seqPAV and myopic Phragmén fail to satisfy (h, α)-monotonicity for all $\alpha < \frac{1}{h}$.*

Proof sketch We first consider both rules based on seqPAV and then extend the argument to the other two rules. Starting with dynamic seqPAV, let $h = 3$ and $j = 6 \cdot y$ for some $y \in \mathbb{N}$ and consider the profile given by

$$2 \times \{a\}, \quad 15 \times \{a, b\}, \quad \left(\frac{j}{2} + 6\right) \times \{b\}, \quad 10 \times \{c\},$$
$$10 \times \{d\}, \quad (j + 6) \times \{a, c, d\}, \quad \left(\frac{j}{3} + 12\right) \times \{e\}.$$

Initially, dynamic seqPAV outputs $r^1 = (a, b, c, d, e)$. After the DM implements $x_1 = b$, dynamic seqPAV outputs $r^2 = (c, d, e, a)$ (where the ordering of c and d might depend on tie-breaking). Now consider the voter group V consisting of all the voters with approval sets $\{a\}$ or $\{a, c, d\}$. In the first iteration, the average satisfaction of this group with the $h = 3$ top-ranked candidates is $\text{avg}_V(r^1_{\leq 3}) = \frac{1}{8+j} \cdot (14 + 2j)$; in the second iteration, it is only $\text{avg}_V(r^2_{\leq 3}) = \frac{1}{8+j} \cdot (12 + 2j)$. For $j \to \infty$, this group of voters makes up nearly 6/11 of the electorate (and the rankings remain unchanged). To extend this example to the case of $h > 3$, we can introduce new candidates with $j/3 + 12$ supporters each (these can be seen as "copies" of candidate e and all its supporters).

For myopic seqPAV, consider the following adaption of the profile where the two terms $j/3$ and $j/2$ in the previous profile are exchanged for j, and we add 4 additional voters approving candidate e only:

$$2 \times \{a\}, \quad 15 \times \{a, b\}, \quad (j + 6) \times \{b\}, \quad 10 \times \{c\},$$
$$10 \times \{d\}, \quad j + 6 \times \{a, c, d\}, \quad (j + 16) \times \{e\}.$$

Here, myopic seqPAV outputs $r^1 = (a, b, c, d, e)$. After the DM again implements $x_1 = b$, the rule outputs $r^2 = (c, d, e, a)$ (where the ordering of c, d and e might depend on tie-breaking). Again consider the same voter group V. As above we have $\text{avg}_V(r^1_{\leq 3}) = \frac{1}{8+j} \cdot (14 + 2j)$ and $\text{avg}_V(r^2_{\leq 3}) = \frac{1}{8+j} \cdot (12 + 2j)$. For $j \to \infty$, V makes up nearly 1/3 of the electorate in this profile. By introducing new candidates with $j/3 + 12$ supporters each (i.e., "copying" e and all its supporters) we can again extend the profile to hold for $h > 3$.

We will now argue that myopic Phragmén also fails to satisfy (h, α)-monotonicity on this example for all $j \in \mathbb{N}$. Recall that this rule computes, for each candidate $c \in C \backslash X$, debts incurred to all voters. The debts relate to buying the candidates in X (in the order of implementation) and afterwards buying candidate c. All candidates then get ranked by comparing the so computed debts of the voters lexicographically. In the first iteration $X = ()$ holds and thus myopic Phragmén is equivalent to AV. The ensuing ranking is $r^1 = (a, b, c, d, e)$, independent of j. Now assume that the DM implements candidate $x_1 = b$. In the second iteration each supporter of b has a debt of $\frac{1}{21+j}$, since there are $21 + j$ voters who approve b and they all share the price of 1 credit for buying b. Note that in this step it is always favorable to balance the debt induced by a candidate equally among its supporters (this might not be the case if a candidate that is only supported by very few voters got implemented before). Thus for every candidate $c \in C \backslash X$ we can compute the debt each of its supporters would have if c would be bought next by $s^{(2)}_c = \frac{1}{|N_c|}(1 + \sum_{i \in N_c} d_i)$, where d_i is voter i's debt induced by X. Myopic Phragmén now ranks the candidates in non-increasing order of $s_c^{(2)}$ since this is the relevant part in comparing the debts of all voters by their maximum as described in the definition of myopic Phragmén. We can compute

$$s_a^{(2)} = \frac{36+j}{(23+j)(21+j)} \quad \text{and} \quad s_c^{(2)} = s_d^{(2)} = s_e^{(2)} = \frac{1}{16+j}.$$

To prove that, for all $j \in \mathbb{N}$, myopic Phragmén fails to satisfy group implementation monotonicity in this example, we have to validate that $s_a^2 > s_e^2$ holds independent of j, which can be done in a straightforward manner:

$$93 + 8j > 0 \Leftrightarrow 576 + 52j + j^2 > 483 + 44j + j^2 \Leftrightarrow (36+j)(16+x) > (23+j)(21+j).$$

The claim regarding dynamic Phragmén can be shown in a similar manner using the first profile again. Here the computation gets far more technical as the debts that get compared during the ranking process of the candidates change in each step of the ranking and not just once per iteration as is the case for the myopic variant of the rule. Nevertheless, we obtain a system of inequalities that ensures that the candidates get ranked by dynamic Phragmén in a similar way as by dynamic seqPAV and thus dynamic Phragmén also violates the monotonicity axiom. We can then again check that these inequalities hold for all j. □

4.2 Weak satisfaction monotonicity

The examples used in the proof of Proposition 6 rely heavily on an implemented candidate that is co-approved with some candidate that is approved by a member of the group under consideration (i.e., there is a $c \in \bigcup_{k \in V} A_k$ with $\{c, x_t\} \subseteq A_i$ for some $i \in N$).[4] It thus makes sense to consider a weakening of the monotonicity axiom that excludes these cases from consideration.

Definition 7 For $h \geq 1$ and $\alpha \in (0, 1]$, a dynamic ranking rule satisfies *weak* (h, α) *-monotonicity* if, for all profiles and all groups of voters $V \subseteq N$ of size $|V| \geq \alpha \cdot |N|$, the following holds: For every iteration t where there is no $c \in \bigcup_{k \in V} A_k$ with $\{c, x_t\} \subseteq A_i$ for some $i \in N$, we have $\text{avg}_V(r_{\leq h}^{t+1}) \geq \text{avg}_V(r_{\leq h}^t)$.

This allows us to obtain positive results for the two myopic rules. Consider a group of voters V and any candidate c that a voter in V approves. If c is not supported by any voter (not necessarily in V) that also supports the candidate that gets implemented next, then the voting power or the debt (depending on the rule we are interested in) of c's supporters does not change from this iteration to the next. Since the voting power (or debt) of the supporters of other candidates can only decrease

[4] In the first instance in the proof of Proposition 6, for example, candidate a is approved by every voter in the group V, but there are also 15 voters who approve a together with b, the candidate that is implemented first.

(or increase, respectively), c's position in the ranking cannot drop when going from one iteration to the next. Thus, the following result holds.

Proposition 8 *Consider the depth-restricted setting for some* $h \geq 3$. *Then, myopic seqPAV and myopic Phragmén satisfy weak* (h, α)*-monotonicity for all values of* α.

On the other hand, for small values of α, both dynamic rules fail to satisfy this weaker axiom.

Proposition 9 *Consider the depth-restricted setting for some* $h \geq 3$. *Then, dynamic seqPAV fails to satisfy weak* (h, α)*-monotonicity for all* $\alpha < \frac{2}{4+h}$.

Proposition 10 *Consider the depth-restricted setting for some* $h \geq 3$. *Then, dynamic Phragmén fails to satisfy weak* (h, α)*-monotonicity for all* $\alpha < \frac{2}{5+h}$.

This means that, if $h = 3$, then there are instances where a group V of voters that makes up nearly one quarter of the electorate might encounter the following situation: A candidate is selected that is neither approved by any voter in V nor by any voter outside of V approving some candidate in $\bigcup_{i \in V} A_i$, but still the average satisfaction of V drops in the next ranking under dynamic Phragmén. For dynamic seqPAV, this can even happen to groups that make up nearly two sevenths ($\approx 29\%$) of the electorate.

Despite the mostly negative results in this section, we rarely found monotonicity violations of any kind in our experiments (see Sect. 6).

5 Proportional representation

We now turn to analyzing the proportionality that is provided by our dynamic ranking rules. The following two sections capture different perspectives on representation, focusing on the proportionality of the ranking r^t at any given iteration t (Sect. 5.1) and on the proportionality of the set X of implemented candidates (Sect. 5.2).

5.1 Proportionality of rankings

In certain applications of dynamic ranking rules, such as the live Q&A platforms mentioned in the introduction, it is desirable for the ranking r^t to provide a representative overview of the opinions (or interests) of the voters at any given iteration t. In this section, we prove proportionality guarantees that are satisfied by ranking r^t for any fixed iteration t.

Measures for the proportionality of rankings have been proposed by Skowron et al. (2017). In particular, *κ-group representation* measures, informally speaking, how far down in the ranking a group of voters needs to go in order to obtain a given amount of satisfaction. In order to adapt the notion of κ-group representation to the dynamic ranking setting, we need the following notation. For iteration t, let $X^t = \{x_1, \ldots, x_{t-1}\}$ denote the set of candidates implemented in the first $t-1$ rounds and, for a group $V \subseteq N$ of voters, let $\lambda^t(V) = |\bigcap_{i \in V} A_i \setminus X^t|$ denote the cohesiveness of V with respect to the remaining candidates $C \setminus X^t$.

Definition 11 Let $\kappa(\alpha, \lambda)$ be a function from $(((0,1] \cap \mathbb{Q}) \times \mathbb{N})$ to $\mathbb{N}$. A dynamic ranking rule satisfies κ-*group representation* if the following holds for all profiles A, groups of voters $V \subseteq N$, rational numbers $\alpha \in (0,1]$, and integers $\lambda, t \leq |C|$: If $|V| \geq \alpha \cdot n$ and $\lambda^t(V) \geq \lambda$, then $\text{avg}_V(r^t_{\leq \kappa(\alpha,\lambda)}) \geq \lambda$.

In words: If a group V of voters makes up an α-fraction of the electorate and has at least λ commonly approved candidates remaining at iteration t, then this group derives an average satisfaction of at least λ from the candidates ranked in the top $\kappa(\alpha, \lambda)$ positions of ranking r^t.[5]

We also consider a quantitative proportionality measure from the field of approval-based *committee* voting, namely the proportionality degree as defined by Skowron (2021). We employ this measure in our setting by fixing a depth restriction h and view the candidates in $r^t_{\leq h}$ as the "committee". The proportionality degree then measures how proportionally the h highest-ranked candidates represent the electorate. Informally, it lower bounds the happiness of voter group V given a certain depth restriction depending on the size and cohesiveness of V.

Definition 12 Consider a profile A, a depth restriction $h \leq |C|$, and a function $g : \mathbb{N} \times \mathbb{N} \rightarrow \mathbb{R}$. We call a set of voters $V \subseteq N$ ℓ-large w.r.t. h if $|V| \geq \ell \cdot \frac{n}{h}$. A dynamic ranking rule $\mathcal{R}$ satisfies h-proportionality degree of g if for all ℓ-large sets of voters V and all iterations $t \in \mathbb{N}$, the ranking $r^{t+1} = \mathcal{R}(A, X^t)$ satisfies

$$\text{avg}_V(r_{\leq h}) \geq \min\left(\lambda^t(V), g(\ell, h)\right).$$

Let $\mathcal{G}_h$ be the set of all such h-proportionality degrees of $\mathcal{R}$. Then, we say that $\mathcal{R}$ satisfies proportionality degree of $d(\ell) = \min_h \sup_{g \in \mathcal{G}_h} g(\ell, h)$. This means that the proportionality degree of $\mathcal{R}$ is the best guarantee on the above objective that holds for all depth restrictions $h \leq |C|$.

Note that as was the case with the κ functions used for group representation that did not only depend on α and λ, but also on the set V and on the sequence X of previously implemented candidates, we simplify notation for the proportionality degree in a similar manner.

[5] A natural lower bound for $\kappa(\alpha, \lambda)$ is given by $\lceil \lambda / \alpha \rceil$. Note that the κ functions used in this section not only depend on α and λ, but also on the set V and on the sequence X of previously implemented candidates. In an attempt to simplify notation, we decided to not make this dependencies explicit in Definition 11.

5.1.1 Dynamic Phragmén

We first consider dynamic Phragmén. Recall that d_i denotes the initial debt of voter i at the end of the first phase of the method, and let $d^V_{\text{avg}} = \frac{1}{|V|}\sum_{i \in V} d_i$ denote the average debt of voters in V.

Theorem 13 *Dynamic Phragmén satisfies κ-group representation for*

$$\kappa(\alpha, \lambda) = \left\lceil \frac{2(\lambda + u + 1) + s \cdot |V|}{\alpha} \right\rceil,$$

where $u = |\bigcup_{i \in V} A_i \cap X|$ and $s = \sum_{i \in V}(d_i - d^V_{\text{avg}})^2$.

Observe that this function is increasing both in the number u of already implemented candidates that are approved by some voter in V and in the variance s of debts of voters in V. For the special case $X = ()$, Theorem 13 implies a group representation of $\lceil \frac{2\lambda+2}{\alpha} \rceil$ for (non-dynamic) sequential Phragmén. For $\lambda \geq 2$, this is an improvement over the κ-group representation bound of $\lceil \frac{5\lambda}{\alpha^2} + \frac{1}{\alpha} \rceil$ proved by Skowron et al. (2017).

The proof of Theorem 13 employs a connection between κ-group representation and the proportionality degree. In particular, we first prove a bound on the proportionality degree of dynamic Phragmén (Theorem 14), using a potential function approach that is similar to the one used by Skowron (2021) for the non-dynamic setting. Then, we establish a relationship between the proportionality degree and group representation (Lemma 15), and use it to translate the bound on the former into a bound on the latter.

Theorem 14 *Dynamic Phragmén satisfies proportionality degree of*

$$d(\ell) \geq \frac{\ell - 1}{2} - \frac{u}{2} - \frac{s \cdot |V|}{4},$$

where $u = |\bigcup_{i \in V} A_i \cap X|$ and $s = \sum_{i \in V}(d_i - d_{\text{avg}})^2$.

If $X = ()$, then $u = s = 0$ and we have the proportionality degree that was also proved by Skowron (2021) for the non-dynamic setting. We prove the results by a similar potential function approach as provided for the respective non-dynamic result (Skowron 2021) while taking into account the added complexity of the dynamic setting.

This result is independent of the iteration t, which makes it rather strong. On the other hand, the definition of proportionality degree (and thus this result) rely on a fixed value of h to determine ℓ-large groups of voters. As we show next, it is possible to translate the proportionality degree defined by Skowron (2021) into κ-group representation as defined by Skowron et al. (2017). Skowron (2021) mentions this connection but, to the best of our knowledge, this is the first explicit translation from one proportionality measure to the other.

Lemma 15 *Let $\mathcal{R}$ be a (dynamic) ranking rule which satisfies proportionality degree of $d(\ell)$ for all $\ell \in \mathbb{Q}$. Then, $\mathcal{R}$ satisfies κ-group representation for*

$$\kappa(\alpha, \lambda) = \left\lceil \frac{d^{-1}(\lambda)}{\alpha} \right\rceil,$$

where d^{-1} denotes the inverse of the function d.

Note that the proportionality degree is usually defined for $\ell \in \mathbb{N}$. For technical reasons, we need the function to be defined for all rational ℓ, which is, however, covered by our proof of Theorem 14.

Proof Consider a group $V \subseteq N$ of voters with proportion $\alpha = |V|/|N|$ and adapted cohesiveness $\lambda^t = |\bigcap_{i \in V} A_i \setminus X|$. Let $h = \lceil 1/\alpha \cdot d^{-1}(\lambda^t) \rceil$. Then, by construction, V is $d^{-1}(\lambda^t)$-large w.r.t. h since

$$|V| = \alpha \cdot |N| = \frac{|N| \cdot d^{-1}(\lambda^t)}{(1/\alpha) \cdot d^{-1}(\lambda^t)} \geq \frac{|N|}{h} \cdot d^{-1}(\lambda^t).$$

Thus, we can apply the proportionality degree with $\ell = d^{-1}(\lambda^t)$ and obtain an average satisfaction for V of

$$\mathrm{avg}_V(r_{\leq h}) \geq \min\left(\lambda^t, d(d^{-1}(\lambda^t))\right) = \lambda^t.$$

It follows that $\mathcal{R}$ satisfies κ-group representation for $\kappa(\alpha, \lambda^t) = h = \lceil 1/\alpha \cdot d^{-1}(\lambda^t) \rceil$. □

Plugging the proportionality degree from Theorem 14 into this lemma we obtain the desired bound on the group representation as mentioned in Theorem 13.

5.1.2 Dynamic seqPAV

For dynamic seqPAV we prove the following generalisation of Theorem 3 by Skowron et al. (2017), where the version of Skowron et al. corresponds to the special case where $X = ()$.

Theorem 16 *Dynamic seqPAV satisfies κ-group representation for*

$$\kappa(\alpha, \lambda) = \left\lceil \frac{2(\lambda + 1 + \mathrm{avg}_V(X))^2}{\alpha^2} \right\rceil.$$

Note that it is not immediately clear how to convert a bound on the group representation into a result on the proportionality degree (i.e., whether an inverse version of Lemma 15 is possible). Still, using the same approach as in the proof of Theorem 16, we are able to prove a bound on the proportionality degree of dynamic seqPAV.

Theorem 17 *Dynamic seqPAV satisfies an h-proportionality degree of*

$$g(\ell, h) = \ell \cdot \sqrt{\frac{1}{2h}} - \text{avg}_V(X) - 1.$$

Note that this is the first such bound in closed form for a seqPAV-variant. There are explicit bounds for small h for the non-dynamic version provided by Skowron (2021). The added generality of our closed form comes at the price of less accuracy when compared to those bounds. While Skowron (2021) shows that for $h = 20$ the proportionality degree of (non-dynamic) seqPAV is greater or equal to $0.7503 \cdot \ell - 1$ and for $h = 200$ it is still greater or equal to $0.694 \cdot \ell - 1$ our bound gives $0.1581 \cdot \ell - 1$ for $h = 20$ and $0.05 \cdot \ell - 1$ for $h = 200$ (when considering the non-dynamic case where $X = ()$).

5.1.3 Myopic rules and AV

AV does not perform any different in the dynamic ranking setting compared to the non-dynamic one. Thus, it satisfies the same bounds on group representation as those stated in Theorem 2 by Skowron et al. (2017). Since myopic seqPAV and myopic Phragmén both agree with AV in the case X = (), the same bounds hold for these two rules.

Proposition 18 *Myopic seqPAV and myopic Phragmén fail* κ*-group representation for*

$$\kappa(\alpha, \lambda) \leq \left\lceil \frac{\lambda \cdot \alpha}{2\alpha - 1} \right\rceil - 1,$$

and for all functions $\kappa(\alpha, \lambda)$ *if* $\alpha \leq \frac{u+1}{u+2}$*, where* $u = |\bigcup_{i \in V} A_i \cap X|$.

Proof The first negative result follows simply by noting that both myopic rules are equal to AV if $X = ()$ and referring to the negative result for AV provided by Skowron et al. (2017). We prove the second negative result by means of a counterexample which is again an adapted version of the one given by Skowron et al. (2017) for AV. Assume λ^t, u and a function $\kappa(\alpha, \lambda)$ are given, set $\alpha \leq \frac{u+1}{u+2}$ and $h = \kappa(\alpha, \lambda^t)$. Let $A = A_X \dot\cup A_V \dot\cup A_G$ with $|A_X| = u$ and $|A_V| = |A_G| = h$ be a set of candidates and $N = V \dot\cup G$ an electorate composed of the disjoint union of voter groups V and G with $|V| < \alpha|N|$. Let the profile be such that all voters in V approve of all candidates in A_X and A_V and all voters in G approve of all A_G and set $X = A_X$. Then, both myopic rules rank all $a \in A_G$ higher than each candidate in A_V and thus $\text{avg}_V(r_{\leq h}) = 0$. □

Since AV and both myopic rules do not allow for any bound on κ-group representation for groups of voters that are not already a majority of the electorate it follows directly from Lemma 15 that they do not allow any bound on the proportionality degree for those groups either.

Corollary 19 *AV does not satisfy any bound on the proportionality degree for groups of size* $\alpha \leq \frac{1}{2}$. *Myopic seqPAV and myopic Phragmén do not satisfy any bound on the proportionality degree for groups of size* $\alpha \leq \frac{u+1}{u+2}$, *where* $u = |\bigcup_{i \in V} A_i \cap X|$.

5.2 Proportionality of implemented candidates

In this section, we study worst-case bounds on the proportionality of the set X of implemented candidates. Clearly, no non-trivial bounds are obtainable without restricting the selection behavior of an adversarial DM. Therefore, we will make the following two assumptions throughout this section:

- **(A1)** The DM is depth-restricted and always implements a candidate from the top h positions of the ranking.
- **(A2)** Every candidate $c \in C$ has sufficiently many "clones,"[6] i.e., candidates c' with identical supporter set $N_{c'} = N_c$.

Assumptions (A1) and (A2) together ensure that the DM can be forced to implement a candidate approved by a voter, by populating the top h positions exclusively with such candidates. Arguably the most natural way to ensure (A2) is to assume that we are in the *party-approval* setting (Brill et al. 2024b), where candidates are interpreted as parties and can be selected arbitrarily often. In the motivating example of live Q&A platforms, party-approval preferences could result from assigning categories (or *tags*) to questions and eliciting participants' approval preferences over categories.[7]

Recall that X^{t+1} denotes the set containing the implemented candidates from the first t rounds. The following property is a natural adaption of the well-studied proportionality axiom *proportional justified representation (PJR)* (Sánchez-Fernández et al. 2017).

Definition 20 A dynamic ranking rule satisfies *proportional justified selection (PJS)* if the following holds for all $t, \ell \in \mathbb{N}$ and for all groups $V \subseteq N$ of voters:

$$\text{If } |V| \geq \frac{\ell}{t} \cdot |N| \text{ and } |\bigcap_{i \in V} A_i| \geq \ell, \text{ then } |X^{t+1} \cap \bigcup_{i \in V} A_i| \geq \ell.$$

A weaker version of this axiom is obtained by fixing $\ell = 1$; in analogy to a well-known notion due to Aziz et al. (2017), we refer to the resulting property as *justified selection*.

We prove the following theorem by interpreting the set X^{t+1} of implemented candidates as a committee.

[6] Given an upper bound T on the number of iterations, $h + T - 1$ clones suffice (as at least h clones will always remain in the ranking).

[7] Note that this is different from the attribute-based representation approach discussed in the introduction: Rather than categorizing *voters* by self-declared attributes, here we refer to a categorization of *candidates* (i.e., questions).

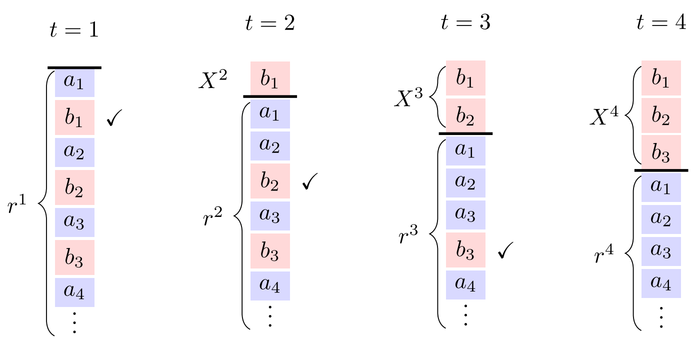

Fig. 3 Rankings discussed in Example 3

Theorem 21 *Under assumptions* (A1) *and* (A2), *myopic Phragmén satisfies PJS.*

Proof First observe that myopic Phragmén always ranks clones consecutively. Due to (A2), there are always at least h clones of each candidate, so that the first h position of each ranking $r^{t'}$ (where $t' \leq t$) will be occupied by a set of candidates that are all clones of each other. Due to (A1), the DM selects a candidate from this top-ranked clone set in each iteration. Now consider the set $X^{t+1} = \{x_1, \dots, x_t\}$ of implemented candidates. Since the assignment of debts under myopic Phragmén mimics the distribution of loads under sequential Phragmén, this set consists precisely of the first t candidates that sequential Phragmén selects on the same instance. Since sequential Phragmén satisfies PJR (Brill et al. 2024b), it follows that myopic Phragmén satisfies PJS. □

Analogously, we can translate a representation guarantee for seqPAV (Sánchez-Fernández et al. 2017) into a guarantee for myopic seqPAV.

Proposition 22 *Under assumptions* (A1) *and* (A2), *myopic seqPAV satisfies justified selection for* $t \leq 5$.

Similar positive results are not possible for the other rules. To see this, consider the following example, which is consistent with assumptions (A1) and (A2). The rankings discussed in this example are depicted in Fig. 3.

***Example* 3** Let $N = V_{\text{blue}} \cup V_{\text{red}}$ be the electorate consisting of two disjoint groups of voters of equal size, i.e., $V_{\text{blue}} \cap V_{\text{red}} = \emptyset$ and $|V_{\text{blue}}| = |V_{\text{red}}|$. Now assume that each voter in V_{blue} approves of all candidates in $\{a_1, a_2, \dots\}$ and each voter in V_{red} approves of all candidates in $\{b_1, b_2, \dots\}$. If we assume alphabetic tie-breaking between the parties, then both dynamic rules will in the first iteration output the ranking $r^1 = (a_1, b_1, a_2, b_2, \dots)$. If we set $h = 4$, then it is possible for an adversarial DM to implement 3 candidates supported by V_{red} before in the fourth iteration we have $r^4_{\leq 4} = \{a_1, a_2, a_3, a_4\}$. In particular, we have $X^3 = (b_1, b_2)$ and $X^4 = (b_1, b_2, b_3)$.

Proposition 23 *Dynamic seqPAV and dynamic Phragmén fail to satisfy justified selection*, *even under assumptions* (A1) *and* (A2) *and for* $t = 2$.

6 Experimental evaluation

In order to better understand the behavior of the dynamic ranking rules considered in this paper, we conducted computational experiments using randomly generated approval profiles. Since we were mainly interested in the proportional representation of groups of voters with similar preferences, we generated profiles according to two probabilistic models that lead to polarized electorates with easily identifiable groups. We measured (i) how the satisfaction of a voter group with the set of implemented candidates varies with the size of the group, and (ii) how the satisfaction of a voter group with the current ranking varies over time.

6.1 Setup

All of our profiles consist of 60 voters and 20 candidates, and the approval sets are generated according to two different models. We first describe how we generate the approval profiles randomly. Since we want to study the satisfaction of groups of voters with roughly similar approval preferences we somehow want to generate profiles with such voter groups. We used two different approaches.

- **Blurred parties model.** Here we assign each of the 60 voters to one of two parties. The size of the parties will vary over the experiments and we will concentrate on the satisfaction of one of the parties. Additionally, we will also associate half of the candidates to one of the parties and the other half to the other, such that each party has 10 candidates associated with them. Here, a party is a group of voters that have the same probability of approving a candidate. Now, for a voter we go over all candidates and say that the voter approves that candidate with probability 0.95, if it is a candidate of the voters party, and with probability 0.05, otherwise. This process is done independently for each voter and for each candidate. Thus, each voter in expectation approves of 95% of the candidates in their party.
- **Spatial model.** This is an adaption of the 4-Gaussian model that was described by Elkind et al. (2017) for the setting of linear preferences. We again group voters and candidates into parties, this time using 3 parties. Thus, two parties are associated with 7 candidates and one party is associated with 6 candidates. The number of voters in the parties again varies but we are always concerned with the satisfaction of the first party, $V \subseteq N$, of size $|V|$ and set the sizes of the other two parties to $\lceil 60 - |V|/2 \rceil$ and $\lfloor 60 - |V|/2 \rfloor$, respectively. In this model we now take a spatial approach using the Euclidean plane. Each of the three parties gets assigned a point—their *center*—that lies on the unit circle. To make the three points equidistant from each other we place them at 0, 120 and 240 degrees. Now the voters and candidates from each party get sampled as points on the Euclidean plane according to a 2-dimensional Gaussian (i.e., normal) distribution with standard deviation

0.4 around their party-center. We say a voter approves of a candidate if that candidate is at Euclidean distance at most 0.8. Note that if we assume that a voter gets assigned their party's center as point in the Euclidean plane, then by construction of the Gaussian distribution with standard distribution 0.4, this voter approves of roughly 95% of the candidates of their party in expectation.

To decide which candidates the DM selects for implementation, we also use a probabilistic approach. Since the ranking that is provided to the DM should somehow factor into the decision which candidate to select, we try to mimic a realistic behavior. To this end, we use an approximation of Google's *click-through rates (CTRs)*. These rates describe how likely it is for a user to click on the first, second, etc. entry in a Google search result. These rates get approximated experimentally and published by various companies. The specific values for the first 15 positions we use for our experiments are as follows.[8] In order to use them as a probabilistic vector, we normalized the values to sum to 1.

Position	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
CTR	32.5	17.6	11.4	8.1	6.1	4.4	3.5	3.1	2.6	2.4	1.0	0.8	0.7	0.6	0.4

We also ran our experiments with several other probability distributions, including harmonically or quadratically decreasing probabilities and probabilities inspired by the notion of *discounted cumulative gain* (Järvelin and Kekäläinen 2002). Since the results were very similar for all probability distributions, we only discuss the results for the click-through rates.

6.2 Results

We now describe our findings with respect to the two measures of satisfaction mentioned above. A graphical representation of the results can be found in Figs. 4 and 5. For each experiment, the figures show the average over 100 runs. Figure 4 shows an overview of the results of the experiments. Additionally, Fig. 5 shows the 10–90 percentile bands for all experiments and all rules (excluding the lowest 10% and the highest 10% of the values of all respective runs) as a measure of variance.

Satisfaction with implemented candidates. We measure the average satisfaction of voter group $V \subseteq N$ with X^{k+1}, where k is the number of candidates associated with that group (i.e., $k = 10$ for the blurred parties model and $k = 7$ for the spatial model). We plot this value against the relative size $\alpha = |V|/|N|$ of the group V. The graphs in the first row of Fig. 4 show that for both models, AV is not proportional: $\text{avg}_V(X^{k+1})$ starts out very low and only jumps up as soon as V becomes the biggest group (which happens at $\alpha = 1/2$ and $\alpha = 1/3$, respectively). In other words, AV underrepresents minorities and overrepresents majorities. The performance of the other four rules are indistinguishable, as all yield proportionally

8 These values are taken from http://www.wikiweb.com/google-ctr/, accessed July 21st, 2023.

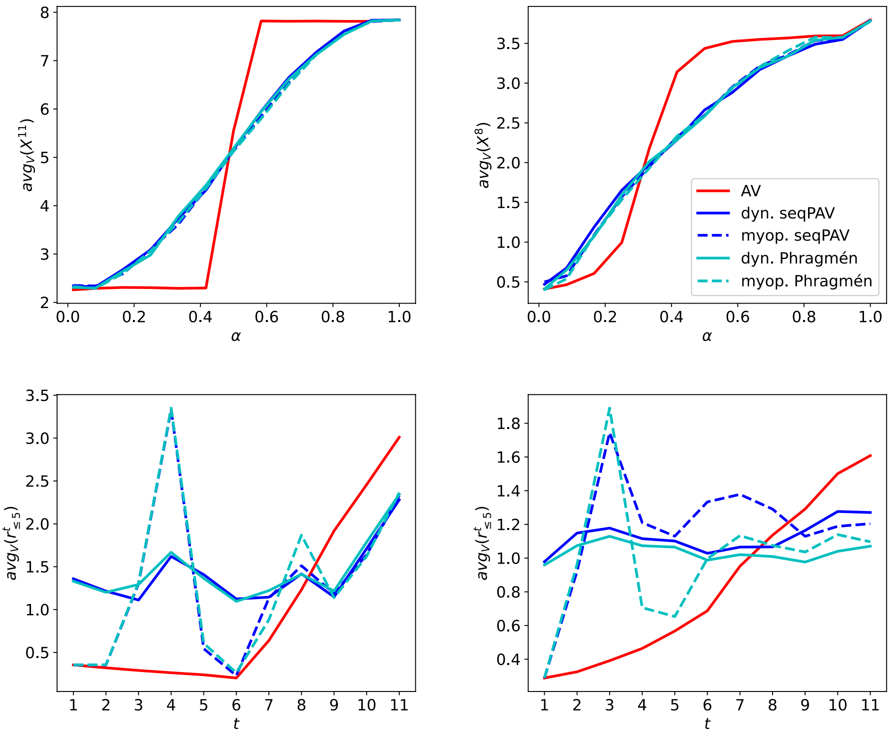

Fig. 4 Experimental results for the blurred parties model (left) and the spatial model (right). The graphs in the first row show the average satisfaction of V with the first k implemented candidates, for relative group size $\alpha \in [0, 1]$. The graphs in the second row show the average satisfaction of V with $r^t_{\leq 5}$, for $1 \leq t \leq 11$

increasing satisfaction values. Figure 5 shows that, especially for the blurred parties model, the spread is rather small for all rules.

Satisfaction with rankings. The graphs in the second row of Fig. 4 depict the average satisfaction of a group V of size $\alpha = 1/4$ with the first 5 candidates of the ranking over the first 11 iterations. Again, AV behaves poorly, as it gives satisfaction to V only once the larger groups have been satisfied. The satisfaction values under the two myopic rules jump heavily from one iteration to the next, as these rules tend to mainly represent one group of voters per iteration. This is also an explanation for the rather large spread (see Fig. 5). Depending on the (random) choice of the DM in the experiments, the group V is represented in different rounds for each run of the experiments, resulting in a large variance in the average satisfaction. On the other hand, the two dynamic rules keep the satisfaction of V relatively constant at around one fourth of the maximum possible satisfaction. These rules provide proportional representation in each single iteration, which is in line with the theoretical results in Sect. 5.1. The dynamic rules also behave differently from the myopic rules in terms of spread (see Fig. 5).

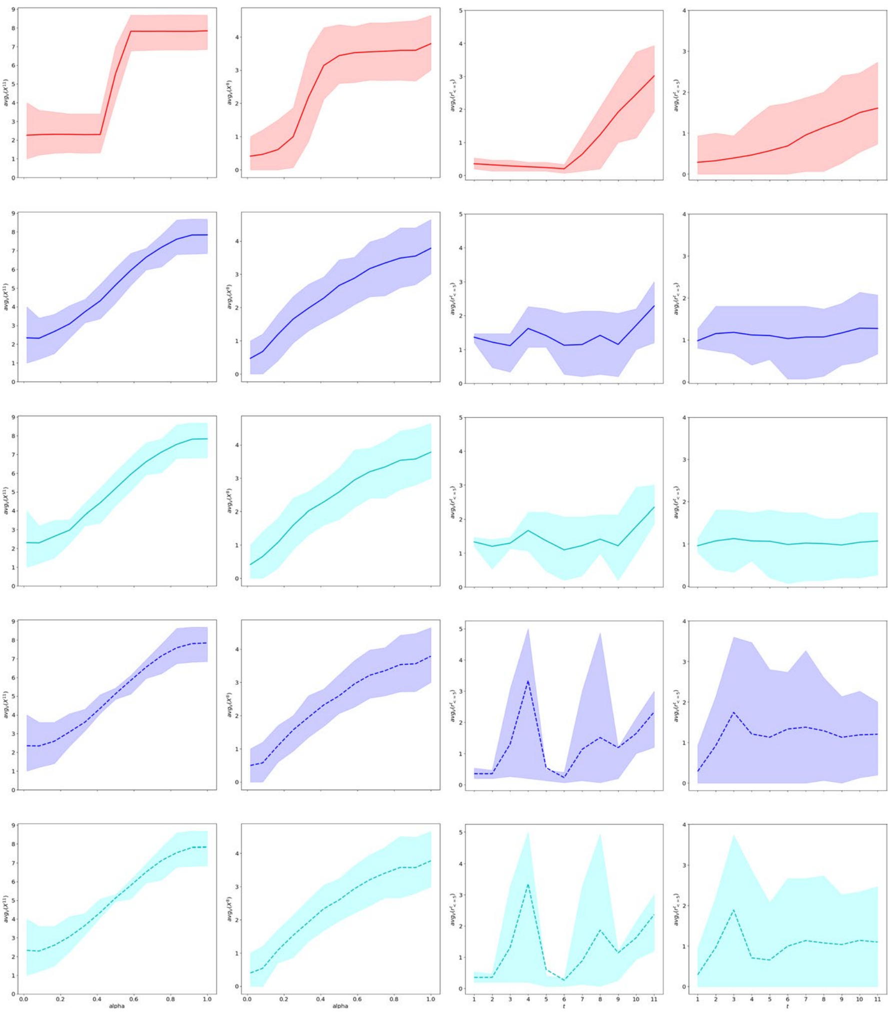

Fig. 5 The 10–90 percentile bands for the five rules over the four experiments. Rows correspond to rules and columns to experiments. The colors of the rules are as in Fig. 4: AV is red, dynamic and myopic seqPAV are dark blue (solid and dotted lines, respectively), and dynamic and myopic Phragmén are light blue (solid and dotted lines, respectively)

7 Conclusion

Motivated by the problem of how questions in a live Q&A session can be ranked in a more representative way, we have introduced dynamic ranking rules. We proposed two paradigms of dynamizing existing ranking rules: under the *dynamic paradigm*, we target proportional representation of voter interests at each individual time step; under the *myopic paradigm*, we try to make the set of implemented candidates as representative as possible. While the former approach lends more flexibility to the decision maker and guarantees a proportional exposure of candidates in each ranking, the latter approach is computationally slightly more efficient and yields stronger selection guarantees. Our experimental results illustrate the difference between the two approaches, and verify that both approaches lead to proportional results. In a practical comparison, the dynamic paradigm seems more reasonable in scenarios where the audience closely observes the ranking's evolution over time. In such scenarios, it is advantageous that the rankings exhibit relatively consistent positions between consecutive time steps, ensuring a smoother and more gradual evolution of the ranking. Conversely, the myopic paradigm might be more suitable for scenarios where there is a strong desire to encourage or even force the decision maker to adhere closely to the goal of proportionality, even if it leads to more pronounced changes in rankings between consecutive time steps.

The application of live Q&A platforms gives rise to some interesting extensions of our model. In realistic scenarios, neither the electorate nor the set of candidates is static, as people enter or leave the audience and new questions come up continuously. Moreover, participants can change their approval preferences throughout the event. Our approach can take these dynamic aspects into account in a straightforward manner: After each implementation, we can apply our ranking rules to the current set of not-yet-implemented candidates and to the current approval preferences—the only necessary information from previous iterations is the sequence of implemented candidates. Another important issue motivated by real-world applications is that participants often do not have the time or resources to thoroughly consider the whole set of candidates (i.e., questions). One way to mitigate this problem is to introduce categories (or tags) for questions and let participants submit preferences over these categories instead of individual questions (as discussed in Sect. 5.2).

The dynamic ranking rules proposed in this paper are applicable to a wide variety of sequential selection procedures in which proportional representation is desired and, at the same time, some flexibility on the part of the decision maker is necessary (e.g., think of human-in-the-loop decision support systems for hiring or budgeting decisions). Other applications of dynamic ranking rules include committee election scenarios in which some part of the committee is fixed (e.g., due to external constraints) and the remaining seats need to be filled in such a way that the committee as a whole is representative.

Appendix A: Algorithmic aspects of dynamic ranking rules

In this section, we study the dynamic ranking rules (introduced in Sect. 3) from an algorithmic perspective. For each of the rules, we provide a pseudocode formulation and give asymptotic bounds on the running time. We consider the two rules based on sequential PAV in Appendix A.1 and the two Phragmén variants in Appendix A.2.

All four rules take as input an approval profile A and a sequence of already implemented candidates X. For a given profile A, we denote the voters given implicitly through the profile by $N(A)$ and the candidates by $C(A)$. The rules output a ranking of all candidates $C(A) \setminus X$ that have not been implemented yet.

In the following, we let $n = |N(A)|$ denote the number of voters and $m = |C(A)|$ the number of candidates. Furthermore, for a set S we use the notation

$$(i \in S \mid \text{sorted } \nearrow \text{ by } f(i))$$

to denote the sequence in which the elements of S are ordered according to non-decreasing $f(i)$-value (and analogously $\searrow$ for non-increasing sequences). For example, $(i \in N \mid \text{sorted} \nearrow \text{ by } d_i)$ is the sequence that orders voters in N non-decreasingly according to d_i and $(c \in C \mid \text{sorted} \searrow \text{ by } \mathrm{mc}(c))$ is the sequence that orders candidates in C non-increasingly according to $\mathrm{mc}(c)$.

A.1 Algorithmic aspects of dynamic and myopic seqPAV

Dynamic seqPAV mimics the non-dynamic variant closely by computing the marginal contribution of a candidate according to all already ranked and already implemented candidates. The rule ranks candidates greedily one by one. In every round it selects a candidate with maximal marginal contribution and appends that candidate to the end of the ranking (see Algorithm 1).

Algorithm 1 Dynamic seqPAV

Input: approval profile A, sequence of implemented candidates X
Output: ranking $r \in \mathcal{L}(C(A) \setminus X)$
$C \leftarrow C(A) \setminus X$ // unranked candidates
$r \leftarrow ()$ // ranking of candidates
while $C \neq \emptyset$ **do**
    for all $c \in C$ **do**
        $\mathrm{mc}_{\mathrm{dyn}}(c) \leftarrow \mathrm{sc}(X \cup r \cup c) - \mathrm{sc}(X \cup r)$
            // compute marginal contribution
    end for
    choose $c \in \arg\max_{c \in C} \mathrm{mc}_{\mathrm{dyn}}(c)$
        // choose candidate with max. mc
    $r \leftarrow (r, c)$ // append that candidate to r
    $C \leftarrow C \setminus \{c\}$
end while
return r

The myopic version of this rule computes the marginal contribution of the candidates once upfront—only with respect to the already implemented candidates—and then simply ranks all candidates according to this score (see Algorithm 2).

Algorithm 2 Myopic seqPAV

Input: approval profile A, sequence of implemented candidates X
Output: ranking $r \in \mathcal{L}(C(A) \setminus X)$

1: $C \leftarrow C(A) \setminus X$ `// unranked candidates`
2: **for all** $c \in C$ **do**
3: $\quad \mathrm{mc}_{\mathrm{myopic}}(c) \leftarrow \mathrm{sc}(X \cup c) - \mathrm{sc}(X)$
`// compute marginal contribution`
4: **end for**
5: $r \leftarrow (c \in C \mid \text{sorted} \searrow \text{by } \mathrm{mc}_{\mathrm{myopic}}(c))$
`// rank candidates by` $\mathrm{mc}_{\mathrm{myopic}}(c)$ `in non-increasing order`
6: **return** r

Proposition 2 *Given an approval profile and a sequence of implemented candidates, the output of dynamic seqPAV can be computed in time $\mathcal{O}(m^3 n)$ and the output of myopic seqPAV can be computed in time $\mathcal{O}(m^2 n)$.*

Proof Termination of both algorithms is straightforward. Concerning the running time of dynamic seqPAV, consider Algorithm 1. First note that $\mathrm{mc}_{\mathrm{dyn}}(c)$ can be computed in time $m \cdot n$. The loops starting in Lines 3 and 4 of Algorithm 1 each have at most m iterations. Choosing a candidate with maximum marginal contribution (Line 7) can be done in an additional time of m. This, however, gets dominated by the running time of the loop in Line 4. Appending and deleting candidates in Lines 8 and 9 is possible in constant time. Thus, the overall running time is in $\mathcal{O}(m^3 n)$.

Concerning the running time of myopic seqPAV, consider Algorithm 2. First note that $\mathrm{mc}_{\mathrm{myopic}}(c)$ can be computed in time $m \cdot n$. The loop starting in Line 2 of Algorithm 2 runs for at most m iterations. Ranking (i.e., sorting) all candidates takes an additional time of $m \log(m)$ which is, however, dominated by the running time of the above loop. Thus, the overall running time is in $\mathcal{O}(m^2 n)$. □

A.2 Algorithmic aspects of dynamic and myopic Phragmén

We present a pseudocode formulation of dynamic Phragmén as Algorithm 3. This algorithm uses two subroutines that we describe in more detail afterwards. For a voter $i \in N(A)$, we denote i's credits with $¢_i$.

First, to compute the initial debts for dynamic Phragmén (and myopic Phragmén, which we cover later in this section), we use a subroutine called `compute_debts` (see Algorithm 4). This algorithm takes an approval profile A and a sequence of implemented candidates X as input and outputs the amount of debt each voter $i \in N(A)$ receives to accommodate the costs of the already implemented candidates in X. Dynamic Phragmén then interprets these debts as negative credits, setting $¢_i = -d_i$. Dynamic Phragmén constructs the output ranking iteratively. In order to find the next candidate to rank, the algorithm searches for the candidate that can be bought by (a subset of) its supporters at the earliest point in time. More precisely, for each unranked candidate c, the algorithm calculates the minimal time we have to wait until the supporters of c can buy c. Then, we rank the candidate with the smallest such time next and let its supporters pay for the candidate. The calculation of this minimal time is done by the subroutine `compute_buying_time`.

Algorithm 3 Dynamic Phragmén

Input: approval profile A, sequence of implemented candidates X
Output: ranking $r \in \mathcal{L}(C(A) \setminus X)$

$C \leftarrow C(A) \setminus X$ // unranked candidates
$N \leftarrow N(A)$ // voters
$r \leftarrow ()$ // ranking of candidates
$(d_i)_{i \in N} \leftarrow \texttt{compute\_debts}(A, X)$ // compute initial debts
for all $i \in N$ **do**
  $¢_i \leftarrow -d_i$ // starting credits
end for
while $C \neq \emptyset$ **do**
  for all $c \in C$ **do**
    $(t_c, V_c) \leftarrow \texttt{compute\_buying\_time}(A, c, (¢_i)_{i \in N})$
  end for
  choose $c \in \arg\min_{c \in C} t_c$ // choose candidate with min. t_c
  for all $i \in V_c$ **do**
    $¢_i \leftarrow 0$ // voters in V_c pay for candidate c
  end for
  for all $i \in N \setminus V_c$ **do**
    $¢_i \leftarrow ¢_i + t_c$ // voters outside V_c get t_c credits
  end for
  $r \leftarrow (r, c)$ // append candidate c to r
  $C \leftarrow C \setminus \{c\}$
end while
return r

Subroutine *compute_debts.* Algorithm 4 computes the initial debts for both Phragmén variants. The input of the algorithm is an approval profile A and a sequence of implemented candidates X; the algorithm computes, for each voter $i \in N(A)$, a non-negative real number d_i which is the amount of debt voter i receives to accommodate the costs of the already implemented candidates in X. Since every candidate has a cost of 1, we have $\sum_{i \in N(A)} d_i = |X|$. The algorithm iterates over X and for each candidate distributes the cost of 1 among all voters approving that candidate. More formally, let $x \in X$ be an already implemented candidate and let N_x be the supporters of x. We divide the cost of 1 for x among N_x in a way that minimizes the maximal debt across all voters in N_x. This is the same approach that is used in the load-balancing version of sequential Phragmén. Note that the difference between debt (or load) of two voters assigned in previous steps of the algorithm can be arbitrarily high.[9] (For example, if the first k implemented candidates in X are only supported by a single voter, this voter gets

[9] This is in contrast to the standard version of sequential Phragmén (Brill et al. 2024b).

assigned a debt of k before any other voter gets assigned any debt.) This may lead to a situation where, in order to minimize the maximal total debt across voters in N_x, we distribute the cost only over a subset of N_x. Thus, for each $x \in X$ we first have to determine the correct subset of N_x to distribute the debt to. We do this by ordering voters in N_x non-decreasingly by d_i and inspect any prefix of voters in this order. For every such prefix $N' \subseteq N_x$ we compute the debt voters in N' have after distributing the additional debt of 1 by

$$d_{\text{new}} = \frac{1 + \sum_{i \in N'} d_i}{|N'|}.$$

Let $v_{+1} \in N_x \backslash N'$ be that supporter of x not in N' with the next lowest debt $d_{v_{+1}}$. It is possible to lower d_{new} by adding v_{+1} to N' if and only if $d_{v_{+1}} < d_{\text{new}}$. The algorithm `compute_debts` does this computation for each $x \in X$ in the order given by the sequence X itself and then outputs the debts computed this way.

Algorithm 4 `compute_debts`

Input: approval profile A, sequence of implemented candidates X
Output: $(d_i)_{i \in N(A)}$, debts for all $i \in N(A)$

$N \leftarrow N(A)$ `// voters`
for all $i \in N$ **do**
  $d_i \leftarrow 0$ `// initial debt`
end for
for $x \in X$ **do**
  $\vec{N}_x \leftarrow (i \in N_x \mid \text{sorted} \nearrow \text{by } d_i)$
  `// supporters of` x `sorted non-decreasingly by` d_i
  for $j \leq |N_x|$ **do**
    $N' \leftarrow (\vec{N}_x)_{\leq j}$ `// first` j `voters in` N_x
    $d_{\text{new}} \leftarrow \frac{1+\sum_{i \in N'} d_i}{j}$ `// distribute debt among` N'
    if $j < |N_x|$ and $d_{\text{new}} \leq d_{j+1}$ **then** `// check debt of next`
      for all $i \leq j$ **do**
        $d_i \leftarrow d_{\text{new}}$ `// assign new debt`
      end for
      break `// break inner for-loop`
    end if
    if $j = |N_x|$ **then** `// all supporters share debt`
      for all $i \in N_x$ **do**
        $d_i \leftarrow d_{\text{new}}$ `// assign new debt`
      end for
    end if
  end for
end for
return $(d_i)_{i \in N(A)}$

Proposition 24 *Given an approval profile A and a sequence of implemented candidates X, the subroutine* `compute_debts` *computes the debts for all voters in $N(A)$ according to X in time $\mathcal{O}(mn^2)$.*

Proof Concerning termination of Algorithm 4, consider a candidate $x \in X$ and its set of supporters $N_x \subseteq N(A)$. The set N_x gets sorted into a sequence (non-decreasingly by d_i) in Line 6. In the loop starting in Line 7, each prefix of voters approving x is considered. For each such prefix N', the algorithm calculates the debt that each voter in N' would get if they together had to pay 1 credit for candidate x (Line 9). As described above, the algorithm then checks whether this debt can be decreased by adding the next candidate in $N_x \backslash N'$ to N' (Line 10). If not, the debt is distributed among voters in N'. Otherwise, the loop continues. The if-statement in Line 16 guarantees that in the end, if no proper subset of candidates $N' \subset N_x$ was singled out, all supporters of x share the debt.

Concerning running time, the loop starting in Line 5 runs for $|X| \leq m$ iterations. Sorting candidates in N_x can be done in time $n\log(n)$. The loop in Line 7 has at most n iterations and the calculation in Line 9 can be performed in $\mathcal{O}(n)$ operations. The assignment of debts in Lines 12 or 18 contributes another n operations, which are, however, dominated by the calculation in Line 9. Since the running time of the loop starting in Line 7 dominates the running time of the operation in Line 6, the overall running time of the algorithm is in $\mathcal{O}(mn^2)$. □

Subroutine `compute_buying_time`**.** As stated above, in each iteration of the ranking process, we calculate for each unranked candidate c the minimal time we have to wait until the supporters of c can buy c. Then, we rank the candidate with the smallest such time next and let the corresponding supporters pay for their candidate. Note that not necessarily all supporters of c have to pay, as it might be the case that some of them have too much debt and the rest of the supporters are able to raise 1 credit before those with a lot of debt can participate in the buying process. To compute that minimal time and the corresponding set of supporters of a candidate c, we use the subroutine `compute_buying_time`. This algorithm takes as input the approval profile A, a candidate $c \in C(A)$ and the current credit balance $(¢_i)_{i\in N}$ of all voters. The function outputs the minimal additional time until a subset of supporters of candidate c will have accumulated a positive budget of 1 credit. Formally, the function finds the minimal $t_c \in [0,1]$ such that there is a set of voters $V \subseteq N_c$ with

$$\sum_{i\in V}(¢_i + t_c) \geq 1.$$

Because of the minimality of t_c we have $¢_i + t_c \geq 0$ for all $i \in V$. Practically, t_c and the corresponding set of supporters can be computed as follows. Set $V = \{i \in N_c \mid ¢_i \geq 0\}$ and sort all remaining supporters $i \in N_c \backslash V$ by non-increasing budget (i.e., voters with less debt are sorted to the top). For each $k = 0, 1, \ldots, |N_c \backslash V|$ consider the set V_k containing voters in V and the first k voters in $N_c \backslash V$. That means $V_0 = V$ and for $k = 1, 2, 3, \ldots$ the set V_k will additionally contain the k voters in $N_c \backslash V$ with the fewest debt. For each k, compute

$$t_c^k = \begin{cases} 0, & \text{if } \sum_{i \in V_k} \text{¢}_i \geq 1 \\ \frac{1-\sum_{i \in V_k}}{\text{¢}_i} |V_k|, & \text{else.} \end{cases}$$

The function `compute_buying_time` then outputs the lowest t_c^k and the corresponding set of voters V_k, breaking ties by smaller k. We now consider the running time of this algorithm. Setting up the sets V and sorting $N_c \backslash V$ can be done in $\mathcal{O}(n \log(n))$. This gets dominated by the running time of the loop over all $k \leq |N_c \backslash V|$. This loop has at most n iterations and for each k, the calculation of t_c^k is possible in $\mathcal{O}(n)$. Thus `compute_buying_time` runs in time $\mathcal{O}(n^2)$. We have therefore proven the following result.

Proposition 25 *Given an approval profile A, a candidate $c \in C(A)$ and the credit balance of all voters $(\text{¢}_i)_{i \in N}$,* `compute_buying_time` *outputs t_c and V_c in time $\mathcal{O}(n^2)$, where t_c is the minimal additional time until a subset of voters $V_c \subseteq N_c$ have a joint budget of* 1 *credit.*

Next, we give a pseudocode formulation of myopic Phragmén (see Algorithm 5). Myopic Phragmén also uses the debts from `compute_debts`, but in a greedy way. Here, we first iterate over all remaining candidates and for each $c \in C(A) \backslash X$ do the following. We append c to the sequence X, obtaining the sequence X_c, and then compute the debts for all voters according to X_c. We then rank those candidates highest for which the maximal voter debt is lowest. This, in a way, checks which candidate is most suitable to be implemented next. Formally, for each $c \in C(A) \backslash X$, we first sort the debts $(d_i^c)_{i \in N}$ of all voters according to X_c non-increasingly. Then, we rank the candidates $c \in C(A) \backslash X$ by comparing the sorted vectors $(\vec{d}_i^c)_{i \in N}$ lexicographically, ranking lexicographically smaller candidates (i.e., candidates that induce a lower maximum debt) higher.

Algorithm 5 Myopic Phragmén

Input: approval profile A, sequence of implemented candidates X
Output: ranking $r \in \mathcal{L}(C(A) \setminus X)$

$C \leftarrow C(A) \setminus X$ `// unranked candidates`
$N \leftarrow N(A)$ `// voters`
for all $c \in C$ **do**
  $X_c \leftarrow (X, c)$ `// append c to X`
  $(d_i^c)_{i \in N} \leftarrow$ `compute_debts`(A, X_c) `// compute debts w.r.t. X_c`
  $(\vec{d}_i^c)_{i \in N} \leftarrow ((d_i^c)_{i \in N} \mid \text{sorted} \searrow)$ `// sort debts non-increasingly`
end for
$r \leftarrow (c \in C \mid \text{sorted} \nearrow \text{lexicographically by } (\vec{d}_i^c)_{i \in N})$
`// rank candidates w.r.t. max. incurred debt`
return r

We are now ready to prove the bounds on the asymptotic runtime of both Phragmén variants.

Proposition 3 *Given an approval profile and a sequence of implemented candidates, the output of dynamic Phragmén can be computed in time $\mathcal{O}(m^2n^2)$ and the output of myopic Phragmén can be computed in time $\mathcal{O}(mn^2)$.*

Proof Termination of both algorithms is straightforward. Regarding the running time of dynamic Phragmén, consider Algorithm 3. We begin with the loops starting in Lines 8 and 9 . Both loops have at most m iterations. By Proposition 25 we know that computing the buying time in Line 10 can be done in $\mathcal{O}(n^2)$. This (in conjunction with the loop in Line 9) dominates the running time of choosing a candidate with minimum t_c (Line 12) and the loop in Line 13. Further, also the running time of the subroutine `compute_debts` of $\mathcal{O}(mn^2)$ is dominated by this. Thus, the overall running time is in $\mathcal{O}(m^2n^2)$.

Regarding the running time of myopic Phragmén, first consider Algorithm 5. The loop in Line 3 runs for at most m iterations. Line 4 takes constant time and Line 5 can be performed in $\mathcal{O}(mn^2)$ operations, as shown in Proposition 24. Sorting the debts with respect to X_c (Line 6) needs $n\log(n)$ operations, which is dominated by the computation of the debts. Lastly, ranking all candidates lexicographically with respect to $(d_i^c)_{i\in N}$ can be done in $\mathcal{O}(nm\log(m))$ operations. This is, however, dominated by the loop starting in Line 3. Thus the overall running time of this formulation of myopic Phragmén is in $\mathcal{O}(m^2n^2)$.

Note that the asymptotic running time for computing myopic Phragmén can be improved based on the following observation. For two distinct candidates $c, c' \in C(A)$, the sequences of candidates X_c and $X_{c'}$ only differ in the last entry. This is because c and c' both get appended to the same sequence X. When computing the debts with respect to X_c in Line 5 of Algorithm 5, the computation of the debts for all candidates in $X \subseteq X_c$ is the same, independent of the choice of candidate c. Thus, it is possible to compute the debts for all voters according to X first, before entering the loop in Line 3, by calling `compute_debts`(A, X) once, and reuse these debts for each candidate $c \in C$ in the loop. To compute the debts with respect to X_c in Line 5, only one more iteration of the calculations of `compute_debts` is needed (i.e., Lines 5 to 15 in Algorithm 4). This is possible in time $\mathcal{O}(n^2)$, bringing the overall running time of myopic Phragmén down to $\mathcal{O}(mn^2)$.[10] □

Appendix B: Omitted proofs

B.1 Weak group monotonicity of dynamic SeqPAV

Proposition 9 *Consider the depth-restricted setting for some $h \geq 3$. Then, dynamic seqPAV fails to satisfy weak (h, α)-monotonicity for all $\alpha < \frac{2}{4+h}$.*

[10] To be able to use `compute_debts` as is, we refrained from formulating this slightly faster version in Algorithm 5.

Proof Consider the following profile with 177 voters and 5 candidates.

$$4 \times \{a\},\ 27 \times \{a,b\},\ 27 \times \{b\},\ 30 \times \{c\},$$
$$9 \times \{c,d\}, 9 \times \{d\},\ 36 \times \{a,d\},\ 35 \times \{e\}.$$

The rule outputs the ranking $r^1 = (a,b,c,e,d)$ in the first iteration. If we now assume that the DM implements candidate $x_1 = b$ then dynamic seqPAV outputs $r^2 = (d,a,e,c)$. Now consider the group of voters V that consists of the 39 supporters of c (i.e., the 30 voters approving only c and the 9 voters approving c and d). For $h = 3$ we have $\text{avg}_V(r^1_{\leq h}) = 1$ but $\text{avg}_V(r^2_{\leq h}) = \frac{9}{39}$ which is a violation of the above axiom. We can see that this again holds for larger h by introducing new candidates with 35 supporters each (i.e., by "copying" candidate e and its supporters).

Similar to what we did in the proof of Proposition 6, we can increase the relative size of V without changing the rankings dynamic seqPAV outputs. For that consider the following adapted profile, where $j = 2 \cdot y$ for some $y \in \mathbb{N}$.

$$4 \times \{a\},\ (2j+27) \times \{a,b\},\ 27 \times \{b\},\ 30 \times \{c\},$$
$$(j+9) \times \{c,d\},\ 9 \times \{d\},\ 36 \times \{a,d\},\ (j/2+35) \times \{e\}.$$

For $j \to \infty$ we have $\frac{|V|}{|N|} \to \frac{2}{7}$. Combining this with the "copying" of candidate e and its $\frac{j}{2} + 35$ supporters we obtain an example where weak (h,α)-monotonicity is violated for every $h \geq 3$ by a group of size nearly $\frac{2}{4+h}$. □

B.2 Weak group monotonicity of dynamic Phragmén

Proposition 10 *Consider the depth-restricted setting for some $h \geq 3$. Then, dynamic Phragmén fails to satisfy weak (h,α)-monotonicity for all $\alpha < \frac{2}{5+h}$.*

Proof We use a similar example as above where we only add an additional copy of candidate e and its 35 supporters.

$$4 \times \{a\},\ 27 \times \{a,b\},\ 27 \times \{b\},\ 30 \times \{c\},$$
$$9 \times \{c,d\},\ 9 \times \{d\},\ 36 \times \{a,d\},\ 35 \times \{e_1\},\ 35 \times \{e_2\}.$$

In the first iteration dynamic Phragmén outputs the ranking $r^1 = (a,c,b,e_1,e_2,d)$. If we again assume that the DM implements candidate $x_1 = b$ then it outputs $r^2 = (d,e_1,e_2,c,a)$. We again consider the group of voters V that consists of the 39 supporters of c. For $h = 3$ we have $\text{avg}_V(r^1_{\leq h}) = 1$ but $\text{avg}_V(r^2_{\leq h}) = \frac{9}{39}$ which is a violation of the above axiom. This again holds for larger h via an introduction of new candidates with 35 supporters each (i.e., "copies" of candidate e_1 and its supporters). To increase the relative size of the voter group V consider the following adaption of the instance for an even natural number j.

$$4 \times \{a\},\ (2j+27) \times \{a,b\},\ 27 \times \{b\},\ 30 \times \{c\},$$
$$(j+9) \times \{c,d\},\ 9 \times \{d\},\ 36 \times \{a,d\},\ (j/2+35) \times \{e_1\},\ (j/2+35) \times \{e_2\}.$$

By checking the inequalities arising from this example in the same manner as described earlier for the stronger axiom we can verify that this example works for all $j \to \infty$. Again combining this with the idea of copying candidate e_1 and its $\frac{j}{2} + 35$ supporters we obtain an example where weak (h, α)-monotonicity is violated by dynamic Phragmén for every $h \geq 3$ by a group of size nearly $\frac{2}{5+h}$. □

B.3 Proportionality degree of dynamic Phragmén

Recall that d_i denotes the starting debt of agent i in dynamic Phragmén. Let $d_{\text{avg}} = \frac{1}{|V|} \sum_{i \in V} d_i$ be the average starting debt of agents in V.

Theorem 14 *Dynamic Phragmén satisfies proportionality degree of*

$$d(\ell) \geq \frac{\ell - 1}{2} - \frac{u}{2} - \frac{s \cdot |V|}{4},$$

where $u = |\bigcup_{i \in V} A_i \cap X|$ *and* $s = \sum_{i \in V} (d_i - d_{\text{avg}})^2$.

Proof Our proof is an adaptation of the proof of Theorem 2 by Skowron (2021), henceforth referred to as Skowron's proof. Let a depth restriction $h \in \mathbb{N}$ and an iteration $t \in \mathbb{N}$ be given. We set $g(\ell, h) = \frac{\ell - 1}{2} - \frac{u}{2} - \frac{s \cdot |V|}{4}$. To make the prove more consistent with Skowron's proof, assume w.l.o.g. that a candidate costs n instead of 1 credit. (This scaling of the cost does not change the workings of the rule.) Towards a contradiction, we assume that there is a set of candidates $r^t_{\leq h}$ that is ranked in the first h positions of the ranking produced by dynamic Phragmén at iteration t, and a group V of voters that is ℓ-large w.r.t. h but has an average satisfaction of $\text{avg}_V(r^t_{\leq h}) < \min(\lambda^t, g(\ell, h))$. Thus, there exists at least one candidate $c \in \bigcap_{i \in V} A_i \setminus (X \cup r_{\leq h})$, i.e., a candidate that is neither ranked nor implemented but approved by all voters in V. We will now investigate the ranking process of dynamic Phragmén in more detail. In order to do that, we imagine the process of buying candidates and waiting for credits for the voters as a time-dependent process. (Note that this process reflects the ranking of r^t for a fixed iteration $t \in \mathbb{N}$; we denote the time elapsing during this process by θ.) For each point in time $\theta > 0$ of the ranking process at iteration t, we define $p_i(\theta)$ to be the amount of credit voter $i \in V$ possesses at that moment, and $p_{\text{avg}}(\theta) = \frac{1}{|V|} \sum_{i \in V} p_i(\theta)$. With that we can define the potential function

$$\phi(\theta) = \sum_{i \in V} (p_i(\theta) - p_{\text{avg}}(\theta))^2.$$

In contrast to Skowron's proof, in our setting might start with negative money, i.e., debt assigned to them by the dynamic Phragmén rule because of implemented candidates in X. This means that at time $\theta = 0$ we might have $p_i(\theta) < 0$ for some voters and thus the potential function at time $\theta = 0$ is not 0 but might be some positive real number $\bar{s} = \phi(0) = \sum_{i \in V}(p_i(0) - p_{\text{avg}}(0))^2 = n^2 \cdot s$. (Here, $\bar{s}$ corresponds to $s = \sum_{i \in V}(d_i - d_{\text{avg}})^2$ scaled according to the new costs of n credits per candidate.)

The proof now proceeds as Skowron's proof. After h time units in the ranking process, at most $h \cdot n$ credits can be amassed by all voters and the procedure cannot stop before that point in time. Until then, the voters in V have at most $|V| \cdot h$ credits. Now fix a point in time θ at which a candidate supported by some voters in V is bought and a voter $j \in V$ that pays for this candidate. That means that the average $p_{\text{avg}}(\theta)$ decreases by $\frac{p_j(\theta)}{|V|}$. We denote by Δ_ϕ the change of the potential function due to that one voter paying for the newly bought candidate. The calculation of Δ_ϕ can be done in the same way as in Skowron's proof, as it does not depend on the starting value of ϕ. For a point in time θ in which a candidate is bought, we get the following.

$$\begin{aligned}
\Delta_\phi &= \sum_{i \in V, j \neq i} \left(p_i(\theta) - \left(p_{\text{avg}}(\theta) \frac{p_j(\theta)}{|V|} \right) \right)^2 \\
&\quad + \left(0 - \left(p_{\text{avg}}(\theta) \frac{p_j(\theta)}{|V|} \right) \right)^2 - \sum_{i \in V} (p_i(\theta) - p_{\text{avg}}(\theta))^2 \\
&= \sum_{i \in V} \left(p_i(\theta) - \left(p_{\text{avg}} - \frac{p_j(\theta)}{|V|} \right) \right)^2 \\
&\quad + \left(p_{\text{avg}} - \frac{p_j(\theta)}{|V|} \right)^2 - \left(p_j(\theta) - \left(p_{\text{avg}} - \frac{p_j(\theta)}{|V|} \right) \right)^2 \\
&\quad - \sum_{i \in V} (p_i(\theta) - p_{\text{avg}}(\theta))^2 \\
&= \sum_{i \in V} \frac{p_j(\theta)}{|V|} \left(2 \cdot p_i(\theta) - 2 \cdot p_{\text{avg}}(\theta) + \frac{p_j(\theta)}{|V|} \right)^2 \\
&\quad - p_j(\theta)^2 + 2 \cdot p_j(\theta) \cdot \left(p_{\text{avg}}(\theta) - \frac{p_j(\theta)}{|V|} \right),
\end{aligned}$$

where in the last step we used the binomial formulas. By definition we have

$$\sum_{i \in V} (2 \cdot p_i(\theta) - 2 \cdot p_{\text{avg}}(\theta)) = 0$$

which lets us conclude

$$\begin{aligned}
\Delta_\phi &= \frac{p_j(\theta)^2}{|V|} + p_j(\theta) \cdot \left(2 \cdot p_{\text{avg}}(\theta) - \frac{p_j(\theta)}{|V|} - p_j(\theta) \right) \\
&= p_j(\theta) \cdot \left(2 \cdot p_{\text{avg}}(\theta) - \frac{p_j(\theta)(|V| + 1)}{|V|} \right).
\end{aligned}$$

Just as in Skowron's proof, we can observe that at each time θ we have $p_{\text{avg}}(\theta) \leq \frac{n}{|V|} \leq \frac{h}{\ell}$. This is because otherwise voters in V would have more than n units of credits left and would have been able to buy a candidate they commonly approve of at an earlier time of the ranking process. Using that fact and setting $y_{\theta,j} = p_j(\theta) - \frac{2|V|}{|V|+1} \cdot \frac{h}{\ell}$ we obtain

$$\Delta_\phi \leq p_j(\theta) \cdot \left(\frac{2h}{\ell} - \frac{y_{\theta,j}(|V|+1)}{|V|} - \frac{2h}{\ell} \right) = -y_{\theta,j}^2 \frac{|V|+1}{|V|} - 2y_{\theta,j}\frac{h}{\ell} \leq -2y_{\theta,j}\frac{h}{\ell}.$$

Following Skowron's proof again, if $x_{\theta,j} > 0$, then ϕ decreases by at least $2|y_{\theta,j}| \cdot \frac{h}{\ell}$ and if $y_{\theta,j} \leq 0$, then ϕ increases by at most $2|y_{\theta,j}| \cdot \frac{h}{\ell}$. Since the potential value is always non-negative and starts at s, the net-change $\sum \Delta_\phi$ has to be greater or equal to $-\bar{s}$, i.e.,

$$-\bar{s} \leq \sum \Delta_\phi \leq \sum_{(\theta,j)\in\mathbb{R}\times V : j \text{ pays at time } \theta} -2y_{\theta,j} \cdot \frac{h}{\ell}.$$

Let $z = |\{(\theta, j) \in \mathbb{N} \times V : j \text{ pays at time } \theta\}|$ be the number of single payments the voters in V issued during the whole procedure. This is less than or equal to the total satisfaction of the group V, i.e.

$$z \leq \sum_{i\in V} |r_{\leq h} \cap A_i|.$$

Rearranging the terms of the bound on $\bar{s}$ above and plugging in the definitions of $y_{\theta,j}$ and z yields

$$\frac{\bar{s}}{2} \cdot \frac{\ell}{h} \geq \sum_{(\theta,j)} p_j(\theta) - \frac{2|V|}{|V|+1} \cdot \frac{h}{\ell} = \sum_{(\theta,j)} p_j(\theta) - z \cdot \frac{2|V|}{|V|+1} \cdot \frac{h}{\ell},$$

where $\sum_{(\theta,j)} p_j(\theta)$ is the total amount of credits voters in V spend for candidates they approved. Our goal now is to lower bound z and with that to lower bound the average satisfaction of voters in V.

For this, first note that we know that $\sum_{(\theta,j)} p_j(\theta) \geq |V| \cdot (h + |X|) - n - \sum_{i\in V} d_i$. Plugging this in the above equation and rearranging terms we obtain

$$\begin{aligned} z &\geq \frac{|V|+1}{2|V|} \cdot \frac{\ell}{h} \cdot \left(|V|(h + |X|) - n - \frac{\bar{s} \cdot \ell}{2h} - \sum_{i\in V} d_i \right) \\ &\geq \frac{1}{2} \cdot \frac{\ell}{h} \cdot \left(|V|(h + |X|) - n - \frac{\bar{s} \cdot \ell}{2h} - \sum_{i\in V} d_i \right) \\ &\geq \frac{1}{2}\left(|V|(\ell - 1) + \frac{|V| \cdot |X| \cdot \ell}{h} - \frac{\bar{s} \cdot \ell^2}{2h^2} - \frac{\ell}{h} \cdot \sum_{i\in V} d_i \right), \end{aligned}$$

where in the last step we used the fact that $n \cdot \frac{\ell}{h} \leq |V|$. We can now use this to get the desired lower bound on the average satisfaction of the voter group V.

$$\frac{1}{|V|}\sum_{i\in V}|r_{\leq h}\cap A_i| \geq \frac{1}{|V|}\cdot z \geq \frac{\ell-1}{2}+\frac{|X|\cdot\ell}{2h}-\frac{1}{2n}\cdot\sum_{i\in V}d_i-\frac{\bar{s}\cdot\ell}{4n\cdot h}$$
$$\geq \frac{\ell-1}{2}-\frac{u}{2}-\frac{\bar{s}\cdot|V|}{4n^2}=\frac{\ell-1}{2}-\frac{u}{2}-\frac{s\cdot|V|}{4},$$

where we again use $n\cdot\frac{\ell}{h}\leq|V|$ and the fact that $u=|\bigcup_{i\in V}A_i\cap X|\geq\frac{1}{n}\sum_{i\in V}d_i$. This contradicts our assumption that $\text{avg}_V(r^t_{\leq h})<g(\ell,h)$ and concludes the proof. □

B.4 Group representation of dynamic SeqPAV

Theorem 16 *Dynamic seqPAV satisfies κ-group representation for*

$$\kappa(\alpha,\lambda)=\left\lceil\frac{2(\lambda+1+\text{avg}_V(X))^2}{\alpha^2}\right\rceil.$$

Proof Fix some $\alpha\in(0,1]$, $\lambda\in\mathbb{N}$ and profile such that in some iteration $t\in\mathbb{N}$, there exists a group of voters $V\subseteq N$ with $|V|\geq\alpha\cdot n$ and $\lambda^t(V)\geq\lambda$. Let r be the ranking dynamic seqPAV outputs in iteration t given the already implemented candidates X. Set $h=\left\lceil\frac{2(\lambda+\text{avg}_V(X)+1)^2}{\alpha^2}\right\rceil$. Towards a contradiction assume that $\text{avg}_V(r_{\leq h})<\lambda$. Now set

$$z=|V|\cdot(\text{avg}_V(r_{\leq h})+\text{avg}_V(X))<|V|\cdot(\lambda+\text{avg}_V(X)).$$

In every step $k\in[h]$ of the ranking procedure of dynamic seqPAV (in iteration t), there exists at least one candidate $c\in\bigcap_{i\in V}A_i\setminus(X\cup r_{\leq h})$, i.e., a candidate that is neither ranked nor implemented but approved by all voters in V. We now consider a step $k\in[h]$ of the ranking process of dynamic seqPAV. Define

$$a_i(k)=|A_i\cap r_{\leq h}|+|A_i\cap X| \quad\text{and}\quad T(k)=\sum_{i\in N}\frac{1}{a_i(k)+1}.$$

Then, $a_i(k)\leq a_i(k+1)$ and $T(k)\geq T(k+1)$ for all $k\in[h]$ with $n\geq T(1)\geq T(2)\geq\cdots\geq T(h+1)\geq 0$. Further, for each $k\in[h]$ it holds

$$\sum_{i\in V}a_i(k)=\sum_{i\in V}|A_i\cap r_{\leq k}|+|A_i\cap X|\leq\sum_{i\in V}|A_i\cap r_{\leq h}|+|A_i\cap X|=z,$$

and thus it holds that

$$\frac{z+|V|}{|V|}\geq\frac{\sum_{i\in V}a_i(k)+|V|}{|V|}=\frac{\sum_{i\in V}(a_i(k)+1)}{|V|}\geq\frac{|V|}{\sum_{i\in V}\frac{1}{a_i(k)+1}},$$

where in the last step we used the inequality of arithmetic and harmonic means. Taking the inverse of this yields

$$\sum_{i\in V}\frac{1}{a_i(k)+1}\geq\frac{|V|^2}{|V|+z}>\frac{|V|^2}{|V|+|V|\cdot(\lambda+\mathrm{avg}_V(X))}=\frac{|V|}{\lambda+\mathrm{avg}_V(X)+1}.$$

Now let V' be the group of voters supporting candidate c' that got ranked in round k instead of c. Since dynamic seqPAV favored c' over c we know that

$$\sum_{i\in V'}\frac{1}{a_i(k)+1}\geq\sum_{i\in V}\frac{1}{a_i(k)+1}>\frac{|V|}{\lambda+\mathrm{avg}_V(X)+1}. \tag{1}$$

We now obtain

$$\begin{aligned}T(k)-T(k+1)&=\sum_{i\in V'}\left(\frac{1}{a_i(k)+1}-\frac{1}{a_i(k)+2}\right)\\&=\sum_{i\in V'}\left(\frac{1}{(a_i(k)+1)(a_i(k)+2)}\right)\\&\geq\sum_{i\in V'}\left(\frac{1}{2(a_i(k)+1)^2}\right).\end{aligned}$$

Using the Cauchy–Schwarz inequality, we can bound this in the following way.

$$\begin{aligned}\sum_{i\in V'}\left(\frac{1}{2(a_i(k)+1)^2}\right)&\geq\frac{1}{2|V'|}\left(\sum_{i\in V'}\frac{1}{a_i(k)+1}\right)^2\\&>\frac{1}{2n}\left(\frac{|V|}{\lambda+\mathrm{avg}_V(X)+1}\right)^2\\&\geq\frac{\alpha^2\cdot n}{2(\lambda+\mathrm{avg}_V(X)+1)^2},\end{aligned}$$

where in the second step we used Eq. (1). This holds for each $k\in[h+1]$ and thus we have

$$T(1)-T(h+1)=\sum_{k\in[h]}T(k)-T(k+1)>h\cdot\frac{\alpha^2\cdot n}{2(\lambda+\mathrm{avg}_V(X)+1)^2}\geq n.$$

This yields $T(k+1)<T(1)-n\leq 0$, which is a contradiction. □

B.5 Proportionality degree of dynamic SeqPAV

Theorem 17 *Dynamic seqPAV satisfies an h-proportionality degree of*

$$g(\ell,h)=\ell\cdot\sqrt{\frac{1}{2h}}-\mathrm{avg}_V(X)-1.$$

Proof Fix some $h \leq |C|$ and let r be the ranking dynamic seqPAV outputs for a given profile and set of already implemented candidates X. Further, let V be a group of voters and $z = \ell \cdot \sqrt{\frac{1}{2h}} - \text{avg}_V(X) - 1$. Towards a contradiction assume that V is ℓ-large w.r.t. h but $\text{avg}_V(r_{\leq h}) < \min\left(|\bigcap_{i\in V} A_i \setminus X|, z\right)$. Thus there exists at least one candidate $c \in \bigcap_{i\in V} A_i \setminus (X \cup r_{\leq h})$, i.e., a candidate that is neither ranked nor implemented but approved by all voters in V. We now consider the steps of the ranking process of dynamic seqPAV. For $k \in [h]$, we again define

$$a_i(k) = |A_i \cap r_{\leq h}| + |A_i \cap X| \quad \text{and} \quad T(k) = \sum_{i\in N} \frac{1}{a_i(k) + 1}.$$

Then, $a_i(k) \leq a_i(k+1)$ and $T(k) \geq T(k+1)$ for all $k \in [h]$ with $n \geq T(1) \geq T(2) \geq \cdots \geq T(h+1) \geq 0$. Further, for each $k \in [h+1]$, the inequality of arithmetic and harmonic means

$$\sum_{i\in V} \frac{1}{a_i(k) + 1} \geq |V|^2 \frac{1}{\sum_{i\in V}(a_i(k) + 1)}$$

and

$$\begin{aligned}
\frac{1}{|V|^2} \sum_{i\in V} a_i(k) + 1 &= \frac{1}{|V|^2} \sum_{i\in V} \left(|A_i \cap r_{\leq h}| + |A_i \cap X| + 1\right) \\
&= \frac{1}{|V|^2} \sum_{i\in V} (|A_i \cap X| + 1) + \frac{1}{|V|} \text{avg}_V(r_{\leq h}) \\
&< \frac{1}{|V|^2} \sum_{i\in V} (|A_i \cap X| + 1) \\
&\quad + \frac{1}{|V|}\left(\ell \cdot \sqrt{\frac{1}{2h}} - \frac{1}{|V|} \cdot \sum_{i\in V} (|A_i \cap X| + 1)\right) \\
&= \frac{1}{|V|} \cdot \frac{\ell}{h} \cdot \sqrt{\frac{h}{2}} = \frac{1}{n} \cdot \sqrt{\frac{h}{2}},
\end{aligned}$$

where we use that, by construction, $\text{avg}_V(r_{\leq h}) < z = \ell \cdot \sqrt{\frac{1}{2h}} - \text{avg}_V(X) - 1$. Plugging the second inequality into the first, we obtain

$$\sum_{i\in V} \frac{1}{a_i(k) + 1} > n \cdot \sqrt{\frac{2}{h}}.$$

Again, let V' be the group of voters supporting candidate c' that got ranked in round k instead of c. Since dynamic seqPAV favored c' over c we know that

$$\sum_{i\in V'} \frac{1}{a_i(k) + 1} \geq \sum_{i\in V} \frac{1}{a_i(k) + 1} > n \cdot \sqrt{\frac{2}{h}}. \tag{2}$$

As in the previous proof, we obtain using the Cauchy–Schwarz inequality

$$T(k) - T(k+1) \geq \frac{1}{2|V'|}\left(\sum_{i \in V'} \frac{1}{a_i(k)+1}\right)^2 > \frac{1}{2n} \cdot \left(n \cdot \sqrt{\frac{2}{h}}\right)^2 = \frac{n}{h},$$

where in the second step we used Eq. (2). This holds for each $k \in [h+1]$ and thus we have

$$T(1) - T(h+1) = \sum_{k \in [h]} T(k) - T(k+1) > h \cdot \frac{n}{h} = n.$$

This yields $T(k+1) < T(1) - n \leq 0$, which is a contradiction. □

Acknowledgements Preliminary versions of this paper have been presented at the *30th International Joint Conference on Artificial Intelligence (IJCAI 2021)* (Israel and Brill 2021) and at the *8th International Workshop on Computational Social Choice (COMSOC 2021)*. We would like to thank reviewers at *IJCAI*, *COMSOC*, and *Social Choice and Welfare* for their insightful comments. This work was partially supported by the Deutsche Forschungsgemeinschaft under grant BR 4744/2-1. We thank Paula Blechschmidt, Cristina Cornelio, Benny Kimelfeld, Phokion Kolaitis, and Julia Stoyanovich for helpful comments and discussions.

Funding Open Access funding enabled and organized by Projekt DEAL.

Data availability The code used to generate the data and to perform the experiments can be accessed via https://git.tu-berlin.de/jonas.israel/dpr.

References

Aziz H, Brill M, Conitzer V, Elkind E, Freeman R, Walsh T (2017) Justified representation in approval-based committee voting. Soc Choice Welf 48(2):461–485

Boutilier C, Procaccia A (2012) A dynamic rationalization of distance rationalizability. In: Proceedings of the 26th AAAI Conference on Artificial Intelligence (AAAI). AAAI Press, pp 1278–1284

Brill M, Peters J (2023) Robust and verifiable proportionality axioms for multiwinner voting. In: Proceedings of the 24th ACM Conference on Economics and Computation (EC), page 301. ACM. Full version: arXiv:2302.01989

Brill M, Freeman R, Janson S, Lackner M (2024a) Phragmén's voting methods and justified representation. Math. Program 203:47–76. https://doi.org/10.1007/s10107-023-01926-8

Brill M, Gölz P, Peters D, Schmidt-Kraepelin U, Wilker K (2024b) Approval-based apportionment. Math Program 203:77–105. https://doi.org/10.1007/s10107-022-01852-1

Casella A (2012) Storable votes: protecting the minority voice. Oxford University Press, Oxford

Elkind E, Faliszewski P, Laslier J-F, Skowron P, Slinko A, Talmon N (2017) What do multiwinner voting rules do? An experiment over the two-dimensional Euclidean domain. In: Proceedings of the 31st AAAI Conference on Artificial Intelligence (AAAI). AAAI Press, pp 494–501
Faliszewski P, Skowron P, Slinko A, Talmon N (2017) Multiwinner voting: A new challenge for social choice theory. In: Endriss U (ed) Trends in Computational Social Choice, Chapt. 2. AI Access
Freeman R, Zahedi SM, Conitzer V (2017) Fair social choice in dynamic settings. In: Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI). IJCAI, pp 4580–4587
Hemaspaandra E, Hemaspaandra LA, Rothe J (2017) The complexity of controlling candidate-sequential elections. Theor Comput Sci 678:14–21
Israel J, Brill M (2021) Dynamic proportional rankings. In: Proceedings of the 30th International Joint Conference on Artificial Intelligence (IJCAI). IJCAI, pp 261–267
Janson S (2016) Phragmén's and Thiele's election methods. Technical report. arXiv:1611.08826 [math.HO]
Janson S (2018) Thresholds quantifying proportionality criteria for election methods. Technical report. arXiv:1810.06377 [cs.GT]
Järvelin K, Kekäläinen J (2002) Cumulated gain-based evaluation of IR techniques. ACM Trans Inf Syst (TOIS) 20(4):422–446
Lackner M (2020) Perpetual voting: Fairness in long-term decision making. In: Proceedings of the 34th AAAI Conference on Artificial Intelligence (AAAI). AAAI Press, pp 2103–2110
Lackner M, Skowron P (2022) Multi-winner voting with approval preferences. Springer, Berlin
Mansbridge J (1999) Should Blacks represent Blacks and women represent women? A contingent "yes". J Polit 61(3):609–891
Monroe BL (1995) Fully proportional representation. Am Polit Sci Rev 89(4):925–940
Oren J, Lucier B (2014) Online (budgeted) social choice. In: Proceedings of the 28th AAAI Conference on Artificial Intelligence (AAAI). AAAI Press, pp 1456–1462
Parkes D, Procaccia A (2013) Dynamic social choice with evolving preferences. In: Proceedings of the 27th AAAI Conference on Artificial Intelligence (AAAI). AAAI Press, pp 767–773
Peters D, Skowron P (2020) Proportionality and the limits of welfarism. In: Proceedings of the 21st ACM Conference on Economics and Computation (ACM-EC). ACM, pp 793–794
Phragmén E (1894) Sur une méthode nouvelle pour réaliser, dans les élections, la représentation proportionnelle des partis. Öfversigt af Kongliga Vetenskaps-Akademiens Förhandlingar 51(3):133–137
Sánchez-Fernández L, Elkind E, Lackner M, Fernández N, Fisteus JA, Basanta Val P, Skowron P (2017) Proportional justified representation. In: Proceedings of the 31st AAAI Conference on Artificial Intelligence (AAAI). AAAI Press, pp 670–676
Schulze M (2011) Free riding and vote management under proportional representation by the single transferable vote. http://m-schulze.9mail.de/schulze2.pdf. Accessed 1 Nov 2023
Skowron P (2021) Proportionality degree of multiwinner rules. In: Proceedings of the 22nd ACM Conference on Economics and Computation (ACM-EC). ACM, pp 820–840
Skowron P, Lackner M, Brill M, Peters D, Elkind E (2017) Proportional rankings. In: Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI). IJCAI, pp 409–415
Tennenholtz M (2004) Transitive voting. In: Proceedings of the 5th ACM conference on electronic commerce (ACM-EC). ACM Press, pp 230–231
Thiele TN (1895) Om flerfoldsvalg. In: Oversigt over det Kongelige Danske Vidensk-abernes Selskabs Forhandlinger, pp 415–441