Stability Analysis of Kinetic Orientation-Based Shape Descriptors

Wouter Meulemans
Department of Mathematics and Computer Science, TU Eindhoven, the Netherlands

Kevin Verbeek
Department of Mathematics and Computer Science, TU Eindhoven, the Netherlands

Jules Wulms
Department of Mathematics and Computer Science, TU Eindhoven, the Netherlands

Abstract

We study three *orientation-based* shape descriptors on a set of continuously moving points: the first principal component, the smallest oriented bounding box and the thinnest strip. Each of these shape descriptors essentially defines a cost capturing the quality of the descriptor and uses the orientation that minimizes the cost. This optimal orientation may be very unstable as the points are moving, which is undesirable in many practical scenarios. If we bound the speed with which the orientation of the descriptor may change, this may lower the quality of the resulting shape descriptor. In this paper we study the trade-off between stability and quality of these shape descriptors.

We first show that there is no *stateless algorithm*, an algorithm that keeps no state over time, that both approximates the minimum cost of a shape descriptor and achieves continuous motion for the shape descriptor. On the other hand, if we can use the previous state of the shape descriptor to compute the new state, we can define "chasing" algorithms that attempt to follow the optimal orientation with bounded speed. We show that, under mild conditions, chasing algorithms with sufficient bounded speed approximate the optimal cost at all times for oriented bounding boxes and strips. The analysis of such chasing algorithms is challenging and has received little attention in literature, hence we believe that our methods used in this analysis are of independent interest.

Keywords and phrases Stability analysis, Time-varying data, Shape descriptors, Approximation algorithms

Digital Object Identifier 10.57717/cgt.v5i1.51

Funding *Wouter Meulemans*: Partially supported by the Netherlands eScience Center (NLeSC); 027.015.G02.
Kevin Verbeek: Supported by the Dutch Research Council (NWO); 639.021.541.
Jules Wulms: Partially supported by the Netherlands eScience Center (NLeSC); 027.015.G02.

Acknowledgements The authors are grateful to Bettina Speckmann for insightful discussions leading up to this paper, and to Christiaan Dirkx, Pantea Haghighatkhah and Aleksandr Popov for preliminary results on the topological stability of oriented bounding boxes.

1 Introduction

Given the amount of data that is widely available nowadays, algorithms play an important part in analyzing this data and finding useful patterns in it. For many applications it is important that an algorithm is *stable*: small changes in the input lead to small changes in the output. These applications include the analysis or visualization of time-varying data. We can effectively visualize how a large set of moving points evolves, for example, by drawing a glyph capturing the direction of movement (e.g. an arrow or line segment) for a few subsets of the points, which gives a clear and comprehensible overview of the data. We can also summarize a set of moving points in a single dimension by projecting all the points to their first principal component, as used in the visualization technique MotionRugs [31]. However,

if the orientation of the glyphs or first principal component changes erratically even with small changes in the data, it becomes hard to see how the data changes over time. In other words, unstable algorithms may result in large visual changes even for small data changes, severely limiting the efficacy of such methods for visualization of time-varying data. Thus, it is important to develop stable algorithms which deal with these discrete changes of the input in an elegant way. As a result, stable algorithms can efficiently work with the available data without losing their effectiveness over time.

The applications mentioned above all use some type of shape descriptor. Shape descriptors are simplified representations of more complex shapes. They are used, for example, as summaries of a large collection of data, where we are not interested in all the details, but simply want to have an overview of the most important features. Shape descriptors play an important role in many fields that perform shape analysis, such as computer vision (shape recognition) [7, 34], computer graphics (bounding boxes for broad-phase collision detection) [20, 29], medical imaging (diagnosis or surgical planning) [10, 35], and machine learning (shape classification) [32, 33]. When computing such shape descriptors, we generally want to optimize how closely the descriptor resembles the input. In this paper we are interested in shape descriptors that specifically capture the orientation of the data – in our case a set of moving points. For example, one could consider a short line segment with small distance to the point set, a line segment connecting the focal points of an ellipse enclosing the points, or any other descriptor that captures the direction in which the point set is stretched.

The three shape descriptors we study in this paper are the *first principal component* (PC), the smallest (area) *oriented bounding box* (OBB) and the thinnest *covering strip* (STRIP). As we will see below, these three shape descriptors are *orientation-based*, that is, the optimization function for such a shape descriptor, in an orientation α on a point set P, is easily expressed (only) in terms of α and P. The orientation α_{OPT} of an optimal descriptor captures the orientation of the described point set P. These three shape descriptors are also commonly used in applications (such as the orientation glyphs or MotionRugs) where we always want to output a reasonable orientation, even when this orientation is not very pronounced. These three shape descriptors are unfortunately also unstable: small changes in the point set they represent can result in discrete "flips" in their orientation (see Figure 1). We analyze the stability of these shape descriptors and develop stable variations on them.

Problem description. The main goal of this paper is to formally analyze the trade-off between quality and stability for orientation-based shape descriptors. Our input consists of a set of n moving points $P = P(t) = \{p_1(t), \ldots, p_n(t)\}$ in two dimensions, where each $p_i(t)$ is a continuous function $p_i \colon [0, T] \to \mathbb{R}^2$. We assume that, at each time t, not all the points are at the same position. The output consists of an orientation $\alpha = \alpha(t)$ of the shape descriptor for

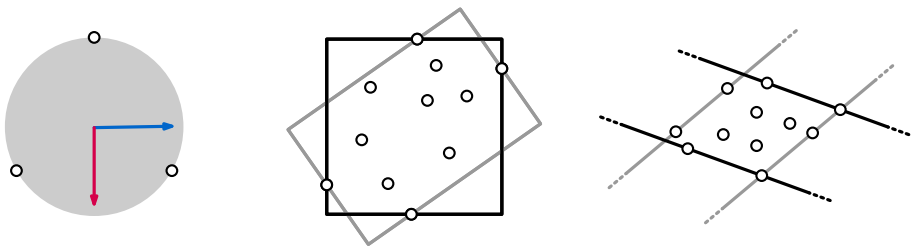

Figure 1 A flip in orientation for PC, OBB and STRIP. Small changes in the positions of the points make one of the orientations optimal over the other.

every point in time $t \in [0, T]$, which need not match the optimal orientation due to stability constraints. To quantify the quality of any output orientation, we define each of the three shape descriptors Π as minimizers of an optimization function $f_\Pi(\alpha, P)$. Let $\mathcal{L}(\alpha)$ be the set of lines with orientation α, and let $d(L, p)$ be the distance between a point p and a line L. Furthermore, let the *extent* of P along a unit vector α be $w_\alpha(P) = \max_{p,q \in P}(p - q) \cdot \alpha$, where $\cdot$ denotes a dot product, and let $\alpha^\perp$ be a vector orthogonal to α. We define the orientation of PC, OBB or STRIP as the orientation α that minimizes the following functions:

Principal component: $f_{\text{PC}}(\alpha, P) = \min_{L \in \mathcal{L}(\alpha)} \sum_{p \in P} d(L, p)^2$
Oriented bounding box: $f_{\text{OBB}}(\alpha, P) = w_\alpha(P) w_{\alpha^\perp}(P)$
Covering strip: $f_{\text{STRIP}}(\alpha, P) = w_{\alpha^\perp}(P)$

Intuitively, PC can be represented by a line that minimizes the sum of squared errors from the points in P to that line – this line is then the direction of largest variation. OBB and STRIP intuitively capture a bounding rectangle of minimum area and a bounding strip of minimum width. Though various other options for these functions are possible, we believe the functions above naturally fit to the shape descriptors, and most importantly these functions quantify the quality of a shape descriptor for any orientation α. This allows us to also consider shape descriptors of suboptimal quality, which do not fully minimize the optimization function. This in turn enables us to make a trade-off between quality and stability. When computing a shape descriptor, we typically compute more than just an orientation, such as the exact position and dimensions of the descriptor. However, the stability is mostly affected by the optimal orientation: if the (not necessarily optimal) orientation changes continuously and the points move continuously, then the shape descriptor changes continuously as well. We therefore ignore other aspects of the shape descriptors to analyze their stability, and assume that these aspects are chosen optimally for the given orientation without any cost with regard to the stability.

Note that an orientation $\alpha(t)$ is an element of the real projective line $\mathbb{RP}^1$, but we typically represent $\alpha(t)$ by a unit vector in $\mathbb{R}^2$ and implicitly identify opposite vectors, which is equivalent. Furthermore, we assume that the output $\alpha(t)$ is computed for all real values $t \in [0, T]$. This assumption simplifies our analysis. In practice, algorithms can be executed only finitely often, once per some defined time step. Introducing such discreteness into the problem may lead to interesting effects, but it is not the focus of this paper. Furthermore, for sufficiently small time steps, the continuous analysis will provide a good approximation.

Kinetic algorithms. Algorithms for kinetic (moving) input can adhere to different models, which may influence the results of the stability analysis. Let $\mathcal{A}$ be an algorithm mapping input to output, and $I(t)$ the input depending on time t. We distinguish the following models.

Stateless algorithms: The output depends only on the input $I(t)$ at a particular point in time, and no other information of earlier times. This in particular means that if $I(t_1) = I(t_2)$, then $\mathcal{A}$ produces the same output at time t_1 and at time t_2.

State-aware algorithms: The algorithm $\mathcal{A}$ has access not only to the input $I(t)$ at a particular time, but also maintains a state S (possibly the entire history) over time; in practice this is typically the output at the previous point in time. Thus, even if $I(t_1) = I(t_2)$, $\mathcal{A}$ may produce different results at time t_1 and t_2 if the states at those times are different.

Clairvoyant algorithms: The algorithm $\mathcal{A}$ has access to the complete function $I(t)$ and can adapt to future inputs. Thus, the complete output over time can be computed offline.

In this paper we consider both stateless and state-aware algorithms. A stateless algorithm can be stable only if it defines a mapping from input to output that is naturally continuous. On the other hand, a state-aware algorithm can easily enforce continuity by using its state to keep track of earlier output. We call such a state-aware algorithm a *chasing algorithm*:

Chasing algorithms: The algorithm $\mathcal{A}$ maintains the most recent output (in our case a shape descriptor) as the state S and uses only the input at the current time $I(t)$ to find a new optimal solution OPT. The algorithm then moves the solution in S towards OPT at the maximum allowed speed (which depends on the required stability). The resulting solution is the output for the current time t and is subsequently stored in S.

The output at time t might not coincide with OPT, as the solution in S may have been too different from OPT and the maximum allowed speed did not allow S to catch up to OPT: the output of our algorithm chases after the optimal solution. The challenge lies in bounding the ratio between the quality of the solutions used by the algorithm and the (unstable) optimal quality. Since we do not consider discrete time steps in this paper, we simply assume that a chasing algorithm maintains a solution over time, and it can choose the rate of change of the solution (e.g., rotation speed/direction) at every point in time.

Stability analysis. Classical considerations to assess algorithms include quality (e.g., optimality, approximation ratio) and efficiency (in e.g. time or memory). These aspects consider the relation between input and output; a worst-case analysis can typically consider a single input. Stability instead considers how outputs relate between related inputs, and a worst-case scenario is a sequence of inputs. This requires a different analysis approach: we use the framework by Meulemans et al. [22]. Below, we restrict their definitions to our setting.

Let $\mathcal{A}$ be an algorithm that takes a point set as input and computes an orientation (where the state S for state-aware algorithms is implicit). For *topological stability*, we aim to analyze how well such an algorithm can perform with respect to the (unstable) optimum, if we restrict its output to move continuously but potentially arbitrarily fast for continuously moving input. The notion of continuity is captured by choosing an appropriate topology for the input space and the output space. This is straightforward: the input space is $\mathbb{R}^{2n}$ and the output space is topologically equivalent to the unit circle which we denote here by O. We choose these topologies for our analysis. The *topological stability ratio* of shape descriptor Π is then defined as

$$\rho_{TS}(\Pi) = \inf_{\mathcal{A}} \sup_{P} \max_{t\in[0,T]} \frac{f_\Pi(P(t), \mathcal{A}(P(t)))}{\min_{\gamma\in O} f_\Pi(P(t), \gamma)}$$

where the supremum is taken over all continuously moving point sets P, and the infimum is taken over all algorithms $\mathcal{A}$ for which $\mathcal{A}(P(t))$ is continuous. Furthermore, in the denominator we minimize over all orientations γ to find an optimal shape descriptor Π (PC, OBB, or STRIP).

The *K-Lipschitz stability ratio* $\rho_{LS}(\Pi, K)$ is defined almost exactly the same: the only difference is that we now take the infimum over all algorithms $\mathcal{A}$ for which $\mathcal{A}(P(t))$ is K-Lipschitz, that is, the output moves continuously with speed bounded by K. We measure the output in radians, and hence the output orientation or angle may change with a speed of at most K radians per time unit. Lipschitz stability requires bounded input speed and scale invariance [22]. Here, we assume that points move with at most unit speed and that the diameter of $P(t)$ is at least 1 at all times. Note that we can always appropriately scale to meet these assumptions, if not all the points coincide in a single location. The unit diameter assumption is only mild, if points represent actual objects such as fish [31] which have a certain size and cannot occupy the same physical space.

Related work. Shape descriptors are a wide topic, studied in various subfields of computer science. Here, we focus on results related to PC, OBB or STRIP, and results on stability.

The three oriented shape descriptors considered in this paper are related to each other, yet slightly different. The ratio between the volume of the bounding box using the orientation of PC and the minimal-volume bounding box is unbounded [13]. Similar to PC, other fitness measures have been considered with respect to oriented lines, such as the sum of distances or vertical distances [11]. Computing OBB for static point sets is a classic problem in computational geometry. In two dimensions, one side of the optimal box aligns with a side of the convex hull and it can be computed in linear time after finding the convex hull [17, 27]; a similar property holds in three dimensions, allowing a cubic-time algorithm [24]. The relevance of bounding boxes in 3D as a component of other algorithms also led to efficient approximation algorithms [5]. Bounding boxes find applications in tree structures for spatial indexing [6, 19, 25, 26] and in speeding up collision detection and ray-tracing techniques [4, 29, 18]. The optimal STRIP can be computed using the same techniques as OBB in two dimensions [17, 27]. Agarwal et al. [2] provide an $O(n + 1/\varepsilon^{O(1)})$-time approximation algorithm via ε-kernels for various measures of a point set, including OBB and STRIP. In terms of kinetic maintenance of these shape descriptors, OBB and STRIP can both be maintained by efficient and compact kinetic data structures [1]. These data structures process $O(n^{2+\varepsilon})$ events under algebraic motion of the points, while both shape descriptors may undergo $\Omega(n^2)$ combinatorial changes even for linear motion of the points. However, a $(1+\varepsilon)$-approximation of OBB and a $(2+\varepsilon)$-approximation of STRIP can be maintained while processing a number of events that depends only on ε [3].

The stability of kinetic 1-center problems has been studied by Bereg et al. [8]. They provide some results on the trade-off between solution quality and speed using chasing algorithms. Their analysis is significantly different from ours, as the structures they chase inherently move continuously: coping with discrete flips is then not necessary. Subsequently, Durocher and Kirkpatrick [16] studied this trade-off for the 1-median problem, which does exhibit discrete flips (even when the 1-median is not restricted to input positions). Durocher and Kirkpatrick [14, 15] also studied the stability of the kinetic 2-center problem; their approach allows a trade-off between solution quality and the speed at which a solution changes. De Berg et al. [12] show similar results in the black-box kinetic-data-structures model. Letscher et al. [21] show that the medial axis for a union of disks changes continuously under certain conditions or if it is pruned appropriately. Meulemans et al. [22] introduced a framework for analyzing stability and apply it to Euclidean minimum spanning trees on a set of moving points. They show bounded topological stability ratios for various topology definitions on the space of spanning trees, and that the Lipschitz stability ratio is at most linear, but also at least linear if the allowed speed for the changes in the tree is too low. Van der Hoog et al. [30] study the topological stability of the k-center problem, showing upper and lower bounds on the stability ratio for various measures. They also provide a clairvoyant algorithm to determine the best ratio attainable for a given set of moving points.

Results and organization. In Section 2 we prove that there exists no stateless algorithm for any of the shape descriptors that is both topologically stable and achieves a bounded approximation ratio for the quality of the optimal shape descriptor. We then consider state-aware algorithms and analyze the topological stability for each of the shape descriptors in Section 3. In Section 4 we analyze the Lipschitz stability of chasing algorithms for all three shape descriptors. We show that chasing algorithms with sufficient speed can achieve a constant approximation ratio for OBB and STRIP, if we indeed assume unit speed for

the points and that the diameter of the point set is always at least 1. To the best of our knowledge, this is the first time a chasing algorithm has been analyzed, which deals with discrete changes. We believe that the new methods that we developed for this challenging stability analysis are of independent interest, and may be applied to the analysis of other chasing algorithms. We conclude the paper in Section 6.

2 Stateless algorithms

We prove that stateless algorithms cannot achieve bounded topological stability ratio for any of the three shape descriptors. This readily implies an unbounded K-Lipschitz stability ratio for any K. The argument is entirely topological. A stateless topologically stable algorithm (with output behaving continuously) is a continuous map from the input space to the output space. Important for the theorem below is that, if all points of set P lie on a single line with orientation α, then $f_\Pi(\gamma, P) = 0$ iff $\gamma = \alpha$, for each considered shape descriptor Π. Intuitively the proof works as follows: We construct a set of inputs that is topologically equivalent to a disk in the input space. All inputs P at the boundary of this disk consist of n points that lie on a single line with orientation α, and hence the ratio between $f_\Pi(\alpha, P)$ and $f_\Pi(\gamma, P)$, with $\gamma \neq \alpha$, will be unbounded. We show that there is no function that is both continuous, and maps the inputs at the boundary of the topological disk to the correct orientations.

▶ **Theorem 1.** *For stateless algorithms $\rho_{TS}(\text{PC}) = \rho_{TS}(\text{OBB}) = \rho_{TS}(\text{STRIP}) = \infty$ if the point set contains at least three points.*

Proof. The idea is to construct a continuous map $j\colon D^2 \to \mathbb{R}^{2n}$ on the two-dimensional closed disk D^2, such that the image of j is a vector consisting of coordinates of valid point sets in $\mathbb{R}^2$ (which do not have all the points at the same position), and that the image of ∂D^2 under j forces the orientation of the shape descriptor on those points. We parameterize D^2 using polar coordinates (r, ϕ) for $0 \leq \phi < 2\pi$ and $0 \leq r \leq 1$. We first construct a map j' as follows, and we interpret each element in its codomain as a sequence of x- and y-coordinate pairs corresponding to n points in $\mathbb{R}^2$ (see Figure 2).

$$j'(r,\phi) = \left(\frac{r}{n}\cos\phi, \frac{r}{n}\sin\phi, \quad \frac{2r}{n}\cos\phi, \frac{2r}{n}\sin\phi, \quad \ldots, \quad \frac{rn}{n}\cos\phi, \frac{rn}{n}\sin\phi\right)$$

When $r \neq 0$, the point set corresponding to j' is spread out over a line, whose orientation depends on ϕ (see Figure 2b). The orientation of the shape descriptor is then always forced (otherwise the approximation ratio is ∞). However, $j'(0, \phi)$ is not a valid point set, since it places all the points at the origin. Now let P^* be a set of n points in $\mathbb{R}^2$, containing at least three non-collinear points. Interpreting P^* as a vector in $\mathbb{R}^{2n}$, we define $j(r, \phi) = j'(r, \phi) + (1 - r)P^*$. By the choice of P^*, $j(r, \phi)$ is always a valid point set. Furthermore, the orientation of the shape descriptors is still fixed for point sets $j(1, \phi) = j'(1, \phi)$, namely $\alpha = \phi \pmod{\pi}$; in that orientation the three optimization functions will evaluate to zero and in any other orientation they evaluate to a non-zero value. As a result, any stateless algorithm with an approximation ratio $\rho < \infty$ defines, along with j, a continuous mapping h from D^2 to circle O (as the quotient space $[0, \pi]/\{0, \pi\}$ for PC and STRIP, and $[0, \frac{\pi}{2}]/\{0, \frac{\pi}{2}\}$ for OBB) where ∂D^2 is mapped multiple times around O, i.e., a double cover of O for PC and STRIP, and a quadruple cover for OBB. The continuous mapping h is simply the output of the stateless algorithm on j. We claim that the mapping h from D^2 to O cannot exist.

For the sake of contradiction, assume that such a map h exists. Let $f, g : O \to D^2$ be continuous functions. Function f maps every point $x \in O$ to the boundary $\partial D^2 \subseteq D^2$ such

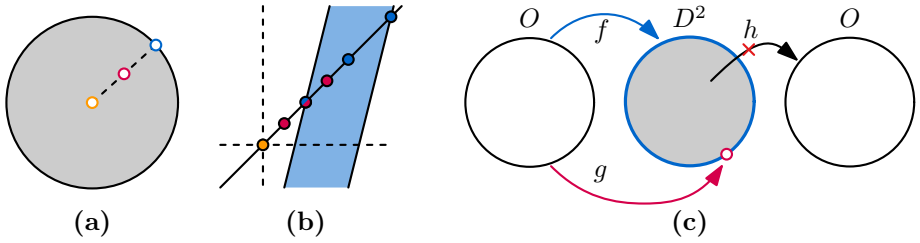

Figure 2 Illustrations for the proof of Theorem 1: **(a)** The closed disk D^2 and elements $j'(0, \pi/4)$, $j'(1/2, \pi/4)$, and $j'(1, \pi/4)$ as colored points in D^2, and **(b)** the corresponding color-coded point sets in $\mathbb{R}^2$. Each point set consists of $n = 3$ points; some points coincide, for example the yellow points. For $j'(1, \pi/4)$ we show the width-0 STRIP in orientation $\pi/2$ as well as a (wider) strip in another orientation. **(c)** The homotopic mappings f and g, and the mapping h that cannot exist.

that the mapping covers the whole boundary once, while g maps all $x \in O$ to a single point $y \in D^2$ (see Figure 2c). We can continuously shrink the image of f to a single point in D^2, in particular the image of g; hence f and g are homotopic. We now consider $h \circ f$ and $h \circ g$ and use the degree of these mappings (as first defined in [9]) to show that h cannot exist. Since f maps O to the boundary $\partial D^2 \subseteq D^2$, and h maps ∂D^2 to a double or quadruple cover of O, we know that the degree of $h \circ f$ is respectively two or four. On the other hand, g maps all of O to a single point in D^2, therefore $h \circ g$ has degree 0. By the Hopf theorem [23] $h \circ f$ and $h \circ g$ cannot be homotopic, as they can be homotopic if and only if their degrees are equal. However, $h \circ f$ and $h \circ g$ must be homotopic, since homotopy equivalence is compatible with function composition and f is homotopic to g. This contradiction implies that h cannot exist. Thus ρ_{TS} is ∞ for stateless algorithms. ◀

3 Topological stability

In this section we turn to state-aware algorithms, and we analyze the topological stability of the shape descriptors. Specifically, we prove the following tight bounds.

▶ **Theorem 2.** *The topological stability ratios of the shape descriptors are:*
- $\rho_{\text{TS}}(PC) = 1$,
- $\rho_{\text{TS}}(OBB) = \frac{5}{4}$,
- $\rho_{\text{TS}}(STRIP) = \sqrt{2}$.

To prove an upper bound on ρ_{TS}, we need an algorithm that produces an output that changes continuously, but may change arbitrarily fast. Our algorithm works as follows. Whenever an optimal solution undergoes a discrete change at time t, our algorithm moves continuously between the solution before and after the discrete change in the interval $T = [t - \epsilon, t + \epsilon]$. When analyzing the topological stability ratio, we consider all suboptimal intermediate solutions in T. However, in our proofs we may assume that this change happens at time t, since the definition of ρ_{TS} asks for an infimum over all algorithms, which may use different ϵ: The limit of T when ϵ approaches zero shrinks to a single point in time. It is therefore sufficient to consider changing the output at time t to find an upper bound on ρ_{TS}, since our output is suboptimal only during this change. On the other hand, to prove a lower bound on ρ_{TS}, we must construct a full time-varying point set such that the corresponding approximation ratio must occur at some point in time during this motion, regardless of which algorithm is used. Note that a lower bound on the topological stability ratio is immediately also a lower bound on the K-Lipschitz stability ratio for any value of K.

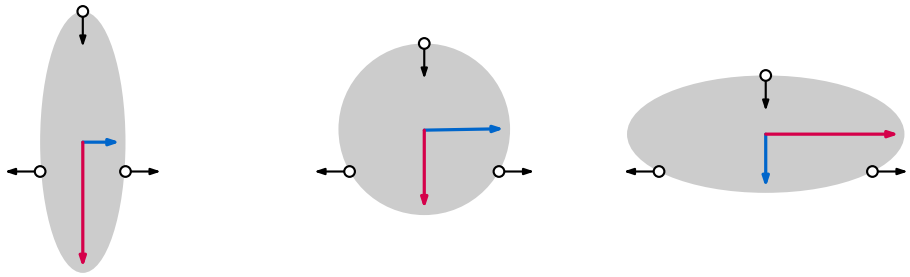

Figure 3 Moving points causing a flip between the first principal component (red) and second principal component (blue).

▶ **Lemma 3.** $\rho_{\mathrm{TS}}(PC) = 1$

Proof. Consider a time t where the first principal component flips between two orientations, represented by unit vectors $\vec{v}_1$ and $\vec{v}_2$ (see Figure 3). The first principal component is the orientation of the line that minimizes the sum of squared distances between the points and the line, as described in Section 1. It can be computed by centering the point set at the mean of the coordinates, computing the covariance matrix of the resulting point coordinates, and extracting the eigenvector of this matrix with the largest eigenvalue. Since eigenvalues change continuously if the data changes continuously [28, Theorem 3.9.1], both $\vec{v}_1$ and $\vec{v}_2$ must have some eigenvalue λ^* at time t. But that means that every interpolated vector $\vec{v} = (1-u)\vec{v}_1+u\vec{v}_2$ also has eigenvalue λ^*, since $C\vec{v} = (1-u)C\vec{v}_1 + uC\vec{v}_2 = (1-u)\lambda^*\vec{v}_1 + u\lambda^*\vec{v}_2 = \lambda^*\vec{v}$. As a result, $f_{\mathrm{PC}}(\vec{v}) = f_{\mathrm{PC}}(\vec{v_1})$, and we can continuously change orientation from $\vec{v}_1$ to $\vec{v}_2$ without decreasing the quality of the shape descriptor. ◀

▶ **Lemma 4.** $\rho_{\mathrm{TS}}(OBB) \leq \frac{5}{4}$

Proof. Consider a time t at which two distinct oriented bounding boxes A and B have minimum area; both are assumed to have the smallest area 1 without loss of generality, as the problem at hand is invariant under scaling. At this time t we continuously change the orientation of the box between that of A and B while making sure that the box still contains all the points (see Figure 4). The goal is to compute the maximal size of the intermediate box in the worst case. Note that we may rotate either clockwise or counterclockwise; we always choose the direction that minimizes this maximal intermediate size.

Let a and b denote the length of the major axes of A and B respectively. Let angle α denote the smallest angle between the orientations of the major axes. Note that $\alpha \in \{0, \pi/2\}$ leads to A and B being identical, and that our problem is invariant under rotation, reflection and translation. We thus assume without loss of generality:

- $b \geq a \geq 1$;
- $0 < \alpha < \pi/2$;
- B is centered at the origin, and A at (dx, dy);
- the major axis of A is horizontal;
- α describes a counterclockwise angle from the major axis of A to the major axis of B.

The points P must all be contained in the intersection of A and B, for otherwise A and B would not be bounding boxes. Furthermore, no side of A may be completely outside B or vice versa, for otherwise one of the boxes could be made smaller. Thus, all sides intersect, and we are interested in four of these intersections $I_1, \ldots, I_4$ (see Figure 4). Specifically,

we want to use the intersections that allow us to derive a valid upper bound on the size of the bounding box during rotation. Since the intersections depend on the direction of rotation, we choose the intersections that allow us to rotate from A to B in counterclockwise direction. The coordinates of the intersections can easily be computed (see Figure 4). For example, for $I_1 = (x_1, y_1)$ we solve for the following two equations: $x_1 = dx - a/2$ and $x_1 \cos\alpha + y_1 \sin\alpha = -b/2$. The resulting coordinates of all intersections are:

- $I_1 = (-\frac{a}{2} + dx, -\frac{b-(a-2dx)\cos\alpha}{2\sin\alpha})$
- $I_2 = (\frac{a}{2} + dx, \frac{b-(a+2dx)\cos\alpha}{2\sin\alpha})$
- $I_3 = ((-\frac{1}{2a} + dy)\frac{\cos\alpha}{\sin\alpha} + \frac{1}{2b\sin\alpha}, -\frac{1}{2a} + dy)$
- $I_4 = ((\frac{1}{2a} + dy)\frac{\cos\alpha}{\sin\alpha} - \frac{1}{2b\sin\alpha}, \frac{1}{2a} + dy)$

Now consider an intermediate box C with angle $\theta \leq \alpha$ with respect to box A. Note that C contains the intersection of A and B as long as it contains $I_1, \ldots, I_4$. We can define two vectors $V_1 = I_2 - I_1$ and $V_2 = I_4 - I_3$, which we project to lines at angle θ and $\theta + \frac{\pi}{2}$ to obtain the lengths of the sides of C. Note that V_1 and V_2 depend only on a, b and α, but not on dx and dy. Thus, using V_1 and V_2 we can obtain a formula for the area of C, which we call $\mathcal{C}$.

$$\begin{aligned}\mathcal{C}(a,b,\alpha,\theta) &= V_1 \cdot (\cos\theta, \sin\theta) \times V_2 \cdot (-\sin\theta, \cos\theta) \\ &= \frac{(b\sin(\alpha-\theta) + a\sin\theta) \times (a\sin(\alpha-\theta) + b\sin\theta)}{ab\sin^2\alpha}\end{aligned}$$

We are now interested in the maximum of $\mathcal{C}$. We start by finding the partial derivative of $\mathcal{C}$ with respect to θ:

$$\frac{\partial\mathcal{C}}{\partial\theta} = \frac{(a^2+b^2-2ab\cos\alpha)\sin(\alpha-2\theta)}{ab\sin^2\alpha}$$

First observe that $\frac{\partial\mathcal{C}}{\partial\theta} = 0$ if and only if $\theta = \alpha/2$ in the chosen domain, which implies that we can set $\theta = \alpha/2$. Using double-angle formulas, we may simplify $\mathcal{C}$ in this setting.

$$\begin{aligned}\mathcal{C}(a,b,\alpha,\alpha/2) &= \frac{(a+b)^2\sin^2(\alpha/2)}{ab\sin^2\alpha} \\ &= \frac{(a+b)^2\sin^2(\alpha/2)}{ab(2\sin(\alpha/2)\cos(\alpha/2))^2} \\ &= \frac{(a+b)^2}{2ab(2\cos^2(\alpha/2))} \\ &= \frac{(a+b)^2}{2ab(1+\cos(\alpha))}\end{aligned}$$

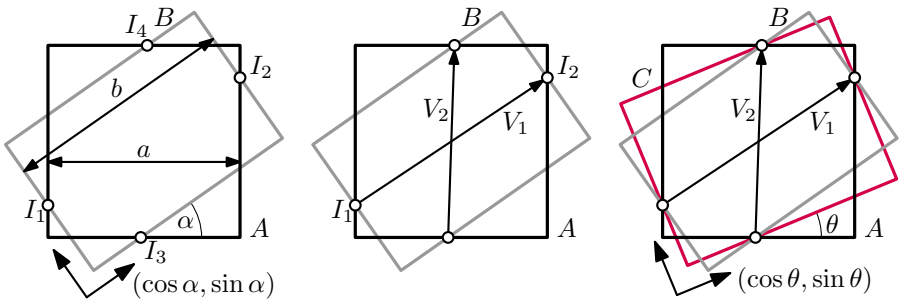

Figure 4 Construction of closed formula for area of intermediate solution C.

We now split the domain of α into $(0, \frac{\pi}{4}]$ and $(\frac{\pi}{4}, \frac{\pi}{2})$, and prove both cases separately.

In the first case, when $\alpha \in (0, \frac{\pi}{4}]$, let $c \geq 1$ be such that $b = ca$ (since $b \geq a$). As b is at most the length of the diagonal of A, we get that $c \leq \sqrt{a^2 + (1/a)^2}/a = \sqrt{1 + 1/a^4} \leq \sqrt{2}$ (since $a \geq 1$). The resulting formula is $\mathcal{C}(a, ca, \alpha, \alpha/2) = \frac{(1+c)^2}{2c(1+\cos\alpha)}$. This function is maximized when c and α are maximized. We thus set $\alpha = \pi/4$ and $c = \sqrt{2}$ to obtain that $\mathcal{C}(1, \sqrt{2}, \pi/4, \pi/8) \leq \frac{1}{2} + \frac{1}{2}\sqrt{2} < \frac{5}{4}$ and thus this case meets the bound claimed.

What remains is to prove the case where $\alpha \in (\frac{\pi}{4}, \frac{\pi}{2})$. It might now be beneficial to rotate A clockwise instead of counterclockwise, to align the minor axis of A with the major axis of B: this clockwise rotation may result in smaller intermediate solutions C. Since $\mathcal{C}$ can only deal with counterclockwise rotation, we have to use different parameters to deal with the described situation. To simulate the clockwise rotation, we use $\mathcal{C}(1/a, b, \pi/2 - \alpha, \theta)$; this reflects the whole setup over direction $\pi/4$ effectively considering the minor axis as the major axis instead. Note that we did not use the assumption that $a \geq 1$ anywhere above, until within the other case where $\alpha \leq \pi/4$. Note that $\frac{\partial \mathcal{C}}{\partial \theta} = 0$ depends on the parameters we fill in, hence we can set $\theta = (\pi/2 - \alpha)/2 = \pi/4 - \alpha/2$ to find a maximum in this case.

For all possible values of a, b and α, we need to find the area of the largest intermediate box C. Since we can choose whether we rotate clockwise or counterclockwise, this area is the minimum of $\mathcal{C}(a, b, \alpha, \alpha/2)$ and $\mathcal{C}(1/a, b, \pi/2 - \alpha, \pi/4 - \alpha/2)$. We first simplify the latter.

$$\begin{aligned}
\mathcal{C}(1/a, b, \pi/2 - \alpha, \pi/4 - \alpha/2) &= \frac{(\frac{1}{a} + b)^2}{2\frac{1}{a}b(1 + \cos(\pi/2 - \alpha))} \\
&= \frac{a^2(\frac{1}{a} + b)^2}{2ab(1 + \sin(\alpha))} \\
&= \frac{(1 + ab)^2}{2ab(1 + \sin(\alpha))}
\end{aligned}$$

To find the maximum of the function, we can use a mathematical program that looks for the values of a, b and α that comply to a set of constraints and maximize a target function. All constraints come from the assumptions, but we add a final constraint similar to what we did in the $\alpha \in (0, \frac{\pi}{4}]$ case. To ensure that all 4 intersection points exist, the projection of the diagonal of A to the major axis of B should be larger than b. Hence we add the constraint $b \leq a\cos\alpha + \frac{1}{a}\sin\alpha$. The mathematical program now looks as follows:

$$\begin{aligned}
\textbf{maximize} \quad & \min\left(\frac{(a+b)^2}{2ab(1+\cos(\alpha))}, \frac{(1+ab)^2}{2ab(1+\sin(\alpha))}\right) \\
\textbf{subject to} \quad & b \leq a\cos\alpha + \frac{1}{a}\sin\alpha \\
& \pi/4 < \alpha < \pi/2 \\
& 1 \leq a \leq b
\end{aligned}$$

Using mathematical software, for example Mathematica 11.2, we can verify that the area is at most $\frac{5}{4}$. ◀

▶ **Lemma 5.** $\rho_{\mathrm{TS}}(\mathit{OBB}) \geq \frac{5}{4}$

Proof. Refer to Figure 5 for illustration. Consider a point set P with four static points $p_1 = (0, 0)$, $p_2 = (2, 1)$, $p_3 = (0.75, 1)$ and $p_4 = (1.25, 0)$, and point p_5 moving linearly

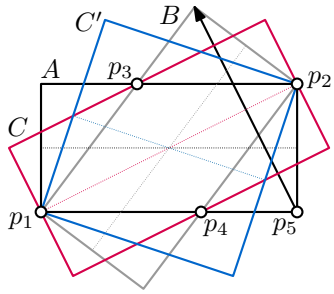

Figure 5 Moving points forcing a flip from bounding box A to B. Rotating continuously, a bounding box at least as big as C or C' is required.

from $(2, 0)$ to $(1.2, 1.6)$. The static points allow two minimal bounding boxes of area 2 and aspect ratio 2: one with orientation 0 (box A) and one with orientation $2\arctan(\frac{1}{2}) \approx 53.13$ degrees (box B). As p_5 is always in A or B, one of these boxes is always optimal. Initially, only A contains p_5 and in the end only box B does. The angles $\arctan(\frac{1}{2})$ (box C) and $\pi/4 - \arctan(\frac{1}{2})$ (box C') give an intermediate box of size $2.5 = \frac{5}{4} \cdot 2$ on the static points. Box C is encountered on a counterclockwise rotation from A to B, and C' on a clockwise rotation. Neither C, C', nor any box rotated more towards B contains the initial location of p_5. Similarly, neither C, C', nor any box rotated towards A contains the final location.

To derive a contradiction, assume a continuously moving OBB exists that achieves a ratio strictly less that $\frac{5}{4}$. This ratio implies that initially the OBB orientation is clockwise between C and C', and at the end of the motion it is not. However, as the assumed OBB moves continuously through O, it must at some point have been in the orientation of C or C'. But this implies a ratio of $\frac{5}{4}$, contradicting our assumption and proving the lower bound. ◀

▶ **Lemma 6.** $\rho_{\text{TS}}(\textit{STRIP}) \leq \sqrt{2}$

Proof. Consider a time t at which there are two thinnest strips A and B of width 1 with different orientations. All the points must be contained in the diamond-shaped intersection D of A and B (see Figure 6). If we continuously rotate a strip C from A to B, then at some point the width of C must be at least the length of one of the diagonals of D. To maximize the length of the shortest diagonal, D must be a square with side length 1. Therefore, the width of C is at most $\sqrt{2}$ during the rotation from A to B. ◀

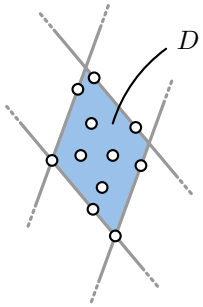

Figure 6 A configuration having two strips of minimal width and overlapping area D.

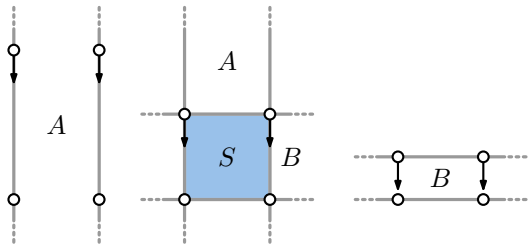

Figure 7 An instance of moving points where the thinnest strip changes orientations. The configuration that leads to the best intermediate solutions is shown in the middle.

▶ **Lemma 7.** $\rho_{\text{TS}}(\textit{STRIP}) \geq \sqrt{2}$

Proof. Let P consist of four points positioned in a unit square S. There are two thinnest strips A and B for P, each of which is oriented along a different pair of parallel edges of S (see Figure 7). If the orientation of a strip is $\pi/4$ away from A and B, then its width is $\sqrt{2}$.

Now assume that the top points of S are moving along the vertical sides of S, starting from high above. Clearly, at the start of this motion, any strip C approximating the thinnest strip must align with A. At the end of this motion, when the top points align with the bottom points of S, the strip C must align with B. Therefore, the strip C must at some point make an angle of $\pi/4$ with A and B. If x is the distance between the top and bottom points of S, then the width of C at this orientation is $(1+x)\sqrt{2}/2$. The minimal width cannot exceed the width of A and B, and hence is at most $\min(x, 1)$. When $x \geq 1$, the ratio between the width of C at orientation $\pi/4$ and the minimal width is $(1+x)\sqrt{2}/2 \geq \sqrt{2}$. This ratio is $(1+x)\sqrt{2}/2x$ when $x < 1$, for which we easily see that $\sqrt{2}/2x + x\sqrt{2}/2x \geq \sqrt{2}/2 + \sqrt{2}/2 = \sqrt{2}$. ◀

4 Lipschitz stability of OBB and STRIP

To derive meaningful bounds on the Lipschitz stability ratio, we assume that the points move with at most unit speed. Furthermore, to achieve Lipschitz stability, the relation between distances and speeds in input and output space should be *scale invariant* [22], that is, inputs and outputs should be affected equally by scaling. This is currently not the case: if we scale the coordinates of the points, then the distances in the input space change accordingly, but the distances in the output space (between orientations) do not. For example, if we scale the input coordinates down, such that the points reside in a smaller area, then the points have to move a shorter distance than before scaling, to achieve the same change in optimum orientation for either of the considered shape descriptors. This means that smaller instances are less (Lipschitz) stable. To remedy this problem, we require that the diameter D of $P(t)$ is at least 1 at every time t. For PC this is not sufficient to prove a bounded Lipschitz stability ratio, as we argue in Section 5. Notice that scaling up the coordinates has the opposite effect and increases (Lipschitz) stability; we need no additional assumptions for such instances.

To produce a K-Lipschitz stable solution we use a chasing algorithm similar to the generic algorithm introduced in Section 1. The algorithm maintains a solution over time, and it can rotate towards the optimal solution at every point in time. However, there is one main difference with the generic algorithm: Instead of chasing the orientation of an optimal solution of OBB or STRIP, we chase the orientation of a diametrical pair. A *diametrical pair* of a point set P is a pair of points (p, q) whose distance $\|p - q\|$ equals the diameter of P. As there can be multiple diametrical pairs of P, we take the following measures to ensure that the diametrical pair we are chasing is well-defined. For our chasing algorithm we start by chasing the orientation of an arbitrary diametrical pair (p, q) until it is no longer a diametrical pair of P. When this happens, we find a new pair (p', q') of points that form a diametrical pair of P, and start chasing the orientation of this pair, again until it is no longer diametrical. This procedure is repeated to prevent unnecessarily switching of the diametrical pair we are chasing. However, we want to point out that dealing with discrete changes is the focal point of our analysis, and hence choosing an arbitrary diametrical pair at any point in time, and chasing towards this orientation would not invalidate our analysis.

Although chasing an optimal shape descriptor would be a more intuitive approach, chasing a diametrical pair is easier to analyze and sufficient to obtain an upper bound on the K-Lipschitz stability ratio for OBB and STRIP. Furthermore, note that, since we assume that the points move with unit speed, a chasing algorithm is K-Lipschitz stable if the solution changes

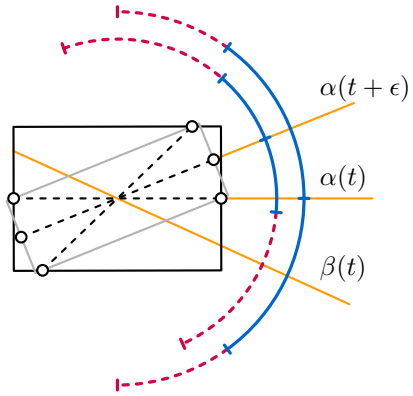

Figure 8 Intervals at time t (outer) and time $t+\epsilon$ (inner). The safe and danger zones are indicated in blue and dashed red respectively. Orientations are shown in yellow, for the diametric box at time t and $t+\epsilon$, and for the chasing algorithm at time t.

by at most K radians during such a unit movement of the points. Hence, in the following we ignore the original definition of K-Lipschitz stable algorithms, and simply impose a maximum speed of K on the solutions.

4.1 Chasing a diametrical pair

We denote the orientation of the diametrical pair we are chasing as $\alpha = \alpha(t)$ and the diameter of P as $D = D(t) \geq 1$. Furthermore, let $W = W(t)$ be the width of the thinnest strip with orientation $\alpha(t)$ covering all points in $P(t)$. The *diametric box* of $P(t)$ is the box containing $P(t)$ with orientation $\alpha(t)$, width $W(t)$ and length $D(t)$. We denote the (inverse) *aspect ratio* of the diametric box by $z = z(t) = W(t)/D(t)$. Finally, our chasing algorithm has orientation $\beta = \beta(t)$ and can change orientation with at most constant speed K. We generally omit the dependence on t if t is clear from the context.

Approach. The main goal is to keep chasing the orientation β as close as possible to the orientation α of an optimal diameter box, specifically within a sufficiently small interval around α. The challenge lies with the discrete flips of α. We must argue that, although flips happen instantaneously, a short time span does not admit many flips over a large angle in the same direction; otherwise we can never keep β close to α with a bounded speed. Furthermore, the size of the interval must depend on the aspect ratio z, since if $z = 0$, the interval around α must have zero size as well to guarantee a bounded approximation ratio.

For the analysis we introduce three functions depending on z: $T(z)$, $H(z)$, and $J(z)$. Function $H(z)$ defines an interval $[\alpha - H(z), \alpha + H(z)]$ called the *safe zone*. We aim to show that, if β leaves the safe zone at some time t, it must return to the safe zone within the time interval $(t, t + T(z)]$. We also define a larger interval $I = [\alpha - H(z) - J(z), \alpha + H(z) + J(z)]$. We refer to the parts of I outside of the safe zone as the *danger zone*. Figure 8 shows I at time t and time $t + \epsilon$. Although β may momentarily end up in the danger zone due to discontinuous changes, it must quickly find its way back to the safe zone. We aim to guarantee that β stays within I at all times. Let $E = E(t)$ refer to an endpoint of I. We call $J(z)$ the *jumping distance* and require that $J(z)$ upper bounds how far E can "jump" instantaneously. Let $\Delta E(z, \Delta t)$ denote how far E moves over a time period of length Δt, starting with a diameter box of aspect ratio z. We then require that $\Delta E(z, 0) \leq J(z)$. Note

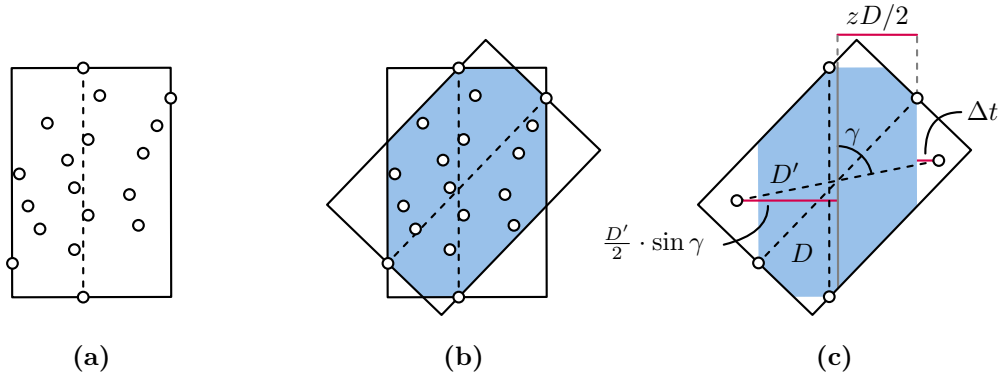

Figure 9 Illustrations supporting the proof of Lemma 8: **(a)** A diametric box for the vertical diametrical pair, with dimensions D, zD containing all the points. **(b)** If another pair of points is also at distance D from each other, all points must lie in the blue area. **(c)** The orientation of other diametrical pairs can change further in the same direction, but only after some points move outside of the blue area and establish a new diameter. The grey line bisects the width of the blue area.

that by defining this upper bound, $J(z)$ is defined recursively through E, since by definition $\Delta E(z, \Delta t)$ is upper bounded by how much α, $H(z)$ and $J(z)$ change over time Δt. Hence we need to carefully choose the right function for $J(z)$. For the other functions we choose $T(z) = z/4$ and $H(z) = c \arcsin(z)$ for a constant c, which we will choose later.

The purpose of c as a variable is that it will appear in the bounds for the speed and approximation ratio of the solution. Choosing a value for c then settles both the stability and solution quality, and ideally leads to a trade-off between these criteria. In our case, however, both are optimized to a reasonable level when choosing c.

Changes in orientation and aspect ratio. To verify that the chosen functions $T(z)$ and $H(z)$ satisfy the intended requirements, and to define the function $J(z)$, we need to bound how much α and z can change over a time period of length Δt. We refer to these bounds as $\Delta\alpha(z, \Delta t)$ and $\Delta z(z, \Delta t)$, respectively. Note that, since the diameter can change discontinuously, we generally have that $\Delta\alpha(z, 0) > 0$ and $\Delta z(z, 0) > 0$.

▶ **Lemma 8.** *$\Delta\alpha(z, \Delta t) \leq \arcsin(z + \Delta t(2 + 2z))$ for $\Delta t \leq (1 - z)/(2 + 2z)$.*

Proof. Refer to Figure 9 for illustrations. Let D be the diameter at time t; the width of the strip containing all the points is zD. Also, let D' be the diameter at time $t + \Delta t$, and let (p'_1, p'_2) be the diametrical pair at that time, such that the diametrical orientation differs by an angle γ from the orientation at time t. Note that $\Delta t \geq |D - D'|/2$, since p'_1 and p'_2 have at most unit speed, and both were inside D at time t. This means that $\Delta t \geq \frac{D'}{2}\sin(\gamma) - \frac{zD}{2}$. As Δt is minimized when $D \geq D'$, we can obtain a lower bound for Δt by equalizing $(D - D')/2 = \frac{D'}{2}\sin(\gamma) - \frac{zD}{2}$. We obtain that $\Delta t \geq \frac{D'(\sin(\gamma) - z)}{2+2z} \geq \frac{\sin(\gamma) - z}{2+2z}$. This is equivalent to $\gamma \leq \arcsin(z + \Delta t(2 + 2z))$, which is well-defined only for $\Delta t \leq \frac{1-z}{2+2z}$. ◀

▶ **Lemma 9.** *$\Delta z(z, \Delta t) \leq z - \frac{\sin(\frac{1}{2}\arcsin(z)) - 2\Delta t}{1+2\Delta t}$ for $\Delta t \leq \sin(\frac{1}{2}\arcsin(z))/2$.*

Proof. Let diameter D at time t be realized by the pair of points (p_1, p_2) with orientation α. The width of the diametric box is determined by two points q_1 and q_2; the distance between q_1 and q_2 is at most D (see Figure 10a).

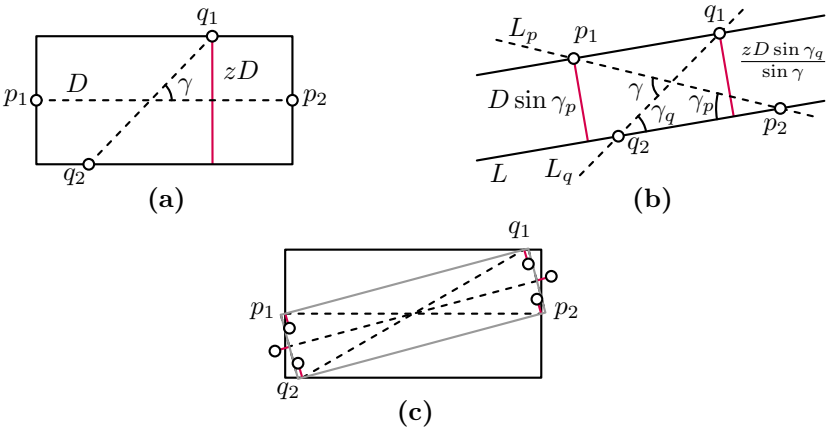

Figure 10 Illustrations for the proof of Lemma 9: **(a)** A diametrical box with aspect ratio z and points p_1, p_2, q_1 and q_2 at the boundary. **(b)** A strip with points located at the boundary, the width of the strip is the maximum of the red lines. **(c)** The smallest diametrical box for time $t + \epsilon$ and in red the distance the points can travel in Δt to further shrink the aspect ratio.

To minimize the aspect ratio of the diametric box at time $t + \Delta t$, we need to find a thinnest strip that contains all of p_1, p_2, q_1, and q_2. All four points are on the boundary of the strip in the worst case, and we assume w.l.o.g. that p_1 and q_1 are on the same side of the strip (same for p_2 and q_2). Consider the following lines: L oriented in the orientation of the thinnest strip (parallel to its boundary), L_p spanned by p_1p_2 and L_q spanned by q_1q_2. Let the angle between L_p and L_q be γ, $\gamma \geq \arcsin(z)$ (see Figure 10b). The distance between q_1 and q_2 is then $zD/\sin(\gamma)$. We denote the angle between L and L_p by γ_p, and between L and L_q by γ_q. We observe that $\gamma_p + \gamma_q = \gamma$, as the orientation of L must bisect the angle γ for the strip to be thinnest.

The width of the strip is $\max(D\sin(\gamma_p), zD\sin(\gamma_q)/\sin(\gamma))$. We show that this width is at least $D\sin(\frac{1}{2}\arcsin(z))$. This is clearly the case if $\gamma_p \geq \frac{1}{2}\arcsin(z)$, so assume the contrary. Since the function $\sin(\gamma_q)/\sin(\gamma) = \sin(\gamma - \gamma_p)/\sin(\gamma)$ is increasing, setting $\gamma = \arcsin(z)$ is optimal. But then $zD\sin(\gamma_q)/\sin(\gamma) = D\sin(\gamma_q) > D\sin(\frac{1}{2}\arcsin(z))$. Thus, the width of the thinnest strip is at least $D\sin(\frac{1}{2}\arcsin(z))$.

The diametric box at time t can flip to the orientation of this thinnest strip. However, the points can move to further shrink the aspect ratio of the diametric box. As a result, the width of the diametric box at time $t + \Delta t$ is at least $D\sin(\frac{1}{2}\arcsin(z)) - 2\Delta t$ (see Figure 10c). For the same reason, diameter D' can increase such that $D' \leq D + 2\Delta t$. The final aspect ratio is then $z' \geq (D\sin(\frac{1}{2}\arcsin(z)) - 2\Delta t)/(D + 2\Delta t)$. Since $D \geq 1$, we obtain that $\Delta z(z, \Delta t) \leq z - (\sin(\frac{1}{2}\arcsin(z)) - 2\Delta t)/(1 + 2\Delta t)$. Note that this bound is meaningful only for $\Delta t \leq \sin(\frac{1}{2}\arcsin(z))/2$. ◀

Jumping distance. We now derive a valid function for $J(z)$. Recall that we require $J(z)$ to be at least the amount $\Delta E(z, \Delta t)$ that E can move in $\Delta t = 0$ time.

▶ **Lemma 10.** *$J(z) = (c + 2)\arcsin(z)$ is a valid jumping distance function.*

Proof. By definition of $\delta E(z, \Delta t)$, we get that $\Delta E(z, 0) \leq \Delta\alpha(z, 0) + H(z) - H(z - \Delta z(z, 0)) + J(z) - J(z - \Delta z(z, 0))$. Lemmata 8 and 9 tell us that $\Delta\alpha(z, 0) \leq \arcsin(z)$ and $\Delta z(z, 0) \leq z - \sin(\frac{1}{2}\arcsin(z))$, respectively, so after simplification we get $\Delta E(z, 0) \leq (1 + c/2)\arcsin(z) +$

$J(z) - J(\sin(\frac{1}{2}\arcsin(z)))$. Since we require that $J(z) \geq \Delta E(z, 0)$, it suffices to show that $J(\sin(\frac{1}{2}\arcsin(z))) \geq (1 + c/2)\arcsin(z)$. Using $J(z) = (c + 2)\arcsin(z)$, we get that $J(\sin(\frac{1}{2}\arcsin(z))) = (c + 2)\arcsin(z)/2$, as required, and hence this $J(z)$ is valid. ◀

▶ **Corollary 11.** *If β is in I, then $|\alpha - \beta| \leq (2c + 2)\arcsin(z)$.*

Bounding the speed. To show that the orientation β stays within the interval I, we argue that over a time period of $T(z)$ we can rotate β at least as far as E. As the endpoint of the safe zone moves at most as fast as E, this implies that if β leaves the safe zone at time t, it returns to it in the time period $(t, t + T(z)]$. Thus we require that $KT(z) \geq \Delta E(z, T(z))$, as β can rotate at most K units when the points move at unit speed. We need to keep up only when the safe zone does not span all orientations, that is, the above inequality must hold only when $H(z) \leq \pi/2$, which can be rewritten into $z \leq \sin(\frac{\pi}{2c})$. At this point in the analysis, we choose a specific value $c = 3$ to reduce the complexity of the calculations for Lemma 14. This results in fixed size of the interval: we need to chase α only when $z \leq \sin(\frac{\pi}{6}) = \frac{1}{2}$. Additionally it leads to a lower bound for K, the speed at which the orientation is allowed to change with respect to the change in input, thus settling the stability.

In our proofs we use the following trigonometric inequalities.

▶ **Lemma 12.** *The following inequalities hold for $0 \leq x \leq 1$:*

1. $\sin(\lambda \arcsin(x)) \leq \lambda x$ *for* $\lambda \geq 1$
2. $\sin(\lambda \arcsin(x)) \geq \lambda x$ *for* $0 < \lambda \leq 1$.

Proof. We first show inequality (1). Let $x = \sin(y)$. We rewrite (1) into $\sin(\lambda y) \leq \lambda \sin(y)$. The derivative with respect of y is $\lambda \cos(\lambda y)$ for the left side and $\lambda \cos(y)$ for the right side. Since $\cos(y) \geq \cos(\lambda y)$ for $0 \leq y \leq \pi/\lambda$ and $\lambda \geq 1$, we get that $\sin(\lambda y) \leq \lambda \sin(y)$ for $0 \leq y \leq \pi/\lambda$. In particular, for $y = \pi/(2\lambda)$ we get that $1 = \sin(\lambda y) \leq \lambda \sin(y)$. Since $\sin(\lambda y) \leq 1$ and $\lambda \sin(y)$ attains it first maximum at $y = \pi/2$, we thus also get that $\sin(\lambda y) \leq \lambda \sin(y)$ for $0 \leq y \leq \pi/2$. Since $x = \sin(y)$ and $\sin(\pi/2) = 1$, the result follows.

For inequality (2), we set $x = \sin(\frac{1}{\lambda}\arcsin(y))$. We can then obtain that either $y \geq \lambda \sin(\frac{1}{\lambda}\arcsin(y))$, or $\frac{1}{\lambda} \geq \sin(\frac{1}{\lambda}\arcsin(y))$. As shown above, this inequality holds for $0 \leq y \leq 1$. Since $y = \sin(\lambda\pi/2) < 1$ implies $x = \sin(\frac{1}{\lambda}\arcsin(y)) = 1$, the inequality holds for $0 \leq x \leq 1$. ◀

▶ **Lemma 13.** $x \leq \arcsin(x) \leq \frac{\arcsin(a)}{a}x$ *for* $0 \leq a \leq 1$ *and* $0 \leq x \leq a$.

Proof. First note that $\arcsin(x)$ is a convex function for $0 \leq x \leq 1$. Since the derivative of x and $\arcsin(x)$ is 1 at $x = 0$, this directly implies that $x \leq \arcsin(x)$. Furthermore, since $\arcsin(x) = \frac{\arcsin(a)}{a}x$ for $x = 0$ and $x = a$, the convexity of $\arcsin(x)$ also directly implies the second inequality. ◀

▶ **Lemma 14.** *If $K \geq 43$, then $|\beta(t) - \alpha(t)| \leq 8\arcsin(z)$ (using $c = 3$) for all t.*

Proof. Consider a time t when $\beta(t)$ leaves the safe zone. We first argue that $\beta(t')$ will be in the safe zone at some time $t' \in (t, t + T(z)]$. To show this, we need to prove that $KT(z) \geq \Delta E(z, T(z))$. As shown earlier, this is relevant only when $z \leq \frac{1}{2}$ if $c = 3$.

To apply the bounds of Lemmata 8 and 9, we must ensure that $T(z) = z/4$ satisfies the bounds for Δt. For Lemma 8, observe that $(1 - z)/(2 + 2z)$ is decreasing and $z/4$ is increasing, and $(1 - z)/(2 + 2z) = \frac{1}{6} \geq z/4$, $z = \frac{1}{2}$. For Lemma 9 we apply Lemma 12 to

show that $\sin(\frac{1}{2}\arcsin(z))/2 \geq z/4$. We thus get the provided bounds on $\Delta\alpha(z, T(z))$ and $\Delta z(z, T(z))$, and as a result a bound on $\Delta E(z, T(z))$. In particular, for $\Delta t \leq T(z)$, we get:

$$\begin{aligned}\Delta E(z,\Delta t) &\leq \Delta\alpha(z,\Delta t) + H(z) - H(z - \Delta z(z,\Delta t)) + J(z) - J(z - \Delta z(z,\Delta t))\\ &= \arcsin(z + \Delta t(2+2z)) + 8\arcsin(z)\\ &\qquad - 8\arcsin\left(\frac{\sin(\frac{1}{2}\arcsin(z)) - 2\Delta t}{1+2\Delta t}\right).\end{aligned}$$

We have that $z \leq \frac{1}{2}$, $\Delta t \leq z/4 \leq \frac{1}{8}$ and $z + \Delta t(2+2z) \leq \frac{7}{8}$. Then, using the inequalities of Lemma 13, where $2\arcsin(\frac{1}{2}) \leq 1.05$ and $\frac{8}{7}\arcsin(\frac{7}{8}) \leq 1.22$, and using Lemma 12, we get:

$$\begin{aligned}\Delta E(z,\Delta t) &\leq 1.22(z + \Delta t(2+2z)) + 8.4z - \frac{4z - 16\Delta t}{1+2\Delta t}\\ &\leq 9.62z + 1.22\Delta t(2+2z),\end{aligned}$$

where the last inequality uses the fact that $\Delta t \leq z/4$, and thus $16\Delta t \leq 4z$. Finally, filling in $\Delta t = T(z) = z/4$, we get:

$$\begin{aligned}\Delta E(z, T(z)) &\leq 10.23z + 0.61z^2\\ &\leq 10.6z && \text{for } z \leq \frac{1}{2}\\ &\leq K\frac{z}{4} && \text{for } K \geq 43.\end{aligned}$$

Finally, we need to argue that $\beta(t)$ does not leave I in the interval $(t, t+T(z)]$. To show this, we need to prove that $K\Delta t \geq \Delta E(z,\Delta t) - J(z)$ for all $\Delta t \in [0, T(z)]$. Using the inequalities above, we have:

$$\begin{aligned}\Delta E(z,\Delta t) - J(z) &\leq \arcsin(z + \Delta t(2+2z)) + 3\arcsin(z)\\ &\qquad - 8\arcsin\left(\frac{\sin(\frac{1}{2}\arcsin(z)) - 2\Delta t}{1+2\Delta t}\right).\end{aligned}$$

We first argue that this function is nondecreasing in z, such that $\Delta E(z,\Delta t) - J(z) \leq \Delta E(\frac{1}{2},\Delta t) - J(\frac{1}{2})$. For that we consider its partial derivative in z:

$$\begin{aligned}\frac{\partial(\Delta E(z,\Delta t) - J(z))}{\partial z} &= \frac{3}{\sqrt{1-z^2}} + \frac{1+2\Delta t}{\sqrt{1-(2\Delta t + z + 2\Delta t z)^2}} -\\ &\qquad \frac{4\cos(\frac{1}{2}\arcsin(z))}{\sqrt{1-z^2}\sqrt{(1+2\Delta t)^2 - (\sin(\frac{1}{2}\arcsin(z)) - 2\Delta t)^2}}\\ &\geq \frac{3}{\sqrt{1-z^2}} + \frac{1+2\Delta t}{\sqrt{1-z^2}} -\\ &\qquad \frac{4}{\sqrt{1-z^2}}\frac{\cos(\frac{1}{2}\arcsin(z))}{\sqrt{1-(\sin(\frac{1}{2}\arcsin(z)) - 2\Delta t)^2}}\\ &\geq \frac{4}{\sqrt{1-z^2}}\left(1 - \frac{\cos(\frac{1}{2}\arcsin(z))}{\sqrt{1-\sin^2(\frac{1}{2}\arcsin(z))}}\right) \geq 0\end{aligned}$$

As a result we can conclude the following:

$$\begin{aligned}\Delta E(z,\Delta t) - J(z) &\leq \Delta E\left(\frac{1}{2},\Delta t\right) - J\left(\frac{1}{2}\right)\\ &= \frac{\pi}{2} + \arcsin\left(\frac{1}{2} + 3\Delta t\right) - 8\arcsin\left(1 - \frac{4+\sqrt{6}-\sqrt{2}}{4+8\Delta t}\right)\end{aligned}$$

Note that this bound is 0 whenever $\Delta t = 0$. It is now sufficient to show that the derivative of this function with respect to Δt is at most K for $0 \leq \Delta t \leq \frac{1}{8}$. Let $a = 4 + \sqrt{6} - \sqrt{2} \approx 5.035$.

$$\begin{aligned}
\frac{\partial \Delta E(\frac{1}{2}, \Delta t)}{\partial \Delta t} &= \frac{6}{\sqrt{4 - (1 + 3\Delta t)^2}} + \frac{64a}{(4 + 8\Delta t)^2 \sqrt{\frac{2a}{4+8\Delta t} - (\frac{a}{4+8\Delta t})^2}} \\
&= \frac{6}{\sqrt{4 - (1 + 3\Delta t)^2}} + \frac{64a}{(4 + 8\Delta t)\sqrt{2a(4 + 8\Delta t) - a^2}} \\
&\leq \frac{6}{\sqrt{4 - (1 + 3\Delta t)^2}} + \frac{16a}{\sqrt{8a - a^2}} \\
&\leq 4.14 + 20.86 \leq 43 \leq K
\end{aligned}$$

Here we used that $3/\sqrt{4 - (1 + 3\Delta t)^2}$ is increasing in Δt and that $\Delta t \leq \frac{1}{8}$. We conclude that $K\Delta t \geq \Delta E(z, \Delta t) - J(z)$ for all $\Delta t \in [0, T(z)]$. Thus, $\beta(t)$ does not leave I during $(t, t + T(z)]$. By repeating this argument whenever $\beta(t)$ leaves the safe zone, we can conclude that $|\beta(t) - \alpha(t)| \leq 8 \arcsin(z)$ for all times t. ◀

4.2 Lipschitz stability ratio

What remains is to analyze the approximation ratio of the chasing algorithm for OBB. Note that, while we already chose $c = 3$ for Lemma 14, we finish the analysis of the solution quality with c as a variable. This demonstrates how the solution quality depends on the values of c, and shows how a different choice of c would lead to a different approximation factor.

Corollary 11 implies that the orientation β of the chasing algorithm is at most an angle $(2c + 2) \arcsin(z)$ away from the orientation of the diameter.

▶ **Lemma 15.** *If* $|\beta - \alpha| \leq (2c + 2) \arcsin(z)$*, then* $f_{OBB}(\beta, P) \leq (4c + 6) \min_\phi f_{OBB}(\phi, P)$.

Proof. Assume that at some time t we have a diametric box with diameter D and aspect ratio z, and let (p_1, p_2) be the diametrical pair we are chasing. Additionally, let points q_1 and q_2 lie on the longer sides of the diametric box, defining its width. The smallest OBB must contain p_1, p_2, q_1, and q_2, and therefore contains the triangles formed by $\{p_1, p_2, q_1\}$ and $\{p_1, p_2, q_2\}$. With the diametrical pair a fraction x along the minor axis in the box, the heights of these triangles with base p_1p_2 are $x \cdot Dz$ and $(1 - x) \cdot Dz$ respectively (see Figure 11a). Their total area is thus $D^2z/2$ and provides a lower bound for the area of OBB.

Now consider the box of the chasing algorithm, where $\Delta\alpha = |\beta - \alpha| \leq (2c + 2) \arcsin(z)$. The major axis (in direction β) has length at most D. Let the minor axis be bounded by two points v_1 and v_2, and γ the angle between the lines spanned by v_1v_2 and by the diametrical pair p_1p_2, on the opposite side of $\Delta\alpha$ with respect to p_1p_2. Let the smallest angle between those two lines be γ'. Note that in the worst case v_1 and v_2 are located on the boundary of the diametric box. Would those points not lie on the boundary, then we could move them there and increase the area of the chasing box. The distance between v_1 and v_2 is therefore bounded by $zD/\sin(\gamma')$. Whenever $\gamma' = \gamma$, the angle between the minor axis of the chasing box and the line through v_1 and v_2 is $\pi/2 - \gamma - \Delta\alpha$ (see Figure 11b). Thus, the length of the minor axis is $zD\cos(\pi/2 - \gamma - \Delta\alpha)/\sin(\gamma) = zD\sin(\gamma + \Delta\alpha)/\sin(\gamma)$.

However, it can also be the case that $\gamma' = \pi - \gamma$ (see Figure 11c). The angle between the minor axis of the box and the line through v_1 and v_2 is now $\pi/2 - \gamma' + \Delta\alpha$. Analogously, we hence find that the length of the minor axis is $zD\cos(\pi/2 - \gamma' + \Delta\alpha)/\sin(\gamma') = zD\sin(\gamma' - \Delta\alpha)/\sin(\gamma')$. Using $\gamma' = \pi - \gamma$, the length of the minor axis can be simplified to $zD\sin(\gamma + \Delta\alpha)/\sin(\gamma)$, which is the same expression as for $\gamma = \gamma'$.

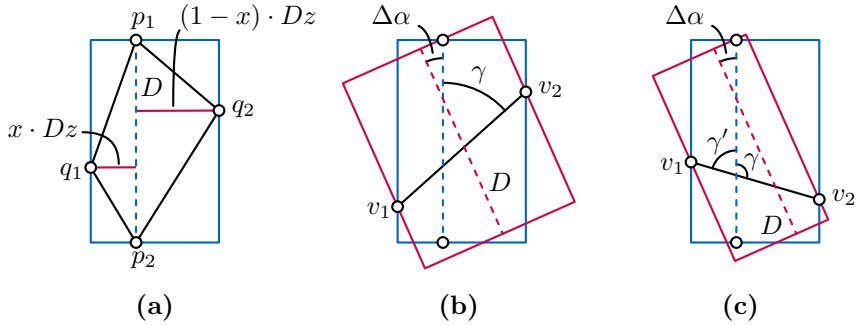

Figure 11 Illustrations supporting proof of Lemma 15: in blue the diametric box, and for **(b)** and **(c)** the box of the chasing algorithm in red.

We now look for the maximum of the function $zD\sin(\gamma+\Delta\alpha)/\sin(\gamma)$, which is obtained for $\gamma+\Delta\alpha\leq\frac{\pi}{2}$. Additionally, we know that $\gamma\geq\arcsin(z)$, since v_1 and v_2 are not the diametrical pair defining the major axis of the diametric box. Hence, under these restrictions the function $zD\sin(\gamma+\Delta\alpha)/\sin(\gamma)$ is decreasing in γ, we attain the maximum when $\gamma=\arcsin(z)$. Substituting $\gamma=\arcsin(z)$, we get $zD\sin(\arcsin(z)+\Delta\alpha)/\sin(\arcsin(z))=D\sin(\arcsin(z)+\Delta\alpha)$. Finally, $\Delta\alpha$ lies between 0 and $(2c+2)\arcsin(z)$. We derived that $\gamma+\Delta\alpha\leq\frac{\pi}{2}$, so if $\Delta\alpha=(2c+2)\arcsin(z)$ would lead to $\gamma+\Delta\alpha>\frac{\pi}{2}$, then we can find a constant x, with $0\leq x\leq(2c+2)$, which ensures that $\gamma+x\arcsin(z)\leq\frac{\pi}{2}$. Multiplying the lengths of the major and minor axes results in an area of at most $D^2\sin((1+x)\arcsin(z))$ for the box of the chasing algorithm. We apply Lemma 12 to get an area between D^2z and $D^2z(2c+3)$. Thus, $f_{\text{OBB}}(\beta,P)\leq(4c+6)min_\phi f_{\text{OBB}}(\phi,P)$. ◀

▶ **Lemma 16.** *If* $|\beta-\alpha|\leq(2c+2)\arcsin(z)$, *then* $f_{STRIP}(\beta,P)\leq(4c+6)\min_\phi f_{STRIP}(\phi,P)$.

Proof. Assume that at some time t we have a diametric box with diameter D and aspect ratio z, and let (p_1,p_2) be the diametrical pair we are chasing. The width of the diametric box is determined by two points q_1 and q_2.

We first derive a lower bound for the width of the thinnest STRIP; note that this follows the same rationale as in the proof of Lemma 9 (and as illustrated in Figures 10a and 10b). Such a strip must contain the points p_1, p_2, q_1 and q_2. As adding points to a point set can only widen the thinnest strip, we consider just these four points for a lower bound. For the thinnest STRIP, all four points are on the boundary of the strip in the worst case, and we assume w.l.o.g. that p_1 and q_1 are on the same side of the strip (same for p_2 and q_2). Consider the following lines: L oriented in the orientation of the strip (parallel to its boundary), L_p spanned by p_1p_2 and L_q spanned by q_1q_2. Let the angle between L_p and L_q be γ, $\gamma\geq\arcsin(z)$. The distance between q_1 and q_2 is then $zD/\sin(\gamma)$. We denote the angle between L and L_p by γ_p, and between L and L_q by γ_q. We observe that $\gamma_p+\gamma_q=\gamma$, as the orientation of L must bisect the γ angle for the strip to be thinnest. The width of the strip is $\max(D\sin(\gamma_p),zD\sin(\gamma_q)/\sin(\gamma))$. We show that this width is at least $D\sin(\frac{1}{2}\arcsin(z))$. This is clearly the case if $\gamma_p\geq\frac{1}{2}\arcsin(z)$, so assume the contrary. Since the function $\sin(\gamma-\gamma_p)/\sin(\gamma)$ is increasing, it is optimal to set $\gamma=\arcsin(z)$. But then $zD\sin(\gamma_q)/\sin(\gamma)=D\sin(\gamma_q)>D\sin(\frac{1}{2}\arcsin(z))$. Thus, the width of the thinnest strip is at least $D\sin(\frac{1}{2}\arcsin(z))\geq Dz/2$ by Lemma 12.

Now consider the strip of the chasing algorithm, with an orientation β differing at most $\Delta\alpha=|\beta-\alpha|\leq(2c+2)\arcsin(z)$ from the orientation of the diametrical pair we are chasing. To prove an upper bound on the width of this strip we follow the same approach as in the proof

of Lemma 15 (and as illustrated in Figures 11b and 11c). Let the width of the strip be bounded by two points v_1 and v_2, where the angle between the line through v_1 and v_2 and the line through the diametrical pair p_1p_2, opposite of $\Delta\alpha$ with respect to p_1p_2, is γ. Let the smallest angle between those two lines be γ'. Note that, the distance between v_1 and v_2 is $zD/\sin(\gamma')$, since they define the width of the diameter box. When $\gamma' = \gamma$, the angle between the vector perpendicular to the orientation of the strip and the line through v_1 and v_2 is $\pi/2 - \gamma - \Delta\alpha$. Thus, the width of the strip is $zD\cos(\pi/2 - \gamma - \Delta\alpha)/\sin(\gamma) = zD\sin(\gamma + \Delta\alpha)/\sin(\gamma)$.

On the other hand, whenever $\gamma' = \pi - \gamma$, the angle between the vector perpendicular to β and the line v_1v_2 is $\pi/2 - \gamma' + \Delta\alpha$. Analogously, we find that the width of the strip is $zD\cos(\pi/2 - \gamma' + \Delta\alpha)/\sin(\gamma') = zD\sin(\gamma' - \Delta\alpha)/\sin(\gamma')$. Using $\gamma' = \pi - \gamma$, the width of the strip can be simplified to $zD\sin(\gamma + \Delta\alpha)/\sin(\gamma)$, which is the same expression as for the case in which $\gamma' = \gamma$.

We now look for the maximum of the function $zD\sin(\gamma + \Delta\alpha)/\sin(\gamma)$, in the same way as for Lemma 15, and find it for $\gamma = \arcsin(z)$. Thus, using $\gamma = \arcsin(z)$, the width of the strip is at most $D\sin((2c+3)\arcsin(z))$, which is at most $Dz(2c+3)$ by Lemma 12. We finally obtain that $f_{\text{STRIP}}(\beta, P) \leq (4c+6)\min_\phi f_{\text{STRIP}}(\phi, P)$. ◀

By combining Lemmata 14, 15, and 16, we obtain the following bounds on the Lipschitz stability of OBB and STRIP.

▶ **Theorem 17.** *The following Lipschitz stability ratios hold for* OBB *and* STRIP*, assuming diameter $D(t) \geq 1$ for all t and points move with at most unit speed:*

- $\rho_{\text{LS}}(\textit{OBB}, 43) \leq 18$,
- $\rho_{\text{LS}}(\textit{STRIP}, 43) \leq 18$.

5 Lipschitz stability of PC

The chasing algorithm does not work for the first principal component. Specifically, our scale normalization by requiring that the diameter is at least one at any point in time does not help. This can intuitively be attributed to the optimization function of PC. Rather than being defined by some form of extremal points, f_{PC} is determined by variance: although the diameter may be large, many close points may still largely determine the first principal component. We formalize this via the lemma below. It implies that requiring a minimal diameter is not sufficient for a chasing algorithm with bounded speed to approximate PC. The proof is inspired by the construction in [13] that shows the ratio on the areas between a bounding box aligned with the principal components and the optimal oriented bounding box can become infinite.

▶ **Lemma 18.** *For any constant K, there exists a point set $P(t)$ with a diameter of at least 1 at all times, such that any shape descriptor that approximates the optimum of f_{PC} must move with speed strictly greater than K.*

Proof. For the sake of contradiction, assume that, for a certain K, all point sets with a diameter of at least 1 at all times admit a shape descriptor that is a ρ-approximation of the optimum of f_{PC} while moving with speed at most K. We derive a contradiction by constructing a point set P for which moving with speed at most K cannot be sufficient to approximate f_{PC}; see Figure 12 for illustration.

P consists of $2 + n$ points. The first two points p_1 and p_2 are stationary at $(-1, 0)$ and $(1, 0)$ – this immediately guarantees that the diameter is always larger than 1. Furthermore, P has n moving points, denoted by set P', that are symmetrically and colinearly placed near the origin at distances in some interval $[b/2, b]$. As such, the mean of P is the origin.

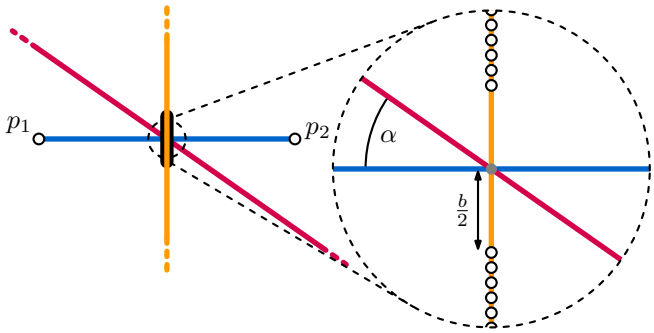

Figure 12 Construction that shows how PC can move arbitrarily fast, despite a minimal diameter. The two points connected by the blue line form a diametrical pair for the whole point set. The dense set of points located at distance $b/2$ from the origin (in grey) can move around the origin in a short amount of time to become vertical (in yellow). The orientation of a K-Lipschitz stable descriptor (in red) cannot move fast enough to match this change.

Recall that f_{PC} measures the sum of squared distances from each point to a line of a given orientation through the mean of the point set – in our case through the origin. Initially, we place P' horizontally making all the points collinear and thus f_{PC} is zero for the horizontal orientation and non-zero for any other orientation. Hence, the assumed shape descriptor must be horizontal initially – otherwise, it would not be a ρ-approximation.

We now rotate the points in P' such that the line through P' becomes vertical (rotation over $\frac{\pi}{2}$) in $\frac{\pi}{4K}$ time. Now, the shape descriptor can rotate over at most $\frac{\pi}{4}$ in this same time. The vertical line has cost 2 (1^2 for each of p_1 and p_2) and thus the optimal orientation has cost at most 2. What remains is to choose values for b and n such that any shape descriptor with angle α at most $\frac{\pi}{4}$ to the horizontal axis has cost strictly larger than 2ρ.

For points moving at unit speed, our described motion is feasible in the given time if $b = \frac{\pi}{4K} / \frac{\pi}{2} = \frac{1}{2K}$. We observe that p_1 and p_2 together add a cost of $2\sin^2\alpha$, and each of the points in P' adds at least $(b/2)^2\cos^2\alpha = \frac{1}{16K^2}\cos^2\alpha$. Since the shape descriptor can reach only angles of at most $\frac{\pi}{4}$ in the alotted time, we know that $\cos^2\alpha \geq \frac{1}{2}$ and $\sin^2\alpha \geq 0$ regardless of α. So, the total cost for the shape descriptor at angle α is at least $2\sin^2\alpha + n\frac{1}{16K^2}\cos^2\alpha \geq \frac{n}{32K^2}$. However, we can add arbitrarily many points in P' for our construction. We obtain a contradiction when $\frac{n}{32K^2} > 2\rho$ and thus we may set $n > 64K^2\rho$. ◀

6 Conclusion

We studied the topological stability (ratio of continuous solutions to optimal discontinuous solutions) and Lipschitz stability (ratio of continuous solutions with bounded speed to optimal discontinuous solutions) of three common orientation-based shape descriptors. Although stateless algorithms cannot achieve topological stability, we proved tight bounds on the topological stability ratio for state-aware algorithms. Our Lipschitz analysis focuses on upper bounds, showing that a chasing algorithm achieves a constant ratio for a constant maximum speed, for two of the three considered descriptors. It remains open to establish whether lower bounds exist that are stronger than those already given by our topological stability results. We believe that our approach for the analysis of the Lipschitz upper bounds are of independent interest, and can be applied to analyze other problems that could be approached via a chasing algorithm.

References

1 P. Agarwal, L. Guibas, J. Hershberger, and E. Veach. Maintaining the extent of a moving point set. *Discrete & Computational Geometry*, 26(3):353–374, 2001. doi:10.1007/s00454-001-0019-x.

2 P. Agarwal, S. Har-Peled, and K. Varadarajan. Approximating extent measures of points. *Journal of the ACM*, 51(4):606–635, 2004. doi:10.1145/1008731.1008736.

3 Pankaj K. Agarwal and Sariel Har-Peled. Maintaining approximate extent measures of moving points. In S. Rao Kosaraju, editor, *Proc. 12th Symposium on Discrete Algorithms (SODA)*, pages 148–157, 2001. URL: http://dl.acm.org/citation.cfm?id=365411.365431.

4 G. Barequet, B. Chazelle, L. Guibas, J. Mitchell, and Ayellet Tal. BOXTREE: A hierarchical representation for surfaces in 3d. *Computer Graphics Forum*, 15(3):387–396, 1996. doi:10.1111/1467-8659.1530387.

5 G. Barequet and S. Har-Peled. Efficiently approximating the minimum-volume bounding box of a point set in three dimensions. *Journal of Algorithms*, 38(1):91–109, 2001. doi:10.1006/jagm.2000.1127.

6 N. Beckmann, H. Kriegel, R. Schneider, and B. Seeger. The R*-tree: An efficient and robust access method for points and rectangles. In *Proc. 1990 ACM SIGMOD*, pages 322–331, 1990. doi:10.1145/93597.98741.

7 S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(4):509–522, 2002. doi:10.1109/34.993558.

8 S. Bereg, B. Bhattacharya, D. Kirkpatrick, and M. Segal. Competitive algorithms for maintaining a mobile center. *Mobile Networks and Applications*, 11(2):177–186, 2006. doi:10.1007/s11036-006-4470-z.

9 L. Brouwer. Über Abbildung von Mannigfaltigkeiten. *Mathematische Annalen*, 71(1):97–115, 1911.

10 D. Brzakovic, X. Mei Luo, and P. Brzakovic. An approach to automated detection of tumors in mammograms. *IEEE Transactions on Medical Imaging*, 9(3):233–241, 1990. doi:10.1109/42.57760.

11 F. Chin, C. An Wang, and F. Lee Wang. Maximum stabbing line in 2d plane. In *Computing and Combinatorics*, pages 379–388, 1999. doi:10.1007/3-540-48686-0_38.

12 M. de Berg, M. Roeloffzen, and B. Speckmann. Kinetic 2-centers in the black-box model. In *Proc. 29th Symposium on Computational Geometry (SOCG)*, pages 145–154, 2013. doi:10.1145/2462356.2462393.

13 D. Dimitrov, C. Knauer, K. Kriegel, and G. Rote. Bounds on the quality of the PCA bounding boxes. *Computational Geometry*, 42(8):772–789, 2009. doi:10.1016/j.comgeo.2008.02.007.

14 S. Durocher and D. Kirkpatrick. The Steiner centre of a set of points: Stability, eccentricity, and applications to mobile facility location. *International Journal of Computational Geometry and Applications*, 16(04):345–371, 2006. doi:10.1142/S0218195906002075.

15 S. Durocher and D. Kirkpatrick. Bounded-velocity approximation of mobile Euclidean 2-centres. *International Journal of Computational Geometry and Applications*, 18(03):161–183, 2008. URL: https://doi.org/10.1142/S021819590800257X.

16 Stephane Durocher and David G. Kirkpatrick. The projection median of a set of points. *Computational Geometry*, 42(5):364–375, 2009. doi:10.1016/J.COMGEO.2008.06.006.

17 H. Freeman and R. Shapira. Determining the minimum-area encasing rectangle for an arbitrary closed curve. *Communications of the ACM*, 18(7):409–413, 1975. doi:10.1145/360881.360919.

18 S. Gottschalk, M. Lin, and D. Manocha. OBBTree: A hierarchical structure for rapid interference detection. In *Proc. 23rd SIGGRAPH*, pages 171–180, 1996. doi:10.1145/237170.237244.

19 A. Guttman. R-trees: A dynamic index structure for spatial searching. In *Proc. 1984 ACM SIGMOD*, pages 47–57, 1984. doi:10.1145/602259.602266.

20 J. Klosowski, M. Held, J. Mitchell, H. Sowizral, and K. Zikan. Efficient collision detection using bounding volume hierarchies of k-DOPs. *IEEE Transactions on Visualization and Computer Graphics*, 4(1):21–36, 1998. doi:10.1109/2945.675649.

21 D. Letscher and K. Sykes. On the stability of medial axis of a union of disks in the plane. In *Proc. 28th Canadian Conference on Computational Geometry (CCCG)*, pages 29–33, 2016.

22 W. Meulemans, B. Speckmann, K. Verbeek, and J. Wulms. A framework for algorithm stability and its application to kinetic Euclidean MSTs. In *Proc. 13th Latin American Theoretical Informatics Symposium (LATIN)*, pages 805–819, 2018. doi:10.1007/978-3-319-77404-6_58.

23 J. Milnor and D. Weaver. *Topology from the differentiable viewpoint*. Princeton U. Press, 1997.

24 J. O'Rourke. Finding minimal enclosing boxes. *International Journal of Parallel Programming*, 14(3):183–199, 1985. doi:10.1007/BF00991005.

25 N. Roussopoulos and D. Leifker. Direct spatial search on pictorial databases using packed r-trees. In *Proc. 1985 ACM SIGMOD*, pages 17–31, 1985. doi:10.1145/318898.318900.

26 T. Sellis, N. Roussopoulos, and C. Faloutsos. The R+-tree: A dynamic index for multi-dimensional objects. In *Proc. 13th International Conference on Very Large Data Bases*, pages 507–518, 1987. URL: http://www.vldb.org/conf/1987/P507.PDF.

27 G. Toussaint. Solving geometric problems with the rotating calipers. In *Proc. IEEE Melecon*, volume 83, page A10, 1983.

28 E. Tyrtyshnikov. *A brief introduction to numerical analysis*. Springer, 2012.

29 G. van den Bergen. Efficient collision detection of complex deformable models using AABB trees. *Journal of Graphics, GPU, & Game Tools*, 2(4):1–13, 1997. doi:10.1080/10867651.1997.10487480.

30 Ivor van der Hoog, Marc J. van Kreveld, Wouter Meulemans, Kevin Verbeek, and Jules Wulms. Topological stability of kinetic k-centers. *Theoretical Computer Science*, 866:145–159, 2021. doi:10.1016/j.tcs.2021.03.026.

31 Jules Wulms, Juri Buchmüller, Wouter Meulemans, Kevin Verbeek, and Bettina Speckmann. Stable visual summaries for trajectory collections. In *Proc. 14th Pacific Visualization Symposium (PacificVis)*, pages 61–70, 2021. doi:10.1109/PacificVis52677.2021.00016.

32 J. Xie, G. Dai, F. Zhu, E. Wong, and Y. Fang. DeepShape: Deep-learned shape descriptor for 3D shape retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(7):1335–1345, 2017. doi:10.1109/TPAMI.2016.2596722.

33 H. Zhang, A. Berg, M. Maire, and J. Malik. SVM-KNN: Discriminative nearest neighbor classification for visual category recognition. In *Proc. 19th Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2126–2136, 2006. doi:10.1109/CVPR.2006.301.

34 Y. Zhong. Intrinsic shape signatures: A shape descriptor for 3d object recognition. In *Proc. 12th International Conference on Computer Vision (ICCV) Workshops*, pages 689–696, 2009. doi:10.1109/ICCVW.2009.5457637.

35 B. Zitová and J. Flusser. Image registration methods: a survey. *Image and Vision Computing*, 21(11):977–1000, 2003. doi:10.1016/S0262-8856(03)00137-9.