Certified Small Well-Separated Pair Decompositions in Practice

Guilherme D. da Fonseca
Aix-Marseille Université and LIS, France

Abstract

Given n points and $\varepsilon > 0$, a well-separated pair decomposition (WSPD) is a collection of *dumbbells* $\{X, Y\}$, where X and Y have diameter at most ε times their distance, such that every pair of points has one point in X and the other in Y for some dumbbell. In a *partition* WSPD every pair is covered by exactly one dumbbell, while a *cover* WSPD allows a pair to be covered by several. Callahan and Kosaraju give a partition WSPD with $O(n/\varepsilon^d)$ dumbbells in $\mathbb{R}^d$, but many point sets have far smaller WSPDs of both kinds. For $d \geq 2$, the known $O(1)$-approximations of the minimum size WSPD that preserve the separation ε have unspecified factors, above 10^7 for $d = 2$ by our estimates.

For cover WSPDs, we give an $O(n^2 \log n)$-time *sweep approximation algorithm* with a factor 40 in the plane for $\varepsilon \leq 1/16$. It extends to $\mathbb{R}^d$ with a factor $2^{O(d)}$. The sweep is also a *lower-bound tool*: its number of steps is a lower bound on the WSPD size. For partition WSPDs, we *analyze quadtree and fair split tree constructions*: in the plane and for $\varepsilon \leq 1/5$, they give constant approximation factors, the smallest of which is 252 for a randomly shifted and scaled quadtree.

In experiments with $\varepsilon = 1/20$, the number of sweep steps is at least 0.91 OPT for $n = 500$, where the optimum OPT is known, and a local search raises it to a lower bound of 0.99 OPT. A greedy variant of the sweep, which is still a certified 40-approximation, is at most 1.28 times the lower bound for $n = 10000$. Followed by a new local-search heuristic that shrinks the WSPD, it is within 0.02% of the optimum on every instance with $n = 500$, and at most 1.10 times the lower bound for $n = 10000$. We then show how to convert the cover WSPDs into partition WSPDs. For $n = 10000$, the conversion gives on average 1.13 times the lower bound, against 1.05 for the cover WSPD itself and 1.55 for the Callahan–Kosaraju partition WSPD.

2012 ACM Subject Classification Theory of computation → Computational geometry; Theory of computation → Approximation algorithms analysis

Keywords and phrases Well-separated pair decomposition, approximation algorithm, local search, set cover, algorithm engineering

Supplementary Material *Software*: `https://github.com/gfonsecabr/small-wspd-experiments`

Acknowledgements Use of AI: This paper was prepared with substantial help from the AI assistant Claude (Anthropic). It took part in developing the algorithms and proofs, writing the code and running the experiments, checking the references, and writing the text. The author checked all results and is responsible for the content.

1 Introduction

Let P be a set of n points in fixed dimension d, and let $\varepsilon \in (0, 1)$. We write $\|pq\|$ for the Euclidean distance between points p and q, $\operatorname{diam}(X)$ for the diameter of a set $X \subseteq P$, and $\operatorname{dist}(X, Y) = \min_{x \in X, y \in Y} \|xy\|$. A *dumbbell* $\{X, Y\}$ consists of two disjoint nonempty sets $X, Y \subseteq P$, its *sides*, and it is *ε-separated* if $\max\{\operatorname{diam}(X), \operatorname{diam}(Y)\} \leq \varepsilon \cdot \operatorname{dist}(X, Y)$.

An *ε-WSPD* of P is a collection $\mathcal{W}$ of ε-separated dumbbells that together cover all $\binom{n}{2}$ *pairs* of points, and $|\mathcal{W}|$ is its *size*. A *partition* WSPD covers every pair exactly once, and we use the term *cover* WSPD to stress that pairs may be covered by multiple dumbbells.

WSPDs were introduced in [7] and are now a standard tool in computational geometry [17, 19]. The Callahan–Kosaraju (CK) construction [7] computes a partition ε-WSPD of size $O(n/\varepsilon^d)$ in $O(n \log n + n/\varepsilon^d)$ time, with dumbbells given as pairs of tree nodes. Several

applications of WSPDs only use that every pair is covered at least once, so they work with cover WSPDs. Choosing one edge per dumbbell gives a t-spanner with $|\mathcal{W}|$ edges [6, 17, 19], and hence an approximate minimum spanning tree [6, 19]; such sparse edge sets also speed up solvers for geometric tour problems [12]. The approximate diameter [19, 14] is found by inspecting every dumbbell once. Other applications, such as the computation of n-body potential fields [7], approximate distance selection [3, 19], and the approximation of the sum of all pairwise distances [14], need a partition WSPD.

The bound $O(n/\varepsilon^d)$ is a worst-case bound, and many point sets have much smaller WSPDs. For example, the n points $p_i = (i, (4/\varepsilon)^i)$ have an ε-WSPD with only $n-1$ dumbbells even in the partition version [5]. Since spanners and many other structures built from a WSPD inherit its size, a WSPD that is as small as possible is desirable.

The input of MINWSPD consists of P and ε, and the goal is a cover ε-WSPD of P of minimum size, which we denote by OPT_ε, or by OPT if ε is clear from the context. A partition WSPD is in particular a cover WSPD. Therefore, every approximation factor with respect to OPT_ε also holds with respect to the partition optimum.

Previous work. MINWSPD was introduced in [5]. It is NP-hard in the plane when ε may depend on n, and in general metric spaces no polynomial-time algorithm achieves a factor n^c, for some constant $c > 0$, unless P = NP [5]. Table 1 lists the known approximation algorithms. For $d \geq 2$, the only one stated for the separation ε is based on the net-tree WSPD [15], and its factor $2^{O(d)}$ is unspecified; our rough estimate from its proof is 10^{11} for $d = 2$. Greedy set cover over all maximal ε-separated dumbbells keeps the separation and gives an $O(\log n)$-approximation whenever their number is polynomial in n, which holds on the line but is not known for $d \geq 2$ [5]. The constant-factor algorithm of [16] works in every metric space but relaxes the separation (the constants 7 and 33 follow from its proof, see also [5]). In $\mathbb{R}^d$, the relaxation can be undone: a grid cuts each side of a $c\varepsilon$-separated dumbbell into at most $\lceil c\sqrt{d}\,\rceil^d$ pieces whose diameter is at most $1/c$ times the larger diameter of the two sides, and the pieces form ε-separated dumbbells. For $d = 2$ and $\varepsilon < 1/33$, the factor becomes $7 \cdot 47^4 \approx 3.4 \cdot 10^7$. For $d \geq 2$, the question of a $(1+\delta)$-approximation, or a simple $O(1)$-approximation, was raised in [5]. A bicriteria approximation scheme, which computes a $(1+\delta)\varepsilon$-WSPD of size at most $(1+\delta)\mathrm{OPT}_\varepsilon$ in $\mathbb{R}^d$ and in doubling metrics, is given in [8], but its running time is doubly exponential in d/δ.

Our results. Table 1 places our algorithms among the known approximation results.

1. **Improved cover WSPD approximation factors** (Theorem 3). For points in the plane and $\varepsilon \leq 1/16$, a sweep algorithm computes a 40-approximation in $O(n^2 \log n)$ time and $O(n^2)$ space. It extends to $\mathbb{R}^d$ for every fixed d (Theorem 11) with the same time and space bounds, with a factor 2048 for $d = 3$ and $2^{O(d)}$ in general. It generalizes the 3-approximation of [5] for $d = 1$ and is a simple $O(1)$-approximation, as asked there.
2. **Improved partition WSPD approximation factors** (Section 7). For points in the plane and $\varepsilon \leq 1/5$, the CK construction on a fair split tree [7] outputs at most $1024\,\mathrm{OPT}$ dumbbells in $O(n \log n + \mathrm{OPT})$ time. The same size bound holds for the two variants of our experiments: *CK box* accepts two tree nodes if their bounding boxes are ε-separated, and *CK exact*, or just *CK*, if their point sets are. In the same time, the maximal separated pairs of same-level quadtree cells, with separation tested on the cells, form a partition ε-WSPD of size at most $529\,\mathrm{OPT}$, and of expected size at most $252\,\mathrm{OPT}$ for a randomly shifted and scaled quadtree. The deterministic bounds extend to $\mathbb{R}^d$.
3. **Small cover WSPD in practice** (Section 5). A greedy variant of the sweep keeps the

approximation factor 40 and runs in $O(n^3)$ time (Theorem 7). The number of steps of the sweep is a lower bound on OPT. A local search on the anchors of the steps enlarges it to a lower bound LB (Section 3.3), which certifies the quality of any WSPD. For $\varepsilon = 1/20$ and $n = 500$, where OPT is known, the greedy sweep is on average 1.03 OPT, against 1.17 OPT for the CK WSPD, and the lower bound is at least 0.99 OPT. For $n = 10000$, where OPT is out of reach, the greedy sweep and the CK WSPD are on average 1.22 LB and 1.55 LB. We present a local-search heuristic (Section 3.2) that post-processes any given WSPD with a single move, *reshape*, never increasing its size. For $n = 500$, it brings the sweep to within 0.02% of OPT, and it reaches OPT on 13 of the 22 instances. For $n = 10000$, the sweep followed by local search is at most 1.10 LB and on average 1.05 LB.

4. **Small partition WSPD in practice** (Section 5). Converting the cover WSPD of the sweep followed by local search into a partition WSPD gives on average 1.006 OPT for $n = 500$ and 1.13 LB for $n = 10000$, against 1.17 OPT and 1.55 LB for CK.

Table 1 Approximation algorithms for MINWSPD. *Both*: outputs a partition WSPD. *: expected.

Setting	Problem	Size	Separation	Time	Reference
$\mathbb{R}^1$	cover	$(1+\delta)\,\mathrm{OPT}_\varepsilon$	ε	$n^{O(1/\delta^2)}$	[5]
$\mathbb{R}^1$	cover	$3\,\mathrm{OPT}_\varepsilon$	ε	$O(n \log n)$	[5]
$\mathbb{R}^1$	both	$10\,\mathrm{OPT}_\varepsilon$	ε	polynomial	[5]
$\mathbb{R}^d$	both	$2^{O(d)}\mathrm{OPT}_\varepsilon$	ε	$2^{O(d)}(n \log n + \mathrm{OPT})^*$	[5, 15]
$\mathbb{R}^d$	cover	$(1+\delta)\,\mathrm{OPT}_\varepsilon$	$(1+\delta)\varepsilon$	$f(d,\delta)\,n^6$	[8]
Any metric	cover	$7\,\mathrm{OPT}_\varepsilon$	33ε	polynomial	[16]
$\mathbb{R}^2$, $\varepsilon \le 1/16$	cover	$40\,\mathrm{OPT}_\varepsilon$	ε	$O(n^2 \log n)$	Theorem 3
$\mathbb{R}^2$, $\varepsilon \le 1/5$	both	$1024\,\mathrm{OPT}_\varepsilon$	ε	$O(n \log n + \mathrm{OPT})$	[7], Section 7
$\mathbb{R}^2$, $\varepsilon \le 1/5$	both	$529\,\mathrm{OPT}_\varepsilon$	ε	$O(n \log n + \mathrm{OPT})$	Section 7
$\mathbb{R}^2$, $\varepsilon \le 1/5$	both	$252\,\mathrm{OPT}_\varepsilon{}^*$	ε	$O(n \log n + \mathrm{OPT})$	Section 7
$\mathbb{R}^d$, $\varepsilon \le 1/16$	cover	$2^{O(d)}\mathrm{OPT}_\varepsilon$	ε	$O(n^2 \log n)$	Theorem 11

Section 2 presents the sweep and its analysis, Section 3 a greedy variant of the sweep and the local search for smaller WSPDs and larger lower bounds, and Section 4 the extension of the sweep to higher dimensions. Section 5 reports the experiments, in the plane and, in Section 5.3, in $\mathbb{R}^3$. Section 6 extends the sweep to larger ε. The partition WSPD constructions are analyzed in Section 7.

2 Sweep Algorithm

This section presents the sweep 40-approximation algorithm for points in the plane. Let $\prec$ denote the lexicographic order on $\mathbb{R}^2$ (by first coordinate, then by second coordinate), and write $p \preceq q$ if $p \prec q$ or $p = q$. We write $\mathbb{B}(c, r)$ for the closed disk of radius r centered at c.

Pieces. For a disk $\mathbb{B}$ of radius ρ, we use a fixed partition of $\mathbb{B}$ into 8 pieces: the concentric disk of radius 0.434ρ, and the 7 sectors of angle $2\pi/7$ of the annulus between radii 0.434ρ and ρ. For the half-disk $\mathbb{B}^+$, which consists of the points of $\mathbb{B}$ whose first coordinate is at least that of its center, we use 5 pieces: the concentric half-disk of radius 0.4ρ, and the 4 sectors of angle $\pi/4$ of the half-annulus between radii 0.4ρ and ρ. Points on the boundary between two pieces are assigned to one of them arbitrarily, so the pieces partition the disk and the half-disk. The inner radii are chosen so that all pieces have roughly the same diameter.

▶ **Lemma 1.** *Every piece of the disk partition has diameter at most* 0.868ρ*, and every piece of the half-disk partition has diameter at most* 0.8ρ*.*

Proof. Assume $\rho = 1$. The diameter of a set is attained by two extreme points of its convex hull. For an annular sector with angle less than π, these are the two inner corners and the points of the outer arc. Two points of the outer arc are at distance at most $2\sin(\varphi/2)$, and the distance from an inner corner to a point of the outer arc increases with the angle between them. Hence an annular sector with inner radius $a < 1$, outer radius 1, and angle $\varphi \le \pi/2$ has diameter $\max\{2\sin(\varphi/2), \sqrt{1 + a^2 - 2a\cos\varphi}, 1 - a\}$. For $\varphi = 2\pi/7$ and $a = 0.434$, these values are 0.8678, 0.804, and 0.566. For $\varphi = \pi/4$ and $a = 0.4$, they are 0.765, 0.771, and 0.6. The central pieces have diameters 0.868 and 0.8. ◀

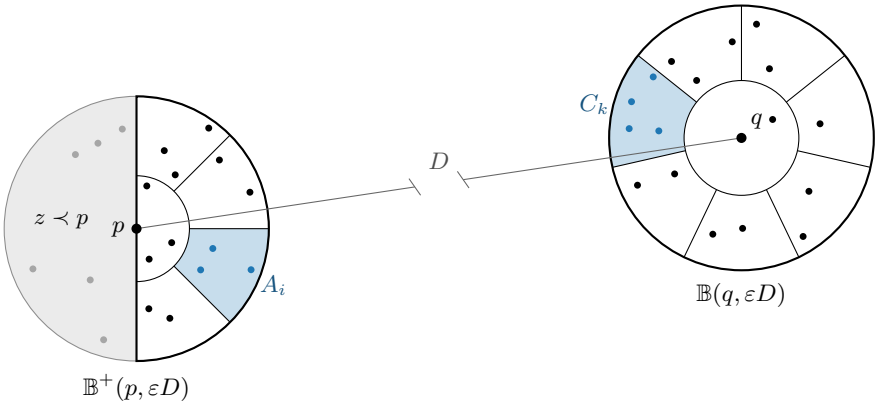


Figure 1 A step of the sweep with anchor $\{p, q\}$. The step outputs a dumbbell $\{A_i, C_k\}$, such as the one highlighted, for every two nonempty pieces.

Algorithm. Initially, no pair is covered. The algorithm processes the points $p \in P$ in the order $\prec$, and for each such p the points $q \succ p$ in arbitrary order. If the pair $\{p, q\}$ is not yet covered, the algorithm performs a *step* with *anchor* $\{p, q\}$, as follows. Let $D = \|pq\|$. Let $A_1, \ldots, A_5$ be the intersections of $\{z \in P : z \succeq p\}$ with the pieces of the half-disk $\mathbb{B}^+(p, \varepsilon D)$, and let $C_1, \ldots, C_8$ be the intersections of P with the pieces of the disk $\mathbb{B}(q, \varepsilon D)$. The step outputs every dumbbell $\{A_i, C_k\}$ whose sides are both nonempty (Figure 1), at most $5 \cdot 8 = 40$ dumbbells. We call these dumbbells the *fixed collection* of the step.

▶ **Observation 2** (Locality). *If an* ε*-separated dumbbell* $\{X, Y\}$ *covers* $\{x, y\}$ *and* $\ell = \operatorname{dist}(X, Y)$*, then* $\ell \le \|xy\| \le (1 + 2\varepsilon)\ell$*, and each side lies in the disk of radius* $\varepsilon\ell$ *centered at any of its points.*

▶ **Theorem 3.** *For* $P \subset \mathbb{R}^2$ *and* $\varepsilon \le 1/16$*, the sweep performs at most* OPT_ε *steps and outputs an* ε*-WSPD of size at most* $40\,\mathrm{OPT}_\varepsilon$ *in* $O(n^2 \log n)$ *time and* $O(n^2)$ *space.*

Proof. *Validity.* The disks $\mathbb{B}(p, \varepsilon D)$ and $\mathbb{B}(q, \varepsilon D)$ are disjoint since $\varepsilon < 1/2$. By Lemma 1, $\operatorname{diam}(A_i), \operatorname{diam}(C_k) \le 0.868\,\varepsilon D$, while $\operatorname{dist}(A_i, C_k) \ge (1 - 2\varepsilon)D$. Since $0.868 \le 1 - 2\varepsilon$ for $\varepsilon \le 1/16$, every output dumbbell is ε-separated. Every pair $\{p, q\}$ with $p \prec q$ is covered once p is processed, since the step with anchor $\{p, q\}$ covers it (p lies in some A_i, q in some C_k).

Invariant. By the validity argument and the processing order, when the step with anchor $\{p, q\}$ starts, every pair $\{x, y\}$ with $\min_\prec\{x, y\} \prec p$ is already covered.

Charging. Fix a minimum ε-WSPD $\mathcal{W}^*$. Charge the step with anchor $\{p, q\}$ to some $Z = \{X, Y\} \in \mathcal{W}^*$ covering $\{p, q\}$, with $p \in X$ and $q \in Y$. Let $\ell = \operatorname{dist}(X, Y) \le D$. By Observation 2, $X \subseteq \mathbb{B}(p, \varepsilon\ell) \subseteq \mathbb{B}(p, \varepsilon D)$ and $Y \subseteq \mathbb{B}(q, \varepsilon D)$.

Let $\{x, y\}$, with $x \in X$ and $y \in Y$, be a pair that is not covered before the step. By the invariant, $x \succeq p$, so $x \in \mathbb{B}^+(p, \varepsilon D)$ lies in some A_i, and y lies in some C_k. Hence, after the step, every pair covered by Z is covered. In particular, no later anchor is covered by Z, and Z is charged only once. Hence distinct steps are charged to distinct dumbbells of $\mathcal{W}^*$, and there are at most $|\mathcal{W}^*|$ steps. Since each step outputs at most $5 \cdot 8 = 40$ dumbbells, the output has size at most $40\,|\mathcal{W}^*|$. The running time is analyzed below. ◀

The condition $\varepsilon \le 1/16$ keeps constants simple. With inner radius $\sin(\pi/7)\rho$ instead of 0.434ρ, every disk piece has diameter at most $2\sin(\pi/7)\rho$, and the factor 40 only needs $\varepsilon \le 1/2 - \sin(\pi/7) \approx 1/15.12$. For larger ε, more pieces give larger factors (Section 6).

Two pairs are *compatible* if a single ε-separated dumbbell covers both of them.

▶ **Observation 4** (Heredity). *If $\{X, Y\}$ is ε-separated, then so is $\{X', Y'\}$ for all nonempty $X' \subseteq X$ and $Y' \subseteq Y$. Hence $\{a, a'\}$ and $\{b, b'\}$ are compatible if and only if $\{\{a, b\}, \{a', b'\}\}$ or $\{\{a, b'\}, \{a', b\}\}$ has disjoint sides and is ε-separated.*

For the step with anchor $\{p, q\}$, let R_{pq} be the set of pairs $\{x, y\}$ that are uncovered when the step starts, are compatible with $\{p, q\}$, and satisfy $x, y \succeq p$.

▶ **Observation 5.** *If every step of the sweep, with anchor $\{p, q\}$, covers R_{pq}, then the anchors are pairwise incompatible, so the number of steps is at most* OPT_ε.

Proof. Let $\{p', q'\}$ be the anchor of a step performed after the step with anchor $\{p, q\}$. Both p' and q' are $\succeq p$, and $\{p', q'\}$ was uncovered when the step with anchor $\{p, q\}$ was performed. If the two anchors were compatible, that step would have covered $\{p', q'\}$, a contradiction. Since no ε-separated dumbbell covers two anchors, every ε-WSPD has at least as many dumbbells as there are steps. ◀

The fixed collection covers R_{pq} for every $\varepsilon < 1/2$ (the condition $\varepsilon \le 1/16$ is needed only for its dumbbells to be ε-separated). Indeed, the charging argument uses only that Z is an ε-separated dumbbell that covers $\{p, q\}$, and if $\{x, y\}$ is compatible with $\{p, q\}$, then, by Observation 4, $\{\{p, x\}, \{q, y\}\}$ or $\{\{p, y\}, \{q, x\}\}$ is such a dumbbell.

▶ **Lemma 6.** *Let $\varepsilon \le 1/16$. Every pair is covered by the output dumbbells of at most* 360 *steps of the sweep, and by at most one output dumbbell of each step.*

Proof. *At most one dumbbell per step.* Let $\{x, y\}$ be covered in the step with anchor $\{p, q\}$, and let $D = \|pq\|$. The disks $\mathbb{B}(p, \varepsilon D)$ and $\mathbb{B}(q, \varepsilon D)$ are disjoint, so one endpoint, say x, lies in $\mathbb{B}(p, \varepsilon D)$ and the other in $\mathbb{B}(q, \varepsilon D)$. The pieces partition each disk, so exactly one output dumbbell of the step covers $\{x, y\}$.

At most 360 *steps.* Let $L = \|xy\|$. We split the steps whose output covers $\{x, y\}$ into two classes, according to whether x lies in the disk around the first or the second point of the anchor, and we bound the size of each class by 180. Consider the class of steps with anchors $\{p_j, q_j\}$ such that $x \in \mathbb{B}(p_j, \varepsilon D_j)$ and $y \in \mathbb{B}(q_j, \varepsilon D_j)$, where $D_j = \|p_j q_j\|$. By the triangle inequality, $(1 - 2\varepsilon)D_j \le L \le (1 + 2\varepsilon)D_j$. Therefore, all p_j lie in $\mathbb{B}(x, r)$ and all q_j lie in $\mathbb{B}(y, r)$, where $r = \varepsilon L/(1 - 2\varepsilon)$. Since $2r < L$ for $\varepsilon < 1/4$, these two disks are disjoint, so the sets $\{p_j, p_k\}$ and $\{q_j, q_k\}$ below are disjoint. For $j \ne k$, the anchors are incompatible

by Observation 5, so the dumbbell $\{\{p_j, p_k\}, \{q_j, q_k\}\}$ is not ε-separated. Since its distance is at least $L - 2r$,

$$\max\{\|p_j p_k\|, \|q_j q_k\|\} > s, \qquad \text{where } s = \varepsilon(L - 2r) = r\,(1 - 4\varepsilon) \geq 3r/4.$$

For $j \neq k$, if the products $\mathbb{B}(p_j, s/2) \times \mathbb{B}(q_j, s/2)$ and $\mathbb{B}(p_k, s/2) \times \mathbb{B}(q_k, s/2)$ of disks shared a point (u, v), then $\|p_j p_k\| \leq \|p_j u\| + \|u p_k\| \leq s$ and, similarly, $\|q_j q_k\| \leq s$. Hence these products are pairwise disjoint. They are contained in $\mathbb{B}(x, r + s/2) \times \mathbb{B}(y, r + s/2)$. The measure of a product of two disks is the product of their areas, so comparing areas, the class has at most $\big((r + s/2)/(s/2)\big)^4 = (1 + 2r/s)^4 \leq (11/3)^4 < 181$ steps. ◀

Proof of the running time in Theorem 3. We store the covered pairs in an $n \times n$ table. For every point $c \in P$, we also store the list of the points of P sorted by distance from c. These lists take $O(n^2 \log n)$ time to build and $O(n^2)$ space. The algorithm inspects each pair once, when its $\prec$-smaller endpoint is processed, and checks the table in $O(1)$ time. This takes $O(n^2)$ time in total. When the algorithm starts processing p, it extracts from the list of p the points $z \succeq p$, in $O(n)$ time. In a step with anchor $\{p, q\}$, the points of $\mathbb{B}(q, \varepsilon D)$ form a prefix of the list of q, and the points $z \succeq p$ of $\mathbb{B}(p, \varepsilon D)$ form a prefix of this extracted list. The step scans these two prefixes and assigns each point to its piece in $O(1)$ time (in the real RAM model, since the sector boundaries have irrational slopes), so it takes $O(1 + m)$ time, where m is the number of scanned points. Every scanned point lies in an output dumbbell of the step: a point of A_i together with the piece C_k that contains q, and a point of C_k together with the piece A_i that contains p. Hence $m \leq \sum(|A| + |C|)$, where the sum is over the output dumbbells $\{A, C\}$ of the step. Marking the pairs covered by $\{A, C\}$ takes $O(|A| \cdot |C|)$ time, and $|A| + |C| \leq |A| \cdot |C| + 1$. Since a step outputs at most 40 dumbbells, all steps together take time linear in the number of steps plus the total number of pair markings. There are at most $\binom{n}{2}$ steps, and by Lemma 6 every pair is marked at most 360 times, so all steps take $O(n^2)$ time. Sorting dominates the time, and the table and the lists the space. ◀

Dependence on ε. The bound of Theorem 3 can also be stated in terms of ε. By Observation 5, the number of steps is at most $\mathrm{OPT}_\varepsilon = O(n/\varepsilon^2)$ by the CK construction [7]. Hence, even if every step scans all of P without sorting, the sweep takes $O(n^2/\varepsilon^2)$ time. Together with Theorem 3, this gives $O(n^2 \min\{\log n, 1/\varepsilon^2\})$ time.

3 Practical Algorithms

We present a greedy variant of the sweep (Section 3.1), and we use local search in two ways: to reduce the size of a given WSPD (Section 3.2), and to enlarge the lower bound given by the steps of the sweep (Section 3.3). They give large practical improvements, but do not improve the approximation bounds.

3.1 Greedy Sweep

The proofs of validity and of the approximation factor in Theorem 3 use only one property of a step with anchor $\{p, q\}$: it covers R_{pq}. This set contains $\{p, q\}$ itself and, by the invariant, every uncovered pair covered by an optimal dumbbell that covers $\{p, q\}$. Therefore, these two proofs hold verbatim if each step outputs *any* collection of ε-separated dumbbells covering R_{pq} whose size is at most that of the fixed collection. The running-time analysis does not carry over, and we analyze the greedy variant separately below.

The sweep of Theorem 3, which we call the *fixed sweep*, is of theoretical interest only: every step outputs a dumbbell for every two nonempty pieces, even if a single dumbbell covers R_{pq}. The *greedy sweep* keeps the proof but not the collection. It covers R_{pq} with dumbbells grown from pairs of R_{pq}, and falls back to the fixed collection only when that collection is smaller. To *grow* a dumbbell from a *seed* pair $\{a, b\}$, we start from $\{\{a\}, \{b\}\}$ and repeatedly add a point that *fits*, that is, a point whose distance to every point of one side is at most $\varepsilon\|ab\|$ and whose distance to every point of the other side is at least $\|ab\|$; the point joins the first side. The dumbbell remains ε-separated, with $\{a, b\}$ as a closest cross pair. Among the points that fit, we choose one that maximizes the number of newly covered pairs of R_{pq}, breaking ties by the number of newly covered pairs overall, and we stop when no point fits. If $|R_{pq}| \leq 16$, the step grows one dumbbell from each pair of R_{pq} and covers R_{pq} by greedy set cover over these candidates. Otherwise, it grows a dumbbell from $\{p, q\}$, and then repeatedly grows a dumbbell from the closest pair of R_{pq} that is not yet covered, until R_{pq} is covered. The second rule avoids growing $|R_{pq}|$ dumbbells when R_{pq} is large.

For $\varepsilon \leq 1/16$, the number of dumbbells grown in a step is capped at 40. If R_{pq} is not covered by then, the step outputs the fixed collection. The cap does not change the output: a greedy cover with more than 40 dumbbells is larger than the fixed collection, which has at most 40 dumbbells, so the step would output the fixed collection anyway. Finally, the greedy sweep deletes, one at a time, the dumbbells whose pairs are all covered by other dumbbells.

▶ **Theorem 7.** *For $P \subset \mathbb{R}^2$ and $\varepsilon \leq 1/16$, the greedy sweep performs at most OPT_ε steps and outputs an ε-WSPD of size at most $40\,\mathrm{OPT}_\varepsilon$ in $O(n^3)$ time and $O(n^2)$ space.*

Proof. Grown dumbbells are ε-separated by construction, and every step covers R_{pq} with at most 40 dumbbells, so the size bounds follow as explained above. The running time is proved below. ◀

We prove the time and space bounds of Theorem 7. The proof holds in every dimension. Let $\lambda = 17/8$. For a step with anchor $\{p, q\}$ and $D = \|pq\|$, let $B_p = P \cap \mathbb{B}(p, \lambda\varepsilon D)$ and $B_q = P \cap \mathbb{B}(q, \lambda\varepsilon D)$. All the work of the step takes place in these two sets. Indeed, every seed $\{a, b\}$ is $\{p, q\}$ or a pair of R_{pq}, so some ε-separated dumbbell covers both $\{a, b\}$ and $\{p, q\}$, and its distance is at most D. By Observation 2, we may name the seed so that $a \in \mathbb{B}(p, \varepsilon D)$, $b \in \mathbb{B}(q, \varepsilon D)$, and $\|ab\| \leq (1 + 2\varepsilon)D$. A point that fits is within distance $\varepsilon\|ab\|$ of a or b, hence within distance $\varepsilon(2 + 2\varepsilon)D \leq \lambda\varepsilon D$ of p or q. Therefore, every pair of R_{pq} and every dumbbell that the step grows or outputs has one side in B_p and the other in B_q. The next lemma extends the second part of Lemma 6 to these larger disks.

▶ **Lemma 8.** *Let $P \subset \mathbb{R}^d$, $\varepsilon \leq 1/16$, and $\lambda = 17/8$. Every pair $\{x, y\}$ satisfies $x \in \mathbb{B}(p, \lambda\varepsilon D)$ and $y \in \mathbb{B}(q, \lambda\varepsilon D)$, where $D = \|pq\|$, for the anchors $\{p, q\}$ of at most $2 \cdot 11^{2d}$ steps of the greedy sweep, in one of the two orders of p and q.*

Proof. Let $L = \|xy\|$. Consider the steps with anchors $\{p_j, q_j\}$ such that $x \in \mathbb{B}(p_j, \lambda\varepsilon D_j)$ and $y \in \mathbb{B}(q_j, \lambda\varepsilon D_j)$, where $D_j = \|p_j q_j\|$; the steps with the roles of x and y exchanged are counted in the same way. By the triangle inequality, $L \geq (1 - 2\lambda\varepsilon)D_j$, so $p_j \in \mathbb{B}(x, r)$ and $q_j \in \mathbb{B}(y, r)$, where $r = \lambda\varepsilon L/(1 - 2\lambda\varepsilon)$. Since $\lambda\varepsilon \leq 17/128 < 1/4$, we have $2r < L$, so these two balls are disjoint. The greedy sweep covers R_{pq} in every step, so by Observation 5, the dumbbell $\{\{p_j, p_k\}, \{q_j, q_k\}\}$ is not ε-separated for $j \neq k$. Its distance is at least $L - 2r$, so

$$\max\{\|p_j p_k\|, \|q_j q_k\|\} > s, \qquad \text{where } s = \varepsilon(L - 2r) = \frac{r\,(1 - 4\lambda\varepsilon)}{\lambda} \geq \frac{15}{68}\,r.$$

Hence the products $\mathbb{B}(p_j, s/2) \times \mathbb{B}(q_j, s/2) \subset \mathbb{R}^{2d}$ are pairwise disjoint. They are contained in $\mathbb{B}(x, r + s/2) \times \mathbb{B}(y, r + s/2)$. Comparing volumes, there are at most $(1 + 2r/s)^{2d} \leq (151/15)^{2d} < 11^{2d}$ such steps. ◀

Proof of the running time in Theorem 7. We store the covered pairs and the sorted lists as in the proof of Theorem 3, in $O(n^2 \log n)$ time, and inspect every pair once. Consider a step, and let $N = |B_p| + |B_q|$ and $M = |B_p| \cdot |B_q|$. Since $\lambda\varepsilon < 1/2$, the sets B_p and B_q are disjoint prefixes of the lists of p and q. Enumerating R_{pq} takes $O(M)$ time, and we mark its pairs in the table, so that membership in R_{pq} is tested in $O(1)$ time. To grow a dumbbell, we maintain for every point of $B_p \cup B_q$ whether it fits and how many pairs it would newly cover. When a point joins a side, these values change only by its own contribution, so they are updated in $O(N)$ time, and a dumbbell is grown in $O(N^2)$ time. Its seed is found by scanning R_{pq} in $O(M)$ time. The greedy set cover for $|R_{pq}| \leq 16$ takes $O(N)$ time, and marking each of the at most 40 output dumbbells takes $O(M)$ time. Since $p \in B_p$ and $q \in B_q$, we have $N \leq 2M$, and $N \leq n$, so the step takes $O(N^2 + M) = O(nM)$ time. By Lemma 8, every pair contributes to M in $O(1)$ steps, so the sum of M over all steps is $O(n^2)$. Hence the greedy sweep runs in $O(n^3)$ time, and the table and the lists take $O(n^2)$ space. ◀

3.2 Local Search for Smaller WSPDs

The sweep of Section 2 never revisits a dumbbell once it is output. We now describe a local-search heuristic that tries to reduce the size of any given ε-WSPD, for instance the CK WSPD or the output of the sweep. It uses a single move, *reshape*, and never increases the size. It uses no solver: unlike exact and heuristic set-cover solvers, it never builds the set-cover instance of MINWSPD, whose size limits these solvers in practice (Section 5).

Reshape. A pair is *private* to a dumbbell of $\mathcal{W}$ if that dumbbell is the only one covering it, and a dumbbell without private pairs is *redundant* and can be deleted. A point p can *join* the side X of an ε-separated dumbbell $\{X, Y\}$ if $p \notin X \cup Y$ and $\{X \cup \{p\}, Y\}$ is still ε-separated. Unlike the greedy sweep of Section 3.1, which keeps the closest cross pair fixed, we allow $\mathrm{dist}(X \cup \{p\}, Y)$ to be smaller than $\mathrm{dist}(X, Y)$. To *grow* a dumbbell, we repeatedly add a point that can join one of its sides, choosing one that newly covers the most pairs private to other dumbbells (ties are broken at random), until no point can join.

The move (Figure 2) is applied to a *target* dumbbell $T \in \mathcal{W}$. If $T = \{X, Y\}$ is redundant, delete it. Otherwise, replace T by the *subdumbbell spanned by its private pairs*, the dumbbell $\{X', Y'\}$ where $X' \subseteq X$ and $Y' \subseteq Y$ consist of the endpoints of these pairs, which is ε-separated by Observation 4. Grow it, and then delete, one at a time, the dumbbells that the new dumbbell makes redundant. Shrinking T first releases the points that lie in no private pair of T, and with them most pairs that other dumbbells also cover, so the new dumbbell can grow in a different direction than the old one. Its growth rule favors pairs that are private to other dumbbells, so repeated reshapes tend to make dumbbells redundant.

The targets are taken in passes over $\mathcal{W}$, each in a new random order, until the time budget ends. Deleted dumbbells are skipped, and dumbbells created during a pass wait for the next one. At the end, the remaining redundant dumbbells are deleted.

Large neighborhood search. For comparison, we also use an integer-programming move. The *large neighborhood search (LNS)* move removes a target T and its k nearest dumbbells, which leaves a set U of uncovered pairs. It re-covers U by a minimum number of dumbbells, chosen among the removed ones and, if $|U| \leq u$, maximal ε-separated dumbbells whose

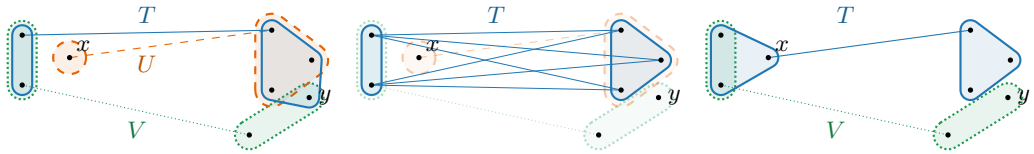


Figure 2 A reshape of T for $\varepsilon = 1/3$, showing three dumbbells, each as its two sides and a closest cross pair. (a) With y in T, x cannot join T: the right side would be too wide for the smaller distance. (b) The pairs private to T (thin lines) span the subdumbbell, and y is released. (c) Free growth adds x, which covers the pairs private to U, so U is deleted.

closest cross pair is in U, with CPLEX. The change is accepted only if it uses fewer dumbbells than it removes. LNS takes its targets in the same passes as reshape, and its neighborhoods and parameters are given in Section 5.1.

▶ **Observation 9.** *Reshape and LNS keep $\mathcal{W}$ an ε-WSPD and do not increase $|\mathcal{W}|$.*

Proof. In a reshape, the new dumbbell contains the subdumbbell spanned by the private pairs of T, so it covers every pair that only T covered. It is ε-separated by Observation 4 and because every added point keeps it ε-separated. Only redundant dumbbells are deleted, and redundancy is checked again before each deletion, so every pair stays covered. A reshape replaces T by one dumbbell and otherwise only deletes dumbbells, so $|\mathcal{W}|$ does not increase. LNS adds ε-separated dumbbells that cover U, and only if the size decreases. ◀

Implementation. A reshape must decide which pairs are private, which dumbbells become redundant, and which points can join a side, each in constant time per pair or point. A table stores, for every pair, the number of dumbbells of $\mathcal{W}$ that cover it. Inserting or deleting $\{X, Y\}$ updates it in $O(|X| \cdot |Y|)$ time. A pair is private if its count is 1, so finding private pairs, testing redundancy, and computing the gain of a point take $O(1)$ time per pair. Shrinking T and deleting dumbbells only decrease counts, so a dumbbell becomes redundant only if the new T covers one of its private pairs. To find it, we need the *owner* of such a pair, the dumbbell to which it was private. Lists of the dumbbells containing each point are expensive: at $n = 10000$, a point lies in hundreds to thousands of dumbbells. Instead, a second table stores, for every pair, the bitwise exclusive or of the indices of the dumbbells that cover it, updated in the same loop as the counts. Since $k \oplus k = 0$, deleting a dumbbell cancels its insertion, so for a pair with count 1 the entry is the index of its owner. Hence the dumbbells that become redundant are found while T is inserted, at no extra asymptotic cost. In a preliminary test with $n = 10000$, this made reshape more than an order of magnitude faster than point-to-dumbbell lists. A point p that can join X satisfies $\|px\| \leq \varepsilon \operatorname{dist}(X, Y)$ for every $x \in X$, so the candidates are the points within distance $\varepsilon \operatorname{dist}(X, Y)$ of the initial X or Y, found by range queries in a kd-tree. For $\varepsilon < 1/2$, each can join only the side it is close to. Diameters only increase and the distance between the sides only decreases, so a candidate that cannot join is discarded for good. For each candidate, we maintain its largest distance to its side, its smallest distance to the other side, and its gain, so that after each addition, updating them and testing separation exactly take constant time per candidate. The points are relabeled in the order of a Hilbert curve, and the tables are packed upper triangles of $\binom{n}{2}$ entries of 4 bytes, stored by rows, so the entries of the pairs of a dumbbell are close in memory.

3.3 Local Search for Larger Lower Bounds

The *compatibility graph* G_ε has the pairs as vertices and an edge between any two compatible pairs. By the proof of Observation 5, every independent set I of G_ε gives the lower bound $|I| \le \mathrm{OPT}_\varepsilon$. The anchors of the greedy sweep form such a set, and we call their number LB_0. We enlarge this set by a local search for large independent sets, starting from the anchors. For the current set I, a pair is *free* if it has no neighbor in I, and *tight* for $e \in I$ if e is its only neighbor in I. These terms and the 1-for-2 swaps below follow the local search of Andrade, Resende, and Werneck [1] for maximum independent sets. The search has three stages.

1. Add free pairs to I until none is left.
2. Repeat passes over I. For each $e \in I$, if two pairs f and g that are tight for e are incompatible with each other, replace e by f and g (a 1-*for*-2 *swap*), and add the pairs that have become free. Stop after a pass without swaps, and let $\mathrm{LB}_1 = |I|$.
3. Make 30 more passes in which, in addition, a pair e without a swap is replaced, with probability 1/2, by a random pair that is tight for e, followed again by the free pairs. These *plateau moves* [18, 13] do not change $|I|$, but they change which pairs are tight, so later passes find new swaps. The lower bound LB is $|I|$ at the end.

A pair joins I only if it is free or tight for the leaving pair, so I stays independent and $\mathrm{LB}_0 \le \mathrm{LB}_1 \le \mathrm{LB} \le \mathrm{OPT}_\varepsilon$.

Implementation. The search keeps a table with, for every pair w, the number $t(w)$ of pairs of I compatible with w. Hence w is free if $t(w) = 0$, and tight for e if $t(w) = 1$ and w is compatible with e. If $\{x', y'\}$ is compatible with $e = \{x, y\}$, then, by Observations 2 and 4 and after possibly exchanging x' and y', we have $x' \in \mathbb{B}(x, \varepsilon\|xy\|)$ and $y' \in \mathbb{B}(y, \varepsilon\|xy\|)$. As in the proof of Theorem 3, these points are prefixes of the lists of the points sorted by distance from x and from y, which gives the enumeration, and the update of the table when e joins or leaves I, in $O(M_e)$ time, where M_e is the product of the numbers of points in the two disks. In a pass, we skip e if the dumbbell formed by the first points and the second points of the tight pairs of e, oriented as above, is ε-separated: then all these pairs are pairwise compatible by Observation 4. Otherwise, we test pairs of tight pairs for compatibility, at most $2 \cdot 10^5$ of them per e and pass. Because of this limit, LB_1 need not be a local optimum for 1-for-2 swaps.

▶ **Proposition 10.** *Let $\varepsilon \le 1/16$. The search takes $O(n^2 \log n + kn^3)$ time and $O(n^2)$ space, where k is the number of passes, and $k \le \mathrm{LB}_1 - \mathrm{LB}_0 + 31$.*

Proof. Sorting takes $O(n^2 \log n)$ time, and the lists and the table take $O(n^2)$ space. The first stage scans all pairs once, and it adds pairs to I without removing any, so its updates take $O(\sum_{e \in I} M_e)$ time for the final set I. The second part of the proof of Lemma 6 uses only that the anchors are pairwise incompatible. Applied to I, it shows that every pair lies in the product of the two disks of at most 360 pairs of I, so $\sum_{e \in I} M_e = O(n^2)$ for every pairwise incompatible set I. In a pass, the pairs processed were in I when the pass started, and the pairs added during the pass stay in I until it ends, so enumerations and updates take $O(n^2)$ time in all. The test that skips e takes time quadratic in the number N_e of points in the two disks, and $N_e^2 \le 2nM_e$ as in the proof of Theorem 7, which gives $O(n^3)$ per pass. The tests of pairs of tight pairs take $O(1)$ time per pair of I. Every pass of the second stage except the last increases $|I|$, so there are at most $\mathrm{LB}_1 - \mathrm{LB}_0 + 31$ passes. ◀

In the experiments, stage 2 makes at most 13 passes, and the time grows like n^2 (Figure 4).

4 Sweep in Higher Dimensions

In this section, $P \subset \mathbb{R}^d$ for a fixed $d \geq 3$, and $\mathbb{B}(c, \rho)$ denotes a closed ball. The order $\prec$ is the lexicographic order on $\mathbb{R}^d$, and the half-ball $\mathbb{B}^+(c, \rho)$ consists of the points of $\mathbb{B}(c, \rho)$ whose first coordinate is at least that of c. Constants in the O-notation may depend on d. The local search does not depend on the dimension; Section 5.3 reports experiments in $\mathbb{R}^3$.

▶ **Theorem 11.** *Let $d \geq 3$ be fixed and $\varepsilon \leq 1/16$. There are pieces for which, for every $P \subset \mathbb{R}^d$, the sweep performs at most OPT_ε steps and outputs an ε-WSPD of size less than $32^d \mathrm{OPT}_\varepsilon$, and at most $2048\,\mathrm{OPT}_\varepsilon$ if $d = 3$, in $O(n^2 \log n)$ time and $O(n^2)$ space. With the same pieces, the greedy sweep satisfies the same bounds, except that it runs in $O(n^3)$ time.*

What carries over. Observations 2 and 4 hold in every dimension. The proof of Theorem 3 uses three properties of the pieces: they partition the ball and the half-ball, they have diameter at most $(1 - 2\varepsilon)$ times the radius, and there are 8 pieces in the ball and 5 in the half-ball. The invariant and the charging also use that a point $x \succeq p$ of $\mathbb{B}(p, \varepsilon D)$ lies in $\mathbb{B}^+(p, \varepsilon D)$, which holds for the lexicographic order in any dimension. Hence, if the ball and the half-ball of radius ρ are partitioned into k and h pieces, respectively, each of diameter at most $(1 - 2\varepsilon)\rho$, the proofs of validity and of the approximation factor apply verbatim and give an ε-WSPD of size at most $kh\,\mathrm{OPT}_\varepsilon$. Observation 5 does not use the geometry of the pieces, and it holds as stated: the number of steps is a lower bound on OPT_ε. The same holds for the greedy sweep, whose cap of 40 becomes kh.

Pieces. Let $\delta = 7/8$, so that $\delta \leq 1 - 2\varepsilon$ for $\varepsilon \leq 1/16$. It suffices to partition $\mathbb{B}(0, 1)$ and $\mathbb{B}^+(0, 1)$ into pieces of diameter at most δ, and to scale and translate them to $\mathbb{B}(c, \rho)$ and $\mathbb{B}^+(c, \rho)$. For fixed d, the pieces below are fixed sets, and a point is assigned to its piece in $O(1)$ time.

The *grid pieces*: let $m_d = \lceil 8\sqrt{d}/7 \rceil$, partition the bounding cube of $\mathbb{B}(0, 1)$ into the $(2m_d)^d$ cubes of side $1/m_d$ of a grid, and intersect them with the ball. The hyperplane through 0 orthogonal to the first axis is a grid hyperplane, so the half-ball meets only $m_d(2m_d)^{d-1}$ of these cubes. Each cube has diameter $\sqrt{d}/m_d \leq \delta$. Hence $k = (2m_d)^d$, $h = m_d(2m_d)^{d-1}$, and $kh = 2^{2d-1} m_d^{2d}$. For $d = 3$, we have $m_3 = 2$, with 64 pieces in the ball and 32 in the half-ball, and $kh = 2048$. For $d = 4$, we have $m_4 = 3$ and $kh = 839808 < 32^4$. A point is assigned to its cube by rounding its coordinates down, and a coordinate equal to 1 is assigned to the last cube.

The *net pieces*: let G be the set of points of the grid $\eta\mathbb{Z}^d$, with $\eta = \delta/(400\sqrt{d})$, at distance at most $1 + \delta/800$ from the origin, and let N be a maximal subset of G whose points are pairwise more than $399\delta/800$ apart, found greedily. Every point of $\mathbb{B}(0, 1)$ is within distance $\delta/800$ of G, and by maximality every point of G is within distance $399\delta/800$ of N, so every point of $\mathbb{B}(0, 1)$ is within distance $\delta/2$ of N. Hence assigning every point to a nearest point of N (ties broken by a fixed order of N) partitions the ball, and the half-ball, into at most $|N|$ pieces of diameter at most δ. The balls of radius $399\delta/1600$ centered at N are pairwise disjoint and contained in $\mathbb{B}(0, 1+401\delta/1600)$, so $|N| \leq (1600/(399\delta)+401/399)^d = (15607/2793)^d < 5.59^d$. Hence $kh < 5.59^{2d} < 31.25^d < 32^d$, which is smaller than the factor of the grid pieces for $d \geq 5$. The set N depends only on d, and a point is assigned to its piece in $O(|N|) = O(1)$ time.

▶ **Lemma 12.** *Let $\varepsilon \leq 1/16$ and $d \geq 2$. Every pair is covered by the output dumbbells of at most $2(11/3)^{2d}$ steps of the sweep in $\mathbb{R}^d$, and by at most one output dumbbell of each step.*

Proof. The proof of Lemma 6 applies verbatim, with balls in place of disks and volumes in $\mathbb{R}^{2d}$ in place of the products of areas. The products $\mathbb{B}(p_j, s/2) \times \mathbb{B}(q_j, s/2) \subset \mathbb{R}^{2d}$ are pairwise disjoint and contained in $\mathbb{B}(x, r + s/2) \times \mathbb{B}(y, r + s/2)$, so each of the two classes has at most $(1 + 2r/s)^{2d} \leq (11/3)^{2d}$ steps. ◀

Proof of Theorem 11. The size bounds follow from the discussion above, with the grid pieces for $d \leq 4$ and the net pieces for $d \geq 5$. The proof of the running time in Theorem 3 applies verbatim, with Lemma 12 in place of Lemma 6, since a point is assigned to its piece in $O(1)$ time. For the greedy sweep, whose cap of 40 becomes kh, the bounds on the number of steps and the size also follow from the discussion above, and the proof of its running time applies verbatim, with Lemma 8 in $\mathbb{R}^d$, and gives $O(n^3)$ time and $O(n^2)$ space. ◀

As in the plane, the number of steps is at most $\mathrm{OPT}_\varepsilon = O(n/\varepsilon^d)$ [7], so the sweep also runs in $O(n^2/\varepsilon^d)$ time.

Dependence on d. The factor of the grid pieces grows as $d^{\Theta(d)}$, and that of the net pieces as $2^{O(d)}$. No choice of pieces can make this analysis subexponential in d. By the isodiametric inequality, a piece of diameter at most $(1 - 2\varepsilon)\rho$ has at most a $((1 - 2\varepsilon)/2)^d$ fraction of the volume of the ball, so $kh \geq 2^{2d-1}/(1 - 2\varepsilon)^{2d} > 2^{2d-1}$.

5 Experiments

We address the following questions. (Q1) How close to optimal are the greedy sweep and the CK WSPD, and how does the gap depend on the input and n? (Q2) What are the running times? (Q3) Can local search close the remaining gap, and how does it compare with LNS? (Q4) How large are partition WSPDs converted from cover WSPDs, compared with those built directly? We use $\varepsilon = 1/20$ (Theorem 3 needs $\varepsilon \leq 1/16$). Sections 5.1 and 5.2 treat points in the plane and Section 5.3 points in $\mathbb{R}^3$.

5.1 Setup

All algorithms are implemented in C++. Points have integer coordinates and ε is rational, so all separation tests are exact, and every output is checked by a separate brute-force verifier. We compare three algorithms. *The sweep* is the greedy sweep of Section 3.1 followed by a pass that removes redundant dumbbells. *CK* is the variant of the CK construction on a fair split tree [7] that accepts two nodes if their point sets are ε-separated, and *CK box* tests the bounding boxes of the two nodes instead, so it outputs at least as many dumbbells. The inputs (Figure 3) are six synthetic families (uniform, Gaussian, clustered, perturbed grid, logarithmic spiral, and annulus) and the non-uniform instances of the CG:SHOP 2019 challenge [10], for n from 200 to 50000. Code, generators, and raw results are available at `https://github.com/gfonsecabr/small-wspd-experiments`. Table 2 gives the local search over time, Table 3 the mean size divided by the reference (Section 5.2) for every input and $n \in \{1000, 10000, 50000\}$, and Table 4 the same for the partition WSPDs of Q4 and $n \in \{1000, 10000\}$. The rest of this subsection gives the details.

Measurements. All experiments ran on an AMD Ryzen 9 9900X (12 cores, 128 GB of RAM, Fedora Linux 44, GCC 16.2 with `-O2`), with up to 11 single-threaded runs in parallel and no other load. Times are wall-clock. Memory is the peak resident set size of a run, which includes the output and the verifier. All methods, including CK, store every dumbbell as

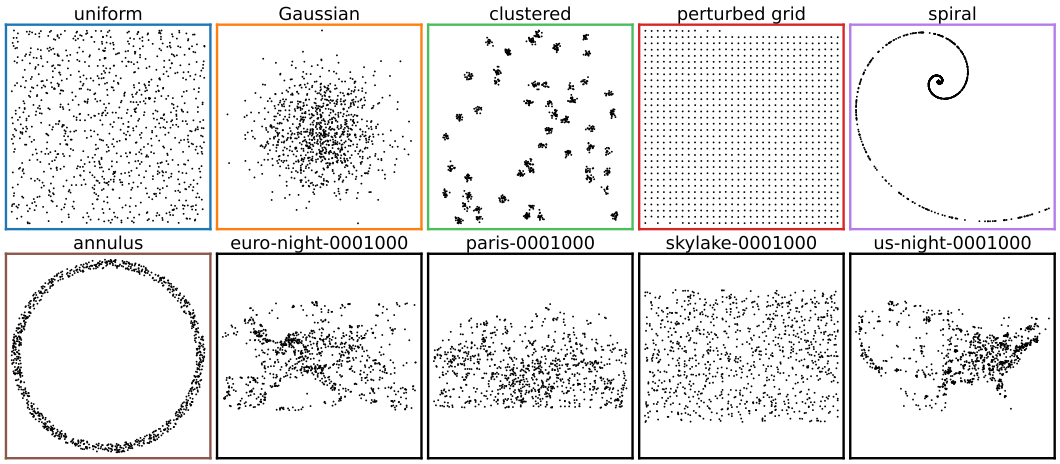


Figure 3 Synthetic and CG:SHOP instances for $n = 1000$.

two lists of points, so the times of CK include listing these points, which the time bounds of Section 7 do not. In Figure 8, the time to 1.001 OPT is the mean over the inputs with proven OPT, and where CPLEX reached its time limit of 1800 s on some input, the mean time of CPLEX is a lower bound.

Implementation. The implementation uses the Pangolin geometry library [9]. With $\varepsilon = a/b$, the separation test is $b^2 \operatorname{diam}(X)^2 \le a^2 \operatorname{dist}(X,Y)^2$, and the verifier checks coverage and exact separation.

Sweep. The final pass that removes redundant dumbbells considers them in the order of increasing $|X| \cdot |Y|$. For each p, the sweep processes the points $q \succ p$ by increasing distance from p. It stores one byte per pair, so memory is quadratic in n (Figure 8). When a step falls back to the fixed collection, it assigns the points to the pieces in floating point, which matters only for points on a boundary between pieces, and the output is verified like every other.

CK. The recursive construction on a fair split tree [7], which splits the longer side of the bounding box of a node at its midpoint. When two nodes are not accepted, the recursion splits the one whose bounding box has the larger diameter. CK box accepts two nodes if the diameters of their boxes are at most ε times the distance between the boxes, and CK tests the point sets exactly when this test fails. These are the two variants of Theorem 20. The original construction [7] tests balls that contain the boxes and splits the node whose box has the longer side (Section 7). Both variants are partition WSPDs, so the redundancy pass would not change them.

Exact optimum. An ε-separated dumbbell $\{X, Y\}$ with $\operatorname{dist}(X,Y) = \|pq\|$, $p \in X$, and $q \in Y$ corresponds to a clique containing p and q in an auxiliary graph on the points of $\mathbb{B}(p, \varepsilon\|pq\|) \cup \mathbb{B}(q, \varepsilon\|pq\|)$. The edges of this graph encode the pairwise conditions $\|xx'\| \le \varepsilon\,\|pq\|$, $\|yy'\| \le \varepsilon\,\|pq\|$, and $\|xy\| \ge \|pq\|$. We enumerate all maximal cliques (Bron–Kerbosch [4]) for every choice of p and q, and solve the resulting set-cover problem by integer programming with CPLEX 22.2. This is exact because some minimum WSPD consists of maximal dumbbells, and every maximal dumbbell arises from a maximal clique. Where CPLEX proves optimality for $n = 1000$ (10 of the 22 instances), the set-cover instance has a median of $6.6 \cdot 10^5$ maximal dumbbells. For the spirals with $n \ge 800$, we first apply exact set-cover reductions (forced columns, row and column dominance) until none applies, and

then run CPLEX on one connected component at a time, with three threads and the sweep followed by reshape as a warm start. For $n = 1000$, OPT is thus known on 13 instances.

Lower bound. The lower bounds come from separate runs of the sweep, which write the anchors to a file, followed by the search of Section 3.3, a separate program, with up to 9 runs in parallel for $n \leq 10000$ and 4 for larger n. The plateau moves use one fixed random seed. In Figure 4, panel (a) shows the mean over the instances with proven OPT (all of them for $n \leq 500$), panel (b) the mean over all instances, and panel (c) the mean running time of these runs, including the sweep. We computed the independence number α of G_ε for every instance with $n \leq 500$ by integer programming with CPLEX: one 0/1 variable per pair, whose sum is maximized, with at most one chosen pair in every maximal dumbbell, since two pairs are compatible if and only if some maximal dumbbell covers both. CPLEX proves optimality on all 88 instances, in at most 2 min each.

Local search. Every run uses one fixed random seed. For LNS, the *scale* of a dumbbell $\{X, Y\}$ is $\text{dist}(X, Y)$. The *neighborhood* of a dumbbell $T = \{A, B\}$ with scale ℓ_T consists of the dumbbells Z that contain a point within distance $2\varepsilon\ell_T$ of a point of T, whose scale is within a factor $(1 + 2\varepsilon)^2$ of ℓ_T, and whose sides, in one of the two orientations, have centroids within distance $2\varepsilon\ell_T$ of the centroids of A and B. Neighbors are ranked by the larger of these two centroid distances, and LNS removes T and its k nearest neighbors. The candidate dumbbells for a pair of U are the first 300 found by the enumeration used to compute OPT (above), and CPLEX has a limit of 2 s per call. We use $(k, u) = (16, 1000)$ for $n \leq 1000$ and $(32, 400)$ for larger n. In Figure 10, a curve begins at the mean time to build the start, over the instances of an input, and is horizontal while reshape is set up (relabeling the points and building its tables), a time that we measured in separate runs. It then shows the mean size for the same reshape time. Within its budget, LNS makes about $2 \cdot 10^5$ moves for $n = 10000$ and between $8 \cdot 10^4$ and $3 \cdot 10^5$ for $n = 50000$, on WSPDs of $3 \cdot 10^6$ to $4 \cdot 10^7$ dumbbells, so for large n it tries only a small fraction of the dumbbells. The parameters of LNS are the best ones among those we tested on the local-search samples (below) and on three instances with $n = 3000$: neighborhoods of radius $1.5\,\varepsilon\ell_T$ to $5\,\varepsilon\ell_T$, 4 to 64 neighbors, and bounds on $|U|$ from 50 to 4000 and no bound, not in all combinations. The limit of CPLEX (0.5 to 10 s) and the number of dumbbells enumerated per pair (50 to 2000) changed the sizes by at most 0.3%.

Inputs. *Synthetic inputs* (Figure 3) are generated at scale $M = 10^6$ around the center $(M/2, M/2)$ and rounded to integer coordinates, with three seeds each: *uniform* in $[0, M]^2$; *Gaussian* (standard deviation $M/8$); *clustered* ($\max\{2, n/20\}$ uniform centers, each point Gaussian around a random center with standard deviation $M/100$); *perturbed grid* (a grid with $\lceil\sqrt{n}\rceil$ columns, each point moved by a Gaussian of standard deviation 2% of the spacing); *logarithmic spiral* (the point $2000\,e^t(\cos 3t, \sin 3t)$ relative to the center, with t uniform in $[0, 6]$); and *annulus* (uniform angle, radius uniform in $[1, 1.1] \cdot M/4$). Duplicate points are discarded and replaced. The *CG:SHOP 2019* instances [10] come from the 2019 Computational Geometry Challenge. We use all of its non-uniform instances whose size is one of the values of n below, after removing duplicates. We use n from 200 to 1000 in steps of 100, from 2000 to 10000 in steps of 1000, and from 20000 to 50000 in steps of 10000 (one seed). The perturbed grid is filled row by row.

For every $n \leq 10000$, there are 18 synthetic and 4 CG:SHOP instances, and for every $n \geq 20000$, 6 and 4. The local-search samples are the first seed of every generator and the four CG:SHOP instances for $n = 1000$, and the first uniform and clustered instances and one CG:SHOP instance (euro-night) for $n = 10000$ and $n = 50000$. Figure 9 and Table 2 show

reshape and LNS on these samples, and Figure 9 also on all instances with $n = 200$. For $n = 50000$, both ran for 5000 s.

5.2 Results

Reference value. We compute OPT as described in Section 5.1. Within 1800 s per instance, CPLEX proves optimality on all instances with $n \leq 500$ and on some with $600 \leq n \leq 1000$, including the spirals with $n \geq 800$. Otherwise we compare against the lower bound LB of Section 3.3, so some values of n mix both references. Where OPT is known (Figure 4), LB is on average 0.999, 0.997, and 0.994 times OPT for $n = 200$, 500, and 1000, at least 0.989 OPT on every instance, and equal to OPT on 7 instances. The number of steps LB_0 is on average 0.98, 0.97, and 0.94 times OPT, and at least 0.90 OPT, and LB_1 is in between. For $n \leq 500$, we computed exactly the independence number α of G_ε, the best bound that independent sets can give: it is at least 0.992 OPT and on average 0.999 OPT. Of the gap between LB_0 and α, the swaps close 76% and the plateau moves another 22%, and $\mathrm{LB} = \alpha$ on 31 of these 88 instances. Ratios to LB overestimate ratios to OPT, probably more so for larger n: $\mathrm{LB}/\mathrm{LB}_0$ grows from 1.05 for $n = 1000$ to 1.10 and 1.12 for $n = 10000$ and 50000.

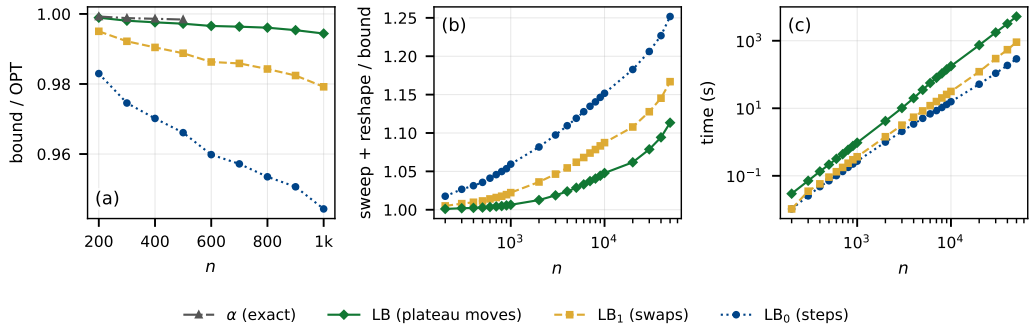


Figure 4 The lower bounds of Section 3.3: means over the instances.

Quality (Q1). Figure 5 summarizes the results, and Figure 6 and Table 3 show them per input. For $n = 500$, the sweep is on average 3% above OPT (at most 9%), and CK box and CK are 21% and 17% above (at most 37% and 27%). For $n = 10000$, the sweep is on average at 1.22 LB and CK at 1.55 LB, and for $n = 50000$ at 1.38 LB and 1.76 LB. Part of this growth may come from the lower bound, whose ratio to OPT decreases with n where OPT is known (Figure 4). The sweep is smallest, followed by CK and CK box, on every input and for every n, except on the perturbed grid, where the three methods are equal for $n = 200$ and CK is smaller than the sweep for $n = 1000$ (by less than 1%) and for $n = 4000$ (by 4%). After the merge pass of Q4, CK is still larger than the sweep on average for every n (1.06 OPT against 1.03 OPT for $n = 500$, and 1.46 LB against 1.22 LB for $n = 10000$). Figure 7 gives the number of dumbbells per point of the sweep followed by reshape. It grows with n on every input except the spiral, where it stays near 22, and for $n = 50000$ it ranges from 23 on the spiral to 981 on the perturbed grid.

Running times (Q2). Figure 8 reports the running times and the memory. On average, CK box and CK take 0.5 s for $n = 10000$ (at most 1.2 s) and 3.7 s and 4.1 s for $n = 50000$, and the sweep takes 0.3 s for $n = 1000$, 15 s for $n = 10000$, and 5 min for $n = 50000$. For $n = 50000$,

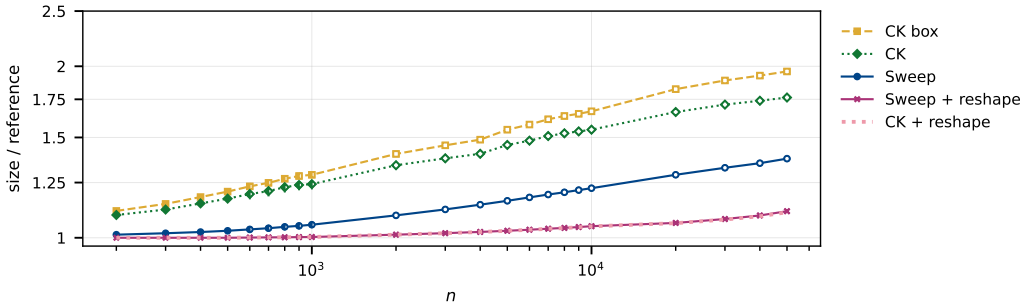


Figure 5 Means. Open (half-filled) markers: the reference is LB on all (some) instances.

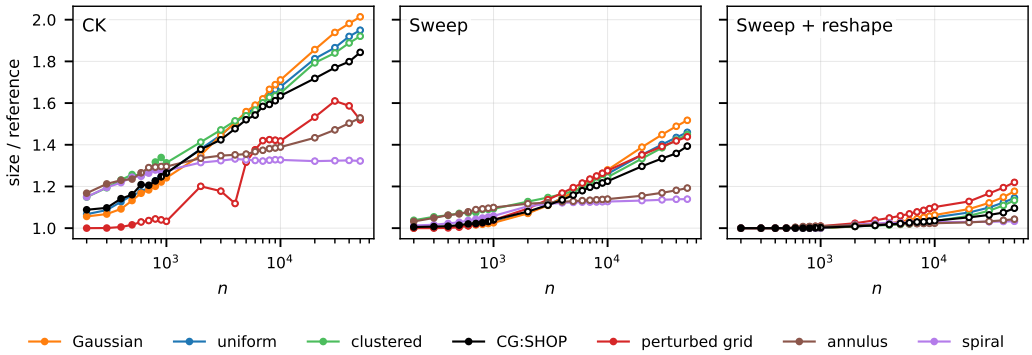


Figure 6 Mean size divided by the reference, per input. Markers as in Figure 5.

CK, the sweep, and reshape use up to 8, 15, and 22 GB. Computing OPT, including the enumeration of the maximal dumbbells, takes 0.07 s for $n = 200$ and 15 s for $n = 1000$ (median where CPLEX proves optimality). After the sweep, computing LB_1 takes on average 16 s for $n = 10000$ and 10 min for $n = 50000$, and computing LB takes 2.7 min and 82 min (Figure 4).

Local search (Q3). We ran reshape (Section 3.2) from CK and from the sweep on every instance, for 100 s for $n \leq 1000$, 1000 s for $2000 \leq n \leq 10000$, and 5000 s above. LNS ran with the same budgets, on every instance for $n \leq 1000$ and on samples of 3 instances for $n = 10000$ and $n = 50000$ (Figure 9 and Table 2). Reshape ends at most as large as LNS from the same start on every instance, and smaller on average for every n. It reaches the final size of LNS within 1% of the budget, while LNS improves CK from 1.66 to 1.39 times the reference for $n = 10000$ and from 1.89 to 1.82 for $n = 50000$, where it tries only a small fraction of the dumbbells within its budget (Section 5.1). On the 164 instances with $n \leq 1000$ where OPT is known, the sweep followed by reshape is on average 0.002% and at most 0.02% above OPT, and it reaches OPT on 101 of them (all 22 for $n = 200$, 13 of 22 for $n = 500$, and 1 of 13 for $n = 1000$), against 0.23%, 1.5%, and 18 for LNS from the sweep. It is on average at 1.05 LB for $n = 10000$ (at most 1.10 LB) and 1.11 LB for $n = 50000$ (at most 1.22 LB), against 1.22 LB and 1.38 LB for the sweep alone (right column of Figure 6). For the same total time, including the time to build the start and to set up reshape, the best results are obtained by CK followed by reshape (Figure 10), although CK is on average larger than the output of the sweep. On every instance with $n \geq 20000$ and on 378 of the 396

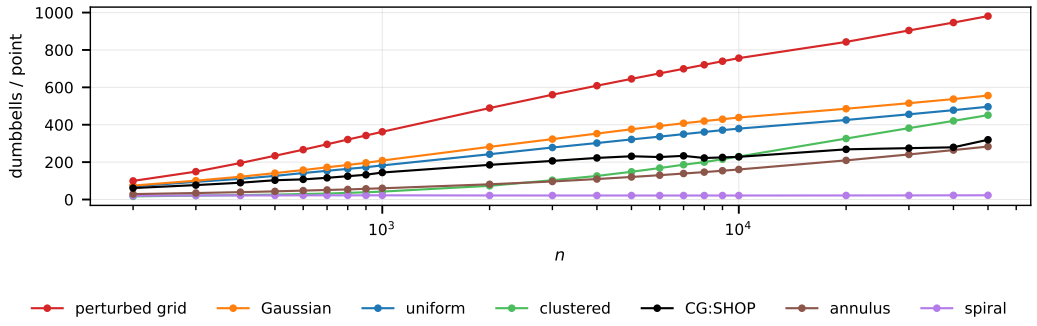


Figure 7 Sweep + reshape: mean number of dumbbells per point, per input.

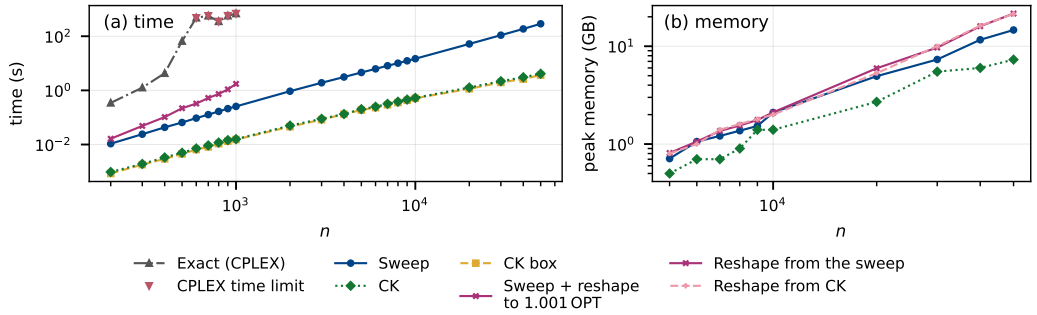


Figure 8 Mean running time and largest peak memory.

instances with $n \leq 10000$, it is smaller than the output of the sweep before the sweep has finished; this takes on average 89% of the running time of the sweep for $n = 10000$ and 76% for $n = 50000$. At every later time and on every instance, the sweep followed by reshape is at most 0.73% smaller, and never smaller for $n \geq 20000$. However, the approximation factor that we prove for CK followed by reshape is larger: since reshape never increases the size, it is 576 for $\varepsilon \leq 1/16$ (Section 7), against 40 for the sweep followed by reshape (Theorem 7). For the same reshape time, the two starts end within 0.42% of each other for $n \leq 10000$. For $n \geq 20000$, where reshape has not converged after 5000 s, the run from CK ends smaller on 39 of the 40 instances, by 0.4% on average for $n = 50000$ and by at most 1.2%. Since the two starts end close to each other, the sweep matters mainly for its lower bound and its

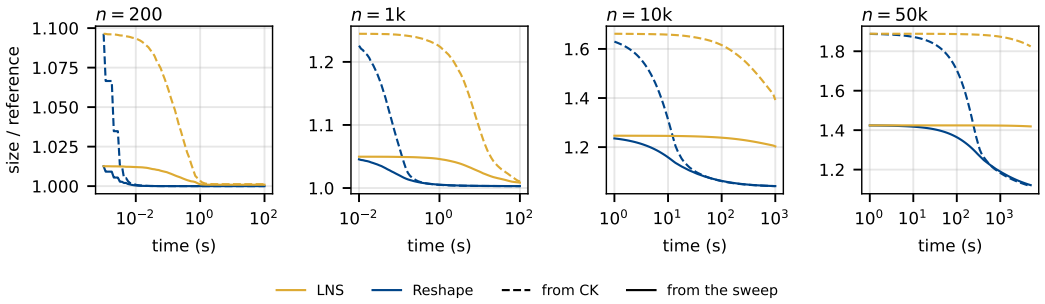


Figure 9 Local search: mean size / reference over time, on all instances for $n = 200$.

approximation factor when reshape follows.

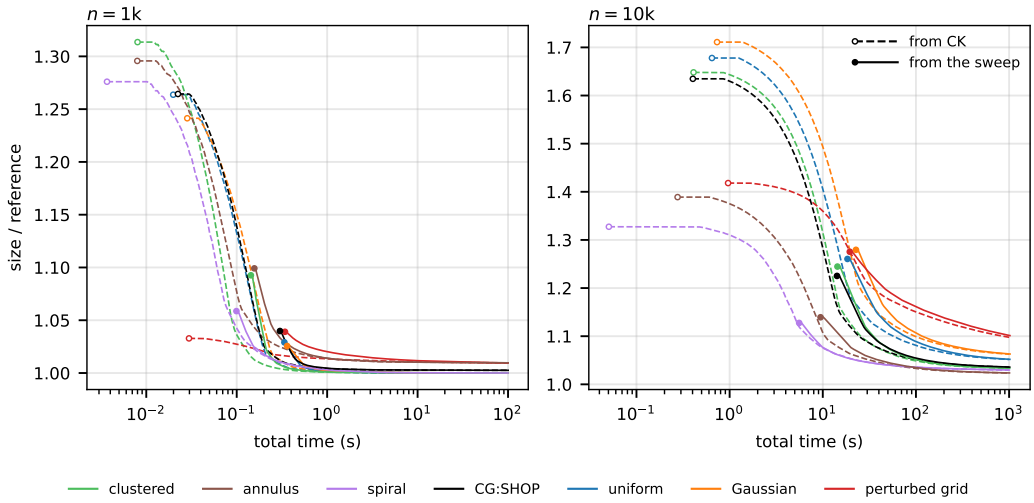


Figure 10 Mean size / reference per input over the total time: build of the start, then reshape.

Partition WSPDs (Q4). We convert a cover WSPD into a partition WSPD by processing its dumbbells one by one. For the current dumbbell $\{X, Y\}$, let U be the set of its pairs that no earlier dumbbell covers. Two points of X are grouped if they have the same partners in Y among the pairs of U, and each group $X' \subseteq X$ with partners $Y' \subseteq Y$ gives the dumbbell $\{X', Y'\}$. The same is done with the roles of X and Y exchanged, and the split into fewer dumbbells is kept. These dumbbells cover the pairs of U exactly once, and they are ε-separated by Observation 4. The order comes from a priority queue that holds, for every dumbbell, the number of dumbbells into which its uncovered pairs split: the next dumbbell has the smallest number, ties broken by size, largest first. The numbers are updated lazily.

A final *merge pass* replaces two dumbbells $\{S, O_1\}$ and $\{S, O_2\}$ with a common side by $\{S, O_1 \cup O_2\}$ if it is ε-separated. The result is still a partition WSPD. The pass tries at most 256 dumbbells per side and is repeated until no merge occurs, at most 20 times. It applies to every partition WSPD, also to CK and the quadtree.

The conversion has no size guarantee (Figure 11 and Table 4). For the sweep followed by reshape, the partition WSPD is on average 1.015, 1.13, and 1.18 times the reference for $n = 1000$, 10000, and 20000 (at most 1.40, on the perturbed grid with $n = 20000$), against 1.003, 1.05, and 1.06 for the cover WSPD, and the conversion takes at most 54 s for $n = 10000$ and 142 s for $n = 20000$. CK, which is a partition WSPD, is at 1.24, 1.55, and 1.66, and at 1.11, 1.46, and 1.59 after the merge pass. It is never smaller than the converted WSPD, also after the merge pass. Converting the optimal WSPDs with $n = 500$ gives partition WSPDs of size $1.004\,\mathrm{OPT}_\varepsilon$ on average and at most $1.0125\,\mathrm{OPT}_\varepsilon$, which bounds the partition optimum on these instances. The quadtree construction of Section 7, with exact tests, is at 1.71, 2.45, and 2.78 times the reference, and at 1.15, 1.98, and 2.36 after the merge pass. Despite its smaller proven factor, it is larger than CK on average for every n, with or without merging.

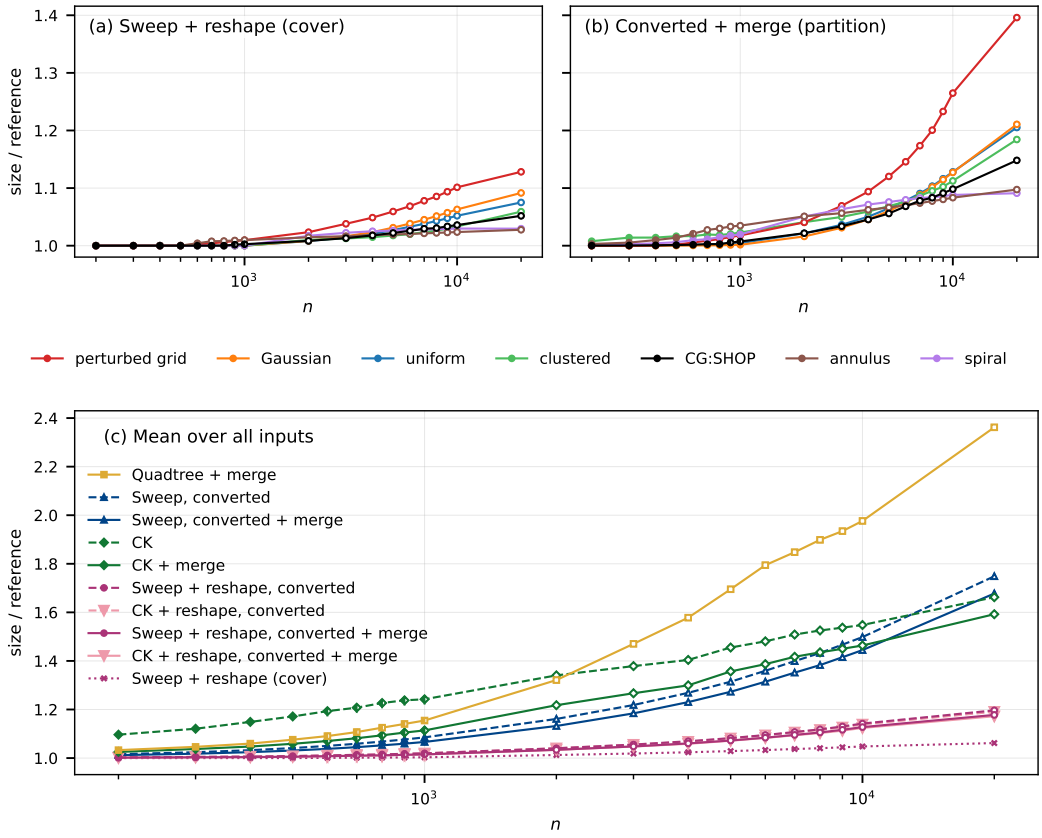


Figure 11 Partition WSPDs: mean size / reference, (a, b) per input. Markers as in Figure 5.

Table 2 Local search over time: mean size divided by the reference. *Build time*: median time to build the start, not included in the times. Bold: best value for each n and time over both starts.

n	1k			10k			50k		
Time	1 s	10 s	100 s	10 s	100 s	1000 s	50 s	500 s	5000 s
Start: CK									
Build time		0.017 s			0.41 s			5.7 s	
Start	1.245	1.245	1.245	1.662	1.662	1.662	1.889	1.889	1.889
Reshape	**1.005**	**1.003**	**1.003**	1.312	**1.061**	**1.041**	1.801	1.240	**1.116**
LNS	1.225	1.093	1.011	1.658	1.616	1.392	1.888	1.881	1.817
Start: Sweep									
Build time		0.31 s			15 s			354 s	
Start	1.050	1.050	1.050	1.247	1.247	1.247	1.424	1.424	1.424
Reshape	**1.005**	**1.003**	**1.003**	**1.158**	1.063	**1.041**	**1.392**	**1.235**	1.121
LNS	1.046	1.026	1.009	1.246	1.238	1.203	1.424	1.424	1.418

Table 3 Mean size divided by the reference, per input: synthetic inputs (three seeds for $n \leq 10000$, one for $n = 50000$) and CG:SHOP 2019 families. †: the reference is the lower bound. Bold: smallest construction of the row.

Input	n	CK box	CK	Sweep	CK + reshape	Sweep + reshape
uniform	1k	1.30	1.26	**1.03**	1.00	1.00
	10k†	1.84	1.68	**1.26**	1.05	1.05
	50k†	2.21	1.95	**1.46**	1.14	1.15
Gaussian	1k	1.27	1.24	**1.03**	1.00	1.00
	10k†	1.88	1.71	**1.28**	1.06	1.06
	50k†	2.30	2.01	**1.52**	1.17	1.18
clustered	1k	1.44	1.31	**1.09**	1.00	1.00
	10k†	1.84	1.65	**1.24**	1.03	1.03
	50k†	2.18	1.92	**1.45**	1.13	1.13
perturbed grid	1k†	**1.03**	**1.03**	1.04	1.01	1.01
	10k†	1.45	1.42	**1.28**	1.10	1.10
	50k†	1.56	1.52	**1.44**	1.21	1.22
spiral	1k	1.28	1.28	**1.06**	1.00	1.00
	10k†	1.33	1.33	**1.13**	1.03	1.03
	50k†	1.32	1.32	**1.14**	1.03	1.03
annulus	1k†	1.38	1.30	**1.10**	1.01	1.01
	10k†	1.49	1.39	**1.14**	1.02	1.02
	50k†	1.66	1.53	**1.19**	1.04	1.04
euro-night	1k	1.30	1.25	**1.04**	1.00	1.00
	10k†	1.80	1.64	**1.23**	1.04	1.04
	50k†	2.02	1.80	**1.36**	1.08	1.09
us-night	1k†	1.34	1.29	**1.04**	1.00	1.00
	10k†	1.77	1.61	**1.21**	1.03	1.03
	50k†	1.97	1.76	**1.34**	1.07	1.08
paris	1k†	1.32	1.27	**1.04**	1.00	1.00
skylake	1k†	1.28	1.24	**1.04**	1.00	1.00
jupiter	10k†	1.81	1.65	**1.22**	1.03	1.03
world	10k†	1.81	1.64	**1.24**	1.04	1.04
protein	50k†	2.15	1.88	**1.43**	1.11	1.11
sun	50k†	2.21	1.94	**1.44**	1.10	1.11

Table 4 Partition WSPDs: mean size divided by the reference (†: the lower bound) per input and n. First column: the cover WSPD of the sweep followed by reshape. The others are partition WSPDs: its conversion and CK, each without and with the merge pass, and the deterministic quadtree with exact separation tests and the merge pass. Bold: smallest partition WSPD of the row.

		Sweep		Converted		CK	Quadtree
Input	n	+ reshape	Converted	+ merge	CK	+ merge	+ merge
uniform	1k	1.00	1.01	**1.00**	1.26	1.12	1.17
	10k†	1.05	1.14	**1.13**	1.68	1.64	2.54
Gaussian	1k	1.00	**1.00**	**1.00**	1.24	1.12	1.19
	10k†	1.06	1.14	**1.13**	1.71	1.68	2.61
clustered	1k	1.00	1.04	**1.02**	1.31	1.14	1.19
	10k†	1.03	1.13	**1.11**	1.65	1.55	2.19
perturbed grid	1k†	1.01	**1.02**	**1.02**	1.03	**1.02**	1.05
	10k†	1.10	1.28	**1.26**	1.42	1.41	1.80
spiral	1k	1.00	1.03	**1.02**	1.28	1.16	1.17
	10k†	1.03	1.10	**1.09**	1.33	1.22	1.23
annulus	1k†	1.01	1.05	**1.03**	1.30	1.13	1.16
	10k†	1.02	1.10	**1.08**	1.39	1.20	1.26
euro-night	1k	1.00	1.01	**1.00**	1.25	1.10	1.15
	10k†	1.04	1.11	**1.10**	1.64	1.54	2.11
us-night	1k†	1.00	**1.01**	**1.01**	1.29	1.11	1.15
	10k†	1.03	1.10	**1.09**	1.61	1.51	2.02
paris	1k†	1.00	**1.01**	**1.01**	1.27	1.12	1.16
skylake	1k†	1.00	**1.01**	**1.01**	1.24	1.11	1.16
jupiter	10k†	1.03	1.11	**1.09**	1.65	1.56	2.27
world	10k†	1.04	1.13	**1.11**	1.64	1.54	2.20

5.3 Experiments in Three Dimensions

We ran the experiments of Sections 5.1 and 5.2 on points in $\mathbb{R}^3$, with $\varepsilon = 1/20$: the sweep, CK box, CK, reshape and LNS, the conversion into partition WSPDs, and the computations of OPT, LB, and α. Table 5 gives the results per input for $n \in \{1000, 10000\}$.

Implementation. The experiments in $\mathbb{R}^3$ use a separate C++ implementation, with its own kd-tree. As in the plane, points have integer coordinates, all separation tests are exact, and every output is checked by a brute-force verifier for coverage and exact separation, and, for partition WSPDs, for covering every pair exactly once. The greedy sweep uses the grid pieces of Section 4: 64 cubes of side $\varepsilon D/2$ in the ball and the 32 of them that meet the half-ball. A step falls back to the fixed collection only if it has fewer nonempty products than the dumbbells grown in the step, which happened in 117 steps over all instances, at most a fraction 10^{-5} of the steps of any instance. CK splits the longest side of the bounding box of a node at its midpoint, and reshape and LNS relabel the points in Morton order. On planar instances embedded in the plane $z = 0$, CK box and CK give exactly the sizes of the planar implementation, and the sweep is within 0.15% of it: ties are broken by the order in which the range queries report the points. The partition WSPDs are the conversion of Q4 with the merge pass, and the octree version of the quadtree construction of Section 7, with exact tests. OPT, LB (with 30 plateau passes), α, and LNS are computed as in the plane (Sections 3.3 and 5.1). Reshape runs from CK and from the sweep on every instance, for 100 s for $n \leq 1000$ and 1000 s for larger n, and LNS from both starts, with the same budgets, on all instances with $n = 200$, on all instances with $n = 1000$, and on three instances with $n = 10000$ (uniform, clustered, and the bunny). All runs are single-threaded, with up to 11 runs in parallel, on the machine of Section 5.1; the times of the sweep, of CK box, and of CK come from separate runs of each.

Inputs. The *synthetic inputs* (Figure 12) are generated in the cube $[0, M]^3$, with $M = 10^6$, and rounded to integer coordinates: *uniform* in the cube; *Gaussian* (around the center of the cube, standard deviation $M/8$ in each coordinate); *clustered* ($\max\{2, n/20\}$ uniform centers, each point Gaussian around a random center with standard deviation $M/100$); *perturbed grid* (a grid with $\lceil n^{1/3} \rceil$ points per side, filled layer by layer, each point moved by a Gaussian of standard deviation 2% of the spacing); *conical spiral* (the point $2000\, e^t(\cos 3t, \sin 3t, 1)$ relative to the center of the cube, with t uniform in $[0, 6]$); and *spherical shell* (around the center of the cube, uniform direction, radius uniform in $[1, 1.1] \cdot M/4$). Duplicate points are discarded and replaced. The *scans* are four meshes of the Stanford 3D Scanning Repository [20]: the bunny, the armadillo, the XYZ RGB dragon, and the happy Buddha, with 35947, 49990, 125066, and 49251 vertices. Each is scaled so that the longest side of its bounding box is 10^6 and rounded, and we use a random sample of n vertices. We use the values of n of Section 5.1 up to 10000: from 200 to 1000 in steps of 100 and from 2000 to 10000 in steps of 1000. For each n, there is one instance per generator and one sample of each scan, so 10 instances, and 180 in all.

Reference value. CPLEX proves optimality on all 90 instances with $n \leq 1000$. Computing OPT takes a median of 3.2 s for $n = 1000$, but up to 55 min on the conical spirals, where the enumeration of the maximal dumbbells dominates. For larger n, the reference is LB. Where OPT is known (Figure 13), LB is on average 0.999, 0.998, and 0.998 times OPT for $n = 200$, 500, and 1000, at least 0.986 OPT on every instance, and equal to OPT on 20 instances. The number of steps LB_0 is on average 0.99, 0.98, and 0.97 times OPT, and at least 0.86 OPT.

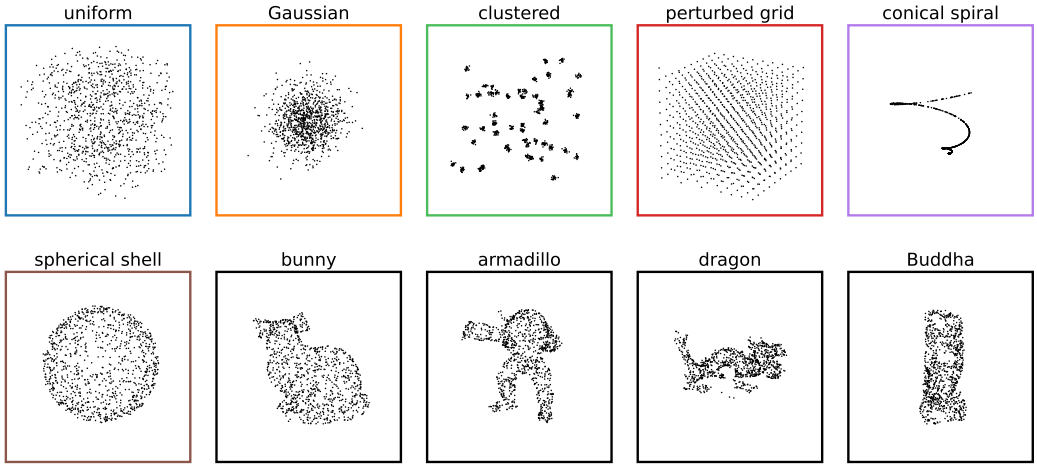


Figure 12 Synthetic instances and scans in $\mathbb{R}^3$ for $n = 1000$.

For $n \leq 500$, α is at least 0.987 OPT and on average at least 0.998 OPT; CPLEX computes it in at most 88 s. Of the gap between LB_0 and α, the swaps close 85% and the plateau moves another 14%, and $\mathrm{LB} = \alpha$ on 31 of these 40 instances. The ratio $\mathrm{LB}/\mathrm{LB}_0$ grows from 1.03 for $n = 1000$ to 1.09 for $n = 10000$.

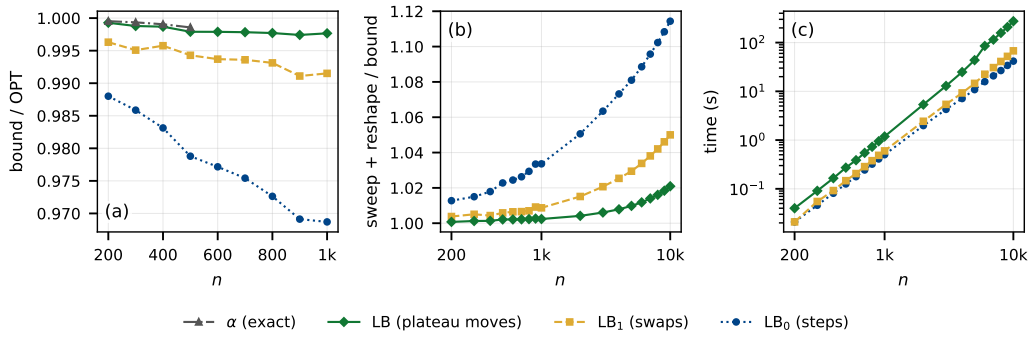


Figure 13 The lower bounds in $\mathbb{R}^3$: means over the instances.

Quality. Figure 14 summarizes the results, and Figure 15 shows them per input. The sweep is never larger than CK and CK box. It is smaller than both on every instance except the perturbed grids with $n \leq 1000$, where every dumbbell of every WSPD is a single pair, so that all methods output $\binom{n}{2}$ dumbbells. For $n = 500$, the sweep is on average 1.4% above OPT (at most 9%), and CK box and CK are 15% and 12% above (at most 63% and 36%). For $n = 1000$, the sweep is at 1.02 OPT and CK at 1.18 OPT. For $n = 10000$, the sweep is on average at 1.13 LB (at most 1.23 LB), and CK box and CK at 1.61 LB and 1.49 LB (at most 2.03 LB and 1.68 LB). Figure 16 gives the number of dumbbells per point of the sweep followed by reshape and the number of pairs per dumbbell of the reference, $\binom{n}{2}$ divided by the reference. For $n \leq 1000$, the uniform, Gaussian, and perturbed grid inputs have at most 1.16 pairs per dumbbell, so almost all dumbbells are single pairs and every method is close to OPT. For $n = 10000$, the number of pairs per dumbbell ranges from 1.16 on the perturbed

grid to 244 on the conical spiral, and it is between 7.6 and 9.0 on the scans; the number of dumbbells per point ranges from 21 on the conical spiral to 4348 on the perturbed grid.

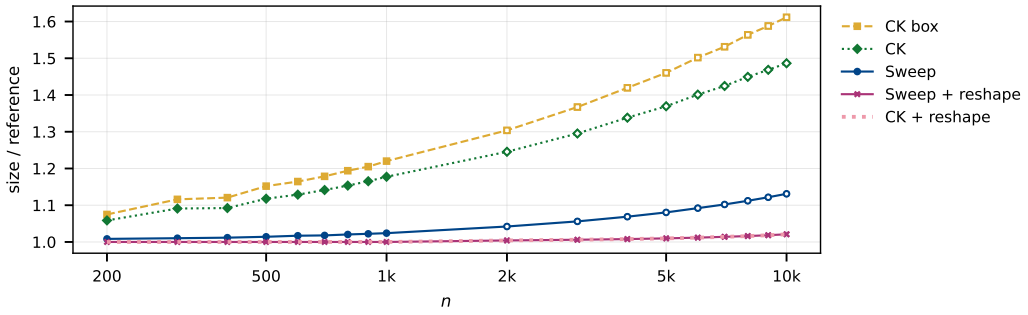


Figure 14 Means in $\mathbb{R}^3$. Open markers: the reference is LB on all instances.

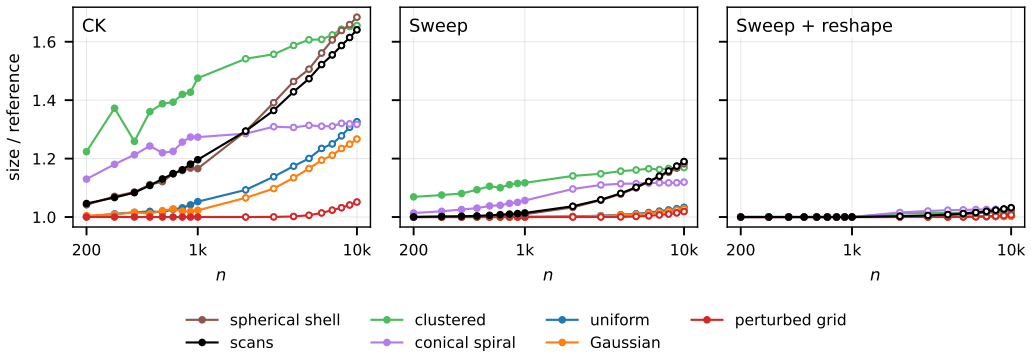


Figure 15 Size divided by the reference in $\mathbb{R}^3$, per input. Markers as in Figure 14.

Running times. Figure 17 reports the running times and the memory. On average, the sweep takes 0.5 s for $n = 1000$ and 45 s for $n = 10000$, and CK box and CK take 1.4 s and 1.5 s for $n = 10000$. The sweep followed by reshape reaches 1.001 OPT in 1.0 s on average for $n = 1000$. After the sweep, computing LB_1 takes 26 s and computing LB 3.9 min on average for $n = 10000$. For $n = 10000$, the sweep and reshape use up to 8.4 GB and CK up to 5.0 GB, on the perturbed grid, whose WSPDs have the most dumbbells.

Local search. Reshape, from either start, is at most 0.03% above OPT on the 90 instances where OPT is known, and reaches OPT on 72 of them from CK and on 69 from the sweep. For $n = 10000$, it is on average at 1.021 LB and at most 1.043 LB, against 1.13 LB for the sweep alone. The two starts end within 0.03% of each other on every instance. Reshape ends at most as large as LNS from the same start in every run of Figure 18. For $n = 1000$, LNS ends on average 0.36% above the reference from CK and 0.19% from the sweep, against less than 0.01% for reshape, and for $n = 10000$, LNS improves CK from 1.55 to 1.34 times the reference and the sweep from 1.15 to 1.10 (means over the three instances), against 1.02 for reshape from both starts. Figure 19 shows reshape from both starts against the total time, including the time to build the start.

Table 5 Size divided by the reference in $\mathbb{R}^3$, per input (one instance each). The column "Pairs" gives $\binom{n}{2}$ divided by the reference. †: the reference is the lower bound. Bold: smallest construction of the row.

Input	n	Pairs	CK box	CK	Sweep	CK + reshape	Sweep + reshape
uniform	1k	1.2	1.06	1.05	**1.00**	1.00	1.00
	10k†	2.2	1.37	1.33	**1.03**	1.00	1.00
Gaussian	1k	1.1	1.03	1.02	**1.00**	1.00	1.00
	10k†	1.9	1.30	1.27	**1.03**	1.00	1.00
clustered	1k	10.9	1.76	1.48	**1.12**	1.00	1.00
	10k†	14.7	2.03	1.66	**1.17**	1.02	1.01
perturbed grid	1k	1.0	**1.00**	**1.00**	**1.00**	1.00	1.00
	10k†	1.2	1.05	1.05	**1.02**	1.01	1.01
conical spiral	1k	24.4	1.28	1.27	**1.06**	1.00	1.00
	10k†	243.8	1.32	1.32	**1.12**	1.03	1.03
spherical shell	1k	1.8	1.19	1.17	**1.01**	1.00	1.00
	10k†	6.0	1.87	1.68	**1.18**	1.03	1.03
bunny	1k	1.9	1.21	1.19	**1.01**	1.00	1.00
	10k†	8.3	1.81	1.67	**1.23**	1.04	1.04
armadillo	1k	2.2	1.23	1.21	**1.02**	1.00	1.00
	10k†	9.0	1.83	1.67	**1.20**	1.04	1.04
dragon	1k	2.1	1.24	1.21	**1.01**	1.00	1.00
	10k†	8.5	1.78	1.62	**1.17**	1.03	1.03
Buddha	1k	2.0	1.21	1.18	**1.01**	1.00	1.00
	10k†	7.6	1.76	1.61	**1.16**	1.02	1.02

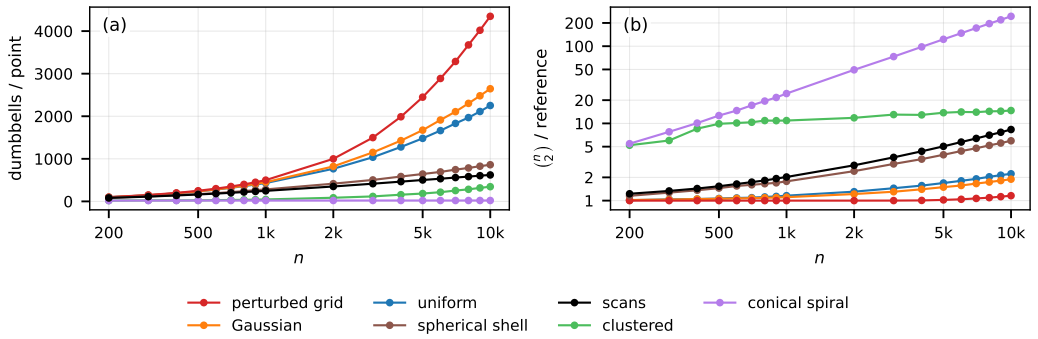


Figure 16 Sweep + reshape in $\mathbb{R}^3$: (a) dumbbells per point, (b) pairs per dumbbell of the reference, per input.

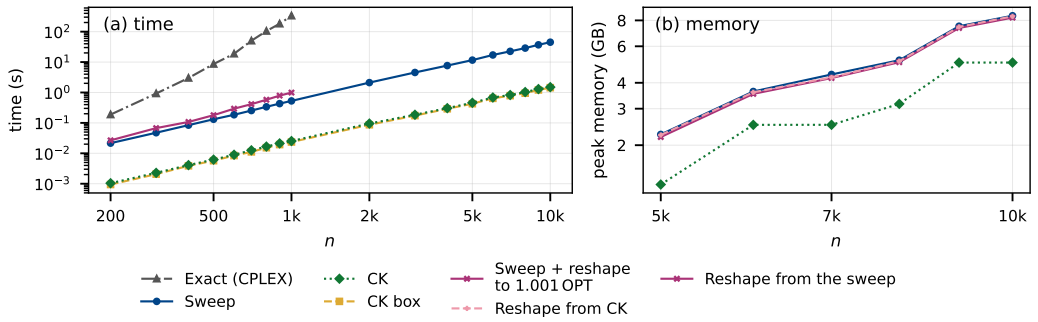


Figure 17 Mean running time and largest peak memory in $\mathbb{R}^3$.

Partition WSPDs. Figure 20 shows the partition WSPDs. For the sweep followed by reshape, the converted partition WSPD is on average 1.005 and 1.055 times the reference for $n = 1000$ and 10000, against 1.000 and 1.021 for the cover WSPD. CK is at 1.18 and 1.49, and at 1.09 and 1.46 after the merge pass. It is never smaller than the converted WSPD, also after the merge pass. The octree construction is at 1.51 and 2.44, and at 1.14 and 2.25 after the merge pass; with the merge pass, it is smaller than CK with the merge pass on 5 instances, all with $n \leq 300$.

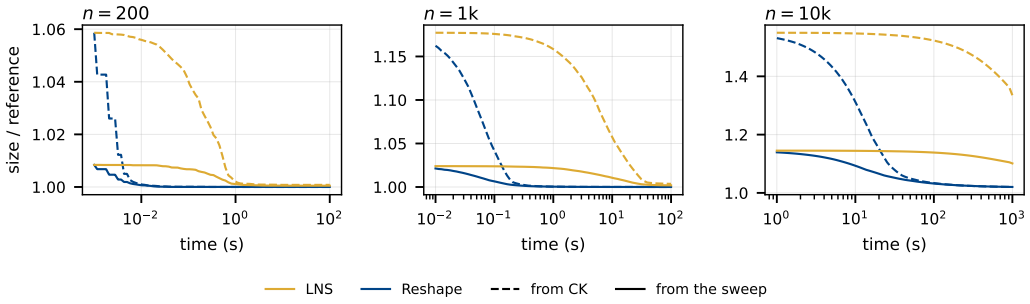


Figure 18 Local search in $\mathbb{R}^3$: mean size / reference over time, on all instances for $n \in \{200, 1000\}$.

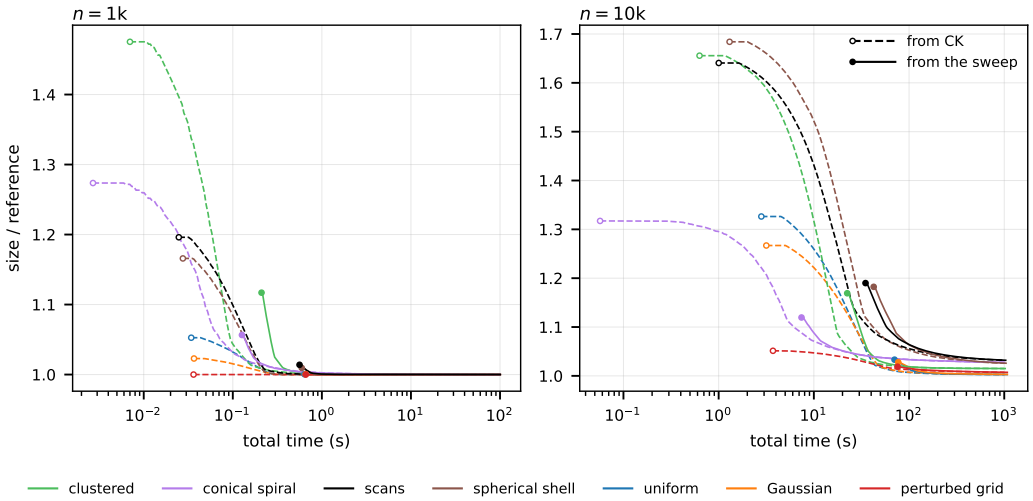


Figure 19 Mean size / reference in $\mathbb{R}^3$ per input over the total time: build of the start, then reshape.

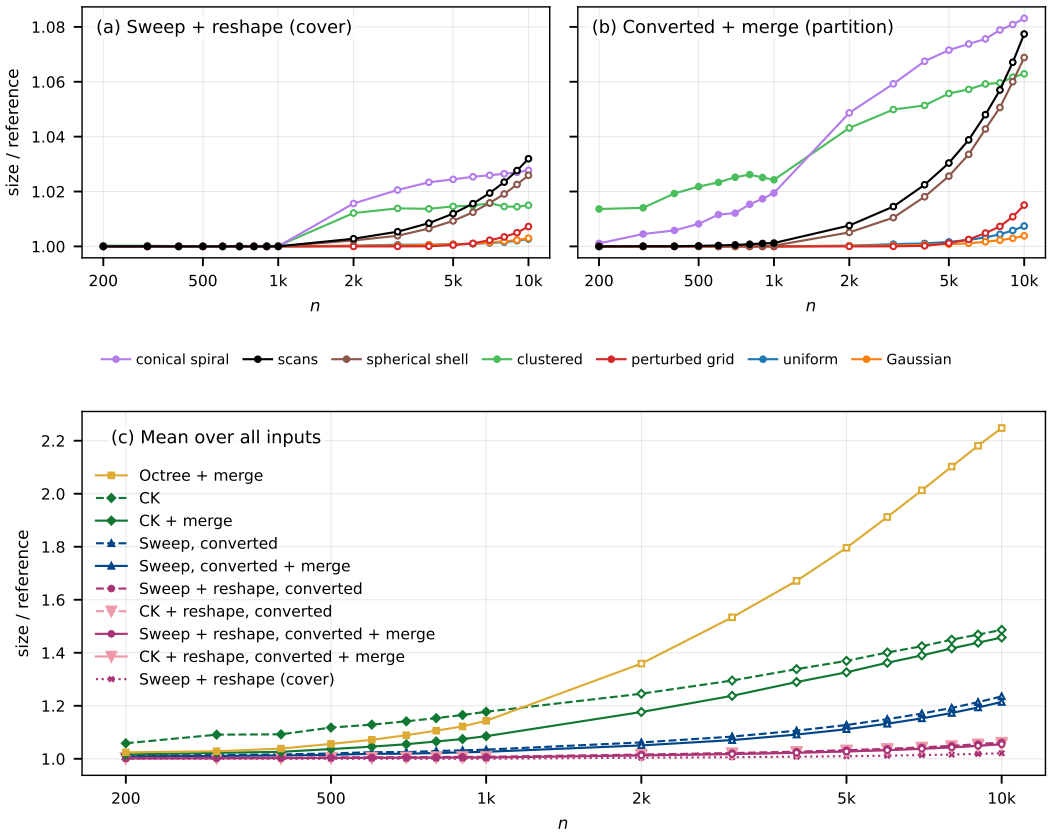


Figure 20 Partition WSPDs in $\mathbb{R}^3$: mean size / reference, (a, b) per input. Markers as in Figure 14.

6 Larger ε

For $\varepsilon > 1/16$, the pieces of Section 2 may give dumbbells that are not ε-separated: the proof of Theorem 3 bounds the distance between two pieces only by $(1-2\varepsilon)D$, which is useless for $\varepsilon \geq 1/2$. We show that pieces that depend on ε, together with a separation test, give both sweeps a constant factor for every $\varepsilon > 0$. The new ingredient is a lower bound on the distance between two pieces that uses a dumbbell covering the anchor. Throughout, $P \subset \mathbb{R}^2$.

Piece families. Let $H = \{z \in \mathbb{B}(0,1) : z_1 \geq 0\}$ be the unit half-disk, where z_1 denotes the first coordinate of z. A *piece family* $\mathcal{L} = (\mathcal{L}^+, \mathcal{L}^\circ)$ consists of a partition $\mathcal{L}^+$ of H and a partition $\mathcal{L}^\circ$ of $\mathbb{B}(0,1)$ into finitely many *pieces*. Each piece Π comes with numbers $0 \leq a_\Pi \leq b_\Pi \leq 1$ and $\beta_\Pi > 0$ such that $a_\Pi \leq \|z\| \leq b_\Pi$ for every $z \in \Pi$ and $\operatorname{diam}(\Pi) \leq \beta_\Pi$. The *size* of $\mathcal{L}$ is $|\mathcal{L}^+| \cdot |\mathcal{L}^\circ|$. For $0 \leq a \leq b \leq 1$ and $0 \leq c \leq d \leq 1$, let

$$M_\varepsilon(a,b;c,d) = \min_{r\in[a,b],\ s\in[c,d]} \max\Big\{1-\varepsilon(r+s),\ r,\ s,\ \frac{1}{1+2\varepsilon}\Big\}.$$

The family is *ε-admissible* if all $\Pi \in \mathcal{L}^+$ and $\Sigma \in \mathcal{L}^\circ$ satisfy

$$\max\{\beta_\Pi, \beta_\Sigma\} \leq \max\big\{1-\varepsilon(b_\Pi + b_\Sigma),\ M_\varepsilon(a_\Pi, b_\Pi; a_\Sigma, b_\Sigma) - \varepsilon(\beta_\Pi + \beta_\Sigma)\big\}. \tag{1}$$

Every term on the right-hand side is nonincreasing in ε, so a family that is ε_0-admissible is ε-admissible for every $\varepsilon \leq \varepsilon_0$. The first term is the bound used in the proof of Theorem 3: the pieces of Section 2, scaled to radius 1, with $a_\Pi = 0$, $b_\Pi = 1$, and the diameters of Lemma 1, form a $1/16$-admissible family of size 40.

The collection of a family. Fix a piece family $\mathcal{L}$. For a step with anchor $\{p,q\}$, where $p \prec q$ and $D = \|pq\|$, let $A_\Pi = \{z \in P : z \succeq p,\ z \in p + \varepsilon D\,\Pi\}$ for $\Pi \in \mathcal{L}^+$, and let $C_\Sigma = P \cap (q + \varepsilon D\,\Sigma)$ for $\Sigma \in \mathcal{L}^\circ$. The *$\mathcal{L}$-collection* of the step consists of the dumbbells $\{A_\Pi, C_\Sigma\}$ whose sides are nonempty and disjoint and that are ε-separated. Both conditions are tested on the points, so the $\mathcal{L}$-collection consists of at most $|\mathcal{L}^+| \cdot |\mathcal{L}^\circ|$ dumbbells, all ε-separated. For the pieces of Section 2 and $\varepsilon \leq 1/16$, every dumbbell $\{A_i, C_k\}$ with nonempty sides passes both tests, so the $\mathcal{L}$-collection is the fixed collection.

▶ **Lemma 13** (Piece separation). *Let $p, q \in \mathbb{R}^2$, $D = \|pq\|$, and let $\{X, Y\}$ be an ε-separated dumbbell with $p, x \in X$ and $q, y \in Y$. Let $A \ni x$ and $C \ni y$ be sets such that $a\varepsilon D \leq \|pz\| \leq b\varepsilon D$ for every $z \in A$, $c\varepsilon D \leq \|qz\| \leq d\varepsilon D$ for every $z \in C$, $\operatorname{diam}(A) \leq \beta\varepsilon D$, and $\operatorname{diam}(C) \leq \beta'\varepsilon D$. Then*

$$\operatorname{dist}(A, C) \geq D \cdot \max\big\{1-\varepsilon(b+d),\ M_\varepsilon(a,b;c,d) - \varepsilon(\beta+\beta')\big\}.$$

Proof. Let $a' \in A$ and $c' \in C$. First, $\|a'c'\| \geq \|pq\| - \|pa'\| - \|qc'\| \geq (1-\varepsilon(b+d))D$. Second, let $\ell = \operatorname{dist}(X,Y)$, $r = \|px\|/(\varepsilon D)$, and $s = \|qy\|/(\varepsilon D)$, so that $r \in [a,b]$ and $s \in [c,d]$. Since $\{X,Y\}$ is ε-separated, $\ell \geq \operatorname{diam}(X)/\varepsilon \geq \|px\|/\varepsilon = rD$, and similarly $\ell \geq sD$. By Observation 2, $\ell \geq D/(1+2\varepsilon)$. Moreover, $\|xy\| \geq \ell$ and $\|xy\| \geq \|pq\| - \|px\| - \|qy\| = (1-\varepsilon(r+s))D$. Hence $\|xy\| \geq M_\varepsilon(a,b;c,d)\,D$, and $\|a'c'\| \geq \|xy\| - \operatorname{diam}(A) - \operatorname{diam}(C) \geq (M_\varepsilon(a,b;c,d) - \varepsilon(\beta+\beta'))D$. ◀

▶ **Lemma 14.** *If $\mathcal{L}$ is ε-admissible, then, in every step, with anchor $\{p,q\}$, the $\mathcal{L}$-collection covers R_{pq}.*

Proof. Let $\{x, y\} \in R_{pq}$. By Observation 4, we may name x and y so that $\{\{p, x\}, \{q, y\}\}$ has disjoint sides and is ε-separated. Its distance is at most D, so $\|px\| \le \varepsilon D$ and $\|qy\| \le \varepsilon D$. Since $x \succeq p$ by the definition of R_{pq}, the first coordinate of x is at least that of p, so $x \in p + \varepsilon D\, H$. Hence $x \in A_\Pi$ for some $\Pi \in \mathcal{L}^+$, and similarly $y \in C_\Sigma$ for some $\Sigma \in \mathcal{L}^\circ$. By the definition of a piece family, $A = A_\Pi$ and $C = C_\Sigma$ satisfy the conditions of Lemma 13 with $(a, b, \beta) = (a_\Pi, b_\Pi, \beta_\Pi)$ and $(c, d, \beta') = (a_\Sigma, b_\Sigma, \beta_\Sigma)$. By Lemma 13 and (1),

$$\operatorname{dist}(A_\Pi, C_\Sigma) \ge D \max\{\beta_\Pi, \beta_\Sigma\} \ge \max\{\operatorname{diam}(A_\Pi), \operatorname{diam}(C_\Sigma)\}/\varepsilon.$$

The left-hand side is positive, so the sides are disjoint, and the dumbbell $\{A_\Pi, C_\Sigma\}$ is ε-separated. Hence it belongs to the $\mathcal{L}$-collection, and it covers $\{x, y\}$. ◀

The sweeps. Let $\mathcal{L}$ be a piece family of size c. The *sweep with* $\mathcal{L}$ outputs the $\mathcal{L}$-collection in every step. In every step, the *greedy sweep with* $\mathcal{L}$ computes a greedy cover of R_{pq} as in Section 3.1, except that, if $|R_{pq}| > 16$, it stops after c grown dumbbells. It outputs the greedy cover if the greedy cover covers R_{pq} and has at most as many dumbbells as the $\mathcal{L}$-collection, and it outputs the $\mathcal{L}$-collection otherwise. For the pieces of Section 2 and $\varepsilon \le 1/16$, these are the fixed sweep of Section 2 and the greedy sweep of Section 3.1. As in Section 3.1, the cap does not change the output: if $|R_{pq}| > 16$, every grown dumbbell belongs to the greedy cover, so a greedy cover with more than c dumbbells is larger than the $\mathcal{L}$-collection.

▶ **Theorem 15.** *Let $\varepsilon > 0$, let $\mathcal{L}$ be an ε-admissible piece family of size c, and let $P \subset \mathbb{R}^2$. The sweep and the greedy sweep with $\mathcal{L}$ perform at most OPT_ε steps and output an ε-WSPD of size at most $c\,\mathrm{OPT}_\varepsilon$. For a fixed family and $\varepsilon \ge 1/16$, both run in $O(n^3)$ time and $O(n^2)$ space.*

Proof. *Size and steps.* Every output dumbbell is ε-separated: the grown dumbbells by construction, and those of the $\mathcal{L}$-collection by its tests. By Lemma 14, every step, with anchor $\{p, q\}$, covers R_{pq}, which contains $\{p, q\}$, with at most c dumbbells. As explained before Theorem 7, the proofs of validity and of the approximation factor in Theorem 3 use only this property, with c in place of 40, and Observation 5 bounds the number of steps by OPT_ε.

Running time. Let $\varepsilon \ge 1/16$. Every 1/16-separated dumbbell is ε-separated, so $\mathrm{OPT}_\varepsilon \le \mathrm{OPT}_{1/16} = O(n)$ by the CK construction [7], and there are $O(n)$ steps. We store the covered pairs in an $n \times n$ table and inspect every pair once, as in the proof of Theorem 3. It remains to show that a step takes $O(n^2)$ time. Every point is assigned to its piece in $O(1)$ time, since the family is fixed. The diameters of all sets A_Π and C_Σ are computed in $O(n^2)$ time. Testing the dumbbell $\{A_\Pi, C_\Sigma\}$ takes $O(|A_\Pi| \cdot |C_\Sigma|)$ time, which sums to $O(n^2)$ over all Π and Σ. Enumerating R_{pq} takes $O(n^2)$ time, since compatibility is tested in $O(1)$ time by Observation 4. A dumbbell is grown in $O(n^2)$ time, as in the proof of Theorem 7, and at most $\max\{c, 16\}$ dumbbells are grown. Marking the pairs covered by each of the at most c output dumbbells takes $O(n^2)$ time. Hence the total time is $O(n^3)$, and the table takes $O(n^2)$ space. ◀

In particular, the number of steps of either sweep is a lower bound on OPT_ε for every $\varepsilon > 0$.

Ring families. Radii $0 = r_0 < r_1 < \cdots < r_K = 1$ and positive integers $n_1, \ldots, n_K$ define a partition of H: for $j = 1, \ldots, K$, the jth *ring*, the set of points $z \in H$ with $r_{j-1} \le \|z\| \le r_j$, is cut into n_j annular sectors of angle π/n_j. Similarly, other radii and integers define a

partition of $\mathbb{B}(0,1)$, with sectors of angle $2\pi/n_j$. These pieces form a *ring family*, in which a piece of the jth ring has $a = r_{j-1}$ and $b = r_j$, and β is its diameter, given by the next lemma. As in Section 2, points on the boundary between two pieces are assigned to one of them arbitrarily. The pieces of Section 2 form a ring family with $K = 2$. Figure 21 shows the first four families of Table 6.

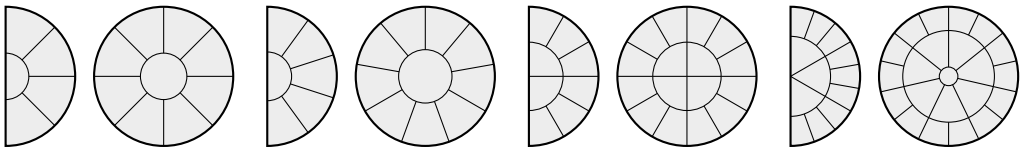
▶ **Lemma 16.** *Let $0 \le a < b$ and $0 < \varphi \le \pi$. The annular sector with radii a and b and angle φ (a circular sector if $a = 0$) has diameter* $\max\{2b\sin(\varphi/2), \sqrt{a^2 + b^2 - 2ab\cos\varphi}\}$.

Proof. Since $\varphi \le \pi$, the convex hull of the sector is bounded by the outer arc, the two radial segments, and the segment between the two inner corners (these corners coincide if $a = 0$). Hence its extreme points are the points of the outer arc and the inner corners, and the diameter is attained by two of them. Two points of the outer arc are at distance at most $2b\sin(\varphi/2)$, with equality for the two outer corners. An inner corner and a point of the outer arc at angle $\theta \in [0, \varphi]$ from it are at distance $\sqrt{a^2 + b^2 - 2ab\cos\theta}$, which increases with θ and is largest for the opposite outer corner. The two inner corners are at distance $2a\sin(\varphi/2) \le 2b\sin(\varphi/2)$. ◀

A central piece of the disk with $n_1 = 1$ is a disk of diameter $2r_1$. Every other piece of a ring family has angle at most π.

▶ **Proposition 17.** *For each ε_0 in Table 6, the ring family given there is ε_0-admissible. Hence, for every $\varepsilon \le \varepsilon_0$, the sweep and the greedy sweep with this family output an ε-WSPD of size at most $c(\varepsilon_0)\,\mathrm{OPT}_\varepsilon$, where $c(\varepsilon_0)$ is its size.*

Proof. In a ring family, a, b, and β depend only on the ring of a piece, so (1) has to be checked only for every pair of rings, one of H and one of $\mathbb{B}(0,1)$. The function minimized in the definition of M_ε is the maximum of four affine functions of (r, s). Hence its minimum over the rectangle $[a, b] \times [c, d]$ is attained at a corner, at a point of an edge where two of the four functions are equal, or at a point where three of them are equal, and M_ε is computed exactly by evaluating these finitely many candidates. We checked (1) at $\varepsilon = \varepsilon_0$ by computer, in floating-point arithmetic, for all pairs of rings of the ten families. The smallest slack is $2 \cdot 10^{-5}$, far above the rounding errors. The second part follows from Theorem 15. ◀



$\varepsilon_0 = 1/10$, $5 \cdot 9 = 45$ | $\varepsilon_0 = 1/8$, $6 \cdot 10 = 60$ | $\varepsilon_0 = 3/16$, $8 \cdot 16 = 128$ | $\varepsilon_0 = 1/4$, $12 \cdot 22 = 264$

Figure 21 The ring families of Table 6 for $\varepsilon_0 = 1/10$, $1/8$, $3/16$, and $1/4$: the pieces of the half-disk H (left) and of the disk $\mathbb{B}(0,1)$ (right), and the size $|\mathcal{L}^+| \cdot |\mathcal{L}^\circ|$ of the family.

▶ **Corollary 18.** *Let $\varepsilon > 0$ and $\beta = \max\{1 - 2\varepsilon, (1 + 2\varepsilon)^{-2}\}$. Some ε-admissible piece family has size at most $4\pi^2(1 + 1/\beta)^4$, which is less than $4 \cdot 10^5 \max\{1, \varepsilon^8\}$. Hence the sweep and the greedy sweep with this family output an ε-WSPD of size at most $4 \cdot 10^5 \max\{1, \varepsilon^8\}\,\mathrm{OPT}_\varepsilon$.*

Proof. Let the pieces be the nonempty intersections of H, and of $\mathbb{B}(0,1)$, with the half-open squares of a grid of side $\beta/\sqrt{2}$, with $a_\Pi = 0$, $b_\Pi = 1$, and $\beta_\Pi = \beta$ for every piece Π. If

ε_0	$c(\varepsilon_0)$		pieces	outer radii of the rings ($\times 1/240$)	pieces per ring
1/10	45	H	5	80, 240	1, 4
		$\mathbb{B}(0,1)$	9	80, 240	1, 8
1/8	60	H	6	84, 240	1, 5
		$\mathbb{B}(0,1)$	10	92, 240	1, 9
3/16	128	H	8	118, 240	2, 6
		$\mathbb{B}(0,1)$	16	118, 240	4, 12
1/4	264	H	12	138, 240	3, 9
		$\mathbb{B}(0,1)$	22	32, 158, 240	1, 7, 14
3/8	578	H	17	108, 180, 240	3, 6, 8
		$\mathbb{B}(0,1)$	34	108, 180, 240	6, 12, 16
1/2	1431	H	27	83, 142, 184, 240	3, 6, 9, 9
		$\mathbb{B}(0,1)$	53	47, 119, 175, 240	1, 9, 21, 22
5/8	2850	H	38	54, 100, 146, 190, 240	2, 5, 10, 11, 10
		$\mathbb{B}(0,1)$	75	40, 90, 136, 180, 240	1, 9, 16, 24, 25
3/4	5151	H	51	35, 84, 120, 149, 189, 240	1, 5, 10, 11, 12, 12
		$\mathbb{B}(0,1)$	101	35, 84, 120, 149, 189, 240	1, 10, 20, 22, 24, 24
7/8	9112	H	68	26, 64, 100, 122, 154, 194, 240	1, 4, 10, 12, 14, 14, 13
		$\mathbb{B}(0,1)$	134	26, 64, 96, 122, 154, 190, 240	1, 8, 16, 27, 28, 25, 29
1	14365	H	85	21, 54, 77, 103, 127, 157, 192, 240	1, 4, 7, 13, 15, 15, 14, 16
		$\mathbb{B}(0,1)$	169	11, 48, 75, 102, 126, 156, 192, 240	1, 7, 14, 26, 30, 30, 29, 32

Table 6 Ring families that are ε_0-admissible, and their sizes $c(\varepsilon_0)$. For each ε_0, the two rows give the rings of the half-disk H and of the disk $\mathbb{B}(0,1)$: the outer radius of each ring, in units of 1/240, and its number of pieces. For $\varepsilon \leq 1/16$, the pieces of Section 2 give $c = 40$.

$\beta = 1 - 2\varepsilon$, then the first term of (1) equals β. Otherwise, $\beta = (1+2\varepsilon)^{-2}$, and since $M_\varepsilon \geq 1/(1+2\varepsilon)$, the second term is at least $1/(1+2\varepsilon) - 2\varepsilon/(1+2\varepsilon)^2 = (1+2\varepsilon)^{-2} = \beta$. Hence the family is ε-admissible. A square that meets $\mathbb{B}(0,1)$ lies in $\mathbb{B}(0,1+\beta)$, since its diameter is β. Comparing areas, at most $\pi(1+\beta)^2/(\beta^2/2) = 2\pi(1+1/\beta)^2$ squares meet $\mathbb{B}(0,1)$, and hence H, so the family has size at most $4\pi^2(1+1/\beta)^4$. If $\varepsilon \leq 1$, then $\beta \geq 1/9$ and $4\pi^2(1+1/\beta)^4 \leq 4\pi^2 \cdot 10^4 < 4 \cdot 10^5$. If $\varepsilon \geq 1$, then $1 + 1/\beta = 1 + (1+2\varepsilon)^2 \leq 10\varepsilon^2$, and the size is less than $4 \cdot 10^5 \varepsilon^8$. ◀

7 Quadtree and Fair Split Tree Partition WSPD Constructions

We first prove Theorem 19, on a quadtree construction, and then bound the size of the CK construction on a fair split tree by the same charging (Theorem 20).

The running times of this section are for dumbbells given as pairs of nodes of a tree, as in [7]. Listing the points of every dumbbell may take $\Omega(n^2)$ time even if $\mathrm{OPT}_\varepsilon = n - 1$. Consider the points $p_i = (i, (4/\varepsilon)^i)$ of Section 1, and let p_j be the point of largest index in an ε-separated dumbbell. If the side of p_j contained another point p_l, its diameter would be at least $(1 - \varepsilon/4)(4/\varepsilon)^j$, while its distance to a point p_i of the other side is at most $\|p_i p_l\| \leq 2(4/\varepsilon)^{j-1} = (\varepsilon/2)(4/\varepsilon)^j$, and $1 - \varepsilon/4 > \varepsilon^2/2$. Hence this side is $\{p_j\}$, a dumbbell with m points covers $m - 1$ pairs, and the dumbbells of every ε-WSPD of these points have at least $\binom{n}{2}$ points in total.

The quadtree construction is a variant of the WSPD construction on a compressed quadtree [14, Chapter 3]. That construction replaces the larger of two cells that are not separated by its children, and tests separation on the cells. Ours replaces both cells, so it outputs pairs of cells of the same level. We analyze it with separation tested on the cells and on the points. For the net-tree construction in doubling metrics, a bound of $O(\mathrm{OPT}_\varepsilon)$ is proved in [5, Section 4] by a similar charging to an optimal WSPD. We make the constant explicit in the plane. The randomized algorithm uses a randomly shifted quadtree [2], [14, Chapter 11], together with a random scale common to all levels, as in the tree embeddings of [11].

Since translations preserve separation, we assume that $P \subset [0, L)^2$ for some $L > 0$. The construction uses a square $R_0 = o + [0, S)^2$ that contains P, with $o \in \mathbb{R}^2$ and $S \geq L$. The deterministic algorithm uses $o = (0, 0)$ and $S = L$. The randomized algorithm draws U uniformly from $[0, 1)$, sets $S = 2^{1+U}L$, and draws o uniformly from $(-S/2, 0]^2$. Then R_0 contains $[0, L)^2$, because $o + S > S/2 \geq L$ in each coordinate. For an integer $i \geq 0$, let $s_i = 2^{-i}S$. The *cells* of level i are the 4^i squares $o + \big([as_i, (a+1)s_i) \times [bs_i, (b+1)s_i)\big)$ with integers $0 \leq a, b < 2^i$. Every point of R_0 lies in exactly one cell of each level, and every cell of level i lies in exactly one cell of each level $j \leq i$, its *ancestor* of level j. For a cell c, let $P_c = P \cap c$. For two distinct cells c and c' of the same level such that P_c and $P_{c'}$ are nonempty, the *box test* accepts if the diameter of the cells is at most ε times the distance between them, and the *exact test* accepts if $\{P_c, P_{c'}\}$ is ε-separated. Fix one of the two tests. A *cell dumbbell* is a set $\{c, c'\}$ of two such cells that the test accepts. Then $\{P_c, P_{c'}\}$ is ε-separated, since $P_c \subseteq c$ and $P_{c'} \subseteq c'$. A cell dumbbell covers the pairs covered by $\{P_c, P_{c'}\}$. A cell dumbbell $\{c, c'\}$ of level i is *maximal* if, for every $j < i$, the ancestors of c and c' of level j do not form a cell dumbbell. Let $\mathcal{W}_\square$ be the collection of the dumbbells $\{P_c, P_{c'}\}$ over all maximal cell dumbbells $\{c, c'\}$. We show that $\mathcal{W}_\square$ satisfies the size bounds of Theorem 19 with both tests, and the running time with the box test.

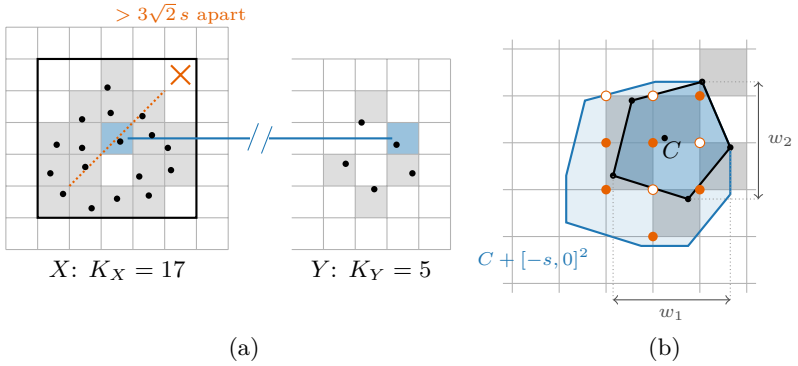


Figure 22 (a) The cells of level i met by the sides X and Y of an optimal dumbbell are shaded. Every dumbbell of $\mathcal{W}_\square$ charged to $\{X, Y\}$ is determined by a pair of shaded cells, such as the blue pair (the distance is not to scale). Here $\varepsilon \leq 1/5$ and $m = 5$: X meets cells of a 5×5 block, but not two opposite corners of it. (b) The lower-left corners of the cells that meet X (filled dots) lie in $C + [-s, 0]^2$, where C is the convex hull of X. Other points of the grid may lie in it too (hollow dots).

▶ **Theorem 19.** *For $P \subset \mathbb{R}^2$ and $\varepsilon \leq 1/5$, a partition ε-WSPD of size at most $529\,\mathrm{OPT}_\varepsilon$ can be computed in $O(n \log n + \mathrm{OPT}_\varepsilon)$ time, and a randomized algorithm with the same running time computes one of expected size at most $252\,\mathrm{OPT}_\varepsilon$, where a dumbbell is given as a pair of*

nodes of a compressed quadtree. For $P \subset \mathbb{R}^d$, one of size at most $(\lfloor 2\sqrt{d}(1+2\varepsilon)\rfloor + 2)^{2d}\mathrm{OPT}_\varepsilon$ can be computed in polynomial time.

Proof. *Partition.* Let $\{x, y\}$ be a pair, and let c_i and c'_i be the cells of level i that contain x and y, respectively. A cell dumbbell of level i covers $\{x, y\}$ only if it is $\{c_i, c'_i\}$. For i large enough, $c_i \neq c'_i$ and both tests accept, the exact test because $P_{c_i} = \{x\}$ and $P_{c'_i} = \{y\}$, and the box test because the diameter of the cells tends to 0 and their distance to $|xy|$. So $\{c_i, c'_i\}$ is a cell dumbbell. Let i_0 be the smallest level for which $\{c_{i_0}, c'_{i_0}\}$ is a cell dumbbell. It is maximal, and no cell dumbbell $\{c_i, c'_i\}$ with $i > i_0$ is maximal, because its ancestors of level i_0 form a cell dumbbell. Hence exactly one maximal cell dumbbell covers $\{x, y\}$. In particular, distinct maximal cell dumbbells cover different pairs and give distinct dumbbells, so $\mathcal{W}_\square$ is a partition ε-WSPD.

Size. Fix a minimum ε-WSPD $\mathcal{W}^*$. Charge every dumbbell of $\mathcal{W}_\square$ to a dumbbell of $\mathcal{W}^*$ that covers one of its pairs. Let $Z = \{X, Y\} \in \mathcal{W}^*$, let $\ell = \mathrm{dist}(X, Y)$, and let $\delta = \varepsilon\ell/(1+2\varepsilon)$, so that $\delta = \varepsilon(\ell - 2\delta)$. Since $\delta < \ell < \sqrt{2}L \leq \sqrt{2}S$, there is a level $i \geq 1$ whose cells have side $s = s_i$ with $\delta/(2\sqrt{2}) < s \leq \delta/\sqrt{2}$. Its cells have diameter at most δ. Let $x \in X$ and $y \in Y$, and let c and c' be the cells of level i that contain them. Every point of c is within distance δ of x, and every point of c' is within distance δ of y. Hence $\mathrm{dist}(c, c') \geq \ell - 2\delta > 0$, so $c \neq c'$, and since $\delta = \varepsilon(\ell - 2\delta)$, the box test accepts. Then the exact test accepts as well, so $\{c, c'\}$ is a cell dumbbell. By the first part, the maximal cell dumbbell that covers $\{x, y\}$ is formed by the ancestors of c and c' of some level, so it is determined by $\{c, c'\}$. Therefore, the number of dumbbells charged to Z is at most $K_X K_Y$, where K_X and K_Y are the numbers of cells of level i that meet X and Y, respectively (Figure 22a).

Cells met by a small set. For $D > 0$, let $m = \lceil D \rceil + 1$, and let $\kappa(D) = m^2 - 2$ if $D \leq (m-2)\sqrt{2}$ and $\kappa(D) = m^2$ otherwise. We claim that a set A of diameter at most Ds meets at most $\kappa(D)$ cells of level i. If two points of A lie in cells whose columns differ by $j \geq 1$, their first coordinates differ by more than $(j-1)s$, so $j - 1 < D$ and $j \leq m - 1$. Hence A meets at most m columns and, similarly, at most m rows. If it meets fewer than m columns or fewer than m rows, it meets at most $m(m-1) \leq m^2 - 2$ cells. Otherwise, it meets only cells of an $m \times m$ block. Two points in opposite corner cells of this block are at distance more than $(m-2)\sqrt{2}\,s$, so if $D \leq (m-2)\sqrt{2}$, then A meets at most one cell of each of the two pairs of opposite corners.

Deterministic bound. We have $\mathrm{diam}(X) \leq \varepsilon\ell = (1+2\varepsilon)\delta < 2\sqrt{2}(1+2\varepsilon)s$, and $2\sqrt{2}(1+2\varepsilon) < 3.96$ for $\varepsilon \leq 1/5$. For $D = 3.96$, we have $m = 5$ and $D \leq 3\sqrt{2}$, so $K_X \leq \kappa(D) = 23$. The same holds for Y. Hence at most $23^2 = 529$ dumbbells are charged to Z, and $|\mathcal{W}_\square| \leq 529\,|\mathcal{W}^*|$.

Randomized bound. In the randomized algorithm, $s = 2^{1+U-i}L$, and the level i depends only on U (for fixed Z). Let $V = \log_2\big(\delta/(\sqrt{2}s)\big)$. Then $V \in [0, 1)$ and $V = \log_2\big(\delta/(2\sqrt{2}L)\big) - U + i$, so V is the fractional part of $\log_2\big(\delta/(2\sqrt{2}L)\big) - U$, and it is uniformly distributed in $[0, 1)$. Let $\gamma = 1.4\sqrt{2}$ and $D_V = \gamma\, 2^V$. For $\varepsilon \leq 1/5$, $\mathrm{diam}(X) \leq (1+2\varepsilon)\delta = (1+2\varepsilon)\sqrt{2}\,2^V s \leq D_V s$, and the same holds for Y, so $K_Y \leq \kappa(D_V)$.

Fix U. The lower-left corners of the cells of level i lie in the lattice $o + s\mathbb{Z}^2$. Since $i \geq 1$, $S/2 = 2^{i-1}s$ is a multiple of s, so each coordinate of o is uniformly distributed modulo s, and the lattice is a uniformly random translate of $s\mathbb{Z}^2$. The expected number of its points in a measurable set is the area of the set divided by s^2. The cell with lower-left corner g meets X if and only if $g \in X + (-s, 0]^2$. This set is contained in $C + [-s, 0]^2$, where C is the convex hull of X. Let $w_1, w_2 \leq \mathrm{diam}(X)$ be the widths of C along the two axes. By Cavalieri's principle, adding a horizontal segment of length s to a convex set increases its area by s times its vertical width, and similarly for a vertical segment, so the area of $C + [-s, 0]^2$ is $\mathrm{area}(C) + s(w_1 + w_2) + s^2$ (Figure 22b). By the isodiametric inequality,

$\text{area}(C) \le \pi \,\text{diam}(X)^2/4$. Therefore, $\mathbb{E}[K_X \mid U] \le e(D_V)$, where $e(D) = \pi D^2/4 + 2D + 1$. Since $K_Y \le \kappa(D_V)$, we get $\mathbb{E}[K_X K_Y \mid U] \le \kappa(D_V)\, e(D_V)$. With the substitution $D = \gamma\, 2^v$,

$$\mathbb{E}[K_X K_Y] \le \int_0^1 \kappa(\gamma\, 2^v)\, e(\gamma\, 2^v)\, dv = \frac{1}{\ln 2} \int_\gamma^{2\gamma} \kappa(D) \Big(\frac{\pi D}{4} + 2 + \frac{1}{D} \Big) dD.$$

The function κ equals 9 on $[\gamma, 2]$, 14 on $(2, 2\sqrt{2}]$, 16 on $(2\sqrt{2}, 3]$, and 23 on $(3, 2\gamma]$. With the antiderivative $\pi D^2/8 + 2D + \ln D$, the right side is less than 251.6. Summing over the dumbbells of $\mathcal{W}^*$ gives $\mathbb{E}|\mathcal{W}_\square| \le 252\, |\mathcal{W}^*|$.

Running time. We use the compressed quadtree of P, which is computed in $O(n \log n)$ time in the real RAM with the floor and logarithm functions [14]. For every node c of the compressed quadtree and every two distinct cells $a, b \subseteq c$ of the next level such that P_a and P_b are nonempty, we call $\textsc{Visit}(a, b)$. Consider a call $\textsc{Visit}(c, c')$. If P_c and $P_{c'}$ are single points, it outputs $\{P_c, P_{c'}\}$. Otherwise, let j be the smallest level greater than that of c at which P_c or $P_{c'}$ meets at least two cells; it is found in the compressed quadtree. From the level of c to level $j-1$, the cells that contain P_c and $P_{c'}$ contain no other point. If the two cells of one of these levels form a cell dumbbell, then so do those of level $j-1$: the exact test gives the same answer at all these levels, and the box test is monotone, because from a level to the next the diameter of the cells halves and their distance does not decrease. Therefore, the call tests the two cells of level $j-1$ and outputs $\{P_c, P_{c'}\}$ if the test accepts. Otherwise, it makes a call $\textsc{Visit}(a, b)$ for every nonempty cell $a \subseteq c$ and every nonempty cell $b \subseteq c'$ of level j. Hence the calls output exactly the dumbbells of $\mathcal{W}_\square$, each as a pair of nodes of the compressed quadtree. Every call that outputs nothing makes at least two calls, and there are $O(n)$ initial calls, so there are $O(n + |\mathcal{W}_\square|)$ calls in total. The deterministic bound holds for every square R_0 that contains P, so $|\mathcal{W}_\square| \le 529\, \text{OPT}_\varepsilon$ in the randomized algorithm too, and there are $O(n + \text{OPT}_\varepsilon)$ calls. With the box test, a call takes $O(1)$ time, since the two cells of level $j-1$ are computed from one point of P_c and one point of $P_{c'}$ with the floor function, and the total time is $O(n \log n + \text{OPT}_\varepsilon)$. This is $O(n \log n + n/\varepsilon^2)$, since $\text{OPT}_\varepsilon = O(n/\varepsilon^2)$ by the CK construction [7]. With the exact test, a call takes $O(n \log n)$ time, computing the diameters from the convex hulls and the distance from the Delaunay triangulation of $P_c \cup P_{c'}$, which contains a closest pair with one point in each side, and the total time is $O((n + \text{OPT}_\varepsilon)\, n \log n)$, which is $O(n^2 \min\{n, 1/\varepsilon^2\} \log n)$ since also $\text{OPT}_\varepsilon \le \binom{n}{2}$. Drawing R_0 takes $O(1)$ time.

Higher dimensions. The deterministic bound extends with cubes of diameter $\sqrt{d}\, s$ in place of squares: the projection of X on each axis is an interval of length less than $2\sqrt{d}(1 + 2\varepsilon)s$, which meets at most $\lfloor 2\sqrt{d}(1 + 2\varepsilon) \rfloor + 2$ of the intervals $[as, (a+1)s)$. ◀

The exact test never gives more dumbbells than the box test. Every cell dumbbell of the box test is also a cell dumbbell of the exact test, so every maximal cell dumbbell of the box test lies in a unique maximal cell dumbbell of the exact test, formed by ancestors of its cells. Every maximal cell dumbbell of the exact test covers some pair and therefore contains the maximal cell dumbbell of the box test that covers this pair. By Markov's inequality, for $\varepsilon \le 1/5$, one run of the randomized algorithm outputs more than $504\, \text{OPT}_\varepsilon$ dumbbells with probability at most $1/2$, so the smallest output of k independent runs has size at most $504\, \text{OPT}_\varepsilon$ with probability at least $1 - 2^{-k}$. For $\varepsilon \le 1/16$, the randomized bound computed with $\gamma = 1.125\sqrt{2}$ is 141, with the same pieces of κ. For every $\varepsilon < 1$, the same proofs give the factors $98^2 = 9604$ (with $D = 6\sqrt{2}$, $m = 10$) and 2812 (with $\gamma = 3\sqrt{2}$). In $\mathbb{R}^d$, the factor is $5^6 = 15625$ for $d = 3$ and $\varepsilon \le 1/16$.

Fair split trees. The charging of Theorem 19 also applies to the CK construction on a fair split tree [7]. We first consider the version used in Section 5, which has the tree of [7] but differs in the separation test and in the choice of the node that is split, and then the original construction. A *box* is an axis-parallel rectangle. Every node u of the tree T has a point set $P_u \subseteq P$, and b_u denotes the smallest box that contains P_u. The root has the point set P, and the leaves are the nodes with one point. Every other node u is split by the line through the center of b_u that is perpendicular to a longest side of b_u: the point sets of its two children are the points of P_u on each side of the line, where the points on the line go to a fixed side. For two nodes u and v with disjoint point sets, the *box test* accepts if $\max\{\operatorname{diam}(b_u), \operatorname{diam}(b_v)\} \leq \varepsilon \operatorname{dist}(b_u, b_v)$, and the *exact test* accepts if the box test accepts or $\{P_u, P_v\}$ is ε-separated. Fix one of the two tests. The procedure $\textsc{Pairs}(u, v)$ outputs $\{P_u, P_v\}$ if the test accepts. Otherwise, it takes the node whose box has the larger diameter, say u, breaking ties arbitrarily, and calls $\textsc{Pairs}(u_1, v)$ and $\textsc{Pairs}(u_2, v)$ for the two children u_1, u_2 of u. This node is not a leaf, since both tests accept two leaves. Let $\mathcal{W}_T$ be the collection of the dumbbells output by the calls $\textsc{Pairs}(u_1, u_2)$ over all nodes of T with children u_1, u_2, where a dumbbell is output as a pair of nodes.

▶ **Theorem 20.** *For $P \subset \mathbb{R}^2$ and $\varepsilon \leq 1/5$, the collection $\mathcal{W}_T$ is a partition ε-WSPD of size at most $1024\,\mathrm{OPT}_\varepsilon$, both with the box test and with the exact test. With the box test, it is computed in $O(n \log n + \mathrm{OPT}_\varepsilon)$ time. For $P \subset \mathbb{R}^d$, its size is at most $\lfloor (2\sqrt{d}(1+2\varepsilon) + 2)^d \rfloor^2 \mathrm{OPT}_\varepsilon$.*

Proof. *Partition.* Every dumbbell of $\mathcal{W}_T$ is ε-separated, because $P_u \subseteq b_u$ for every node u. Let $\{x, y\}$ be a pair. The calls $\textsc{Pairs}(u, v)$ with $x \in P_u$ and $y \in P_v$, or vice versa, form a chain: it starts with the call for the two children of the lowest common ancestor of the leaves of x and y, every call of the chain that outputs nothing makes exactly one call of the chain, and the chain ends with an output, since two leaves are accepted. Hence exactly one dumbbell of $\mathcal{W}_T$ covers $\{x, y\}$.

Size. Fix a minimum ε-WSPD $\mathcal{W}^*$, and charge every dumbbell of $\mathcal{W}_T$ to a dumbbell of $\mathcal{W}^*$ that covers one of its pairs. Let $Z = \{X, Y\} \in \mathcal{W}^*$, and let $\ell = \operatorname{dist}(X, Y)$ and $\delta = \varepsilon\ell/(1+2\varepsilon)$, as in the proof of Theorem 19. We regard a node as an ancestor of itself. The diameter of the box does not decrease from a node to its parent. For a point x, let $u(x)$ be the highest ancestor of the leaf of x whose box has diameter at most δ. The box of the root has diameter at least $\ell > \delta$, so $u(x)$ is not the root, and the box of every proper ancestor of $u(x)$ has diameter greater than δ.

Let $x \in X$ and $y \in Y$. Every point of $b_{u(x)}$ is within distance δ of x, and every point of $b_{u(y)}$ is within distance δ of y. Hence $\operatorname{dist}(b_{u(x)}, b_{u(y)}) \geq \ell - 2\delta > 0$, and since $\delta = \varepsilon(\ell - 2\delta)$, both tests accept $u(x)$ and $u(y)$. In particular, $P_{u(x)}$ and $P_{u(y)}$ are disjoint, so the first call of the chain of $\{x, y\}$ is on proper ancestors of $u(x)$ and $u(y)$ or on these nodes themselves. We claim that every call $\textsc{Pairs}(u, v)$ of the chain, with $x \in P_u$, is on an ancestor u of $u(x)$ and an ancestor v of $u(y)$. Consider such a call that outputs nothing. Then $u \neq u(x)$ or $v \neq u(y)$. If $u = u(x)$, then v is a proper ancestor of $u(y)$, so $\operatorname{diam}(b_v) > \delta \geq \operatorname{diam}(b_u)$ and the call splits v. Similarly, if $v = u(y)$, the call splits u. Otherwise, both nodes are proper ancestors. In each case, the next call of the chain is on ancestors of $u(x)$ and $u(y)$. Therefore, the dumbbell of $\mathcal{W}_T$ that covers $\{x, y\}$ is output by a call on ancestors of $u(x)$ and $u(y)$. The calls of the chain depend only on these ancestors, so this dumbbell is determined by $u(x)$ and $u(y)$. The number of dumbbells charged to Z is thus at most $K_X K_Y$, where K_X is the number of distinct nodes $u(x)$ with $x \in X$, and K_Y the number of distinct nodes $u(y)$ with $y \in Y$.

Packing. We bound K_X using the outer rectangles of [7]. Let λ_u be the length of a longest side of b_u. The *outer rectangle* $\hat{b}_u$ of the root u is a square of side λ_u that contains b_u. For a child u of a node v, $\hat{b}_u$ is the part of $\hat{b}_v$ on the side of P_u of the line that splits v. Then $b_u \subseteq \hat{b}_u$, and the outer rectangles of two nodes have disjoint interiors if neither is an ancestor of the other. Moreover, both sides of $\hat{b}_u$ have length at least $\lambda_v/2$, where v is the parent of u. For the sides perpendicular to the line that splits v, this holds because $b_v \subseteq \hat{b}_v$ extends $\lambda_v/2$ on each side of the line. The other sides are as long as those of $\hat{b}_v$, whose length is λ_v if v is the root, and at least $\lambda_{v'}/2 \geq \lambda_v/2$ by induction otherwise, where v' is the parent of v.

Let $m = \delta/(2\sqrt{2})$. The box of the parent v of $u(x)$ has diameter greater than δ, so $\lambda_v > \delta/\sqrt{2}$ and both sides of $\hat{b}_{u(x)}$ are longer than m. Hence $\hat{b}_{u(x)}$ contains an axis-parallel square of side m that contains x. No node $u(x)$ is a proper ancestor of another one, so these squares, one for each of the K_X nodes, have disjoint interiors. They lie in $C + [-m, m]^2$, where C is the convex hull of X. As in the proof of Theorem 19, the area of this set is $\text{area}(C) + 2m(w_1 + w_2) + 4m^2$, where $w_1, w_2 \leq \text{diam}(X)$ are the widths of C along the two axes, and $\text{area}(C) \leq \pi\,\text{diam}(X)^2/4$. If $\text{diam}(X) \leq Dm$, the area is thus at most $(\pi D^2/4 + 4D + 4)\,m^2$. We have $\text{diam}(X) \leq \varepsilon\ell = (1 + 2\varepsilon)\delta = 2\sqrt{2}(1 + 2\varepsilon)m$, which is less than $3.96\,m$ for $\varepsilon \leq 1/5$. Hence $K_X \leq \pi D^2/4 + 4D + 4 < 33$ for $D = 3.96$, that is, $K_X \leq 32$, and the same holds for K_Y. At most $32^2 = 1024$ dumbbells are charged to Z, and $|\mathcal{W}_T| \leq 1024\,|\mathcal{W}^*|$.

Running time. The tree is computed in $O(n \log n)$ time [7]. Every call that outputs nothing makes two calls, and there are $n - 1$ initial calls, so there are $O(n + |\mathcal{W}_T|)$ calls. With the box test, a call takes $O(1)$ time. Since $|\mathcal{W}_T| \leq 1024\,\text{OPT}_\varepsilon$, the total time is $O(n \log n + \text{OPT}_\varepsilon)$, which is $O(n \log n + n/\varepsilon^2)$ since $\text{OPT}_\varepsilon = O(n/\varepsilon^2)$ [7].

Higher dimensions. In $\mathbb{R}^d$, a node is split by the hyperplane through the center of its box that is perpendicular to a longest side, and the same proof applies with $m = \delta/(2\sqrt{d})$: the outer box of $u(x)$ contains a cube of side m that contains x, and X lies in a cube of side $\text{diam}(X) \leq 2\sqrt{d}(1 + 2\varepsilon)m$. These cubes lie in a cube of side $\text{diam}(X) + 2m$, so $K_X \leq (2\sqrt{d}(1 + 2\varepsilon) + 2)^d$. ◀

For $\varepsilon \leq 1/16$, the same proof gives the factor $24^2 = 576$ (with $D = 1.125 \cdot 2\sqrt{2}$), and for every $\varepsilon < 1$, the factor $94^2 = 8836$ (with $D = 6\sqrt{2}$). In $\mathbb{R}^3$, the factor is $205^2 = 42025$ for $\varepsilon \leq 1/16$. With the exact test, a call takes $O(n \log n)$ time, as in the proof of Theorem 19.

The proof also applies to the original construction of [7], with two changes. First, its test, for a separation $s > 0$, accepts two nodes u and v if b_u and b_v are contained in two balls of the same radius r that are at distance at least sr from each other. For $s = 2/\varepsilon$, this test accepts only ε-separated dumbbells, since $\text{diam}(P_u), \text{diam}(P_v) \leq 2r$ and $\text{dist}(P_u, P_v) \geq 2r/\varepsilon$. It accepts $u(x)$ and $u(y)$: for $r = \max\{\text{diam}(b_{u(x)}), \text{diam}(b_{u(y)})\}/2 \leq \delta/2$, every point of the balls of radius r centered at the centers of the two boxes is within distance $2r \leq \delta$ of x or of y, so the distance between the balls is at least $\ell - 2\delta = \delta/\varepsilon \geq sr$. Second, Pairs splits the node whose box has the longer side. Then $u(x)$ is defined as the highest ancestor of the leaf of x with $\lambda_{u(x)} \leq \delta/\sqrt{2}$, so that its box has diameter at most δ and the box of its parent has a side longer than $\delta/\sqrt{2}$, and the rest of the proof is unchanged. Hence, for $P \subset \mathbb{R}^2$ and $\varepsilon \leq 1/5$, the construction of [7] with separation $2/\varepsilon$ computes a partition ε-WSPD of size at most $1024\,\text{OPT}_\varepsilon$ in $O(n \log n + \text{OPT}_\varepsilon)$ time.

8 Conclusion

We gave approximation algorithms for minWSPD with explicit factors that keep the separation ε. For cover WSPDs, the sweep is a 40-approximation in the plane for $\varepsilon \leq 1/16$, a

2048-approximation in $\mathbb{R}^3$, and a $2^{O(d)}$-approximation in $\mathbb{R}^d$. With pieces that depend on ε, it keeps a constant factor for every $\varepsilon > 0$. For partition WSPDs, CK on a fair split tree and the quadtree construction give factors 1024 and 529 in the plane for $\varepsilon \leq 1/5$, and 252 in expectation for a randomly shifted and scaled quadtree, in $O(n \log n + \mathrm{OPT})$ time. The sweep also gives a lower bound on OPT: its anchors are pairwise incompatible, so its number of steps is at most OPT.

On the practical side, the greedy sweep keeps the factor 40, and reshape shrinks any given WSPD without increasing its size. A local search on the anchors of the sweep enlarges its lower bound to LB, which is at least 0.989 OPT on every planar instance where OPT is known, and certifies the quality of any WSPD on inputs where OPT is out of reach. In the plane, the greedy sweep followed by reshape is within 0.02% of OPT for $n = 500$ and on average 1.05 LB for $n = 10000$. Converting its output gives partition WSPDs of on average 1.13 LB, against 1.55 LB for CK. In $\mathbb{R}^3$, reshape is at most 0.03% above OPT where OPT is known, and on average 1.021 LB for $n = 10000$.

Open problem. Without the final deletion of redundant dumbbells, which can only decrease the size, the largest ratio we observed between the output of the greedy sweep and the smallest known WSPD is 1.503, for `jupiter-0020000` with $\varepsilon = 1/20$. Hence its approximation factor for small ε is between 1.503 and 40. What is it?

The largest observed ratio grows with n, from 1.41 for $n = 10000$ to 1.50 for $n = 20000$, so the lower end may be far from the truth. The upper end is the product of two bounds: the number of steps is at most OPT, and every step outputs at most 40 dumbbells. An analysis that bounds each step separately cannot improve it much: for $\varepsilon = 1/16$, we found point sets in which the set R_{pq} of a single step needs 23 dumbbells. A smaller factor seems to need an amortized argument, which pays for a step whose R_{pq} needs many dumbbells with optimal dumbbells that cover pairs outside R_{pq}.

References

1 Diogo V. Andrade, Mauricio G. C. Resende, and Renato F. Werneck. Fast local search for the maximum independent set problem. *Journal of Heuristics*, 18(4):525–547, 2012. doi:10.1007/s10732-012-9196-4.

2 Sanjeev Arora. Polynomial time approximation schemes for Euclidean traveling salesman and other geometric problems. *J. ACM*, 45(5):753–782, 1998. doi:10.1145/290179.290180.

3 Sergei Bespamyatnikh and Michael Segal. Fast algorithms for approximating distances. *Algorithmica*, 33(2):263–269, 2002. doi:10.1007/s00453-001-0114-7.

4 Coen Bron and Joep Kerbosch. Algorithm 457: Finding all cliques of an undirected graph. *Communications of the ACM*, 16(9):575–577, 1973. doi:10.1145/362342.362367.

5 Kevin Buchin, Jacobus Conradi, Sariel Har-Peled, Antonia Kalb, Abhiruk Lahiri, Lukas Plätz, Carolin Rehs, and Sampson Wong. On small pair decompositions for point sets. In *Proc. 34th Annual European Symposium on Algorithms (ESA)*, volume 388 of *LIPIcs*, pages 102:1–102:22, 2026. Full version: arXiv:2601.22728. doi:10.4230/LIPIcs.ESA.2026.102.

6 Paul B. Callahan and S. Rao Kosaraju. Faster algorithms for some geometric graph problems in higher dimensions. In *Proc. 4th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 291–300, 1993.

7 Paul B. Callahan and S. Rao Kosaraju. A decomposition of multidimensional point sets with applications to k-nearest-neighbors and n-body potential fields. *J. ACM*, 42(1):67–90, 1995. doi:10.1145/200836.200853.

8 Guilherme D. da Fonseca. A bicriteria approximation scheme for the minimum well-separated pair decomposition, 2026. arXiv:2610.13650. doi:10.48550/arXiv.2610.13650.

9 Guilherme D. da Fonseca. Pangolin: Plane geometry library. `https://github.com/gfonsecabr/pgl`, 2026.

10 Erik D. Demaine, Sándor P. Fekete, Phillip Keldenich, Dominik Krupke, and Joseph S. B. Mitchell. Area-optimal simple polygonalizations: The CG challenge 2019. *ACM Journal of Experimental Algorithmics*, 27:2.4:1–2.4:12, 2022. `doi:10.1145/3504000`.

11 Jittat Fakcharoenphol, Satish Rao, and Kunal Talwar. A tight bound on approximating arbitrary metrics by tree metrics. *J. Comput. Syst. Sci.*, 69(3):485–497, 2004. `doi:10.1016/j.jcss.2004.04.011`.

12 Sándor P. Fekete, Phillip Keldenich, Dominik Krupke, and Eike Niehs. Edge sparsification for geometric tour problems. *Comput. Geom. Topol.*, 3(1):1:1–1:23, 2024. `doi:10.57717/cgt.v3i1.20`.

13 Andrea Grosso, Marco Locatelli, and Wayne Pullan. Simple ingredients leading to very efficient heuristics for the maximum clique problem. *Journal of Heuristics*, 14(6):587–612, 2008. `doi:10.1007/s10732-007-9055-x`.

14 Sariel Har-Peled. *Geometric Approximation Algorithms*, volume 173 of *Mathematical Surveys and Monographs*. American Mathematical Society, 2011. `doi:10.1090/surv/173`.

15 Sariel Har-Peled and Manor Mendel. Fast construction of nets in low-dimensional metrics and their applications. *SIAM J. Comput.*, 35(5):1148–1184, 2006. `doi:10.1137/S0097539704446281`.

16 Sariel Har-Peled, Benjamin Raichel, and Eliot W. Robson. Well-separated pairs decomposition revisited, 2025. arXiv:2509.05997.

17 Giri Narasimhan and Michiel H. M. Smid. *Geometric Spanner Networks*. Cambridge University Press, 2007. `doi:10.1017/CBO9780511546884`.

18 Wayne Pullan and Holger H. Hoos. Dynamic local search for the maximum clique problem. *Journal of Artificial Intelligence Research*, 25:159–185, 2006. `doi:10.1613/jair.1815`.

19 Michiel H. M. Smid. The well-separated pair decomposition and its applications. In Teofilo F. Gonzalez, editor, *Handbook of Approximation Algorithms and Metaheuristics, Volume 2: Contemporary and Emerging Applications*, chapter 4, pages 71–84. Chapman and Hall/CRC, 2nd edition, 2018. `doi:10.1201/9781351235426`.

20 Stanford Computer Graphics Laboratory. The Stanford 3D scanning repository. `http://graphics.stanford.edu/data/3Dscanrep/`. Meshes used: `https://github.com/alecjacobson/common-3d-test-models`.