International Journal of Computational Geometry & Applications

AN OPTIMAL MORPHING BETWEEN POLYLINES

SERGEI BESPAMYATNIKH*
Department of Computer Science, Duke University, Box 90129, Durham, NC 27708, USA

Received 07 May 2001
Revised 01 November 2001
Communicated by (J. S. B. Mitchell)

We address the problem of continuously transforming or morphing one simple polyline into another so that every point p of the initial polyline moves to a point q of the final polyline using the geodesic shortest path from p to q. We optimize the width of the morphing, that is, the longest path made by a point on the polyline. We present an algorithm for finding the minimum width morphing in $O(n^2)$ time using $O(n)$ space, where n is the total number of vertices of polylines. This improves the previous algorithm [7] by a factor of $\log^2 n$.

Keywords: Morphing, polylines, skeleton, shortest path.

1. Introduction

In computer graphics and computer vision a recent area of interest has been *morphing*, i.e., continuously transforming one shape into another. Morphing algorithms have numerous uses in shape interpolation, animation, video compression [1, 4, 7, 8, 11, 12]. In general there are numerous ways [2, 3, 7] to interpolate between two shapes. There are different criteria for the quality of a morph and notions of optimality. We consider the following morphing problem. Let α and β be two shapes which are polylines. A polyline is a finite sequence of segments in the plane that are connected into a simple curve. We want to produce a continuous transformation of the polyline α to the polyline β. Rendering this transformation is an animation problem.

Guibas *et al.* [8] considered the problem when α and β are *parallel* polygons with the same number of vertices, i.e., polygons with the same sequence of angles. They showed that any two parallel polygons can be morphed into one another such that every interpolating polygon is also parallel to the initial and final polygons. They also showed that the morphing transformation can be computed in $O(n \log n)$ time.

*besp@cs.duke.edu

Recently Efrat *et al.* [7] considered morphing between polylines such that the interpolating polyline is always between bounding polylines and the objective is to minimize the longest path made by a point moving from one polyline to another. They obtained an algorithm with $O(n^2 \log^2 n)$ running time and $O(n^2)$ space. Their algorithm is based on parametric searching and therefore is rather complicated to implement.

The main result of this paper is that an optimal morphing between polylines can be obtained by a simpler algorithm in $O(n^2)$ time using $O(n)$ space.

Efrat *et al.* [7] also presented a 2-factor approximation to the morphing width that can be computed in near linear time. Another variant of the morphing width, using a slightly different distance function, was previously investigated in [6]. Alt and Godau [2] considered the problem of computing the Fréchet distance between two polygonal curves. They show that the Fréchet distance can be computed in $O(mn)$ time. The Fréchet distance can be used to define a morph from one polyline to another. The Fréchet distance can be viewed as a simpler function than the width between polylines, which is minimized in our algorithm. Alt *et al.* [5] considered the matching of two polygonal curves under translations with respect to the Fréchet distance.

The paper is organized as follows. In Section we specify the problem. In Section we introduce a notion of matching skeleton and prove its properties. In Section we describe an algorithm and analyze its running time.

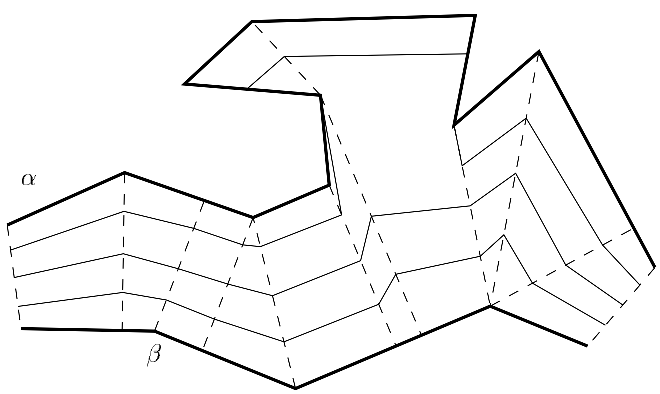


Fig. 1. Morphing between two polylines α and β.

2. Geometric Preliminaries

The initial and final polylines α and β can be represented as parameterized curves $f_\alpha, f_\beta : [0,1] \to \mathbb{R}^2$, i.e., the polyline α is the set of points $\{f_\alpha(u) \mid u \in [0,1]\}$ and the polyline β is the set of points $\{f_\beta(u) \mid u \in [0,1]\}$. The polylines α and β

are simple (without self-intersections) and oriented, i.e., the polyline α is oriented from $\alpha_0 = f_\alpha(0)$ to $\alpha_1 = f_\alpha(1)$ and the polyline β is oriented from $\beta_0 = f_\beta(0)$ to $\beta_1 = f_\beta(1)$. We assume that the parameterizations f_α and f_β are *monotone*, i.e., a point $f_\alpha(u_1)$ lies in the subpolyline of α from α_0 to $f_\alpha(u_2)$ if $0 \le u_1 \le u_2 \le 1$ (similarly for β).

A *morphing* is defined by a function $F : [0,1]^2 \to \mathbb{R}^2$ of two parameters $F(u,t)$ such that $F(u,0) = f_\alpha(u)$ and $F(u,1) = f_\beta(u)$ for any $u \in [0,1]$. The parameter t corresponds to time and, at any moment $t \in [0,1]$, the morphing function F defines the *interpolating polyline* $\gamma(t)$ which is the set of points $\{F(u,t)|u \in [0,1]\}$. Fixing another parameter u we obtain the *trace* of a point that moves from $f_\alpha(u)$ to $f_\beta(u)$. The points $f_\alpha(u)$ and $f_\beta(u)$ are called *matching points*. Every morphing function F induces the *matching* $M(F) = \{(f_\alpha(u), f_\beta(u)) \mid u \in [0,1]\}$.

We consider the morphing problem that is defined as follows. We assume that the polylines α and β are disjoint and, furthermore, the segments of α and β and two segments $[f_\alpha(0), f_\beta(0)]$ and $[f_\alpha(1), f_\beta(1)]$ form a simple polygon P, see Fig. 1. The morphing function F should satisfy the following property.

Trace property. The trace of each point is the shortest path in the polygon P from $f_\alpha(u)$ to $f_\beta(u)$ for some $u \in [0,1]$.

A morphing function satisfying the trace property is determined by the matching $M(F)$. For two points $p, q \in P$, let $\pi(p,q)$ be the shortest path from p to q that lies inside the polygon P. Let $d_P(p,q)$ denote the length of the path $\pi(p,q)$. We want to find a morphing function satisfying the trace property and minimizing the longest trace, which is called the *morphing width* [7]

$$W(F) = W(f_\alpha, f_\beta) = \max_{u \in [0,1]} \{d_P(f_\alpha(u), f_\beta(u))\}.$$

The width can be expressed in terms of matching

$$W(M) = \max_{(a,b) \in M} \{d_P(a,b)\}.$$

The input of the matching problem is the two polylines α and β, and the output is the minimum width morphing. The output can be expressed in terms of parametrizations f_α and f_β since the morphing function (and the matching) can be derived from them†. A more detailed representation of the optimal morphing is discussed in the next section.

3. Matching Skeleton

We define the notion of a *matching skeleton* that allows us to characterize an optimal morphing and efficiently compute it. Consider an arbitrary morphing function F

† We even can assume that one of the parametrizations is fixed. For example, f_β can be fixed to be the *arc-length parametrization*, i.e., $f_\beta(u)$ is the point of β so that the length of the subpolyline of β from $f_\beta(0)$ to $f_\beta(u)$ is $u|\beta|$ where $u \in [0,1]$ and $|\beta|$ is the length of β. Then the problem is to find a parametrization f_α minimizing the width $W(f_\alpha, f_\beta)$.

for the polylines α and β. Let a be a vertex of the polyline α. We define a *match* of a as

$$M(a) = \{b \in \beta \mid (a, b) \in M(F)\}.$$

Symmetrically we define the match of a vertex of β.

Observation 1 *The match of a vertex $a \in \alpha$ defined for any morphing F is a subpolyline of β.*

Proof. I □.t follows from the monotonicity of parameterization f_β.

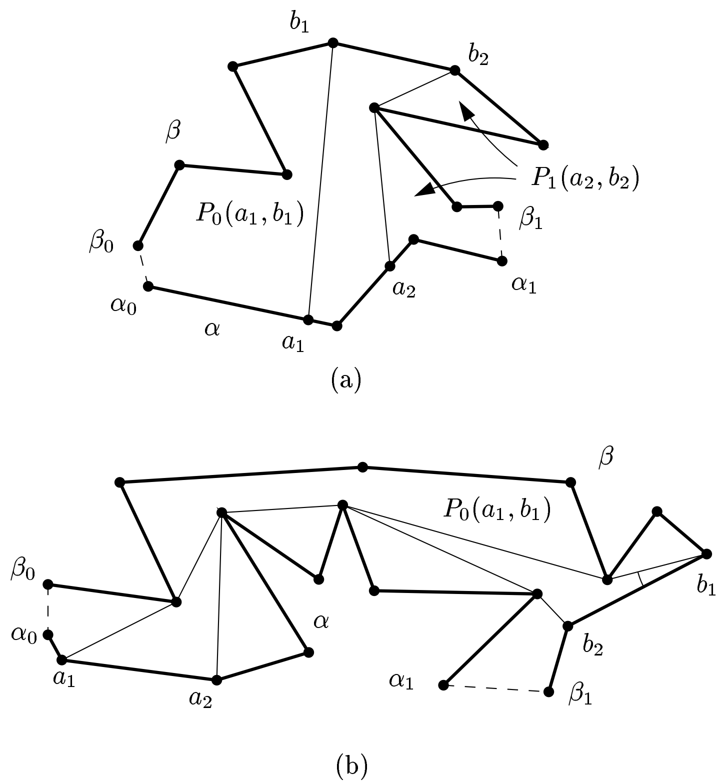


Fig. 2. Critical matches.

The shortest path $\pi(a, b)$, for $a \in \alpha$ and $b \in \beta$ splits the polygon P into two subpolygons. Here we admit a large class of polygons that is more general than the class of simple polygons. We denote by $P_i(a, b)$, $i \in \{0, 1\}$ the subpolygon containing the segment $[\alpha_i, \beta_i]$. We call a pair (a, b) a *critical match* if either a is a vertex of α and b is an endpoint of $M(a)$, or b is a vertex of β and a is an endpoint of $M(b)$. The monotonicity of parameterizations f_α and f_β implies that the shortest paths of critical matches (a_1, b_1) and (a_2, b_2) do not *cross*, i.e., either $P_0(a_1, b_1) \subseteq P_0(a_2, b_2)$

or $P_1(a_1, b_1) \subseteq P_1(a_2, b_2)$. In other words, two shortest paths $\pi(a_1, b_1)$ and $\pi(a_2, b_2)$ either (i) are disjoint or (ii) share common edges and can be drawn along opposite sides of common edges, see Fig. 2.

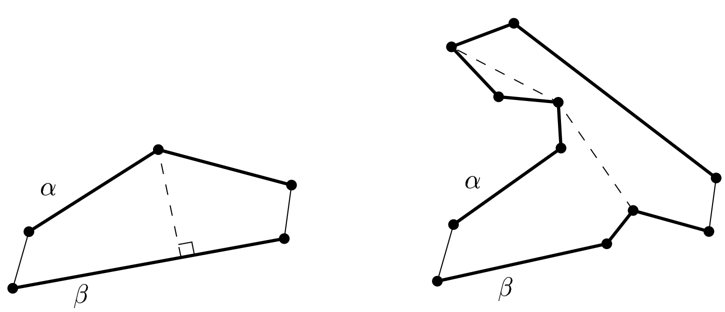
We define a *skeleton* $S(F)$ of a morphing function F as the union of the shortest paths of its critical matches. The main result of this Section is the following Theorem.

Theorem 1 *The width of an optimal morphing F is the length of a shortest path in its skeleton.*

Proof. L □.et n be the total number of vertices of the polylines α and β. There are at most $2n-2$ distinct critical matches (a_i, b_i), $i = 1, 2, \ldots, m$ in the skeleton S. Let $l(S)$ be the largest length of a shortest path in S, i.e., $l(S) = \max_{1 \leq i \leq m} d_P(a_i, b_i)$. Clearly, the width of the morphing F is greater than or equal to $l(S)$. We prove that $W(F) \leq l(S)$.

Let $Q_i, 1 \leq i < m$ be the polygon formed by two shortest paths $\pi(a_i, b_i)$ and $\pi(a_{i+1}, b_{i+1})$, i.e., $Q_i = P_0(a_{i+1}, b_{i+1}) \cap P_1(a_i, b_i)$. Notice that the part of the polyline α between points a_i and a_{i+1} is the segment $[a_i a_{i+1}]$. Similarly $[b_i b_{i+1}]$ is the part of the polyline β. It suffices to show that there is a morphing F_i of the polyline $a_i a_{i+1}$ into the polyline $b_i b_{i+1}$ inside Q_i such that $W(F_i) \leq \max(d_P(a_i, b_i), d_P(a_{i+1}, b_{i+1}))$. We prove it by induction on the number of vertices of Q_i.

Induction base. Obviously, the claim holds if Q_i contains two or three vertices.



(a) (b)

Fig. 3. Two cases of the minimum width morphing.

Induction step. Suppose that Q_i contains at least four vertices. Consider first the case $a_i = a_{i+1}$. We show that at least one of the angles of Q_i at vertices b_i or b_{i+1} is acute. Indeed, let p be the last common vertex of the paths $\pi(a_i, b_i)$ and $\pi(a_{i+1}, b_{i+1})$ and Q'_i be the simple polygon formed by the segment $[b_i, b_{i+1}]$ and the paths $\pi(p, b_i)$ and $\pi(p, b_{i+1})$. The polygon Q'_i has a funnel shape and all its vertices except p, b_i, and b_{i+1} are reflex. Let m be the number of vertices of Q_i'. The sum of the angles of Q_i' is $(m-2)180°$ and, thus, the sum of the angles at p, b_i, and b_{i+1} is at most $180°$. Therefore at least one of the angles at vertices b_i or b_{i+1} is acute. Without loss of generality suppose it is b_i — see Fig. 2(b) where the angle at b_1 is acute.

There is a sufficiently small $\varepsilon > 0$ such that replacing the vertex b_i by a point $p_\varepsilon = (1-\varepsilon)b_i + \varepsilon b_{i+1}$ preserves the number of vertices of Q_i and acute angle at p_ε. Clearly, this property does not hold for $\varepsilon = 1$ when $p_\varepsilon = b_{i+1}$. Hence there is some smallest $\varepsilon > 0$ such that either (i) the modified Q_i contains a smaller number of vertices, or (ii) the angle at p_ε is $90°$. We match a_i to all the points $[b_i, p_\varepsilon]$. Note that the distance from a_i to any of these points is at most $d_P(a_i, b_i)$. In the first case there is a morphing function from a_i to $[p_\varepsilon, b_{i+1}]$ by the induction assumption and we are done. In the second case we apply the symmetric approach for b_{i+1} if the corresponding angle is acute. Let $[b_i', b_{i+1}']$ be the reduced segment (i.e., $b_i' = p_\varepsilon$). The angles at b_i' and b_{i+1}' are right since the induction argument cannot be applied to both b_i' and b_{i+1}'. But one of these angles must be acute by the above argument if $b_i' \neq b_{i+1}'$. Hence $b_i' = b_{i+1}'$ and the polygon Q_i reduces to the path $\pi(a_i, b_i')$ completing the morph F_i. More important, it shows that $\pi(a_i, b_i')$ is shorter than both $\pi(a_i, b_i)$ and $\pi(a_i, b_{i+1})$.

In the general case $a_i \neq a_{i+1}$, the sum of the four angles at a_i, a_{i+1}, b_i and b_{i+1} is at most $360°$. At least one of these angles is acute and we apply the reduction described above. For example, if the angle at a_i is acute, we slide p_ε away from a_i toward a_{i+1} to shorten $\pi(b_i, p_\varepsilon)$. This leads to the case in which all four angles are right and, therefore, Q_i is a rectangle. Clearly, the translation of the segment $[a_i a_{i+1}]$ to $[b_i b_{i+1}]$ is the required morphing.

The proof of Theorem 1 implies the following Corollary, which also can be derived from the analysis of [7].

Corollary 1 *Let a be a point of α and b_1 and b_2 be two points of an edge of β. The shortest path in P from a to any point $b \in [b_1, b_2]$ has length at most $\max(d_P(a, b_1), d_P(a, b_2))$.*

Theorem 1 allows us to reduce the morphing problem to finding the minimum width skeleton.

Lemma 1 *There is a minimum width morph such that each critical match either*

(i) *is a pair of vertices from α and β, or*

(ii) *corresponds to the shortest distance from a vertex of a polyline to an edge of another polyline.*

Proof. S □.uppose that one of the critical matches, say (a, b), does not satisfy the conditions of the Lemma. Without loss of generality suppose a is a vertex of α. Then b is not a vertex of β. Let e be an edge of β containing b. The shortest path ab meets e with a non-right angle. If the polygon $P_0(a, b)$ has an acute angle at b, then we change the match (a, b) by moving b to the left by a sufficiently small amount, see Fig. 4(a). Note that we need to change preceding critical matches with endpoint b. If the polygon $P_0(a, b)$ has an obtuse angle at b, then we move b to the right, see Fig. 4(b). Again, some of the critical matches will be changed too. All the modified shortest paths reduce their lengths since they meet b with an acute angle.

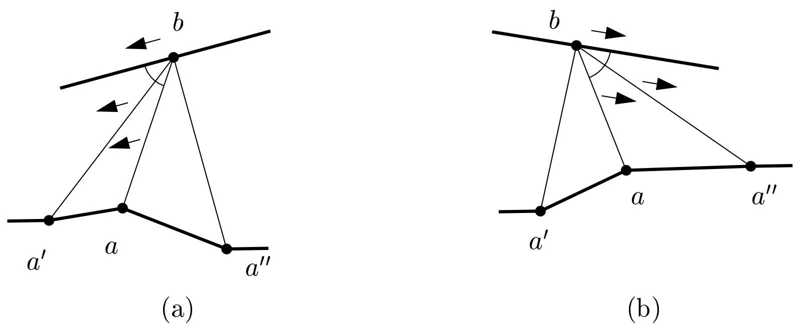


Fig. 4. Changing critical matches (a) $a'b$ and ab, and (b) ab and $a''b$.

Corollary 2 *The minimum width $W(M(F))$ over all morphing functions F is a shortest distance in P either between two vertices of P (one from α and another from β) or between a vertex and an edge of P (vertex from α (resp. β) and edge from β (resp. α)).*

By Corollary 2 there are $O(n^2)$ *candidate shortest paths* for an optimal skeleton. In the next Section we show that they can be found in $O(1)$ amortized time per path.

4. Algorithm

We show that the optimal matching and morph between two polylines α and β can be found in $O(n^2)$ time using dynamic programming.

Lemma 1 provides a class of morphs containing an optimal morph. A morph F satisfying the conditions of Lemma 1 can be described using $O(n)$ shortest paths $S(F)$ corresponding to the critical matches. Note that the total complexity of these paths, the number of segments in the skeleton, is linear since the shortest paths form a planar graph with $O(n)$ vertices. Lemma 1 allows us to use dynamic programming as follows. Let A be the set of pairs (a, b) such that one of the following conditions holds

- a is a vertex of α and b is a vertex of β, or
- a is a vertex of α, b is an edge of β and the shortest path from a to a point of b is achieved at an interior point of b, or

• a is an edge of α, b is a vertex of β and the shortest path from b to a point of a is achieved at an interior point of a.

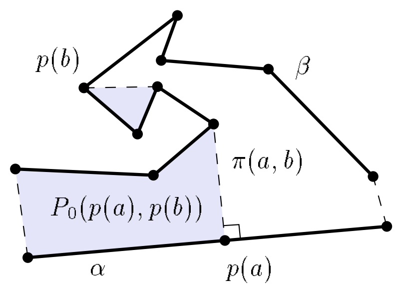


Fig. 5. Morph $F_{a,b}$ in shaded polygon.

Let $\pi(a,b)$ be the shortest path from a to b, $(a,b) \in A$. The dynamic program, for each pair $(a,b) \in A$, finds an optimal morph $F_{a,b}$ between parts of α and β truncated by endpoints of $\pi(a,b)$ such that the morph is restricted by the polygon P_0 that is cut off by $\pi(a,b)$, see Fig 5. More formally, let $p(a)$ and $p(b)$ be the endpoints of the shortest path $\pi(a,b)$. $F_{a,b}$ is an optimal morph of the polyline $\alpha_0 p(a)$ into the polyline $\beta_0 p(b)$ restricted by the polygon $P_0(p(a),p(b))$. Let $w(a,b),(a,b) \in A$ denote the width of the morph $F_{a,b}$. The algorithm spends $O(n^2)$ time although there are $\Theta(n^2)$ pairs in A and the shortest path of a pair $(a,b) \in A$ can have linear size.

For a morph F, the shortest paths of $S(F)$ do not cross and they can be sorted in order of increasing area of the polygon $P_0(p(a),p(b))$. We define $A(F)$ as the set of all matches of F that are also in A. Let $(a_0,b_0),(a_1,b_1),\ldots,(a_l,b_l)$ be the ordered sequence of pairs of $A(F)$. By Lemma 1 we can assume that $A(F)$ contains all the critical matches of F. We define a *partial order* on A, i.e., $(a,b) < (a',b')$ if $P_0(p(a),p(b)) \subseteq P_0(p(a'),p(b'))$. We say that (a_i,b_i) and (a_{i+1},b_{i+1}) satisfy the *transition property* if (a_i,b_i) is an immediate predecessor of (a_{i+1},b_{i+1}) in A, see Fig. 6. In order to compute the minimum width morph efficiently we prove the following Lemma.

Lemma 2 (Transition) *There is an optimal width morphing function F such that all consecutive pairs of $A(F)$ satisfy the transition property.*

Proof. L □.et F be an optimal morph satisfying the conditions of Lemma 1. Suppose that there is a vertex, say $a \in \alpha$, whose two critical matches from (a_i,b_i) to (a_{i+1},b_{i+1}), where $a_i = a_{i+1} = a$, do not make a transition. By Corollary 1 the pairs (a,b) of A, where b is "between" b_i and b_{i+1} (each b, b_i, b_{i+1} can be either a vertex or an edge), can be added to $A(F)$ without increasing its width. These pairs form a sequence satisfying the transition property.

If two consecutive pairs (a_i,b_i) and (a_{i+1},b_{i+1}) share a common edge, say $a =$

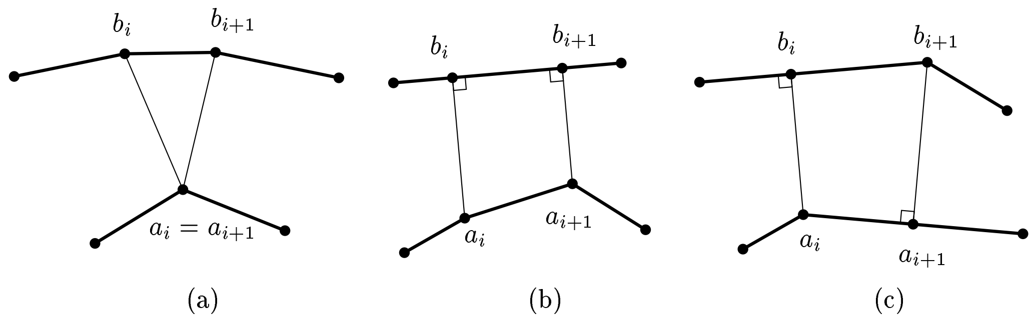


Fig. 6. Transition property for (a_i, b_i) and (a_{i+1}, b_{i+1}).

$a_i = a_{i+1}$, then b_i and b_{i+1} are adjacent vertices of P (since shortest paths of F do not cross).

If $a_i \neq a_{i+1}$ and $b_i \neq b_{i+1}$ then these pairs make a transition since there is no vertex between a_i and b_i and there is no vertex between a_{i+1} and b_{i+1}.

We compute all the widths $w(a,b), (a,b) \in A$ in lexicographical order (an edge of α with endpoints a and a' is between a and a' in the order). For a vertex $a_i \in \alpha$, the dynamic program stores all the pairs $(a_i, b) \in A$ and the minimum width $w(a_i, b)$ for each such pair. Initially $i = 0$ and the shortest paths from a_0 to the vertices of the polyline β can be found in linear time using the *shortest path tree* T_{a_0} [9, 10], see Fig. 7(a). Using the tree T_{a_0} we find the shortest paths from a_0 to edges of β and compute the distances $d_P(a_0, b)$, $(a_0, b) \in A$. The values $w(a_0, b)$ can be computed using the recurrence

$$w(a_0, b) = \max(d_P(a_0, b), w(a_0, b')), \text{ where } (a_0, b') \text{ precedes } (a_0, b) \text{ in } A.$$

Let $e = [a_i a_{i+1}]$ be the edge of α such that the widths $w(a_i, b)$, $(a_i, b) \in A$ are computed. To find the values $w(a_{i+1}, b)$, $(a_{i+1}, b) \in A$ we construct the shortest path tree $T_{a_{i+1}}$ and compute the distances $d_P(a_{i+1}, b)$, $(a_{i+1}, b) \in A$ in linear time. We also need to find the distances $d_P(e, b)$, $(e, b) \in A$.

Shortest paths to the edge e. Note that, if $(e, b) \in A$, then b is a vertex of β and the shortest path $\pi(e, b)$ meets the edge e with angle $90°$. A part of the polyline β visible from the edge e in the direction perpendicular to e can be found in linear time. We can use ray shooting to find the range R of vertices of β that participates in pairs $(e, b) \in A$. Two rays are emanated from a_i and a_{i+1} in the direction normal to e. The edges of P hit by the rays can be found by simply checking all the edges of P in $O(n)$ time. Alternatively, we can find the vertices in R using the fact that the shortest paths from a vertex of R to endpoints of e form acute angles at both ends of e.

We find the shortest paths $\pi(e, b), b \in R$ as follows. We cut off the vertices

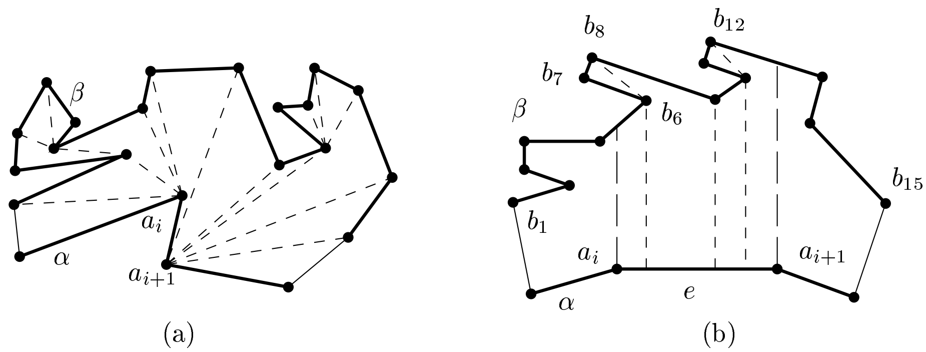


Fig. 7. (a) Shortest path tree $T_{a_{i+1}}$. (b) Vertices $b_6, b_7, \ldots, b_{12}$ form shortest paths to e.

outside the range R and add a point p_∞ at infinity behind the edge e. We construct the tree of shortest paths from p_∞ using the algorithm [9, 10] in slightly general setting. Truncating each path by the segment e we obtain the distances $d_P(e,b)$, $b \in R$, see Fig. 7(b).

Let $R = \{b_j, b_{j+1}, \ldots, p_k\}$. The dynamic program computes the widths $w(e, b_l)$, $b_l \in R$ using the following recurrence.

$$\begin{aligned} w(e, b_j) &= \max(d_P(e, b_j), w(a_i, b_j)) \\ w(e, b_l) &= \max(d_P(e, b_l), w(e, b_{l-1})) \text{ where } j < l \le k \end{aligned}$$

The widths $w(a_i, b)$, $(a_i, b) \in A$ can be found similarly.

The correctness follows by Lemma 2. The total running time of the algorithm is $O(n^2)$ since each transition takes $O(n)$ time.

Theorem 2 *An optimal morphing between two polylines can be found in $O(n^2)$ time using $O(n)$ space.*

Proof. The above algorithm computes the optimal width in $O(n^2)$ time using linear space. It remains to show how to compute a morph F of the optimal width. A morph can be described by its skeleton (the information about the match can be derived easily from the skeleton). One way to compute the skeleton is to modify the above dynamic program and store all the possible matches (a, b) and pointers between them indicating the transition matches. Then the optimal match $M(F)$ can be found by backtracking. The obvious drawback of this approach is that the space requirement is $\Omega(n^2)$.

To achieve a linear space we use the following approach. Suppose that the number of vertices of α is at least the number of vertices of β. Let p_i be the middle vertex of α, i.e., $i = \lceil k/2 \rceil$ where $p_1 = \alpha_0, p_2, \ldots, p_k = \alpha_1$ are the vertices of α. The point p_k partition the polyline α into two subpolylines. We apply the dynamic

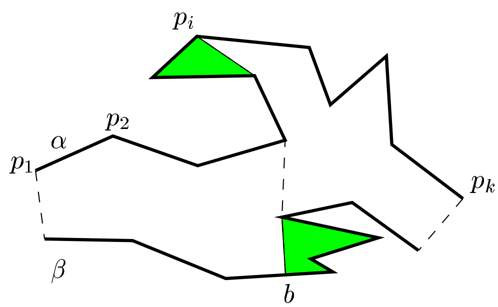


Fig. 8. Pockets are shaded.

program twice: for morphing the polyline p_1, p_i to β and for morphing the polyline p_i, p_k to β. The only difference is that we use the dynamic program in "opposite" directions in the sense that the two morphs end at p_i. The two dynamic programs in the last step produce the optimal widths for every pair $(p_i, b) \in A$. These widths correspond to two sides of the shortest path $\pi(p_i, b)$. Using all these widths we can find the shortest path that participates in the optimal morph between α and β (it is included in $S(F)$ too). This reduces the problem to subproblems of smaller size. If the shortest path we found is just a segment then the subproblems can be solved recursively. Otherwise some subpolylines make pockets, see Fig. 8. We can slightly modify the dynamic program to handle the pockets. We store segments of shortest paths found recursively in a list that eventually corresponds to a skeleton.

The space required by the modified algorithm is linear. Let $T(n)$ be the running time of the algorithm where n is the total number of segments of α and β. It implies the recurrence

$$T(n) = O(n^2) + T(n_1) + T(n_2)$$

where $n_1, n_2 \leq 3n/4$ and $n_1 + n_2 \leq n + 1$. The solution of the recurrence is $T(n) = O(n^2)$ and we are done. $\square$.

5. Conclusion

In this paper, we study the problem of finding an optimal morphing from one polyline to another minimizing the morphing width. We characterized the solution and proved that there is an optimal morph in a certain class of morphs. We presented an algorithm for finding an optimal morphing in $O(n^2)$ time. Future directions for study are (i) extensions to higher dimensions, (ii) various shapes (points, segments, polygons, etc.), and (iii) other morphing objectives.

Acknowledgements

I would like to thank the anonymous referees for useful comments and suggestions.

References

1. O. Aichholzer, H. Alt, and G. Rote, "Matching shapes with a reference point", *Internat. J. Comput. Geom. Appl.*, **7** (1997) 349–363.
2. H. Alt and M. Godau, "Computing the Fréchet distance between two polygonal curves", *Internat. J. Comput. Geom. Appl.* **5** (1995) 75–91.
3. H. Alt and L. J. Guibas, "Discrete geometric shapes: Matching, interpolation, and approximation", in *Handbook of Computational Geometry*, eds. J.-R. Sack and J. Urrutia, chapter 3 (Elsevier 2000) 121–153.
4. H. Alt, K. Mehlhorn, H. Wagener, and E. Welzl, "Congruence, similarity and symmetries of geometric objects", *Discrete Comput. Geom.* **3** (1998) 237–256.
5. H. Alt, C. Knauer, and C. Wenk, "Matching polygonal curves with respect to the Fréchet distance", in *Proc. 18th Internat. Symp. on Theoretical Aspects of Computer Science* (2001) pp. 63–74.
6. A. Efrat, L. J. Guibas, S. Har-Peled, D. C. Lin, J. S. B. Mitchell, and T. M. Murali, "Sweeping simple polygons with a chain of guards", in *Proc. 11th ACM-SIAM Sympos. Discrete Algorithms* (2000) pp. 927–936.
7. A. Efrat, L. J. Guibas, S. Har-Peled, and T.M. Murali, "Morphing between polylines", in *Proc. 12th ACM-SIAM Sympos. Discrete Algorithms* (2001) pp. 680–689.
8. L. Guibas, J. Hershberger, and S. Suri, "Morphing simple polygons", *Discrete Comput. Geom.* **24** (2000) 1–34.
9. L. J. Guibas, J. Hershberger, D. Leven, M. Sharir, and R. E. Tarjan, "Linear-time algorithms for visibility and shortest path problems inside triangulated simple polygons", *Algorithmica* **2** (1987) 209–233.
10. J. Hershberger and J. Snoeyink, "Computing minimum length paths of a given homotopy class", *Comput. Geom. Theory Appl.* **4** (1994) 63–98.
11. A. W. F. Lee, D. Dobkin, W. Sweldens, and P. Schröder "Multiresolution mesh morphing", in *Computer Graphics Proceedings* (1999) pp. 343–350. ACM Siggraph, 1999.
12. S. Venkatasubramanian, "Geometric Shape Matching and Drug Design", Ph. D. thesis, Stanford University (1999).