Controllable Morphing of Compatible Planar Triangulations

VITALY SURAZHSKY
and
CRAIG GOTSMAN
Department of Computer Science
Technion—Israel Institute of Technology
Technion City, Haifa 32000, Israel

Two planar triangulations with a correspondence between the pair of vertex sets are compatible (*isomorphic*) if they are topologically equivalent. This work describes methods for morphing compatible planar triangulations with identical convex boundaries in a manner that guarantees compatibility throughout the morph. These methods are based on a fundamental representation of a planar triangulation as a matrix that unambiguously describes the triangulation. Morphing the triangulations corresponds to interpolations between these matrices.

We show that this basic approach can be extended to obtain better control over the morph, resulting in valid morphs with various natural properties. Two schemes, which generate the linear trajectory morph if it is valid, or a morph with trajectories close to linear otherwise, are presented. An efficient method for verification of validity of the linear trajectory morph between two triangulations is proposed. We also demonstrate how to obtain a morph with a natural evolution of triangle areas and how to find a smooth morph through a given intermediate triangulation.

Categories and Subject Descriptors: I.3.5 [**Computing Methodologies**]: Computer Graphics—*Curve, surface, solid, and object representations*; *Geometric algorithms, languages, and systems*

General Terms: Algorithms

Additional Key Words and Phrases: Controllable Morphing, Compatible triangulations, Isomorphic triangulations, Local Control, Linear Morph, Morphing, Self-intersection elemination

1. INTRODUCTION

Morphing, also known as metamorphosis, is the gradual transformation of one shape (the *source*) into another (the *target*). Morphing has wide practical use in areas such as computer graphics, animation and modeling. Currently, to achieve more spectacular, impressive and accurate results, the morphing process requires a lot of the work to be done manually. A major research challenge is to develop techniques that will automate this process as much as possible.

The morphing problem has been investigated in many contexts, e.g., morphing of two-dimensional images [Beier and Neely 1992; Fujimura and Makarov 1998; Tal and Elber 1999], polygons and polylines [Sederberg and Greenwood 1992; Sederberg et al. 1993; Shapira and Rappoport 1995; Goldstein and Gotsman 1995; Carmel and Cohen-Or 1997; Alexa et al. 2000], free-form curves [Samoilov and Elber 1998] and even voxel-based volumetric representations [Cohen-Or et al. 1998]. The morphing process always consists of solving two main problems. The first one is to find a correspondence between elements (features) of the two shape representations. The second problem is to find trajectories that corresponding elements travel during the morphing process. Regrettably, a formal definition of a successful correspondence does not exist, as well as a definition of a successful solution to the trajectory problem.

The naive solution to the trajectory problem, once a correspondence has been established, is to choose the trajectories to be straight lines, where every feature of the source shape travels with a constant velocity towards the corresponding feature of the target shape. A morph generated in this manner is said to be a *linear* morph. Unfortunately, this simple approach can lead to some undesirable results. The

intermediate shapes can vanish, namely, degenerate into a single point. Moreover, intermediate shapes may self-intersect, even though the source and target shapes do not. Even if the linear morph is free from self-intersections and degenerate regions, its intermediate shapes may have areas or distances between the shape elements far from those of the given shapes, resulting in a 'misbehaved' looking morph.

Most of the research on solving the trajectory problem for morphing concentrates on the elimination of self-intersections and on the preservation of geometrical properties of the intermediate shapes. Sederberg et al. [1993] morph piecewise linear curves (polylines) using a *heuristic* algorithm which interpolates the angles between adjacent edges as well as the length of the edges. Shapira and Rappoport [1995] use a *skeleton* representation of the geometry of both closed polylines (polygons), which takes into account the interiors of the polygons as well as the boundaries. The work of Samoilov and Elber [1998] concentrates on self-intersection elimination using two methods. The first method builds a 3D homotopy of two planar curves and projects it back into the plane to obtain a sequence of planar curves. The second method flips segments of the curves involved in self-intersection to eliminate it. Goldstein and Gotsman [1995] precompute *multiresolution representations* of two polygons, based on *curve evolutions*. A morphing sequence is reconstructed from the intermediate representations in the different resolutions of the curves. Alexa et al. [2000] compatibly triangulate the source and target polygons, and optimize the vertex trajectories during the morph in an attempt to maintain compatibility and preserve shape. This results in non-linear vertex trajectories, in which individual triangle orientations seem to be preserved on a local scale throughout the morph. However, on a global scale, there is nothing to prevent the intermediate polygons from self-intersecting. All of the above methods achieve good results for many inputs. However, none guarantees that the resulting morph will be self-intersection free.

Image *warping* deals with deformations of images. Image morphing is a computer animation technique, which aims to continuously transform one image into another. Beier and Neely [1992] proposed an image morphing approach, which is based on attaching a set of corresponding line segments to images such that image pixels have coordinates with respect to this set. This approach performs linear interpolation on the line segments and determines pixel values using their line coordinates. Since each line segment is morphed linearly, the method may induce self-intersection of the line segments, resulting in foldover of image regions. Fujimura and Makarov [1998] presented an image warping method, which exploits a time-varying triangulation to obtain a foldover-free warp. The method permits points, line-segments and even polygons as corresponding features of the image. However, it requires as input also trajectories that the image features travel during the warping process. Tal and Elber [1999] triangulate the interior of the source and target polygons in a compatible manner prior to morphing. The main motivation for this is that texture mapping be well-defined on the interiors of the triangulated polygons as a piecewise-affine mapping. However, the actual vertex trajectories used by Tal and Elber are the simple linear ones, hence do not guarantee that the triangulated polygons are compatible throughout the morph.

Floater and Gotsman [1999] introduced an innovative approach for morphing planar *triangulations*. Triangulations are ubiquitously used in computer graphics as a representation and a parameterization (e.g., for texture mapping) of surfaces and planar shapes. Two triangulations with a correspondence between their vertex sets are said to be compatible (*isomorphic*), if they are topologically equivalent. That work presents a robust technique for morphing compatible triangulations based on a convex representation of triangulations. They show that the method, applied to compatible triangulations with an identical boundary, always yields a valid morph in the form of a continuous sequence of valid triangulations, namely, the triangulation does not self-intersect during the morph.

Our work is based on that of Floater and Gotsman [1999], where only the basic approach was presented. Here we analyze this approach in depth and investigate its properties and capabilities. In addition to analysis of the basic global scheme, we present several extensions that allow more local control over the morphs, namely, self-intersection free morphs with prescribed properties may be obtained. Two linear-reducible schemes are proposed. Both schemes produce linear trajectory morphs if possible, or a morph with close to linear trajectories otherwise. These schemes may also be combined to obtain a morph, which will be as close as possible to the linear trajectory morph. Moreover, we introduce an efficient method for verifying of the validity of the linear morph between two triangulations. We also show how to obtain a morph through a predefined intermediate triangulation. As another demonstration of the extensibility of the basic approach, this work presents a method that generates a morph with a natural (almost uniform) evolution of triangle areas.

2. BACKGROUND

First, some standard definitions from graph theory: a simple graph $\mathcal{G} = \mathcal{G}(V, E)$ is a set of vertices $V = \{1, \ldots, |V|\}$ and a set of edges E, such that E is a subset of all unordered pairs of vertices $\{i, j\}$, when $i \neq j$. Two graphs $\mathcal{G}_0$ and $\mathcal{G}_1$ are *isomorphic* if there is a 1–1 correspondence between their vertices and edges in such a way that corresponding edges link corresponding vertices.

A simple graph $\mathcal{G}$ is said to be *planar* if it can be drawn in the plane in such a way that the following holds:

(i) there exists a 1–1 mapping $\wp$ of vertices V onto a set of distinct points of the plane;
(ii) each edge $\{i, j\} \in E$ corresponds to a simple curve with endpoints $\wp(i)$ and $\wp(j)$;
(iii) the only intersections between curves are at common endpoints.

This representation of a planar graph $\mathcal{G}$ is called a *plane graph*, denoted by an ordered pair $(\mathcal{G}, \wp)$. A mapping $\wp\colon V \to \mathbb{R}^2$ may also be viewed as a point sequence $\{i \mapsto (x_i, y_i) \mid i\colon 1, \ldots, |V|\}$. The following notations are equivalent: $\wp(i)$, p_i, (x_i, y_i).

A plane graph partitions the plane into connected regions called *faces*. Obviously, a plane graph has a single unbounded face, denoted by the *outer* face. The set of vertices and the set of edges adjoining to the *outer* face form a subgraph $\partial\mathcal{G}$, called the *boundary* of a plane graph $(\mathcal{G}, \wp)$. Note that different drawings of a planar graph in the plane may result in different boundaries. Thus different plane graphs of the same planar graph may partition the plane topologically in different ways.

A plane graph is said to be *triangulated* if all its bounded faces have exactly three edges. A *planar triangulation* $\mathcal{T} = \mathcal{T}(\mathcal{G}, \wp)$ is a simple triangulated plane graph such that its edges are represented by straight lines. In this work, we deal only with *planar* triangulations, and they will be called simply *triangulations*. We call a triangulation *valid* if it satisfies the above definition, in particular, the only intersections between its edges are at common endpoints. Otherwise (if the edges intersect at interior points), a triangulation is called *invalid*.

Intuitively, two (valid) triangulations are *isomorphic* if they are topologically equivalent. We will define this more formally using a notion of *orientation*. A face f is an ordered triplet of its vertices (i, j, k). An *orientation* of face f is said to be *counterclockwise* if the face vertices i, j, k, in this specific order, lie in the counterclockwise direction relative to the centroid of the face. Analogously, an *orientation* is said to be *clockwise* if the face vertices are ordered in the clockwise direction. Since the boundary $\partial\mathcal{G}$ of $\mathcal{G}$ is a simple closed polyline in the plane, it also has a well-defined orientation relative to the interior faces.

***Definition* 2.1** Two triangulations $\mathcal{T}_0 = \mathcal{T}(\mathcal{G}_0, \wp_0)$ and $\mathcal{T}_1 = \mathcal{T}(\mathcal{G}_1, \wp_1)$ are *isomorphic* if the following conditions hold.

(1) The two graphs $\mathcal{G}_0$ and $\mathcal{G}_1$ are isomorphic.
(2) There is a 1–1 correspondence between the bounded faces of $\mathcal{T}_0$ and $\mathcal{T}_1$ such that corresponding faces join corresponding vertices.
(3) The boundaries $\partial\mathcal{G}_0$ and $\partial\mathcal{G}_1$ have the same orientation, i.e. there exist two sequences of vertices of $\partial\mathcal{G}_0$ and $\partial\mathcal{G}_1$ in counterclockwise direction relatively to the interior faces such that the sequences correspond.
Note that (3) is equivalent to (3′).
(3′) All corresponding bounded faces have the same orientation.

Two sequences of N points $\wp_0$ and $\wp_1$ are said to be *compatible* if there exists a planar graph $\mathcal{G}$ such that two (valid) triangulations $\mathcal{T}_0 = \mathcal{T}(\mathcal{G}, \wp_0)$ and $\mathcal{T}_1 = \mathcal{T}(\mathcal{G}, \wp_1)$ are isomorphic. A sequence of N points $\wp_0$ is said to be *compatible with* a triangulation $\mathcal{T}_1 = \mathcal{T}(\mathcal{G}, \wp_1)$ if $\mathcal{T}_0 = \mathcal{T}(\mathcal{G}, \wp_0)$ is a (valid) triangulation and $\mathcal{T}_0$ and $\mathcal{T}_1$ are isomorphic. To be consistent with other works, we also call isomorphic triangulations *compatible* triangulations. Figure 1 shows some compatible and non-compatible triangulations.

It is easy to verify (decide) whether two triangulations are compatible (isomorphic) or whether two point sequences are compatible with respect to a given triangulation. The problem arises when it is necessary to *find* isomorphic triangulations for given point sequences. The decision problem determining whether two point sequences are compatible has been investigated [Saalfeld 1987; Aronov et al. 1993] and is believed to be NP-hard. Souvaine and Wenger [1994] demonstrated that two sequences of N points may be made compatible by adding $O(N^2)$ *Steiner* (extraneous) points. Moreover, there exist N-point sequences requiring $\Omega(N^2)$ points to be added to obtain compatibility.

2.1 Barycentric Coordinates

Given a polygon with k vertices p_1, p_2, ..., p_k, $k \geq 3$, any point p in the plane can be expressed as:

$$p = \sum_{i=1}^{k} \lambda_i \cdot p_i, \quad \sum_{i=1}^{k} \lambda_i = 1. \tag{1}$$

The coefficients $\lambda_1, \ldots, \lambda_k$ in these equations are said to be *barycentric coordinates* of p relative to $p_1, \ldots, p_k$. When p lies in the convex hull of a polygon, it can be expressed as a convex combination of the polygon vertices, namely, all barycentric coordinates have positive values. In this paper we consider the case when p lies in the kernel of a star-shaped polygon.

We denote by $\mathcal{S}(a, b, c)$ the area of a triangle with vertices a, b, c. It is well known that $\mathcal{S}(a, b, c)$ may be expressed using the vertex coordinates as follows:

$$\mathcal{S}(a, b, c) = \pm\frac{1}{2}\begin{vmatrix} 1 & 1 & 1 \\ x_a & x_b & x_c \\ y_a & y_b & y_c \end{vmatrix} \tag{2}$$

We are now interested in finding barycentric coordinates of p with respect to vertices of a polygon. The special case when the polygon has three vertices, namely, p lies in a triangle $\triangle(p_1, p_2, p_3)$, is simple. The barycentric coordinates of p with respect to p_1, p_2, p_3 are unique:

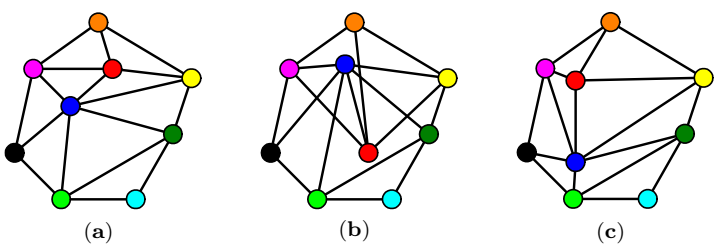
$$\lambda_1 = \frac{\mathcal{S}(p, p_2, p_3)}{\mathcal{S}(p_1, p_2, p_3)}, \quad \lambda_2 = \frac{\mathcal{S}(p, p_3, p_1)}{\mathcal{S}(p_1, p_2, p_3)}, \quad \lambda_3 = \frac{\mathcal{S}(p, p_1, p_2)}{\mathcal{S}(p_1, p_2, p_3)}.$$

A point p lying in the kernel of a star-shaped polygon with more than three vertices has non-unique barycentric coordinates. A simple solution is to choose any triangle containing p whose vertices are vertices of the polygon. Barycentric coordinates of p now may be defined as non-zero coordinates with respect to the triangle vertices, and zeros for all other vertices.

Floater [1997] presented a scheme that produces more uniform barycentric coordinates. His scheme combines barycentric coordinates generated using the simple method described above applied to a number of triangles. For each vertex p_i of the k vertices of the polygon there exists a triangle $\triangle_i$ such that p_i is one of the triangle vertices, and $\triangle_i$ contains p. These k different sets of barycentric coordinates of p, not necessarily distinct, are averaged to produce the final barycentric coordinates. The drawback of this scheme is that it produces barycentric coordinates which are only $\mathbf{C^0}$-continuous with respect to change in the location of p. We will see later in Section 4 that higher order continuity is essential to generate natural (smooth) morphs.

We introduce a new scheme having $\mathbf{C}^\infty$-continuity in most cases. The scheme finds barycentric coordinates $\lambda = (\lambda_1, \ldots, \lambda_k)$ of a point p with respect to a polygon with k vertices $p_1, \ldots, p_k$ satisfying the following condition: The weighted variance between the coordinates is minimal, where the weights are the distances between p and the polygon vertices. This can be formulated as follows:

$$\begin{aligned} &\text{minimize} && f(\lambda_1, \ldots, \lambda_k) = \sum_{i=1}^{k} \|p - p_i\| \, (\frac{1}{k} - \lambda_i)^2, \\ &\text{subject to} && p = \sum_{i=1}^{k} \lambda_i \cdot p_i, \qquad \sum_{i=1}^{k} \lambda_i = 1. \end{aligned} \tag{3}$$



(**a**) (**b**) (**c**)

Fig. 1. (**a**) Triangulation of 9 points in the plane. (**b**) Triangulation not compatible with (a). Vertex correspondence coded in colors. (**c**) Triangulation compatible with (a).

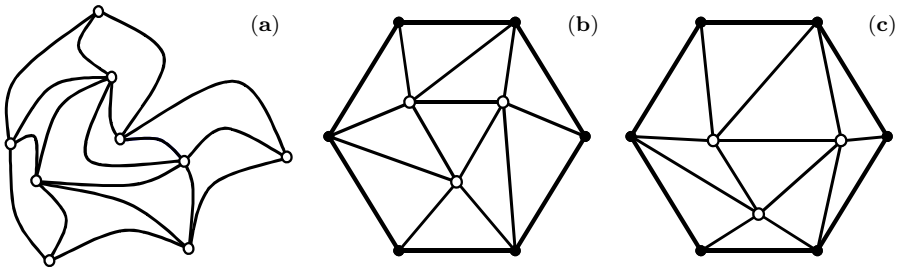


Fig. 2. Constructing a straight line representation of a triangulated plane graph: (**a**) a triangulated plane graph; (**b**) a corresponding triangulation—the boundary is a convex polygon, the interior vertices are centroids of their neighboring vertices; (**c**) each interior vertex is an arbitrary convex combination of its neighboring vertices.

This linear least squares problem can be solved using Lagrange multipliers. In rare cases, this scheme can result in barycentric coordinates with negative values. However, they usually are very close to zero and, in general, almost negligible. To eliminate these negative barycentric coordinates entirely we can use the Kuhn-Tucker method [Luenberger 1973] to solve (3) with extra inequality constraints: $\lambda_i \geq 0, i = 1, \ldots, k$. In these cases our scheme will be only $\mathbf{C^0}$-continuous.

Recently, Sugihara [1999] presented a new scheme based on Voronoi diagrams to generate barycentric coordinates. For the case of star-shaped polygons, the scheme is $\mathbf{C}^\infty$-continuous, and moreover, is simple to calculate—it does not require an explicit computation of the Voronoi diagram. Unfortunately, Sugihara's scheme may also produce negative barycentric coordinates, which are small, but not negligible.

2.2 Drawing Triangulated Graphs

It has been shown by Fáry [1948] that every planar graph has a *straight line representation*. Hence, for every planar triangulated graph $\mathcal{G}$, there exists a point sequence $\wp$ such that $\mathcal{T} = \mathcal{T}(\mathcal{G}, \wp)$ is a (valid) triangulation. Tutte [1963] described the following method to generate $\wp$: The boundary vertices of $\mathcal{G}$ are mapped to an arbitrary convex polygon with the same number of vertices and the same vertex order. Then, the interior vertices are placed in such a way that every vertex is the centroid of the polygon of its neighboring vertices, see Figure 2(**b**). This scheme was extended by Floater [1997]. Each interior vertex can be *any* convex combination of its neighbors, see Figure 2(**c**). In terms of barycentric coordinates, any positive barycentric coordinates for each interior vertex may be chosen with respect to its neighbors.

To compute $\wp$, we use the following method. Let $\mathcal{G} = \mathcal{G}(V, E)$ be a simple triangulated graph, with $|V| = N$. We assume that boundary vertices of $\mathcal{G}$ have been identified. These boundary vertices correspond to some planar representation of $\mathcal{G}$. Let V_I be the set of the interior vertices and V_B be the set of the boundary vertices such that $|V_I| = n$ and $|V_B| = N - n = k$. Without loss of generality assume $V_I = \{1, \ldots, n\}$ and $V_B = \{n+1, \ldots, N\}$. Now we wish to find coordinates of the graph vertices, namely, for each vertex $i \in V$ to find $\wp(i) = (x_i, y_i)$. We define $\wp$ for each vertex $i \in V_B$ to be coordinates of the vertices of a k-sided convex polygon with the same vertex order as of $\partial\mathcal{G}$.

For each interior vertex we may choose arbitrary non-negative barycentric coordinates relative to its neighbors, namely, for each vertex $i \in V_I$ a set of scalars $\lambda_{i,j}$ for $j = 1, \ldots, N$ such that

$$\begin{aligned} &\lambda_{i,j} = 0, \quad \{i,j\} \notin E, \\ &\lambda_{i,j} \geq 0, \quad \{i,j\} \in E, \quad \text{barycentric coordinate of } i \text{ relatively to } j, \\ &\sum_{j=1}^{N} \lambda_{i,j} = 1, \quad \forall i = 1, \ldots, n. \end{aligned} \tag{4}$$

Next we define $p_1, \ldots, p_n$ to be the solution of the following system of linear equations

$$p_i = \sum_{i=1}^{N} \lambda_{i,j} \cdot p_j, \quad i = 1, \ldots, n. \tag{5}$$

This system contains n linear equations with n variables. Floater [1997] showed that the coefficient matrix of these equations is non-singular. Therefore, a unique solution always exists.

3. MORPHING USING NEIGHBORHOOD MATRICES

For a given triangulation $\mathcal{T} = \mathcal{T}(\mathcal{G}, \wp)$, it is possible to define a *neighborhood matrix* that describes the connectivity of the graph $\mathcal{G}$ as well as the mutual disposition of the interior vertices of $\mathcal{G}$ using barycentric coordinates. Let $A = A(\mathcal{T})$ be a $N \times N$ matrix defined as follows:

$$A(i,j) = \begin{cases} \lambda_{i,j}, & i = 1,\ldots,n, \quad j = 1,\ldots,N \\ 1, & i > n, \quad j = i, \\ 0, & i > n, \quad j \neq i. \end{cases} \tag{6}$$

The general form of A is:

$$A = \left[\begin{array}{ccc|ccc} \lambda_{1,1} & \cdots & \lambda_{1,n} & \lambda_{1,n+1} & \cdots & \lambda_{1,N} \\ \vdots & & \vdots & \vdots & & \vdots \\ \lambda_{n,1} & \cdots & \lambda_{n,n} & \lambda_{n,n+1} & \cdots & \lambda_{n,N} \\ \hline \multicolumn{3}{c|}{0} & \multicolumn{3}{c}{I} \end{array}\right] = \left[\begin{array}{c|c} A_I & A_B \\ \hline 0 & I \end{array}\right] \tag{7}$$

Note that all rows of A sum to unity due to (4). We call a neighborhood matrix *legal* if it satisfies (6), and *illegal* otherwise.

Let $x = x(\mathcal{T}) = (x_1, \ldots, x_N)$ be a column vector of x-components of the triangulation vertex coordinates $\wp(\mathcal{T})$, and $y = y(\mathcal{T}) = (y_1, \ldots, y_N)$ the vector of y-components respectively. Every row i of A satisfies the following equation: $p_i = \sum_{i=1}^{N} \lambda_{i,j} \cdot p_j$. Putting these equations together for $i = 1, \ldots, N$ we obtain:

$$A \cdot x = x, \qquad A \cdot y = y. \tag{8}$$

Namely, x and y are eigenvectors of A.

The matrix A may be constructed for a given triangulation $\mathcal{T}$ by calculating barycentric coordinates for the interior vertices of $\mathcal{T}$. It is also possible to build a triangulation given a matrix A and coordinates of the boundary vertices. The trivial zero solution for $A \cdot x = x$ is impossible due to the fixed (non-zero) boundary vertices. We show how this can be reduced to the solution of the linear system (5).

The vector x may be partitioned into two vectors $x_I = (x_1, \ldots, x_n)$ and $x_B = (x_{n+1}, \ldots, x_N)$. y is partitioned in the same manner. x_I and y_I stand for vectors of coordinates of the interior vertices, while x_B and y_B — for the boundary vertices. We can now write $A \cdot x = x$, namely, $(A - I) \cdot x = 0$ as follows:

$$\left[\begin{array}{c|c} A_I - I & A_B \\ \hline 0 & 0 \end{array}\right] \cdot \left(\left[\begin{array}{c} x_I \\ \hline 0 \end{array}\right] + \left[\begin{array}{c} 0 \\ \hline x_B \end{array}\right] \right) = 0 \tag{9}$$

$$(A_I - I) \cdot x_I + A_B \cdot x_B = 0 \tag{10}$$

Equation (10) has n variables in x_I. So, we get another form of (5). It was proven by Floater [1997] that the $n \times n$ matrix $(A_I - I)$ is not singular, and thus we always have a unique solution for x_I.

There is a one-to-many correspondence between a triangulation and neighborhood matrices. Given a neighborhood matrix and the corresponding boundary points, the interior points are uniquely determined. On the other hand, given a triangulation $\mathcal{T}$, it is possible to build a corresponding neighborhood matrix $A(\mathcal{T})$ in many different ways, since the barycentric coordinates of the triangulation's interior vertices are not unique. We denote by $\mathcal{A}(\mathcal{T})$ the set of all possible neighborhood matrices $A(\mathcal{T})$, and by $\mathbb{A}(\mathcal{T})$ the set of all neighborhood matrices corresponding to different triangulations compatible with $\mathcal{T}$. $\mathbb{A}(\mathcal{T})$ is called the *A-space* of $\mathcal{T}$. If $\mathcal{T}_0$ and $\mathcal{T}_1$ are compatible, then $\mathbb{A}(\mathcal{T}_0) = \mathbb{A}(\mathcal{T}_1)$. While $\mathbb{A}(\mathcal{T})$ describes only the "connectivity" information, $\mathcal{A}(\mathcal{T})$ describes also the "geometry" information. These definitions imply: $A(\mathcal{T}) \in \mathcal{A}(\mathcal{T})$, $\mathcal{A}(\mathcal{T}) \subset \mathbb{A}(\mathcal{T})$. $\mathcal{A}(\mathcal{T})$ is the equivalence class in $\mathbb{A}(\mathcal{T})$ that consists of all neighborhood matrices describing the "geometry" of $\mathcal{T}$.

3.1 The Morphing Scheme

A *morph* between two compatible triangulations $\mathcal{T}_0 = \mathcal{T}(\mathcal{G}, \wp_0)$ and $\mathcal{T}_1 = \mathcal{T}(\mathcal{G}, \wp_1)$ is a gradual transformation of $\mathcal{T}_0$ into $\mathcal{T}_1$. This transformation may be viewed as a continuous function $\mathcal{T}(t)$, where $0 \le t \le 1$ and $\mathcal{T}(0) = \mathcal{T}_0$, $\mathcal{T}(1) = \mathcal{T}_1$. A morph between $\mathcal{T}_0$ and $\mathcal{T}_1$ is *valid* if for all t, $0 \le t \le 1$, $\mathcal{T}(t)$ is a valid triangulation compatible with $\mathcal{T}_0$ (and $\mathcal{T}_1$). In this work we consider triangulations $\mathcal{T}_0$ and $\mathcal{T}_1$ such that the boundaries of the triangulations coincide. The boundaries of $\mathcal{T}(t)$ for $0 \le t \le 1$ will naturally also coincide, namely, $p_i(t) = p_i(0) = p_i(1)$ for $n < i \le N$ and $0 \le t \le 1$. To find $\mathcal{T}(t)$ means to find $p_i(t)$ for $1 \le i \le n$ in such a way that the point sequence $\wp_t$ is compatible with $\mathcal{T}_0$ (and $\mathcal{T}_1$).

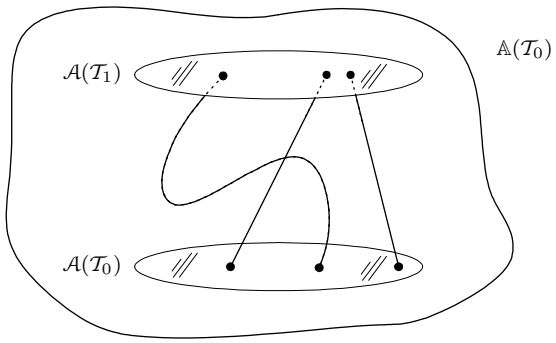


Fig. 3. Any curve in $\mathbb{A}(\mathcal{T}_0)$ $(= \mathbb{A}(\mathcal{T}_1))$ with endpoints in $\mathcal{A}(\mathcal{T}_0)$ and $\mathcal{A}(\mathcal{T}_1)$ defines a valid morph between $\mathcal{T}_0$ and $\mathcal{T}_1$. The two straight line segments are the convex combination morphs.

We show now how to find $\mathcal{T}(t)$ using neighborhood matrices. Neighborhood matrices A_0 and A_1 are generated corresponding to $\mathcal{T}_0$ and $\mathcal{T}_1$. The next step is to find a continuous function $A(t)$ such that $A(0) = A_0$, $A(1) = A_1$ and for all t, $0 \le t \le 1$, $A(t)$ is a legal neighborhood matrix describing the same connectivity as that of A_0 and A_1. In other words, we need to find a curve in $\mathbb{A}(\mathcal{T}_0)$ with endpoints in $\mathcal{A}(\mathcal{T}_0)$ and $\mathcal{A}(\mathcal{T}_1)$, see Figure 3. We can then find $\wp_t$ by solving $A(t) \cdot x(t) = x(t)$ and $A(t) \cdot y(t) = y(t)$. $\mathcal{T}(t) = \mathcal{T}(\mathcal{G}, \wp_t)$ is a valid triangulation since $A(t)$ is a legal neighborhood matrix with the connectivity of $\mathcal{G}$. $\mathcal{T}(0)$, $\mathcal{T}(1)$ and $\mathcal{T}(t)$ are compatible due to the common fixed boundary.

3.2 The Convex Combination Morph

The most straightforward method for generating a curve $A(t)$, $0 \le t \le 1$ in $\mathbb{A}(\mathcal{T}_0)$ with endpoints in $\mathcal{A}(\mathcal{T}_0)$ and $\mathcal{A}(\mathcal{T}_1)$ was proposed by Floater and Gotsman [1999]. Just use a straight line segment connecting arbitrary points $A_0 \in \mathcal{A}(\mathcal{T}_0)$ and $A_1 \in \mathcal{A}(\mathcal{T}_1)$, see Figure 3. Thus $A(t) = (1-t) \cdot A_0 + t \cdot A_1$ when $0 \le t \le 1$. It is easy to see [Floater and Gotsman 1999] that $A(t) \in \mathbb{A}(\mathcal{T}_0)$ for $0 \le t \le 1$. This solution is called *the convex combination morph*, since $A(t)$ is a convex combination of A_0 and A_1.

We show now that trajectories traveled by the interior vertices during the morph are continuous and smooth, namely, that $p_i(t)$, $1 \le i < n$ are $\mathbf{C^0}$ and $\mathbf{C^1}$ continuous. Since $A(t)$ depends continuously and smoothly on t, the exact solution of the linear system (5) is continuous and smooth. More specifically, the ith component x_i of the solution has the form:

$$x_i(t) = \frac{P_i^x(t)}{P_\Delta(t)}, \tag{11}$$

where $P_i^x(t)$ and $P_\Delta(t)$ are polynomials of degree n on t; $P_\Delta(t) \neq 0$ being the determinant of the corresponding non-singular [Floater 1997] matrix for the linear system (5).

It is important that $\mathcal{T}(t)$ never passes through any triangulation $\mathcal{T}'$ twice; and even a small advance of t from 0 to 1 achieves a non-zero change in $\mathcal{T}(t)$ towards $\mathcal{T}_1$. The following theorem formalizes this:

Theorem 3.1 *$\mathcal{T}(t)$ obtained by the convex combination morph is injective.*

We need the following two lemmas to prove the theorem:

Lemma 3.2 *The set $\mathcal{A}(\mathcal{T})$ does not contain interior points, namely, if $A \in \mathcal{A}(\mathcal{T})$ then $A \in \partial\mathcal{A}(\mathcal{T})$.*

Proof. Let $\mathcal{T}_0$ and $\mathcal{T}_1$ be two isomorphic triangulations, and let $A_0 \in \mathcal{A}(\mathcal{T}_0)$, $A_1 \in \mathcal{A}(\mathcal{T}_1)$ be two arbitrary neighborhood matrices corresponding to $\mathcal{T}_0$ and $\mathcal{T}_1$ respectively. Let $A(s)$ be a line segment between A_0 and A_1 expressed as $A(s) = s \cdot A_1 + (1-s) \cdot A_0$, $0 \le s \le 1$. We wish to show that if $\mathcal{A}(\mathcal{T}_0) \neq \mathcal{A}(\mathcal{T}_1)$ then $A(s) \notin \mathcal{A}(\mathcal{T}_0)$ for $s > 0$. For $x_0 = x(\mathcal{T}_0)$ and for every $A \in \mathcal{A}(\mathcal{T}_0)$ we have $A \cdot x_0 = x_0$. It is necessary to show that $A(s) \cdot x_0 \neq x_0$ for $s > 0$. Assume by contraposition that $A(s) \cdot x_0 = x_0$ for $s > 0$. Then

$$\begin{aligned} A(s) \cdot x_0 &= (s \cdot A_1 + (1-s) \cdot A_0) \cdot x_0 \\ &= (s \cdot (A_1 - A_0) + A_0) \cdot x_0 \\ &= s \cdot (A_1 \cdot x_0 - A_0 \cdot x_0) + A_0 \cdot x_0 \\ &= s \cdot (A_1 \cdot x_0 - x_0) + x_0 \end{aligned}$$

Using the assumption:

$$\begin{aligned} s \cdot (A_1 \cdot x_0 - x_0) + x_0 &= x_0 \\ s \cdot (A_1 \cdot x_0 - x_0) &= 0 \\ A_1 \cdot x_0 &= x_0 \quad \text{since } s > 0 \end{aligned}$$

For $\mathcal{A}(\mathcal{T}_0) \neq \mathcal{A}(\mathcal{T}_1)$ we obtain a contradiction. $A_1 \cdot x_0 \neq x_0$ since $A_1 \notin \mathcal{A}(\mathcal{T}_0)$. □

Lemma 3.3 *The set $\mathcal{A}(\mathcal{T})$ is convex.*

Proof. Let $A_0 \in \mathcal{A}(\mathcal{T})$ and $A_1 \in \mathcal{A}(\mathcal{T})$ be arbitrary points in $\mathcal{A}(\mathcal{T})$. Let $A(s)$ be a convex combination of A_0 and A_1. We need to prove that $A(s) \in \mathcal{A}(\mathcal{T})$ for $0 \leq s \leq 1$. For $x = x(\mathcal{T})$ and for every $A \in \mathcal{A}(\mathcal{T})$ it holds that $A \cdot x = x$. Hence it suffices to show that $A(s) \cdot x = x$ for $0 \leq s \leq 1$.

$$\begin{aligned} A(s) \cdot x &= (s \cdot A_1 + (1-s) \cdot A_0) \cdot x \\ &= s \cdot A_1 \cdot x + (1-s) \cdot A_0 \cdot x \\ &= s \cdot x + (1-s) \cdot x = x \end{aligned}$$

□

Now we can proceed with the proof of Theorem 3.1.

Proof. Assume by contraposition that there exist $\mathcal{T}(t_1)$ and $\mathcal{T}(t_2)$ such that $\mathcal{T}(t_1) = \mathcal{T}(t_2)$ and $t_1 \neq t_2$. Suppose, without loss of generality, that $t_1 < t_2$. Let $\mathcal{T}' = \mathcal{T}(t_1) = \mathcal{T}(t_2)$. Since $\mathcal{T}(t_1) = \mathcal{T}(t_2)$, $A(t_1) \in \mathcal{A}(\mathcal{T}')$ and $A(t_2) \in \mathcal{A}(\mathcal{T}')$. Obviously, $A(t_1) \neq A(t_2)$, because $A(t)$ is a convex combination of A_0 and A_1 when $A_0 \neq A_1$. Due to Lemma 3.3, $\mathcal{A}(\mathcal{T}')$ is convex. Therefore $A(t) \in \mathcal{A}(\mathcal{T}')$ for $t_1 \leq t \leq t_2$. $\mathcal{A}(\mathcal{T}_0)$ and $\mathcal{A}(\mathcal{T}(t_2))$ have no internal points by Lemma 3.2. Consequently, for $A(t)$, being a line segment between A_0 and $A(t_2)$, it holds that $A(t) \notin \mathcal{A}(\mathcal{T}_0)$ and $A(t) \notin \mathcal{A}(\mathcal{T}(t_2))$ when $0 < t < t_2$. This is a contradiction, since $A(t_1) \in [\mathcal{A}(\mathcal{T}(t_2)) = \mathcal{A}(\mathcal{T}')]$ when $t_1 < t_2$. □

An important property of morphing schemes is invariance to affine transformations. Thus it is particularly encouraging that the convex combination morph is invariant to affine transformations due to the inherent nature of barycentric coordinates and neighborhood matrix properties.

3.3 Approaching the Linear Morph

A morph of two isomorphic triangulations is said to be *linear* if the interior vertices traverse linear trajectories with constant velocities, i.e. $p_i(t) = (1-t) \cdot p_i(0) + t \cdot p_i(1)$, $1 \leq i \leq n$. A morphing scheme is called *linear-reducible* if it generates the linear morph when that morph is valid, and some other valid morph otherwise. It is easy to see that when the triangulations have a single interior point, the convex combination morph is linear. This follows from the dependency of the interior vertex solely on the fixed boundary vertices. In contrast, the convex combination morph of two triangulations with more than one interior vertex is *never* linear. This follows from the fact that equation (11) cannot be reduced to the form $(1-t) \cdot x_i(0) + t \cdot x_i(1)$ for $n > 2$, since the polynomials $P_i^x(t)$ and $P_\Delta(t)$ are of full degree n in t. For that reason, in some cases, the linear morph is to be preferred over the convex combination morph. For example, if two triangulations have subsets of vertices with identical positions, the vertices will remain fixed during the linear morph, while the convex combinations morph will distort them. See Figure 4 for an example.

To obtain a morph that is as close as possible to the linear one, consider the following interpolant between $A(0)$ and $A(1)$:

$$A(t) = (1-t) \cdot A_0^m + t \cdot A_1^m, \quad m \in \mathbb{N}. \tag{12}$$

This generates a morph that approaches the linear morph as the power m increases. For m approaching infinity the generated morph is linear (but might not be valid).

To see why this is true, observe that any triangulation can be viewed as a graph of states with transitions. Given a triangulation $\mathcal{T} = \mathcal{T}(\mathcal{G}, \wp)$, define a corresponding transition (directed) graph $G = G(V, E)$ that has the same vertices as $\mathcal{G}(\mathcal{T})$, namely, $V(G) = V(\mathcal{G}(\mathcal{T}))$, see Figure 5. For each undirected edge of $\mathcal{G}(\mathcal{T})$ which links two interior vertices, define two directed edges in $E(G)$ connecting these vertices in the opposite directions. Every edge between a boundary vertex v_B and an interior vertex v_I corresponds to a directed edge $(v_I \rightarrow v_B)$. In addition, we attach loop edges to vertices corresponding to the boundary vertices of $\mathcal{G}(\mathcal{T})$.

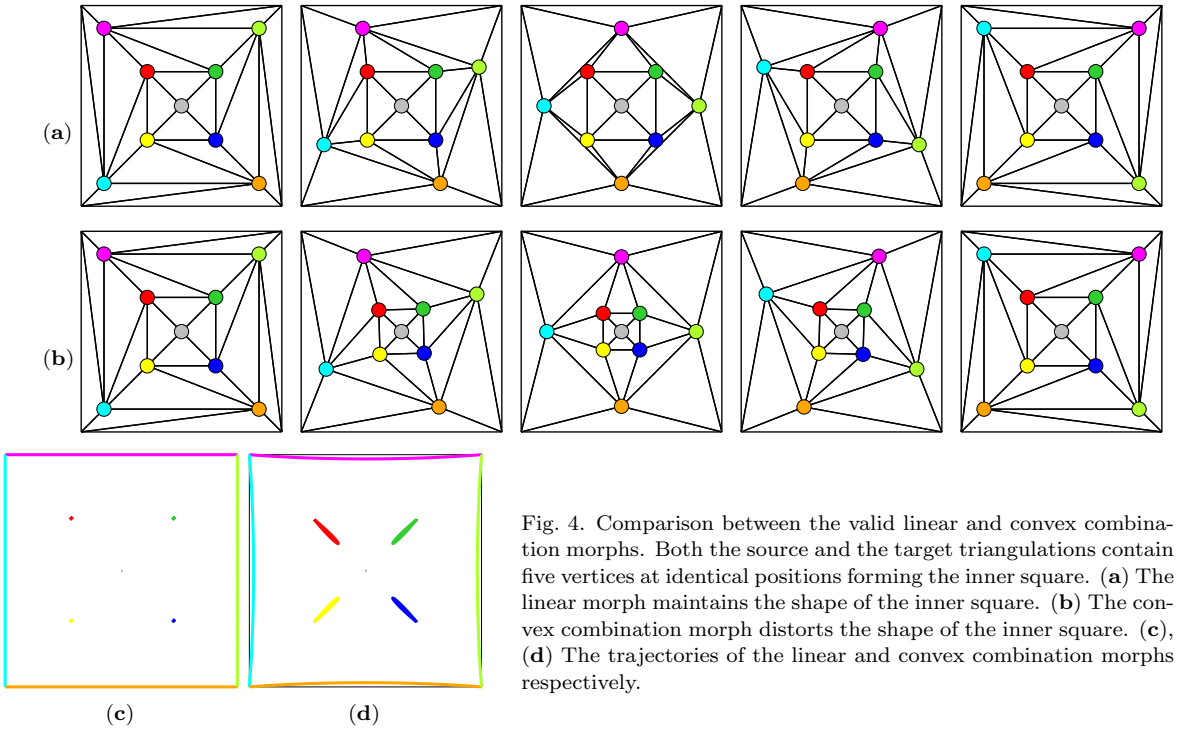


Fig. 4. Comparison between the valid linear and convex combination morphs. Both the source and the target triangulations contain five vertices at identical positions forming the inner square. (**a**) The linear morph maintains the shape of the inner square. (**b**) The convex combination morph distorts the shape of the inner square. (**c**), (**d**) The trajectories of the linear and convex combination morphs respectively.

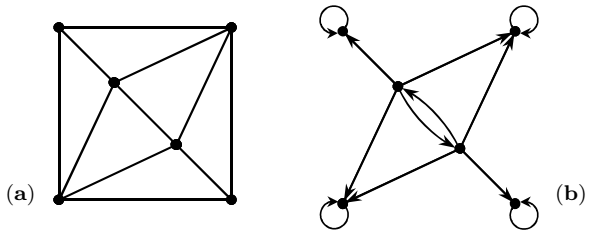


Fig. 5. A triangulation viewed as a graph of states with transitions: (**a**) a triangulation; (**b**) the corresponding transition graph.

The barycentric coordinates of the interior vertices of $\mathcal{T}$ may be viewed as probabilities of transitions in G. To each directed edge $(i \to j) \in E(G)$, $i \neq j$, assign a probability of transition from state i to state j to be the corresponding barycentric coordinate of vertex i with respect to vertex j in $\mathcal{T}$. To the loop edges $(i \to i) \in E(G)$ assign unit probability. Since barycentric coordinates sum to unity, G has legal transition probabilities.

Consequently, the neighborhood matrix of $\mathcal{T}$ is a stochastic matrix describing the transition process in G, known as a *Markov chain*. Let $q = (q_1, \ldots, q_N)$ be a vector of probabilities, where q_i is a probability to be in state i. The neighborhood matrix A, being a stochastic matrix, describes the probabilities to be in one of the N states after a single transition. Namely, $A \cdot q$ is the vector of probabilities after a single transition, and the stochastic matrices $A^2, A^3, \ldots$ describe the probabilities after a number of transitions.

We are interested in A^m when m approaches infinity. For states corresponding to the boundary vertices, our transition graph G has probability 1 to stay in these states. These states are *absorbing states* of the Markov chain. From every state in G corresponding to an interior vertex there exists a path to the absorbing states. Therefore $\lim_{m\to\infty} A^m$ exists and the asymptotic probability of being in a non-absorbing state is zero (after $m \to \infty$ transitions from any state). Denote $\lim_{m\to\infty} A^m$ by A^∞. The entries of A^∞ are:

$$A^\infty(i,j) = \begin{cases} 0, & \text{if } 1 \le i \le n, \quad 1 \le j \le n, \quad \text{(non-absorbing states)} \\ a_{i,j} > 0, & \text{if } 1 \le i \le n, \quad n < j \le N, \quad \text{(absorbing states)} \\ A(i,j), & \text{otherwise.} \end{cases}$$

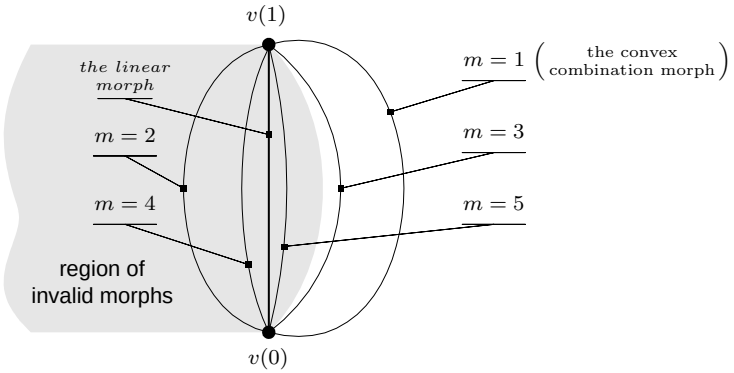


Fig. 6. Trajectories of a vertex v during morphs generated with different powers m of neighborhood matrices. Note that the morphs corresponding to even powers are even worse ("more invalid") than the linear morph, since they are more distant from the guaranteed valid convex combination morph ($m = 1$) than the linear morph.

Thus, the matrix A^∞ has the following form:

$$A^\infty = \left[\begin{array}{c|c} 0 & A_B^\infty \\ \hline 0 & I \end{array}\right] \tag{13}$$

Now, we wish to show that a morph generated using (12) is linear when m approaches infinity. First note that $A(t)$ has the form of (13), since A_0^∞ and A_1^∞ have this form. The equation $A(t) \cdot x(t) = x(t)$ may be written as:

$$(A_I(t) - I) \cdot x_I(t) + A_B(t) \cdot x_B = 0,$$

as in (10). Since $A(t)$ has the form of (13), $A_I(t) = 0$, and $x_I(t) = A_B(t) \cdot x_B$ for $0 \le t \le 1$:

$$\begin{aligned} x_I(t) &= A_B(t) \cdot x_B \\ &= [(1-t) \cdot A_B(0) + t \cdot A_B(1)] \cdot x_B \\ &= (1-t) \cdot A_B(0) \cdot x_B + t \cdot A_B(1) \cdot x_B \\ &= (1-t) \cdot x_I(0) + t \cdot x_I(1) \end{aligned}$$

So, the generated morph is linear when m approaches infinity. But it is known that the linear morph is not always a valid morph. More precisely, while A_0 and A_1 are valid neighborhood matrices, A_0^∞ and A_1^∞ are not necessarily such. In general, for $m > 1$, A^m is not a valid neighborhood matrix. Therefore the point sequence $\wp_t$ generated using $A(t)$ is not necessarily compatible with $\mathcal{T}_0$ (and $\mathcal{T}_1$). Nevertheless, the following two conjectures allow to obtain a morph that is closer to the linear morph than the convex combination morph.

Conjecture 3.4 *If the linear morph is valid, namely, all $\wp(t)$, $0 \le t \le 1$, are compatible with $\mathcal{T}_0$ (and $\mathcal{T}_1$), then for all odd $m \in \mathbb{N}$, $A(t)$ in (12) defines a valid morph, which approaches the linear morph as m increases.*

Conjecture 3.5 *If the linear morph is invalid, then there exists a $M \ge 1$ such that for all odd $m \le M$, $A(t)$ in (12) defines a valid morph, which approaches the linear morph as m increases; and for all $m > M$ the resulting morph is invalid.*

In order to obtain a morph that is closer to the linear morph than the convex combination morph, powers of neighborhood matrices are used. Higher powers ($m > 1$) result in morphs which approach the linear one. In practice, odd powers tend to produce valid morphs, while even powers do not. In fact, even powers generate morphs, which are worse ("more invalid") than the linear one, since they are more distant from the convex combination morph ($m = 1$) than the linear morph. Figure 6 demonstrates the common behavior of trajectories of an interior vertex for various powers of neighborhood matrices and Figure 7 shows this on a concrete example. Conjecture 3.5 allows to choose the maximum m, namely $m = M$ that guarantees a valid morph. A simple algorithm to find M sequentially checks morphs for every $m > 1$, incrementing m by 2. The number of morphs checked may be significantly reduced to

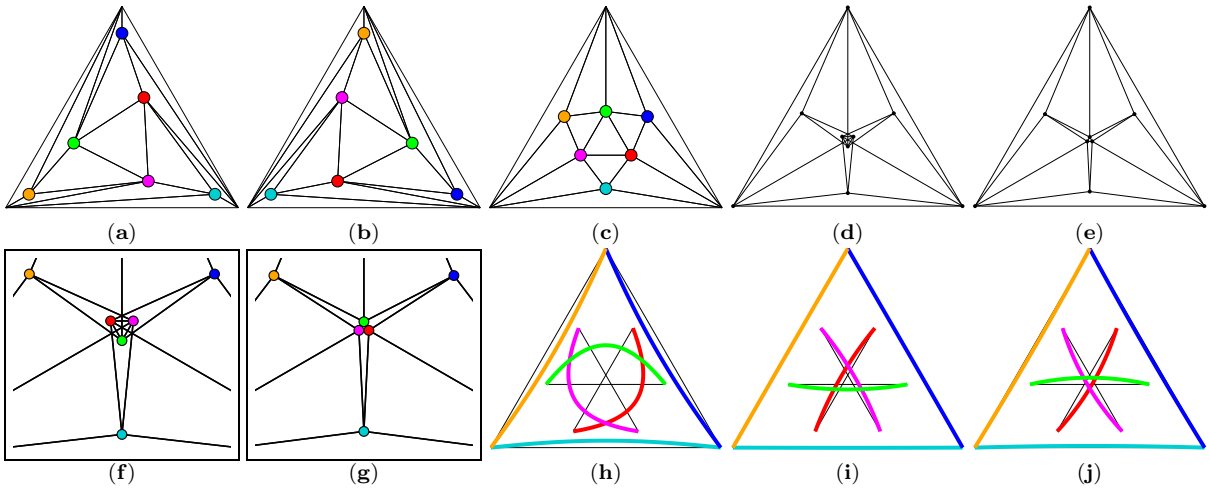


Fig. 7. Morphs generated by raising neighborhood matrices to various powers. (**a**), (**b**) The source and the target triangulations. Correspondence is color coded. (**c**) The convex combination morph at $t = 0.5$. (**d**) An invalid morph is generated by raising neighborhood matrices to power $m = 2$ ($t = 0.5$). (**e**) A valid morph is generated by raising neighborhood matrices to power $m = 3$ ($t = 0.5$). (**f**), (**g**) Zoom in on the triangulations in (d) and (e). (**h**) Trajectories of the convex combination morph. (**i**), (**j**) Trajectories of morphs generated by raising neighborhood matrices to power $m = 2$ and $m = 3$ respectively. Note the positions of the trajectories relative to the straight lines (the linear morph).

$O(\log M)$. First, we find an upper bound m_{max} by doubling m until the morph is invalid. Then, the resulting M is found by binary search in the interval $[\frac{m_{max}}{2}, m_{max}]$.

Furthermore, a morph that is even closer to the linear morph than a morph defined by $m = M$ may be obtained. Consider the following definition of $A(t)$:

$$A(t) = (1-t) \cdot \left[(1-d) \cdot A_0^m + d \cdot A_0^{m+2}\right] + t \cdot \left[(1-d) \cdot A_1^m + d \cdot A_1^{m+2}\right] \tag{14}$$

This equation averages the neighborhood matrices of the valid morph defined by $m = M$ with the neighborhood matrices of the invalid morph, defined by $m = M + 2$. Using power m for neighborhood matrices and a parameter d, equation (14) may be viewed as a morph with a non-integer power for neighborhood matrices $m + d \in \mathbb{R}$ ($m + d \geq 1$). In order to obtain a morph, which is the closest possible to the linear morph, the maximal parameter d may be chosen by binary search in the interval $[0, 1]$, verifying the morph validity at every step. See Figure 11(b) for a morph generated by this scheme.

3.4 Morphing with an Intermediate Triangulation

This section demonstrates how to find a morph between two triangulations $\mathcal{T}_0$ and $\mathcal{T}_1$ such that at a given time $t = t_m$ the morph interpolates a given triangulation $\mathcal{T}_m$. The triangulations $\mathcal{T}_0$, $\mathcal{T}_1$ and $\mathcal{T}_m$ are compatible and with identical boundaries. A naive solution is to find two convex combination morphs independently: the first—between $\mathcal{T}_0$ and $\mathcal{T}_m$, and the second morph—between $\mathcal{T}_m$ and $\mathcal{T}_1$. The problem with this is that while the two independent morphs are continuous and smooth, the combined morph will usually have a $\mathbf{C^1}$ discontinuity at the intermediate vertices.

In order to find a smooth morph, it is necessary to smoothly interpolate A_0, A_m and A_1 in $\mathbb{A}(\mathcal{T}_0)$. Consequently, the corresponding elements of the three matrices should be smoothly interpolated. Given three points $(0, \lambda_{i,j}(0))$, $(t_m, \lambda_{i,j}(t_m))$ and $(1, \lambda_{i,j}(1))$ in $\mathbb{R}^2$, it is necessary to find an interpolation $\lambda_{i,j}(t)$ for all $t \in [0, 1]$, see Figure 8. Since the entries of the matrices are barycentric coordinates, the interpolation must satisfy $0 \leq \lambda_{i,j}(t) \leq 1$. An interpolation within the bounded region $[0, 1] \times [0, 1]$ may be found as a piecewise Bézier curve, since any Bézier curve is located in the convex hull of its control points.

An important point is that interpolations for the matrix entries are performed independently. But every row i, $1 \leq i \leq n$, of $A(t)$, being barycentric coordinates of the interior vertex i, should sum to unity. Due to the independent interpolations, this might not be the case. Normalizing the elements of each row can solve this problem. The normalized entry $\overline{\lambda}_{i,j}(t)$ is defined as follows:

$$\overline{\lambda}_{i,j}(t) = \frac{\lambda_{i,j}(t)}{\sum_{k=1}^{N} \lambda_{i,k}(t)} \tag{15}$$

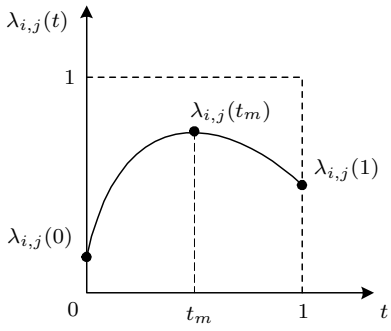


Fig. 8. An interpolation of three points $(0, \lambda_{i,j}(0))$, $(t_m, \lambda_{i,j}(t_m))$ and $(1, \lambda_{i,j}(1))$ when $0 \le t_m \le 1$ is in the bounded region $[0,1] \times [0,1]$.

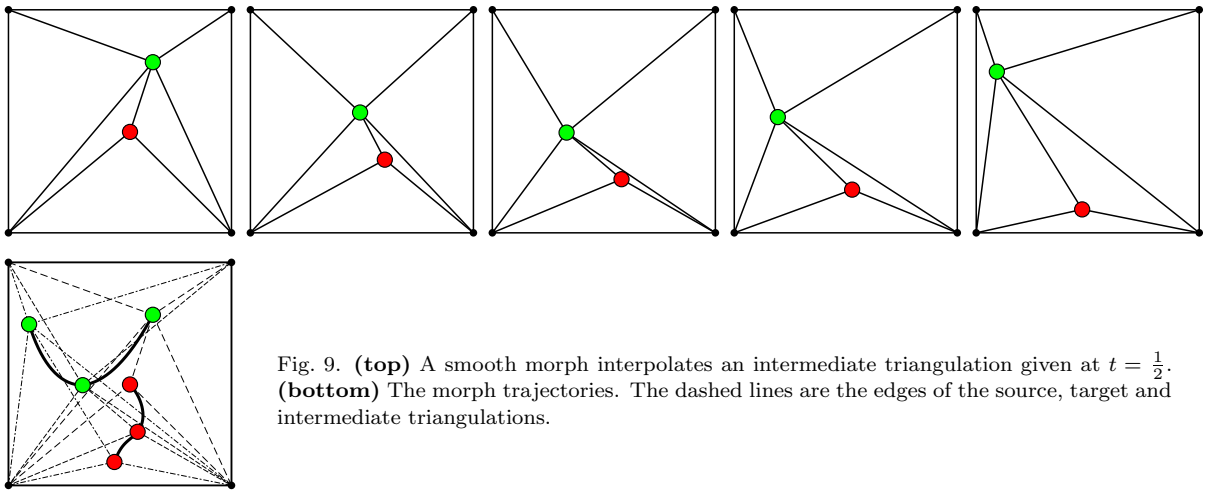


Fig. 9. **(top)** A smooth morph interpolates an intermediate triangulation given at $t = \frac{1}{2}$. **(bottom)** The morph trajectories. The dashed lines are the edges of the source, target and intermediate triangulations.

Since $\lambda_{i,j}(t)$ is smooth, the sum of $\lambda_{i,j}(t)$'s is also smooth. Therefore the normalized $\overline{\lambda}_{i,j}(t)$ is a smooth interpolation. See an example demonstrating a smooth morph in Figure 9.

4. MORPHING WITH LOCAL CONTROL

A well-behaved morphing scheme should have properties like those described in Section 3.2. Trajectories traveled by the interior vertices should be smooth and even (not jerky and not bumpy). It would be useful if the scheme would be linear-reducible. When the linear morph is invalid, the natural requirement is to generate a morph as close as possible to the linear one. It would also be useful to be able to control triangle areas in such a way that they transform naturally (uniformly) during the morph. This may help to prevent shrinking/swelling of triangles, that result in an unnatural-looking morph. The schemes presented in this section allow the control of trajectories of the interior vertices, triangle areas etc. in a local manner.

To find a morph between two triangulations $\mathcal{T}_0$ and $\mathcal{T}_1$ means to find a curve $A(t)$ for $0 \le t \le 1$ in $\mathbb{A}(\mathcal{T}_0)$ with endpoints in $\mathcal{A}(\mathcal{T}_0)$ and $\mathcal{A}(\mathcal{T}_1)$. We will do this by constructing each row of $A(t)$ (corresponding to each interior vertex) separately.

We define $\mathcal{T}'(\mathcal{G}', \wp')$ to be a subtriangulation of a triangulation $\mathcal{T}(\mathcal{G}, \wp)$ if $\mathcal{T}'$ is a valid triangulation, $\mathcal{G}'$ is a subgraph of $\mathcal{G}$ and the coordinates of the corresponding vertices of $\mathcal{T}'$ and $\mathcal{T}$ are identical. The triangulations $\mathcal{T}_0$ and $\mathcal{T}_1$ may be decomposed into n subtriangulations in the following manner: each interior vertex i, $1 \le i \le n$ corresponds to a subtriangulation that consists of the interior vertex, its neighbors and edges connecting these vertices, see Figure 10. A subtriangulation defined above is said to be a *star* denoted by $\mathcal{Z}_i$. Namely, every star $\mathcal{Z}_i$ corresponds to the interior vertex i, and $\bigcup_{i=1}^{n} \mathcal{Z}_i = \mathcal{T}$.

Let $\mathcal{Z}_i(0)$ when $1 \le i \le n$ be stars of the triangulation $\mathcal{T}_0$; stars $\mathcal{Z}_i(1)$ are defined analogously for $\mathcal{T}_1$. Clearly, $\mathcal{Z}_i(0)$ and $\mathcal{Z}_i(1)$ are two isomorphic triangulations, since they are the same subgraph of two isomorphic triangulations $\mathcal{T}_0$ and $\mathcal{T}_1$. Barycentric coordinates of the interior vertex in star $\mathcal{Z}_i$ with respect to the boundary vertices of that star are also barycentric coordinates of the interior vertex i

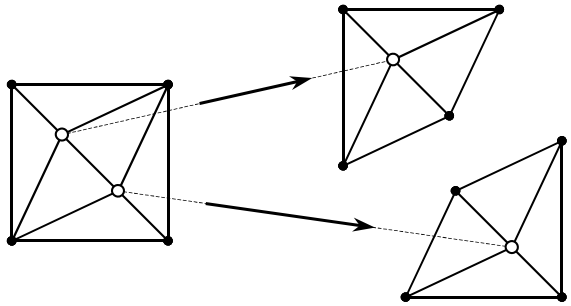


Fig. 10. A triangulation is decomposed into stars; each interior vertex defines a separate star.

in a triangulation $\mathcal{T}$ with respect to its neighbors. Thus all $\mathcal{Z}_i(0)$'s when $1 \leq i \leq n$ together define a neighborhood matrix A_0 in $\mathcal{A}(\mathcal{T}_0)$; $\mathcal{Z}_i(1)$ for $1 \leq i \leq n$ define A_1 respectively. In the same manner, we would like to define $A(t)$ for a specific t using stars $\mathcal{Z}_i(t)$, $1 \leq i \leq n$. The question is how to find $\mathcal{Z}_i(t)$ for $0 < t < 1$ such that it will define barycentric coordinates with some intermediate values between barycentric coordinates of $\mathcal{Z}_i(0)$ and $\mathcal{Z}_i(1)$. Obviously, a smoth morph of two stars $\mathcal{Z}_i(0)$ and $\mathcal{Z}_i(1)$ should suffice to obtain this. But only a smooth $A(t)$ will define a smooth morph between the triangulations. For that reason it is important to use a method to generate barycentric coordinates of the interior vertices that is at least $\mathbf{C^1}$-continuous, such as that described in Section 2.1. A morphing scheme that generates $A(t)$ by morphing separately the stars of two triangulations is said to be a *local scheme*.

There is a simple way to morph two stars $\mathcal{Z}(0)$ and $\mathcal{Z}(1)$. First, translate the two stars in such a way that the interior vertices of the both stars are at the origin. Then a morph may be defined by linear interpolation of the polar coordinates of the corresponding boundary vertices. One can find a proof for the correctness of this morph in [Shapira and Rappoport 1995; Floater and Gotsman 1999; Surazhsky 1999], where there are also recommendations on how to choose the polar coordinates in order to obtain a valid morph. Note that the validity of star morphs, based on the translation of interior vertices to the origin, depends only on how angle components of the polar coordinates of the boundary vertices vary during the morphs. Arbitrary variations in the radial direction of the boundary vertices do not affect the validity of the morph.

4.1 The Local Linear-Reducible Scheme

The local scheme morphs separately the corresponding stars of $\mathcal{T}_0$ and $\mathcal{T}_1$ and is based on the translation of the source and target stars to the origin. Two translated stars $\mathcal{Z}_i(0)$ and $\mathcal{Z}_i(1)$ are morphed in the following manner. If the linear morph of two stars is valid, we adopt it. Otherwise, an arbitrary valid morph is taken. It can be the morph that averages the polar coordinates of the boundary vertices, as described in Section 4; or translated trajectories of the boundary vertices during the convex combination morph. In the latter case the row corresponding to the star $\mathcal{Z}_i$ in $A(t)$ is equal to the i'th row of $A(t)$ generated by the convex combination morph.

The following theorem shows that this scheme is linear-reducible.

Theorem 4.1 *The linear morph of two triangulations $\mathcal{T}_0$ and $\mathcal{T}_1$ is valid iff the linear morphs of all component stars are valid.*

Proof. *Validity:* If the linear morph between $\mathcal{T}_0$ and $\mathcal{T}_1$ is valid, then any triangulation $\mathcal{T}(t)$ for $0 \leq t \leq 1$ is valid and compatible with $\mathcal{T}_0$ (and $\mathcal{T}_1$) and thus may be decomposed into valid stars. Each $\mathcal{Z}_i(t)$ when $1 \leq i \leq n$ is a valid morph between $\mathcal{Z}_i(0)$ and $\mathcal{Z}_i(1)$. If all morphs of the stars $\mathcal{Z}_i(t)$ are valid, then we have a legal neighborhood matrix function $A(t)$ for $0 \leq t \leq 1$, and thus $\mathcal{T}(t)$ is valid.

Linearity: Let $p_{i,j}$ be the coordinates of the vertex i in the star $\mathcal{Z}_j$. First, we prove that if the morph between $\mathcal{T}_0$ and $\mathcal{T}_1$ is linear then the morphs of all stars are linear. We have $p_i(t) = (1-t) \cdot p_i(0) + t \cdot p_i(1)$ for $0 \leq t \leq 1$. Every star $\mathcal{Z}_j(t)$ is translated in such a way that the interior vertex is at the origin. Thus,

$p_{i,j}(t) = p_i(t) - p_j(t)$. Combining both equations:

$$\begin{aligned} p_{i,j}(t) &= p_i(t) - p_j(t) \\ &= [(1-t)\cdot p_i(0) + t\cdot p_i(1)] - [(1-t)\cdot p_j(0) + t\cdot p_j(1)] \\ &= (1-t)\cdot [p_i(0) - p_j(0)] + t\cdot [p_i(1) - p_j(1)] \\ &= (1-t)\cdot p_{i,j}(0) + t\cdot p_{i,j}(1) \end{aligned}$$

For the opposite direction, we prove that if the morphs of all stars are linear then the morph between $\mathcal{T}_0$ and $\mathcal{T}_1$ is linear. The linear morphs of the stars imply that:

$$p_{i,j}(t) = (1-t)\cdot p_{i,j}(0) + t\cdot p_{i,j}(1) \qquad \text{for } 0 \le t \le 1. \tag{16}$$

Let $\mathcal{T}(t)$ be the linear morph between $\mathcal{T}_0$ and $\mathcal{T}_1$, namely,

$$p_i(t) = (1-t)\cdot p_i(0) + t\cdot p_i(1) \qquad \text{for } 0 \le t \le 1. \tag{17}$$

It remains to show that $A(t)$, defined by the stars $\mathcal{Z}_j(t)$ for $1 \le j \le n$, satisfies $A(t) \in \mathcal{A}(\mathcal{T}(t))$. We will show that every star $\mathcal{Z}_j(t)$ defines the same barycentric coordinates of the interior vertex i as the corresponding star j of $\mathcal{T}(t)$, and thus $A(t) \in \mathcal{A}(\mathcal{T}(t))$. Clearly, $\mathcal{Z}_j(t)$ is isomorphic with the corresponding star j of $\mathcal{T}(t)$. The following states that the vertex coordinates of $\mathcal{Z}_j(t)$ are translated coordinates of the corresponding star j of $\mathcal{T}(t)$. Due to the initial translations of the interior vertices to the origin:

$$\begin{aligned} p_{i,j}(0) &= p_i(0) - p_j(0) \\ p_{i,j}(1) &= p_i(1) - p_j(1) \end{aligned} \tag{18}$$

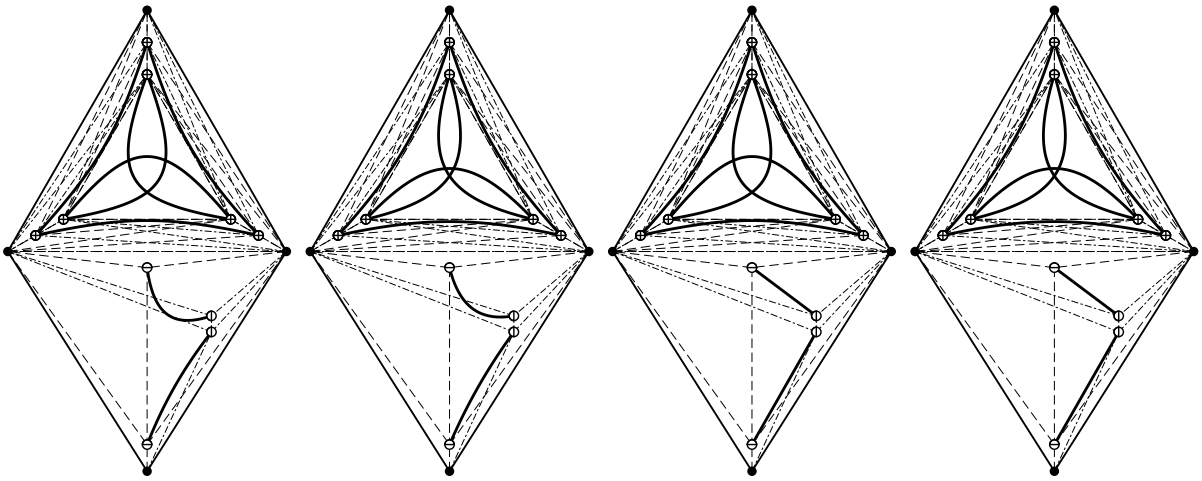
We can now express (16) as:

$$\begin{aligned} p_{i,j}(t) &= (1-t)\cdot p_{i,j}(0) + t\cdot p_{i,j}(1) \\ &= (1-t)\cdot [p_i(0) - p_j(0)] + t\cdot [p_i(1) - p_j(1)] \\ &= [(1-t)\cdot p_i(0) + t\cdot p_i(1)] - [(1-t)\cdot p_j(0) + t\cdot p_j(1)] \\ &= p_i(t) - p_j(t) \end{aligned} \tag{19}$$

Hence, after the translation of $p_j(t)$ the coordinates of every vertex in the star $\mathcal{Z}_j(t)$ are equal to the coordinates of the corresponding vertex in $\mathcal{T}(t)$. Since barycentric coordinates are invariant to a translation (as a special case of affine transformations), the stars $\mathcal{Z}_j(t)$ for $1 \le j \le n$ define $A(t) \in \mathcal{A}(\mathcal{T}(t))$. □

This work presents two linear-reducible schemes: the scheme described in Section 3.3 and the scheme introduced in this section. It is important to emphasize the principal difference between these two schemes. The first scheme approaches the linear morph using neighborhood matrices raised to a power. This approaching significantly affects *all* trajectories of the interior vertices and is a *global* approach to the linear morph. It allows to choose a degree of approximation to the linear morph by specifying the power of the neighborhood matrices. However, even the morph closest to the linear morph does not allow each individual trajectory to be as 'linear' as possible. The *global* convergence may be blocked by a single problematic trajectory (which invalidates the morph), preventing the others from being straightened further, see Figures 11(**a**) and 11(**b**). On the other hand, the local linear-reducible scheme, morphing the component stars separately, may affect a group of trajectories of adjacent vertices almost independently of other trajectories of the morph. However, the local scheme does not attempt to approximate the linear morph for stars for which the linear morph is invalid. Thus, vertices of triangulation regions that cannot be morphed linearly have trajectories similar to those generated by the convex combination morph, and vertices of regions that may be morphed linearly have trajectories very close to straight lines, see Figure 11(**c**). Knowing the properties of both the linear-reducible schemes, it is possible to choose the most suitable for specific triangulations and specific applications.

These two schemes may also be combined to obtain a morph that is closer to the linear morph than a morph generated separately by each of the schemes. First, the scheme of Section 3.3 is used to generate a valid morph $\mathcal{T}_P(t)$ with a maximal power for neighborhood matrices. Then the scheme of this section is applied, morphing each of the stars separately. For stars for which the linear morph is invalid, corresponding trajectories from $\mathcal{T}_P(t)$ are used. For the rest of the stars, the linear morph is used. The resulting morph is valid, since the morphs of all component stars are valid. The morph in Figure 11(**d**) was generated using this combined scheme.



(**a**) (**b**) (**c**) (**d**)

Fig. 11. Trajectories of various morphs approaching the linear morph. The dashed lines are the edges of the source and target triangulations. (**a**) Trajectories of the convex combination morph. (**b**) Trajectories of the valid morph generated by raising neighborhood matrices to power $m = 1.3$. All trajectories are closer to straight lines than the trajectories of the convex combination morph. However, the two lower trajectories could still potentially be straight lines without affecting the validity of the morph. (**c**) Trajectories of the valid morph generated using the local linear-reducible scheme. The two lower trajectories are straight lines, but the rest are identical to the corresponding trajectories of the convex combination morph. (**d**) Trajectories of the valid morph generated by combination of two linear-reducible schemes. The two lower trajectories are linear. The rest approach straight lines similar to the corresponding trajectories of the morph with power 1.3.

4.2 Testing Validity of the Linear Morph

The linear-reducible scheme, described in the previous section, morphs an individual star linearly only if the linear morph is valid. A natural question is how to determine whether the linear morph between $\mathcal{Z}(0)$ and $\mathcal{Z}(1)$ will be valid or not. Clearly, the naive test which verifies whether $\mathcal{Z}(\frac{1}{2})$ is a valid triangulation is not enough. To verify whether $\mathcal{Z}(t)$ is a valid triangulation for all $0 \leq t \leq 1$ is impossible in practice, since $[0, 1]$ is a continuum. Appendix A presents a robust and fast (linear time complexity) method to perform the test. This method can also be applied to check the validity of linear morphs for general triangulations. According to Theorem 4.1 it is sufficient to check the validity of the linear morphs for all corresponding stars of the two triangulations. The complexity of this test is $O(V(\mathcal{T}))$, namely, linear in the size of the triangulations.

4.3 Improving Triangle Area Behavior

This section describes a method for improving the behavior of the triangle areas during the morph. The triangle areas do not always evolve uniformly during the morph when using the methods described in the previous sections. In fact, the triangle areas may evolve linearly *only* when the triangulations have a single interior vertex. For a specific triangle i, we would like its area, denoted by $\mathcal{S}_i$, to behave for $0 \leq t \leq 1$ like:

$$\mathcal{S}_i(t) = (1-t) \cdot \mathcal{S}_i(0) + t \cdot \mathcal{S}_i(1) \tag{20}$$

This cannot be satisfied for all triangles of the triangulation for all $0 < t < 1$. Consider the following equation for the area of a triangle with vertices i, j and k:

$$\mathcal{S}(t) = \frac{1}{2} \begin{vmatrix} 1 & 1 & 1 \\ x_i(t) & x_j(t) & x_k(t) \\ y_i(t) & y_j(t) & y_k(t) \end{vmatrix} \tag{21}$$

Areas of triangles with two boundary vertices are transformed uniformly only when the third vertex travels linearly with a constant velocity. Areas of triangles with a single boundary vertex are quadratically (not uniformly) transformed, since the two non-boundary vertices travel linearly with constant velocities, by (21).

The problem of the triangle area improvement may be formulated as follows. Denote by $\overline{\mathcal{S}}_i(t)$ the

desired area of a triangle i that evolves linearly:

$$\overline{\mathcal{S}}_i(t) = (1-t)\cdot\mathcal{S}_i(0) + t\cdot\mathcal{S}_i(1). \tag{22}$$

Thus, a morph between two triangulations should minimize a cost function such as:

$$\sum_i \left|\overline{\mathcal{S}}_i(t) - \mathcal{S}_i(t)\right|, \quad \text{for all } t\in[0,1]. \tag{23}$$

We now show how to improve the triangle area evolution using the local scheme, such that the resulting morph is at least closer to (23) than the convex combination morph.

Since it is difficult to improve the triangle areas for the entire triangulation, the improvement may be done separately for the stars of the triangulation. This is performed after a morph of a specific star is defined. To preserve the validity of the morph, φ-components of the boundary vertices are preserved. The improvement is done by a variation of the boundary vertices in the radial direction relative to the origin.

First, we consider an improvement such that all triangles within the star have exactly the desired areas, namely, the area of each triangle is $\overline{\mathcal{S}}_i(t)$. This approach, however, has a serious drawback. While every triangle has its desired area within the star, its shape significantly differs from the shape it assumes in the entire triangulation. Furthermore, since a triangle belongs to a number of stars, its shapes in the different stars might contradict each other considerably. Therefore the resulting morph is very unstable. The trajectories that the interior vertices travel are tortuous. The triangle areas are far from uniform and hardly better than those generated by the convex combination morph.

All this means that the evolution of the triangle areas within the stars must also take into account the triangle shapes. For a specific triangle i, one of its vertices is the interior vertex of the star, that is placed at the origin for $0\le t\le 1$. The angle adjacent to the interior vertex cannot be changed, because it may affect the validity of the morph. Therefore an improvement of the triangle area is achieved by a variation of the lengths of its two edges adjacent to the interior vertex. Every edge adjacent to the interior vertex belongs to exactly two triangles. Consequently, the length of the edge after an improvement for one triangle does not always coincide with the length of the edge within the second triangle. We improve the triangle areas separately for each triangle, and the resulting length of the edge is the average of the two lengths.

We propose a simple method that improves the area of a single triangle and also preserves the triangle shape. This method changes the positions of the triangle vertices in the radial direction such that the triangle area evolves linearly and the lengths of the radial edges maintain the proportions they would have had, had the edge lengths evolved linearly. Let a and b be the lengths of the edges, and θ be the angle between them. The area of the triangle is:

$$\mathcal{S} = \frac{1}{2}\,a\,b\,\sin(\theta) \tag{24}$$

Denote by $a(t)$ and $b(t)$ the lengths of the edges as they evolve linearly:

$$a(t) = (1-t)\cdot a(0) + t\cdot a(1), \qquad b(t) = (1-t)\cdot b(0) + t\cdot b(1) \tag{25}$$

The resulting $\overline{a}(t)$ and $\overline{b}(t)$ are the lengths of the edges such that the triangle area is $\overline{\mathcal{S}}(t)$ defined by (22). In order to find $\overline{a}(t)$ and $\overline{b}(t)$ it is necessary to solve the following system of equations with the unique solution.

$$\begin{cases} \overline{a}(t)\cdot\overline{b}(t) = \dfrac{2\,\overline{\mathcal{S}}(t)}{\sin(\theta(t))}, & \text{to satisfy } \overline{\mathcal{S}} = \frac{1}{2}\,\overline{a}(t)\,\overline{b}(t)\,\sin(\theta(t)) \\ \dfrac{\overline{a}(t)}{\overline{b}(t)} = \dfrac{a(t)}{b(t)}, & \text{for preserving the relation between the edges.} \end{cases} \tag{26}$$

See an example of a morph generated using this method in Figure 12.

5. EXPERIMENTAL RESULTS: MORPHING POLYGONS

In practice, morphing is performed more frequently on planar figures than on planar triangulations. Luckily, many types of planar figures may be embedded in triangulations as a subset of the triangulation edges. Thus the problem of morphing planar figures may be reduced to that of morphing triangulations, and the edges not part of the figure are ignored in the resulting morph. Two popular cases of planar figures are planar *polygons* and planar *stick figures*. The former is a cycle of edges, and the latter a

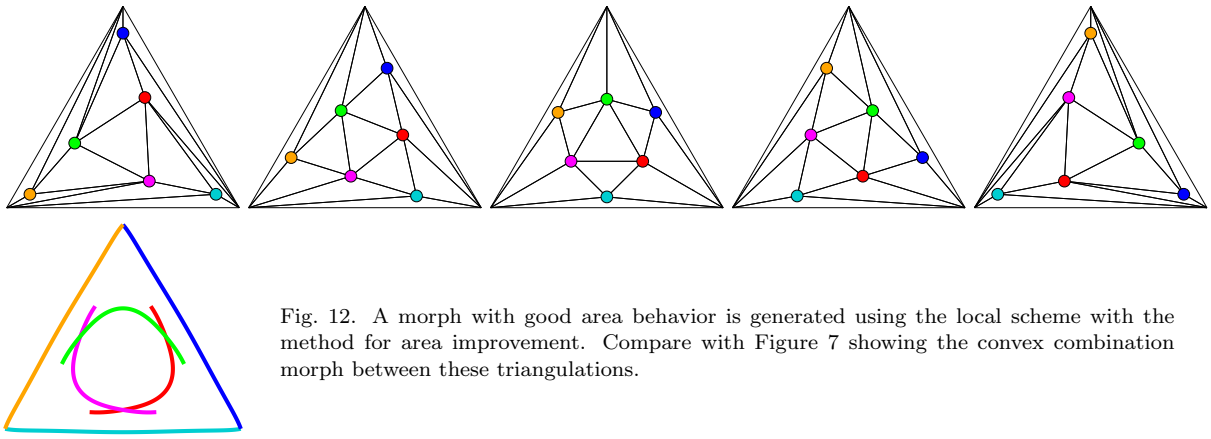


Fig. 12. A morph with good area behavior is generated using the local scheme with the method for area improvement. Compare with Figure 7 showing the convex combination morph between these triangulations.

connected straight line graph. Embedding these types of figures in a triangulation in an efficient manner is a difficult problem in itself, and has been treated separately by us in [Gotsman and Surazhsky 2001] and [Surazhsky and Gotsman 2001]. Here we will assume that these embeddings have been done, and investigate the effect of the various triangulation morphing techniques described in the previous sections on the results.

Figure 13 shows morphs between two polygons—the shapes of the two letters U and S. Figure 14 shows morphs between two stick figures—the shapes of a scorpion and a dragonfly. These examples have been embedded in planar convex *tilings* [Floater and Gotsman 1999] (the faces are not necessarily triangles), for which all the theory developed in this paper holds too. Being convex, these tilings may be easily triangulated if needed. In both examples the linear morph self-intersects (Figure 13(a) and Figure 14(a)). The convex combination morph is valid, but has an unpleasant behavior (Figure 13(b) and Figure 14(b)). The local scheme, which averages the polar coordinates of star boundary vertices, provides good results when parts of the figures should be rotated during the morph. Figure 13(c) and Figure 14(c) demonstrate that this results in a rather natural morph. Unfortunately, the morph of Figure 13(c) undergoes some exaggerated shrinking. This may be avoided by using the local scheme with area improvement, as in Figure 13(d).

Figure 14(d) shows how to approach the linear morph while still preserving the morph validity, by using the scheme that raises the neighborhood matrices to power 17. Note that the tail travels a path similar to that of the linear morph, but by shrinking avoids self-intersection. Also note that some parts of the rest of the triangulations self-intersect, since the trajectories of *all* interior vertices approach the linear ones. But since we are interested only in the validity of the stick figure itself, we can ignore the behavior of other edges.

6. CONCLUSION

We have described a robust approach for morphing planar triangulations. This approach always yields a valid morph, free of self-intersections, based on the only known analytical method for generating morphs guaranteed to be valid [Floater and Gotsman 1999]. The approach, having many degrees of freedom, may be used to produce a variety of morphs, and, thus, can be tuned to obtain morphs with many desirable characteristics.

6.1 Discussion

Morphing thru an intermediate triangulation poses the following interesting problem. Find a morph through an intermediate triangulation at a given time t_m, in which only a *subset* of the interior vertices have prescribed positions. This contrasts with the scenario treated in Section 3.4, where *all* vertices of the intermediate triangulation have prescribed positions. While constraining only a subset of the vertices might seem easier than constraining all the vertices, it is actually more difficult, since if all vertices are constrained, the user supplies a complete geometry compatible with the triangulation. Supplying only part of the vertex geometry leaves the algorithm the task of finding compatible geometries for the other vertices, which is difficult, especially since they might not exist.

This (static) problem is interesting in its own right, and has applications in the generation of texture coordinates. It is only recently that Eckstein et al. [2001] have shown how to solve this problem by the introduction of (extraneous) Steiner vertices. The solution with a minimal number of Steiner vertices,

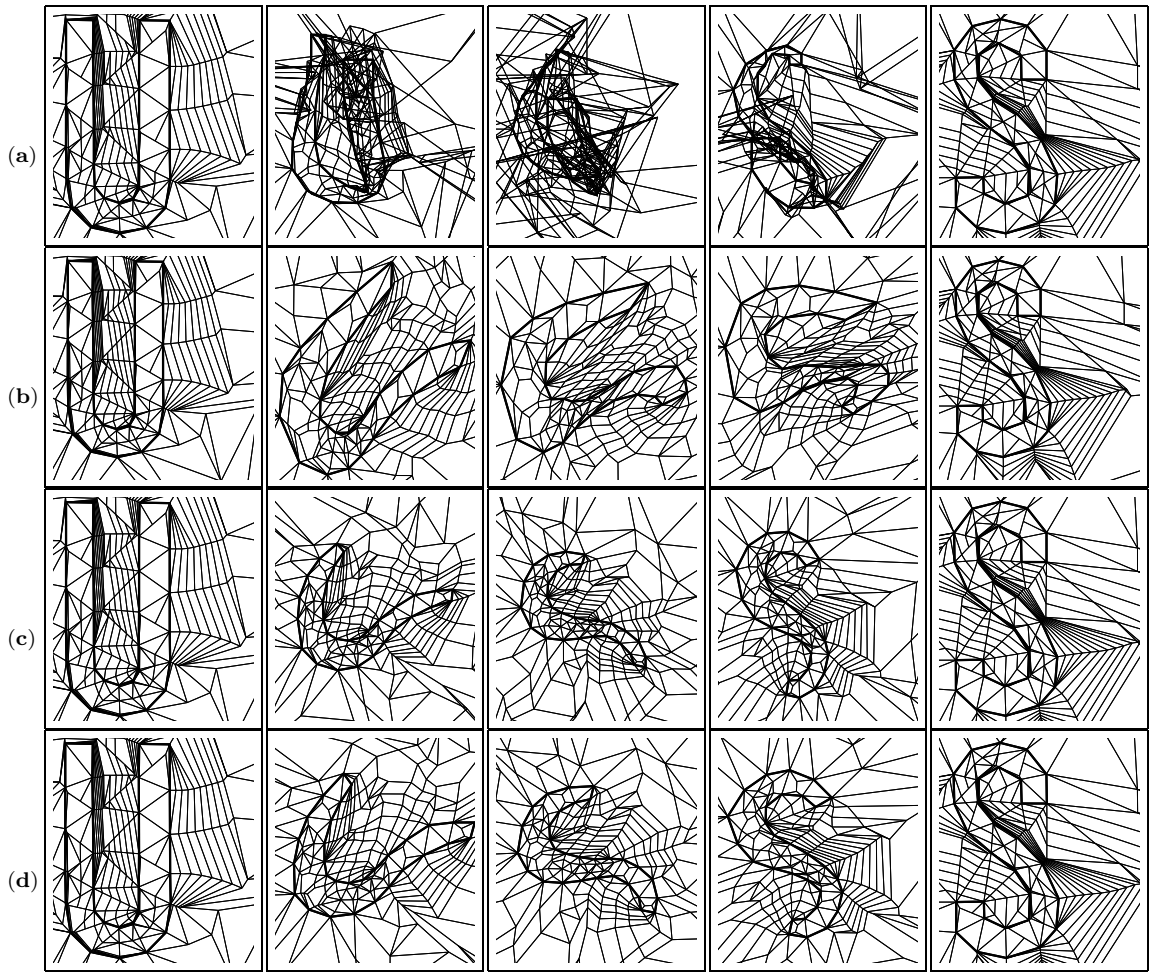


Fig. 13. Morphing simple polygons—the shapes of two letters S and U: (**a**) The linear morph is invalid — the polygon self-intersects. (**b**) The convex combination morph is valid, but unnatural. (**c**) Morph generated by the local scheme that averages polar coordinates. It behaves naturally, accounting for the rotation of the lower part of the S, but shrinks in an exaggerated manner. (**d**) Morph generated by the local scheme with area improvement. It is similar to the morph in (c), but with much less shrinking of the shape.

and in particular, with none when it is possible, is still open. In general, the main difficulty stems from the fact that our morphing techniques use neighborhood matrices, which always result in a *global* solution to the morphing problem, making it virtually impossible to precisely control an individual vertex location (or trajectory).

In Section 3.3, two conjectures are used to generate a morph that approaches the linear morph. Numerous examples support these conjectures, but a proof still eludes us. Since the matrices used to generate morphs by that method are not legal neighborhood matrices, the proof requires more a profound comprehension of the method.

Section 4.3 presents a heuristic for improving the evolution of triangle areas. Further analysis of the correlation between vertex trajectories as well as triangle area behavior within the stars and behavior of these elements in the triangulation, may provide insight to more successful heuristics, perhaps even some optimal approximation to the desired triangle areas.

It is important to make the techniques presented in this work applicable to real-world scenarios. As mentioned in Section 5, the techniques have already been applied to morph simple planar polygons and stick figures by embedding them in triangulations. In practice, the triangulations are built *around* them. For example, the triangulations in which simple polygons are embedded are constructed by compatibly triangulating the interior of the polygons and an annular region in the exterior of the polygon between the polygon boundary and a fixed convex enclosure. See [Gotsman and Surazhsky 2001; Surazhsky and Gotsman 2001] for more details. These works have yet to be extended to morph planar figures with arbitrary (e.g. disconnected) topologies.

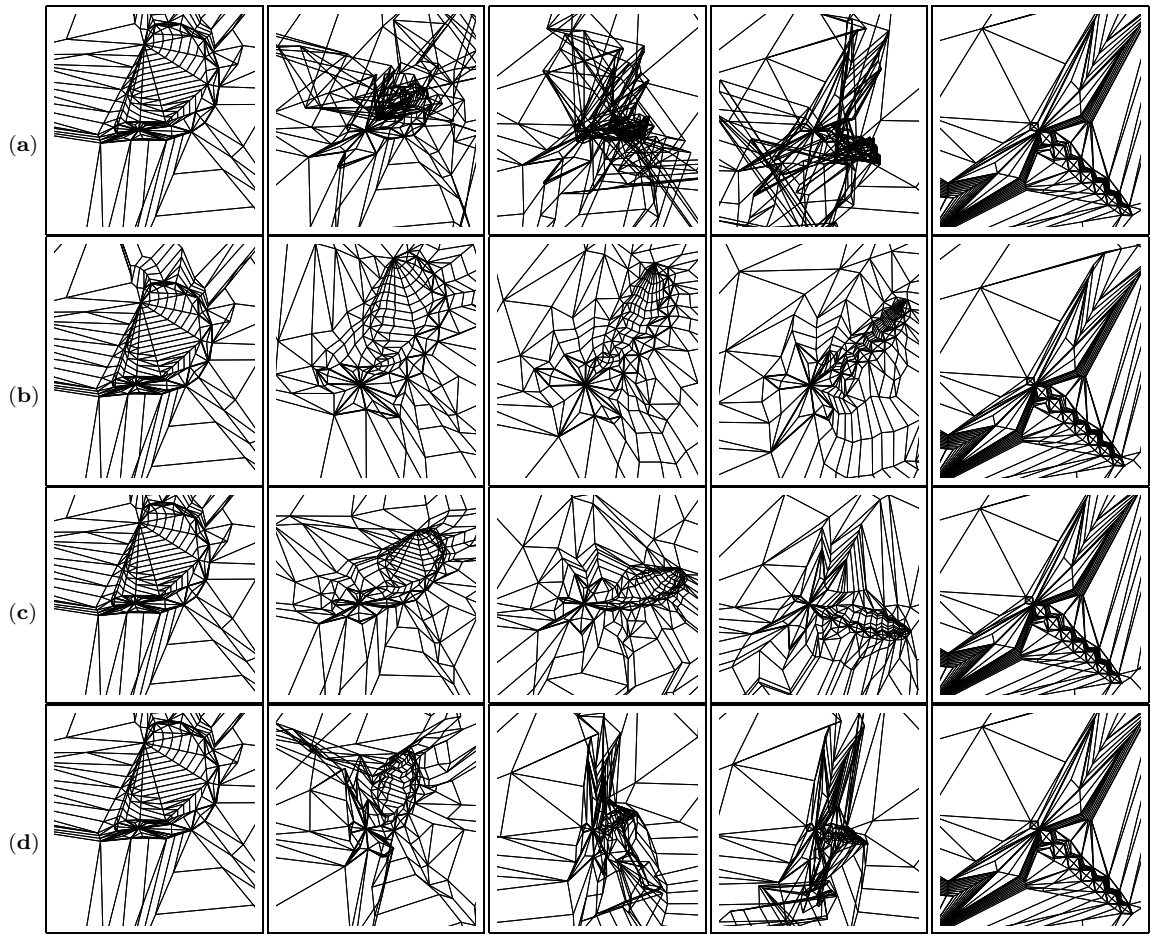


Fig. 14. Morphing between figures of a scorpion and a dragonfly: (**a**) The linear morph is invalid — the figure self-intersects. (**b**) The convex combination morph is valid, but unnatural. (**c**) Morph generated by the local scheme that averages polar coordinates. It behaves naturally, accounting for the rotation of the tail. (**d**) Morph generated by the raising the neighborhood matrices to power 17. Note that the tail travels a path similar to that of the linear morph (a), but it shrinks in order to avoid self-intersection.

Morphing triangulations is usually useful as a means to morph planar figures, and in that case the fixed convex boundary is not restrictive. However, if the objective is to actually morph two triangulations (e.g. for image warping), then a fixed common convex boundary might be restrictive. Fortunately, using the methods of [Gotsman and Surazhsky 2001; Surazhsky and Gotsman 2001], it is possible to overcome this by embedding the source and target triangulations with *different* boundaries, in two larger triangulations with a common fixed boundary. In practice this is done by compatibly triangulating the annulus between the original and new boundary, possibly introducing Steiner vertices. See Figure 15.

6.2 Future Work

A challenging research subject would be to extend the techniques of this work to three dimensions, certainly, starting from an extension of [Tutte 1963]. Furthermore, it would be interesting to address the problems in [Aronov et al. 1993; Babikov et al. 1997; Souvaine and Wenger 1994; Etzion and Rappoport 1997] for 3D.

APPENDIX

A. TESTING VALIDITY OF THE LINEAR MORPH OF A STAR

Let v_0 be the interior vertex of the stars with degree d. The corresponding boundary vertices are indexed without loss of generality as $v_1, \ldots, v_d$ in a counterclockwise order with respect to the interior vertex. Let $\big(x_1(t), y_1(t)\big), \ldots, \big(x_d(t), y_d(t)\big)$ be the boundary vertex coordinates during the linear morph. We denote by $\big(\rho_i(t), \varphi_i(t)\big)$ polar coordinates of the vertex i. Let $\theta_i(t) = \varphi_{i+1}(t) - \varphi_i(t)$ be the angle of a triangle i adjacent to the interior vertex. Note that all calculations with indices are performed modulo d. It is

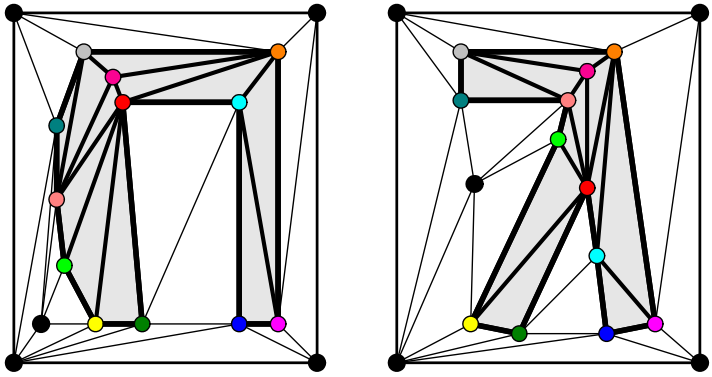


Fig. 15. Embedding two compatible triangulations (shaded regions) with different boundaries into larger triangulations with a common fixed boundary.

assumed that the polar coordinates of the vertices for $t = 0$ and $t = 1$ are chosen in such a way that $0 < \theta_i(0) < \pi$ and $0 < \theta_i(1) < \pi$. Now, it is necessary to check that the linear morph preserves the triangle orientations, namely, it should be verified that $0 < \theta_i(t) < \pi$ for $0 \le t \le 1$. To verify this, it is sufficient to check the extrema of $\theta_i(t)$ on $[0, 1]$. The extremum points may be found by solving the equation $\theta_i'(t) = 0$.

For notational simplicity, we denote $i = a$ and $i + 1 = b$. Due to the linear traversals of the vertices, we have:

$$x_a(t) = (1-t) \cdot x_a(0) + t \cdot x_a(1) \qquad y_a(t) = (1-t) \cdot y_a(0) + t \cdot y_a(1) \tag{27}$$

$$x_b(t) = (1-t) \cdot x_b(0) + t \cdot x_b(1) \qquad y_b(t) = (1-t) \cdot y_b(0) + t \cdot y_b(1) \tag{28}$$

The φ-component of the polar coordinates is expressed as:

$$\varphi(t) = \operatorname{sign}(y(t)) \cdot \arccos\left(\frac{x(t)}{\sqrt{x^2(t) + y^2(t)}}\right), \qquad \text{where } \operatorname{sign}(z) = \begin{cases} +1, & z \ge 0 \\ -1, & z < 0 \end{cases} \tag{29}$$

The next step is to derive $\varphi_b'(t) - \varphi_a'(t)$. But the $\operatorname{sign}(z)$ function is not convenient for the derivation. To overcome this problem we perform some substitutions for $\varphi(t)$. Since $|\varphi_a(1) - \varphi_a(0)| < \pi$ we can rotate both vertices $v_a(0)$ and $v_a(1)$ by the same angle ω_a round the origin such that the vertices are placed in the upper half plane, namely, the y-components of the vertices are positive. We denote the rotated coordinates by $(\tilde{x}, \tilde{y})$ with the polar φ-component $\tilde{\varphi}$. Thus, we have

$$\varphi_a(0) = \tilde{\varphi}_a(0) - \omega_a, \qquad \varphi_a(1) = \tilde{\varphi}_a(1) - \omega_a. \tag{30}$$

Since the rotation is an affine transformation, it is easy to see that for the line segment defined by (27):

$$\varphi_a(t) = \tilde{\varphi}_a(t) - \omega_a. \tag{31}$$

Clearly, $\tilde{y}_a(t) > 0$ for $0 \le t \le 1$ due to $\tilde{y}_a(0) > 0$, $\tilde{y}_a(1) > 0$ and (27). Consequently, $\varphi_a(t)$ defined as in (29) may now be expressed as:

$$\varphi_a(t) = \arccos\left(\frac{\tilde{x}(t)}{\sqrt{\tilde{x}^2(t) + \tilde{y}^2(t)}}\right) - \omega_a \tag{32}$$

Now, $\varphi_a(t)$ may easily be derived and after the simplification we get:

$$\varphi_a'(t) = \frac{\tilde{x}_a(0) \cdot \tilde{y}_a(1) - \tilde{x}_a(1) \cdot \tilde{y}_a(0)}{\tilde{x}_a^2(t) + \tilde{y}_a^2(t)} \tag{33}$$

Due to the rotational invariance of the nominator and the denominator, we can return to the original (not rotated) coordinates:

$$\varphi_a'(t) = \frac{x_a(0) \cdot y_a(1) - x_a(1) \cdot y_a(0)}{x_a^2(t) + y_a^2(t)} \tag{34}$$

The similar procedure of a rotation may be performed for the vertex b, since for b it also holds that $|\varphi_a(1) - \varphi_a(0)| < \pi$. Therefore we can write $\theta_i'(t) = 0$, when $\theta_i(t) = \varphi_b(t) - \varphi_a(t)$ as:

$$\frac{x_b(0) \cdot y_b(1) - x_b(1) \cdot y_b(0)}{x_b^2(t) + y_b^2(t)} - \frac{x_a(0) \cdot y_a(1) - x_a(1) \cdot y_a(0)}{x_a^2(t) + y_a^2(t)} = 0 \tag{35}$$

The expression $\tilde{x}^2(t) + \tilde{y}^2(t)$, being ρ-components of the polar coordinates, is strictly positive. Hence, 35 is equivalent to:

$$\begin{aligned} &\big[x_b(0) \cdot y_b(1) - x_b(1) \cdot y_b(0)\big] \cdot \big[x_a^2(t) + y_a^2(t)\big] - \\ &\big[x_a(0) \cdot y_a(1) - x_a(1) \cdot y_a(0)\big] \cdot \big[x_b^2(t) + y_b^2(t)\big] = 0 \end{aligned} \tag{36}$$

Since $x(t)$ and $y(t)$ are linear in t, (36) is a quadratic equation in t and may be solved analytically.

REFERENCES

Alexa, M., Cohen-Or, D., and Levin, D. 2000. As-rigid-as-possible polygon morphing. *Proceedings of SIGGRAPH '2000*, 157–164.

Aronov, B., Seidel, R., and Souvaine, D. L. 1993. On compatible triangulations of simple polygons. *Computational Geometry: Theory and Applications 3*, 27–35.

Babikov, M., Souvaine, D. L., and Wenger, R. 1997. Constructing piecewise linear homeomorphisms of polygons with holes. *Proceedings of 9th Canadian Conference on Computational Geometry*.

Beier, T. and Neely, S. 1992. Feature-based image metamorphosis. *Computer Graphics (SIGGRAPH '92) 26,* 2, 35–42.

Carmel, E. and Cohen-Or, D. 1997. Warp-guided object-space morphing. *The Visual Computer 13*, 465–478.

Cohen-Or, D., Levin, D., and Solomovici, A. 1998. Three-dimensional distance field metamorphosis. *ACM Transactions on Graphics 17,* 2 (Apr.), 116–141.

Eckstein, I., Surazhsky, V., and Gotsman, C. 2001. Texture mapping with hard constraints. *Proceedings of Eurographics, Manchester*. To appear.

Etzion, M. and Rappoport, A. 1997. On compatible star decompositions of simple polygons. *IEEE Trans. on Visualization and Computer Graphics 3,* 1, 87–95.

Fáry, I. 1948. On straight line representation of planar graphs. *Acta Univ. Szeged Sect. Sci. Math. 11*, 229–233.

Floater, M. S. 1997. Parameterization and smooth approximation of surface triangulation. *Computer Aided Geometric Design 14*, 231–250.

Floater, M. S. and Gotsman, C. 1999. How to morph tilings injectively. *Computational and Applied Mathematics 101*, 117–129.

Fujimura, K. and Makarov, M. 1998. Foldover-free image warping. *Graphical Models and Image Processing 60,* 2 (Mar.), 100–111.

Goldstein, E. and Gotsman, C. 1995. Polygon morphing using a multiresolution representation. *Proceeding of Graphics Interface*.

Gotsman, C. and Surazhsky, V. 2001. Guaranteed intersection-free polygon morphing. *Computers and Graphics 25,* 1, 67–75.

Luenberger, D. G. 1973. *Introduction to linear and nonlinear programming.* Addison-Wesley.

Saalfeld, A. 1987. Joint triangulations and triangulation maps. *Proceedings of 3rd Annual ACM Symposium on Computational Geometry*, 195–204.

Samoilov, T. and Elber, G. 1998. Self-intersection elimination in metamorphosis of two-dimensional curves. *The Visual Computer 14*, 415–428.

Sederberg, T. W., Gao, P., Wang, G., and Mu, H. 1993. 2D shape blending: an intrinsic solution to the vertex path problem. *Computer Graphics (SIGGRAPH '93) 27*, 15–18.

Sederberg, T. W. and Greenwood, E. 1992. A physically based approach to 2D shape blending. *Computer Graphics (SIGGRAPH '92) 26*, 25–34.

Shapira, M. and Rappoport, A. 1995. Shape blending using the star-skeleton representation. *IEEE Trans. on Computer Graphics and Application 15,* 2, 44–51.

Souvaine, D. L. and Wenger, R. 1994. Constructing piecewise linear homeomorphisms. Tech. Rep. 94–52, DIMACS. Dec.

Sugihara, K. 1999. Surface interpolation based on new local coordinates. *Computer-Aided Design 31*, 51–58.

Surazhsky, V. 1999. Morphing planar triangulations. M.S. thesis, Technion—Israel Institute of Technology.

Surazhsky, V. and Gotsman, C. 2001. Morphing stick figures using optimized compatible triangulations. *Proceedings of Pacific Graphics, Tokyo*. To appear.

Tal, A. and Elber, G. 1999. Image morphing with feature preserving texture. *Computer Graphics Forum (Eurographics '99 Proceedings) 18,* 3, 339–348.

Tutte, W. T. 1963. How to draw a graph. *Proc. London Math. Soc. 13*, 743–768.

Received April 1999; accepted May 2001; revised June 2001.