When a Nearby Center Is Not Enough: Service-Aware Individual Fairness for Min-Sum-Radii Clustering

Anonymous authors
Paper under double-blind review

Abstract

Neighborhood-radius individual fairness typically bounds each point's distance to the selected center set, not to the center that actually serves it. For standard pointwise objectives, nearest-center reassignment enforces fair service without increasing cost. We formalize nearest-assignment stability—coordinatewise no-farther reassignment cannot increase cost—as a sufficient transfer condition. In contrast, minimum sum of radii (MSR) depends on the realized partition and admits a linear gap: on a tree with $n = 2k$ points and standard n/k-neighbor radii, the exact selected-set-fair and service-aware optima are 2 and k, respectively. To our knowledge, we give the first study of service-aware individually fair MSR, requiring every retained point's assigned center to meet its personalized service bound. To handle this non-mergeable constraint, we introduce minimum-priority capture, which constructs centers and assignments jointly. For arbitrary priorities, the algorithm is deterministic and fixed-parameter tractable (FPT) in $k + z$ for fixed ε. Whenever exact λ-service is feasible, it returns at most k centers and at most z outliers, with cost at most $(2 + \varepsilon)$ times the exact-service optimum and service-fairness factor 2. We establish a matching service lower bound: for every fixed $\delta \in (0, 1)$, a budget-preserving deterministic FPT algorithm guaranteeing factor $2 - \delta$ on exact-service-feasible inputs would imply $\mathrm{W}[2] = \mathrm{FPT}$, even with standard neighborhood radii, no outliers, and unrestricted cost. The framework extends to fixed polynomial-time computable monotone symmetric norms of cluster radii and to min-sum-diameters.

1 Introduction

Clustering with individual fairness asks whether every point receives service at a scale appropriate to its local neighborhood. For minimum sum of radii (MSR), the cost depends on both the selected centers and the realized partition: the objective sums each cluster's maximum assignment distance. We study the distinction between access to a nearby center and the service actually delivered.

Neighborhood-radius fairness assigns each point x a personalized scale $\rho(x)$, commonly its $\lceil n/k \rceil$-neighborhood radius, and bounds its distance to a selected center set C by $d(x, C) \leq \lambda\rho(x)$ (Jung et al., 2020; Mahabadi & Vakilian, 2020). For standard pointwise objectives such as k-median, k-means, and k-center, nearest-center reassignment preserves this certificate and cannot increase cost. Subsequent work improves guarantees and scalability for these objectives (Negahbani & Chakrabarty, 2021; Vakilian & Yalçıner, 2022; Bateni et al., 2024; Ebbens et al., 2025; Wu et al., 2026), with priority and outlier variants also studied (Bajpai et al., 2021; Han et al., 2023; Maity et al., 2025). Assignment-level service itself has a close precedent in Liu et al. (2026b), who explicitly assign clients within local-density radii in facility location and give a bicriteria individually fair k-median result. For these pointwise costs, the assigned-center and nearby-center views remain compatible through nearest reassignment.

MSR breaks this compatibility. Moving a point to a nearer center can enlarge the receiving cluster's radius without reducing the original cluster's radius. Optimal MSR partitions need not be Voronoi with respect to their centers (Bagheri Nezhad et al., 2025). Consequently, a fair center set need not certify low-cost fair service. We make this obstruction quantitative using canonical neighbor-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

hood radii: on a tree with $n = 2k$ points, the exact selected-set-fair optimum is 2, while the exact service-aware optimum is k. Nearest reassignment of the constructed cost-2 solution costs $k + 1$ (Lemma 4). This is a gap in cost, not feasibility: a selected-set certificate still guarantees that some fair assignment exists.

Existing constrained-MSR results do not directly resolve this gap. MSR and min-sum-diameters (MSD) have a long approximation history (Charikar & Panigrahy, 2001; Gibson et al., 2012). Fair-MSR work studies group-composition constraints (Drexler et al., 2023; Carta et al., 2024; Bagheri Nezhad et al., 2025), while Gadekar (2026a) studies group representation among centers with outliers and general radius norms. Neither constraint is a point-specific service bound. Generic FPT approximations cover mergeable constraints (Bandyapadhyay & Chen, 2025; Gadekar, 2026b). Grandoni et al. (2026) give a $(1 + \varepsilon)$ FPT scheme for unconstrained MSR and a deterministic $(2 + \varepsilon)$ guarantee for polynomial-time verifiable mergeable constraints via their MSD scheme. Further work treats outliers, colors, capacities, and parameterized complexity (Buchem et al., 2024; Banerjee et al., 2025; Chen et al., 2024; Kumar et al., 2026; Liu et al., 2026a; Filtser & Gadekar, 2026). Our designated-service constraint is non-mergeable (Lemma 5), so these frameworks do not directly apply.

We therefore study *service-aware individually fair MSR*: every retained point x must be assigned to a designated center $\sigma(x)$ satisfying $d(x, \sigma(x)) \leq \lambda\rho(x)$. The algorithmic challenge is to recenter clusters without losing these personalized service bounds. Radius/profile guessing and factor-two recentering are standard FPT MSR tools (Chen et al., 2024; Bandyapadhyay & Chen, 2025), but radius control alone does not ensure point-specific service. We develop *minimum-priority capture*, whose key property ensures whole-cluster capture under simultaneous radius and factor-two service bounds. If a minimum-priority remaining point p belongs to a cluster of an optimal λ-service-fair solution, exact service through that cluster's center gives every remaining member x the bound $d(p, x) \leq \lambda(\rho(p) + \rho(x)) \leq 2\lambda\rho(x)$. Under a dominating radius profile, a ball centered at p and truncated by these service bounds therefore captures the entire remaining optimal cluster, even if its optimal center is no longer among the remaining points. Any additional captured points also satisfy both bounds. This property lets the search construct centers and assignments simultaneously, with outlier branching providing budget control. The result addresses designated service rather than improving the unconstrained MSR cost frontier.

1.1 Our contributions

A transfer condition and a linear assignment gap. We formalize *nearest-assignment stability* as a sufficient condition for cost-preserving transfer from selected-set fairness to service fairness (Lemma 3). Standard pointwise objectives satisfy this condition, but MSR admits a $\Theta(k)$ gap between the two optima under canonical radii. Our tree construction has $n = 2k$ points and exact optima $\mathrm{OPT}_1^{\mathrm{set}} = 2$ and $\mathrm{OPT}_1^{\mathrm{serv}} = k$, showing that the separation persists even when the service-aware optimum is free to choose a different center set (Lemma 4). This is a cost gap rather than a feasibility gap: a nearby-center certificate guarantees a fair assignment, but not a low-cost one. We also prove non-mergeability of designated service (Section 3), explaining why existing FPT frameworks for mergeable constraints do not directly apply.

An FPT algorithm and a matching lower bound on the service-fairness factor. Whenever exact λ-service is feasible, minimum-priority capture yields a deterministic algorithm with cost at most $(2 + \varepsilon)$ times the exact-service optimum $\mathrm{OPT}_\lambda^{\mathrm{serv}}$, service-fairness factor 2, and no center or outlier budget violations. The key capture property combines a minimum-priority pivot with an upward-rounded radius profile to capture all remaining members of a corresponding optimal cluster while satisfying both radius and service bounds. It remains valid even after that cluster's optimal center has left the remaining set, allowing centers and assignments to be constructed jointly rather than repaired after center selection. For arbitrary priorities and up to z outliers, its running time is $(k/\varepsilon)^{O(k)}(k + 1)^{k+z}n^{O(1)}$, hence FPT in $k + z$ for fixed $\varepsilon \in (0, 1]$ (Theorem 11). For every fixed $\delta \in (0, 1)$, improving this service-fairness factor to $2 - \delta$ with a budget-preserving deterministic FPT algorithm would imply $\mathrm{W}[2] = \mathrm{FPT}$, already with canonical radii, no outliers, exact-service-feasible inputs, and unrestricted output cost (Theorem 13). Tightness concerns the service-fairness factor, not the cost factor.

Extensions and supporting validation. The algorithmic framework extends to every fixed polynomial-time computable monotone symmetric norm of cluster radii and, with a modified capture radius, to service-aware min-sum-diameters (Theorems 14 and 15). Both extensions preserve the $(2+\varepsilon)$ cost bound relative to their respective exact-service optima, factor-two service, and the original center and outlier budgets. The corresponding proofs exploit coordinatewise radius or diameter control, allowing the same capture framework to handle these objectives. The factor-two lower bound on service stretch applies to both extensions as well. Exact diagnostics on subsamples of three standard datasets and exhaustive tests on tiny metrics check the modeling gap and theorem-to-code consistency. These checks compare exact selected-set and service-aware optima and exercise the complete profile search under arbitrary priorities, norm objectives, and diameter objectives. They are supporting validation, not a separate empirical contribution.

Section 2 fixes the model and notation. Section 3 establishes the assignment gap. Sections 4–6 give the algorithm, service lower bound, and extensions. Section 7 reports the computational diagnostics.

2 PRELIMINARIES

Let (P, d) be a finite metric space with $|P| = n$ and centers restricted to P. The input contains integers $k \geq 1$ and $z \geq 0$ satisfying $k \leq n - z$, priorities $\rho : P \to \mathbb{R}_{\geq 0}$, and a fairness parameter $\lambda \geq 1$.

Definition 1 (Service-aware individually fair MSR)**.** A solution consists of an outlier set $O \subseteq P$ with $|O| \leq z$ and nonempty centered clusters $(c_i, X_i)_{i=1}^m$, where $m \leq k$, the sets X_i partition $P \setminus O$, and $c_i \in X_i$. Write $C = \{c_1, \ldots, c_m\}$ and $\sigma(x) = c_i$ for $x \in X_i$. Its cluster radii and objective are $R_i = \max_{x \in X_i} d(x, c_i)$, $\text{cost} = \sum_{i=1}^m R_i$. The solution is λ-service-fair if every retained point is served by its designated cluster center within its personalized bound:

$$d(x, c_i) \leq \lambda \rho(x) \qquad \text{for every } x \in X_i. \tag{1}$$

Let $\mathrm{OPT}_\lambda^{\mathrm{serv}}$ denote the minimum cost of such a solution, with $\mathrm{OPT}_\lambda^{\mathrm{serv}} = +\infty$ if none exists.

A service-fairness factor (service stretch) of $t \geq 1$ means $d(x, \sigma(x)) \leq t\lambda\rho(x)$ for every retained point. Exact λ-service has $t = 1$. Our cost guarantees are relative to $\mathrm{OPT}_\lambda^{\mathrm{serv}}$ even when $t > 1$. For comparison, a solution is λ-selected-set-fair if $d(x, C) \leq \lambda\rho(x)$ for every $x \in P \setminus O$, where $d(x, C) = \min_{c \in C} d(x, c)$. The nearby center need not serve x, and cost is still computed from the designated partition. Let $\mathrm{OPT}_\lambda^{\mathrm{set}}$ be the minimum cost, taking $+\infty$ if infeasible. The feasible center/outlier choices coincide with those of service fairness: nearest reassignment enforces (1), but may raise MSR cost. Thus $\mathrm{OPT}_\lambda^{\mathrm{set}} \leq \mathrm{OPT}_\lambda^{\mathrm{serv}}$, with a possible gap in cost rather than feasibility.

Although our algorithm accepts arbitrary input priorities, a common outlier-aware neighborhood choice (Han et al., 2023) is fixed before selecting O:

$$q = \left\lceil \frac{n - z}{k} \right\rceil, \qquad \rho_{k,z}(x) = \min\{r \geq 0 : |B(x, r) \cap P| \geq q\}, \tag{2}$$

where $B(x, r) = \{y \in P : d(x, y) \leq r\}$, so the ball count includes x. The rank uses the input budget z, and priorities remain fixed after outlier removal. Other work keeps an n/k-based rank even with outliers (Maity et al., 2025). Because our algorithm treats ρ as input, either convention and application-specific scales can be supplied directly.

We use at most k clusters. Requiring exactly k gives the same optimum when $n - z \geq k$. While fewer than k clusters remain, split a non-center point from a cluster of size at least two into a zero-radius singleton. Cost does not increase, and service fairness is preserved. A formal statement covering norm objectives appears in Appendix A.

3 THE ASSIGNMENT GAP

In this section, we identify when selected-set fairness transfers to service fairness without increasing cost, establish a linear assignment gap for MSR, and prove the non-mergeability of designated service.

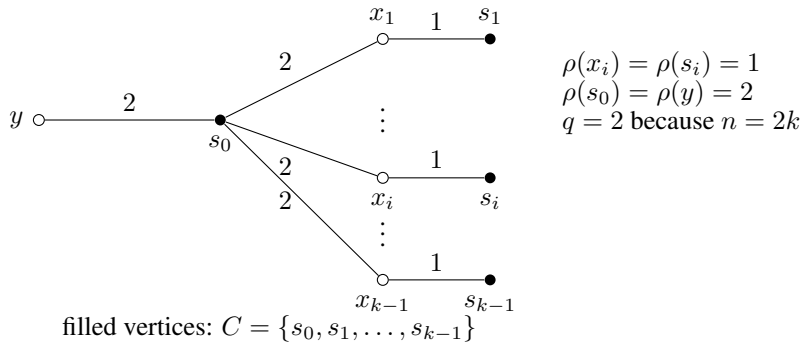


Figure 1: The tree used in Lemma 4: representative arms are shown. The filled set C certifies exact selected-set fairness, but the cost-2 MSR assignment sends every x_i to s_0 rather than to its nearby witness s_i.

3.1 When selected-set fairness transfers to service fairness

We first give a sufficient condition under which selected-set fairness transfers to service fairness without increasing the objective value.

Definition 2 (Nearest-assignment stability)**.** A clustering objective $F(C, \sigma)$ is nearest-assignment stable if, for every retained set S, fixed center set $C \subseteq S$, and assignments $\sigma, \tau : S \to C$ satisfying $\sigma(c) = \tau(c) = c$ for every $c \in C$, $d(x, \tau(x)) \leq d(x, \sigma(x))\ \forall x \in S$ implies $F(C, \tau) \leq F(C, \sigma)$.

Lemma 3. *For any nearest-assignment-stable objective and every $\lambda \geq 1$, any λ-selected-set-fair solution can be converted, with the same centers and outliers and without increasing cost, into a λ-service-fair solution by nearest-center reassignment. Consequently, when both formulations use the same budgets and priorities, their optimum values are equal.*

Proof. Fix the retained set S, centers C, and the assignment σ of a selected-set-fair solution. Assign each $x \in S$ to a nearest center $\tau(x) \in C$, choosing $\tau(c) = c$ for every $c \in C$. Then $d(x, \tau(x)) = d(x, C) \leq \lambda\rho(x)$ for every retained point, so τ is service-fair. Moreover $d(x, \tau(x)) \leq d(x, \sigma(x))$ for every x, and nearest-assignment stability therefore implies $F(C, \tau) \leq F(C, \sigma)$. Conversely, every service-fair solution is selected-set-fair. Applying the two directions to respective optima proves equality. □

Nearest-assignment stability is a sufficient transfer condition, not a converse characterization. Every objective of the form $G((d(x, \sigma(x)))_{x \in S})$ with coordinatewise nondecreasing G is stable, including weighted k-median, k-means, pointwise ℓ_p objectives, ordered monotone pointwise costs, and k-center. MSR is not stable because it groups assignment distances by designated center, takes one maximum per group, and then sums those maxima. The next lemma gives both a fixed-center witness and a linear gap between optima, even when the centers may be reselected.

3.2 A linear assignment gap for MSR

We next show that this distinction can be quantitatively large by establishing a linear assignment gap for MSR.

Lemma 4. *For every integer $k \geq 2$, there is a tree metric on $n = 2k$ points with $z = 0$ and canonical priorities $\rho_{k,0}$ such that $\mathrm{OPT}_1^{\mathrm{set}} = 2$ and $\mathrm{OPT}_1^{\mathrm{serv}} = k$. Hence the selected-set optimum and the service-aware individually fair MSR optimum differ by a factor $k/2$. For the displayed selected-set-optimal center set, nearest-center assignment costs $k + 1$.*

Proof. Let $m = k - 1$. Create vertices s_0, y and pairs x_i, s_i for $i \in [m]$, with tree edges s_0y, s_0x_i of length 2 and x_is_i of length 1 (Figure 1). Since $n = 2k$, the canonical rank is $q = 2$, so $\rho(x_i) = \rho(s_i) = 1$ and $\rho(s_0) = \rho(y) = 2$.

For selected-set fairness, choose $C = \{s_0, s_1, \ldots, s_m\}$. Every point has a center within its priority. Assign $s_0, y, x_1, \ldots, x_m$ to s_0 and keep each s_i singleton, at cost 2. Every radius-1 ball contains at most two points. Since all distances are integral, a solution of cost < 2 has at most one two-point cluster and otherwise only singletons. It covers at most $k + 1 < 2k$ points, a contradiction. Hence $\mathrm{OPT}_1^{\mathrm{set}} = 2$.

Exact service forces a center in each arm pair $\{x_i, s_i\}$, whose members have no outside point within priority 1, and a center in $\{y, s_0\}$ to serve y. These k disjoint requirements exhaust the budget, so each pair contains exactly one center and every arm contributes radius at least 1. If s_0 is selected, serving y adds radius 2. Otherwise y is selected, and s_0 must be served by y or a selected x_i, adding radius 2 at y or forcing an arm radius of at least 2. Thus the cost is at least $m + 1 = k$.

This bound is attained with centers $y, x_1, \ldots, x_m$: assign each s_i to x_i, assign s_0 to x_1, and keep y singleton. One arm has radius 2 and the other $m - 1$ arms radius 1. Finally, nearest reassignment for the displayed set C sends x_i to s_i and y to s_0, giving cost $m + 2 = k + 1$. □

3.3 Non-mergeability of designated service

Finally, we show that the distinction also has an algorithmic consequence: designated service is not, in general, mergeable. Following the constrained-MSR literature (Bandyapadhyay & Chen, 2025), call a cluster property mergeable if any two disjoint feasible clusters have a feasible union. Here the input priorities remain fixed, and the union may choose any member as its designated center.

Lemma 5. *For every fixed $\lambda \geq 1$, the cluster property "there exists a designated center in the cluster that serves every member within λ times its priority radius" is not mergeable.*

Proof. Take four points in two pairs, with within-pair distance $a = 1/(3\lambda)$ and all cross-pair distances 1. This is a metric. For $k = 2$ and $z = 0$, the canonical rank is $q = 2$, so every point has priority $a > 0$. Each pair is λ-service-fair when centered at either member. Their union is not: any designated center serves the opposite pair at distance $1 > \lambda a = 1/3$. Thus non-mergeability holds even for positive canonical priorities. □

4 The Minimum-Priority Capture Algorithm

In this section, we develop minimum-priority capture to construct centers and assignments jointly, addressing the obstructions in Section 3. We enumerate upward-rounded radius profiles, describe the capture search, and establish its approximation and running-time guarantees.

4.1 Upward-rounded radius profiles

We first construct a list containing a profile that dominates the optimal radii with only a small increase in their sum. Every cluster radius lies in $\mathcal{D} = \{d(u, v) : u, v \in P\}$. Profiles are ordered, with coordinates serving as labels for the optimal clusters. We round upward because $2\widehat{R}_j$ will be a hard capture threshold.

Lemma 6. *For every $\eta \in (0, 1]$, one can enumerate a list $\mathcal{L}_\eta$ of at most $|\mathcal{D}|\big(\lceil k/\eta \rceil + 1\big)^k + 1 \leq n^2\big(\lceil k/\eta \rceil + 1\big)^k + 1$ ordered vectors with the following property. For every $R^* \in \mathcal{D}^k$, some $\widehat{R} \in \mathcal{L}_\eta$ satisfies $R_i^* \leq \widehat{R}_i$ for all i and $\sum_i \widehat{R}_i \leq (1 + \eta) \sum_i R_i^*$. The list is generated in polynomial time per vector.*

Proof. Include the all-zero vector. For each positive $\Delta \in \mathcal{D}$, set $\delta = \eta\Delta/k$ and enumerate $\{0, \delta, 2\delta, \ldots, \lceil k/\eta \rceil \delta\}^k$. For nonzero R^*, choose $\Delta = \max_i R_i^*$ and set $\widehat{R}_i = \delta\lceil R_i^*/\delta \rceil$. Because $R_i^* \leq \Delta$, this vector is enumerated. Moreover, $R_i^* \leq \widehat{R}_i < R_i^* + \delta$, so

$$\sum_i \widehat{R}_i < \sum_i R_i^* + k\delta = \sum_i R_i^* + \eta\Delta \leq (1 + \eta) \sum_i R_i^*.$$

□

4.2 MINIMUM-PRIORITY CAPTURE SEARCH

We next describe the minimum-priority capture search for a fixed profile $\widehat{R} = (\widehat{R}_1, \ldots, \widehat{R}_k)$, keeping ρ fixed throughout. A state has unprocessed points $U \subseteq P$, unused labels $J \subseteq [k]$, and b declared outliers. Each path keeps its own partial solution. For $p \in U$, $j \in J$, define

$$A(p, j, U) = \{x \in U : d(p, x) \le 2\widehat{R}_j,\ d(p, x) \le 2\lambda\rho(x)\}. \tag{3}$$

The pivot satisfies both guards, so every capture is nonempty. Under a dominating profile, a minimum-priority pivot in an optimal cluster captures all its remaining members using the matching label, even if that cluster's center is no longer in U (Lemma 7). Any additional captured points satisfy both the radius and service guards.

For $\varepsilon \in (0, 1]$, set $\eta = \varepsilon/2$ and run Algorithm 1 from $(P, [k], 0)$ for every $\widehat{R} \in \mathcal{L}_\eta$. Explore all branches and return a candidate of minimum actual MSR cost, breaking ties deterministically. Report failure if none exists. Success certifies relaxed service, not necessarily exact-service feasibility.

Algorithm 1 Minimum-Priority Capture Search for one profile

Input: A metric space (P, d), priorities ρ, service parameter λ, outlier budget z, a fixed radius profile $\widehat{R}$, a fixed tie order, and a state (U, J, b) with its path's partial solution.
Output: Candidate centered partitions and outlier sets recorded at successful leaves.

```
procedure SEARCH(U, J, b)
if U = ∅ then record the centered partition and outliers as a candidate and return
if J = ∅ and b = z then return failure
choose the first p ∈ arg min_{x∈U} ρ(x) under the fixed tie order
if b < z then branch: declare p an outlier and call SEARCH(U \ {p}, J, b + 1)
for each j ∈ J do
    A ← A(p, j, U)
    branch: open p, assign A to p, and call SEARCH(U \ A, J \ {j}, b)
end for
end procedure
```

4.3 APPROXIMATION AND RUNNING-TIME ANALYSIS

Finally, we prove the capture property and derive the approximation and running-time guarantees. Assume $\mathrm{OPT}_\lambda^{\mathrm{serv}} < \infty$ and fix an optimal λ-service-fair solution with outlier set O^* and exactly k labeled clusters (c_j^*, X_j^*), using zero-radius singleton splitting if necessary. Let R_j^* be its radii, and fix a profile $\widehat{R} \ge R^*$ from Lemma 6.

Lemma 7. *At a search state with remaining set U, let p minimize ρ on U. For any unused label j, if $p \in X_j^*$, then $X_j^* \cap U \subseteq A(p, j, U)$.*

Proof. For any $x \in X_j^* \cap U$, the triangle inequality and the radius bound give $d(p, x) \le d(p, c_j^*) + d(c_j^*, x) \le 2R_j^* \le 2\widehat{R}_j$. The λ-service fairness of the optimum gives $d(p, x) \le d(p, c_j^*) + d(c_j^*, x) \le \lambda\rho(p) + \lambda\rho(x)$. Because $p, x \in U$ and p has minimum priority, $\rho(p) \le \rho(x)$, hence $d(p, x) \le 2\lambda\rho(x)$. Both tests in (3) hold. □

Lemma 8. *Every successful leaf defines a centered partition with at most k centers and at most z outliers, and every retained point x is assigned to its designated center within $2\lambda\rho(x)$.*

Proof. Each capture contains its pivot and is removed from U, so successful termination gives disjoint nonempty centered clusters partitioning the retained points. Each cluster consumes one unused label, and outlier branches require $b < z$. The second test in (3) ensures the service bound. □

Lemma 9. *For the profile dominating R^*, the search tree contains a successful root-to-leaf path with at most k cluster branches and at most z outlier branches.*

Proof. Maintain that $X_j^* \cap U = \emptyset$ for every used label j and that all declared outliers belong to O^*. This holds at the root. If the pivot $p \in O^*$, then p is undeclared, so $b < |O^*| \le z$ and the outlier branch is available. Otherwise $p \in X_j^*$ for a unique j, which is unused by the invariant. Lemma 7 ensures that taking this label removes $X_j^* \cap U$ and preserves the invariant. Over-capture preserves the invariant, and every additional point satisfies both guards in (3). Thus every nonempty state on this path has an available child, and each step removes its pivot and consumes a label or an outlier slot. The path reaches $U = \emptyset$ within $k + z$ steps. Appendix B gives the full induction. □

Lemma 10. *For any fixed profile $\widehat{R}$, every successful leaf has cost at most $2\sum_j \widehat{R}_j$.*

Proof. Index the output radii by used labels and set $R_j^A = 0$ for unused labels. Each label is used at most once, and (3) gives $R_j^A \le 2\widehat{R}_j$ for every j. Summing proves the claim. □

Theorem 11. *Assume $\mathrm{OPT}_\lambda^{\mathrm{serv}} < \infty$. For every $\varepsilon \in (0, 1]$ and $\lambda \ge 1$, there is a deterministic algorithm for service-aware MSR with outliers that returns at most k centers and at most z outliers and satisfies $\mathrm{cost}(A) \le (2+\varepsilon)\mathrm{OPT}_\lambda^{\mathrm{serv}}$ and $d(x, \sigma_A(x)) \le 2\lambda\rho(x)$ for every $x \notin O_A$. Its running time is $(k/\varepsilon)^{O(k)}(k+1)^{k+z}n^{O(1)}$.*

Proof. Set $\eta = \varepsilon/2$. For the profile approximating the optimal radii, Lemmas 9 and 10 produce a candidate satisfying

$$\mathrm{cost} \le 2\sum_j \widehat{R}_j \le 2(1+\varepsilon/2)\sum_j R_j^* = (2+\varepsilon)\mathrm{OPT}_\lambda^{\mathrm{serv}}.$$

Lemma 8 applies to every candidate, so choosing the cheapest across all profiles preserves validity and cannot increase this bound.

The number of profiles is at most $n^2(\lceil 2k/\varepsilon \rceil + 1)^k + 1$. In a search tree, each branch either consumes a label or increases b, so depth is at most $k+z$ and branching factor at most $k+1$. Pivots and absorbed sets are computed in polynomial time after distance preprocessing. Multiplying terms and absorbing polynomial factors yields the stated bound. □

Corollary 12. *Whenever the exactly service-fair comparison optimum is finite, for $\rho = \rho_{k,z}$ from (2) and $\lambda = 1$, the algorithm has cost at most $(2+\varepsilon)$ times the optimum exactly service-fair cost, assigns every retained point within $2\rho_{k,z}(x)$, and violates neither budget. When $z = 0$, the running time is $(k/\varepsilon)^{O(k)}n^{O(1)}$.*

5 The Factor-Two Service Lower Bound

In this section, we establish an algorithm-independent lower bound of 2 on the service-fairness factor for deterministic FPT algorithms without center augmentation. Appendix D shows that both factors in Lemma 7 are tight for the pivot-capture template. This lower bound does not establish optimality of the $(2+\varepsilon)$ MSR cost factor, as the reduction and verifier never evaluate the MSR objective.

Theorem 13. *Fix any constant $\delta \in (0, 1)$. Unless $\mathrm{FPT} = \mathrm{W}[2]$, there is no deterministic algorithm that runs in $f(k+z)n^{O(1)}$ time on every input and, whenever exact 1-service is feasible, preserves the center and outlier budgets and guarantees service stretch at most $2 - \delta$. This holds already for $z = 0$ in $\{0, 1, 2\}$ metrics with canonical $\rho_{k,0} \equiv 1$, even with no constraint on the output cost.*

Proof. Reduce from the W[2]-complete Dominating Set problem (Downey & Fellows, 1995), with $G = (V, E)$, $h = |V|$, closed neighborhoods $N_G[v]$, and $3 \le k \le h$. Other cases are directly decidable. Create one candidate s_v per vertex, $M = k + 2$ filler types with $L = h + h\binom{M}{2}$ copies each, and demands $x_{u,A}$ for $u \in V$ and $A \in \binom{[M]}{2}$. Candidates and fillers form a distance-1 clique.

A demand $x_{u,A}$ is at distance 1 exactly from candidates s_v with $u \in N_G[v]$ and fillers of a type in A. All other distinct pairs have distance 2. Nonzero distances lie in $\{1, 2\}$, so the triangle inequality holds. The metric has $N = (k+3)L = O(hk^3)$ points, is constructible in $O(N^2)$ time, and leaves k unchanged. Since $k \ge 3$, its canonical rank satisfies $2 \le \lceil N/k \rceil \le 2L$. Every radius-0 ball

is a singleton, whereas each radius-1 ball contains either two full filler types or the candidate/filler clique. Thus all canonical priorities equal 1.

Given a dominating set of size at most k, select the corresponding candidates and serve every point at distance at most 1. Selected centers serve themselves. Conversely, suppose at most k centers give exact service, and fix a graph vertex u not dominated by the vertices corresponding to the selected candidates. If the solution represents b filler types, then at most $k-b$ demands associated with u can themselves be centers, whereas $\binom{k+2-b}{2} > k-b$ pairs avoid all represented filler types. One corresponding unselected demand has no selected center at distance 1, a contradiction. Thus exact service is feasible exactly when G has a dominating set of size at most k.

Run a hypothetical $(2-\delta)$-service algorithm and verify in polynomial time that its output is a centered partition of all points satisfying the center budget and service bound. On a yes-instance, exact feasibility guarantees a passing output. Since every priority is 1 and every distance lies in $\{0,1,2\}$, any passing output has service distance at most 1. Thus no output can pass on a no-instance, regardless of the algorithm's behavior outside its feasibility promise. The assumed all-input running-time bound yields an FPT decision procedure for DOMINATING SET, contradicting W[2]-hardness unless FPT = W[2]. Appendix G gives the full details. □

6 EXTENSIONS TO RADIUS NORMS AND DIAMETERS

In this section, we extend minimum-priority capture to monotone symmetric norms of cluster radii and to service-aware min-sum-diameters.

6.1 MONOTONE SYMMETRIC NORMS OF RADII

We extend the cost guarantee to monotone symmetric norms using the coordinatewise bounds on output radii. Let $\mathrm{OPT}^{\mathrm{serv}}_{\lambda,\Phi}$ denote the minimum $\Phi(R)$ over the exact λ-service-fair solutions of Definition 1, where Φ is a monotone symmetric norm on $\mathbb{R}^k$ and radius vectors are padded with zeros.

Theorem 14. *For every fixed polynomial-time computable monotone symmetric norm Φ with $\mathrm{OPT}^{\mathrm{serv}}_{\lambda,\Phi} < \infty$, Theorem 11 remains valid with $\Phi(R^A) \le (2+\varepsilon)\mathrm{OPT}^{\mathrm{serv}}_{\lambda,\Phi}$ and the same service, budget, and running-time guarantees.*

Proof sketch. For a Φ-optimal R^*, Appendix C gives a profile $\widehat{R} \ge R^*$ with $\Phi(\widehat{R}) \le (1+\eta)\Phi(R^*)$. Rounding adds at most $\eta\Delta/k$ per coordinate for $\Delta = \|R^*\|_\infty$, while symmetry, monotonicity, and the triangle inequality give $\Phi(\mathbf{1}) \le k\Phi(e_1)$ and $\Phi(R^*) \ge \Delta\Phi(e_1)$. Lemmas 9 and 10 give a successful leaf with $R^A \le 2\widehat{R}$ after indexing by labels and zero-padding. Thus $\Phi(R^A) \le 2\Phi(\widehat{R})$. Set $\eta = \varepsilon/2$ and minimize $\Phi(R^A)$ over successful leaves from all profiles. □

6.2 SERVICE-AWARE MIN-SUM-DIAMETERS

We adapt the capture guard to obtain the same guarantees for service-aware min-sum-diameters. We retain the feasible solutions of Definition 1, including designated in-cluster centers and the service constraint, but minimize $\sum_i D_i$, where $D_i = \max_{x,y\in X_i} d(x,y)$. Zero-diameter singleton splitting again permits exactly k clusters when $n - z \ge k$.

Theorem 15. *Let $\mathrm{OPT}^{\mathrm{MSD}}_{\lambda} < \infty$ denote the exact λ-service-fair MSD optimum. For every $\varepsilon \in (0,1]$, a deterministic algorithm preserves both budgets and guarantees $\sum_i D_i^A \le (2+\varepsilon)\mathrm{OPT}^{\mathrm{MSD}}_{\lambda}$ and $d(x,\sigma_A(x)) \le 2\lambda\rho(x)$ for every retained point x, in the same asymptotic running time as Theorem 11. Any fixed polynomial-time computable monotone symmetric norm of zero-padded diameter vectors has the same guarantees against its exact-service optimum.*

Proof sketch. Enumerate upward-rounded diameter profiles. In a label-j branch, keep the service guard but replace $d(p,x) \le 2\widehat{R}_j$ by $d(p,x) \le \widehat{D}_j$. For $p,x \in X_j^* \cap U$, $d(p,x) \le D_j^* \le \widehat{D}_j$, and minimum priority gives the same service bound, preserving the consistent path. The triangle inequality bounds each output diameter by $2\widehat{D}_j$. Minimize $\sum_i D_i^A$ (or $\Phi(D^A)$ for norms) over all successful leaves and profiles. Rounding with $\eta = \varepsilon/2$ gives the claim (Appendix E). □

Scope of tightness. Theorem 13 uses only service feasibility, so its lower bound applies to both extensions. Unless $\mathrm{W}[2] = \mathrm{FPT}$, no deterministic FPT algorithm preserving both budgets can guarantee service stretch $2 - \delta$ for any fixed $\delta \in (0, 1)$, even with unrestricted cost, $z = 0$, and canonical $\rho_{k,0} \equiv 1$. This establishes service-factor optimality, not cost-factor optimality.

7 Geometric Diagnostics and Computational Verification

In this section, we examine the assignment gap on geometric instances and check the correspondence between the algorithm and its implementation.

Data and exact comparison protocol. We use Euclidean metrics from the Iris, Wine, and Breast Cancer Wisconsin Diagnostic datasets in scikit-learn (Pedregosa et al., 2011). After standardizing each full feature matrix, we take 20 fixed 12-point subsamples per dataset. Labels are unused. With $k = 3$, $z = 0$, $\lambda = 1$, and rank-4 canonical priorities, exhaustive center-set enumeration and a mask dynamic program compute the exact unconstrained, selected-set-fair, and service-aware MSR optima. We compute t_{opt}, the smallest service stretch $t \geq 1$ among all selected-set-optimal solutions. Thus $t_{\mathrm{opt}} > 1$ rules out any cost-preserving repair to exact service.

The assignment gap on geometric instances. The service-aware optimum strictly exceeds the selected-set optimum in 52 of 60 samples (65–100% per dataset, Table 1). Sensitivity checks for $k \in \{2, 3, 4\}$ find the gap at each budget, with the largest median ratios at $k = 4$ (Appendix F).

Table 1: Exact geometric diagnostic over 20 fixed subsamples per dataset ($n = 12$, $k = 3$). Set/Unc. and Serv./Set are exact optimum ratios. Ratios and stretches are medians unless labeled "max". Percentages give trial frequencies.

Dataset	Trials	Set/Unc.	Serv./Set	Serv./Set max	$t_{\mathrm{opt}} > 1$	median t_{opt}
Iris	20	1.117	1.069	1.680	65%	1.301
Wine	20	1.050	1.543	2.300	95%	1.299
Breast Cancer	20	1.134	1.491	1.900	100%	1.265

Direct validation of the profile search. With $\varepsilon = 1$, the complete profile search on the same 60 metrics has maximum cost ratio $\mathrm{cost}(A)/\mathrm{OPT}_1^{\mathrm{serv}}$ of 1.391 and maximum service stretch of 1.987, within the respective bounds 3 and 2 in Theorem 11. Neither budget is violated. A separate tiny-instance suite covers random and exhaustive graph metrics, arbitrary priorities, radius norms, the separation family, and exact MSD instances, while independent assignment enumeration cross-validates the exact solvers. These finite checks do not replace proofs. Appendix F and the supplement provide detailed diagnostics and reproduction materials.

8 Conclusion

A nearby selected center does not by itself certify low-cost fair service for a partition-dependent objective. Nearest-assignment stability suffices for cost-preserving transfer, whereas MSR admits a $\Theta(k)$ gap between the exact selected-set and service-aware optima. For arbitrary priorities, minimum-priority capture is deterministic and FPT in $k + z$ for fixed ε. Whenever exact λ-service is feasible, it guarantees cost at most $(2 + \varepsilon)\mathrm{OPT}_\lambda^{\mathrm{serv}}$, service stretch 2, and no budget violations. The same mechanism extends to fixed polynomial-time computable monotone symmetric radius norms and service-aware MSD. Unless $\mathrm{W}[2] = \mathrm{FPT}$, budget-preserving deterministic FPT algorithms cannot improve the service-fairness factor below 2 by any fixed positive constant, even with unrestricted cost. This establishes optimality of the service-fairness factor, not the MSR cost approximation factor. Future work could seek a $(1+\varepsilon)$ cost approximation while preserving factor-two service, attain sub-two service through extra centers or randomization, and incorporate fairness constraints into outlier selection.

AI USE STATEMENT

Generative AI tools were used only for language polishing of the manuscript and assistance in drafting validation and diagnostic code. All literature discovery was conducted by the authors, and all research ideas, problem formulation, algorithm design, theoretical results and proofs, experimental design, analysis, and substantive scientific content were developed by the authors. The authors independently reviewed and verified all AI-assisted material against the relevant literature, formal arguments, and executable diagnostics, and take full responsibility for the final manuscript, code, experiments, and results.

ETHICS STATEMENT

Distance-based individual fairness is a mathematical surrogate, not a complete certificate of equitable deployment. Its validity depends on the metric, the meaning of the priority radius, the service represented by assignment, and the population included in the data. In particular, allowing outliers can exclude people from service and may create harms not measured by the retained-point guarantee. Any deployment should justify these choices with domain expertise and stakeholder input, and should evaluate group-level and procedural concerns in addition to the local distance condition studied here.

REPRODUCIBILITY STATEMENT

All proofs needed for the main guarantees are included in the paper and appendices. The supplement contains the exact comparison solver, the complete profile-search implementation, the fixed-k implementation runtime diagnostic, arbitrary-priority/norm and MSD verification, the real-data diagnostic with fixed sample indices, the $k \in \{2, 3, 4\}$ sensitivity check, and canonical-radius lower-bound verification. It also includes trial-level CSVs, JSON summaries, and independent brute-force cross-validation. The finite experiments are intended as geometric diagnostics, theorem-to-code checks, and implementation sanity checks rather than deployment or large-scale scalability benchmarks.

REFERENCES

Sina Bagheri Nezhad, Sayan Bandyapadhyay, and Tianzhi Chen. Polynomial-time constant-approximation for fair sum-of-radii clustering. In *Proceedings of the 33rd Annual European Symposium on Algorithms*, pp. 62:1–62:16, 2025.

Tanvi Bajpai, Deeparnab Chakrabarty, Chandra Chekuri, and Maryam Negahbani. Revisiting priority k-center: Fairness and outliers. In *Proceedings of the 48th International Colloquium on Automata, Languages, and Programming*, pp. 21:1–21:20, 2021.

Sayan Bandyapadhyay and Tianzhi Chen. Improved FPT approximation for sum of radii clustering with mergeable constraints. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM*, pp. 23:1–23:17, 2025.

Sandip Banerjee, Yair Bartal, Lee-Ad Gottlieb, and Alon Hovav. Improved fixed-parameter bounds for min-sum-radii and diameters k-clustering and their fair variants. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 15481–15488, 2025.

MohammadHossein Bateni, Vincent Cohen-Addad, Alessandro Epasto, and Silvio Lattanzi. A scalable algorithm for individually fair k-means clustering. In *Proceedings of the 27th International Conference on Artificial Intelligence and Statistics*, pp. 3151–3159, 2024.

Moritz Buchem, Katja Ettmayr, Hugo K. K. Rosado, and Andreas Wiese. A $(3 + \epsilon)$-approximation algorithm for the minimum sum of radii problem with outliers and extensions for generalized lower bounds. In *Proceedings of the 2024 Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 1738–1765, 2024.

Lena Carta, Lukas Drexler, Annika Hennes, Clemens Rösner, and Melanie Schmidt. FPT approximations for fair k-min-sum-radii. In *Proceedings of the 35th International Symposium on Algorithms and Computation*, pp. 16:1–16:18, 2024.

Moses Charikar and Rina Panigrahy. Clustering to minimize the sum of cluster diameters. In *Proceedings of the 33rd Annual ACM Symposium on Theory of Computing*, pp. 1–10, 2001.

Xianrun Chen, Dachuan Xu, Yicheng Xu, and Yong Zhang. Parameterized approximation algorithms for sum of radii clustering and variants. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 20666–20673, 2024.

Rod G. Downey and Michael R. Fellows. Fixed-parameter tractability and completeness I: Basic results. *SIAM Journal on Computing*, 24(4):873–921, 1995.

Lukas Drexler, Annika Hennes, Abhiruk Lahiri, Melanie Schmidt, and Julian Wargalla. Approximating fair k-min-sum-radii in Euclidean space. In *Proceedings of the 21st International Workshop on Approximation and Online Algorithms*, pp. 119–133, 2023.

Matthijs Ebbens, Nicole Funk, Jan Höckendorff, Christian Sohler, and Vera Weil. A subquadratic time approximation algorithm for individually fair k-center. In *Proceedings of the 28th International Conference on Artificial Intelligence and Statistics*, pp. 2287–2295, 2025.

Arnold Filtser and Ameet Gadekar. FPT approximations for capacitated sum of radii and diameters. In *Proceedings of the 42nd International Symposium on Computational Geometry*, pp. 48:1–48:18, 2026.

Ameet Gadekar. FPT approximations for fair sum of radii with outliers and general norm objectives. *arXiv preprint arXiv:2602.05476*, 2026a.

Ameet Gadekar. On the parameterized approximability of (mergeable) sum of radii clustering. *arXiv preprint arXiv:2605.06398*, 2026b.

Matt Gibson, Gaurav Kanade, Erik Krohn, Imran A. Pirwani, and Kasturi Varadarajan. On clustering to minimize the sum of radii. *SIAM Journal on Computing*, 41(1):47–60, 2012.

Fabrizio Grandoni, Anupam Gupta, and Jatin Yadav. FPT approximation schemes for min-sum radii and min-sum diameters clustering. *arXiv preprint arXiv:2605.10895*, 2026.

Lu Han, Dachuan Xu, Yicheng Xu, and Ping Yang. Approximation algorithms for the individually fair k-center with outliers. *Journal of Global Optimization*, 87(2):603–618, 2023.

Christopher Jung, Sampath Kannan, and Neil Lutz. Service in your neighborhood: Fairness in center location. In *Proceedings of the 1st Symposium on Foundations of Responsible Computing*, pp. 5:1–5:15, 2020.

Pankaj Kumar, Haiko Müller, Sebastian Ordyniak, and Melanie Schmidt. On the parameterized complexity of min-sum-radii. In *Proceedings of the 20th Scandinavian Symposium on Algorithm Theory*, pp. 26:1–26:18, 2026.

Shuilian Liu, Gregory Gutin, Yicheng Xu, and Yong Zhang. Constant FPT approximation algorithms for colorful sum of radii. *Theoretical Computer Science*, 1076:115968, 2026a.

Shuilian Liu, Yicheng Xu, and Yong Zhang. Individual preference facility location: A dual-fitting framework and its extensions. *Theory of Computing Systems*, 70(2):29, 2026b.

Sepideh Mahabadi and Ali Vakilian. Individual fairness for k-clustering. In *Proceedings of the 37th International Conference on Machine Learning*, pp. 6586–6596, 2020.

Binita Maity, Shrutimoy Das, and Anirban Dasgupta. Linear programming based approximation to individually fair k-clustering with outliers. In *Proceedings of the 2025 IEEE International Conference on Data Mining Workshops*, pp. 340–347, 2025.

Maryam Negahbani and Deeparnab Chakrabarty. Better algorithms for individually fair k-clustering. In *Proceedings of the 35th International Conference on Neural Information Processing Systems*, pp. 13340–13351, 2021.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12: 2825–2830, 2011.

Ali Vakilian and Mustafa Yalçıner. Improved approximation algorithms for individually fair clustering. In *Proceedings of the 25th International Conference on Artificial Intelligence and Statistics*, pp. 8758–8779, 2022.

Di Wu, Qilong Feng, and Jianxin Wang. Better guarantees for individual fairness k-median. *Theoretical Computer Science*, 1074:115900, 2026.

A Additional Preliminaries and Boundary Cases

A.1 At most k versus exactly k clusters

Lemma 16. *Assume $n - z \geq k$. Any feasible service-aware solution with $m < k$ nonempty clusters can be transformed into one with exactly k nonempty clusters without increasing any original cluster radius, any monotone symmetric norm of the zero-padded radius vector, or any retained point's assignment distance. The outlier set is unchanged.*

Proof. Every feasible solution serves at least $n-z \geq k$ points. If it has $m < k$ clusters, some cluster X_i contains at least two points. Choose $x \in X_i \setminus \{c_i\}$, delete it from X_i, and create the singleton centered cluster $(x, \{x\})$. The original cluster remains nonempty and keeps center c_i. Deleting a point cannot increase its radius. The singleton has radius zero and is exactly service-fair. Repeating gives exactly k clusters. Match each original cluster to its remaining cluster and each new singleton to a padded zero coordinate. The new radius vector is coordinatewise no larger, so monotonicity and symmetry give the norm claim. □

A.2 Zero priorities and zero radii

If $\rho(x) = 0$, any finite-factor service guarantee requires the assigned center to be at distance zero from x. Since d is a metric, every retained zero-priority point must therefore be its own center. Algorithm 1 enforces this because the second test in (3) becomes $d(p, x) = 0$. If a profile coordinate is zero, the first test similarly admits only points at zero distance from the pivot. In every cluster branch, the pivot itself satisfies both tests, so recursion still progresses. The all-zero optimal profile is included explicitly in Lemma 6. The grid construction is used only when $\Delta > 0$.

A.3 Fewer than z outliers

The comparison optimum and the algorithm may use fewer than z outliers. A successful search stops as soon as $U = \emptyset$, independently of the current value of b. In the consistent-path proof, an optimal outlier that is absorbed by an earlier fair cluster does not require an outlier branch. Thus early absorption cannot make the path exceed the outlier budget.

A.4 Centers and assignments

Each cluster center is its pivot and belongs to its absorbed set because both relevant distances are zero. Hence every successful search returns a legitimate centered partition, not merely a covering by external balls. Points are removed from U immediately after assignment, so the produced clusters are disjoint and together with the declared outliers partition P. A successful search certifies 2λ-service, not necessarily exact λ-service: the algorithm is not an exact-service feasibility test. Its approximation guarantee is conditional on $\mathrm{OPT}_\lambda^{\mathrm{serv}} < \infty$.

B Formal Search Invariant and Complete Path Induction

By Lemma 16, fix an optimal exact λ-service-fair solution $(O^*, (c_j^*, X_j^*)_{j=1}^k)$ and a profile $\widehat{R} \geq R^*$ coordinatewise. At a state (U, J, b) on the intended path, let O_A be the declared outlier set. Maintain:

- (I1) $X_j^* \cap U = \emptyset$ for every used label $j \notin J$.
- (I2) $O_A \subseteq O^*$ and $b = |O_A| \leq z$.
- (I3) the assigned clusters and O_A are pairwise disjoint and partition $P \setminus U$. Each used label j has one nonempty cluster containing its pivot, of radius at most $2\widehat{R}_j$ and satisfying 2λ-service.

At the root $(P, [k], 0)$, no cluster has been assigned and $O_A = \emptyset$, so (I1)–(I3) hold.

Lemma 17. *Suppose a state satisfies (I1)–(I3) and $U \neq \emptyset$. Then the algorithm has an available child satisfying the invariant. This child may already have an empty remaining set.*

Proof. Let p be the minimum-priority pivot. If $p \in O^*$, then fewer than $|O^*| \le z$ optimal outliers have been declared so far: declared outliers are distinct and belong to O^* by (I2), while p is an undeclared member of O^*. Thus $b < z$ and the outlier child is available. Removing p preserves all invariant clauses.

Otherwise $p \in X_j^*$ for a unique optimal cluster. Label j is unused. Indeed, if it had been used, (I1) would imply that all then-remaining points of X_j^* had been removed, contradicting $p \in U$. By Lemma 7, the branch using j removes every point of $X_j^* \cap U$, whether or not c_j^* is still in U. Marking label j as used preserves (I1), and (I2) is unchanged. The absorbed set lies in U, contains p, and satisfies both capture guards, so (I3) is preserved as well. □

Repeatedly applying Lemma 17 proves Lemma 9. Each successor decreases the nonnegative integer $|J| + z - b$ by one, from its initial value $k + z$. A nonempty state cannot exhaust both resources, since the lemma supplies an available child. Thus the path reaches $U = \emptyset$ within $k + z$ steps. Over-captured points, including optimal outliers or optimal centers, already receive valid service. An unused label may have no remaining optimal points, so it need not be consumed before successful termination.

C Profile Enumeration for Monotone Symmetric Norms

The proof in Theorem 14 can be isolated as a profile-list lemma.

Lemma 18. *Let $\eta \in (0, 1]$ and let Φ be a monotone symmetric norm on $\mathbb{R}^k$, with monotonicity understood on nonnegative vectors. For every $R^* \in \mathcal{D}^k$, the list $\mathcal{L}_\eta$ of Lemma 6 contains $\widehat{R} \ge R^*$ coordinatewise with*

$$\Phi(\widehat{R}) \le (1 + \eta)\Phi(R^*).$$

Proof. If $R^* = 0$, select the all-zero profile. Otherwise choose the correct $\Delta = \|R^*\|_\infty$, let $\delta = \eta\Delta/k$, and round upward as in Lemma 6. Then $0 \le \widehat{R} - R^* \le \delta\mathbf{1}$. Put $a = \Phi(e_1) > 0$, where e_1 is the first standard basis vector. Symmetry and the triangle inequality give $\Phi(\mathbf{1}) \le ka$, while monotonicity and a maximum coordinate of R^* give $\Phi(R^*) \ge \Delta a$. Therefore

$$\Phi(\widehat{R}) \le \Phi(R^*) + \delta\Phi(\mathbf{1}) \le \Phi(R^*) + \eta\Delta a \le (1 + \eta)\Phi(R^*).$$ □

Both the list and the rounded vector chosen in this proof are independent of Φ. For each chosen norm, minimize Φ of the zero-padded radius vector over all successful leaves, rather than reuse the leaf minimizing the sum of radii. The stated running time assumes polynomial-time evaluation of this fixed norm. Examples include ℓ_1 MSR, ℓ_∞ maximum radius, and intermediate ℓ_p norms of cluster radii. The selected leaf may depend on Φ.

D Template Tightness of the Factor Two

Lemma 19. *There is a three-point path with prescribed priorities on which, for a valid fixed tie order, the search with the exact optimal radius profile produces cost ratio 2 against the exact-service optimum and service stretch 2.*

Proof. Let the path be $p - c - x$ with both edges of length 1, let $k = 1$, $z = 0$, $\lambda = 1$, and set $\rho(p) = \rho(c) = \rho(x) = 1$. The exactly service-fair optimum centers the single cluster at c, has radius 1, and hence cost 1. Its exact profile is $\widehat{R}_1 = 1$.

Fix the deterministic tie order so that p is selected first. The absorbed set contains all three points: c is at distance 1, and x is at distance $2 = 2\widehat{R}_1 = 2\rho(x)$. The only search branch therefore centers the cluster at p, giving radius and cost 2. Point x is served at exactly twice its priority. This shows tightness of the two triangle-inequality steps used by the present capture template. These priorities are not the canonical radii for $k = 1$, and the example is not an algorithm-independent cost lower bound. Appendix G gives the separate canonical-radius hardness result for service stretch $2 - \delta$, for every fixed $\delta \in (0, 1)$, in the budget-preserving deterministic FPT regime. □

E Full Proof of the Min-Sum-Diameters Extension

For $\varepsilon \in (0, 1]$, assume $\mathrm{OPT}_\lambda^{\mathrm{MSD}} < \infty$ and set $\eta = \varepsilon/2$. As in Lemma 16, singleton splitting preserves service and cannot increase a diameter, so fix an optimal exact λ-service-fair MSD solution with exactly k labeled centered clusters (c_j^*, X_j^*) and diameters D_j^*. Every diameter, including zero, is a pairwise input distance, so Lemma 6 applies to D^*. For any enumerated profile $\widehat{D}$, define

$$A_D(p, j, U) = \{x \in U : d(p, x) \le \widehat{D}_j,\ d(p, x) \le 2\lambda\rho(x)\}.$$

The search is Algorithm 1 with A replaced by A_D.

Lemma 20. *Suppose $\widehat{D} \ge D^*$ coordinatewise. If p is the minimum-priority remaining point and $p \in X_j^*$ for an unused label j, then $X_j^* \cap U \subseteq A_D(p, j, U)$.*

Proof. For $x \in X_j^* \cap U$, the definition of diameter gives $d(p, x) \le D_j^* \le \widehat{D}_j$. Exact service fairness through c_j^* gives $d(p, x) \le \lambda\rho(p) + \lambda\rho(x) \le 2\lambda\rho(x)$ because $\rho(p) \le \rho(x)$. □

The label-availability, outlier, and termination arguments of Lemmas 9 and 8 are unchanged. A cluster created with label j lies inside the $\widehat{D}_j$-ball around p, so for any two assigned points x, y,

$$d(x, y) \le d(x, p) + d(p, y) \le 2\widehat{D}_j.$$

Index output diameters by used labels and put $D_j^A = 0$ for unused labels. Thus every successful leaf satisfies $D^A \le 2\widehat{D}$. For the profile supplied by Lemma 6, the consistent path exists and

$$\sum_j D_j^A \le 2 \sum_j \widehat{D}_j \le 2(1 + \eta) \sum_j D_j^* = (2 + \varepsilon)\mathrm{OPT}_\lambda^{\mathrm{MSD}}.$$

Return the leaf of minimum actual sum of diameters across all profiles. The profile count and search tree are unchanged, and computing actual diameters takes polynomial time.

For a fixed polynomial-time computable monotone symmetric norm Φ, instead fix a Φ-optimal exact-service solution. Lemma 18 gives $\widehat{D} \ge D^*$ with $\Phi(\widehat{D}) \le (1 + \eta)\Phi(D^*)$. The consistent path and coordinatewise diameter bound imply $\Phi(D^A) \le 2\Phi(\widehat{D}) \le (2 + \varepsilon)\Phi(D^*)$. Minimizing this norm over all successful leaves proves the norm extension with the same service, budget, and running-time guarantees.

F Additional Details for the Computational Diagnostics

Exact service-aware comparison optimum. For each nonempty subset $S \subseteq P$, the subset-DP audit computes

$$g_\lambda(S) = \min_{c \in S} \left\{ \max_{x \in S} d(x, c) : d(x, c) \le \lambda\rho(x)\ \forall x \in S \right\},$$

with $g_\lambda(S) = +\infty$ when no eligible center exists. Let $F(U, \ell)$ be the minimum cost of partitioning U into at most ℓ feasible centered clusters. For nonempty U and $\ell \ge 1$, fix a point $u(U) \in U$ and use

$$F(U, \ell) = \min_{\substack{S \subseteq U \\ u(U) \in S}} \{g_\lambda(S) + F(U \setminus S, \ell - 1)\}.$$

Set $F(\emptyset, \ell) = 0$ for $\ell \ge 0$ and $F(U, 0) = +\infty$ for nonempty U. Minimize $F(P \setminus O, k)$ over all $O \subseteq P$ with $|O| \le z$. Requiring $u(U) \in S$ removes redundant cluster orderings. Priorities remain fixed across outlier choices.

Exact selected-set versus service-aware solver on real data. For the $z = 0$ real-data instances and a fixed center set C, each center c has candidate radius values among $\{d(c, x) : x \in P\}$. A radius r covers the assignment mask $\{x : d(c, x) \le r\}$. The mask dynamic program starts with the empty covered mask at cost zero and processes each center once: choosing radius r unions its assignment mask with the current mask and adds r to the cost. It keeps the minimum cost for each

resulting mask. The full-mask value is the optimum for C, or $+\infty$ if no full mask is reachable. For the selected-set relaxation we first require $d(x, C) \le \rho(x)$ for all x and leave assignment masks unrestricted. For service factor t, the mask for (c, r) is instead truncated to points satisfying both $d(c, x) \le r$ and $d(c, x) \le t\rho(x)$. Enumerating every center set of size at most k is exact for the tiny diagnostic samples. Although these masks can overlap, they encode the centered-partition model exactly: assign every selected center to itself and each other point to any selected mask containing it. Each center belongs to its own mask, so this gives disjoint nonempty clusters without increasing any radius or violating a service guard. Conversely, any admissible centered partition supplies covering masks at its actual radii. Thus the covering and partition optima coincide for fixed C.

We report t_{opt}, the least factor $t \ge 1$ achieved by any selected-set-optimal solution. Enumerate the exactly selected-set-fair center sets whose unrestricted assignment cost equals $\mathrm{OPT}_1^{\mathrm{set}}$, then binary-search $\{1\} \cup \{\max\{1, d(c, x)/\rho(x)\} : c, x \in P,\ \rho(x) > 0\}$. For a retained point with zero priority, only zero-distance assignments are eligible at any finite factor. The test asks whether one of the enumerated center sets has truncated assignment cost at most $\mathrm{OPT}_1^{\mathrm{set}}$. Exact selected-set fairness of that center set forces equality. This monotone test finds the smallest feasible breakpoint. Thus $t_{\mathrm{opt}} > 1$ is equivalent to $\mathrm{OPT}_1^{\mathrm{serv}} > \mathrm{OPT}_1^{\mathrm{set}}$ on these samples. The separate global threshold permits arbitrary center sets and asks for cost *at most* $\mathrm{OPT}_1^{\mathrm{set}}$, not necessarily equality. It is never larger than t_{opt} and is not a proxy for the stretch of a selected-set-optimal solution.

The real-data script uses the Iris, Wine, and Breast Cancer Wisconsin Diagnostic feature matrices shipped with scikit-learn. Each full feature matrix is standardized with `StandardScaler` before subsampling. For trial $j \in \{0, \dots, 19\}$, the seed is $73000 + 1000a + j$, with $a = 0, 1, 2$ for Iris, Wine, and Breast Cancer, respectively. The code uses `numpy.random.default_rng` to sample 12 indices without replacement, sorts them, and computes Euclidean distances. We use $k = 3$, $z = 0$, $\lambda = 1$, hence rank $q = 4$. Labels are never used. Trial-level indices and costs are stored in `real_data_results/real_data_trials.csv`. An independent brute-force program explicitly enumerates center sets and all center assignments on 30 random six-point metrics. All 90 objective/mode comparisons match the mask-DP solver, and the independently enumerated t_{opt} values match as well.

Numerical convention. "Exact" refers to exhaustive combinatorial optimization, not symbolic real arithmetic. The real-data solver uses floating-point distances, with absolute tolerance 10^{-9} for coverage and DP updates and 10^{-8} for optimum-cost ties and tests of $t_{\mathrm{opt}} > 1$. Priorities at most 10^{-9} are treated as zero when constructing service-stretch breakpoints. The recorded environment is Python 3.13.5, NumPy 2.3.5, and scikit-learn 1.8.0. The supplement contains `TESTED_ENVIRONMENT.txt` and executable reproduction commands.

Sensitivity to the center budget. To check that the diagnostic is not an artifact of $k = 3$, we rerun the first 10 fixed subsamples per dataset for $k \in \{2, 3, 4\}$ (with the corresponding standard rank $\lceil n/k \rceil$). The table reports medians and maxima of exact optimum ratios, with ratios and stretches rounded to three decimal places. "Need > 1" denotes $t_{\mathrm{opt}} > 1$.

k	Dataset	Trials	Serv./Set	max	Need > 1 / median t_{opt}
2	Iris	10	1.000	1.507	40% / 1.000
2	Wine	10	1.000	1.499	30% / 1.000
2	Breast Cancer	10	1.068	1.370	50% / 1.018
3	Iris	10	1.080	1.680	70% / 1.346
3	Wine	10	1.543	2.300	90% / 1.150
3	Breast Cancer	10	1.493	1.791	100% / 1.265
4	Iris	10	1.306	1.836	90% / 1.489
4	Wine	10	1.863	2.117	100% / 1.565
4	Breast Cancer	10	1.663	1.944	100% / 1.535

The effect is weaker for $k = 2$ on Iris/Wine but remains present in some trials, while for $k = 4$ exact service-aware cost is substantially larger than selected-set cost across all three datasets. This is a sensitivity diagnostic, not a scaling experiment.

Exact implementation of the proposed algorithm. The MSR audit enumerates the all-zero profile and, for every positive pairwise distance Δ, the complete ordered grid with $\eta = \varepsilon/2$. For each fixed profile it memoizes the minimum remaining MSR cost at states (U, J, b) and follows the minimum-priority, fixed-index tie rule of Algorithm 1. Memoization is valid because future captures depend only on this state and MSR cost is additive over created clusters. The norm audit instead carries the accumulated radii to each leaf and evaluates the complete zero-padded vector. It does not assume an additive recurrence for arbitrary norms. The extended audit enumerates all 728 connected labeled unweighted graphs on five vertices and checks their shortest-path metrics. Separate randomized checks test arbitrary priorities, including zeros, with $\lambda \in \{1, 1.5, 2\}$, and ℓ_2/ℓ_∞ radius norms. The MSD checker independently computes exact service-aware diameter optima and tests 80 tiny tree/Euclidean instances. The lower-bound audit checks the canonical-rank and pair-counting inequalities and a symmetry-reduced center-budget feasibility condition. It does not enumerate center sets on the full expanded metric.

Interpretation. The real-data experiment shows that the assignment mismatch occurs on subsamples of these three standardized feature datasets. It does not establish domain-specific fairness or large-scale utility. The finite exhaustive/random checks do not replace proof. Their purpose is to make objective definitions, recurrences, and theorem-to-code translations inspectable and to expose small counterexamples if a boundary case has been mishandled.

Direct profile-search diagnostic. Using $\varepsilon = 1$, we additionally run the complete profile search on all 60 $n = 12, k = 3$ metrics. The implementation visits $1+|\mathcal{D}_+|(\lceil 2k/\varepsilon \rceil+1)^k$ profiles, where $\mathcal{D}_+$ is the set of distinct positive distances. Repeated vectors across scale grids are counted separately. Here this is $1 + 66 \cdot 7^3 = 22{,}639$ visits per instance. Against the exact $\mathrm{OPT}_2^{\mathrm{serv}}$ benchmark under service factor 2, the algorithm/$\mathrm{OPT}_2^{\mathrm{serv}}$ ratio has median 1.106 and maximum 1.443. The realized service stretch has median 1.234 and maximum 1.987. Against the stricter exact $\mathrm{OPT}_1^{\mathrm{serv}}$ benchmark, the corresponding cost ratios are 0.699 and 1.391. Values below one are possible because the algorithm may use service stretch above one. Only $\mathrm{OPT}_1^{\mathrm{serv}}$ is the comparison optimum in Theorem 11 for this run. The $\mathrm{OPT}_2^{\mathrm{serv}}$ comparison is an additional same-service-envelope diagnostic. Trial-level outputs, fixed sample indices, and the dedicated reproduction script are included in the supplement.

Fixed-k implementation runtime diagnostic. To make the literal exhaustive profile-search implementation inspectable beyond $n = 12$, we additionally run it at fixed $k = 3$, $z = 0$, and $\varepsilon = 1$ on two deterministic subsamples per dataset and size for $n \in \{12, 16, 20\}$. We omit exact comparison optima in this timing experiment. This is a runtime sanity check rather than a quality or scalability benchmark. Across the six runs at each n, the results are:

n	profile visits	median full-search time (s)	max service stretch
12	22,639	0.713	1.682
16	41,161	1.538	1.611
20	65,171	2.800	1.658

Times are machine- and implementation-dependent. The purpose of this table is only to expose the behavior of the released literal search as n increases at fixed k. It is not evidence of large-scale practical scalability. The supplementary script records the fixed sample indices, seeds, profile counts, costs, service stretches, and per-instance wall-clock times.

G Deterministic FPT Service-Factor Optimality Without Center Augmentation

We expand the reduction in Theorem 13. It is parameter preserving, forces the standard neighborhood priorities rather than supplying artificial priorities, and never invokes the MSR objective. The hardness conclusion concerns deterministic FPT algorithms under exact center-budget preservation.

Construction. We reduce from Dominating Set, which is W[2]-complete parameterized by the requested size k (Downey & Fellows, 1995). It suffices to consider $3 \le k \le h := |V|$: smaller

values are decidable by direct enumeration, while $k > h$ is trivial. Let $G = (V, E)$ and let $N_G[v]$ denote the closed neighborhood of v. Put

$$M = k + 2, \qquad D = h\binom{M}{2}, \qquad L = h + D.$$

The metric contains three kinds of points. For each $v \in V$, create a set candidate s_v. For each type $i \in [M]$, create filler copies $f_{i,1}, \ldots, f_{i,L}$. Finally, create a demand $x_{u,A}$ for every $u \in V$ and pair $A \in \binom{[M]}{2}$, giving D demands.

Distinct set candidates and filler copies are at distance 1 from one another. A demand $x_{u,A}$ is at distance 1 from s_v exactly when $u \in N_G[v]$, and from every filler copy of a type in A. Every other pair of distinct points is at distance 2.

Lemma G.1. *The construction is a metric on $N = (k + 3)L = O(hk^3)$ points, with a distance matrix constructible in $O(N^2)$ time from the adjacency matrix of G. For center budget k and $z = 0$, its canonical priorities satisfy $\rho_{k,0}(x) = 1$ for every point x.*

Proof. All nonzero distances lie in $\{1, 2\}$. For distinct a, b, the triangle inequality is immediate if the intermediate point equals an endpoint. Otherwise both legs are at least 1, so their sum is at least $2 \geq d(a, b)$.

Because $h + D = L$ and $M = k + 2$, the number of points is $N = h + ML + D = (k + 3)L$. The canonical rank from (2) is $q = \lceil N/k \rceil \leq 2L$ for $k \geq 3$, and $q \geq 2$. A demand $x_{u,A}$ has in its radius-1 ball itself, all $2L$ fillers of its two types, and s_u, hence at least $2L + 2 \geq q$ points. Every set candidate or filler has all $h + ML > 2L \geq q$ set-candidate and filler points in its radius-1 ball. Radius-0 balls are singletons, so the least radius attaining rank q is exactly 1 for every point. □

Lemma G.2. *The graph G has a dominating set of size at most k if and only if the constructed instance has an exact 1-service-fair solution using at most k centers.*

Proof. Suppose $S \subseteq V$ dominates G and $|S| \leq k$. Select $\{s_v : v \in S\}$. Every set candidate and filler is within distance 1 of every selected set candidate. For each demand $x_{u,A}$, domination gives $v \in S$ with $u \in N_G[v]$, so $d(x_{u,A}, s_v) = 1$. Assign selected centers to themselves and every other point to an eligible selected candidate. This gives nonempty centered clusters partitioning all points and, by Lemma G.1, exact service.

Conversely, let C be any center set of size at most k that serves every point within distance 1, and define $S = \{v : s_v \in C\}$. Assume for contradiction that some $u \in V$ is not dominated by S. Let $J \subseteq [M]$ be the filler types represented in C and write $b = |J|$. At least b centers are fillers, so the number c of selected demands of the form $x_{u,A}$ is at most $k - b$ (selected set candidates, other demands, and extra filler copies can only reduce this number).

Exactly $\binom{M-b}{2}$ pairs A avoid J. Writing $t = k - b \geq 0$,

$$\binom{M-b}{2} = \binom{k+2-b}{2} = \binom{t+2}{2} > t = k - b \geq c,$$

where the strict inequality follows from $\binom{t+2}{2} - t = (t^2 + t + 2)/2 > 0$. Hence there is a pair A disjoint from J for which $x_{u,A} \notin C$. This demand is unselected. It is at distance 2 from every selected set candidate, since u is undominated, and from every selected filler, since $A \cap J = \emptyset$. It is also at distance 2 from every other demand. Thus it has no center within distance 1, contradicting exact service. Therefore S dominates G, and $|S| \leq |C| \leq k$. □

Why the gadgets are needed. The L copies per filler type make every demand's canonical radius equal to 1 without allowing radius 0. Pair-indexed demands prevent non-set centers from bypassing domination: after b filler types consume budget, more than the remaining $k - b$ pair demands avoid those types. Selecting individual demands cannot help one another because distinct demands are at distance 2.

Proof of Theorem 13. Assume an algorithm $\mathcal{A}$ with the stated guarantee exists. Given (G, k), build the metric above in polynomial time and run $\mathcal{A}$ with $z = 0$ and canonical priorities. Its size $N =$

$O(hk^3)$ and its center parameter is still k. All distances have constant-size encodings. The assumed running-time bound holds on every input, including those without an exact-service solution.

We then apply a polynomial-time verifier. Reject a failure report or malformed output. Otherwise check that the output is a centered partition of all points, uses at most k centers, and satisfies $d(x, \sigma(x)) \leq (2 - \delta)\rho_{k,0}(x)$ for every point. If G is a yes-instance, Lemma G.2 supplies exact feasibility, so $\mathcal{A}$ must pass this test. Lemma G.1 and the $\{0, 1, 2\}$ distance set imply

$$d(x, \sigma(x)) \leq 2 - \delta < 2 \quad \Longrightarrow \quad d(x, \sigma(x)) \leq 1,$$

so every passing output is in fact exactly 1-service-fair. If G is a no-instance, Lemma G.2 therefore implies that no output can pass the verifier, regardless of what $\mathcal{A}$ does outside its promise.

Thus "run $\mathcal{A}$ and verify" decides DOMINATING SET in time $f(k)N^{O(1)} = f'(k)h^{O(1)}$, contradicting W[2]-hardness unless FPT = W[2]. The verifier never inspects the MSR cost, so the lower bound is objective independent and applies unchanged to the radius-norm and MSD variants. □

Sharpness and scope. The threshold is literal on the reduction: because every pairwise distance is at most 2 and every priority is 1, choosing any one point as a center already gives service stretch 2. By contrast, every stretch strictly below 2 collapses to exact distance-1 service and exposes the dominating-set instance. Budget preservation is also essential to this theorem: selecting one filler from each of the $M = k + 2$ types gives exact service for every graph, so the construction does not rule out sub-two service under center augmentation. The result matches the service factor in the no-budget-violation regime of Theorems 11, 14, and 15, not their cost factors. It does not by itself exclude randomized FPT algorithms under the stated deterministic complexity assumption.