Chapter 4

Matching Special Shape Classes Under Translations

In this chapter we again consider matching two shapes under translations. However, this time we exploit the structure of certain special shape classes to speed up the algorithms. We mainly consider the (directed) Hausdorff distance, for which speed-ups can be achieved for a special type of terrains (see Section 4.1.2) and for convex polyhedra (see Section 4.2). For terrains we also consider the perpendicular distance under translations, since the perpendicular distance is a natural distance measure on terrains.

4.1 Matching Terrains

Since matching of arbitrary polyhedral sets is very time consuming we consider in this section the case of terrains: To the best of our knowledge there is no other algorithm known which matches two terrains in higher dimensions.

In Section 4.1.1 we compute the translation that minimizes the perpendicular distance between two terrains in two or three dimensions. In particular, if the complexities of the terrains are m and n, respectively, the runtime for $d = 2$ is $O(mn \log(mn)\alpha(mn))$, and for $d = 3$ it is $O_\delta((mn)^2)$. Here, $\alpha(mn)$ is the extremely slow growing inverse Ackermann function, see [67].

In Section 4.1.2 we compute the directed Hausdorff distance between two terrains F and G under translations, where we require that G is a certain type of terrain (an ε-terrain), whereas F is allowed to be an arbitrary polyhedral set. The results are summarized in Table 4.1. This notion generalizes to the undirected Hausdorff distance, when both F and G are ε-terrains. An ε-terrain is a special type of terrain which guarantees a terrain-like property for its ε-neighborhood.

4.1.1 Perpendicular Distance

In this subsection let two polyhedral terrains $F = \{(x, f(x)) \mid x \in D_f\}$ and $G = \{(x, g(x)) \mid x \in D_g\}$ in $\mathbb{R}^d$ be given. We will concentrate on the case that $d = 2$ or $d = 3$. Since each terrain intersects every line in $\vec{e}_d$-direction at most once, it is natural to consider the height difference between points of F and G lying above each other (with respect to the $\vec{e}_d$-coordinate) as a distance measure. We therefore consider the perpendicular distance. Our

Table 4.1: Results of Section Section 4.1.2 for the decision problem of finding a translation which does not exceed a given directed Hausdorff distance between a polyhedral set F and a polyhedral ε-terrain G, $|F| = n$ and $|G| = m$.

Metric	Dimension	Runtime
L_2, polyhedral	$d = 2$	$O(n \log^2 n + mn \log(mn)\alpha(mn))$
polyhedral	$d = 3$	$O_\delta(m^2 n^4)$
polyhedral	$d \geq 4$	$O(m^d n^{d^2-d} \alpha^d(n))$

task is to translate F in $\mathbb{R}^d$ such that $D_f \subseteq D_g$ and the maximum perpendicular distance between F and G restricted to the translated domain of F is minimized. Due to the special $\vec{e}_d$-direction of terrains we consider a translation $t' = (t_1, \ldots, t_{d-1}, t_d) \in \mathbb{R}^d$ to be composed of a translation $t = (t_1, \ldots, t_{d-1}) \in \mathbb{R}^{d-1}$ and a translation $t_d \in \mathbb{R}$, thus $t' = (t, t_d)$. Using this notation we have $F + t' = \{(x, f(x-t) + t_d) \mid x \in D_f\}$. The problem we wish to solve in this subsection is:

Problem 5 *Let two polyhedral terrains $F = \{(x, f(x)) \mid x \in D_f\}$ and $G = \{(x, g(x)) \mid x \in D_g\}$ in $\mathbb{R}^d$ of complexity m and n, respectively, be given. We wish to compute a translation $(t^*, t_d^*) \in \mathbb{R}^d$, $t^* \in \mathbb{R}^{d-1}$, $t_d^* \in \mathbb{R}$, such that*

$$\delta_\perp(F + (t^*, t_d^*), G) = \min_{\substack{t \in \mathbb{R}^{d-1} \\ D_f + t \subseteq D_g}} \min_{t_d \in \mathbb{R}} \delta_\perp(F + t', G) \tag{4.1}$$

Rewriting $\delta_\perp(F + t', G)$ we obtain

$$\begin{aligned}
\delta_\perp(F + t, G) &= \max_{x \in D_f + t} |f(x-t) - g(x) + t_d| \\
&= \max\{ \max_{x \in D_f} (g(x+t) - f(x)) - t_d \ , \ t_d - \min_{x \in D_f} (g(x+t) - f(x)) \ \} \\
&= \max\{ \max_{x \in D_f} h_x(t) - t_d \ , \ t_d - \min_{x \in D_f} h_x(t)) \ \}
\end{aligned}$$

with $h_x(t) := g(x+t) - f(x)$. Then for each $t \in \mathbb{R}^{d-1}$ the inner maximum is the pointwise maximum, that is the upper envelope of the functions h_x for all $x \in D_f$. Observe that $D_f + t \subseteq D_g$ is equivalent to $t \in \overline{\overline{D_g} \oplus (-D_f)} =: D_h$. Then let

$$\overline{h} : D_h \longrightarrow \mathbb{R} \ ; \quad t \mapsto \max_{x \in D_f} h_x(t)$$

be the upper envelope, and

$$\underline{h} : D_h \longrightarrow \mathbb{R} \ ; \quad t \mapsto \min_{x \in D_f} h_x(t)$$

be the lower envelope. Let $\overline{H}$ and $\underline{H}$ be the respective graphs of $\overline{h}$ and $\underline{h}$. These graphs are polyhedral terrains and are, respectively, the upper and lower envelopes of the functions h_x, for all $x \in D_f$.

Lemma 6 *Let $F, G, \overline{H}, \underline{H}$ be as defined above. Then $\overline{H}$ ($\underline{H}$) is the upper (lower) envelope, with respect to the $\vec{e}_d$-axis, of $G \oplus (-F)$ restricted to the region above D_h.*

Proof:

$$G \oplus (-F) = \bigcup_{z \in D_g} \bigcup_{x \in D_f} (z - x, g(z) - f(x)) = \bigcup_{t \in \mathbb{R}^{d-1}} \bigcup_{\substack{x \in D_f \\ x + t \in D_g}} (t, g(x+t) - f(x))$$

The lemma now follows directly from the definitions of $\overline{H}$ and $\underline{H}$. □

Reformulating (4.1) we thus have

$$\delta_\perp(F + (t^*, t_d^*), G) = \min_{t \in D_h} \min_{t_d \in \mathbb{R}} \max\{\overline{h}(t) - t_d, t_d - \underline{h}(t)\} \quad (4.2)$$

which can easily be simplified to

$$\delta_\perp(F + (t^*, t_d^*), G) = \min_{t \in D_h} \frac{\overline{h}(t) - \underline{h}(t)}{2} = \frac{\overline{h}(t^*) - \underline{h}(t^*)}{2} \quad (4.3)$$

$$\text{with} \quad t_d^* = (\overline{h}(t^*) + \underline{h}(t^*))/2. \quad (4.4)$$

$(t^*, t_d^*) \in \mathbb{R}^d$ with $t^* \in \mathbb{R}^{d-1}$ and $t_d^* \in \mathbb{R}$ is the translation which attains the minimum in (4.3).

Since f and g are piecewise linear, $G \oplus (-F)$ is polyhedral. Let $\overline{L}$ be the upper and $\underline{L}$ be the lower envelope of $G \oplus (-F)$. Clearly $\overline{L}$ and $\underline{L}$ are also piecewise linear, thus polyhedral terrains. The restrictions $\overline{H}$ of $\overline{L}$ and $\underline{H}$ of $\underline{L}$ to the region above D_h are also piecewise linear since D_h is piecewise linear. Therefore, $h^- := (\overline{h} - \underline{h})/2$ is also piecewise linear, as well as the terrain $H^- := \{(x, h^-(x)) \mid x \in D_h\}$ it induces. The simplicial partition M_{h^-} of its domain is by Lemma 6 the overlay of $M_{\overline{L}}$ and $M_{\underline{L}}$ restricted to D_h. The linear function associated with each cell C in M_{h^-} is then simply half the difference of the linear functions assigned to the two cells in $M_{\overline{h}}$ and $M_{\underline{h}}$ that contain C. In order to compute t^{opt}, it suffices to minimize (4.3) over all vertices in M_{h^-}. Since H^- is piecewise linear this yields the optimal solution. This leads to the following algorithm:

1. *Compute the upper envelope $\overline{L}$ and the lower envelope $\underline{L}$ of $G \oplus (-F)$.* The Minkowski sum of two unions of objects is the union of the pairwise Minkowski sums. We consider for each $(d-1)$-simplex ϕ in $-F$ and each $(d-1)$-simplex γ in G the Minkowski sum $\gamma \oplus \phi$. This Minkowski sum is the (d-dimensional) convex hull of d^2 points and has therefore a constant complexity and can also be computed in constant time for d is considered to be constant. $G \oplus (-F)$ is the union of all $\gamma \oplus \phi$, but we do not have to compute this whole union. We only compute the upper envelope of all $\gamma \oplus \phi$. We have nm different (γ, ϕ)-pairs. Each $\gamma \oplus \phi$ is a convex d-polytope (the Minkowski sum of two convex sets is convex), whose surface can be triangulated, i.e. partitioned into a constant number of $(d-1)$-simplices, see e.g. [61]. Altogether we have $O(mn)$ $(d-1)$-simplices in d dimensions. Their lower and upper envelope has complexity $O((mn)^{d-1}\alpha((mn))$ and can be computed with a randomized algorithm in expected $O((mn)^{d-1}\alpha((mn))$ time and space, or deterministically in $O_\delta((mn)^{d-1})$, see [67].

2. *Compute $D_h = \overline{\overline{D_g} \oplus (-D_f)}$.* We compute the volume representation for D_h, as in Section 3.4.2, in $O((mn)^{d-1})$ time.

3. *Overlay* $\overline{L}$ *and* $\underline{L}$*, and clip to within* D_h*.* We compute the overlay of $M_{\overline{L}}$ and $M_{\underline{L}}$, additionally superimposed with the $O(mn)$ hyperplanes defining D_h. By Lemma 6 this yields the simplicial partition M_{h^-}.

4. *Compute (4.3) for every vertex in* M_{h^-}*.* Let t^* be a vertex that assumes the minimum. Plugging t^* into (4.4) yields t^*_d.

In particular, for $d = 2$ we compute $M_{\overline{L}}$ and $M_{\underline{L}}$ in $O(mn\log(mn)\alpha(mn))$ time. In this case, each domain is a set of $O(mn\alpha(mn))$ intervals on a line, such that the overlay has the same complexity and can be computed with a simple sweep in $O(mn\alpha(mn))$ time. Now the clipping is done by additionally overlaying the obtained interval partition with D_h, and computing the desired intervals, again with a simple sweep in $O(mn\alpha(mn))$ time.

For $d = 3$, note that the volume representation for D_h consists of the arrangement of $O(mn)$ lines, see Section 3.4.2. In [67] it is shown that the overlay of $M_{\overline{L}}$ (or $M_{\underline{L}}$) with $O(mn)$ lines does not increase the complexity of the envelope, and it can also be computed in the same time. A result in [3, 67] states that the overlay L of $M_{\overline{L}}$ and $M_{\underline{L}}$ can be computed in $O_\delta((mn)^2)$ time. Both results can be combined to compute the overlay of $M_{\overline{L}}$ and $M_{\underline{L}}$ additionally superimposed with the $O(mn)$ lines defining D_h in $O_\delta((mn)^2)$ time. The minimization over all vertices using (4.3) needs time proportional to the number of vertices processed. We have thus proven the following theorem:

Theorem 4 *Let F and G be two polyhedral terrains with complexities m and n. A translation which minimizes the perpendicular distance between F and G can be computed for $d = 2$ in time $O(mn\log(mn)\alpha(mn))$, and for $d = 3$ in time $O_\delta((mn)^2)$.*

Note that for dimension $d = 4$ Koltun and Sharir [58, 59] showed that the worst-case complexity of the overlay of the subdivisions of the domains of the lower envelopes of arbitrary trivariate functions of constant algebraic description complexity equals the worst-case combinatorial complexity of the envelopes. However there is no result that actually computes the overlay within the same time bounds. For dimensions $d > 4$ there are currently no results for the overlay known, only for the region sandwiched between two envelopes, see [41] and also Theorem 5. *

4.1.2 Directed Hausdorff Distance

In this subsection we consider a faster way of matching ε-terrains, which are a special type of terrains.

Definition 16 *For a given $\varepsilon > 0$, and a given metric, a terrain $G = \{(x, g(x)) \mid x \in D_g\}$ in $\mathbb{R}^d$ is called an* ε-terrain *iff $g(x)$ is continuous over each connected component of D_g, and the intersection of G^ε with an arbitrary line in direction $\vec{e}_d$ is either an interval or empty.*

Let a parameter $\varepsilon > 0$, a polyhedral set $F = \{(x, f(x)) \mid x \in D_f\}$ of complexity m, and an ε-terrain $G = \{(x, g(x)) \mid x \in D_g\}$ in $\mathbb{R}^d$ be given. We consider the decision problem Problem 4 for which we have to decide if there exists a $t \in \mathbb{R}^d$ such that the directed Hausdorff

*After submission of this thesis, Koltun and Wenk [57] generalized the results for overlays of envelopes of piecewise linear functions, and furthermore Theorem 4, to arbitrary dimension.

distance $\vec{\delta}_{\mathrm{H}}(F+t, G) \leq \varepsilon$. Rewriting the result of Lemma 2 we know that this is equivalent to checking if

$$I := \overline{\overline{G^\varepsilon} \oplus (-F)} \neq \emptyset \ . \tag{4.5}$$

We will see that ε-terrains have special properties wich allow us to speed up the algorithm of Section 3.4.

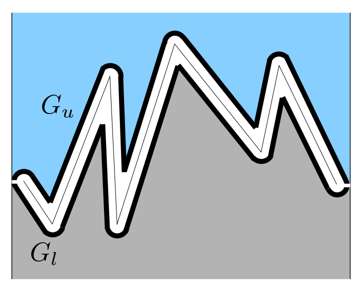
ε-Terrains

We call a metric *projectable* if the projection in direction $\vec{e}_d$ of its closed d-dimensional unit ball results in its $(d-1)$-dimensional unit ball. Clearly the commonly used L_2-metric, L_1-metric and L_∞-metric are projectable.

We consider different classes and properties of ε-terrains. The first important property is the following.

Lemma 7 *Let $G = \{(x, g(x)) \mid x \in D_g\}$ be an ε-terrain in $\mathbb{R}^d$ for a projectable metric. Then there exist interior-disjoint terrains G_u, G_l both over the domain D_g^ε, and a set S of line segments in direction $\vec{e}_d$ connecting the boundaries of G_u and G_l, such that $\partial G^\varepsilon = G_u \cup G_l \cup S$.*

Proof:
Let G_u (G_l) be the set of all upper (lower) endpoints of all line segments which are obtained as the intersection of a line in direction $\vec{e}_d$ with G^ε. Endpoints of degenerate intervals that consist of one point only are assigned to G_u and to G_l. Let S consist of all line segments in direction $\vec{e}_d$ that are fully contained in ∂G^ε; clearly such a line segment can degenerate to a point. Then G_u and G_l are both terrains, and $\partial G^\varepsilon = G_u \cup G_l \cup S$. The projection of $\mathbf{B}_\varepsilon^d$ in direction $\vec{e}_d$ onto $\mathbb{R}^{d-1}$ is $\mathbf{B}_\varepsilon^{d-1}$, by the definition of a projectable metric. Thus G_u and G_l both project onto D_g^ε.



□

Lemma 8 *Every terrain in $\mathbb{R}^d$ which is continuously defined over a convex domain is an ε-terrain for any $\varepsilon > 0$, with respect to any convex polyhedral metric or the Euclidean metric.*

Proof: Let G be the given continuously defined terrain, and let $\varepsilon > 0$. Let l be an arbitrary line in direction $\vec{e}_d$, and assume for the sake of contradiction that there are three points a, b, c on l, a below b below c, such that $a, c \in G^\varepsilon$ but $b \notin G^\varepsilon$. This implies that there cannot be any point of G in $\mathbf{B}_\varepsilon^d(b)$. Now let $p, q \in G$ such that $a \in \mathbf{B}_\varepsilon^d(p)$ and $c \in \mathbf{B}_\varepsilon^d(q)$. This means in particular that both p and q are contained in the cylinder $l \oplus \mathbf{B}_\varepsilon^d$.

Let p' (and q') be the projection of p (and q, respectively) onto the domain D_g of G. By the convexity of D_g the line segment s between p' and q' has to be contained in D_g. From the continuity property we know that there must exist a path π in G from p to q whose projection onto D_g is s. Hence, π is contained in $l \oplus \mathbf{B}_\varepsilon^d$. However, observe that $\mathbf{B}_\varepsilon^d(b)$ divides the cylinder $l \oplus \mathbf{B}_\varepsilon^d$ into two parts, and p lies below and q above $\mathbf{B}_\varepsilon^d(b)$. Thus π has to intersect $\mathbf{B}_\varepsilon^d(b)$, which is a contradiction. □

Since every connected subset of $\mathbb{R}$ is connected, we know that for $d = 2$ the result of Lemma 8 holds for any connected domain.

Lemma 9 *Let G be a terrain of complexity n in $\mathbb{R}^3$, and let $\varepsilon > 0$. Then the complexity of G^ε is $O_\delta(n^2)$. And even if G is defined over a convex domain there is a lower bound example of complexity $\Omega(n^2)$.*

Proof: G consists of $O(n)$ triangles, segments, and points which do not intersect in their relative interior. In [6] it has been shown that the complexity and the computation time for the ε-neighborhood for such a collection of objects is $O_\delta(n^2)$. For the lower bound example we construct a terrain over a rectangular domain which consists of $n/2$ rectangles and $n/2$ pyramidal spikes. See Figure 4.1 for an illustration. Let γ be a small constant, and let $h \geq \varepsilon$. The rectangles are normal to $\vec{e}_3$ at height h, are adjacent to each other, have width $2\varepsilon/(n+1)$ in $\vec{e}_1$-direction and have length $(\varepsilon+\gamma)n/2$. They build the upper half of a convex polyhedron. The spikes are located right next to the rectangles at distance $\varepsilon/(n+1)$, one after the other at spacing $2\varepsilon + \gamma$ in $\vec{e}_2$-direction, i.e., in the direction of the long side of the rectangles. The spikes have $\vec{e}_3$-height $h + \varepsilon$. The holes between the spikes and rectangles are filled with the necessary surface patches to build a convex domain. The spikes and the rectangles are located in such a way, that the ε-neighborhood of every spike intersects the ε-neighborhood of every rectangle in the overall ε-neighborhood G^ε for a total of $\Omega(n^2)$ intersections. This construction does not depend on the exact shape of the unit ball, and therefore works for the L_2-metric as well as for other convex metrics. □

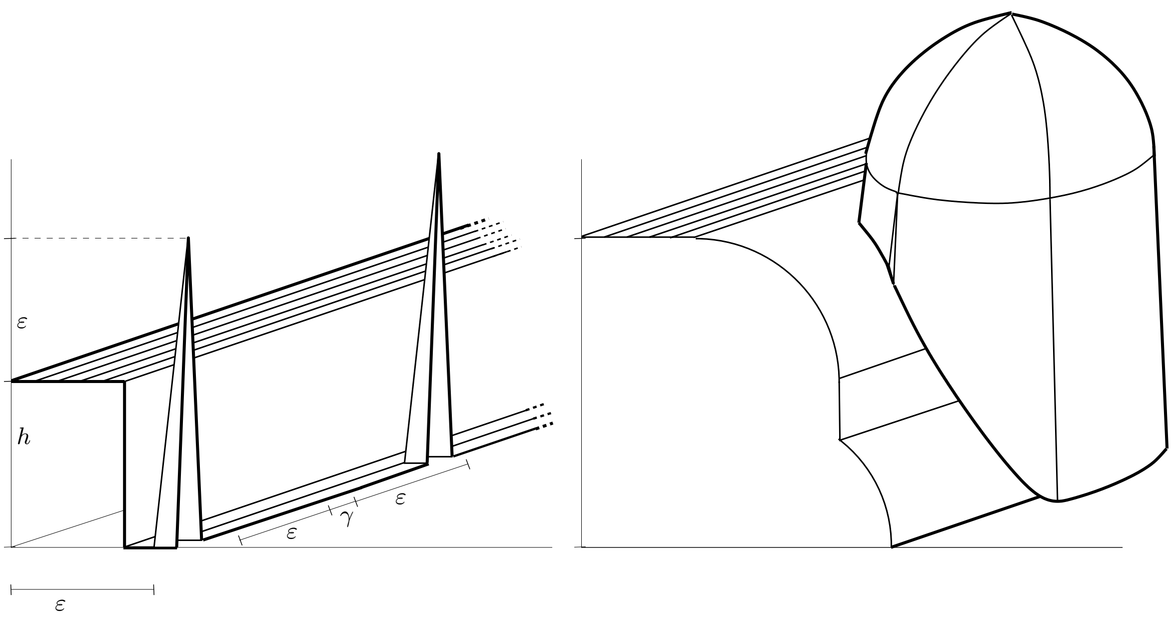


Figure 4.1: Example of a terrain in $\mathbb{R}^3$ defined over a convex domain, whose ε-neighborhood has quadratic complexity. Left: The terrain. Right: Part of the ε-neighborhood of the terrain.

The Structure of I for ε-Terrains

In the remainder of this subsubsection we assume that G is an ε-terrain for a given parameter $\varepsilon > 0$. From Lemma 7 we know that for an ε-terrain G the boundary of G^ε can be split into two separate terrains G_u and G_l over the domain D_g^ε. Thus, over the domain D_g^ε, $\overline{G^\varepsilon}$ can be split into two disjoint parts, $\overline{G^\varepsilon} = U \dot\cup L$, where U is the "upper" and L the "lower" part, with respect to $\vec{e}_d$. Both U and L are unbounded in the $\vec{e}_d$-direction and bounded by G_u from below and by G_l from above, respectively.

It suffices to consider $\overline{G^\varepsilon}$ only over the domain D^ε_g. Thus we can substitute $\overline{G^\varepsilon} = U \dot\cup L$ into (4.5) and obtain directly

$$I = \overline{U \oplus (-F)} \cap \overline{L \oplus (-F)} \, . \tag{4.6}$$

Since G_u and G_l are both terrains we know that

$$\begin{aligned} U \oplus (-F) = H_1 \quad &:= \quad \text{lower envelope of} \quad G_u \oplus (-F) \\ \text{and} \quad L \oplus (-F) = H_2 \quad &:= \quad \text{upper envelope of} \quad G_l \oplus (-F) \end{aligned}$$

Let $H_1\!\downarrow$ be the (d-dimensional) region below H_1 and similarly $H_2\!\uparrow$ the region above H_2 (where the above/below direction is with respect to the $\vec{e}_d$-direction). Plugging this into (4.6) yields

$$I = H_1\!\downarrow \cap H_2\!\uparrow \; . \tag{4.7}$$

I is often referred to as *the region sandwiched* between the two envelopes H_1 and H_2. Let $\gamma_d(n)$ denote the combinatorial complexity of G^u and G_l. We can construct I by performing the following steps (implementation details are given in the next section):

- Compute G_u (and G_l) as the upper (resp., lower) envelope of the ε-neighborhoods of all simplices of G.

- Compute the envelopes H_1 and H_2. G_u and G_l are composed of $\gamma(n)$ surface patches whose shape depend on the underlying metric. We compute H_1 and H_2 by computing for each such patch and each $a \in F$ the individual Minkowski sums of which there are $O(m\gamma(n))$ many, and finally computing the envelope of these Minkowski sums.

- Compute the sandwich region $H_1\!\downarrow \cap H_2\!\uparrow$.

Computing I for ε-Terrains

Let us now consider how fast the algorithm for computing I for ε-terrains can be implemented for various metrics in various dimensions.

In $d = 2$ dimensions G consists of n pairwise interior-disjoint line segments and points. For any convex metric, the ε-neighborhoods of single segments or points are pseudo-disks, i.e., the boundaries of two ε-neighborhoods intersect at most twice. In [55] it has been shown that the union of n pseudo-disks has complexity $O(n)$, hence $\gamma_2(n) = O(n)$. Since G is an ε-terrain we consider the Minkowski sum of each segment in G with the unit disk $\mathbf{B}^2_\varepsilon$, and then obtain G_u as the upper envelope and G_l as the lower envelope of all these Minkowski sums. Since we know that their complexity is $O(n)$, the envelopes can be computed in $\gamma'_2(n) = O(n \log n)$ time, [67].

For the Euclidean metric, G_l and G_u consist of line segments and circular arcs, two of which can cross each other at most two times. Now we need to compute the Minkowski sum of each such line segment or circular arc with each $a \in F$. The case of a line segment is easy because the Minkowski sum is simply the convex hull of the vector sums of each pair of endpoints, one from each line segment. For circular arcs the computation becomes a bit more

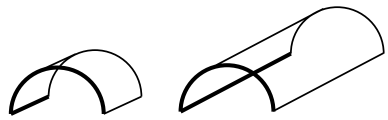


Figure 4.2: Two examples of the construction of the Minkowski sum of a line segment and a circular arc.

tricky, but can still be carried out in constant time by basically copying the circular arc at both ends of the line segment and adding a copy of the segment between the two points on the arcs at which the segment is tangent. See Figure 4.2 for an illustration of two cases that can arise. Therefore H_1 and H_2 also consist of line segments and circular arcs, have complexity $O(mn2^{\alpha(mn)})$, and can be computed in $O(mn2^{\alpha(mn)}\log(mn))$ time, see [67]. Since H_1 and H_2 are again x-monotone they can cross each other only $O(mn2^{\alpha(mn)})$ times, such that a linear-time sweep suffices to check if I is empty or not. In two dimensions for the L_2 metric we therefore have a runtime of $O(mn2^{\alpha(mn)}\log(mn)) = O(mn2^{\alpha(mn)}\log(mn))$. For $d = 2$ and projectable polyhedral metrics of constant description complexity, G_u and G_l are piecewise linear functions, i.e. they consist of $O(n)$ line segments each. In that case the envelopes H_1 and H_2 have a complexity of $\theta(mn\alpha(n))$ each and can be computed in $O(mn\alpha(n)\log(n))$ time, [67]. This yields a total runtime of $O(mn\alpha(n)\log(n))$.

Let us now turn to three dimensions, and let us consider projectable convex polyhedral metrics of constant description complexity. For $d = 3$ we have seen in Lemma 9 that G^ε and therefore G_u and G_l have complexity $O_\delta(n^2)$. But since G is an ε-terrain we can as well consider the Minkowski sum of each triangle, segment, or point in G with the polyhedron $\mathbf{B}^3_\varepsilon$, and then compute the upper envelope of all the obtained simplices (which yields G_u), and the lower envelope (which yields G_l). The Minkowski sum of two polyhedra is the convex hull of the vector sums of each pair of extreme points, one from each of the two polyhedra. Thus these Minkowski sums can be computed in constant time. The complexity of the envelope of $O(n)$ simplices is $O(n^2\alpha(n))$ and it can be computed in deterministic $O_\delta(n^2)$ time, for $\delta > 0$, or in randomized expected $O(n^2\alpha(n))$ time, [67]. Now we compute the Minkowski sums for each of the $O(n^2\alpha(n))$ surface patches (of constant description complexity) on G_u or G_l with each $a \in F$. Note that in the case of the L_2-metric G_u and G_l can also contain pieces of ε-spheres and ε-cylinders. For these the computation of the Minkowski sum with a triangle of F is not straight-forward. In principle a similar approach as the two-dimensional case should apply, however the handling of tangent planes and intersection curve segments seems tricky. We therefore consider only polyhedral metrics. After that we compute the region I between the two envelopes H_1 and H_2 in $O_\delta((mn^2\alpha(n))^2)$ time, see [3], for $\delta > 0$. The algorithm of [3] computes the overlay of the domains (so called *minimization diagrams*) of the envelopes H_1 and H_2 and provides a representation for the two envelopes above each cell. Employing this information we can check if $I = H_1{\downarrow} \cap H_2{\uparrow}$ is non-empty by simply traversing the overlay. Hence, altogether we need $O_\delta(m^2n^4)$ time.

For arbitrary $d > 3$ and projectable convex polyhedral metrics of constant description complexity one can try to apply the same technique. The complexity of G_u and G_l is $O(n^{d-1}\alpha(n))$, and they can be computed in deterministic $O_\delta(n^{d-1})$ time, or in randomized expected $O(n^{d-1}\alpha(n))$ time, see [67]. However for this case it is only known that the region between two envelopes has the same complexity as the envelopes themselves, therefore in our case $O((mn^{d-1}\alpha(n))^{d-1}\alpha(mn^{d-1}\alpha(n)))$, see [41]. Also note the recent result by Koltun and

Sharir [58, 59], who bound the complexity of the overlay of two minimization diagrams of two envelopes of arbitrary trivariate functions of constant algebraic description complexity. However, we are not aware of an algorithm that actually computes either the region sandwiched between two envelopes, or the overlay of the minimization diagrams for dimensions $d > 3$.[†] A brute force approach to construct I is to compute the d-dimensional arrangment of all $O(mn^{d-1}\alpha(n))$ Minkowski sums of a triangle on G_l or G_u with a triangle on F, and then traverse all d-cells of this arrangement in order to check if at least one cell is contained in I. This needs time $O((mn^{d-1}\alpha(n))^d) = O(m^d n^{d^2-d}\alpha^d(n))$.

Altogether we showed the following results:

Theorem 5 *Let F be a polyhedral set and G be a polyhedral ε-terrain in $\mathbb{R}^d$, with complexities m and n respectively. We can solve Problem 4, i.e., answer the decision problem for the directed Hausdorff distance under translations for the following cases:*

- *For $d = 2$ and any projectable convex polyhedral metrics of constant description complexity, we can solve Problem 4 in $O(mn\alpha(n)\log(n))$ time.*
- *For $d = 2$ and the Euclidean metric, we can solve Problem 4 in $O(mn2^{\alpha(mn)}\log(mn))$ time.*
- *For $d = 3$ and any projectable convex polyhedral metrics of constant description complexity we can solve Problem 4 in $O_\delta(m^2n^4)$ time.*
- *For $d \geq 4$ and any projectable convex polyhedral metrics of constant description complexity we can solve Problem 4 in $O(m^d n^{d^2-d}\alpha^d(n))$ time.*

If both F and G are ε-terrains, then we can apply the same approach as before. We split ∂F into F_u and F_l, and define U^F and L^F to be the region above F_u and below F_l. We call those regions for G now U^G and L^G. Then we have to compute $I^F \cap I^G$ where

$$\begin{aligned} I^F &:= \overline{U^F \oplus (-G)} \cap \overline{L^F \oplus (-G)} \\ I^G &:= \overline{U^G \oplus (-F)} \cap \overline{L^G \oplus (-F)}\,. \end{aligned}$$

This is equivalent to computing

$$H_1^G\!\downarrow \cap H_1^F\!\downarrow \cap H_2^G\!\uparrow \cap H_2^G\!\uparrow$$

where H_1^G is the lower envelope of $G_u \oplus (-F)$, H_2^G the upper envelope of $G_l \oplus (-F)$, H_1^F the lower envelope of $F_u \oplus (-G)$, H_2^F the upper envelope of $F_l \oplus (-G)$.

Since we have two lower envelopes and two upper envelopes we can simply compute $H_1^{F,G}$ as the lower envelope of $G_u \oplus (-F) \cup F_u \oplus (-G)$, and $H_2^{F,G}$ as the upper envelope of $G_l \oplus (-F) \cup F_l \oplus (-G)$, and know that the following holds:

$$H_1^{F,G}\!\downarrow \cap H_2^{F,G}\!\uparrow \;=\; H_1^G\!\downarrow \cap H_1^F\!\downarrow \cap H_2^G\!\uparrow \cap H_2^G\!\uparrow$$

Thus the same argumentation as above applies to this case, however with the time complexities symmetric in n and m:

[†]Note that in [59] Koltun and Sharir give a rough sketch how to compute the space of hyperplane transversals in four dimensions, which is a special case of the region sandwiched between two envelopes. If this technique applies to the general case, then we could apply it to our problems, too.

Corollary 5 *Let F and G be polyhedral ε-terrains in $\mathbb{R}^d$, with complexities m and m respectively. Let $N = m + n$. We can solve Problem 2, i.e., answer the decision problem for the undirected Hausdorff distance under translations for the following cases:*

- *For $d = 2$ and any projectable convex polyhedral metrics of constant description complexity, we can solve Problem 4 in $O(N^2\alpha(N^2)\log(N))$ time.*
- *For $d = 2$ and the Euclidean metric, we can solve Problem 4 in $O(N^2 2^{\alpha(N^2)}\log(N))$ time.*
- *For $d = 3$ and any projectable convex polyhedral metrics of constant description complexity we can solve Problem 4 in $O_\delta(N^6)$ time.*
- *For $d \geq 4$ and any projectable convex polyhedral metrics of constant description complexity we can solve Problem 4 in $O(N^{d^2}\alpha^d(N))$ time.*

Note that the restriction to ε-terrains is by Lemma 8 no restriction for terrains defined continuously over a convex domain.

We would now like to be able to apply an optimization scheme like parametric search to finally solve the optimization problems. However we are not aware of parallel versions of the envelope algorithms we used. But we can of course apply binary search on ε. Note that this can only be done if the terrains under consideration are ε-terrains for the range of ε to be searched. Recall that terrains defined over a convex domain are ε-terrains for any $\varepsilon > 0$.

4.2 Matching Convex Polyhedra

In this section let A and B be two bounded convex polyhedra in $\mathbb{R}^d$. Amenta has shown in [19] that the computation of the minimum Hausdorff distance under translations for two convex polyhedra can be expressed as a lexicographic convex program. However an actual implementation has only been given for two dimensions. For higher dimensions it has been left open how to construct these convex constraints such that the lexicographic minimum point for a constant number of constraints can be computed in constant time. For two dimensions it has been stated in [19] that this can indeed be done in constant time, although it is not completely clear how to construct the corresponding cones and compute their intersections, especially for the L_2-metric. Notice that Amenta [19] considers the slightly more general case of homotheties, which are combinations of translations and scalings.

We first describe how to solve the decision Problem 2: Given an $\varepsilon > 0$, does there exist a $t \in \mathbb{R}^d$ such that the Hausdorff distance $\delta_\mathrm{H}(A + t, B) \leq \varepsilon$? For this we consider first the question concerning the directed Hausdorff distance, if there exists a $t \in \mathbb{R}^d$ such that $\vec{\delta}_\mathrm{H}(A + t, B) \leq \varepsilon$. From (3.1) we know that this is equivalent to

$$t \in I(\varepsilon) := \overline{\overline{B^\varepsilon} \oplus (-A)}\ . \tag{4.8}$$

So our goal is to check if $I(\varepsilon) \neq \emptyset$, and if so to find one point in $I(\varepsilon)$. For this we will construct $I(\varepsilon)$. The following construction is based on polyhedral metrics of constant description complexity. Unfortunately it does not directly carry over to the case of the L_2 metric. After having solved the decision problem we will see that the construction easily generalizes to solving the optimization Problem 1.

Two common ways to efficiently represent polyhedra are the *V-representation* and the *H-representation*. In the V-representation a polyhedron is represented by a finite number of points in $\mathbb{R}^d$ whose convex hull forms the polyhedron. In the H-representation the polyhedron is represented by a finite number of half-spaces whose intersection forms the polyhedron. We assume that A and B are bounded convex polyhedra which are given in V-representation $A = CH\{a_1, ..., a_m\}$ and $B = CH\{b_1, ..., b_n\}$ with $a_1, ..., a_m, b_1, ..., b_n \in \mathbb{R}^d$. Note that we consider the solid polyhedra A and B and not only their boundaries.

First we construct the V-representation for $-A = CH\{-a_1, ..., -a_m\}$ by reflecting the points representing A at the origin. Now note that B^ε is, in the case of a convex polyhedral metric of constant description complexity, the convex hull of all points $a + p$ with $a \in \{a_1, ..., a_m\}$ and p an arbitrary point of the constant number of extreme points on the unit ball $\mathbf{B}^d_\varepsilon$. Thus B^ε is the convex hull of $O(n)$ points.

Lemma 10 *Let P and Q be two convex polyhedra in $\mathbb{R}^d$, $d \geq 2$. Then $\overline{\overline{P} \oplus Q}$ is also a convex polyhedron. If P is given in H-representation as the intersection of m half-spaces, then $\overline{\overline{P} \oplus Q}$ is the intersection of m half-spaces. If furthermore Q is given in V-representation with n vertices, then the H-representation for $\overline{\overline{P} \oplus Q}$ can be computed in $O(mn)$ randomized expected time. For $d = 2$, $\overline{\overline{P} \oplus Q}$ can be computed in $O(m + n)$ time.*

Proof: Let P be given in H-representation, $P = \bigcap_{i=1}^m h_i^+$, where each h_i is an oriented hyperplane in $\mathbb{R}^d$, h_i^+ the associated half-space in direction of the normal vector, and h_i^- the associated half-space in the opposite direction. And let Q be given in V-representation, $Q = CH\{q_1, \ldots, q_n\}$. Then

$$\overline{\overline{P} \oplus Q} = \overline{(\bigcup_{i=1}^{m} h_i^-) \oplus Q} = \overline{\bigcup_{i=1}^{m} (h_i^- \oplus CH\{q_1, \ldots, q_n\})}$$

Now $h_i^- \oplus CH\{q_1, \ldots, q_n\} = \bigcup_{j=1}^n (h_i^- + q_j) = h_i^- + q_i^*$, where q_i^* is a vertex in $\{q_1, \ldots, q_n\}$ with maximum signed distance[‡] to h_i. Thus

$$\overline{\overline{P} \oplus Q} = \overline{\bigcup_{i=1}^{m} (h_i^- + q_i^*)} = \bigcap_{i=1}^{m} (h_i^+ + q_i^*)$$

is convex. q_i^* can be simply computed for each h_i using an LP-solver, for example, in $O(n)$ randomized expected time[§] [66]; resulting in $O(mn)$ time altogether. For $d = 2$ we use a result of [49] which states that $\overline{\overline{P} \oplus Q}$ can be computed in $O(n + m)$ time. □

[‡]Note that the distance in translation space is the Euclidean distance.
[§]Note that d is assumed to be constant.

Let $d > 2$. Clearly $B^\varepsilon = CH\{\cup_{i=1}^n \mathbf{B}_\varepsilon^d(b_i)\}$ which is the convex hull of $O(n)$ points, assuming $\mathbf{B}_\varepsilon^d$ to be a polyhedron of constant description complexity. Thus we compute an H-representation for B^ε by applying a convex hull algorithm which runs in $O(n^{\lfloor (d+1)/2 \rfloor})$ time, see [33] for example. The complexity, i.e., the number of faces of all dimensions is $O(n^{\lfloor d/2 \rfloor})$. We take all $O(n^{\lfloor d/2 \rfloor})$ facets and extend them to their affine hulls. This yields an H-representation for B^ε. Now we can apply Lemma 10 to compute the H-representation for $I(\varepsilon)$, which consists of $O(n^{\lfloor d/2 \rfloor})$ half-spaces, in $O(mn^{\lfloor d/2 \rfloor})$ expected time. In order to obtain one point in $I(\varepsilon)$, or to check if $I(\varepsilon)$ is empty, we apply an LP solver which runs in time proportional to the number of defining half-spaces, thus in $O(n^{\lfloor d/2 \rfloor})$ expected time. For $d = 2$, B^ε can be computed in $O(n)$ time and $I(\varepsilon)$ can be computed in $O(m + n)$ time, see [49].

Theorem 6 *Let A and B be two convex polyhedra in $\mathbb{R}^d$ given in V-representation with m and n vertices, respectively. Let $\varepsilon > 0$. Then we can find a translation t with $\vec{\delta}_{\mathrm{H}}(A+t, B) \leq \varepsilon$, if there exists one, in $O(mn^{\lfloor d/2 \rfloor} + n^{\lfloor (d+1)/2 \rfloor})$ randomized expected time for $d > 2$, and in $O(m + n)$ time for $d = 2$.*

We can apply this result directly to the case of the undirected Hausdorff distance: For this we have to check if $I(\varepsilon) \cap I'(\varepsilon) \neq \emptyset$ with $I(\varepsilon) := \overline{\overline{B^\varepsilon} \oplus (-A)}$ and $I'(\varepsilon) := \overline{\overline{A^\varepsilon} \oplus (-B)}$. So we construct the H-representation for $I(\varepsilon)$, and symmetrically the H-representation for $I'(\varepsilon)$. This can be done in $O(mn^{\lfloor d/2 \rfloor} + n^{\lfloor (d+1)/2 \rfloor} + nm^{\lfloor d/2 \rfloor} + m^{\lfloor (d+1)/2 \rfloor})$ expected time, resulting in a collection of $O(n^{\lfloor d/2 \rfloor} + m^{\lfloor d/2 \rfloor})$ half-spaces, which we plug into an LP solver to obtain a $t \in I(\varepsilon) \cap I'(\varepsilon)$ if there exists one.

Corollary 6 *Let A and B be two convex polyhedra in $\mathbb{R}^d$ given in V-representation with m or n vertices, respectively. Let $N = n + m$ and let $\varepsilon > 0$. Then we can find a translation t with $d_H(A+t, B) \leq \varepsilon$, if there exists one, in $O(mn^{\lfloor d/2 \rfloor} + n^{\lfloor (d+1)/2 \rfloor} + nm^{\lfloor d/2 \rfloor} + m^{\lfloor (d+1)/2 \rfloor}) = O(N^{\lceil (d+1)/2 \rceil})$ randomized expected time for $d > 2$, and in $O(m + n)$ time for $d = 2$.*

In order to solve the optimization problem we add ε as another dimension to the translation/parameter space: Instead of considering only translations in $\mathbb{R}^d$ for a fixed $\varepsilon > 0$, we consider tuples $(t, \varepsilon) \in \mathbb{R}^{d+1}$ with a translation $t \in \mathbb{R}^d$. We construct $I = (I(\varepsilon), \varepsilon) \subseteq \mathbb{R}^{d+1}$ and search for the minimum ε such that $I(\varepsilon) \neq \emptyset$. The only change in the construction of I in this higher-dimensional space is that $(B^\varepsilon, \varepsilon)$ for varying ε is the Minkowski sum of B with the convex cone $C = (\mathbf{B}_\varepsilon^d, \varepsilon)$ in ε-direction with base $\mathbf{B}_\varepsilon^d$. Note that for $\varepsilon, \varepsilon' > 0$, $\mathbf{B}_\varepsilon^d$ is the scalar multiple $(\varepsilon/\varepsilon')\mathbf{B}_{\varepsilon'}^d = \{(\varepsilon/\varepsilon')x \mid x \in \mathbf{B}_{\varepsilon'}^d\}$ of $\mathbf{B}_{\varepsilon'}^d$. Thus $B^\varepsilon = B \oplus \mathbf{B}_\varepsilon^d = B \oplus ((\varepsilon/\varepsilon')\mathbf{B}_{\varepsilon'}^d)$. It is well-known that Minkowski sums with scalar multiples are combinatorially isomorphic for all positive scalars, see [51] for example. Thus we obtain an H-representation for C by extending every half-space of the H-representation for B^ε to a half-space in $\mathbb{R}^{d+1}$, by adding the appropriate direction vector. This direction vector can be obtained from an extreme boundary point on the facet of B^ε that is contained in the hyperplane bounding the half-space. Hence, C is the intersection of $O(n^{\lfloor d/2 \rfloor})$ half-spaces, and we can apply Lemma 10 in the same way¶, which yields that I is convex and can be computed within the same time bounds as $I(\varepsilon)$. Thus Theorem 6 and Corollary 6 carry over to the following Corollary:

¶The proof in Lemma 10 did not assume boundedness of the polyhedra.

Corollary 7 *Let A and B be two convex polyhedra in $\mathbb{R}^d$ given in V-representation with m or n vertices, respectively. Let $N = n + m$. Then we can solve Problem 3, i.e., find a translation t that minimizes $\vec{\delta}_{\mathrm{H}}(A + t, B)$ in $O(mn^{\lfloor d/2 \rfloor} + n^{\lfloor (d+1)/2 \rfloor})$ randomized expected time for $d > 2$, and in $O(m + n)$ time for $d = 2$.*

We can solve Problem 1, i.e., find a translation t that minimizes $\delta_{\mathrm{H}}(A + t, B)$, in $O(mn^{\lfloor d/2 \rfloor} + n^{\lfloor (d+1)/2 \rfloor} + nm^{\lfloor d/2 \rfloor} + m^{\lfloor (d+1)/2 \rfloor}) = O(N^{\lceil (d+1)/2 \rceil})$ randomized expected time for $d > 2$, and in $O(m + n)$ time for $d = 2$.

Note that for two dimensions the time bounds match the bounds given in [19].