

Low Cost, Fair, and Representative Committees in a Metric Space

Christopher Jerrett and Elliot Anshelevich

Department of Computer Science, Rensselaer Polytechnic Institute

jerrec@rpi.edu, eanshel@cs.rpi.edu

October 27, 2025

Abstract

We study the problem of selecting a representative committee of k agents from a collection of n agents in a common metric space. This problem is related to choosing k facilities in facility location and k -median problems. However, unlike in more traditional facility location where each agent only cares about the *closest* selected facility, in the settings we consider each agent desires that *all* selected committee members are close to them. More precisely, we look at the sum objective, in which the goal is to minimize the total distance from all agents to all members of the chosen committee. We show that it is always possible to find a committee which is both *low-cost* according to this objective, and also *fair* according to many existing notions of fairness and proportionality defined for clustering settings. Moreover, we introduce a new desirable axiom for representative committees we call *NORP*, which prevents *over-representation* of any subset of agents. While all existing algorithms for fair committee selection do not satisfy this intuitive property, we provide new algorithms which form simultaneously *low-cost*, *fair*, and *NORP* solutions, thus showing that it is always possible to form low-cost, fair, and representative committees for our settings.

1 Introduction

Selecting a small set of representatives from a large set of points in a metric space is an important and foundational problem in AI and social choice. The distances in the metric space may encode, for example: physical proximity (e.g., for choosing a set of facilities in facility location), ideological alignment (e.g., political spectra and spatial voting [Enelow and Hinich \(1990\)](#); [Merrill and Grofman \(1999\)](#)), or general dissimilarity (e.g., dissimilarity between images when attempting to select a small set of images which are representative of a large collection [Kalayci et al. \(2024\)](#); [Sun et al. \(2012\)](#)). Consider forming a committee: there are n agents (voters) embedded in some ideological metric space, and we want to choose a small number of these voters as our committee of representatives. Crucially, the agents care about all of the chosen representatives' ideologies (as modeled by their locations in the metric space), not just their closest one. If all representatives have equal voting power, then agents should care about how much they like each chosen representative equally. We model this by attempting to minimize the total distance from all agents to all chosen representatives.

Formally, in our problem we are given a set N of n agents, an integer $k \leq n$ and a (pseudo)metric¹

¹We consider pseudometrics since we allow multiple different agents to be located at the same point, i.e., have distance 0 between them.

$d : N \times N \rightarrow \mathbb{R}^+$. We wish to select a k element subset $X \subseteq N$ which we will call the “representatives” of N . We consider X to be a good solution if it minimizes the natural sum cost function discussed above, i.e., $\sum_{v \in N} \sum_{x \in X} d(v, x)$. Note that, unlike other typical clustering objectives (including k -median, k -means, and k -center), efficiently computing a solution minimizing our cost function is trivial using a simple greedy algorithm.

The sum objective we consider here has been extensively studied in the heterogeneous facility location literature (see e.g., [Anastasiadis and Deligkas \(2018\)](#); [Deligkas et al. \(2023\)](#); [Serafino and Ventre \(2014\)](#); [Kanellopoulos et al. \(2023\)](#)). In this setting, we wish to construct k facilities in a metric space that minimize the sum objective. For example, consider the situation where a city planner must decide where to construct a collection of heterogeneous public facilities (a school, a post office, a court house, a city hall, a library, and a park). Importantly, *each facility serves a different purpose*, meaning that agents will need to visit and use all the facilities we build rather than rely on one or a few of them. This is distinct from clustering or more standard facility location problems, where agents are assigned to just one (typically the closest) facility. See Related Work for further discussion on this topic.

Fair and Low-cost Solutions As discussed above, minimizing the sum objective often makes sense for committee selection, heterogeneous facility location, and other settings where the goal is to select a small set of representative agents with each agent caring about their distance to *all* of the selected representatives. Thus, for the settings we are interested in, why not simply choose the solution optimizing this cost function, which we will call the *min-cost solution*? Unfortunately, while such a solution has clear advantages, and is easy to compute efficiently, it also has some important downsides.

Perhaps the main disadvantage of the min-cost solution is that it can be very *unfair*. When deciding on a committee, we often want different groups to have representation proportional to their size [Pukelsheim \(2017\)](#), even if it hurts the sum objective. Consider a very simple example on the line segment $[0, 1]$ where there are 51% of agents located at 0 and 49% located at 1. It seems reasonable, and consistent with basic principles of proportionality [Aziz and Lee \(2020\)](#); [Brill and Peters \(2023\)](#), to have about half of the representatives at 0 and half at 1. The min-cost solution, however, does not do this: all representatives are chosen to be at 0. Intuitively, this seems unfair; the agents located at 1 deserve to have at least one representative nearby, and preferably about $\frac{k}{2}$ representatives. This reveals a tension: the sum objective optimizes global efficiency but may ignore local fairness.² Yet in many applications—from political representation to public facilities to diverse image selection—both are essential. We thus would like to include some notion of fairness and proportionality in our idea of a desirable solution. Our main goal in this paper is to show how to form simultaneously **fair and low-cost solutions**. By “fair” we mean one that satisfies certain fairness axioms, discussed below, and by “low-cost” we mean one that is a constant factor approximation of the optimal solution in the sum objective.³

Proportionally Representative Fairness Many fairness properties have been introduced for our setting during the last few years (see detailed discussion in Related Work). We begin by focusing on Proportional Representative Fairness (PRF), introduced by [Aziz et al. \(2024\)](#). Suppose that we have $\ell \frac{n}{k}$ agents located at a single point. It is reasonable to require that at least ℓ of the k

²More generally, the min-cost solution may not be fair or proportional for any reasonable notion of proportionality, including PRF, mJR, α -1-core, mPJR, NORP, etc (see Related Work for discussion of these fairness concepts).

³Note that it is *not* possible to do this for many objectives, including k -median, k -means, and k -center (see Related Work). Unlike these clustering objectives, the sum objective permits both fairness and constant-factor approximations *simultaneously*.

representatives should also be located at that point, to ensure proportional representation, since these agents form $\frac{\ell}{k}$ fraction of the total. More generally, if a group of $\ell \frac{n}{k}$ agents is *not* located at a single point, it still makes sense to require that there be at least ℓ representatives nearby. For a cohesive group where all agents are near each other (a densely populated city), “nearby” should mean quite close, while for a less cohesive group where the agents may be far away from each other (a sparse collection of small rural villages), it would make sense for “nearby” to mean farther as well. In particular, if we define $D(S) = \max_{v,w \in S} d(v,w)$ to be the diameter of a set S of agents, then “nearby” can mean “proportional to the diameter”. This brings us to the following formal definition, which simply states that for any large set S , there are at least ℓ representatives within distance at most $D(S)$ from the set S .

Definition 1.1. A representative set X of size k satisfies *Proportionally Representative Fairness* (PRF) if the following holds. For any set of agents $S \subseteq N$ of size $|S| \geq \ell \frac{n}{k}$, it must be that $|x \in X : \exists v \in S \text{ s.t. } d(v, x) \leq D(S)| \geq \ell$.

While we study other fairness concepts as well, we choose to focus on PRF for several reasons: (1) PRF solutions always exist, and can be computed efficiently.⁴ This is in contrast to many other fairness concepts, for which only approximately fair solutions are guaranteed to exist. (2) We believe PRF neatly captures the basic intuition behind fairness and proportionality: a large set should have many representatives near it, where “near” means proportional to how spread out the set is. (3) The previous reason is, of course, subjective, and the reader may prefer one of the other known fairness concepts, described in Related Work. However, most of our results also hold for other fairness concepts in addition to PRF. In particular, we show that almost all of our results also hold for Metric Justified Representation (mJR, see Section 3 for definition), which due to Kellerhals and Peters (2024) immediately implies results for Proportional Fairness, Transferable Core, and Individual Fairness. Thus, if the reader prefers a different fairness notion to PRF, using Kellerhals and Peters (2024) our results for PRF and mJR likely extend to that as well.

No Over-Representation Property (NORP) Our work consists of two main contributions. First, we show that the sum objective is compatible with many existing notions of fairness, including PRF and mJR, and give efficient algorithms to compute simultaneously low-cost and fair solutions. Second, we introduce a novel fairness axiom to address the problem of over-representation, and show that it is possible to satisfy this property without losing much, i.e., while still forming fair and low-cost solutions.

An undesirable feature of many existing fairness axioms is that they can allow a small group of agents to be over-represented. Consider the following simple example, with large n and $k = 3$ given in Figure 1. Usual fairness axioms would allow all three representatives to be placed in the center. From the point of view of the $n - 3$ agents which are far away from the center, however, the 3 agents in the center are extremely privileged: why do those 3 agents get all 3 representatives right next to them? Moreover, the three chosen representatives lack diversity, as they are all at essentially the same point. If we instead place one representative in the center and two in the outer groups, then fairness axioms are still preserved while solving the above concern about the center location being over-represented.

The example in Figure 1 can be easily generalized to arbitrary k by using a star graph topology with k agents in the center and $\frac{n}{k} - 1$ agents at each of the arms of the star (see Figure 3). To the

⁴We study the setting in which the set of agents is the same as the set of possible representatives. In Aziz et al. (2024) they show that if the entire metric space is the candidate set then PRF exists; however they do this by only considering the locations of the agents as candidates, so their results apply to our setting as well.

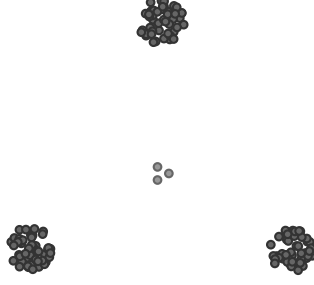


Figure 1: An example with a large number of agents in the plane and $k = 3$. We have three large groups of $\frac{n}{k} - 1$ agents placed symmetrically around the origin, with 3 agents at the origin. Then standard fairness axioms would allow all three representatives to be placed in the center.

best of our knowledge, all fairness notions existing in the literature so far would allow all or nearly all representatives to be placed in the center in such examples. All existing algorithms developed for our setting would also place all or nearly all representatives on top of the k center agents, giving those k agents disproportionate power, while also creating envy from almost all other agents. Why does this very small group of k agents in the middle have almost all the representatives directly on top of them? This is an undesirable property from a few different perspectives. The first is that agents could be envious of the small group of “elite” agents where representation is concentrated. In real settings, people are not only concerned with having proportionally enough, but also resent if someone else has disproportionately too much. Second, by allowing too many representatives to be concentrated in a small region, the solution could become homogeneous and lack diversity, making it less robust. Thus, we want a fairness concept that says that if a small area in the space has a large number of representatives, there should be a proportional number of agents nearby, so that area is truly deserving of so many representatives. To form this fairness concept, we consider arbitrary sets S of ℓ representatives. How many agents should be near S to “deserve” S being there, and what does “near” mean? We say that there are enough agents nearby if there are enough to justify $\ell - 1$ representatives, along with at least one more agent to justify adding another. So in total there must be strictly larger than $(\ell - 1)\frac{n}{k}$ agents. Similar to PRF, we can consider how cohesive S is, and define “near” to be within distance $D(S)$.

Definition 1.2. A set of representatives X of size k is said to satisfy *NORP* if for any set $S \subseteq X$ of size $|S| \geq \ell$ the following holds: $|v \in N : \exists x \in X \text{ s.t. } d(v, x) \leq D(S)| > (\ell - 1)\frac{n}{k}$.

Informally, this means that any set of ℓ representatives needs to have at least one more agent nearby than enough agents to justify $\ell - 1$ representatives. This new fairness concept is orthogonal to all existing work, since it requires that sets are not *over-represented* instead of ensuring they are not *under-represented*.

Our Contributions

We begin with establishing lower bounds, showing that it is *not* possible to always form either a PRF or a NORP solution which is better than a 2-approximation to the min-cost solution. We then show that this bound is tight, by providing an algorithm which forms a PRF and 2-approximate solution (Algorithm 2), and another which forms a NORP and 2-approximate solution (Algorithm 3). All of our algorithms are polynomial-time, except for Algorithm 2, which is poly-time in many relevant special cases, and can be transformed into a PTAS. Thus, we show that it is possible to

form low-cost PRF solutions, and low-cost NORP solutions, but what if we want all three at the same time?

Our main result (Section 5) shows that we can efficiently compute a fully PRF and NORP solution, which is a 4-approximation for the sum objective. This shows that all our fairness properties (as well as all the ones implied by PRF and mJR using Kellerhals and Peters (2024)) are compatible with having a low-cost solution at the same time, although the approximation factor is no longer the best possible (4 instead of 2). Finally, we show how relaxing axioms of fairness can improve the cost of the solution, by providing algorithms that form a 2-approximate 2-PRF solution in guaranteed poly-time, and an algorithm that provides a $2 + \frac{2}{\alpha}$ approximate, α -PRF and NORP solution all at once, allowing us to quantify the tradeoff between the cost of a solution and its level of fairness. We summarize the upper bounds given by our algorithms in Table 1.

One take-away message from our work is that NORP is a desirable new fairness property, whose addition does not really hurt. It is still possible to form solutions which satisfy most of the existing fairness concepts, while simultaneously satisfying NORP and producing solutions with small overall cost (and thus high agent happiness).

2 Related Work

A very large body of work exists on clustering and facility location in a metric space; this work is too much to survey here, see for example Ackermann et al. (2010); Jain et al. (1999); Jain (2010); Chang et al. (2016) and references therein. In most of this clustering work, the goal is to choose facilities in order to minimize the distances from each agent to their *closest* facility (e.g., k -median, k -means, or k -center objectives). In our work, however, we focus on applications where the agents do not care only about their closest chosen facility, but about *all* chosen facilities, i.e., the sum objective. Note that for objectives like k -median, it is *not possible* to achieve fairness properties that we study (e.g., PRF or mJR) at the same time as a constant approximation to the desired objective. Consider an example on a line where $n - k + 1$ agents are located at zero and $k - 1$ agents are positioned at $1, 1 + \varepsilon, 1 + 2\varepsilon, \dots, 1 + (k - 2)\varepsilon$. To optimize k -median, k -means, or k -center objectives, we would place $k - 1$ centers at $1, 1 + \varepsilon, 1 + 2\varepsilon, \dots, 1 + (k - 2)\varepsilon$ and one center at zero. However, this solution is inherently unfair: it under-represents the dense cluster on the left while over-representing the sparse cluster on the right. On the other hand, the sum objective permits both fairness and constant-factor approximations *simultaneously*.

As mentioned previously, the sum objective that we are interested in has been studied in the context of heterogeneous facility location problems. Such problems mostly have been considered in the mechanism design context, where the agents are strategic and can lie about their locations,

Algorithm	Sum Approximation	PRF	mJR	NORP
Algorithm 2	2	1	2	∞
Algorithm 3	2	∞	∞	1
Algorithm 4	4	1	1	1
Algorithm 5	2	2	2	∞
Algorithm 6	$2 + \frac{2}{\alpha}$	α	α	1

Table 1: Summary of upper bounds given by the solutions X given by each algorithm. The first column is the upper bound on the ratio $\frac{\text{cost}(X)}{\text{cost}(\text{OPT})}$. The other three columns give an upper bound on each of the fairness axioms we consider: 1 means that the property holds exactly, while anything larger means it holds approximately.

see [Chan et al. \(2021\)](#) for a survey. Because of this, the goal is usually to form algorithms which are *strategyproof*, i.e., the agents have no incentive to lie [Serafino and Ventre \(2014\)](#); [Xu et al. \(2021\)](#); [Kanellopoulos et al. \(2023\)](#); [Zhao et al. \(2024\)](#); [Kanellopoulos et al. \(2025\)](#); [Deligkas et al. \(2025\)](#). Various related objectives are also used in existing work; in addition to our sum objective the distance to the farthest facility (e.g., [Lotfi and Voudouris \(2024\)](#); [Deligkas et al. \(2025\)](#)) or to the q 'th closest facility (e.g., [Caragiannis et al. \(2022\)](#); [Ebadian and Micha \(2025\)](#)) have been studied. In contrast to most of this existing work, we do not assume that the agents are strategic, and instead focus on fairness properties in general metric spaces. Similar objectives have also been considered in the context of *metric distortion*, where the true locations of the agents are unknown and only their ordinal preferences are known, see e.g., [Goel et al. \(2018\)](#); [Caragiannis et al. \(2022\)](#). Extending our results to other objectives is the subject of our future work.

Research on fairness in metric clustering problems has recently received a lot of interest in the fields of Machine Learning, AI, and Social Choice; see survey [Chhabra et al. \(2021\)](#). Many compelling fairness properties have been defined for this setting, including Proportional Fairness [Chen et al. \(2019\)](#); [Micha and Shah \(2020\)](#), Individual Fairness [Jung et al. \(2020\)](#), and Transferable Core [Li et al. \(2021\)](#). There has also been a significant amount of work (e.g., [Mahabadi and Vakilian \(2020\)](#); [Negahbani and Chakrabarty \(2021\)](#); [Vakilian and Yalciner \(2022\)](#); [Bateni et al. \(2024\)](#)) forming approximately fair solutions that also approximate the best fair solution in the k -median or k -means objectives. Note that, as discussed above, it is *impossible* to form an approximately fair solution with respect to any of these fairness concepts that is also a good approximation to the optimum k -median solution; the work mentioned above forms an approximation with respect to the best *fair* solution. Unlike this existing work, we seek solutions which are a constant factor away from *any* solution for our objective, not just any *fair* solution.

While useful for many clustering applications, the fairness concepts mentioned in the previous paragraph have a major weakness: they require only a single representative to be close enough to each set of $\frac{n}{k}$ agents. For example, consider a line segment with 1 percent of agents at 0 and 99 percent at 1. If $k = 100$ then the solution that places 99 centers at 0 and 1 at 1 is considered fair by all the above definitions, despite greatly under-representing 99% and over-representing 1%. Because many settings, including the ones we discuss in the Introduction, desire some sort of proportional representation as well, several lines of work have generalized these fairness concepts to address this drawback. First, [Aziz et al. \(2024\)](#) introduced PRF, as defined in the Introduction. They showed that PRF solutions always exist, and can be efficiently computed. We use the concept of PRF and show that it is compatible with the sum objective as well as NORP. Second, [Kellerhals and Peters \(2024\)](#) defined metric versions of justified and proportional justified representation [Brill and Peters \(2023\)](#), which they call mJR and mPJR respectively. They showed both always exist and can be computed via Expanding Approvals [Aziz and Lee \(2020\)](#). mPJR is a strengthening of mJR and implies PRF⁵. They also showed various implications between mJR, mPJR, Proportional Fairness, Individual Fairness, Transferable Core, and q -core [Kellerhals and Peters \(2024\)](#). Importantly, they showed that mJR implies the best possible bounds for Proportional Fairness and Individual Fairness, as well as bounds for Transferable Core. Since the solutions formed by our algorithms satisfy mJR, this immediately means that they approximately satisfy these other fairness properties also. Finally, [Kalayci et al. \(2024\)](#) considered the metric committee selection problem with the same motivation as ours: the agents care about more than just their closest representative. They define a new fairness property which they call *Proportionally Representative* (distinct from PRF from [Aziz et al. \(2024\)](#)), which is a strengthening of Transferable Core. They show a variety of results

⁵A reader already familiar with mJR from [Kellerhals and Peters \(2024\)](#) may wonder why we use mJR in our work, instead of mPJR, which extends mJR to sets of size $\ell \frac{n}{k}$. In fact, mPJR is incompatible with NORP, and we believe this is evidence that mPJR is too strong of a property: see Appendix B.

and algorithms computing approximately fair solutions using their fairness concept. Unfortunately, exact Proportionally Representative solutions do not always exist, and resource augmentation (allowing fewer than the deserved amount ℓ representatives to be located near a set of size $\ell \frac{n}{k}$) is required to ensure that approximately fair solutions exist for this concept. While this existing work is very relevant to ours, we focus instead on forming fair solutions which are also compatible with low-cost solutions, and on satisfying NORP, a novel fairness concept which is orthogonal to all existing work, since it requires that sets are not *over-represented* instead of ensuring they are not *under-represented*.

3 Model and Techniques

We are given a set N of n agents in a pseudo-metric space with distance function d . The goal is to select a set X of size k from N . For ease of presentation, we will assume that k divides n in all our algorithms, so $\frac{n}{k}$ is an integer. All of our results, however, also hold for the more general case of arbitrary n and k . Rather than extending each of our results separately, we give a general reduction in Appendix A which shows how to extend all of our algorithms and results for the case when k does not divide n .

We will now give some basic notation and definitions that we will use the rest of the paper. We are interested in computing solutions X that have low cost in the sum objective, which we define as $\text{cost}_{(N,d)}(X) = \sum_{v \in N} \sum_{x \in X} d(v, x)$. We will omit N and d when clear from context. Likewise, the optimal solution OPT is defined as a set of size k with minimum cost, i.e., the set X which minimizes $\sum_{v \in N} \sum_{x \in X} d(v, x)$. We will refer to any solution X with $\text{cost}(X) \leq \alpha \cdot \text{cost}(\text{OPT})$ as an α -approximation.

Throughout the paper, we denote the diameter of a set S by $D(S) = \max_{v,w \in S} d(v, w)$, and the radius by $R(S) = \min_{c \in N} \max_{v \in S} d(c, v)$, the size of the smallest ball centered at some agent in the entire collection of agents N that contains the entire set. We will use the notation $B_S(x, \delta) = \{v \in S \mid d(v, x) \leq \delta\}$ for the set of points in S that are in the closed ball of radius δ centered at x , and $B_S^\circ(x, \delta) = \{v \in S \mid d(v, x) < \delta\}$ for the open ball. When the set S is clear from context or consists of the entire space, we will omit S .

We now review the axioms of fairness that we will consider in this paper, adjusted slightly for our setting in which any agent can be chosen as a representative. The main fairness concepts we use are PRF and NORP, already defined in the Introduction. Note that using the above notation, PRF means that there should be at least ℓ representatives in $\cup_{v \in S} B(v, D(S))$, and NORP means that there should be more than $(\ell - 1) \frac{n}{k}$ agents in $\cup_{x \in S} B(x, D(S))$. Likewise, we define mJR from Kellerhals and Peters (2024):

Definition 3.1. An outcome $X \subseteq N$ with $|X| = k$ satisfies *metric Justified Representation* (mJR) if for all $S \subseteq N$ with $|S| = \frac{n}{k}$, we have that $\cup_{v \in S} B(v, R(S))$ contains at least one element of X .

We analogously define approximate fairness properties: X satisfies α -approximate PRF (which we will abbreviate to simply α -PRF) if it satisfies PRF, but with $\alpha D(S)$ instead of $D(S)$. In other words, the needed representatives are allowed to be farther than members of S , but still not too far. Similarly, α -mJR means satisfying mJR, but with distance requirement $\alpha R(S)$ instead of $R(S)$.

Techniques Before presenting detailed results, we give an overview of the techniques used in our algorithms. All of our algorithms are similar to the Greedy Capture algorithm of Chen et al. (2019), the Truncated Greedy Capture algorithm of Kalayci et al. (2024), and the Expanding Approvals algorithm of Aziz and Lee (2020). At the start of the algorithm, we begin with all agents in N

being *uncovered*. At a high level, all of our algorithms consist of selecting a set S_i of agents of size $\frac{n}{k}$ at each iteration i , and then selecting a representative x_i from this set. We will call agents in set S_i as *covered* by set S_i . We will also sometimes refer to representative x_i as *covering* all the agents in set S_i . Once an agent is covered in iteration i , we will never select them as a future member of a set S_j for $j > i$. We continue in this manner until all agents are covered; since we cover exactly $\frac{n}{k}$ agents at each iteration, we end up with exactly k representatives.

Notice that simply running previously known versions of Greedy Capture or Truncated Greedy Capture (which choose S_i to be the ball with smallest radius, and choose the center of this ball as x_i), can result in solutions which are not a good approximation to OPT, and do not satisfy NORP. Because of this, we must be careful about how we choose sets S_i and representatives x_i . Satisfying one or two of our desired properties is not too difficult, but as we add more constraints, choosing the right agent as representative becomes more complex, and requires a much more sophisticated analysis to show our bounds. Below we give a high-level sketch of how our algorithms behave; for full details see Sections 4, 5, 6, and the Appendix.

Algorithm 2: The set S_i is the smallest-diameter uncovered set, and x_i is the agent in S_i closest to OPT. When picking the smallest diameter sets we can select anything in S_i as a representative and still have the solution be PRF; by selecting x_i in S_i that is closest to OPT we simultaneously get a 2-approximation to OPT.

Algorithm 3: The representative x_i is the uncovered agent closest to OPT. The set S_i are the $\frac{n}{k}$ closest uncovered agents to x_i . By only selecting uncovered agents closest to OPT as representatives, we get a 2-approximation. Then by covering everything in a ball around it, we ensure that other representatives are far enough away that we achieve a NORP solution.

Algorithm 4: The set S_i is the smallest-radius uncovered set, and x_i is the agent in its center. Our algorithm is closely related to the many variants of Greedy Capture and Expanding Approvals algorithms from the literature (e.g., Aziz and Lee (2020, 2021)), and is most similar to the Truncated Greedy Capture algorithm of Kalayci et al. (2024). However, there are two crucial differences between our algorithm and all previous approaches in the literature. First, we stop growing balls for agents that are covered; this is crucial to avoid over-representation, as illustrated by Example 1 discussed in the Introduction. Unlike algorithms in existing work, this allows us to satisfy NORP and similar properties. Second, we move the last representative x_k to the closest member of its ball from OPT, instead of leaving it in the center of the ball. This is necessary to make our solutions become a good approximation to OPT, as without this change our algorithm (as well as all algorithms in the literature, to the best of our knowledge) can result in an arbitrarily bad approximation ratio, see 5. Despite the difference in the choice of the last representative, we can still show that this algorithm achieves PRF and mJR, while also satisfying NORP and guaranteeing a 4-approximation to OPT.

Algorithm 5: The sets S_i are the same as in Algorithm 4 but we move all representatives to the agent in S_i closest to OPT. This can violate NORP, and is only 2-PRF, but gives a 2-mJR solution and a 2-approximate solution in polynomial time.

Algorithm 6: Grows the sets S_i in the same way as Algorithm 4, except for the uncovered agent that is closest to OPT that has its radius grow at a rate α times faster than all other agents. x_i is selected the same as in Algorithm 4. This gives a $2 + \frac{2}{\alpha}$ approximate and α -PRF solution, while also satisfying NORP.

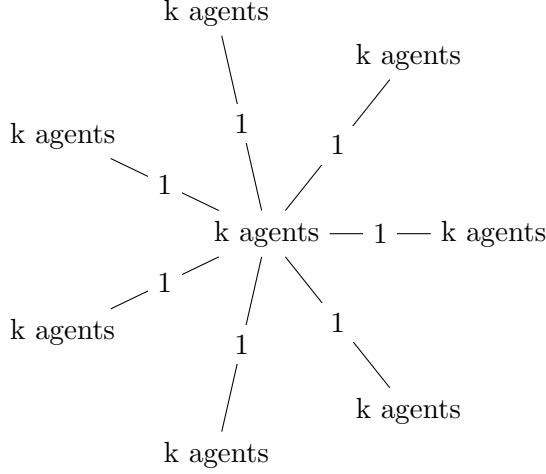


Figure 2: Example illustrating the incompatibility of PRF, NORP, and $(2 - \epsilon)$ -approximate solutions.

4 Basic Compatibility of Low-Cost and Fairness

We begin by establishing basic impossibility results: satisfying either PRF or NORP (even approximately) is not compatible with forming a better than 2-approximation for our cost objective.⁶

Proposition 4.1. *For any $\epsilon > 0$, there exists an instance where all solutions with cost at most $(2 - \epsilon) \text{cost}(\text{OPT})$ do not satisfy α -PRF (for any finite $\alpha \geq 1$) or NORP.*

Proof. See the example in Figure 2. In this example assume that $k = \sqrt{n}$ and n is a perfect square. We have k agents located at each leaf of a star graph, and k at the center; the distances from the center to each leaf are all equal to 1, and there are $k - 1$ leaves. Consider any set of agents of size $\frac{n}{k} = k$ located at the same point. This is a set of diameter zero, so for a solution to be α -PRF (for any finite α), it must place at least one representative at that point. Thus, we see that for a solution to be α -PRF we must place a representative at each arm of the star graph, and one at the center. Likewise, NORP would allow at most 1 representative in the center, since otherwise the 2 representatives in the center of the graph would constitute a set with diameter zero, but with only $\frac{n}{k}$ agents distance zero away from them. We have $k - 1$ arms. Thus the cost of any α -PRF or NORP solution is $k(k - 1)(1 + 2(k - 2)) + k(k - 1) > 2k(k - 1)(k - 2)$. On the other hand, OPT places all representatives in the center, thus the cost of OPT is $k^2(k - 1)$. Thus the approximation ratio between the cost of any α -PRF or NORP solution, and the cost of OPT, approaches 2 as n and k approach ∞ . $\square$

Our main result in this paper is given in Section 5, where we give an algorithm that provides a simultaneously PRF, NORP, and mJR solution that is at worst a 4-approximation to OPT. In the rest of this section, however, we first consider PRF and NORP individually, and show that the above lower bound is tight by proving that there always exists a 2-approximate solution satisfying PRF, and the same is true for NORP. The algorithms to compute these solutions, and the proofs of these results, are comparatively simple, although we will use ideas from both algorithms in Section 5.

⁶In fact, it is co-NP-hard to determine if a PRF solution exists which is better than a 2-approximation to OPT, see Appendix C.

First, we will introduce an algorithm that gives us a solution satisfying PRF that simultaneously is also at most 2 times the cost of the optimal solution. The algorithm proceeds as follows: we start with all agents “uncovered” then over the course of the algorithm we consider agents covered when a representative is placed nearby. The algorithm works by repeatedly finding the smallest diameter subset of the uncovered agents with size $\frac{n}{k}$. Then we select the agent in the set that minimizes its average distance to OPT. By finding these aforementioned small sets, we ensure that PRF is maintained, and we also ensure that an efficient and low-cost solution is found by selecting representatives in each set that are relatively close to the optimal solution, OPT. Unfortunately, in general, implementing our algorithm is NP-Hard for general metric spaces, via a trivial reduction from *Clique*. (We will later give a polynomial time approximately fair algorithm, as well as a perfectly fair 4-approximate solution). Fortunately our algorithm is polynomial time when the set of points lies in $\mathbb{R}^2$ with the L_2 metric (Aggarwal et al., 1989) and is fixed parameter tractable for the L_∞ metric (Eppstein and Erickson, 1994). In three dimensions there is also a PTAS that for any $\epsilon > 0$ finds a set that has diameter at most $(1 + \epsilon)$ away from the diameter of the smallest set in time $O(n \log n + 2^{O(1/(\epsilon^{1.5} \log \epsilon))})$ (Jiang, 2007). To the best of our knowledge, it is unknown whether the problem of selecting a subset of points of k elements in R^n with the norm L_2 that minimizes the diameter is NP-Hard. Note that if we use the PTAS given by (Jiang, 2007), we get a solution that is still a two-approximation for our objective, but is an approximate $(1 + \epsilon)$ -PRF solution.

Algorithm 1 Forms 2-approximate, 1-PRF solution

1:	$Z \leftarrow N$	▷ The set of agents that are uncovered so far
2:	while $Z \neq \emptyset$ do	
3:	$S_i \leftarrow \arg \min_{\substack{S \subseteq Z \\ S = \frac{n}{k}}} \max_{j, j' \in S} d(j, j')$	▷ Greedily find the agents that we cover
4:	$Z \leftarrow Z \setminus S_i$	▷ Remove the agents that we covered
5:	$\delta_i \leftarrow \max_{j, j' \in S_i} d(j, j')$	▷ The diameter of the set that we found
6:	$s_i \leftarrow \arg \min_{v \in S_i} \sum_{o \in \text{OPT}} d(v, o)$	▷ Pick the agent that is closest to OPT
7:	end while	
8:	$X \leftarrow \{s_i \mid i \in 1, \dots, k\}$	

In the above algorithm, as in all our algorithms, if there is a tie for selecting a set S_i , or a representative s_i , then one can be chosen arbitrarily. The algorithm will give us all the desired properties with any tie-breaking rule. We will now introduce some terminology that we will use frequently in the following proofs. Similar concepts will be used in the proofs for other algorithms later on. We will call the set Z consisting of a subset of agents the uncovered agents. Notice that in each iteration Z has the set S_i removed from it, and $S_i \subseteq Z$ before the set is removed from Z on line 4. Therefore, in each iteration we remove $\frac{n}{k}$ agents, effectively covering these agents. Likewise, we will also refer to representatives that cover agents. We say that a representative s_i covers an agent v if v was covered on the same iteration as in which s_i was selected, or equivalently $v \in S_i$. The next fact to note is that the values of δ_i are non-decreasing. This fact leads to an equivalent formulation of Algorithm 1 as a continuous process given by Algorithm 2.

The continuous formulation of the above algorithm is presented in Algorithm 2. The algorithm starts with an initial diameter $\delta = 0$, and continuously increases the diameter until a set of size $\frac{n}{k}$ is found that has a diameter of δ . Notice that the next set we find is always precisely one of the sets in $\arg \min_{\substack{S \subseteq Z \\ |S| = \frac{n}{k}}} \max_{j, j' \in S} d(j, j')$. The continuous version allows us to introduce the notion of time. We can treat δ as the time. In our proofs and discussion when we refer to time t we refer to

Algorithm 2 Forms 2-approximate, 1-PRF solution continuously

```
1:  $Z \leftarrow N$  ▷ The set of agents that are uncovered so far
2:  $\delta \leftarrow 0$ 
3: while  $Z \neq \emptyset$  do
4:   while  $\{S \mid (S \subseteq Z) \wedge (|S| = \frac{n}{k}) \wedge (D(S) = \delta)\} = \emptyset$  do
5:     smoothly increase  $\delta$ 
6:   end while
7:    $S_i \in \{S \mid (S \subseteq Z) \wedge (|S| = \frac{n}{k}) \wedge (D(S) = \delta)\}$ 
8:    $Z \leftarrow Z \setminus S_i$  ▷ Remove the agents that we covered
9:    $\delta_i \leftarrow \delta$ 
10:   $s_i \leftarrow \arg \min_{v \in S_i} \sum_{o \in \text{OPT}} d(v, o)$  ▷ Pick the agent that is closest to OPT
11: end while
12:  $X \leftarrow \{s_i \mid i \in 1, \dots, k\}$ 
```

the point in the running time of the continuous formulation of Algorithm 1 when $\delta = t$. For the rest of the paper we will present algorithms as continuous processes, but in the same way that we have done here, we can convert them to discrete algorithms that require k iterations. Now we can move on and show the fairness and efficiency guarantees of Algorithm 1.

Lemma 4.2. *The solution X given by Algorithm 1 satisfies PRF.*

Proof. Let $S \subseteq N$ be a set of agents of size $\ell \frac{n}{k}$ with diameter $D(S)$. Consider two cases: the first where $D(S) < \delta_k$, and the case where $D(S) \geq \delta_k$.

First, we consider the case where $D(S) < \delta_k$. Then after time $D(S)$, it must be that fewer than $\frac{n}{k}$ agents are left uncovered in S . This is because otherwise we would have added an additional representative to cover some of the set S , since any subset of S has diameter at most $D(S)$. Notice that each chosen representative can cover at most $\frac{n}{k}$ agents in S , and therefore there are at least ℓ representatives that cover something in S after time $D(S)$. For any s_i that covers some agent $v \in S$ by time $D(S)$, it must be that $d(v, s_i) \leq \delta_i \leq D(S)$, since s_i was chosen at time δ_i before time $D(S)$. Thus, all of these ℓ representatives are at most $D(S)$ away from something in S , and so PRF is satisfied for S .

Now suppose $D(S) \geq \delta_k$. At the end of the algorithm, all agents are covered. Since each representative can only cover $\frac{n}{k}$ agents, and each representative is at most distance δ_k from the agents they cover, it must be that there are at least ℓ representatives within distance $D(S)$ from something in S . $\square$

Now we will show that $\text{cost}(X) \leq 2 \cdot \text{cost}(\text{OPT})$. The proof is quite general and simple. So, first, we will state a general proposition from which the cost bounds follow directly.

Proposition 4.3. *Let X be a set of representatives, and $P = \{P_1, P_2, \dots, P_k\}$ be a partition of N into k sets of equal size. Then if $X = \{\arg \min_{v \in P_i} \sum_{o \in \text{OPT}} d(v, o) \mid P_i \in P\}$ (with ties broken arbitrarily), then $\text{cost}(X) \leq 2 \cdot \text{cost}(\text{OPT})$.*

Proof. Let $x_i \in X$ be the representative in X such that $x_i \in \arg \min_{v \in P_i} \sum_{o \in \text{OPT}} d(v, o)$. Then we

have that the cost of X is given by the following:

$$\sum_{x_i \in X} \sum_{v \in N} d(v, x_i) \leq \frac{1}{k} \sum_{x_i \in X} \sum_{v \in N} \sum_{o \in \text{OPT}} [d(v, o) + d(x_i, o)] \quad (1)$$

$$= \text{cost}(\text{OPT}) + \frac{n}{k} \sum_{x_i \in X} \sum_{o \in \text{OPT}} d(x_i, o) \quad (2)$$

$$= \text{cost}(\text{OPT}) + \frac{n}{k} \sum_{x_i \in X} \sum_{o \in \text{OPT}} \frac{k}{n} \sum_{v \in P_i} d(x_i, o) \quad (3)$$

$$= \text{cost}(\text{OPT}) + \sum_{x_i \in X} \sum_{o \in \text{OPT}} \sum_{v \in P_i} d(x_i, o) \quad (4)$$

where the first line comes from applying the triangle inequality $d(v, x_i) \leq d(v, o) + d(o, x_i)$ for all $o \in \text{OPT}$ simultaneously (and averaging the k inequalities). We recall that the bound $\sum_{o \in \text{OPT}} d(v, o) \geq \sum_{o \in \text{OPT}} d(x_i, o)$ holds for all $v \in P_i$, since we chose an element of P_i which minimizes $\sum_{o \in \text{OPT}} d(v, o)$ as x_i . Thus we obtain the following bounds:

$$\text{cost}(\text{OPT}) + \sum_{x_i \in X} \sum_{o \in \text{OPT}} \sum_{v \in P_i} d(x_i, o) \leq \text{cost}(\text{OPT}) + \sum_{x_i \in X} \sum_{v \in P_i} \sum_{o \in \text{OPT}} d(v, o) \quad (5)$$

$$= \text{cost}(\text{OPT}) + \sum_{v \in N} \sum_{o \in \text{OPT}} d(v, o) \quad (6)$$

$$= 2 \text{cost}(\text{OPT}). \quad (7)$$

Thus, $\text{cost}(X) \leq 2 \cdot \text{cost}(\text{OPT})$. $\square$

Corollary 4.4. *The solution X given by Algorithm 1 has cost $\text{cost}(X) \leq 2 \cdot \text{cost}(\text{OPT})$.*

Proof. Notice that the sets S_i form a partition and $s_i \in \arg \min_{v \in S_i} \sum_{o \in \text{OPT}} d(v, o)$. $\square$

Next, we will show a useful reduction lemma that allows us to not have to repeatedly prove approximate mJR bounds. We show that PRF implies a 2 approximation to mJR.

Lemma 4.5. *If X is α -PRF then X is 2α -mJR.*

Proof. Let X be a PRF solution. Then let $S \subseteq N$ be a subset of size $\frac{n}{k}$. Recall that the radius of S is defined as the smallest δ such that there is an agent $v \in N$ with $B(v, \delta) \supseteq S$. We then need to show that there is a representative in $\cup_{v \in S} B(v, \alpha R(S))$ for X to satisfy α -mJR.

Let v be the agent in N such that $B(v, R(S)) \supseteq S$, and v_1 and v_2 be the agents in S such that $d(v_1, v_2) = \max_{v, v' \in S} d(v, v')$. Note that $D(S) = d(v_1, v_2) \leq d(v_1, v) + d(v, v_2) \leq 2R(S)$. Thus, we have $D(S) \leq 2R(S)$, and by α -PRF there is a representative in $\cup_{v \in S} B(v, \alpha D(S)) \supseteq \cup_{v \in S} B(v, 2\alpha R(S))$. $\square$

Corollary 4.6. *The solution returned by Algorithm 1 is 2 -mJR.*

Proof. Follows from Lemma 4.2 and Lemma 4.5. $\square$

We have now seen an algorithm which returns the best possible approximation for OPT (assuming we want some sort of fairness), while forming a PRF solution. Unfortunately, this algorithm may return a solution that is not NORP. To see this, consider the example in Figure 3. Algorithm 1 would select each arm and the closest central agent at each iteration, then place the representative

in the central region. Thus, there are 6 representatives with a diameter of ϵ and only 6 agents within ϵ of a representative in the solution.

Now that we have shown that PRF is compatible with a 2-approximate solution, we ask the same question about NORP: is it always possible to form a 2-approximate solution which is NORP? Intuitively, satisfying NORP can be easy: simply place representatives far away from each other. However, requiring a solution which is both low-cost and NORP is trickier, since a low cost solution should have most of its representatives be centrally located. Thus, we need the representatives to be centrally located while also avoiding overrepresentation.

The next algorithm works by repeatedly growing the ball centered on the uncovered agent v in OPT that minimizes $\sum_{o \in \text{OPT}} d(v, o)$. Then, when the ball contains $\frac{n}{k}$ uncovered agents, we add the agent at the center of the ball to the set of representatives and cover the $\frac{n}{k}$ agents closest to v . Thus, the algorithm runs for k iterations and places k representatives, each of which covers $\frac{n}{k}$ agents. We therefore have a 2-approximate solution and a NORP solution simultaneously in polynomial time. By removing the closest $\frac{n}{k}$ agents from never being a representative in the future we guarantee that the solution will have a so-called “nesting” property, meaning for any two representatives x and x' , if x' is added after x , there will be $\frac{n}{k}$ agents with distance to x less than $d(x, x')$. Like previously,

Algorithm 3 $(2, \infty, 1)$ algorithm

```

1:  $Z \leftarrow N$  ▷ Uncovered Agents
2:  $\delta \leftarrow 0$ 
3:  $i \leftarrow 1$ 
4: while  $Z \neq \emptyset$  do
5:    $s_i \leftarrow \arg \min_{v \in Z} \sum_{o \in \text{OPT}} d(v, o)$  ▷ Agent closest to OPT, whose ball we grow
6:   while  $|B(s_i, \delta) \cap Z| < \frac{n}{k}$  do
7:     smoothly increase  $\delta$ 
8:   end while
9:    $S_i \leftarrow \arg \min_{\substack{T \subseteq B(s_i, \delta) \cap Z \\ |T| = \frac{n}{k}}} \sum_{v \in T} d(v, s_i)$ 
10:   $Z \leftarrow Z \setminus S_i$ 
11:   $\delta_i \leftarrow \delta$ 
12:   $i \leftarrow i + 1$ 
13: end while
14:  $X \leftarrow \{s_i \mid i \in 1, \dots, k\}$ 

```

we define the set Z as the set of uncovered agents and $s_i = x_i$ to cover v in the same iteration when s_i was selected, v was covered. Also, like previously, we refer to δ as the time in the continuous version of the algorithm.

In addition, we observe that the returned solution may not be α -PRF for any $\alpha > 0$. To see this, consider the example in Figure 4.

Theorem 4.7. *Solution X formed by Algorithm 3 is NORP and has cost $\text{cost}(X) \leq 2 \cdot \text{cost}(\text{OPT})$.*

We now present our first proof that a solution satisfies NORP. The high-level idea is that each representative has a ball around it when it is selected. Then by selecting representatives that are not inside each other’s balls we “nest” them so that the representatives are some minimum distance from each other, ensuring that $D(S)$ is sufficiently large for any subset S of X .

Lemma 4.8. *Solution X formed by Algorithm 3 satisfies NORP.*

Proof. Let $S \subseteq X$ be a set of representatives of size $\ell > 1$ (or else NORP is trivially satisfied). Note that due to the algorithm, each representative s_i that is chosen has a corresponding value δ_i ; the set S_i of agents which are covered by s_i is guaranteed to lie in $B(s_i, \delta_i)$.

Let x and y be arbitrary representatives in the set S . Let δ_x and δ_y be the values of δ when x and y were selected as representatives in the algorithm. To prove our result, we first prove that $D(S) \geq \min(\delta_x, \delta_y)$.

Suppose that x was added after y . Then we know that $x \notin B^\circ(y, \delta_y)$, or otherwise we would have covered x when selecting y . Thus, we have $d(x, y) \geq \delta_y$. Since y may have been added after x instead, this means that we always have either $d(x, y) \geq \delta_y$ or $d(x, y) \geq \delta_x$. Together, this means that $D(S) \geq \min(\delta_x, \delta_y)$.

To prove the lemma, we need to show that there are enough agents near S to justify having ℓ representatives. Consider the two representatives in S with the first and second largest value of δ_i ; we will call these representatives x_l and x_s (respectively), and their corresponding δ_i values will be denoted by δ_l and δ_s . By the above argument, we know that $D(S) \geq \min(\delta_s, \delta_l) = \delta_s$. Thus, if we can prove that $\cup_{s_i \in S} B(s_i, \delta_s)$ contains a lot of agents, then we are done, since $\cup_{s_i \in S} B(s_i, \delta_s) \subseteq \cup_{s_i \in S} B(s_i, D(S))$.

Let $S' = S \setminus \{x_l\}$. We know that $\delta_i \leq \delta_s$ for all $s_i \in S'$, since δ_s is the second largest δ value in set S . Recall that each ball centered on representative $s_i \in S'$ covered $\frac{n}{k}$ agents, all of which were at most δ_i away from s_i . In addition, no agent was removed by two different sets, so for all $s_i \in S'$ the sets they cover are disjoint. So, there are at least $(\ell - 1) \frac{n}{k}$ agents in $\cup_{s_i \in S'} B(s_i, \delta_s)$, and thus in $\cup_{s_i \in S'} B(s_i, D(S))$. Now consider the ball of radius $D(S)$ centered on the last representative, x_l . There is at least one additional agent that is covered by representative x_l located at x_l , since we chose x_l as a representative and we only choose uncovered agents as representatives. Thus, we have at least $(\ell - 1) \frac{n}{k} + 1$ agents in $\cup_{x \in S} B(x, D(S))$ for every S , so X satisfies NORP. $\square$

The following lemma follows directly from an application of Proposition 4.3. However, the fact that NORP is compatible (with such a low approximation factor) with forming a low-cost solution is somewhat surprising. The specific cost function we consider here ends up with OPT being a dense low-diameter set, while NORP requires that the solution is the opposite: sparse and spread out. Despite the seeming incompatibility, we arrive at a simple approximation bound, that is also the best possible according to Proposition 4.1.

Lemma 4.9. *Solution X formed by Algorithm 3 has cost $\text{cost}(X) \leq 2 \cdot \text{cost}(\text{OPT})$.*

Proof. The proof follows immediately from Proposition 4.3. $\square$

5 Main Result: Low-cost, Fair, and NORP at the same time

Now that we have an algorithm that provides a low-cost fair solution, and another that gives a low-cost NORP solution, we will attempt to combine the ideas of both algorithms and introduce our next algorithm, which will do both *simultaneously*. Algorithm 4 gives a solution that satisfies all our desired fairness concepts: PRF, NORP, and mJR. At the same time, the solution it forms has cost that is at most four times that of the optimal solution. Our algorithm starts by smoothly growing balls around all agents in N ; if multiple agents are located at the same point, we grow each ball independently. Eventually, when a ball has at least $\frac{n}{k}$ uncovered agents, we add the agent whose ball contained $\frac{n}{k}$ total uncovered agents to our set. We then greedily cover agents from the closest to the center outward (including the agent that grew the ball), so that the only agents that are not covered must be at least δ_i (the radius of the ball when x_i was added) away from the center x_i . We then stop growing the balls around all the agents that are covered. We continue in this

manner: each time there is a ball with agents uncovered $\frac{n}{k}$, we then add the representative at the center of this ball and remove the closest $\frac{n}{k}$ from it from ever becoming a center. Recall that k divides n so when we place k representatives we would have removed $k \cdot \frac{n}{k} = n$ agents; we then stop growing the balls when we cover all agents. Lastly, we move the last center x_k to the agent in the set that x_k covered that minimizes its average distance to the optimal solution. Note that we present our algorithm as a continuous process, although it is easy to discretize our algorithm and transform it into one that is polynomial in time, similarly to the technique in [Chen et al. \(2019\)](#).

Algorithm 4 (4,1,1) algorithm

```

1:  $Z \leftarrow N$ 
2:  $\delta \leftarrow 0$ 
3:  $i \leftarrow 1$ 
4: while  $Z \neq \emptyset$  do
5:   while  $\exists x \in Z : |B(x, \delta) \cap Z| \geq \frac{n}{k}$  do
6:      $S_i \leftarrow \arg \min_{\substack{T \subseteq B(x, \delta) \cap Z \\ |T| = \frac{n}{k}}} \sum_{v \in T} d(v, x)$ 
7:      $Z \leftarrow Z \setminus S_i$ 
8:      $s_i \leftarrow x$ 
9:      $\delta_i \leftarrow \delta$ 
10:     $i \leftarrow i + 1$ 
11:   end while
12:   smoothly increase  $\delta$ 
13: end while
14:  $X \leftarrow \{s_i | i < k\} \cup \{\arg \min_{v \in S_k} d(v, \text{OPT})\}$ 

```

Our algorithm is closely related to the many variants of Greedy Capture and Expanding Approvals algorithms from the literature (e.g., [Aziz and Lee \(2020, 2021\)](#)), which are most similar to the Truncated Greedy Capture algorithm of [Kalayci et al. \(2024\)](#). However, there are two crucial differences between our algorithm and all previous approaches in the literature. First, we stop growing balls for agents that are covered; this is crucial to avoid over-representation as depicted in [Example 1](#). Unlike algorithms in existing work, this allows us to satisfy NORP and similar properties. Second, we move the last representative to the closest member of its ball from OPT, instead of leaving it in the center of the ball. This is necessary to make our solutions become a good approximation to OPT, as without this change our algorithm (as well as all algorithms in the literature, to the best of our knowledge) can result in an arbitrarily bad approximation ratio. For example, consider the simple example in [Figure 5](#): running variants of Greedy Capture without moving the last center results in a solution that is arbitrarily bad compared to the optimal solution.

Theorem 5.1. *Let X be the solution returned by Algorithm 4. Then X satisfies PRF, mJR, and NORP, with $\text{cost}(X) \leq 4 \cdot \text{cost}(\text{OPT})$.*

Note that the sets S_i form a partition of the agents N , since at each iteration we remove S_i from Z , and each S_i is of exactly the size $\frac{n}{k}$. In addition, each S_i has exactly one representative x_i of X as a member: for $i < k$ this is just $x_i = s_i$, and for $i = k$ this is the agent $x_k \in S_k$ that minimizes the distance to OPT. As described above, we will refer to an agent v as *covered* by the set S_i (and representative x_i) if this agent $v \in S_i$. We will also refer to the solution at *time* t to mean the sets S_i and centers s_i chosen so far by Algorithm 4 when $\delta = t$, since the value of δ is continuously increasing. We will refer to the set Z at time δ as *uncovered* agents.

The proof that Algorithm 4 satisfies PRF follows the same logic as the PRF proof for Algorithm 2.

Lemma 5.2. *The solution X of Algorithm 4 satisfies PRF.*

Proof. Let $S \subseteq N$ be a set of agents of size $\ell \frac{n}{k}$ with diameter $D(S)$. Consider two cases: the first where $D(S) < \delta_k$, and the case where $D(S) \geq \delta_k$.

First, we consider the case where $D(S) < \delta_k$. Then after time $D(S)$, it must be that fewer than $\frac{n}{k}$ agents are left uncovered in S . This is because otherwise we would have added an additional representative to cover some of the set S , since any subset of S has diameter at most $D(S)$, and thus a ball centered at one of the uncovered agents with radius $D(S)$ would contain all uncovered agents of S . Notice that each chosen representative can cover at most $\frac{n}{k}$ agents in S , and therefore there are at least ℓ representatives that cover something in S after time $D(S)$. Since $D(S) < \delta_k$, all the representatives chosen by time $D(S)$ are centers of the chosen balls, since we only move the last representative in X . Thus, for any $s_i \in X$ that covers some agent $v \in S$ by time $D(S)$, it must be that $d(v, s_i) \leq \delta_i \leq D(S)$, since s_i was chosen at time δ_i before time $D(S)$. Thus, all of these ℓ representatives are at most $D(S)$ away from something in S , and so PRF is satisfied for S .

Now suppose $D(S) \geq \delta_k$. At the end of the algorithm, all agents are covered. Since each representative can only cover $\frac{n}{k}$ agents, it must be that there are at least ℓ representatives covering something in S . Note that with the exception of the last representative, all representatives are at most δ_k away from all agents in their covered set, since representative s_i covers agents in a ball of radius $\delta_i \leq \delta_k$. Thus, if there are ℓ representatives that are not the last representative that cover something in S , then we have at least ℓ representatives in $\cup_{v \in S} B(v, \delta_k) \subseteq \cup_{v \in S} B(v, D(S))$, and S satisfies the PRF condition. Now suppose that there are only $\ell - 1$ representatives that are not the last representative covering something in S . Since $|S| = \ell \frac{n}{k}$, this means that there are exactly ℓ representatives covering something in S , with each covering exactly $\frac{n}{k}$ agents of S , and with one of these being the last representative x_k . Moreover, this means that for every representative covering something in S , its corresponding set S_i (the set of agents it covers) is entirely contained in S , i.e., $S_i \subseteq S$. But the i 'th representative in our algorithm is always chosen from the set S_i : this is true for the last representative as well, since the last representative x_k is not necessarily the center of S_k , but it is still chosen from among the elements of S_k . Therefore, all ℓ representatives covering elements of S are in fact members of S themselves in this case, and so trivially $\cup_{v \in S} B(v, 0) \subseteq \cup_{v \in S} B(v, D(S))$ contains at least ℓ representatives. This completes the proof that S always satisfies the PRF condition, with S being arbitrary. $\square$

Now we prove that the algorithm outputs a NORP solution. Again, the argument works similarly as before, except we now need to show that moving the last representative cannot create a situation that causes over representation. Note that the following argument does rely on the fact that the radius of the balls is increasing over time; this is in contrast to the proof of NORP for Algorithm 1 where the radius of the chosen balls is not monotone with time.

Lemma 5.3. *The solution X of Algorithm 4 satisfies NORP.*

Proof. Let $S \subseteq X$ be a set of representatives of size $\ell > 1$ (or else NORP is trivially satisfied). Let $x_l \in S$ be the representative added the latest in S by Algorithm 4, and likewise $x_s \in S$ be the second latest representative in S selected by Algorithm 4. We will also define $S' = S \setminus \{x_l\}$ to be the set of representatives in S , excluding the representative added the latest in S . Let δ_s be the time when x_s was selected. Now we consider the diameter of this set S . First, we see that the last center, x_l , must be at least δ_s from x_s since we cover (and thus will never select as representatives)

all agents strictly closer than δ_s to x_s when we picked it (since we cover agents greedily). Thus, we can conclude that the diameter $D(S) \geq \delta_s$.

Now we need to show that there are enough agents near S to justify having ℓ representatives. We know that $\delta_j \leq \delta_s$ for all $x_j \in S'$, since x_j was added earlier in the algorithm than x_s . Recall that each ball centered on representative $x_j \in S'$ covered $\frac{n}{k}$ agents, all of which were at most δ_j away from x_j . In addition, no agent was removed by two different centers, so for all $x_j \in S'$ the sets they cover are disjoint. So, there are at least $(\ell - 1)\frac{n}{k}$ agents in $\cup_{x \in S'} B(x, \delta_s)$, and thus in $\cup_{x \in S'} B(x, D(S))$. Now consider the ball of radius $D(S)$ centered on the last representative, x_ℓ . There is at least one additional agent that is not covered before time δ_ℓ at x_ℓ , since we chose x_ℓ as a representative and we only choose uncovered agents as representatives. Thus, we have at least $(\ell - 1)\frac{n}{k} + 1$ agents in $\cup_{x \in S} B(x, D(S))$ for every S , so X satisfies NORP. $\square$

Lemma 5.4. *The solution X of Algorithm 4 satisfies mJR.*

Proof. Let $S \subseteq N$ be a set of agents of size $\frac{n}{k}$ with radius $R(S)$. Like in a previous lemma, we will consider two cases: $R(S) < \delta_k$ and $R(S) \geq \delta_k$. Let c be the agent in N such that $B(c, R(S)) \cap S \supseteq S$. If $c \in X$, then X meets the properties of mJR immediately. We may therefore proceed assuming that $c \notin X$.

First, suppose $R(S) < \delta_k$. Then we would have grown the ball around c , and if there were $\frac{n}{k}$ uncovered agents at time $R(S)$, we would have added the representative c . Otherwise, there was some agent in S that was already covered at time $R(S)$. But since $R(S) < \delta_k$, any agent covered by time $R(S)$ can be at most $R(S)$ away from its covering representative, fulfilling mJR.

Now suppose that $R(S) \geq \delta_k$. At the end of the algorithm, all agents are covered. Again we can assume that we did not pick c or else mJR would immediately be satisfied. If something in S was covered by a set other than S_k then we are done since the representatives in S_i for $i < k$ are at most $\delta_i \leq \delta_k \leq R(S)$ away from all agents in S_i . Finally, if all agents of S are covered by S_k , then $S = S_k$. Since X contains a representative in S_k , we know that mJR is satisfied. $\square$

Theorem 5.5. *The solution X of Algorithm 4 has $\text{cost}(X) \leq 4 \cdot \text{cost}(\text{OPT})$.*

Proof. First, we will write the cost of X and apply the triangle inequality simultaneously to the distances $d(v, x_i)$ by going through all $o \in \text{OPT}$ and averaging the distances.

$$\text{cost}(X) = \sum_{x_i \in X} \sum_{v \in N} d(v, x_i) \tag{8}$$

$$\leq \frac{1}{k} \sum_{x_i \in X} \sum_{v \in N} \sum_{o \in \text{OPT}} [d(v, o) + d(o, x_i)] \tag{9}$$

$$\leq \sum_{v \in N} \sum_{o \in \text{OPT}} d(v, o) + \frac{1}{k} \sum_{x_i \in X} \sum_{v \in N} \sum_{o \in \text{OPT}} d(o, x_i) \tag{10}$$

$$\leq \text{cost}(\text{OPT}) + \frac{n}{k} \sum_{x_i \in X} \sum_{o \in \text{OPT}} d(o, x_i) \tag{11}$$

We will now focus on bounding the last term by $3 \cdot \text{cost}(\text{OPT})$. First we must pull out the last representative, x_k . Let $X' = \{x_i \in X \mid i \neq k\}$. Now we rewrite $\frac{n}{k} \sum_{x_i \in X} \sum_{o \in \text{OPT}} d(o, x_i)$ as

$$\frac{n}{k} \sum_{x_i \in X'} \sum_{o \in \text{OPT}} d(o, x_i) + \frac{n}{k} \sum_{o \in \text{OPT}} d(o, x_k). \tag{12}$$

We separate X' from x_k since in the algorithm we specifically selected x_k to minimize the distance to OPT of all agents in S_k . Because of this, we know that $\sum_{o \in \text{OPT}} d(o, x_k) \leq \sum_{o \in \text{OPT}} d(o, v)$ for any $v \in S_k$. To bound the first term above, we will apply the triangle inequality by going through the average of all agents in S_i , the set of agents covered by x_i :

$$\frac{n}{k} \sum_{x_i \in X} \sum_{o \in \text{OPT}} d(o, x_i) = \frac{n}{k} \sum_{x_i \in X'} \sum_{o \in \text{OPT}} d(o, x_i) + \frac{n}{k} \sum_{o \in \text{OPT}} d(o, x_k) \quad (13)$$

$$\leq \frac{n}{k} \frac{k}{n} \sum_{x_i \in X'} \sum_{o \in \text{OPT}} \sum_{v \in S_i} [d(v, x_i) + d(v, o)] + \sum_{v \in S_k} \sum_{o \in \text{OPT}} d(v, o) \quad (14)$$

$$= \sum_{x_i \in X'} \sum_{o \in \text{OPT}} \sum_{v \in S_i} [d(v, x_i) + d(v, o)] + \sum_{v \in S_k} \sum_{o \in \text{OPT}} d(v, o) \quad (15)$$

$$= \sum_{x_i \in X'} \sum_{o \in \text{OPT}} \sum_{v \in S_i} d(v, x_i) + \sum_{o \in \text{OPT}} \sum_{v \in N} d(v, o) \quad (16)$$

$$= \sum_{x_i \in X'} \sum_{o \in \text{OPT}} \sum_{v \in S_i} d(v, x_i) + \text{cost}(\text{OPT}) \quad (17)$$

So far, we have shown that $\text{cost}(X) \leq 2 \cdot \text{cost}(\text{OPT}) + \sum_{x_i \in X'} \sum_{o \in \text{OPT}} \sum_{v \in S_i} d(v, x_i)$. Thus, we can focus on the bounding of the term $\sum_{x_i \in X'} \sum_{o \in \text{OPT}} \sum_{v \in S_i} d(v, x_i)$. We do this by bounding $\sum_{x_i \in X'} \sum_{v \in S_i} d(v, x_i)$ for each $o \in \text{OPT}$ independently. We will show how to bound this term in the following key lemma. Note that we are bounding the distances from representatives in X' to agents S_i that they cover (so this does not include the last representative x_k), by the distances from o to *all agents*, including those covered by x_k . Once again, it is crucial that x_k is treated differently, as the following lemma does not hold if X' on the left side is replaced by X .

Lemma 5.6. *For any $o \in \text{OPT}$ we have that*

$$\sum_{x_i \in X'} \sum_{v \in S_i} d(v, x_i) \leq 2 \sum_{x_i \in X} \sum_{v \in S_i} d(v, o) = 2 \sum_{v \in N} d(v, o).$$

Proof. We will proceed via a charging argument. The terms $d(v, x_i)$ will be charged to $d(w, o)$, for some w in S_j for $j \geq i$.

First, let us define $N^- = \{v : \exists j < k, v \in S_j\}$, the set of agents covered by representatives not in S_k . We will create an injective mapping g_o from N^- to N with the condition that for any $v \in N^-$, if v is covered by x_i , then $d(v, x_i) \leq 2d(g_o(v), o)$. Formally, we introduce the following proposition:

Proposition 5.7. *For each $o \in \text{OPT}$, there exists an injective mapping $g_o : N^- \rightarrow N$ such that for any $v \in N^-$ and x_i covering v , we have that $d(v, x_i) \leq 2d(g_o(v), o)$.*

We will reserve the details on how to construct this injective function for the end of the proof. For now we will proceed assuming that the desired injective function g_o exists. We can then conclude that the following bounds for $\sum_{x_i \in X'} \sum_{v \in S_i} d(v, x_i)$ hold.

$$\begin{aligned} \sum_{x_i \in X'} \sum_{v \in S_i} d(v, x_i) &\leq 2 \left[\sum_{x_i \in X'} \sum_{v \in S_i} d(g_o(v), o) \right] \\ &\leq 2 \sum_{x_i \in X} \sum_{v \in S_i} d(v, o) \end{aligned}$$

as desired. $\square$

With the above lemma, finishing the proof of Theorem 5.5 is simple:

$$\begin{aligned} \text{cost}(X) &\leq 2 \text{cost}(\text{OPT}) + \sum_{x_i \in X'} \sum_{o \in \text{OPT}} \sum_{v \in S_i} d(v, x_i) \\ &\leq 2 \text{cost}(\text{OPT}) + 2 \sum_{o \in \text{OPT}} \sum_{v \in N} d(v, o) \\ &\leq 4 \text{cost}(\text{OPT}). \end{aligned}$$

Now to finish the main technical contribution of this proof, all we have left is to prove the existence of an injective mapping g_o given by Proposition 5.7. We do this below.

Proof of Proposition 5.7. At a high level we will build the mapping g_o by looking at each ball around x_i of radius δ_i , that consists of agents in the set S_i , in the opposite order in which they were added by the algorithm. We will consider two types of agents: “good” agents who prefer their covering center x_i by at least a factor of 2, i.e., $d(v, x_i) \leq 2d(v, o)$; and “bad” agents who do not, i.e., $d(v, x_i) > 2d(v, o)$. Then we will show that at time δ_i , when we choose x_i , there cannot be too many agents in the open ball of radius $\frac{1}{2}\delta_i$ around any $o \in \text{OPT}$. This allows us to map “good” agents to themselves, and “bad” agents to other agents in a later ball that are sufficiently far away from o . Lastly, through a counting argument, we show that we always have enough available slots for the “bad” agents to charge to.

The first step in the proof is to show that the open ball $B^\circ(o, \frac{1}{2}\delta_i)$ has few uncovered agents, as follows.

Lemma 5.8. *At time δ_i there are strictly fewer than $\frac{n}{k}$ uncovered agents in $B^\circ(o, \frac{1}{2}\delta_i)$.*

Proof. Suppose to the contrary that there exist at least $\frac{n}{k}$ uncovered agents in $B^\circ(o, \frac{1}{2}\delta_i)$. Then there exists an uncovered $v \in B^\circ(o, \frac{1}{2}\delta_i)$ such that there is a ball of radius strictly less than δ_i that contains $\frac{n}{k}$ uncovered agents (in fact this holds for all uncovered $v \in B^\circ(o, \frac{1}{2}\delta_i)$). But we should have selected the ball around v earlier in the algorithm, yielding a contradiction. Note that this holds even if v would have been the last ball, as it is not necessary that we make v a representative, only that we choose v ’s ball. $\square$

Continuing with our proof, we will give an algorithm to construct the injective mapping from $N^- \rightarrow N$ with our desired property. To do so, we will consider the representatives in reverse order. Now we will define two sets of agents with respect to each $x_i \in X'$ and to the $o \in \text{OPT}$ that we are considering.

$$S_{(i,o)}^- = \{v \in S_i \mid 2d(v, o) < d(v, x_i)\}$$

and

$$S_{(i,o)}^+ = \{v \in S_i \mid d(v, x_i) \leq 2d(v, o)\}.$$

We can then immediately assign $g_o(v) = v$ for all $v \in S_{(i,o)}^+$ since we see that $d(v, x_i) \leq 2d(v, o)$ so the desired property that $d(v, x_i) \leq 2d(v, o) = 2d(g_o(v), o)$ holds for these $v \in S_{(i,o)}^+$. These are the “good” agents described earlier in the high level overview.

Now consider the agents $v \in S_{(i,o)}^-$. Notice that these agents in $S_{(i,o)}^-$ are less than δ_i from x_i . We then can conclude that since $2d(v, o) < d(v, x_i) \leq \delta_i$, we then have that $d(v, o) < \frac{1}{2}\delta_i$, so these agents are included in the open ball $B^\circ(o, \frac{1}{2}\delta_i)$ we considered earlier. The set of agents in $S_{(i,o)}^-$ are the aforementioned “bad” agents that we described earlier. We will need to charge these agents to

later uncovered agents. We will now give an iterative algorithm that can construct the remaining values of g_o , starting by forming values $g_o(v)$ for $v \in S_{k-1}$ and proceeding in reverse order.

While constructing g_o we will also form useful sets A_i , which we initialize to $A_{k-1} = S_k$. Each set A_i will have the following three properties:

1. $|A_i| = \frac{n}{k}$;
2. $A_i \subseteq S_{i+1} \cup \dots \cup S_k$;
3. For all $j > i$ there does not exist a $v \in S_j$ such that $g_o(v) \in A_i$.

Note that these properties hold for $A_{k-1} = S_k$ since $|S_k| = \frac{n}{k}$, and the third property holds since g_o is only defined on the domain N^- .

Now consider building the mapping g_o for S_i , for some $1 \leq i \leq k-1$. We will proceed inductively, so suppose that so far we have computed the mapping for all $v \in S_j$ for $j > i$, and we have constructed the set A_i satisfying the properties above.

As described previously, for $v \in S_{(i,o)}^+$, we simply set $g_o(v) = v$. Let $m = |S_{(i,o)}^-|$: this is how many agents of S_i we still need to assign using g_o . Now consider the uncovered agents at time δ_i in Algorithm 4, right before we chose x_i . Both S_i and A_i are uncovered, by property 2 of A_i above, since at time δ_i only $S_1 \cup \dots \cup S_{i-1}$ have been covered. As we argued above, $S_{(i,o)}^- \subseteq B^\circ(o, \frac{1}{2}\delta_i)$, but Lemma 5.8 states that there can only be strictly fewer than $\frac{n}{k}$ uncovered agents in this ball. Thus, since m of the agents in S_i are also in $B^\circ(o, \frac{1}{2}\delta_i)$, there must be at most $\frac{n}{k} - 1 - m$ agents $v \in A_i$ inside this ball, and thus there can be no fewer than $\frac{n}{k} - (\frac{n}{k} - 1 - m) = m + 1$ agents $v \in A_i$ with $d(v, o) \geq \frac{1}{2}\delta_i$. We can then map the agents of $S_{(i,o)}^-$ each to a unique agent in A_i with $d(v, o) \geq \frac{1}{2}\delta_i$. Now g_o satisfies the desired properties since for $v \in S_{(i,o)}^-$ we have that,

$$d(v, x_i) \leq \delta_i \leq 2d(g_o(v), o).$$

We then have that the desired property holds for all $v \in S_{(i,o)}^-$. Note that due to the third property of A_i , this mapping is injective, since nothing else has been chosen to map to agents in A_i until now.

Next, we will show how to construct A_{i-1} to maintain the invariant properties stated above. Let $g_o(S_i)$ be the set of $\frac{n}{k}$ elements which we mapped the agents $v \in S_i$ onto. These are either in S_i (for $v \in S_{(i,o)}^+$) or in A_i (for $v \in S_{(i,o)}^-$). Now set $A_{i-1} = (A_i \cup S_i) \setminus g_o(S_i)$. To complete the proof of this proposition, we need to show that A_{i-1} still has the three invariant properties defined above. $|A_{i-1}| = \frac{n}{k}$, since S_i and A_i are disjoint sets of size $\frac{n}{k}$. The second property clearly holds, since $A_{i-1} \subseteq S_i \cup A_i \subseteq S_k \cup S_{k-1} \cup \dots \cup S_i$. The third property holds as well, since we made sure that A_{i-1} consists only of agents which g_o has not mapped anything onto yet, and we have already finished forming the mapping for all $S_i \cup \dots \cup S_{k-1}$.

This completes the proof of the proposition, since the above process has been shown inductively to construct an injective mapping with the desired property. $\square$

Now that we have shown that g_o exists, that completes our proof of the theorem, and we obtain an approximation bound of 4, as desired. $\square$

6 Approximate Fairness

In previous sections, we gave 2-approximation algorithms which achieve PRF or NORP separately, as well as a 4-approximation algorithm which achieves PRF, NORP, and mJR simultaneously.

In this section, we will investigate what happens if we relax the fairness notions and consider approximate notions of fairness. We first show that allowing a 2-PRF solution, we can find an 2-approximate solution in polynomial time. We also give a version of Algorithm 4 that computes an approximately PRF solution (instead of 1-PRF), but with a lower cost.

Our first approximately fair algorithm starts by growing n balls, each ball centered on a unique agent in N . Then, when a ball centered on an agent $v \in N$ has $\frac{n}{k}$ uncovered agents, we greedily cover all agents in the ball (from the center out). Lastly, we choose the agent $v' \in S_i$, that is, the set of agents covered by v , that minimizes the distance to all the points in OPT. Notice that we never stop growing a ball and can select the same ball multiple times, but the covered sets are disjoint. Thus, we will never select the same agent twice to be a representative (since an agent can never be covered twice). This algorithm can be seen as a polynomial-time approximate version of the Algorithm 1 that finds sets that are at most 2 times the diameter of the smallest set. Meaning that although we have a solution that is two approximation to OPT we suffer by finding a solution that is only at worst a two approximate PRF solution. However, we do not suffer the same loss for mJR and find a 2 approximation to mJR (as opposed to a 4 approximation as implied by Lemma 4.5).

Algorithm 5 Computes a 2-PRF, 2-mJR, 2-approx to OPT in poly-time

```

1:  $Z \leftarrow N$  ▷ Uncovered agents
2:  $\delta \leftarrow 0$ 
3:  $i \leftarrow 1$ 
4: while  $Z \neq \emptyset$  do
5:   while  $\exists x \in N : |B(x, \delta) \cap Z| \geq \frac{n}{k}$  do ▷ Select a ball centered at
6:      $S_i \leftarrow \arg \min_{\substack{T \subseteq B(x, \delta) \cap Z \\ |T| = \frac{n}{k}}} \sum_{v \in T} d(v, x)$ 
7:      $Z \leftarrow Z \setminus S_i$ 
8:      $s_i \leftarrow x$ 
9:      $\delta_i \leftarrow \delta$ 
10:     $i \leftarrow i + 1$ 
11:  end while
12:  smoothly increase  $\delta$ 
13: end while
14:  $X \leftarrow \{\arg \min_{v \in S_i} \sum_{o \in \text{OPT}} d(v, o) \mid i \in [k]\}$ 

```

Theorem 6.1. *Let X be the solution returned by Algorithm 5. Then X satisfies 2-PRF and 2-mJR, and $\text{cost}(X) \leq 2 \text{cost}(\text{OPT})$.*

Lemma 6.2. *Let X be the solution returned by Algorithm 5. Then X is 2-PRF.*

Proof. Let $S \subseteq N$ be a set of agents of size $\ell \frac{n}{k}$ with diameter $D(S)$. After time $D(S)$ there must be strictly fewer than $\frac{n}{k}$ uncovered agents in S . Since each representative covers at most $\frac{n}{k}$ agents and no agent is covered by the same representative twice we have that there must be ℓ distinct representatives covering something in S by time $D(S)$.

Recall that the algorithm works by choosing a ball with center s_i and radius δ_i containing a set S_i , and then choosing a representative x_i to be the closest element of S_i to OPT. Let x_i be a representative covering some $v \in S$ by time $D(S)$, and s_i be the center of the set S_i . Then we have that $d(v, x_i) \leq d(v, s_i) + d(s_i, x_i) \leq 2\delta_i \leq 2D(S)$. The last inequality is because x_i was placed before time $D(S)$ and thus $\delta_i \leq D(S)$. Thus there are at least ℓ representatives in $\cup_{v \in S} B(v, 2 \cdot D(S))$. $\square$

Lemma 6.3. *Let X be the set of representatives returned by Algorithm 5. Then the clustering X has cost $\text{cost}(X) \leq 2 \cdot \text{cost}(\text{OPT})$.*

Proof. Follows from Proposition 4.3. □

Lemma 6.4. *Let X be the solution returned by Algorithm 5. Then X is 2-mJR.*

Proof. Let $S \subseteq N$ be a subset of N of size $\frac{n}{k}$ and let S have radius $R(S)$. Let $c \in N$ be the agent in N such that $S \subseteq B(c, R(S))$ (recall that c need not be in S). Note that the set S must have something covered after time $R(S)$ or else there is a ball of radius $R(S)$ centered at c that contains S that would cover all of S .

Now consider how far an agent v can be from some x_i that covers it by time $R(S)$. Let s_i be the center of the ball with radius $\delta_i \leq R(S)$ chosen when forming the set S_i . Then, $d(v, x_i) \leq d(v, s_i) + d(s_i, x_i) \leq 2\delta_i \leq 2R(S)$. Thus, there must be a representative at most $2R(S)$ away from an agent in S . □

Our final algorithm is parameterized by $\alpha \geq 1$ which is the PRF approximation factor. The algorithm works identically to Algorithm 4 with the exception that it maintains a special agent whose ball is grown α times faster than the rest. The special agent is the one who is closest to $o_1 = \arg \min_{o \in \text{OPT}} \sum_{v \in N} d(v, o)$. By growing the balls near o_1 α times faster we can improve the approximation bounds to $2 + \frac{2}{\alpha}$, with the PRF approximation growing to α and all while maintaining NORP.

Algorithm 6 $(2 + \frac{2}{\alpha}, \alpha, 1)$ algorithm

```

1:  $Z \leftarrow N$ 
2:  $\delta \leftarrow 0$ 
3:  $i \leftarrow 1$ 
4:  $a \leftarrow \arg \min_{v \in Z} d(v, o_1)$  ▷ Break ties in the arg min arbitrarily
5: while  $Z \neq \emptyset$  do
6:   while  $\exists x \in Z : (|B(x, \delta) \cap Z| \geq \frac{n}{k}) \vee ((x = a) \wedge (|B(x, \alpha \cdot \delta) \cap Z| \geq \frac{n}{k}))$  do
7:      $S_i \leftarrow \arg \min_{\substack{T \subseteq Z \\ |T| = \frac{n}{k}}} \sum_{v \in T} d(v, x)$ 
8:      $Z \leftarrow Z \setminus S_i$ 
9:      $s_i \leftarrow x$ 
10:     $\delta_i \leftarrow \delta$ 
11:     $i \leftarrow i + 1$ 
12:     $a \leftarrow \arg \min_{v \in Z} d(v, o_1)$ 
13:   end while
14:   smoothly increase  $\delta$ 
15: end while
16:  $X \leftarrow \{s_i | i < k\} \cup \{\arg \min_{v \in S_k} d(v, o_1)\}$ 

```

Lemma 6.5. *Let X be the solution returned by Algorithm 6. Then X is α -PRF.*

Proof. The proof is identical to the proof of Lemma 5.2, with the exception that the balls now are at most $\alpha\delta$ in radius. □

Lemma 6.6. *Let X be the solution returned by Algorithm 6. Then X is α -mJR.*

Proof. The proof is identical to the proof of Lemma 5.4, with the exception that the balls now are at most $\alpha\delta$ in radius and we consider the two cases $R(S) \leq \alpha\delta_k$ and $R(S) > \alpha\delta_k$. $\square$

Lemma 6.7. *Let X be the solution returned by Algorithm 6. Then X satisfies NORP.*

Proof. The proof is almost identical to the proof of Lemma 5.3 since we will never select a representative that is within each of the previous representative's ball. The only difference is that instead of setting x_s to be the second-to-last representative selected, we set x_s to be the representative with the largest radius of the set it covers (either δ_i or $\alpha\delta_i$ depending on how x_i was selected). $\square$

Theorem 6.8. *Algorithm 6 outputs a solution X with $\text{cost}(X) \leq (2 + \frac{2}{\alpha}) \text{cost}(\text{OPT})$.*

Proof. First, we will write the cost of X and apply the triangle inequality to the distances $d(v, x_i)$ by going through $o_1 \in \text{OPT}$ such that $o_1 \in \arg \min_{o \in N} \sum_{v \in N} d(v, o)$.

$$\text{cost}(X) = \sum_{x_i \in X} \sum_{v \in N} d(v, x_i) \quad (18)$$

$$\leq \sum_{x_i \in X} \sum_{v \in N} [d(v, o_1) + d(o_1, x_i)] \quad (19)$$

$$\leq k \sum_{v \in N} d(v, o_1) + n \sum_{x_i \in X} d(o_1, x_i) \quad (20)$$

$$\leq \sum_{v \in N} \sum_{o \in \text{OPT}} d(v, o) + n \sum_{x_i \in X} d(o_1, x_i) \quad (21)$$

$$\leq \text{cost}(\text{OPT}) + n \sum_{x_i \in X} d(o_1, x_i) \quad (22)$$

We will now focus on bounding the last term by $(1 + \frac{2}{\alpha}) \cdot \text{cost}(\text{OPT})$. First we must pull out the the last representative, x_k . Let $X' = \{x_i \in X \mid i \neq k\}$. Now we rewrite $n \sum_{x_i \in X} d(o_1, x_i)$ as

$$n \sum_{x_i \in X'} d(o_1, x_i) + nd(o_1, x_k). \quad (23)$$

We separate X' from x_k since in the algorithm we specifically selected x_k to minimize the distance to o_1 of all agents in S_k . Because of this, we know that $d(o_1, x_k) \leq d(o_1, v)$ for any $v \in S_k$.

Once again we need to split X' into two sets, those who were selected when chosen as a in the algorithm (excluding x_k) and those who were not. We call the two aforementioned sets Y and Y' (resp). Y is the set of representatives that were chosen with radius $\alpha\delta$ and were the closest uncovered agent to o_1 at that time. Y' is the rest of representatives chosen in the “normal” way, because the ball of radius δ centered at that agent contained $\frac{n}{k}$ uncovered agents. In the inequalities below, we use the triangle inequality, and the fact that for any $x_i \in Y$, $v \in S_i$, it must be that

$d(x_i, o_1) \leq d(v, o_1)$, since x_i was the closest to o_1 at the time it was chosen.

$$n \sum_{x_i \in X'} d(o_1, x_i) + nd(o_1, x_k) = n \sum_{x_i \in Y} d(o_1, x_i) + n \sum_{x_i \in Y'} d(o_1, x_i) + nd(o_1, x_k) \quad (24)$$

$$\leq n \sum_{x_i \in Y} d(o_1, x_i) + n \frac{k}{n} \sum_{x_i \in Y'} \sum_{v \in S_i} [d(v, x_i) + d(v, o_1)] + k \sum_{v \in S_k} d(v, o_1) \quad (25)$$

$$= n \sum_{x_i \in Y} d(o_1, x_i) + k \sum_{x_i \in Y'} \sum_{v \in S_i} [d(v, x_i) + d(v, o_1)] + k \sum_{v \in S_k} d(v, o_1) \quad (26)$$

$$\leq k \sum_{x_i \in Y} \sum_{v \in S_i} d(v, o_1) + k \sum_{x_i \in Y'} \sum_{v \in S_i} [d(v, x_i) + d(v, o_1)] + k \sum_{v \in S_k} d(v, o_1) \quad (27)$$

$$\leq \text{cost}(\text{OPT}) + k \sum_{x_i \in Y'} \sum_{v \in S_i} d(v, x_i) \quad (28)$$

Let us now focus on the bounding of the term $k \sum_{x_i \in Y'} \sum_{v \in S_i} d(v, x_i)$. We will show how to bound this term in the following key lemma. Note that we are bounding the distances from representatives in Y' to agents S_i that they cover (so this does not include the last representative x_k), by the distances from o_1 to *all agents*, including those covered by x_k .

Lemma 6.9. *We have that*

$$\sum_{x_i \in Y'} \sum_{v \in S_i} d(v, x_i) \leq \frac{2}{\alpha} \sum_{x_i \in Y' \cup \{x_k\}} \sum_{v \in S_i} d(v, o_1).$$

Proof. We will proceed via a charging argument. The terms $d(v, x_i)$ will be charged to $d(w, o_1)$, for some w in S_j for $j \geq i$ and $x_j \in Y' \cup \{x_k\}$.

First, let us define $N^- = \cup_{x_j \in Y'} S_j$, which is the set of agents covered by representatives other than the representative a at the time it was selected (and not representative x_k). Likewise we define $N^+ = N^- \cup \{S_k\}$. We will create an injective mapping g from N^- to N^+ with the condition that for any $v \in N^-$, if v is covered by x_i , then $d(v, x_i) \leq \frac{2}{\alpha} d(g(v), o_1)$. Formally, we introduce the following proposition defining the mapping we desire to construct.

Proposition 6.10. *There exists an injective mapping $g : N^- \rightarrow N^+$ such that for any $v \in N^-$ and x_i covering v , we have that $d(v, x_i) \leq \frac{2}{\alpha} d(g(v), o_1)$.*

We will reserve the details on how to construct this injective function for the end of the proof. For now we will proceed assuming that the desired injective function g exists. We can then conclude that the following bounds for $\sum_{x_i \in Y'} \sum_{v \in S_i} d(v, x_i)$ hold.

$$\begin{aligned} \sum_{x_i \in Y'} \sum_{v \in S_i} d(v, x_i) &\leq \frac{2}{\alpha} \left[\sum_{x_i \in Y'} \sum_{v \in S_i} d(g(v), o_1) \right] \\ &\leq \frac{2}{\alpha} \sum_{x_i \in Y' \cup \{x_k\}} \sum_{v \in S_i} d(v, o_1) \end{aligned}$$

as desired. □

With the above lemma, finishing the proof of Theorem 5.5 is simple:

$$\begin{aligned}
\text{cost}(X) &\leq 2 \text{cost}(\text{OPT}) + k \sum_{x_i \in Y'} \sum_{v \in S_i} d(v, x_i) \\
&\leq 2 \text{cost}(\text{OPT}) + k \frac{2}{\alpha} \sum_{x_i \in Y' \cup \{x_k\}} \sum_{v \in S_i} d(v, o_1) \\
&\leq 2 \text{cost}(\text{OPT}) + k \frac{2}{\alpha} \sum_{x_i \in X} \sum_{v \in S_i} d(v, o_1) \\
&= 2 \text{cost}(\text{OPT}) + k \frac{2}{\alpha} \sum_{v \in N} d(v, o_1) \\
&\leq 2 \text{cost}(\text{OPT}) + \frac{2}{\alpha} \sum_{v \in N} \sum_{o \in \text{OPT}} d(v, o) \\
&\leq \left(2 + \frac{2}{\alpha}\right) \text{cost}(\text{OPT}).
\end{aligned}$$

Now to finish the main technical contribution of this proof, all we have left is to prove the existence of an injective mapping g given by Proposition 6.10. We do this below.

Proof of Proposition 6.10. At a high level we will build the mapping g by looking at each ball around x_i of radius δ_i , that consists of agents in the set S_i , in the opposite order in which they were added by the algorithm. We will consider two types of agents: “good” agents who prefer their covering center x_i by at least a factor of $2/\alpha$, i.e., $d(v, x_i) \leq \frac{2}{\alpha} d(v, o_1)$; and “bad” agents who do not, i.e., $d(v, x_i) > \frac{2}{\alpha} d(v, o_1)$. Then we will show that at time δ_i , when we choose x_i , there cannot be too many agents in the open ball of radius $\frac{1}{2}\alpha\delta_i$ around o_1 . This allows us to map “good” agents to themselves, and “bad” agents to other agents in a later ball that are sufficiently far away from o_1 . Lastly, through a counting argument, we show that we always have enough available slots for the “bad” agents to charge to.

The first step in the proof is to show that the open ball $B^\circ(o_1, \frac{1}{2}\alpha\delta_i)$ has few uncovered agents, as follows.

Lemma 6.11. *At time δ_i there are strictly fewer than $\frac{n}{k}$ uncovered agents in $B^\circ(o_1, \frac{1}{2}\alpha\delta_i)$.*

Proof. Suppose to the contrary that there exist at least $\frac{n}{k}$ uncovered agents in $B^\circ(o_1, \frac{1}{2}\alpha\delta_i)$ at time δ_i . Then the uncovered agent a closest to o_1 will have all of these agents contained in the ball $B^\circ(a, \alpha\delta_i)$, and so these agents will become covered by our algorithm before time δ_i , giving a contradiction. $\square$

Continuing with our proof, we will give an algorithm to construct the injective mapping from $N^- \rightarrow N^+$ with our desired property. To do so, we will consider the representatives in reverse order. Now we will define two sets of agents with respect to each $x_i \in Y'$.

$$S_i^- = \left\{ v \in S_i \mid \frac{2}{\alpha} d(v, o_1) < d(v, x_i) \right\}$$

and

$$S_i^+ = \left\{ v \in S_i \mid d(v, x_i) \leq \frac{2}{\alpha} d(v, o_1) \right\}.$$

We can then immediately assign $g(v) = v$ for all $v \in S_i^+$ since we see that $d(v, x_i) \leq \frac{2}{\alpha}d(v, o)$ so the desired property that $d(v, x_i) \leq \frac{2}{\alpha}d(v, o) = \frac{2}{\alpha}d(g(v), o)$ holds for these $v \in S_i^+$. These are the “good” agents described earlier in the high level overview.

Now consider the agents $v \in S_i^-$. Notice that these agents in $S_{(i,o)}^-$ are less than δ_i from x_i . We then can conclude that since $\frac{2}{\alpha}d(v, o_1) < d(v, x_i) \leq \delta_i$, we then have that $d(v, o_1) < \alpha \frac{1}{2}\delta_i$, so these agents are included in the open ball $B^\circ(o, \frac{1}{2}\delta_i)$ we considered earlier. The set of agents in S_i^- are the aforementioned “bad” agents that we described earlier. We will need to charge these agents to later uncovered agents. We will now give an iterative algorithm that can construct the remaining values of g , starting by forming values $g(v)$ for $v \in S_{k-1}$ and proceeding in reverse order. First, we will relabel the sets S_i and the representatives x_i . Let $k' = |Y'| + 1$, which is the number of representatives not equal to a when selected, as well as the last representative. Let y_i be the i th representative added in Y' and S'_i be its respective covering set, with $y_{k'} = x_k$. Notice that g then takes agents in $N^- = S'_1 \cup S'_2 \cup \dots \cup S'_{k'-1}$ and maps them to agents in $N^+ = S'_1 \cup S'_2 \cup \dots \cup S'_{k'}$.

While constructing g we will also form useful sets A_i , which we initialize to $A_{k'-1} = S'_{k'}$. Each set A_i will have the following three properties:

1. $|A_i| = \frac{n}{k}$;
2. $A_i \subseteq S'_{i+1} \cup \dots \cup S'_{k'}$;
3. For all $j > i$ there does not exist a $v \in S'_j$ such that $g(v) \in A_i$.

Note that these properties hold for $A_{k'-1} = S'_{k'}$ since $|S'_{k'}| = \frac{n}{k}$, the second property holds since all sets S'_i are disjoint, and the third property holds since the domain of g does not include $S'_{k'}$.

Now consider building the mapping g for S'_i , for some $1 \leq i \leq k' - 1$. We will proceed inductively, so suppose that so far we have computed the mapping for all $v \in S'_j$ for $j > i$, and we have constructed the set A_i satisfying the properties above.

As described previously, for $v \in S_i^+$, we simply set $g(v) = v$. Let $m = |S_i^-|$: this is how many agents of S'_i we still need to assign using g . Now consider the uncovered agents at time δ_i in Algorithm 6, right before we choose y_i . Both S'_i and A_i are uncovered at this time, by property 2 of A_i above. As we argued above, $S_i^- \subseteq B^\circ(o_1, \frac{1}{2}\delta_i)$, but Lemma 6.11 states that there can only be strictly fewer than $\frac{n}{k}$ uncovered agents in this ball. Thus, since m of the agents in S'_i are also in $B^\circ(o, \frac{1}{2}\alpha\delta_i)$, there must be at most $\frac{n}{k} - 1 - m$ agents $v \in A_i$ inside this ball, and thus there can be no fewer than $\frac{n}{k} - (\frac{n}{k} - 1 - m) = m + 1$ agents $v \in A_i$ with $d(v, o_1) \geq \frac{\alpha}{2}\delta_i$. We can then map the agents of S_i^- each to a unique agent in A_i with $d(v, o) \geq \frac{1}{2}\alpha\delta_i$. Now g satisfies the desired properties since for $v \in S_i^-$ we have that,

$$d(v, y_i) \leq \delta_i \leq \frac{2}{\alpha}d(g(v), o_1).$$

We then have that the desired property holds for all $v \in S_i^-$. Note that due to the third property of A_i , this mapping is injective, since nothing else has been chosen to map to agents in A_i until now.

Next, we will show how to construct A_{i-1} to maintain the invariant properties stated above. Let $g(S'_i)$ be the set of $\frac{n}{k}$ elements which we mapped the agents $v \in S'_i$ onto. These are either in S'_i (for $v \in S_i^+$) or in A_i (for $v \in S_i^-$). Now set $A_{i-1} = (A_i \cup S'_i) \setminus g(S'_i)$. To complete the proof of this proposition, we need to show that A_{i-1} still has the three invariant properties defined above. $|A_{i-1}| = \frac{n}{k}$, since S_i and A_i are disjoint sets of size $\frac{n}{k}$. The second property clearly holds, since $A_{i-1} \subseteq S'_i \cup A_i \subseteq S'_{k'} \cup S'_{k'-1} \cup \dots \cup S'_i$. The third property holds as well, since we made sure

that A_{i-1} consists only of agents which g has not mapped anything onto yet, and we have already finished forming the mapping for all $S'_i \cup \dots \cup S'_{k'-1}$.

This completes the proof of the proposition, since the above process has been shown inductively to construct an injective mapping with the desired property. $\square$

Now that we have shown that g exists, that completes our proof of the theorem, and we obtain an approximation bound of $2 + \frac{2}{\alpha}$, as desired. $\square$

7 Conclusion

In this manuscript, we showed that many known fairness axioms are compatible with good solutions according to the sum-cost objective, i.e., that we can find simultaneously low-cost and fair solutions. We also defined NORP, a new fairness axiom which focuses on avoiding over-representation. We proved that NORP is compatible with many other fairness axioms as well, and that we can find fair and NORP solutions which approximate the optimum for the sum objective within a constant factor.

There are still many open questions which remain. Can a solution always be found that is PRF and NORP while also better than a four approximation to OPT? Can a 2-approximate and PRF solution always be found in polynomial time for a general metric space? Finally, what other objectives are compatible with NORP and fairness axioms, i.e., could we form constant approximations to these objectives while satisfying fairness constraints? One promising class of such objectives is the ℓ -centrum objectives Slater (1978); Tamir (2001); Peeters (1998); Han et al. (2023): can similar results be shown for that class, perhaps with the approximation ratio depending on ℓ ?

References

- M. R. Ackermann, J. Blömer, and C. Sohler. Clustering for metric and nonmetric distance measures. *ACM Transactions on Algorithms (TALG)*, 6(4):1–26, 2010.
- A. Aggarwal, H. Imai, N. Katoh, and S. Suri. Finding k points with minimum spanning trees and related problems. In *Proceedings of the Fifth Annual Symposium on Computational Geometry (SoCG)*, pages 283–291, 1989.
- E. Anastasiadis and A. Deligkas. Heterogeneous facility location games. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS)*, pages 623–631, 2018.
- H. Aziz and B. E. Lee. The expanding approvals rule: improving proportional representation and monotonicity. *Social Choice and Welfare*, 54(1):1–45, 2020.
- H. Aziz and B. E. Lee. Proportionally representative participatory budgeting with ordinal preferences. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, volume 35, pages 5110–5118, 2021.
- H. Aziz, B. Lee, S. Morota Chu, and J. Vollen. Proportionally representative clustering. In *Proceedings of the 20th Conference on Web and Internet Economics (WINE)*, 2024.

- M. Bateni, V. Cohen-Addad, A. Epasto, and S. Lattanzi. A scalable algorithm for individually fair k-means clustering. In *International Conference on Artificial Intelligence and Statistics*, pages 3151–3159. PMLR, 2024.
- M. Brill and J. Peters. Robust and verifiable proportionality axioms for multiwinner voting. *Proceedings of the 24th ACM Conference on Economics and Computation*, page 301, 2023.
- I. Caragiannis, N. Shah, and A. A. Voudouris. The metric distortion of multiwinner voting. *Artificial Intelligence*, 313:103802, 2022.
- H. Chan, A. Filos-Ratsikas, B. Li, M. Li, and C. Wang. Mechanism design for facility location problems: A survey. In *30th International Joint Conference on Artificial Intelligence*, pages 4356–4365. International Joint Conferences on Artificial Intelligence Organization, 2021.
- C.-S. Chang, W. Liao, Y.-S. Chen, and L.-H. Liou. A mathematical theory for clustering in metric spaces. *IEEE Transactions on Network Science and Engineering*, 3(1):2–16, 2016.
- X. Chen, B. Fain, L. Lyu, and K. Munagala. Proportionally fair clustering. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 1032–1041, 2019.
- A. Chhabra, K. Masalkovaitė, and P. Mohapatra. An overview of fairness in clustering. *IEEE Access*, 9:130698–130720, 2021.
- A. Deligkas, A. Filos-Ratsikas, and A. A. Voudouris. Heterogeneous facility location with limited resources. *Games and Economic Behavior*, 139:200–215, 2023.
- A. Deligkas, M. Lotfi, and A. A. Voudouris. Agent-constrained truthful facility location games. *Journal of Combinatorial Optimization*, 49(2):24, 2025.
- S. Ebadian and E. Micha. Boosting sortition via proportional representation. *AAMAS-25: Proc. of 24th International Conference on Autonomous Agents and Multiagent Systems*, 2025.
- J. M. Enelow and M. J. Hinich. *Advances in the spatial theory of voting*. Cambridge University Press, 1990.
- D. Eppstein and J. Erickson. Iterated nearest neighbors and finding minimal polytopes. *Discrete & Computational Geometry*, 11:321–350, 1994.
- A. Goel, R. Hulett, and A. K. Krishnaswamy. Relating metric distortion and fairness of social choice rules. In *Proceedings of the 13th Workshop on Economics of Networks, Systems and Computation (NetEcon)*, 2018.
- Y. Han, C. Jerrett, and E. Anshelevich. Optimizing multiple simultaneous objectives for voting and facility location. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2023.
- A. K. Jain. Data clustering: 50 years beyond k-means. *Pattern recognition letters*, 31(8):651–666, 2010.
- A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: a review. *ACM computing surveys (CSUR)*, 31(3):264–323, 1999.
- M. Jiang. On finding a large number of 3d points with a small diameter. *Discrete Applied Mathematics*, 155(17):2355–2361, 2007.

- C. Jung, S. Kannan, and N. Lutz. Service in your neighborhood: Fairness in center location. *Foundations of Responsible Computing (FORC)*, 2020.
- Y. Kalayci, D. Kempe, and V. Kher. Proportional representation in metric spaces and low-distortion committee selection. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, volume 38, pages 9815–9823, 2024.
- P. Kanellopoulos, A. A. Voudouris, and R. Zhang. On discrete truthful heterogeneous two-facility location. *SIAM Journal on Discrete Mathematics*, 37(2):779–799, 2023.
- P. Kanellopoulos, A. A. Voudouris, and R. Zhang. Truthful two-facility location with candidate locations. *Theoretical Computer Science*, 1024:114913, 2025.
- L. Kellerhals and J. Peters. Proportional fairness in clustering: A social choice perspective. *Advances in Neural Information Processing Systems (NeurIPS)*, 37:11299–11317, 2024.
- B. Li, L. Li, A. Sun, C. Wang, and Y. Wang. Approximate group fairness for clustering. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 6381–6391. PMLR, 2021.
- M. Lotfi and A. A. Voudouris. On truthful constrained heterogeneous facility location with max-variant cost. *Operations Research Letters*, 52:107060, 2024.
- S. Mahabadi and A. Vakilian. Individual fairness for k -clustering. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 6586–6596. PMLR, 2020.
- S. Merrill and B. Grofman. *A unified theory of voting: Directional and proximity spatial models*. Cambridge University Press, 1999.
- E. Micha and N. Shah. Proportionally fair clustering revisited. In *47th International Colloquium on Automata, Languages, and Programming (ICALP)*, 2020.
- M. Negahbani and D. Chakrabarty. Better algorithms for individually fair k -clustering. *Advances in Neural Information Processing Systems (NeurIPS)*, 34:13340–13351, 2021.
- P. H. Peeters. Some new algorithms for location problems on networks. *European Journal of Operational Research*, 104(2):299–309, 1998.
- F. Pukelsheim. *Proportional representation: Apportionment Methods and Their Applications*. Springer, 2017.
- P. Serafino and C. Ventre. Heterogeneous facility location without money on the line. In *Proceedings of the 21st European Conference on Artificial Intelligence (ECAI)*, pages 807–812, 2014.
- P. J. Slater. Centers to centroids in graphs. *Journal of graph theory*, 2(3):209–222, 1978.
- F. Sun, M. Wang, D. Wang, and X. Wang. Optimizing social image search with multiple criteria: Relevance, diversity, and typicality. *Neurocomputing*, 95:40–47, 2012.
- A. Tamir. The k -centrum multi-facility location problem. *Discrete Applied Mathematics*, 109(3): 293–307, 2001.
- A. Vakilian and M. Yalciner. Improved approximation algorithms for individually fair clustering. In *International conference on artificial intelligence and statistics*, pages 8758–8779. PMLR, 2022.

- X. Xu, B. Li, M. Li, and L. Duan. Two-facility location games with minimum distance requirement. *Journal of Artificial Intelligence Research*, 70:719–756, 2021.
- Q. Zhao, W. Liu, Q. Nong, and Q. Fang. Constrained heterogeneous two-facility location games with sum-variant. *Journal of Combinatorial Optimization*, 47(4):65, 2024.

A Extending our results when k does not divide n

Recall that in the body of the paper, we considered only instances where k divides n , and only proved our results about solution costs and fairness under that assumption. We now justify that this assumption is reasonable, and show that all our results in this work extend to the more general case where k may not divide n . Rather than form new algorithms for this more general case, we instead give a reduction from the general case where k may not divide n to the special case when it does. Our reduction will show that if an algorithm provides low-cost, or fair, or NORP solutions for the special case, then it can be transformed into an algorithm for the general case with the same properties.

Before we present our reduction, we note that it only works for algorithms with the *point-NORP* property, defined below. We will need to consider balls of radius 0, which are defined as $B(v) = \{v' \in N \mid d(v, v') = 0\}$, which are the agents from the set N that lie at the same point in the metric space. Note that since multiple agents are allowed to be at the same point (i.e., our space is a pseudo-metric), balls of radius 0 may have many agents in them.

Definition A.1 (Point-NORP). Let X be a set of representatives of size k . For any set $S \subseteq X$ of size $|S| \geq \ell$ such that $S \subseteq B(v)$ for some v (i.e., the maximum distance between representatives in S is 0) the following holds: $|B(v)| > (\ell - 1)\frac{n}{k}$.

Point-NORP is simply a weaker version of NORP applied only to sets S of diameter 0, as it says that any coalition of diameter 0 cannot have too many representatives. We say that an algorithm satisfies Point-NORP if it always produces committees which satisfy Point-NORP. Before we proceed with our reduction showing how to extend results to the case when k does not divide n , we first prove that all the algorithms presented in this paper satisfy this property. This is trivially true for all algorithms which satisfy NORP (since Point-NORP is a special case of NORP for sets of diameter zero), so we just need to show this still holds for Algorithms 1 and 5.

Proposition A.1. *Algorithm 1 satisfies Point-NORP if ties are broken so that the set with the most agents closest to OPT is selected first.*

Proof. Suppose we have ℓ representatives in some $B(v)$. We will argue that at most one representative x_i in $B(v)$ contains agents in S_i outside of $B(v)$.

So see why this is true, first note that when there is a set of $\frac{n}{k}$ agents of diameter zero, then this set will be chosen first by the algorithm. Thus, at the very beginning of the algorithm (when $\delta = 0$), these sets will be selected and covered, with each set S_i contained entirely inside $B(v)$. This continues until less than $\frac{n}{k}$ uncovered agents are left in $B(v)$.

We now claim that at most one more representative will be chosen in $B(v)$. Suppose to the contrary that x_i and x_j in $B(v)$ are both chosen, and without loss of generality let $i < j$. Then at the time of choosing x_i , there exist less than $\frac{n}{k}$ uncovered agents v with $d(v, x_i) = 0$, so our tie breaking rule would have kicked in and we would have selected those agents as part of the set S_i . Thus we have that at most one representative x_i in $B(v)$ contains agents in S_i outside of $B(v)$.

Since at least $\ell - 1$ of the representatives in $B(v)$ have their entire sets S_i contained in $B(v)$, and since each S_i contains exactly $\frac{n}{k}$ agents, then together with the ℓ 'th representative which is also in $B(v)$, it must be that $|B(v)| > (\ell - 1)\frac{n}{k}$. $\square$

Proposition A.2. *Algorithm 5 satisfies Point-NORP if we greedily cover $B(x_i)$ first.*

Proof. The argument is mostly the same as the previous proposition. By similar reasoning, all representatives in $B(v)$ which are chosen before $B(v)$ had less than $\frac{n}{k}$ uncovered agents will have their entire set S_i in $B(v)$.

We change the algorithm slightly when $\delta > 0$. Instead, we will let

$$x_i = \arg \min_{v \in B(x, \delta) \cap Z} \sum_{o \in \text{OPT}} d(v, o),$$

and then we cover the agents

$$S_i = (B(x_i) \cap Z) \cup \arg \min_{T \subseteq B(x, \delta) \cap Z, |T| = \frac{n}{k} - |B(x_i) \cap Z|} \sum_{v \in T} d(v, x)$$

at each iteration.

Suppose that we have β representatives in $B(v)$, then since we greedily form S_i out from this point, there must be at least $(\beta - 1)\frac{n}{k} + 1 \geq (\beta - 1)k + 1$ agents in $B(v)$, by exactly the same argument as in the previous proposition.

Notice that after this change we still return a solution that is a 2-approximation to OPT, since the representative $x_i \in S_i$ still satisfies $\sum_{o \in \text{OPT}} d(o, x_i) \leq \sum_{o \in \text{OPT}} d(o, v)$ for all $v \in S_i$.

Likewise, the proof of PRF never used the fact that we covered *greedily* out of the center agent, only that we choose S_i to be *some* set of $\frac{n}{k}$ agents in the ball $B(x, \delta)$. $\square$

Now that we know all the algorithms in our work satisfy Point-NORP, we proceed with the reduction showing how to extend our results to the case when k does not divide n . Suppose we have an Algorithm A which satisfies Point-NORP, and works when k divides n . Given a set of agents N of size n , and an integer k which does *not* divide n , we form a new instance with the set of agents $N' = N \times \{1, 2, \dots, k\}$, and k remaining the same. In other words, for every agent $v \in N$, we now have k identical agents in N' , which we'll denote as $v_1, v_2, \dots, v_k$. The distances remain the same, i.e., for any $v, w \in N$, and any i and j , we have that $d(v_i, w_j) = d(v, w)$. Note that the number of agents in N' is exactly $n' = kn$, so k divides the number of agents in the new instance. We can therefore run algorithm A on the new instance, and obtain a committee $X' \subseteq N'$ of size k . Finally, we set $X \subseteq N$ to be a committee in the original instance as follows. For every $v \in N$, consider $B(v)$ in N and $B(v_1)$ in N' . For every representative in X' located in $B(v_1)$, we select an arbitrary agent in $B(v)$ and add it to X . This completes our reduction, as we used an algorithm which works on the case when k divides n to form a committee X for the more general setting when k does not divide n .

To show that our reduction above is well-defined, we need to argue that it is actually possible to form a committee X as above, i.e., that there are enough agents in $B(v)$ to select. In other words, we need to show that $|B(v)| \geq |X' \cap B(v_1)|$. Due to our construction, we know that $|B(v_1)| = k|B(v)|$. Since Algorithm A satisfies Point-NORP, we also know that the number of representatives in X' chosen in $B(v_1)$ is strictly less than $|B(v_1)|\frac{k}{|N'|} + 1$. Thus,

$$|X' \cap B(v_1)| \leq |B(v_1)|\frac{k}{|N'|} = k|B(v)|\frac{k}{kn} = \frac{k}{n}|B(v)| \leq |B(v)|,$$

and so our reduction is well-defined. Also we clearly select at most k representatives. For intuition, our reduction essentially simulates a fractional version of A where instead we cover $\frac{1}{k}$ fractions of agents. We now prove that if algorithm A behaves nicely for the case when k divides n , then the above reduction also has the same properties for arbitrary k and n .

Theorem A.3. *Let X be the output of the reduction above, X' be the intermediate instance created in the reduction, and OPT' be the optimal solution of N' . We then have the following:*

1. *If $\text{cost}_{N'}(X') \leq \alpha \text{cost}_{N'}(\text{OPT}')$ then $\text{cost}_N(X) \leq \alpha \text{cost}_N(\text{OPT})$.*
2. *If X' is an α -approximate PRF solution, then X is α -approximate PRF.*
3. *If X' is an α -approximate mJR solution, then X is an α -approximate mJR.*
4. *If X' is an α -approximate NORP solution, then X is α -approximate NORP.*

Proof.

1. Notice that the cost is given by

$$\text{cost}_N(X) = \sum_{v \in N} \sum_{x \in X} d(v, x) \quad (29)$$

$$= \frac{k}{k} \sum_{v \in N} \sum_{x \in X} d(v, x) \quad (30)$$

$$= \frac{1}{k} \sum_{v \in N'} \sum_{x \in X'} d(v, x) \quad (31)$$

$$= \frac{1}{k} \text{cost}_{N'}(X') \quad (32)$$

$$\leq \frac{1}{k} \alpha \text{cost}_{N'}(\text{OPT}') \quad (33)$$

Now let us compare the costs of OPT' and of OPT . Let $O \subseteq N'$ be the solution obtained by taking each $o \in \text{OPT}$, and adding the agent $o_1 \in N'$ to O . In other words, O is exactly OPT mapped to N' . Then, $\text{cost}_{N'}(O) = k \text{cost}_N(\text{OPT})$. But since OPT' is the optimal solution for N' , we know that $\text{cost}_{N'}(\text{OPT}') \leq \text{cost}_{N'}(O)$. Thus, we have that

$$\text{cost}_N(X) \leq \frac{1}{k} \alpha \text{cost}_{N'}(\text{OPT}') \quad (34)$$

$$\leq \frac{1}{k} \alpha \text{cost}_{N'}(O) \quad (35)$$

$$= \alpha \text{cost}_N(\text{OPT}) \quad (36)$$

as desired.

2. Suppose X is a solution given by the above reduction. Let $S \subseteq N$ be a set of $\ell \frac{n}{k}$ agents. Then the set $S' \subseteq N'$ corresponding to duplicating each agent k times has $\ell \frac{nk}{k} = \ell n = \ell \frac{n'}{k}$ agents. Thus, balls $\cup_{v \in S'} B(v, \alpha D(S'))$ have at least ℓ agents in X' . Notice that $D(S') = D(S)$ and so $\cup_{v \in S} B(v, \alpha D(S))$ has ℓ representatives as well.
3. Similarly to the above argument, we consider $S \subseteq N$ with size $\frac{n}{k}$. We see that the set $S' \subseteq N'$ has $\frac{n'}{k}$ agents, so $\cup_{v \in S'} B(v, \alpha R(S'))$ has a representative in X' and thus $\cup_{v \in S} B(v, \alpha R(S))$ has a representative in X .

4. Let $S \subseteq X$ be a set of ℓ representatives. Then the set $\cup_{x \in S'} B(x, \alpha D(S))$ has $\ell \frac{n'}{k}$ agents in N' and thus there are at least $\ell \frac{n}{k}$ agents in N in $\cup_{x \in S} B(x, \alpha D(S))$.

□

B Other fairness concepts

We also remark that mPJR as defined by Kellerhals and Peters (2024) is incompatible with NORP; See Figure 6. We believe that this example shows that mPJR is too strong of a property. While an mPJR solution always exists, this example shows that it may not be very “fair” or robust for our applications, since it concentrates all the representatives in a very small region.

For completeness, we also include the statement of the following theorem, stating that approximate mJR implies many other fairness concepts. Kellerhals and Peters (2024) already proved these results for exact mJR; we omit the proof of this theorem since it is a direct and simple extension of the proofs in Kellerhals and Peters (2024).

Theorem B.1 (Theorem 5, (Kellerhals and Peters, 2024)). *Let X be an outcome satisfying α -mJR. Then it also satisfies $(1 + \sqrt{1 + \alpha})$ -proportional fairness, $(1 + \alpha)$ -individual fairness, and is in the $(\gamma, \frac{\gamma(\alpha+1)}{\gamma-1})$ -transferable core for any $\gamma > 1$.*

C Hardness of computing fair and low-cost solutions

Theorem C.1. *For any $\gamma \in (0, 1)$ it is co-NP-Hard to determine if there exists a PRF or mJR solution that is better than a $1 + \gamma - O\left(\frac{1}{\sqrt{n}}\right)$ approximation to OPT. In particular, it is co-NP-Hard to determine if a PRF or mJR solution that is a $2 - O\left(\frac{1}{\sqrt{n}}\right)$ approximation exists.*

Proof. We will show this through a reduction from CLIQUE. Note that for any constant $\gamma \in (0, 1)$ the problem of determining if a clique in a graph $G = (V, E)$ with size $|V| \cdot \gamma$ exists is NP-Hard.

Let $\beta = \gamma - \frac{1}{|V|} \in (0, 1)$ for $|V|$ sufficiently large. Construct the instance as follows:

1. Create $|V|$ copies of G ;
2. All edges in G have distance $1 - \epsilon$;
3. Create $\frac{1}{\beta} \cdot |V|$ agents with a distance zero from each other and 1 from all other agents;
4. All other distances are the metric completion;
5. Let $k = |V| \frac{1}{\beta}$.

Now we will show that there exists a clique of size at least $\gamma|V| = \beta|V| + 1$ if and only if there does not exist a PRF or mJR solution that is a $(1 + \gamma - O\left(\frac{1}{|V|}\right))$ -approximation to OPT. First observe that for $|V| > 1$ the solution that minimizes the sum objective places all representatives at the $|V| \frac{1}{\beta}$ center agents and has cost $\frac{|V|^3}{\beta}$. Note that the number of agents in this instance is $n = |V| \left(|V| + \frac{1}{\beta}\right)$ and the quota for a coalition is $\frac{n}{k} = \frac{(|V|^2 + \frac{|V|}{\beta})}{|V| \frac{1}{\beta}} = \beta|V| + 1$. First, we will show the forward direction.

Suppose that there exists a clique of size $\gamma|V| = \frac{n}{k}$. Then to satisfy PRF or mJR, for each copy of G we need to place one of the representatives in the graph instead of on the center agents,

since the clique has diameter $1 - \epsilon$. Thus, the cost is at least $|V|^2(k - |V|) + 2|V|^2(|V| - 1) + |V|(|V| - 1)(1 - \epsilon) = |V|^2\left(\frac{|V|}{\beta} - |V|\right) + 2|V|^2(|V| - 1) + |V|(|V| - 1)(1 - \epsilon)$ (ignoring the cost of the k central agents) while the cost of OPT is $|V|^2k = |V|^3\frac{1}{\beta}$. We then have that the approximation ratio is given by:

$$\begin{aligned} \frac{|V|^2\left(\frac{|V|}{\beta} - |V|\right) + 2n^2(|V| - 1) + |V|(|V| - 1)(1 - \epsilon)}{|V|^3\frac{1}{\beta}} &= 1 + \beta - \frac{\beta}{|V|} - \frac{\beta}{|V|^2} - \frac{\beta\epsilon}{|V|} + \frac{\beta\epsilon}{|V|^2} \\ &\geq 1 + \beta - O\left(\frac{1}{|V|}\right) \\ &= 1 + \gamma - O\left(\frac{1}{|V|}\right) \\ &= 1 + \gamma - O\left(\frac{1}{\sqrt{n}}\right). \end{aligned}$$

Now for the reverse direction, suppose that there exists a PRF solution that strictly better than a $(1 + \gamma - O(\frac{1}{\sqrt{n}}))$ - approximation, then by before we know it cannot be the case that each duplicate of the original graph G has a representative, thus there cannot be a clique of size $\gamma|V|$. This means that for any $\gamma \in (0, 1)$ determining if a solution exists that is $1 + \gamma - O(\frac{1}{\sqrt{n}})$ approximate to OPT and is PRF or mJR solution is computationally harder than to determine if a clique of size $\gamma|V|$ exists. $\square$

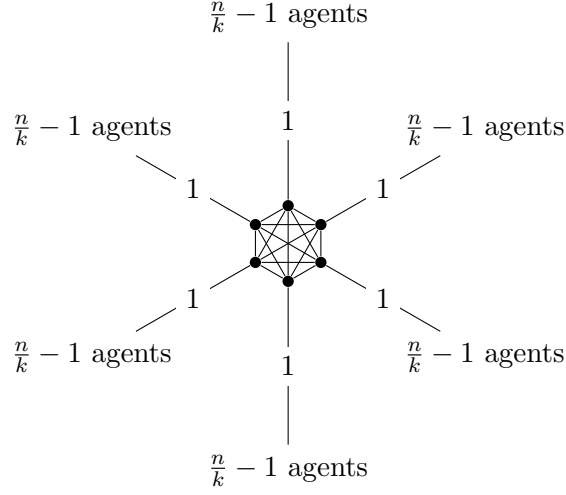


Figure 3: Example where Algorithm 1 gives a solution that is not NORP. $k = 6$ and there are 6 central agents that are all a distance ϵ from each other; each central agent has a spoke with $\frac{n}{k} - 1$ agents a distance 1 away, all of which are also distance ϵ from each other. Previous fairness axioms would allow representatives to be placed in the inner clique. This is unfair since the k agents in the middle have a large number of representatives within a distance of ϵ , yet only 6 agents. A fairer solution (that satisfies all usual fairness properties) would be to place the representatives on the outermost 5 points with $\frac{n}{k} - 1$ agents and one representative in the middle.

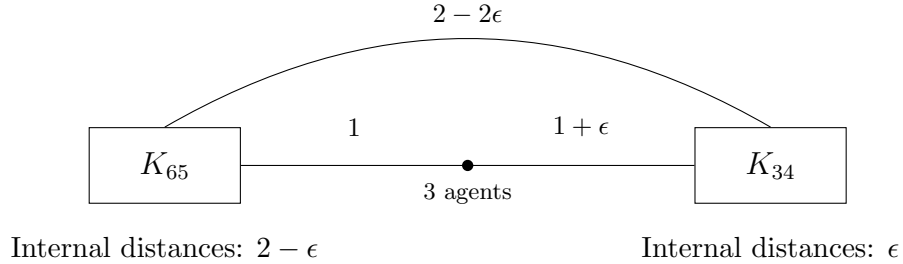


Figure 4: Example where Algorithm 3 does not satisfy PRF. There are 102 agents in total, broken into three groups. Three agents in the center on top of each other, 65 on the left and 34 on the right. We use K_r to denote a complete graph with r nodes. The 65 on the left are $2 - 2\epsilon$ from each other and the 34 on the right are ϵ from each other. All agents in K_{65} and K_{34} are $2 - 2\epsilon$ from each other. The three agents in the middle are 1 and $1 + \epsilon$ away from the two groups. The three middle agents are the optimal solution for $k = 3$ and $\frac{n}{k} = 34$. The algorithm instead picks one central representative and two on the left. Then the group on the right does not have a representative within distance ϵ , which is the diameter of that set.

$n - 2$ agents ————— 1 ————— 1 agent ——— ϵ ——— 1 agent

Figure 5: In the above example we have $n - 2$ agents on the left hand side each distance 0 from each other and a single agent each in the middle and on the right. Standard Truncated Greedy Capture (along with Greedy Capture and Expanding Approvals if we place extra representatives on the left) would first place $k - 1$ representatives on the left, then the last representative in the middle. The optimal solution, however, places all representatives on the left. The cost of the solution returned by the greedy capture algorithm is at least $(k - 1)(2 - \epsilon) + (n - 2)(1 - \epsilon)$. The cost of the optimal solution is $k(2 - \epsilon)$. As $n \rightarrow \infty$ the approximation factor becomes arbitrarily large.

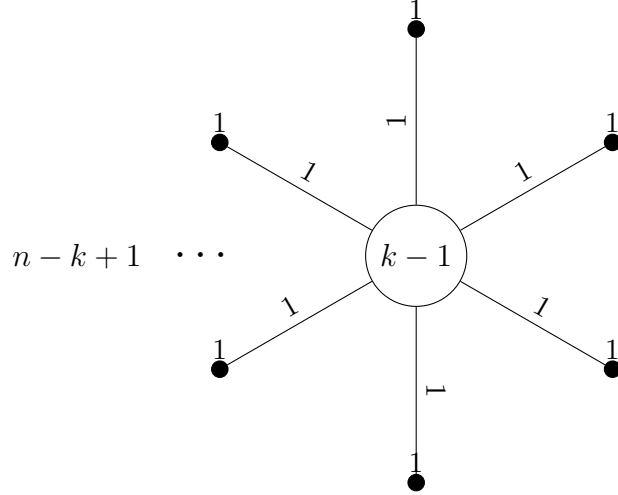


Figure 6: A instance where mPJR is incompatible with NORP for any approximation factor. Assume $k \leq \sqrt{\frac{n}{2}}$. There are $n - k + 1$ agents on the outside spokes, each $1 + \epsilon$ from each other, and $k - 1$ agents in the middle. Assume there are $\ell \leq k$ representatives on the outer spokes. Then there exists a set $S \subseteq N$ consisting of $n - 2k + 1 > (k - 1)\frac{n}{k}$ agents all on the spokes that are not representatives. This group has $k - 1$ agents all at most distance 1 from all agents in S , therefore it does not satisfy mPJR unless at least $k - 1$ representatives are placed in the center. But placing $k - 1$ representatives in the center clearly violates NORP.