

Planar and Toroidal Morphs Made Easier

Jeff Erickson

Patrick Lin

University of Illinois Urbana-Champaign

Abstract

We present simpler algorithms for two closely related morphing problems, both based on the barycentric interpolation paradigm introduced by Floater and Gotsman, which is in turn based on Floater’s asymmetric extension of Tutte’s classical spring-embedding theorem.

First, we give a very simple algorithm to construct piecewise-linear morphs between planar straight-line graphs. Specifically, given isomorphic straight-line drawings Γ_0 and Γ_1 of the same 3-connected planar graph G , with the same convex outer face, we construct a morph from Γ_0 to Γ_1 that consists of $O(n)$ unidirectional morphing steps, in $O(n^{1+\omega/2})$ time. Our algorithm entirely avoids the classical edge-collapsing strategy dating back to Cairns; instead, in each morphing step, we interpolate the pair of weights associated with a single edge.

Second, we describe a natural extension of barycentric interpolation to geodesic graphs on the flat torus. Barycentric interpolation cannot be applied directly in this setting, because the linear systems defining intermediate vertex positions are not necessarily solvable. We describe a simple scaling strategy that circumvents this issue. Computing the appropriate scaling requires $O(n^{\omega/2})$ time, after which we can compute the drawing at any point in the morph in $O(n^{\omega/2})$ time. Our algorithm is considerably simpler than the recent algorithm of Chambers *et al.* and produces more natural morphs. Our techniques also yield a simple proof of a conjecture of Connelly *et al.* for geodesic torus triangulations.

1 Introduction

Computing morphs between geometric objects is a fundamental problem that has been well studied, with many applications in graphics, animation, modeling, and more. A particularly well-studied setting is that of morphing between planar straight-line graphs. Formally, a morph between two isomorphic planar straight-line graphs Γ_0 and Γ_1 consists of a continuous family of planar straight-line graphs Γ_t starting at Γ_0 and ending at Γ_1 .

We describe an extremely simple morphing algorithm for planar graphs, which simultaneously obtains properties of two earlier approaches: Floater and Gotsman’s barycentric interpolation method [28, 30, 50–52] results in morphs that are natural and visually appealing but are represented implicitly; variations on Cairns’ edge-collapse method [1, 9, 10, 35, 53] result in efficient explicit representations of morphs that are not useful for visualization. Our new algorithm efficiently computes an explicit piecewise-linear representation of a morph between drawings of the same 3-connected planar graph, that are potentially more useful for visualization than morphs based on Cairns’ method.

We also extend Floater and Gotsman’s planar morphing algorithm to geodesic graphs on the flat torus. Recent results of Luo, Wu, and Zhu [41] imply that Floater and Gotsman’s method directly generalizes to morphs between geodesic triangulations on surfaces of negative curvature, but a direct generalization to the torus generically fails [48]. Our extension is based on

simple scaling strategy, and it yields more natural morphs than previous algorithms based on edge collapses [12]. Finally, our arguments yield a straightforward proof of a conjecture of Connelly, Henderson, Ho, and Starbird [18] on the structure of the deformation space of geodesic triangulations of the torus.

1.1 Related Work

1.1.1 Planar Morphs

The history of morphing arguably begins with Steinitz [49, p. 347], who proved that any 3-dimensional convex polyhedron can be continuously formed into any other convex polyhedron with the same 1-skeleton.

Cairns [9, 10] was the first to prove the existence of morphs between arbitrary isomorphic planar straight-line triangulations, using an inductive argument based on the idea of collapsing an edge from a low-degree vertex to one of its neighbors. Thomassen [53] extended Cairns' proof to arbitrary planar straight-line graphs. Cairns and Thomassen's proofs are constructive, but yield morphs consisting of an exponential number of steps.

Floater and Gotsman [28] proposed a more direct method to construct morphs between planar graphs, based on an extension by Floater [26] of Tutte's classical spring embedding theorem [56]. Let Γ be a straight-line drawing of a planar graph G , such that the boundary of every face of Γ is a strictly convex polygon. Then every interior vertex in Γ is a strict convex combination of its neighbors; that is, we can associate a positive weight $\lambda_{u \rightarrow v}$ with each half-edge or *dart* $u \rightarrow v$ in G , such that the vertex positions p_v in Γ satisfy the linear system

$$\sum_{u \rightarrow v} \lambda_{u \rightarrow v} (p_v - p_u) = (0, 0) \quad \text{for every interior vertex } u \quad (1)$$

Floater [26] proved that given arbitrary¹ positive weights $\lambda_{u \rightarrow v}$ and an arbitrary convex outer face, solving linear system (1) yields a straight-line drawing of G with convex faces. Tutte's original spring-embedding theorem [56] is the special case of this result where every dart has weight 1, but his proof extends verbatim to arbitrary symmetric weights, where $\lambda_{u \rightarrow v} = \lambda_{v \rightarrow u}$ for every edge uv [33, 46, 54].

Floater and Gotsman [28] construct a morph between two convex drawings of the same planar graph G , with the same outer face, by linearly interpolating between weights $\lambda_{u \rightarrow v}$ consistent with the initial and final drawings. Appropriate initial and final weights can be computed in $O(n)$ time using, for example, Floater's mean-value coordinates [27, 34]. The resulting morphs are natural and visually appealing. However, the motions of the vertices are only computed implicitly; vertex positions at any time can be computed in $O(n^{\omega/2})$ time by solving a linear system via nested dissection [4, 38], where $\omega < 2.37286$ is the matrix multiplication exponent [3, 37].² Gotsman and Surazhsky generalized Floater and Gotsman's technique to arbitrary planar straight-line graphs [30, 50–52].

A long series of later works, culminating in a paper by Alamdari, Angelini, Barrera-Cruz, Chan, Da Lozzo, Di Battista, Frati, Haxell, Lubiw, Patrignani, Roselli, Singla, and Wilkinson [1], describe an efficient algorithm to construct planar morphs with explicit piecewise-linear vertex

¹Floater's presentation assumes that $\sum_{u \rightarrow v} \lambda_{u \rightarrow v} = 1$ for every interior vertex v , but this assumption is clearly unnecessary.

²By solving the underlying linear system symbolically, it is possible to express the motion of each vertex as a rational function of degree $n - 1$, but computing vertex coordinates at any particular time from this representation requires $O(n^2)$ time.

trajectories, all ultimately based on Cairns’ inductive edge-collapsing strategy. Given any two isomorphic straight-line drawings (with the same rotation system and nesting structure) of the same n -vertex planar graph, the resulting algorithm constructs a morph consisting of $O(n)$ *unidirectional* morphing steps, in which all vertices move along parallel lines at fixed speeds. Thus, each vertex moves along a piecewise-linear path of complexity $O(n)$, and the entire morph has complexity $O(n^2)$. These results require several delicate arguments; in particular, to perturb the *pseudomorphs* defined by edge collapses and their reversals into true morphs. Recent results of Klemz [36] imply that this algorithm can be implemented to run in $O(n^2 \log n)$ time on an appropriate real RAM model of computation. The resulting morph contracts all vertices into an exponentially small neighborhood and then expand them again, so it is not useful for visualization.

Angelini, Da Lozzo, Frati, Lubiw, Patrignani, and Roselli Angelini [5] consider the setting of *convexity-preserving* morphs between convex drawings; Kleist, Klemz, Lubiw, Schlipf, Staals, and Strash [35] consider morphing to convexify any 3-connected planar drawing. Both describe algorithms that produce piecewise-linear morphs consisting of $O(n)$ steps, and that can be implemented to run in time $O(n^{1+\omega/2})$. (Klemz [36] conjectures that both running times can be improved to $O(n^2 \log n)$.) Combining these algorithms results in an alternative piecewise-linear morph between 3-connected planar drawings.

1.1.2 Toroidal Morphs

Until recently, very little was known about morphing graphs on the torus or other more complex surfaces.

Tutte’s spring-embedding theorem was generalized to simple triangulations of surfaces with non-positive curvature by Colin de Verdière [17] and independently by Hass and Scott [31]. Delgado-Friedrichs [22], Lovász [39], and Gortler *et al.* [29] also independently proved an extension of Tutte’s theorem to graphs on the flat torus whose universal covers are simple and 3-connected. For any toroidal graph and any assignment of positive *symmetric* weights to the darts, solving a linear system similar to (1) yields vertex positions of a geodesic drawing with strictly convex faces [25, 29]; see Section 2 for details. Thus, if two isotopic geodesic torus graphs Γ_0 and Γ_1 can both be described by symmetric dart weights, linearly interpolating those weights yields a morph from Γ_0 to Γ_1 [16]; in light of the authors’ toroidal Maxwell–Cremona correspondence [25], this can be seen as a natural toroidal analogue of Steinitz’s theorem on morphing convex polyhedra [49].

The restriction to symmetric weights is both nontrivial and significant. In a torus graph with convex faces, every vertex can be described as a convex combination of its neighbors, but not necessarily with symmetric weights. Moreover, the linear system expressing vertex positions as convex combinations of its neighbors is rank-deficient, and therefore is not solvable in general; see Appendix A for an example. Thus, Floater’s asymmetric extension of Tutte’s theorem does *not* directly generalize to the flat torus.

For similar reasons, Floater and Gotsman’s planar morphing algorithm also does not generalize. Suppose we are given two isotopic geodesic torus graphs Γ_0 and Γ_1 , each with dart weights that express their vertices as convex combinations of their neighbors. Unfortunately, in general, interpolating those weights yields linear systems that have no solution; we give a simple example in Appendix A.

Steiner and Fischer [48] modify the system by fixing a single vertex, restoring full rank. However, solving the modified system does not necessarily yield a crossing-free drawing, be-

cause the fixed vertex may not lie in the convex hull of its neighbors.³ Moreover, even though the initial and final weights are consistent with crossing-free drawings, averages of those weights may not be. We give an example of this bad behavior in Appendix A.

Chambers, Erickson, Lin, and Parsa [12] described the first algorithm to morph between arbitrary essentially 3-connected geodesic torus graphs. Their algorithm uses a combination of Cairns’ edge-collapsing strategy and spring embeddings to construct a morph consisting of $O(n)$ unidirectional morphing steps, in $O(n^{1+\omega/2})$ time. Like planar morphs built from edge collapses, these toroidal morphs contract vertices into small neighborhoods and thus are not suitable for visualization.

Recently, Luo *et al.* [41] generalized Floater’s theorem to geodesic triangulations of arbitrary closed Riemannian 2-manifolds with strictly negative curvature, extending the spring-embedding theorems of Colin de Verdière [17] and Hass and Scott [31] to asymmetric weights. Their result immediately implies that if two geodesic triangulations of such a surface are homotopic, then linearly interpolating the dart weights yields a continuous family of crossing-free geodesic drawings, or in other words, a morph. Their result applies only to surfaces with negative Euler characteristic; alas, the torus has Euler characteristic 0.

1.2 New Results

We describe two applications of Floater and Gotsman’s barycentric interpolation strategy, which yield simpler algorithms for morphing planar and toroidal graphs.

First we describe a very simple algorithm to construct piecewise-linear morphs between planar straight-line graphs. Given two isomorphic planar straight-line graphs Γ_0 and Γ_1 with strictly convex faces and the same outer face, we construct a morph from Γ_0 to Γ_1 that consists of $O(n)$ unidirectional morphing steps, in $O(n^{1+\omega/2})$ time. Our morphing algorithm computes barycentric weights for the darts in Γ_0 and Γ_1 in a preprocessing phase, and then for each morphing step, interpolates only the pair of weights associated with a single edge. Our key observation is that changing the weights for a single edge e moves all vertices in the Floater drawing along lines parallel to e . (The same observation was made for *symmetric* edge weights by Chambers *et al.* [12].) Our algorithm is significantly simpler than that of Angelini *et al.* [5] for computing convexity-preserving morphs. We then extend our algorithm to drawings with non-convex faces, using a simpler approach than Kleist *et al.* [35]. Fig. 1 shows a morph computed by our algorithm; in each frame, the weights of the red edge are about to change.

Next, we describe a natural extension of Floater and Gotsman’s method to geodesic graphs on the flat torus. Our key observation is that barycentric dart weights can be *scaled* so that barycentric interpolation works. Specifically, we call a weight assignment *morphable* if every *column* of the resulting Laplacian linear system sums to zero; averages of morphable weights are morphable. Given any weight assignment consistent with any convex drawing, we can guarantee morphability by scaling the weights of all darts leaving each vertex v —or equivalently, scaling each *row* of the linear system—by a common positive scalar α_v . This scaling obviously has no effect on the solution space of the system. Positivity of the scaling vector α follows from a weighted directed version of the matrix-tree theorem [8, 20, 55]. We can compute the appropriate scaling in $O(n^{\omega/2})$ time, after which we can compute any intermediate drawing in $O(n^{\omega/2})$ time, matching the performance of Floater and Gotsman exactly. The resulting morphs are natural and visually appealing, and our proofs of correctness are considerably simpler than those of Chambers *et al.* [12]. However, unlike Chambers *et al.*, our new morphing algorithm

³Steiner and Fischer incorrectly claim [48, Section 2.2.1] that the resulting drawing has no “foldovers” except at the fixed vertex and its neighbors; see Appendix A.

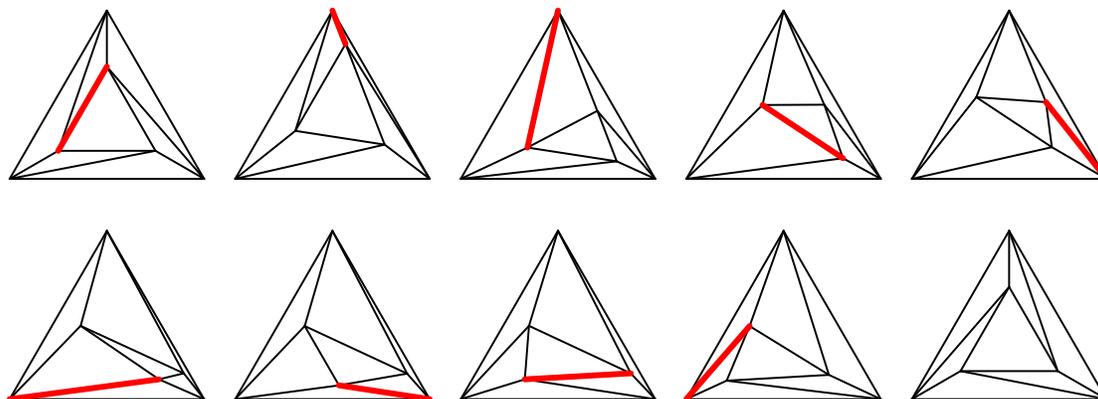


Figure 1. Incrementally morphing between planar graphs.

does not compute explicit vertex trajectories. Fig. 2 shows a morph computed by our algorithm between two randomly shifted 6×6 toroidal grids. (The authors’ Python implementation is available on request.)

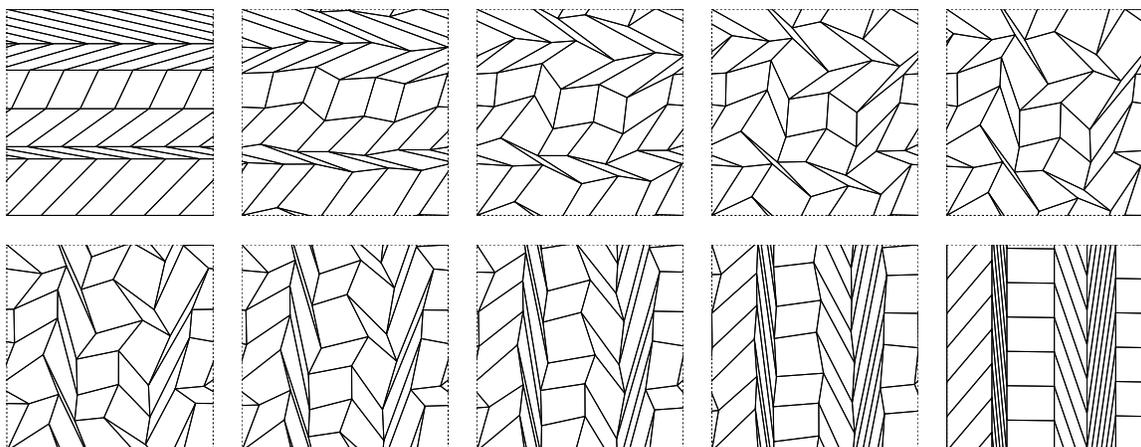


Figure 2. Morphing between randomly shifted toroidal grids.

It remains an open question whether our results can be combined to compute explicit low-complexity piecewise-linear toroidal morphs without edge collapses. We offer some preliminary observations in Appendix B.

2 Definitions and Notation

2.1 Planar Graphs

Any planar straight-line drawing Γ can be represented by a *position* matrix $P \in \mathbb{R}^{n \times 2}$, each row p_v of which gives the location of some vertex v . Thus, each edge uv is drawn as the straight-line segment $p_u p_v$. We call a planar drawing *convex* if it is crossing-free, every bounded face is a convex polygon, and the outer face is the complement of a convex polygon.

Formally, we regard each edge of any graph as a pair of opposing half-edges or *darts*, each directed from its *tail* to its *head*. We write $rev(d)$ to denote the reversal of any dart d . For simple graphs, we write $u \rightarrow v$ to denote the dart with tail u and head v . A *barycentric weight*

vector for Γ assigns a positive real number $\lambda_{u \rightarrow v}$ to every dart $u \rightarrow v$ of a graph, so that the vertex positions p_v satisfy Floater's linear system (1). Conversely, for a fixed graph G with a fixed convex outer face, the *Floater drawing* Γ^λ of G with respect to a positive weight vector λ is the unique drawing whose vertex positions p_v satisfy system (1).

A *morph* between two planar drawings Γ_0 and Γ_1 is a continuous family of crossing-free drawings Γ_t parametrized by time, starting at Γ_0 and ending at Γ_1 . A morph is *linear* if each vertex moves along a straight line at uniform speed, and *piecewise-linear* if it is the concatenation of linear morphs. Any piecewise-linear morph can be described by a finite sequence of straight-line drawings, or their position matrices. A linear morph is *unidirectional* if vertices move along parallel lines.

2.2 Torus Graphs

The *flat torus* is the quotient space $\mathbb{T} = \mathbb{R}^2 / \mathbb{Z}^2$, also obtained by identifying opposite sides of the unit square $[0, 1]^2$. A *geodesic* on the flat torus is the image of a line segment in \mathbb{R}^2 under the projection map $\pi: \mathbb{R}^2 \rightarrow \mathbb{T}$ where $\pi(x, y) = (x \bmod 1, y \bmod 1)$.

A (crossing-free) *geodesic torus drawing* Γ of a graph G maps its vertices to distinct points in \mathbb{T} and its edges to simple, interior-disjoint geodesics. We explicitly consider geodesic drawings of graphs with loops and parallel edges. We write $d: u \rightarrow v$ to declare that d is a dart with tail u and head v ; we emphasize that (unlike in planar setting) there may be more than one such dart.

Every geodesic torus drawing Γ of a graph G is the projection of an infinite, doubly-periodic planar straight-line graph $\tilde{\Gamma}$, called the *universal cover* of Γ [12]. We call Γ *essentially simple* if its universal cover $\tilde{\Gamma}$ is simple, and *essentially 3-connected* if $\tilde{\Gamma}$ is 3-connected [43, 44]. Finally, we call Γ a *convex* drawing if every face of $\tilde{\Gamma}$ is strictly convex. Every convex torus drawing is both essentially simple and essentially 3-connected, since every infinite planar graph with strictly convex faces is 3-connected [21].

Coordinate representations. Following Chambers *et al.* [12], we use a *coordinate representation* (P, τ) for geodesic torus drawings that records

- a *position* vector $p_v \in \mathbb{R}^2$ for each vertex v , and
- a *translation* vector $\tau_d \in \mathbb{Z}^2$ for each dart d , such that $\tau_{\text{rev}(d)} = -\tau_d$.

These vectors indicate that each dart $d: u \rightarrow v$ is drawn as the projection of a line segment from p_u to $p_v + \tau_d$ in the universal cover $\tilde{\Gamma}$. In particular, if we normalize all vertex positions to the half-open unit square $[0, 1)^2$, then each translation vector τ_d indicates the number of times d crosses the vertical boundary of the unit square to the right, and the number of times d crosses the horizontal boundary of the unit square upward.

Two crossing-free drawings of the same graph on the torus are *isotopic* if one can be deformed into the other through a continuous family of (not necessarily geodesic) crossing-free drawings; such a deformation is called an *isotopy*. Two crossing-free drawings are isotopic if and only if their coordinate representations can be *normalized* so that their translation vectors agree; this condition can be tested in $O(n)$ time [12, Theorem A.1], [15]. A *geodesic isotopy* or *morph* is an isotopy in which all intermediate drawings are geodesic.

Barycentric weights. In any convex torus drawing Γ , the position p_v of each vertex v can be expressed as a convex combination of its neighbors, as follows. We can assign a weight $\lambda_d > 0$

to each dart d such that any coordinate representation (P, τ) of Γ satisfies the linear system

$$\sum_v \sum_{d:u \rightarrow v} \lambda_d (p_v - p_u + \tau_d) = (0, 0) \quad \text{for every vertex } u. \quad (2)$$

We can express this linear system in matrix notation as $L^\lambda P = H^\lambda$, where

$$L_{ij}^\lambda = \begin{cases} \sum_k \sum_{d:i \rightarrow k} \lambda_d & \text{if } i = j \\ \sum_{d:i \rightarrow j} -\lambda_d & \text{otherwise} \end{cases} \quad \text{and} \quad H_i^\lambda = \sum_j \sum_{d:i \rightarrow j} \lambda_d x_d \quad (2')$$

The (unnormalized, asymmetric) Laplacian matrix L^λ has rank $n - 1$ [48]. We call any positive weight vector λ satisfying system (2) *barycentric* for Γ . Barycentric weights for any convex torus drawing can be computed in $O(n)$ time using, for example, Floater's mean-value coordinates [27, 34].

On the other hand, suppose we fix the graph G and translation vectors τ_d consistent with an essentially 3-connected (but not necessarily geodesic) drawing of G . Then for any positive weight vector λ , any solution to linear system (2) gives the vertex positions p_v of a convex drawing Γ^λ of G [29]. In this case, we say that the *Floater drawing* Γ^λ *realizes* the weight vector λ , and we call the weight vector λ *realizable* for the graph G . Every realizable weight vector is realized by a two-dimensional family of drawings that differ by translation.

Every *symmetric* positive weight vector (where $\lambda_d = \lambda_{\text{rev}(d)}$) is realizable: for any assignment of positive weights to the *edges* of G , there is a corresponding convex torus drawing [17, 22, 29, 31, 39]. Realizable weights are not necessarily symmetric: there are convex torus drawings with only asymmetric barycentric weights. Conversely, positive asymmetric weights are not always realizable.

3 Morphing Planar Graphs Edge by Edge

We describe a very simple algorithm to morph planar straight-line graphs that combines the benefits of both the Floater and Gotsman approach [28, 30, 50–52] and the Cairns approach [1, 9, 10, 35, 53]. Our algorithm constructs a morph consisting of $O(n)$ unidirectional morphing steps, in $O(n^{1+\omega/2})$ time. Because our morphs do not use edge collapses, they are also potentially good for visualization.

Fix a planar graph G and a convex outer face. Let p_v^λ denote the position of vertex v in the Floater drawing Γ^λ with respect to weight vector λ . The following lemma is a planar asymmetric version of Lemma 5.1 of Chambers *et al.* [12]. Intuitively, it states that changing the weights of the darts of a single edge e moves each vertex in the Floater drawing along lines parallel to e .

Lemma 3.1. *Let λ and μ be arbitrary positive weight vectors such that $\lambda_d \neq \mu_d$ or $\lambda_{\text{rev}(d)} \neq \mu_{\text{rev}(d)}$ for some dart d , but $\lambda_{d'} = \mu_{d'}$ for all darts $d' \notin \{d, \text{rev}(d)\}$. For each vertex w , the vector $p_w^\mu - p_w^\lambda$ is parallel to the drawing of d in Γ^λ .*

Proof: Suppose d has tail u and head v , and (by rotating the drawing if necessary) that d is drawn parallel to the x -axis. For each vertex i , let y_i^λ and y_i^μ be the y -coordinates of points p_i^λ and p_i^μ , respectively, so that $y_u^\lambda = y_v^\lambda$. We need to prove that $y_w^\lambda = y_w^\mu$ for every vertex w .

Projecting linear system (1) for λ onto the y -axis gives us

$$\sum_{i \rightarrow j} \lambda_{i \rightarrow j} (y_j^\lambda - y_i^\lambda) = 0 \quad \text{for each vertex } i. \quad (3)$$

Swapping entries of λ with corresponding entries of μ in the system (3) changes at most two constraints, corresponding to the two endpoints u and v of d . Moreover, in each changed constraint, the single changed coefficient is multiplied by $y_u^\lambda - y_v^\lambda = y_v^\lambda - y_u^\lambda = 0$, so the y_i^λ 's also solve the corresponding system for μ . Since the system (3) and its counterpart for μ each have a unique solution, we conclude that $y_w^\lambda = y_w^\mu$ for every vertex w . \square

Under the assumptions of Lemma 3.1, linearly interpolating vertex positions from Γ^λ to Γ^μ yields a unidirectional linear morph [1, Corollary 7.2], [12, Lemma 5.2]. It follows that we can morph between isomorphic convex drawings through a sequence of at most $3n-9$ unidirectional linear morphing steps, one for each internal edge, following the algorithm in Fig. 3. Initial and final barycentric weight vectors can be found in $O(n)$ time using, for example, Floater's mean-value method [27, 34]. Each intermediate drawing can be computed in $O(n^{\omega/2})$ time using nested dissection [4, 38], for a total running time of $O(n^{1+\omega/2})$.

```

MORPHCONVEX( $\Gamma_{\text{start}}, \Gamma_{\text{end}}$ ):
   $\lambda \leftarrow$  barycentric weights for  $\Gamma_{\text{start}}$ 
   $\mu \leftarrow$  barycentric weights for  $\Gamma_{\text{end}}$ 
   $k \leftarrow 0$ 
  for each internal edge  $e$ 
     $k \leftarrow k + 1$ 
     $d \leftarrow$  a dart of  $e$ 
     $\lambda_d \leftarrow \mu_d$ 
     $\lambda_{\text{rev}(d)} \leftarrow \mu_{\text{rev}(d)}$ 
     $\Gamma_k \leftarrow \Gamma^\lambda$ 
  return  $\Gamma_{\text{start}}, \Gamma_1, \Gamma_2, \dots, \Gamma_k \llbracket = \Gamma_{\text{end}} \rrbracket$ 

```

Figure 3. Algorithm for morphing between convex planar drawings.

Because all Floater drawings are convex, Lemma 5.2 of Chambers *et al.* [12] implies that MORPHCONVEX actually produces a *convexity-preserving* piecewise-linear morph; all faces remain convex throughout the morph. The existence of convexity-preserving morphs was first proved by Thomassen [53]; Angelini, Da Lozzo, Frati, Lubiw, Patrignani, and Roselli [5] described a piecewise-linear convexity-preserving morph consisting of $O(n)$ steps. Our algorithm is significantly simpler than that of Angelini *et al.* [5].

Theorem 3.2. *Given any two isomorphic convex planar drawings with n vertices and the same convex outer face, we can compute a morph between them consisting of at most $3n-9$ unidirectional linear morphing steps, in $O(n^{1+\omega/2})$ time.*

The previous algorithm can be extended to morph between non-convex drawings by introducing intermediate convex drawings, as follows: Add edges to the initial and final drawings to decompose every face into convex polygons, compute barycentric weights for the resulting convex drawing, and then reduce the weights of each added edge (one-by-one) to zero, effectively deleting that edge. Dropping the added edges yields a piecewise-linear morph from each input

drawing to a convex drawing. Again, each intermediate drawing can be computed in $O(n^{\omega/2})$ time. Our complete morphing algorithm is shown in Fig. 4. Our algorithm CONVEXIFY is considerably simpler than that of Kleist *et al.* [35]; however, unlike Kleist *et al.*, our algorithm is *not* necessarily convexity-increasing.

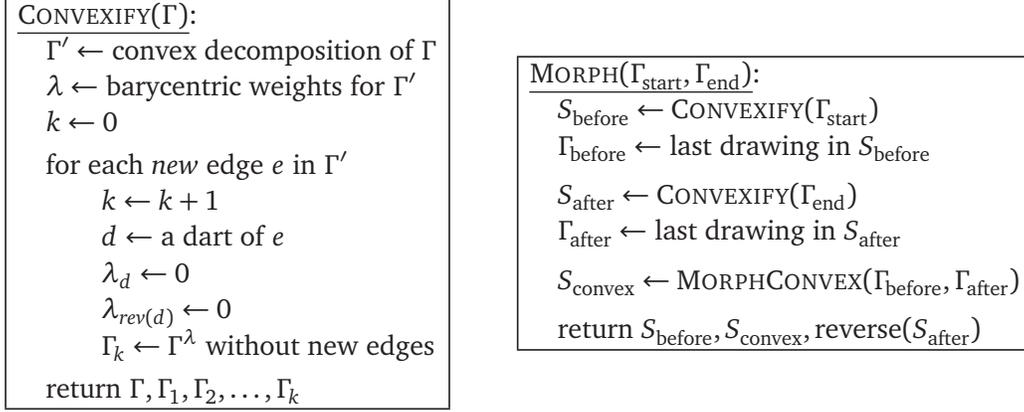


Figure 4. Algorithm for morphing between general planar straight-line drawings.

In total, we perform one morphing step for each internal edge of G , plus at most $2(k-3)$ morphing steps for each bounded face with degree k . Euler’s formula implies that a 3-connected planar graph has between $1.5n$ and $3n-6$ edges, and thus at most $3n-9$ internal edges. Thus, we need to add at most $1.5n-6$ edges to convexify the initial and final faces, so our morph consists of at most $4.5n-15$ linear morphing steps. In summary:

Theorem 3.3. *Given any two isomorphic 3-connected planar straight-line drawings with n vertices and the same convex outer face, we can compute a morph between them consisting of at most $4.5n-15$ unidirectional linear morphing steps, in $O(n^{1+\omega/2})$ time.*

Models of computation. The algorithm of Alamdari *et al.* [1] require a slightly nonstandard real RAM model of computation that supports exact square and exact cube roots. In contrast, Floater’s mean-value weights can be expressed in terms of areas and Euclidean lengths [34], which require only square roots to evaluate exactly. If initial and final barycentric weights are given, both Floater and Gotsman’s morphing algorithm [28] and our incremental algorithm use only basic arithmetic operations: addition, subtraction, multiplication, and division.

Even without exact roots, any integer-RAM or floating-point implementation of our morphing algorithm must contend with precision issues. A careful implementation of Alon and Yuster’s nested dissection algorithm [4] solves Floater’s linear system (1) exactly in $O(n^{1+\omega/2} \text{polylog } n)$ bit operations, assuming all dart weights $\lambda_{u \rightarrow v}$ are $O(\log n)$ -bit integers [4,6]. Thus, at least then all weights are given as part of the input, an exact implementation of our morphing algorithm runs in $O(n^{2+\omega/2} \text{polylog } n)$ on a standard integer RAM. Coordinates of Tutte/Floater drawings can require $\Omega(n)$ bits of precision to avoid collapsing or crossing edges [23,24]; a canonical bad example is shown in Figure 5. Thus, the near-linear cost of exact arithmetic is unavoidable in the worst case.

Shen, Jiang, Zorin, and Panozzo [47] observe that floating-point implementations of Tutte’s algorithm suffer from robustness issues in practice. Shen *et al.* describe an iterative procedure to repair floating-point-Tutte drawings; however, it is unclear whether a similar procedure can be used to avoid precision issues in our algorithm while maintaining continuity of the resulting morph. It is also unclear whether precision issues in our algorithm can be avoided, or at

least minimized, by carefully choosing the order in which edge weights are changed and/or by morphing through a carefully chosen intermediate drawing.

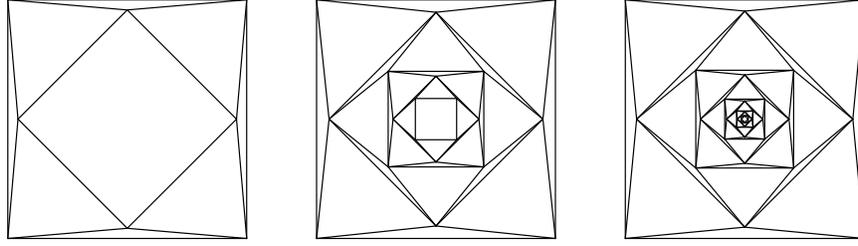


Figure 5. A family of weighted spring embeddings consisting of nested squares, with 2, 5, and 10 layers, respectively.

4 Morphable Weight Vectors on the Flat Torus

As observed by Steiner and Fischer [48], Floater and Gotsman’s morphing algorithm does not directly generalize to the toroidal setting, since not all positive weight vectors λ are realizable. In particular, given arbitrary barycentric weights $\lambda(0)$ and $\lambda(1)$ of two isotopic convex torus drawings, intermediate weights $\lambda(t) := (1 - t)\lambda(0) + t\lambda(1)$ are not necessarily realizable; see Appendix A for an example. Thus, interpolating barycentric weights does not necessarily give us a morph.

To bypass this issue, we identify a subspace of *morphable* weight vectors, such that every convex torus drawing has a morphable barycentric weight vector, every morphable weight vector is realizable, and convex combinations of morphable weights are morphable. Specifically, a positive weight vector λ is *morphable* if each *column* of the matrices L^λ and H^λ sums to 0.⁴ The following lemma is immediate:

Lemma 4.1. *Convex combinations of morphable weight vectors are morphable.*

Lemma 4.2. *Every morphable weight vector is realizable.*

Proof: If λ is a morphable weight vector, then the n th row of the linear system $L^\lambda P = H^\lambda$ is implied by the other $n - 1$ rows, so we can remove it. The resulting abbreviated linear system still has rank $n - 1$, so it has a (unique) solution. \square

Lemma 4.3. *Given a barycentric weight vector λ for a convex torus drawing Γ , a morphable barycentric weight vector for Γ can be computed in $O(n^{\omega/2})$ time.*

Proof: The matrix L^λ has rank $n - 1$, so there is a one-dimensional space of (row) vectors $\alpha = (\alpha_1, \dots, \alpha_n)$ such that $\alpha L^\lambda = (0, \dots, 0)$. We can compute a non-zero vector α in $O(n^{\omega/2})$ time using nested dissection [2, 4, 38].

A directed version of the matrix tree theorem [8, 20, 55] implies that we can choose all α_i to be positive. Specifically, let G^\pm be the weighted directed graph whose weighted arcs correspond to the weighted darts of G . An *inward directed spanning tree* is an acyclic spanning subgraph of G^\pm where every vertex except one (called the *root*) has out-degree 1. The weight of an inward

⁴Directed graph Laplacians whose columns sum to zero are also called *Eulerian*; Cohen et al. [14] refer to the scaling process described by Lemma 4.3 as *Eulerian scaling*.

directed spanning tree is the product of the weights of its arcs. For each i , let α_i be the sum of the weights of all inward directed spanning trees rooted at vertex i ; we have $\alpha_i > 0$ because all dart weights are positive. The directed matrix tree theorem implies that $\alpha L = 0$, as required; for an elementary proof, see De Leenheer [20, Theorem 3]. (See also Cohen *et al.* [14, Lemma 1] for an alternate proof using the Perron-Frobenius theorem.)

Define a new weight vector μ by setting $\mu_d := \alpha_{\text{tail}(d)} \lambda_d$ for each dart d . For each index i , we immediately have $L_i^\mu P = \alpha_i L_i^\lambda P = \alpha_i H_i^\lambda = H_i^\mu$, where P is the position matrix for Γ , so μ is in fact a barycentric weight vector for Γ . Finally, we observe that $(1, \dots, 1)L^\mu = \alpha L^\lambda = (0, \dots, 0)$ and $(1, \dots, 1)H^\mu = \alpha H^\lambda = \alpha L^\lambda P = (0, \dots, 0)P = (0, 0)$, which imply that μ is morphable. \square

Theorem 4.4. *Given coordinate representations of two isotopic essentially 3-connected geodesic torus drawings Γ_0 and Γ_1 , we can efficiently compute a morph from Γ_0 to Γ_1 . Specifically, after $O(n^{\omega/2})$ preprocessing time, we can compute any intermediate drawing during the morph in $O(n^{\omega/2})$ time.*

Proof: Suppose Γ_0 and Γ_1 are convex drawings. First, if necessary, we normalize the given coordinate representations so that their translation vectors agree, in $O(n)$ time [12, Theorem A.1]. Then we find barycentric weight vectors $\lambda(0)$ and $\lambda(1)$ for Γ_0 and Γ_1 , respectively, in $O(n)$ time, for example using Floater’s mean-value coordinates [27, 34]. Following Lemma 4.3, we derive morphable weights $\mu(0)$ and $\mu(1)$ from $\lambda(0)$ and $\lambda(1)$, respectively, in $O(n^{\omega/2})$ time. Finally, given any real number $0 < t < 1$, we set $\mu(t) := (1-t)\mu(0) + t\mu(1)$ and solve the linear system $L^{\mu(t)}P^{(t)} = H^{\mu(t)}$ for the position matrix $P^{(t)}$ of an intermediate drawing $\Gamma^{\mu(t)}$; Lemmas 4.1 and 4.2 imply that this system is solvable. The function $t \mapsto \Gamma^{\mu(t)}$ is a convexity-preserving morph between Γ_0 and Γ_1 .

If the faces of Γ_0 or Γ_1 are not convex, we morph through an intermediate convex drawing, similarly to Chambers *et al.* [12, Theorem 8.1]. Let Γ_* be the Floater drawing of G obtained by setting every dart weight to 1. Compute any triangulation T_0 of Γ_0 , and then triangulate the convex faces Γ_* using the same diagonals, to obtain a triangulation T_* isotopic to T_0 . Assign weight 0 to the darts of the diagonals in $T_* \setminus \Gamma_*$ to obtain a barycentric weight vector μ_* for T_* , which is symmetric and therefore morphable. Derive morphable weights μ_0 for T_0 using mean-value coordinates [27, 34] and Lemma 4.3. Then we can morph from T_0 to T_* by weight interpolation, using the weight vector $\mu(t) := (1-2t)\mu_0 + 2t\mu_*$ for any $0 \leq t \leq 1/2$. Ignoring the diagonal edges gives us a morph from Γ_0 to Γ_* . A symmetric procedure yields a morph from Γ_* to Γ_1 . \square

4.1 Deformation Space of Geodesic Torus Drawings

Our formulation of morphable weights provides a straightforward solution to a conjecture of Connelly *et al.* [18] about the deformation space of geodesic torus triangulations. Bloch, Connelly, and Henderson [7] proved that for any planar straight-line triangulation Γ of a convex polygon P , the space of all planar straight-line triangulations of P that are homeomorphic to Γ is contractible. (Cairns’ morphing theorem [9, 10] asserts only that this space is connected.) Simpler proofs of this theorem were recently given by Cerf [11] and Luo [40]; in particular, Luo observed that the Bloch–Connelly–Henderson theorem follows immediately from Floater’s barycentric embedding theorem.

Connelly *et al.* [18] conjectured that every isotopy class of geodesic triangulations on any surface S with constant curvature is homotopy-equivalent to the group $Isom_0(S)$ of isometries of S that are homotopic to the identity. In particular, $Isom_0(S^2)$ is the rotation group $SO(3)$ and

$Isom_0(\mathbb{T})$ is the translation group $S^1 \times S^1$; for every other orientable surface S without boundary, $Isom_0(S)$ is trivial [45]. Very recently, Luo, Wu, and Zhu proved this conjecture for all surfaces of genus at least 2 [41] and for the flat torus [42]; both proofs use nontrivial extensions of Floater’s theorem.

Here we offer a simpler proof for torus graphs; in fact, we prove a more general result about convex drawings instead of just triangulations.

Theorem 4.5. *For any convex drawing Γ on the flat torus \mathbb{T} , the space of all convex drawings isotopic to Γ is homotopy equivalent to \mathbb{T} .*

Proof: Fix a convex drawing Γ of a graph G with n vertices and m edges; without loss of generality, assume some vertex v is positioned at $(0, 0)$. Let $X = X(\Gamma)$ denote the space of all convex drawings of G isotopic to Γ , and let $X_0 = X_0(\Gamma)$ be the subspace of drawings in X where vertex v is positioned at $(0, 0)$. Every drawing in X is a translation of a unique drawing in X_0 , so $X = X_0 \times S^1 \times S^1$. Thus, to prove the theorem, it suffices to prove that X_0 is contractible.

Call a weight vector $\lambda \in \mathbb{R}_+^{2m}$ for Γ *normalized* if $\sum_d \lambda_d = 1$, where the sum is over all darts of Γ . Let $R = R(\Gamma)$ denote the set of all realizable weight vectors for T , and let $\overline{M} = \overline{M}(\Gamma)$ denote the set of all normalized morphable weight vectors for T .

Lemma 4.1 implies that \overline{M} is convex and therefore contractible. (Specifically, \overline{M} is the interior of a $(2m - n - 2)$ -dimensional convex polytope in \mathbb{R}^{2m} .)

Call two realizable weight vectors $\lambda, \lambda' \in R$ *equivalent* if there is a scaling vector $\alpha \in \mathbb{R}_+^n$ such that $\lambda'_d = \alpha_{tail(d)} \lambda_d$ for every dart d . Because the Laplacian matrix L^λ has rank $n - 1$, Lemma 4.3 implies that every realizable weight $\lambda \in R$ is equivalent to a *unique* normalized morphable weight $\mu \in \overline{M}$. It follows that R is homeomorphic to $\overline{M} \times \mathbb{R}_+^n$ and therefore contractible.

Now we follow the proof of Theorem 1.4 in Luo *et al.* [41]. Because every morphable weight is realizable, solving linear system (2) gives us a continuous map $\Phi: R \rightarrow X_0$. Floater’s mean-value weights [27, 34] give us a continuous map $\Psi: X_0 \rightarrow R$ such that $\Phi \circ \Psi$ is the identity map on X_0 . Because R is contractible, the function $\Psi \circ \Phi$ is homotopic to the identity map on R . We conclude that X_0 is homotopy equivalent to R and therefore contractible. \square

5 Open Questions

It is natural to ask whether our “best-of-both-worlds” planar morph can be extended to graphs on the flat torus. In Appendix B, we prove a toroidal analog of Lemma 3.1 for *realizable* weight vectors; unfortunately, the main roadblock is that not all weight vectors are realizable. In particular, given a realizable weight vector (morphable or not), it is not clear when changing the weights for a single edge results in another realizable weight vector.

Several previous planar morphing algorithms [1, 5, 19, 35] rely on a certain convexifying procedure [13, 32, 35, 36], and are (potentially) faster than our algorithm via the implementation recently described by Klemz [36]. It is an open question whether the procedure can be extended to geodesic torus graphs.

One can also ask if the result can be extended to surfaces of higher genus. The recent results of Luo *et al.* [41] imply that Floater and Gotsman’s planar morphing algorithm [28] extends to geodesic triangulations on higher-genus surfaces of negative curvature; however, the existence of (any reasonable analog of) piecewise-linear morphs on such surfaces remains unknown.

Acknowledgments

We thank Anna Lubiw for asking questions about Lemma 5.1 of Chambers *et al.* [12], whose answers ultimately led to the discovery of Theorem 3.3, and for other helpful feedback. We also thank Yanwen Luo for making us aware of his recent work [40–42]. Finally, we thank the anonymous reviewers for their comments and helpful suggestions for improvement.

References

- [1] Soroush Alamdari, Patrizio Angelini, Fidel Barrera-Cruz, Timothy M. Chan, Giordano Da Lozzo, Giuseppe Di Battista, Fabrizio Frati, Penny Haxell, Anna Lubiw, Maurizio Patrignani, Vincenzo Roselli, Sahil Singla, and Bryan T. Wilkinson. [How to morph planar graph drawings](#). *SIAM J. Comput.* 46(2):824–852, 2017.
- [2] Lyudmil Aleksandrov and Hristo Djidjev. [Linear algorithms for partitioning embedded graphs of bounded genus](#). *SIAM J. Discrete Math.* 9(1):129–150, 1996.
- [3] Josh Alman and Virginia Vassilevska Williams. [A refined laser method and faster matrix multiplication](#). *Proc. 32nd Ann. ACM-SIAM Symp. Discrete Algorithms*, 2021. arXiv:2010.05846.
- [4] Noga Alon and Raphael Yuster. [Matrix sparsification and nested dissection over arbitrary fields](#). *J. ACM* 60(4):25:1–25:8, 2013.
- [5] Patrizio Angelini, Giordano Da Lozzo, Fabrizio Frati, Anna Lubiw, Maurizio Patrignani, and Vincenzo Roselli. [Optimal morphs of convex drawings](#). *Proc. 31st Int. Symp. Comput. Geom.*, 126–140, 2015. Leibniz Int. Proc. Informatics 34. arXiv:1503.09021.
- [6] Erwin H. Bareiss. [Sylvester’s identity and multistep integer-preserving Gaussian elimination](#). *Math. Comput.* 22(103):565–578, 1968.
- [7] Ethan D. Bloch, Robert Connelly, and David W. Henderson. [The space of simplexwise linear homeomorphisms of a convex 2-disk](#). *Topology* 23(2):161–175, 1984.
- [8] Carl W. Borchardt. [Ueber eine der interpolation entsprechende Darstellung der Eliminations-Resultante](#). *J. Reine Angew. Math.* 57:111–121, 1860.
- [9] Stewart S. Cairns. [Deformations of plane rectilinear complexes](#). *Amer. Math. Monthly* 51(5):247–252, 1944.
- [10] Stewart S. Cairns. [Isotopic deformations of geodesic complexes on the 2-sphere and on the plane](#). *Ann. Math.* 45(2):207–217, 1944.
- [11] Jean Cerf. [About the Bloch–Connolly–Henderson theorem on the simplexwise linear homeomorphisms of a convex 2-disk](#). Preprint, October 2019. arXiv:1910.00240.
- [12] Erin Wolf Chambers, Jeff Erickson, Patrick Lin, and Salman Parsa. [How to morph graphs on the torus](#). *Proc. 32nd Ann. ACM-SIAM Symp. Discrete Algorithms*, 2759–2778, 2021. arXiv:2007.07927.
- [13] Marek Chrobak, Michael T. Goodrich, and Roberto Tamassia. [Convex drawings of graphs in two and three dimensions \(preliminary version\)](#). *Proc. 12th Ann. Symp. Comput. Geom.*, 319–328, 1996.

- [14] Michael B. Cohen, Jonathan Kelner, John Peebles, Richard Peng, Aaron Sidford, and Adrian Vladu. [Faster algorithms for computing the stationary distribution, simulating random walks, and more](#). *Proc. 57th Ann. IEEE Symp. Found. Comput. Sci.*, 583–592, 2016. arXiv:1608.03270.
- [15] Éric Colin de Verdière and Arnaud de Mesmay. [Testing graph isotopy on surfaces](#). *Discrete Comput. Geom.* 51(1):171–206, 2014.
- [16] Éric Colin de Verdière, Michel Pocchiola, and Gert Vegter. [Tutte’s barycenter method applied to isotopies](#). *Comput. Geom. Theory Appl.* 26(1):81–97, 2003.
- [17] Yves Colin de Verdière. [Comment rendre géodésique une triangulation d’une surface?](#) *L’Enseignement Mathématique* 37:201–212, 1991.
- [18] Robert Connelly, David W. Henderson, Chung Wu Ho, and Michael Starbird. On the problems related to linear homeomorphisms, embeddings, and isotopies. *Continua, Decompositions, Manifolds: Proc. Texas Topology Symposium, 1980*, 229–239, 1983. Univ. Texas Press.
- [19] Giordano Da Lozzo, Giuseppe Di Battista, Fabrizio Frati, Maurizio Patrignani, and Vincenzo Roselli. [Upward planar morphs](#). *Algorithmica* 82(10):2985–3017, 2020.
- [20] Patrick De Leenheer. [An elementary proof of a matrix tree theorem for directed graphs](#). *SIAM Review* 62(3):716–726, 2020. arXiv:1904.12221.
- [21] Olaf Delgado-Friedrichs. [Barycentric drawings of periodic graphs](#). *Proc. 11th Symp. Graph Drawing*, 178–189, 2003. Lecture Notes Comput. Sci. 2912.
- [22] Olaf Delgado-Friedrichs. [Equilibrium placement of periodic graphs and convexity of plane tilings](#). *Discrete Comput. Geom.* 33(1):67–81, 2004.
- [23] Giuseppe Di Battista and Fabrizio Frati. From Tutte to Floater and Gotsman: On the resolution of planar straight-line drawings and morphs. Preprint, 2021. arXiv:2108.09483.
- [24] Peter Eades and Patrick Garvan. [Drawing stressed planar graphs in three dimensions](#). *Proc. 2nd Symp. Graph Drawing*, 212–223, 1995. Lecture Notes Comput. Sci. 1027.
- [25] Jeff Erickson and Patrick Lin. [A toroidal Maxwell-Cremona-Delaunay correspondence](#). *Proc. 36th Int. Symp. Comput. Geom.*, 40:1–40:17, 2020. Leibniz Int. Proc. Informatics 164, Schloss Dagstuhl–Leibniz-Zentrum für Informatik. arXiv:2003.10057.
- [26] Michael S. Floater. [Parametric tilings and scattered data approximation](#). *Int. J. Shape Modeling* 4(3–4):165–182, 1998.
- [27] Michael S. Floater. [Mean value coordinates](#). *Comput. Aided Geom. Design* 20(1):19–27, 2003.
- [28] Michael S. Floater and Craig Gotsman. [How to morph tilings injectively](#). *J. Comput. Appl. Math.* 101(1–2):117–129, 1999.
- [29] Steven J. Gortler, Craig Gotsman, and Dylan Thurston. [Discrete one-forms on meshes and applications to 3D mesh parameterization](#). *Comput. Aided Geom. Design* 23(2):83–112, 2006.

- [30] Craig Gotsman and Vitaly Surazhsky. [Guaranteed intersection-free polygon morphing](#). *Computers & Graphics* 25(1):67–75, 2001.
- [31] Joel Hass and Peter Scott. [Simplicial energy and simplicial harmonic maps](#). *Asian J. Math.* 19(4):593–636, 2015. [arXiv:1206.2574](#).
- [32] Seok-Hee Hong and Hiroshi Nagamochi. [Convex drawings of hierarchical planar graphs and clustered planar graphs](#). *J. Discrete Algorithms* 8(3):282–295, 2010.
- [33] John E. Hopcroft and Peter J. Kahn. [A paradigm for robust geometric algorithms](#). *Algorithmica* 7(1–6):339–380, 1992.
- [34] Kai Hormann and Michael S. Floater. [Mean value coordinates for arbitrary planar polygons](#). *ACM Trans. Graphics* 25(4):1424–1441, 2006.
- [35] Linda Kleist, Boris Klemz, Anna Lubiw, Lena Schlipf, Frank Staals, and Darren Strash. [Convexity-increasing morphs of planar graphs](#). *Comput. Geom. Theory Appl.* 84:69–88, 2019.
- [36] Boris Klemz. [Convex drawings of hierarchical graphs in linear time, with applications to planar graph morphing](#). *Proc. 29th Ann. Europ. Symp. Algorithms*, 57:1–57:15, 2021. Leibniz Int. Proc. Informatics 204.
- [37] François Le Gall. [Powers of tensors and fast matrix multiplication](#). *Proc. 25th int. Symb. Alg. Comput.*, 296–303, 2014.
- [38] Richard J. Lipton, Donald J. Rose, and Robert Endre Tarjan. [Generalized nested dissection](#). *SIAM J. Numer. Anal.* 16:346–358, 1979.
- [39] László Lovász. [Discrete analytic functions: An exposition](#). *Eigenvalues of Laplacians and other geometric operators*, 241–273, 2004. Surveys in Differential Geometry 9, Int. Press.
- [40] Yanwen Luo. [Spaces of geodesic triangulations of surfaces](#). Preprint, August 2020. [arXiv:1910.03070v3](#).
- [41] Yanwen Luo, Tianqi Wu, and Xiaoping Zhu. [The deformation space of geodesic triangulations and generalized Tutte’s embedding theorem](#). Preprint, May 2021. [arXiv:2105.00612](#).
- [42] Yanwen Luo, Tianqi Wu, and Xiaoping Zhu. [The deformation space of geodesic triangulations of flat tori](#). Preprint, July 2021. [arXiv:2107.05159](#).
- [43] Bojan Mohar. [Circle packings of maps—The Euclidean case](#). *Rend. Sem. Mat. Fis. Milano* 67(1):191–206, 1997.
- [44] Bojan Mohar. [Circle packings of maps in polynomial time](#). *Europ. J. Combin.* 18(7):785–805, 1997.
- [45] Douglas A. Norris. [Isometries homotopic to the identity](#). *Proc. Amer. Math. Soc.* 105(3):692–696, 1989.
- [46] Jürgen Richter-Gebert. [Realization spaces of polytopes](#). Lecture Notes Math. 1643. Springer, 1996.

- [47] Hanxiao Shen, Zhongshi Jiang, Denis Zorin, and Daniele Panozzo. [Progressive embedding](#). *ACM Trans. Graphics* 38(4):32:1–32:13, 2019. Proc. SIGGRAPH 2019.
- [48] Dvir Steiner and Anath Fischer. Planar parameterization for closed 2-manifold genus-1 meshes. *Proc. 9th ACM Symp. Solid Modeling Appl.*, 83–91, 2004.
- [49] Ernst Steinitz and Hans Rademacher. *Vorlesungen über die Theorie der Polyeder: unter Einschluß der Elemente der Topologie*. Grundlehren der mathematischen Wissenschaften 41. Springer-Verlag, 1934. Reprinted 1976.
- [50] Vitaly Surazhsky and Craig Gotsman. [Controllable morphing of compatible planar triangulations](#). *ACM Trans. Graphics* 20(4):203–231, 2001.
- [51] Vitaly Surazhsky and Craig Gotsman. [Morphing stick figures using optimal compatible triangulations](#). *Proc. 9th Pacific Conf. Comput. Graphics Appl.*, 40–49, 2001.
- [52] Vitaly Surazhsky and Craig Gotsman. [Intrinsic morphing of compatible triangulations](#). *Int. J. Shape Modeling* 9(2):191–201, 2003.
- [53] Carsten Thomassen. [Deformations of plane graphs](#). *J. Comb. Theory Ser. B* 34(3):244–257, 1983.
- [54] Carsten Thomassen. [Tutte’s spring theorem](#). *J. Graph Theory* 45(4):275–280, 2004.
- [55] William T. Tutte. The dissection of equilateral triangles into equilateral triangles. *Math. Proc. Cambridge Phil. Soc.* 44(4):463–482, 1948.
- [56] William T. Tutte. [How to draw a graph](#). *Proc. London Math. Soc.* 13(3):743–768, 1963.

A Some Bad Examples

Here we provide several concrete examples showing that barycentric methods for drawing and morphing planar graphs do not immediately generalize to graphs on the torus.

First, we give an infinite family of non-realizable positive weight vectors. Let G be any graph on the torus, and consider the weight vector λ that assigns weight 2 to a single dart d and weight 1 to every dart other than d . (There is obviously nothing special about the values 1 and 2 here.)

Lemma A.1. λ is not a realizable weight vector for G .

Proof: Let $u = \text{tail}(d)$ and $v = \text{head}(d)$. For the sake of argument, suppose the linear system $L^\lambda P = H^\lambda$ has a solution; let Γ^λ be the resulting drawing. This system remains solvable if we remove row u and arbitrarily fix p_u [48]. All dart weights in this truncated linear system are equal to 1, which implies that the drawing Γ^λ is identical to the Tutte drawing Γ^1 for the all-1s weight vector. Comparing the two linear systems, we conclude that $p_v - p_u + \tau_d = (0, 0)$; that is, the edge of d has length zero in $\Gamma^\lambda = \Gamma^1$. But this is impossible; every edge in a Tutte drawing has non-zero length [29, Lemma B.5]. \square

Next, we give an example of two realizable weight vectors for the same torus graph whose averages are not realizable. Consider the toroidal drawings of K_7 shown in Fig. 6, which differ only in the position of vertex 2. We computed mean-value weights λ and μ for these drawings,

normalized so that the weights of all edges leaving each vertex sum to 1 [27, 34]. Routine calculations (which we implemented in Python) now imply that the average weight $(\lambda + \mu)/2$ is not realizable.

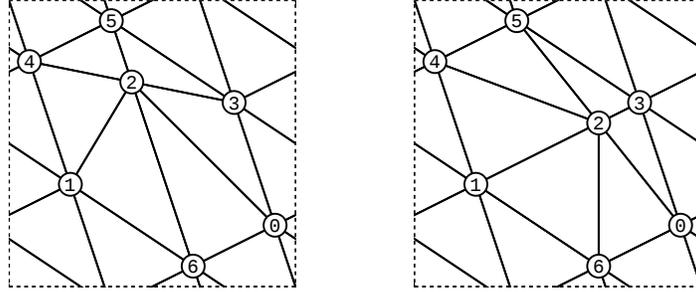


Figure 6. Isotopic drawings of K_7 whose normalized mean-value weights are not morphable.

Finally, we consider Steiner and Fischer’s approach [48] of fixing a single vertex, which restores the Laplacian linear system to full rank. The top row of Fig. 7 shows two isotopic drawings of a 12×12 toroidal grid, one with a single row of vertices shifted $1/2$ to the left, the other with a single column of vertices shifted $1/2$ downward. Let λ^{\leftarrow} and λ^{\downarrow} respectively denote the normalized mean-value weights for these drawings [27, 34]. The bottom left image in Fig. 7 shows the Steiner-Fischer drawing for the weight $\lambda = (2\lambda^{\leftarrow} + \lambda^{\downarrow})/3$, with the red edges indicating the fixed vertex. This drawing is clearly not crossing-free; it also follows that the weight vector λ is not realizable.

The bottom right of Fig. 7 shows the corresponding Floater drawing for the realizable weight vector $\mu = (2\mu^{\leftarrow} + \mu^{\downarrow})/3$, where μ^{\leftarrow} and μ^{\downarrow} are *morphable* weights derived by rescaling λ^{\leftarrow} and λ^{\downarrow} , as described in Lemma 4.3.

Steiner and Fischer claim [48, Section 2.2.1] that their drawings have no “foldovers” except possibly at the fixed vertex and its neighbors. Here a “foldover” is a vertex in the drawing whose incident faces overlap (where formally, faces are determined by the reference drawing used to define the translation vectors). Close examination of Fig. 7 shows that this claim is incorrect.

B Trying to Morph Torus Graphs Edge by Edge

The following lemma generalizes Lemma 3.1 to the toroidal setting; it also generalizes Lemma 5.1 of Chambers *et al.* [12] to work for *asymmetric* weights.

Lemma B.1. *Let λ and μ be arbitrary realizable weight vectors such that $\lambda_d \neq \mu_d$ or $\lambda_{\text{rev}(d)} \neq \mu_{\text{rev}(d)}$ for some dart d , and $\lambda_{d'} = \mu_{d'}$ for all darts $d' \notin \{d, \text{rev}(d)\}$. For every vertex w , the vector $p_w^\mu - p_w^\lambda$ is parallel to the drawing of d in Γ^λ .*

Proof: Suppose d has tail u and head v . By the definition of crossing vectors, dart d appears in Γ^λ as the projection of a dart in the universal cover from p_u^λ to $p_v^\lambda + \tau_d$. Fix a non-zero vector $\sigma \in \mathbb{R}^2$ orthogonal to the vector $p_v^\lambda - p_u^\lambda + \tau_d$ and thus orthogonal to darts $\{d, \text{rev}(d)\}$ in Γ^λ . For each vertex i , let $z_i^\lambda = p_i^\lambda \cdot \sigma$ and $z_i^\mu = p_i^\mu \cdot \sigma$, and for each dart d' , let $\chi_{d'} = \tau_{d'} \cdot \sigma$. Our choice of σ implies that $\chi_d = z_u^\lambda - z_v^\lambda$. We need to prove that $z_i^\lambda = z_i^\mu$ for every vertex i .

Let $X^\lambda = H^\lambda \cdot \sigma$ and $X^\mu = H^\mu \cdot \sigma$. The real vector $Z^\lambda = (z_i^\lambda)_i$ is a solution to the linear system $L^\lambda Z = X^\lambda$; in fact, Z^λ is the *unique* solution such that $z_n^\lambda = 0$. Similarly, $Z^\mu = (z_i^\mu)_i$ is the unique solution to an analogous equation $L^\mu Z = X^\mu$ with $z_n^\mu = 0$. We will prove that $L^\mu Z^\lambda = X^\mu$, so that in fact $Z^\lambda = Z^\mu$.

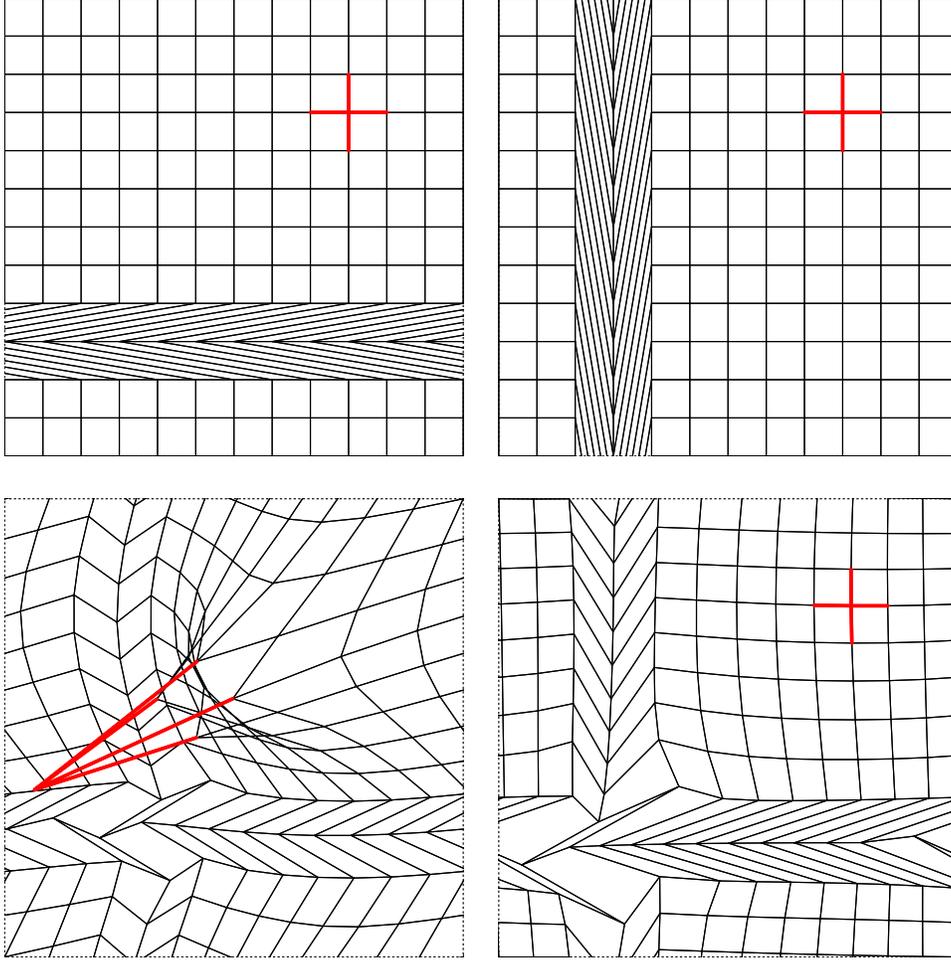


Figure 7. A bad example of fixed-vertex weight interpolation; see the text for explanation.

Let $\delta = \mu_d - \lambda_d$ and $\varepsilon = \mu_{\text{rev}(d)} - \lambda_{\text{rev}(d)}$. The matrices L^λ and L^μ differ in only four locations:

$$L_{ij}^\mu - L_{ij}^\lambda = \begin{cases} \delta & \text{if } (i, j) = (u, u) \\ -\delta & \text{if } (i, j) = (u, v) \\ -\varepsilon & \text{if } (i, j) = (v, u) \\ \varepsilon & \text{if } (i, j) = (v, v) \\ 0 & \text{otherwise} \end{cases}$$

More concisely, we have $L^\mu = L^\lambda + (\delta \mathbf{e}_u - \varepsilon \mathbf{e}_v)(\mathbf{e}_u - \mathbf{e}_v)^T$. Similar calculations imply $H^\mu = H^\lambda + \tau_d(\delta \mathbf{e}_u - \varepsilon \mathbf{e}_v)$ and therefore $X^\mu = X^\lambda + \chi_d(\delta \mathbf{e}_u - \varepsilon \mathbf{e}_v)$. It follows that

$$\begin{aligned} L^\mu Z^\lambda &= L^\lambda Z^\lambda + (\delta \mathbf{e}_u - \varepsilon \mathbf{e}_v)(\mathbf{e}_u - \mathbf{e}_v)^T Z^\lambda \\ &= X^\lambda + (\delta \mathbf{e}_u - \varepsilon \mathbf{e}_v)(z_u^\lambda - z_v^\lambda) \\ &= X^\lambda + (\delta \mathbf{e}_u - \varepsilon \mathbf{e}_v) \chi_d \\ &= X^\mu, \end{aligned}$$

completing the proof. □

Colin de Verdière [17] showed that all *symmetric* positive weights are realizable on the flat torus (see also [22, 29, 31, 39]); Chambers *et al.* [12] exploit this observation in their morphing algorithm. In the asymmetric case, however, it is unclear when changing the weights for a single edge in a realizable weight vector results in another realizable weight vector. We obtain the following partial result, via the analyses of Lemmas 4.3 and B.1.

Lemma B.2. *Let λ be a realizable weight vector, and let α be a positive row vector such that $\alpha L^\lambda = (0, \dots, 0)$ and $\alpha H^\lambda = (0, 0)$. Let μ be another positive weight vector such that $\lambda_d \neq \mu_d$ or $\lambda_{\text{rev}(d)} \neq \mu_{\text{rev}(d)}$ for some dart d , and $\lambda_{d'} = \mu_{d'}$ for all darts $d' \notin \{d, \text{rev}(d)\}$. Set $\delta := \mu_d - \lambda_d$ and $\varepsilon := \mu_{\text{rev}(d)} - \lambda_{\text{rev}(d)}$. If $\delta \alpha_{\text{tail}(d)} = \varepsilon \alpha_{\text{head}(d)}$, then μ is realizable.*

Proof: Suppose d has tail u and head v . The analysis of Lemma B.1 gives us

$$\begin{aligned} L^\mu &= L^\lambda + (\delta \mathbf{e}_u - \varepsilon \mathbf{e}_v)(\mathbf{e}_u - \mathbf{e}_v)^T \\ H^\mu &= H^\lambda + \tau_d(\delta \mathbf{e}_u - \varepsilon \mathbf{e}_v) \end{aligned}$$

Because $\alpha L^\lambda = (0, \dots, 0)$ and $\alpha H^\lambda = (0, 0)$, we immediately have $\alpha L^\mu = (\delta \alpha_u - \varepsilon \alpha_v)(\mathbf{e}_u - \mathbf{e}_v)^T$ and $\alpha H^\mu = \alpha_d(\delta \alpha_u - \varepsilon \alpha_v)$.

If $\delta \alpha_u = \varepsilon \alpha_v$, then $\alpha L^\mu = (0, \dots, 0)$ and $\alpha H^\mu = (0, 0)$. It follows that a suitable scaling of μ is morphable, and therefore realizable, which implies that μ itself is also realizable. \square

In particular, when λ is symmetric, then we can choose α to be the all-1s vector, which implies that any weight μ satisfying the conditions of Lemma B.2 is also symmetric.