Trajectory Grouping Structure under Geodesic Distance

Irina Kostitsyna[1], Marc van Kreveld[2], Maarten Löffler[2], Bettina Speckmann[1], and Frank Staals[2]

1 **Department of Mathematics and Computer Science, TU Eindhoven, The Netherlands**
{i.kostitsyna|b.speckmann}@tue.nl

2 **Department of Information and Computing Sciences, Utrecht University, The Netherlands**
{m.j.vankreveld|m.loffler|f.staals}@uu.nl

Abstract

In recent years trajectory data has become one of the main types of geographic data, and hence algorithmic tools to handle large quantities of trajectories are essential. A single trajectory is typically represented as a sequence of time-stamped points in the plane. In a collection of trajectories one wants to detect maximal groups of moving entities and their behaviour (merges and splits) over time. This information can be summarized in the *trajectory grouping structure*.

Significantly extending the work of Buchin et al. [WADS 2013] into a realistic setting, we show that the trajectory grouping structure can be computed efficiently also if obstacles are present and the distance between the entities is measured by geodesic distance. We bound the number of *critical events*: times at which the distance between two subsets of moving entities is exactly ε, where ε is the threshold distance that determines whether two entities are close enough to be in one group. In case the n entities move in a simple polygon along trajectories with τ vertices each we give an $O(\tau n^2)$ upper bound, which is tight in the worst case. In case of *well-spaced* obstacles we give an $O(\tau(n^2 + m\lambda_4(n)))$ upper bound, where m is the total complexity of the obstacles, and $\lambda_s(n)$ denotes the maximum length of a Davenport-Schinzel sequence of n symbols of order s. In case of general obstacles we give an $O(\tau \min\{n^2 + m^3\lambda_4(n), n^2m^2\})$ upper bound. Furthermore, for all cases we provide efficient algorithms to compute the critical events, which in turn leads to efficient algorithms to compute the trajectory grouping structure.

1998 ACM Subject Classification F.2.2 Analysis of Algorithms and Problem Complexity

Keywords and phrases moving entities, trajectories, grouping, computational geometry

Digital Object Identifier 10.4230/LIPIcs.SOCG.2015.674

1 Introduction

Tracking moving entities like humans, vehicles and animals is becoming more and more commonplace, with applications in security (what human movement is suspicious behavior?), the social sciences (which people move together? what regions do they avoid?), biology (what are migration routes and what are the stopping places?), and traffic analysis. Technology like GPS, RFID, and video has led to large data sets with trajectories, representing the movement of entities. At a similar pace, more and more algorithmic tools to analyze such data are being developed within the areas of Geographic Information Science, data mining, and computational geometry.

In most cases, each trajectory is represented by a sequence of time-stamped points in the plane or in space. As such, trajectories can be seen as a form of time-series data with a

31st International Symposium on Computational Geometry (SoCG'15).
Editors: Lars Arge and János Pach; pp. 674–688
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

geometric component. Collections of trajectories can be processed for retrieving patterns like clusters, flocks, leadership, encounter, and many more [1, 7, 10, 11, 12]. Trajectory data can also be linked to the environment, available in other spatial data sets, to determine more types of patterns [3, 4].

Recent research has gone beyond identifying flocks or moving clusters separately by modelling all joint movements into the trajectory grouping structure [2]. This structure captures the joining and splitting of groups of entities by employing methods from computational topology, in particular, the Reeb graph [6]. Distances between moving entities are among the main criteria to decide if entities belong to the same group (see below for a precise definition). In this paper we significantly extend the trajectory grouping structure by incorporating obstacles and measuring distances as geodesic distances. The geodesic distance between two entities is the distance that needs to be traversed for one entity to reach the other entity. This approach gives a more natural notion of groups because it separates entities moving on opposite sides of obstacles like fences or water bodies. A threshold distance denoted by ε determines whether two entities are close enough to be in the same group. Hence we examine the number of times that a threshold distance occurs among n moving entities. Only threshold distances between the closest two entities of different groups matters, so we analyze the number of events of this type for various obstacle settings.

The combination of moving points and specific structures defined by these points has been a topic of major interest in computational geometry; for example, one of the main open problems in the area is the question "How many times can the Delaunay triangulation change its combinatorial structure in the worst case, when n points move along straight lines in the plane?" Other related research on movement in geometric algorithms concerns kinetic data structures. To our knowledge, our paper is the first to combine continuously moving points with geodesic distances in the plane. We expect that our analysis will be of interest to other distance problems on moving points than the trajectory grouping structure. For example, in a similarity measure for two trajectories that incorporates obstacles.

Terminology and notation. We are given a set $\mathcal{X}$ of n entities, each moving along a piecewise linear trajectory with τ vertices, and a set of pairwise disjoint polygonal obstacles $\mathbb{O} = \{\mathcal{O}_1, .., \mathcal{O}_h\}$. Let m denote the total complexity of $\mathbb{O}$.

We denote the position of entity a at time t by $a(t)$. Let $\|pq\|$ denote the Euclidean distance between points p and q, and let $\xi_{ab}(t) = \|a(t)b(t)\|$ denote the (Euclidean) distance between entities a and b at time t. A *path* $P = p_1, .., p_k$ from p_1 to p_k is a polygonal line with vertices $p_1, .., p_k$, and has length $\varsigma(P) = \sum_{i=1}^{k-1} \|p_i p_{i+1}\|$. A path is *obstacle-avoiding* if it is disjoint from the interior of all obstacles in $\mathbb{O}$. A path between p and q is a *geodesic*, denoted $g(p, q)$, if it has minimum length among all obstacle-avoiding paths. We refer to the length of $g(p, q)$ as the *geodesic distance* between p and q. We denote the geodesic distance between a and b at time t by $\varsigma_{ab}(t) = \varsigma(g(a(t), b(t)))$.

To determine if a set of entities may form a group, we have to decide if they are close together. Analogous to Buchin et al. [2] we model this by a spatial parameter ε. More specifically, two entities a and b are *directly connected* at time t if they are within (geodesic) distance ε from each other, that is, $\varsigma_{ab}(t) \leq \varepsilon$. A set of entities $\mathcal{X}'$ is *ε-connected* at time t if for any pair $a, b \in \mathcal{X}'$ there is a sequence $a = a_0, a_1, .., a_k = b$ such that a_i and a_{i+1} are directly connected. We refer to a time at which a and b become directly connected or disconnected as an *ε-event*. At such a time the distance between a and b is exactly ε. If an ε-event also connects or disconnects the maximal ε-connected set(s) containing a and b, it is a *critical event*. A (maximal) ε-connected set of entities $\mathcal{X}'$ is a *group* if it is ε-connected at any time t in a time interval of length at least δ, and it has at least a certain size.

Table 1 The number of critical events (i.e. the size of $\mathcal{R}$), and the time required to construct $\mathcal{R}$. Note that the input size is $\Theta(\tau n + m)$.

	Lower bound	Upper bound	Algorithm
Simple polygon	$\Omega(\tau n^2)$	$O(\tau n^2)$	$O(\tau n^2(\log^2 m + \log n) + m)$
Well-spaced obstacles	$\Omega(\tau(n^2 + nm))$	$O(\tau(n^2 + m\lambda_4(n)))$	$O(\tau n^2 m \log n)$
General obstacles	$\Omega(\tau(n^2 + nm \min\{n, m\}))$	$O(\tau \min\{n^2 + m^3\lambda_4(n), n^2m^2\})$	$O(\tau n^2 m^2 \log n + m^2 \log m)$

Trajectory grouping structure. Since the objective of our methods is to compute the trajectory grouping structure as defined by Buchin et al. [2], we review their structure here. It captures not just the groups, but also how and when they arise, merge, split, or stop to be a group. Only maximal groups are considered, where groups can be maximal in size and in duration.

The evolution of the maximal ε-connected sets as the entities move is directly represented by a directed acyclic graph (DAG) $\mathcal{R}$. Edges of the graph correspond to the maximal ε-connected sets and the nodes correspond to structural changes, that is, critical events. For example, a node may represent a critical event where two maximal ε-connected sets get close enough to become one ε-connected set: the node will have in-degree 2 and out-degree 1 and represents a join. This DAG $\mathcal{R}$ is a Reeb graph [6]. Each entity is associated with a directed path in $\mathcal{R}$ in the natural way.

Groups are defined as above. We are interested only in maximal groups: a subset S for a time interval I is a maximal group if (i) S is in an ε-connected subset during I, (ii) I has length at least δ, (iii) S has at least the required size, (iv) no proper superset of S or proper superinterval of I exists with the same properties. Maximal groups are associated with a directed (sub)path in $\mathcal{R}$ in a natural way.

Buchin et al. [2] show that when there are no obstacles the maximum complexity of $\mathcal{R}$ is $\Theta(\tau n^2)$ in the worst case, and it can be computed in $O(\tau n^2 \log n)$ time. Furthermore, there are $\Theta(\tau n^3)$ maximal groups in the worst case, and they can be reported in $O(\tau n^3 \log n + N)$ time, where N is the output size (which is $O(\tau n^4)$).

Results and organization. We extend the results of Buchin et al. [2] to the case where the entities move amidst obstacles, and we thus measure the distance between two entities a and b by their geodesic distance ς_{ab}. Instead of having $O(\tau n^2)$ events that correspond to the nodes in $\mathcal{R}$, we can have more events, depending on the obstacles and their complexity.

We study three settings for the obstacles. In the simplest case, all entities move inside a simple polygon with m vertices. In the most general case, obstacles can have any shape, location, and complexity, but they are disjoint and have total complexity m. As an intermediate case we assume that the distance between any two non-adjacent obstacle edges is at least ε. We say that the obstacles are *well-spaced*.

Our results on the number of critical events, and thus the size of the Reeb-graph, for the three cases, are listed in Table 1. For the simple polygon case, which we treat in Section 3, our bounds are tight. The upper bounds for the well-spaced obstacles case, and the general obstacles case include a $\lambda_4(n)$ term, where $\lambda_s(n)$ denotes the maximum length of a Davenport-Schinzel sequence of order s with n symbols. Since $\lambda_4(n)$ is only slightly superlinear, our bound for the well-spaced obstacles case is almost tight. We present these results in Sections 4 and 5, respectively. For all cases we also bound the total number of ε-events, and we show how to compute $\mathcal{R}$ efficiently. Omitted proofs can be found in the full version.

Once we have the Reeb graph $\mathcal{R}$ describing connectivity events of the entities in $\mathcal{X}$, we can use the existing analysis by Buchin et al. [2] to bound the number of maximal groups as well as their algorithm(s) to compute these groups. So the interesting part is in analyzing the complexity of $\mathcal{R}$ and determining how to compute it.

2 Distance Functions

Let a and b be two entities, each moving along a straight line during interval I, and let p be a fixed point in $\mathbb{R}^2$. During I the Euclidean distance $\xi_{ap}(t)$ between a and p is a convex hyperbolic function in t that has the form $\sqrt{Q(t)}$, for some quadratic function Q. The Euclidean distance between a and b during I is a convex hyperbolic function of the same form. Since ξ_{ap} is convex, there are at most two times in I such that $\xi_{ap}(t) = \varepsilon$. The same applies for ξ_{ab}.

The geodesic distance $\varsigma_{ap}(t)$ between a and p is a piecewise function. At times where the geodesic $g(a(t), p)$ consists of a single line segment, the geodesic distance is simply the Euclidean distance. When the geodesic consists of more than one line segment we can decompose it into two parts: a line segment $g(a(t), u) = \overline{a(t)u}$, and a path $g(u, p)$, where u is the first obstacle vertex on $g(a(t), p)$. Similarly, if the geodesic $g(a(t), b(t))$ between a and b consists of more than one segment we can decompose it into three parts $\overline{a(t)u}$, $g(u, v)$, and $\overline{vb(t)}$ (we may have $u = v$). It follows that each piece of ς_{ap} is convex and hyperbolic. The pieces of ς_{ab} are convex as well, since they are of the form $\xi_{au}(t) + C + \xi_{vb}(t) = \sqrt{Q_1(t)} + C + \sqrt{Q_2(t)}$, for some quadratic functions Q_1 and Q_2 and a constant C. Therefore, we again have that on each piece there are at most two times where $\varsigma_{ap}(t)$ is exactly ε. The same applies for $\varsigma_{ab}(t)$.

We obtain the same results when a and b move on piecewise linear trajectories, rather than lines. The functions then simply consist of more pieces.

▶ **Lemma 1.** *Let $\mathcal{F} = f_1, .., f_n$ be a set of n piecewise (partial) functions, each function f_i consisting of τ pieces $f_i^1, .., f_i^\tau$, such that any two pieces f_i^k and f_j^ℓ intersect each other at most s times. The lower envelope $\mathcal{L}$ of $\mathcal{F}$ has complexity $O(\tau\lambda_{s+2}(n))$.*

Analogous to Lemma 1 we can show that the upper envelope of $\mathcal{F}$ has complexity $O(\tau\lambda_{s+2}(n))$.

3 Simple Polygon

We first focus our attention on entities moving in a simply-connected polygonal domain.

3.1 Lower Bound

Buchin et al. [2] show that the number of critical events for n entities moving in $\mathbb{R}^2$ without obstacles can be $\Omega(\tau n^2)$. Clearly, this lower bound also holds for entities moving inside a simple polygon.

3.2 Upper Bound

Let a and b be two entities, each moving along a line during interval I, and let $\varsigma(t) = \varsigma_{ab}(t)$ be the function describing the geodesic distance between a and b during interval I.

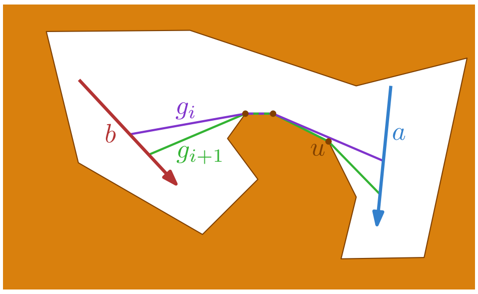

Figure 1 Geodesics g_i (purple) and g_{i+1} differ by at most one vertex; the first vertex u on g_{i+1}.

▶ **Lemma 2.** *The function ς is convex.*

Proof Sketch. Let $[t_{i-1}, t_i]$ and $[t_i, t_{i+1}]$ be two consecutive time intervals, corresponding to pieces ς_i and ς_{i+1} of ς. We now show that ς is convex on $[t_{i-1}, t_{i+1}]$.

Let g_i and g_{i+1} denote the geodesic shortest paths corresponding to ς_i and ς_{i+1}, respectively. Geodesics g_i and g_{i+1} differ by at most one vertex u (assuming general position of the obstacle vertices), and this vertex occurs either at the beginning or the end of the geodesic. Consider the case that u is the first vertex of g_{i+1}, and u does not occur on g_i. See Figure 1. All other cases are symmetric. Let v be the second vertex of g_{i+1} (and thus the first vertex of g_i). We have $\varsigma_i(t) = \|a(t)v\| + \varsigma(v, b(t))$ and $\varsigma_{i+1}(t) = \|a(t)u\| + \|uv\| + \varsigma(v, b(t))$. It follows that the individual pieces ς_i and ς_{i+1} are (convex) hyperbolic functions, that $\varsigma_i(t_i) = \varsigma_{i+1}(t_i)$, and that for any time $t \in [t_{i-1}, t_{i+1}]$, $\varsigma_{i+1}(t) \geq \varsigma_i(t)$. We use these properties to show that for any three times $s, m, t \in [t_{i-1}, t_{i+1}]$, with $s \leq m \leq t$, the point $\varsigma(m)$ lies below the line segment (function) $\overline{\varsigma(s)\varsigma(t)}$, that is $\varsigma(m) \leq \overline{\varsigma(s)\varsigma(t)}(m)$. Since ς_i and ς_{i+1} are convex, the only interesting case is when s lies on ς_i and t lies on ς_{i+1}. We prove this by case distinction on m. It follows that ς is convex on $[t_{i-1}, t_{i+1}]$. ◀

▶ **Theorem 3.** *Let $\mathcal{X}$ be a set of n entities, each moving in a simple polygon along a piecewise linear trajectory with τ vertices. The number of ε-events is at most $O(\tau n^2)$.*

Proof. Fix a pair of entities a and b. Both a and b move along trajectories with τ vertices. So there are $2\tau - 1$ intervals during which both a and b move along a line. During each such interval ς_{ab} is convex (Lemma 2). So there are at most two times in each interval at which $\varsigma_{ab}(t) = \varepsilon$. The lemma follows. ◀

3.3 Algorithm

Next, we describe how to compute all ε-events. The high level overview of our algorithm is as follows. For each pair of entities a and b, we first find a time $t_{\min}$ such that the geodesic distance $\varsigma(t) = \varsigma_{ab}(t)$ between a and b is minimal. Clearly, if $\varsigma(t_{\min}) > \varepsilon$ there is no time at which a and b are at distance ε. Otherwise, we use the fact that ς is convex (Lemma 2). This means that on $I^- = (-\infty, t_{\min}]$ it is monotonically decreasing, and on $I^+ = [t_{\min}, \infty)$ it is monotonically increasing. Hence, there are at most two times t^- and t^+ such that $\varsigma(t) = \varepsilon$, and we have that $t^- \in I^-$ and $t^+ \in I^+$. We now find t^- and t^+ using parametric search [13]: t^- (t^+) is the smallest (largest) time in I^- (I^+) such that $\varsigma(t) \leq \varepsilon$. To actually find $t_{\min}$, we basically use the same approach. At $t_{\min}$ the derivative ς' of ς is zero. Since ς is convex, its derivative is monotonically increasing. Therefore, we can find $t_{\min}$ using a parametric search: $t_{\min}$ is the smallest time such that $\varsigma'(t) \geq 0$.

Finding the times $t_{\min}$, t^-, and t^+. We use parametric search [13] to find $t_{\min}$, t^-, and t^+. The global idea is as follows. For a more detailed description of parametric search and its application to our problem we refer to the full version of this paper.

To find $t_{\min}$ we use $\mathcal{P}(t) = \varsigma'(t) \geq 0$ as predicate. To find t^- and t^+ we use $\mathcal{P}(t) \leq \varepsilon$, and $\mathcal{P}(t) \geq \varepsilon$, respectively. In all these cases we need an algorithm $\mathcal{A}$ that can test $\mathcal{P}(t)$ for a given time t. This means that we need an efficient algorithm to compute $\varsigma(t)$ and a functional description of ς. To this end, we preprocess the input polygon for shortest path queries. We triangulate the polygon in $O(m)$ time [5], and build the data structure $\mathcal{D}$ of Guibas and Hershberger [8]. This also takes $O(m)$ time, and allows us to find the length of the shortest path between two fixed points p and q in $O(\log m)$ time. In particular, this means that for a given time t, we can compute $\varsigma(t)$ and $\varsigma'(t)$ in $O(\log m)$ time.

A query, and thus our algorithm $\mathcal{A}$, takes $O(\log m)$ time. It now immediately follows that we can compute $t_{\min}$, t^-, and t^+ in $O(\log^2 m)$ time each. We obtain the following result.

▶ **Lemma 4.** *Let $\mathcal{X}$ be a set of n entities, each moving in a simple polygon along a piecewise linear trajectory with τ vertices. We can compute all ε-events in $O(\tau n^2 \log^2 m + m)$ time, where m is the number of vertices in the polygon.*

To compute $\mathcal{R}$ we can now use the algorithm as described by Buchin et al. [2]. This algorithm maintains the connected components in a dynamic graph G; at each ε-event we insert or delete an edge in G. This takes $O(\log n)$ time per ε-event, and thus $O(\tau n^2 \log n)$ time in total [2, 14]. We conclude:

▶ **Theorem 5.** *Let $\mathcal{X}$ be a set of n entities, each moving in a simple polygon along a piecewise linear trajectory with τ vertices. The Reeb graph $\mathcal{R}$ representing the movement of the entities in $\mathcal{X}$ has size $O(\tau n^2)$ and can be computed in $O(\tau n^2(\log^2 m + \log n) + m)$ time, where m is the number of vertices in the polygon.*

4 Well-spaced Obstacles

Next, we consider the situation where the entities move in a domain with multiple polygonal obstacles. We first assume that the obstacles are well-spaced, that is, the distance between any pair of non-adjacent obstacle edges is at least ε.

4.1 Lower Bound

▶ **Lemma 6.** *The total number of critical events for a set of n entities, each moving amidst a set of well-spaced obstacles $\mathcal{O}$ along a piecewise linear trajectory with τ vertices, is $\Omega(\tau(n^2 + nm))$, where m is the total complexity of $\mathcal{O}$.*

Proof. We describe a construction in which the entities move along lines that yields $\Omega(nm)$ critical events. We repeat this construction in $\Omega(\tau)$ steps. Since we already have a $\Omega(\tau n^2)$ lower bound for entities moving in $\mathbb{R}^2$ without obstacles, the lemma then follows.

The construction that we use is sketched in Fig. 2. We have two horizontal lines ℓ_A and ℓ_B that are within vertical distance ε of each other. Our obstacles essentially form a wall separating the two lines that has $\Theta(m)$ openings. Each obstacle is triangular, and thus well-spaced. Furthermore, the obstacles are at distance at least ε from each other, so $\mathcal{O}$ is well-spaced. Our set of entities consists of two equal-sized subsets A and B. The entities move in pairs; one entity a from A and one entity b from B. Throughout the movement they maintain $a_x = b_x$, and stay far away from any other entities. It is easy to see that this yields $\Omega(nm)$ critical events as desired. ◀

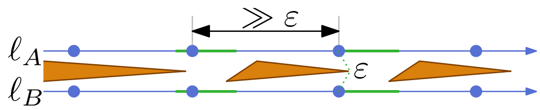

Figure 2 The lower bound construction for well spaced obstacles. The entities of a pair a, b are within distance ε from each other when both move in a green interval.

4.2 Upper Bound

In this case our obstacles are well spaced, so if two entities are at geodesic distance ε the geodesic consists of at most two line segments. We now start with some bounds on the total number of ε-events.

▶ **Observation 7.** *There are at most $O(\tau n^2)$ ε-events where the geodesic between the two entities involved is a single line segment.*

▶ **Lemma 8.** *Let $\mathcal{X}$ be a set of n entities, each moving amidst a set of well-spaced obstacles $\mathcal{O}$ along a piecewise linear trajectory with τ vertices. The number of ε-events is at most $O(\tau n^2 m)$, where m is the total complexity of $\mathcal{O}$.*

Proof. By Observation 7 there are only $O(\tau n^2)$ ε-events in which the geodesic is a single line segment. We now bound the number of ε-events for which the geodesic contains an obstacle vertex v by $O(\tau n^2)$. The lemma then follows. Fix two entities a and b. Each trajectory edge intersects the ε-disk centered at v at most once. Hence, there are $O(\tau)$ time intervals during which both a and b move along a line, and are within distance ε from v. Clearly, all ε-events occur within one of these intervals. Since the obstacles are well spaced, the ε-disk contains at most three edges: the two edges connected to v and at most one edge adjacent to both these edges. It follows that the function ς_{ab} consists of at most $O(1)$ pieces during such an interval. Hence, there can be at most a constant number of ε-events per interval. ◀

Next, we show that the number of critical events can only be $O(\tau(n^2 + m\lambda_4(n)))$. Clearly, the number of critical events at which the geodesic is a single line segment is also at most $O(\tau n^2)$ (Observation 7). We now bound the number of critical events where two sets of entities become ε-connected or ε-disconnected, and the geodesic between them consists of two line segments, connected via an obstacle vertex v.

Let $\mathcal{D}$ be the disk of radius ε centered at v, and consider a subdivision of $\mathcal{D}$ into six equal size sectors or *wedges*. See Fig. 3. We make sure that the obstacle containing v intersects at least two wedges. Let W be such a wedge. For any pair of points p and q in W, the Euclidean distance between p and q is at most ε. Let $\mathcal{X}_W(t) \subseteq \mathcal{X}$ denote the set of entities that that lie in W at time t.

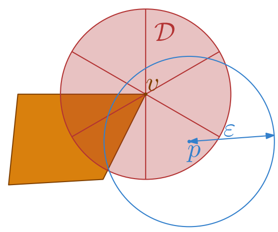

Figure 3 The ε-disk $\mathcal{D}$ (red) centered at v subdivided into six wedges. The distance between any pair of points p and q in the same wedge is at most ε.

▶ **Observation 9.** *At any time t, there is at most one maximal set of ε-connected entities G that has entities in wedge W, that is, for which $G \cap \mathcal{X}_W(t) \neq \emptyset$.*

▶ **Corollary 10.** *At any time t, there is at most one maximal set of ε-connected entities G such that $\mathcal{X}_W(t) \subseteq G$.*

When two maximal sets of ε-connected entities $\mathcal{X}_R$ and $\mathcal{X}_B$ become ε-connected or ε-disconnected at time t via vertex v, then the entities $r \in \mathcal{X}_R$ and $b \in \mathcal{X}_B$ that form their

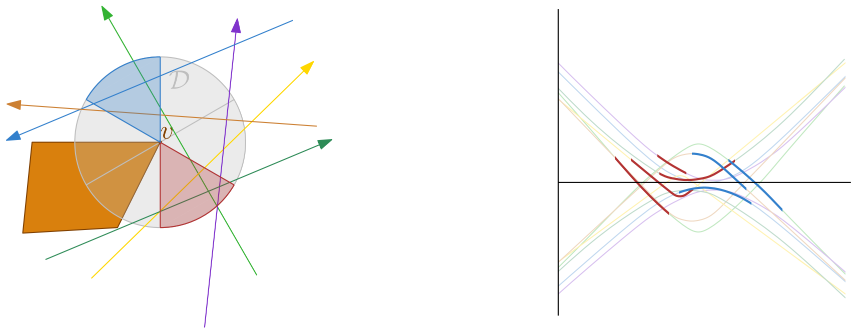

Figure 4 A set of entities on the left, and the corresponding sets of partial functions $\mathcal{R}$ (red) and $\mathcal{B}$ (blue). Critical events correspond to intersections between the lower envelope of $\mathcal{R}$ and the upper envelope of $\mathcal{B}$.

closest pair must both lie in $\mathcal{D}$ at time t. More specifically, since the geodesic between r and b uses vertex v, r and b must lie in different wedges. Let R and B denote the wedges that contain r and b, respectively. We now show that the total number of critical events involving entities in wedges R and B is $O(\tau\lambda_4(n))$. By Corollary 10 it then follows that each such event corresponds to exactly one pair of ε-connected sets. Since there are only 15 pairs of wedges, there are also at most $O(\tau\lambda_4(n))$ times when two maximal sets of ε-connected entities are at distance exactly ε and are connected via vertex v.

▶ **Lemma 11.** *The total number of critical events involving entities in wedges R and B is $O(\tau\lambda_4(n))$.*

Proof. Given an entity $a \in \mathcal{X}$ we define two partial functions ϱ_a and β_a as follows:

$$\varrho_a(t) = \begin{cases} \xi_{av}(t) - \varepsilon/2 & \text{if } a \in \mathcal{X}_R(t) \\ \bot & \text{otherwise,} \end{cases} \qquad \beta_a(t) = \begin{cases} -\xi_{av}(t) + \varepsilon/2 & \text{if } a \in \mathcal{X}_B(t) \\ \bot & \text{otherwise,} \end{cases}$$

where $\bot$ denotes undefined. Furthermore, let $\mathcal{R} = \{\varrho_r \mid r \in \mathcal{X}\}$ and $\mathcal{B} = \{\beta_b \mid b \in \mathcal{X}\}$. See Fig. 4. It now follows that for any two entities $r \in \mathcal{X}_R(t)$ and $b \in \mathcal{X}_B(t)$ the length of the path from r via v to b is ε if and only if $\varrho_r(t) = \beta_b(t)$. Thus, the number of times entities in R become ε-connected or ε-disconnected via vertex v is at most the number of intersection points between the lower envelope of $\mathcal{R}$ and the upper envelope of $\mathcal{B}$. Next, we show that this number of intersection points is at most $O(\tau\lambda_4(n))$.

Each trajectory consists of $\tau - 1$ edges, each of which intersects wedge R in a single line segment. Hence, for each entity a, the function ϱ_a is defined on at most $\tau - 1$ maximal contiguous intervals $I_a^1, .., I_a^{\tau-1}$. Thus, by Lemma 1 the lower envelope $\mathcal{L}$ of $\mathcal{R}$ has complexity at most $O(\tau\lambda_4(n))$. Similarly, the upper envelope $\mathcal{U}$ of $\mathcal{B}$ has complexity $O(\tau\lambda_4(n))$. It follows that there are also $O(\tau\lambda_4(n))$ time intervals such that both $\mathcal{L}$ and $\mathcal{U}$ are represented by a simple hyperbolic function. In each such interval $\mathcal{L}$ and $\mathcal{U}$ intersect each other at most twice. Hence, the total number of intersection points is $O(\tau\lambda_4(n))$. ◀

It now follows that the total number of critical events at which the geodesic contains an obstacle vertex is $O(m\tau\lambda_4(n))$. We conclude:

▶ **Theorem 12.** *Let $\mathcal{X}$ be a set of n entities, each moving amidst a set of well-spaced obstacles $\mathcal{O}$ along a piecewise linear trajectory with τ vertices. The number of critical events is at most $O(\tau(n^2 + m\lambda_4(n)))$, where m is the total complexity of $\mathcal{O}$.*

4.3 Algorithm

We now show how to compute the Reeb graph $\mathcal{R}$ in case the entities move among well-spaced obstacles. At first glance, it seems that we can compute all critical events using the same approach as used in the upper bound proof. Indeed, this allows us to find all times at which critical events occur. However, to construct the Reeb graph we also need to know the sets of entities involved at each critical event, e.g. we want to know that a set $\mathcal{X}'$ splits into subsets R and B. Unfortunately, there does not seem to be an efficient, i.e. sub-linear, way to obtain this information, nor can we easily maintain the ε-connected sets of entities without considering all ε-events. It is easy to compute all ε-events in $O(\tau n^2 m)$ time, using the approach described in Lemma 8. Once we have computed all ε-events, we can construct the Reeb graph using the same method described by Buchin et al. [2]. This takes $O(\log n)$ time per ε-event. Thus, we conclude:

▶ **Theorem 13.** *Let $\mathcal{X}$ be a set of n entities, each moving amidst a set of well-spaced obstacles $\mathbb{O}$ along a piecewise linear trajectory with τ vertices. The Reeb graph $\mathcal{R}$ representing the movement of the entities in $\mathcal{X}$ has size $O(\tau(n^2 + m\lambda_4(n)))$ and can be computed in $O(\tau n^2 m \log n)$ time, where m is the total complexity of $\mathbb{O}$.*

5 General Obstacles

Finally, we study the most general case in which the entities move amidst multiple obstacles, and there are no restrictions on the locations, shape, or size of the obstacles.

5.1 Lower Bound

▶ **Lemma 14.** *The total number of critical events for a set of n entities, each moving amidst a set of obstacles $\mathbb{O}$ along a piecewise linear trajectory with τ vertices, is $\Omega(\tau(n^2 + nm\min\{n, m\}))$, where m is the total complexity of $\mathbb{O}$.*

Proof. We describe a construction in which the entities move along lines that yields $\Omega(nmk)$ critical events, with $k = \min\{n, m\}$. We again repeat this construction $\Omega(\tau)$ times.

The basic idea is to create $\Omega(k)$ stationary entities, $\Omega(n)$ moving entities, and $\Omega(m)$ "entrances" from which a moving entity can become connected with a stationary entity. Each stationary entity is surrounded by an obstacle. The distance from such a stationary entity s to an entrance leading to s, will be approximately ε. So an entity gets ε-connected with s only if it is directly in front of the entrance. We make sure that each stationary entity is reachable from all entrances. Hence, each time that one of the $\Omega(n)$ moving entities passes an entrance it will generate $\Omega(k)$ critical events. Since all $\Omega(n)$ moving entities encounter all $\Omega(m)$ entrances we get at least $\Omega(nmk)$ critical events as desired.

Let c be a point in the plane, let $\delta > 0$ be a small value, and let P be a set of $\Omega(k)$ points on the lower half of the circle with center c and radius $\varepsilon/2 + \delta$. We place a large rectangular obstacle $\mathcal{O}$ containing c and all points in P such that the (shortest) distance from c to the top side h of O is smaller than $\varepsilon/2 - \delta$. See Fig. 5(a).

We now carve $\Omega(k + m)$ passages through $\mathcal{O}$. The first $k' = \Omega(k)$ connect c to each point in P. The remaining $m' = \Omega(m)$ connect c to the top side h of the obstacle $\mathcal{O}$. The first k' passages all have length exactly $\varepsilon/2 + \delta$, and we make sure that the remaining m' passages all have length exactly $\varepsilon/2 - \delta$. We can do this with at most one bend in each passage. See Fig. 5(a). The distance from any point in P to the top side of $\mathcal{O}$, via any of the m' passages, is now ε, and the distance between any two points in P is strictly larger than ε.

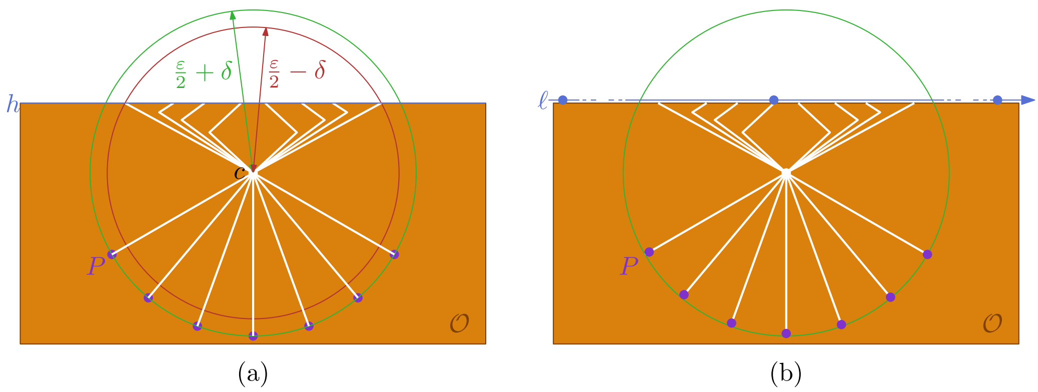

Figure 5 The lower bound construction for general obstacles. (a) Constructing the passages through obstacle $\mathcal{O}$. (b) The final construction.

We place a stationary entity on each point in P, and we let $\Omega(n)$ entities move from left to right on a horizontal line ℓ containing h (we can move ℓ upwards a bit later to make sure the entities do not intersect the obstacle). We make sure that at any time the distance between two of these moving entities is larger than ε, so they are never in the same ε-connected set. When an entity e arrives at an entrance, that is, an opening of one of the top passages, it is at distance ε to the points in P. Hence, we have a critical event where e connects with all entities at points in P. We can make sure that e generates an event with (the entity on) *each* point in P by moving each point in P by a small unique amount towards c. Fig. 5(b) shows the resulting construction. ◀

5.2 Upper Bound

We again start by bounding the total number of ε-events.

▶ **Lemma 15.** *Let $\mathcal{X}$ be a set of n entities, each moving amidst a set of obstacles $\mathcal{O}$ along a piecewise linear trajectory with τ vertices. The number of ε-events is at most $O(\tau n^2 m^2)$, where m is the total complexity of $\mathcal{O}$.*

As in the case of well-spaced obstacles, ε-events are not necessarily critical events. We now fix an obstacle vertex v, and show that there are at most $O(\tau m^2 \lambda_4(n))$ critical events involving v. To this end, we again decompose the (geodesic) ε-disk centered at v into regions such that each region corresponds to at most one maximal set of ε-connected entities. Each critical event involving v also involves two maximal ε-connected sets, and thus two regions in this decomposition. We show that we have to consider only $O(m)$ pairs of such regions, and that for each pair there can be at most $O(\tau m \lambda_4(n))$ critical events. Since we have $O(m)$ obstacle vertices this gives us a total bound of $O(\tau m^3 \lambda_4(n))$. When m is at most $O(n^2/\lambda_4(n))$, this is an improvement over the bound in Lemma 15. It follows that the total number of critical events is thus at most $O(\tau \min\{n^2 + m^3\lambda_4(n), n^2m^2\})$.

Let $\mathcal{D}_\varepsilon$ denote the geodesic ε-disk centered at v, and let $\mathcal{D}_{\varepsilon/2}$ denote the geodesic $(\varepsilon/2)$-disk centered at v. Clearly, the geodesic distance between any two points in $\mathcal{D}_{\varepsilon/2}$ is at most ε, thus we observe:

▶ **Observation 16.** *At any time t there is at most one maximal ε-connected set of entities G such that $G_{\mathcal{D}_{\varepsilon/2}}(t) \neq \emptyset$, and thus $\mathcal{X}_{\mathcal{D}_{\varepsilon/2}}(t) \subseteq G$.*

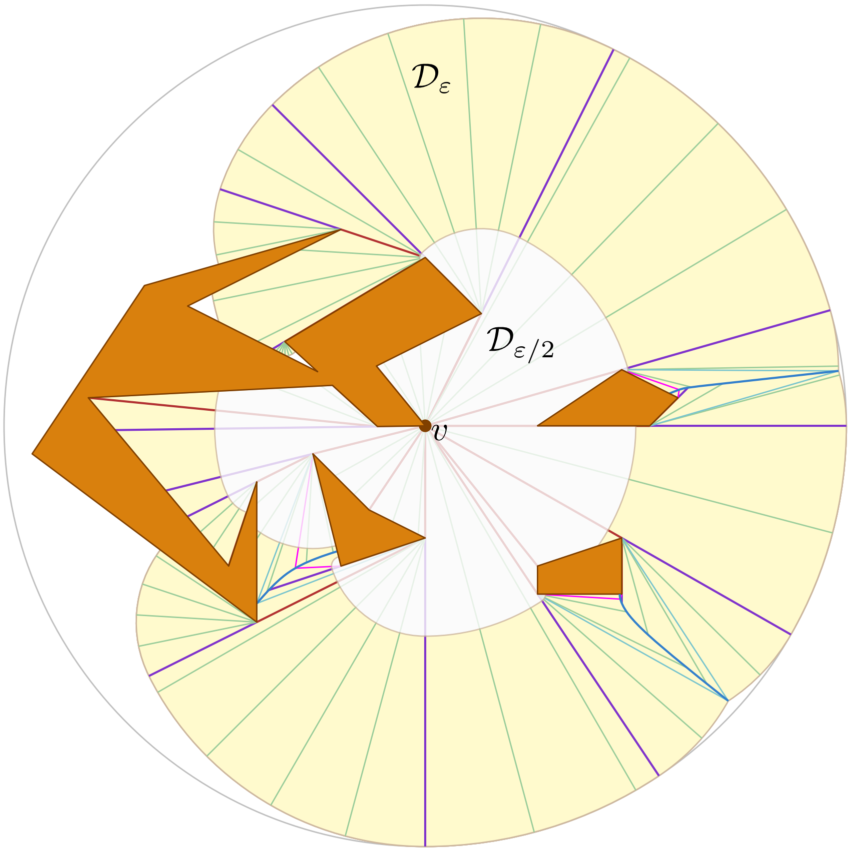

Figure 6 Subdivision Φ. The color of the edge indicates its type: the red edges originate from shortest paths, the purple and blue edges from the shortest path map, the cyan edges from the subdivision in "triangular sectors", the light green edges guarantee that the maximum angle at the routing point is at most $\pi/12$, and the pink edges guarantee monotonicity.

Let $\mathcal{A} = \mathcal{D}_\varepsilon \setminus \mathcal{D}_{\varepsilon/2}$. We decompose $\mathcal{A}$ into $O(m)$ regions such that for each region R we have that (i) the geodesic distance between two points $p, q \in R$ is at most ε, (ii) any two points $p, q \in R$ have the same (combinatorial) geodesic to v, and (iii) the boundary of R has constant complexity.

Let Φ denote this decomposition of $\mathcal{A}$. It follows that at any time, each region R in Φ contains entities from at most one maximal ε-connected set G. That is, $\mathcal{X}_R(t) \subseteq G$. It is now easy to see that any critical event involving v involves the maximal set of ε-connected entities $G_{\varepsilon/2}$ corresponding to $\mathcal{D}_{\varepsilon/2}$, and a maximal set of ε-connected entities G_R corresponding to a region R of Φ. Hence, there are only $O(m)$ pairs of regions that can be associated with a critical event involving v. We now show how to construct Φ, and how to bound the number of events corresponding to a single pair of regions.

Obtaining subdivision Φ. Let Φ' be the overlay of the shortest path map with root v (restricted to $\mathcal{D}_\varepsilon$), and all shortest paths from v to obstacle vertices in $\mathcal{D}_\varepsilon$.

▶ **Observation 17.** *Φ' has complexity $O(m)$.*

The edges of Φ' are either line segments or hyperbolic arcs [9]. Since Φ' is a refinement of the shortest path map, all points in a region R in Φ' have the same geodesic g to v (except for the starting edge). Hence, each region R is star-shaped, and has a vertex c that lies inside the kernel. This vertex c is the second vertex on each geodesic g. We refer to c as the *routing point* of R.

Next, we further subdivide each region R in Φ'. We add edges $\overline{cu}$ between the routing point c and all boundary vertices u of R. Each region is now bounded by two line segments $\overline{cu}$ and $\overline{cw}$ and a segment $\widetilde{cw}$. The segment $\widetilde{cw}$ is either a line segment, or a hyperbolic arc. We further add edges $\overline{cz}$ between c and points z on $\widetilde{cw}$ such that the angle at c is at most $\theta = \pi/12$. In case $\widetilde{cw}$ is a hyperbolic arc we make sure that the hyperbolic function describing this arc is monotonic. To this end, we add at most one additional edge $\overline{cz}$ to the point z on $\widetilde{cw}$ with maximum curvature. All these new edges are contained in R and do not intersect each other. It follows that the total complexity, summed over all regions in the subdivision, is still $O(m)$. Let Φ denote the resulting subdivision, restricted to $\mathcal{A}$. See Fig. 6.

▶ **Lemma 18.** *Let R be a region in Φ. For any two points $p, q \in R$ the Euclidean distance $\|pq\|$ between p and q is at most $\varepsilon\sqrt{29/4 - 4\sqrt{3}}$.*

▶ **Lemma 19.** *Let R be a region in Φ. For any two points $p, q \in R$ the geodesic distance $\varsigma(p,q) = \varsigma(g(p,q))$ between p and q is at most ε.*

▶ **Lemma 20.** *Subdivision Φ has complexity $O(m)$ and each region $R \in \Phi$ has the following properties:*

(i) *the geodesic distance between two points $p, q \in R$ is at most ε,*
(ii) *any two points $p, q \in R$ have the same geodesic to v (excluding the starting edge), and*
(iii) *the boundary of R has constant complexity.*

Proof. Property (i) follows directly from Lemma 19, and Property (ii) follows from the fact that Φ is a refinement of the shortest path map. Each region is bounded by three or four segments, depending if the routing point c lies in $\mathcal{A}$ or not. If $c \in \mathcal{A}$, region R is bounded by three segments. Otherwise, R is bounded by three segments and a part of $D_{\varepsilon/2}$. However, as all shortest paths from points in R to v use point c, it follows that this part of $D_{\varepsilon/2}$ is also a single hyperbolic segment. This proves Property (iii). ◀

Bounding the number of critical events for a pair of regions. Next, we fix a region R in Φ, and show that the number of critical events involving v, R, and $D_{\varepsilon/2}$, is at most $O(\tau\lambda_4(n))$.

▶ **Lemma 21.** *Let R be any region of Φ, and let G_R be the maximal set of ε-connected entities corresponding to R. The (geodesic) distance between G_R and v is given by a piecewise hyperbolic function with $O(\tau\lambda_4(n))$ pieces.*

Proof. The boundary of R has constant complexity, so each entity in G_R intersects region R in $O(\tau)$ time-intervals. Furthermore, all points in R have the same combinatorial geodesic, so during any such an interval, the distance to v is given by a simple hyperbolic function. Thus, the distance function between G_R and v corresponds to the lower envelope of a set of hyperbolic functions. Lemma 1 now completes the proof. ◀

Fix a region R, let

$$\beta_a(t) = \begin{cases} -\varsigma(a(t), v) + \varepsilon & \text{if } a(t) \in R \\ \bot & \text{otherwise,} \end{cases}$$

and let $\mathcal{U}$ be the upper envelope of $\{\beta_a(t) \mid a \in \mathcal{X}\}$. It follows from Lemma 21 that $\mathcal{U}$ has complexity $O(\tau\lambda_4(n))$.

Now consider the entities in the inner region $\mathcal{D}_{\varepsilon/2}$. The function ς_{av} expressing the geodesic distance between a and v is piecewise hyperbolic and consists of $O(m\tau)$ pieces. Let $\mathcal{L}$ denote the lower envelope of all functions ϱ_a, $a \in \mathcal{X}$, where $\varrho_a(t) = \varsigma_{av}(t)$ if $\varsigma_{av}(t) \leq \varepsilon/2$ and $\perp$ otherwise. It follows from Lemma 1 that $\mathcal{L}$ has complexity $O(m\tau\lambda_4(n))$.

As with the well-spaced obstacles, all critical events in which the entities involved lie in $\mathcal{D}_{\varepsilon/2}$ and R at the time of the event correspond to intersections of $\mathcal{L}$ and $\mathcal{U}$. To bound the number of intersections, and thus the number of critical events, we now (again) partition the domain of $\mathcal{L}$ and $\mathcal{U}$ (i.e., time) into sets $D_1, .., D_k$ such that in each D_i the lower envelope $\mathcal{L}$ and the upper envelope $\mathcal{U}$ intersect at most twice. It is easy to partition the domain into $k = O(|\mathcal{L}| + |\mathcal{U}|) = O(\tau\lambda_4(n) + m\tau\lambda_4(n)) = O(m\tau\lambda_4(n))$ intervals with this property. Hence, we get $O(m\tau\lambda_4(n))$ critical events involving vertex v and the pair of regions $(R, \mathcal{D}_{\varepsilon/2})$. This gives a total of $O(m^3\tau\lambda_4(n))$ critical events. Together with the bound on the number of ε-events (Lemma 15) this gives us the following result:

▶ **Theorem 22.** *Let $\mathcal{X}$ be a set of n entities, each moving amidst a set of obstacles $\mathcal{O}$ along a piecewise linear trajectory with τ vertices. The number of critical events is at most $O(\tau \min\{n^2 + m^3\lambda_4(n), n^2m^2\})$, where m is the total complexity of $\mathcal{O}$.*

5.3 Algorithm

We again explicitly compute all ε-events in order to construct the Reeb graph $\mathcal{R}$. We follow the approach from Lemma 15. That is, we compute the shortest path map Ψ with root v, and for each pair of entities a and b we trace their trajectories through Ψ. For each of the $O(\tau m)$ pairs of regions visited, we construct ς_{ab} and find the ε-events. Computing the shortest path map with root v takes $O(m \log m)$ time [9]. Tracing the trajectories and computing the distance functions takes time proportional to the number of regions visited. Hence, we spend $O(\tau m)$ time for each pair. It follows that the total time required to compute all ε-events is $O(m(m \log m + n^2\tau m)) = O(\tau n^2m^2 + m^2 \log m)$. Computing $\mathcal{R}$ again takes $O(\log n)$ time per ε-event. We obtain the following result.

▶ **Theorem 23.** *Let $\mathcal{X}$ be a set of n entities, each moving amidst a set of obstacles $\mathcal{O}$ along a piecewise linear trajectory with τ vertices. The Reeb graph $\mathcal{R}$ representing the movement of the entities in $\mathcal{X}$ has size $O(\tau \min\{n^2 + m^3\lambda_4(n), n^2m^2\})$ and can be computed in $O(\tau n^2m^2 \log n + m^2 \log m)$ time, where m is the total complexity of $\mathcal{O}$.*

6 Concluding Remarks

We study the trajectory grouping structure for entities moving amidst obstacles. To this end, we analyze the number of times when two sets of entities are at distance ε from each other. Our results for various types of obstacles can be found in Table 1. These bounds on the number of critical events also give a bound on the size of the Reeb graph $\mathcal{R}$. This in turn gives bounds on the number of maximal groups: if the Reeb graph has size $O(|\mathcal{R}|)$ there are $O(|\mathcal{R}|n)$ maximal groups [2]. Furthermore, we present efficient algorithms to compute $\mathcal{R}$, which leads to efficient algorithms to compute the grouping structure.

One intruiging open question is whether the Reeb graph can be constructed using only the critical events, that is, in an output-sensitive manner. The difficulty with the approach as described in [2] appears to be that one would need a dynamic data structure for maintaining a subdivision of a set (the groups), that supports efficient split and merge operations. Thus, there may be fundamental graph-theoretical obstacles to this approach. However, it is not clear that this is the only possible approach to compute $\mathcal{R}$.

An other direction of future work is to extend the grouping structure for entities moving in more realistic environments, for instance modeled by weighted regions. This starts with interesting modeling questions since distances are related to the speed of the entities. For example: should the distance for two entities, say sheep, to be directly connected be larger on a muddy field than it is on a concrete courtyard, or do the sheep need to be closer together in the field to be considered a group?

Although we developed the technical machinery in this paper with the goal of extending the trajectory grouping structure, we foresee wider applications for our techniques. We believe our work will serve as a starting point for more general research related to moving entities and geodesic distances. For example, we can consider trajectory similarity measures in the presence of obstacles.

Acknowledgments. M. L., F. S., I. K., and B. S. are supported by the Netherlands Organisation for Scientific Research (NWO) under grants 639.021.123, 612.001.022, 612.001.106, and 639.023.208 respectively.

References

1. Marc Benkert, Joachim Gudmundsson, Florian Hübner, and Thomas Wolle. Reporting flock patterns. *Computational Geometry*, 41(3):111–125, 2008.
2. Kevin Buchin, Maike Buchin, Marc van Kreveld, Bettina Speckmann, and Frank Staals. Trajectory grouping structure. In *Proc. 2013 WADS Algorithms and Data Structures Symposium*, LNCS, pages 219–230. Springer, 2013.
3. Maike Buchin, Somayeh Dodge, and Bettina Speckmann. Context-aware similarity of trajectories. In *Geographic Information Science*, volume 7478 of *LNCS*, pages 43–56. Springer, 2012.
4. Maike Buchin, Anne Driemel, and Bettina Speckmann. Computing the Fréchet distance with shortcuts is NP-hard. In *Symposium on Computational Geometry*, page 367. ACM, 2014.
5. Bernard Chazelle. Triangulating a simple polygon in linear time. *Discrete Comput. Geom.*, 6(5):485–524, 1991.
6. Herbert Edelsbrunner and John L. Harer. *Computational Topology – an introduction.* American Mathematical Society, 2010.
7. Joachim Gudmundsson and Marc van Kreveld. Computing longest duration flocks in trajectory data. In *Proc. 14th ACM International Symposium on Advances in Geographic Information Systems*, GIS'06, pages 35–42. ACM, 2006.
8. Leonidas J. Guibas and John Hershberger. Optimal shortest path queries in a simple polygon. *Journal of Computer and System Sciences*, 39(2):126–152, 1989.
9. John Hershberger and Subhash Suri. An Optimal Algorithm for Euclidean Shortest Paths in the Plane. *SIAM Journal on Computing*, 28(6):2215–2256, 1999.
10. Hoyoung Jeung, Man Lung Yiu, Xiaofang Zhou, Christian S. Jensen, and Heng Tao Shen. Discovery of convoys in trajectory databases. *PVLDB*, 1:1068–1080, 2008.
11. Panos Kalnis, Nikos Mamoulis, and Spiridon Bakiras. On discovering moving clusters in spatio-temporal data. In *Advances in Spatial and Temporal Databases*, volume 3633 of *LNCS*, pages 364–381. Springer, 2005.
12. Patrick Laube, Marc van Kreveld, and Stephan Imfeld. Finding REMO – detecting relative motion patterns in geospatial lifelines. In *Developments in Spatial Data Handling*, pages 201–215. Springer, 2005.

13 Nimrod Megiddo. Applying parallel computation algorithms in the design of serial algorithms. *J. ACM*, 30(4):852–865, 1983.
14 Salman Parsa. A deterministic $O(m \log m)$ time algorithm for the Reeb graph. In *Proc. 28th ACM Symposium on Computational Geometry*, pages 269–276, 2012.