How to Morph Graphs on the Torus

Erin Wolf Chambers[†]   Jeff Erickson[‡]   Patrick Lin[‡]   Salman Parsa[†]

Abstract

We present the first algorithm to morph graphs on the torus. Given two isotopic essentially 3-connected embeddings of the same graph on the Euclidean flat torus, where the edges in both drawings are geodesics, our algorithm computes a continuous deformation from one drawing to the other, such that all edges are geodesics at all times. Previously even the existence of such a morph was not known. Our algorithm runs in $O(n^{1+\omega/2})$ time, where ω is the matrix multiplication exponent, and the computed morph consists of $O(n)$ parallel linear morphing steps. Existing techniques for morphing planar straight-line graphs do not immediately generalize to graphs on the torus; in particular, Cairns' original 1944 proof and its more recent improvements rely on the fact that every planar graph contains a vertex of degree at most 5. Our proof relies on a subtle geometric analysis of 6-regular triangulations of the torus. We also make heavy use of a natural extension of Tutte's spring embedding theorem to torus graphs.

1 Introduction

Computing a morph between two given geometric objects is a fundamental problem, with applications to questions in graphics, animation, and modeling. In general, the goal is twofold: ensure the morphs are as low complexity as possible, and ensure that the intermediate objects retain the same high level structure throughout the morph.

Morphs between planar drawings are well studied in the topology, graph drawing, and computer graphics literature, with many variants. A morph between two planar straight-line embeddings Γ_0 and Γ_1 of the same planar graph is a continuous family of planar embeddings Γ_t parametrized by time, starting at Γ_0 and ending at Γ_1. In the most common formulation, all edges must be straight line segments at all times during the morph; there are then many variables of how to optimize the morph.

In this paper, we consider the more general setting of morphs between two isotopic embeddings of the same graph on the flat torus. To our knowledge, ours is the first algorithm to morph graphs on any higher-genus surface. In fact, it is the first algorithm to compute *any* form of isotopy between surface graphs; existing algorithms to test whether two graphs on the same surface are isotopic are non-constructive [29]. Our algorithm outputs a morph consisting of $O(n)$ steps; within each step, all vertices move along parallel geodesics at (different) constant speeds, and all edges remain geodesics ("straight line segments"). Our algorithm runs in $O(n^{1+\omega/2})$ time; the running time is dominated by repeatedly solving a linear system encoding a natural generalization of Tutte's spring embedding theorem.

1.1 Prior Results (and Why They Don't Generalize).
Cairns [20, 21] was the first to prove the existence of a straight-line continuous deformation between any two isomorphic planar straight-line triangulations. A long series of later works, culminating in papers by Alamdari et al. [1] and Kleist et al. [54], improved and generalized Cairns' argument to apply to arbitrary planar straight-line graphs, to produce morphs with polynomial complexity, and to derive efficient algorithms for computing those morphs. (For a more detailed history of these results, we refer the reader to Alamdari et al. [1] and Roselli [76].) Cairns' inductive argument and its successors fundamentally rely on two simple observations: (1) Every planar graph has at least one vertex of degree at most five, and (2) Every polygon with at most five vertices has at least one vertex in its visibility kernel. Thus, every planar straight-line graph contains at least one vertex that can be collapsed to one of its neighbors while preserving the planarity of the embedding.

Unfortunately, the first of these observations fails for graphs on the torus; it is easy to construct a triangulation of the torus in which every vertex has degree 6. Moreover, not every star-shaped hexagon has a vertex in its visibility kernel. Thus, it is no longer immediate that in *any* geodesic toroidal triangulation, one can move a vertex to one of its neighbors while maintaining a proper geodesic embedding. (Indeed, the fact that we can actually collapse such an edge is the main topic of Section 4.)

Floater and Gotsman [42] described an alternative method for morphing planar triangulations using a generalization of Tutte's spring-embedding theorem [84]. Every interior vertex in a planar triangulation can be expressed

as a convex combination of its neighbors. Floater and Gotsman's morph linearly interpolates between the coefficients of these convex combinations; Tutte's theorem implies that at all times, the interpolated coordinates are consistent with a proper straight-line embedding. Gotsman and Surazhsky later generalized Floater and Gotsman's technique to arbitrary planar straight-line graphs [47, 80, 81].

At its core, Floater and Gotsman's algorithm relies on the fact that the system of linear equations expressing vertices as convex combinations of their neighbors has full rank. An analogous system of equations describes equilibrium embeddings of graphs on the torus [38, 46]; however, for graphs with n vertices, this linear system has $2n$ equations over $2n$ variables (the vertex coordinates), but its rank is only $2n - 2$. If the linear system happens to have a solution, that solution is consistent with a proper embedding [28, 32, 46, 62]; unfortunately, the system is not always solvable.

When the coefficients associated with each edge are symmetric, the linear system has a two-dimensional set of solutions, which correspond to proper embeddings that differ only by translation. (See our Theorem 2.1 below.) Thus, if two given triangulations can both be described by symmetric coefficients, linearly interpolating those coordinates yields an isotopy [30]. Otherwise, however, even if the initial and final coefficient vectors are feasible, weighted averages of those coefficients might not be. Steiner and Fisher [79] modify the linear system by fixing one vertex, restoring full rank. However, while the solution to this linear system always describes a geodesic drawing of the graph, edges in that drawing can cross. In either setting, linearly interpolating the edge coefficients does not yield a morph.

Both of these approaches produce planar morphs that require high numerical precision to describe exactly. Barerra-Cruz et al. [11] describe an algorithm to morph between two isomorphic *weighted Schnyder drawings* of the same triangulation, each determined by a Schnyder wood together with an assignment of positive weights to the faces. The resulting morph consists of $O(n^2)$ steps, where after each step, all vertices lie on a $6n \times 6n$ integer grid. The algorithm relies crucially on the fact that the set of Schnyder woods of a planar triangulation is a distributive lattice [41]. Despite some initial progress by Barerra-Cruz [9], it is still an open question whether this algorithm can be extended to arbitrary planar triangulations, or even to arbitrary planar straight-line graphs. Beyond that, it is also not clear whether this result can be extended to toroidal graphs. Castelli Aleardi et al. [22] and Gonçalves and Lévêque [45] describe natural generalizations of Schnyder woods to graphs on the torus; however, the Schnyder woods (or 3-orientations) of a toroidal triangulation do not form a distributive lattice.

Considerably less is known about morphing graphs on higher-genus surfaces. Like earlier planar morphing algorithms, our algorithm follows the same inductive strategy as several earlier algorithms for transforming combinatorial embeddings into geodesic embeddings on the torus [55, 65, 66]. Our algorithm most closely resembles an algorithm of Kocay et al. [55], which transforms any essentially 3-connected toroidal embedding into an isotopic geodesic embedding, by repeatedly collapsing vertices with degree at most 5 until the embedding becomes a 6-regular triangulation.

1.2 Our Results. We begin by reviewing relevant definitions and background in Section 2. Most importantly, we review a natural generalization of Tutte's spring embedding theorem [84] to graphs on the flat torus, first proved by Y. Colin de Verdière [28]; see Theorem 2.1. We present a technical overview of our contributions in Section 3, deferring details to later sections for clarity.

Like many previous planar morphing papers, most of our paper is devoted to computing *pseudomorphs* between *triangulations*. A pseudomorph is a continuous deformation in which vertices are allowed to coincide during the motion but edges are not allowed to cross. Our pseudomorph algorithm uses two different operations that reduce the complexity of the graph: *direct collapses*, which move one vertex to one of its neighbors, and *spring collapses*, which increase the weight of one edge to infinity while maintaining an equilibrium embedding, as described by Theorem 2.1. The heart of our result is a novel analysis of 6-regular toroidal triangulations in Section 4, which implies that every non-trivial toroidal triangulation contains at least one edge that can be directly collapsed without introducing any crossings. We regard this analysis as the main technical contribution of our paper. We describe and analyze spring collapses in Section 5, again relying on Theorem 2.1. We describe and analyze the base case of our pseudomorph algorithm in Section 6: a special class of triangulations we call *zippers*, where every vertex is incident to a loop.

In Section 7, we show that a mild generalization of techniques from Alamdari et al. [1] can be used to perturb our pseudomorph into a proper morph; this perturbation technique gives us our final morphing algorithm for triangulations. Finally, in Section 8, we describe a simple reduction from morphing essentially 3-connected geodesic toroidal embeddings to morphing triangulations, again using Theorem 2.1. We conclude in Section 9 with some open problems and future directions to consider.

2 Background and Definitions

2.1 The Flat Torus. The *(square) flat torus* $\mathbb{T}$ is the metric space obtained by identifying opposite sides of the
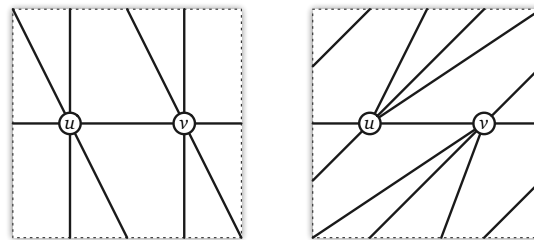
unit square $[0,1]^2$ in the Euclidean plane via $(x,0) \sim (x,1)$ and $(0,y) \sim (1,y)$. See Figure 2.1. Equivalently, $\mathbb{T}$ is the quotient space $\mathbb{T} = \mathbb{R}^2/\mathbb{Z}^2$, obtained by identifying every pair of points whose x- and y-coordinates differ by integers. The function $\pi: \mathbb{R}^2 \to \mathbb{T}$ defined by $\pi(x,y) = (x \bmod 1, y \bmod 1)$ is called the *covering* map or **projection** map.

A **geodesic** in $\mathbb{T}$ is the projection of any line segment in $\mathbb{R}^2$; geodesics are the natural analogues of "straight line segments" on the flat torus.[1] We emphasize that a geodesic is *not* necessarily the shortest path between its endpoints; indeed, there are infinitely many geodesics between any two points on $\mathbb{T}$. A **closed geodesic** in $\mathbb{T}$ is any geodesic whose endpoints coincide; the two ends of any closed geodesic are locally collinear.

2.2 Toroidal Embeddings.
Geodesic toroidal drawings are the natural generalizations of straight-line planar graphs to the flat torus. Formally, a **geodesic toroidal drawing** Γ of a graph G is a mapping of vertices to distinct points of $\mathbb{T}$ and edges to non-intersecting geodesics between their endpoints. Following standard usage in topology, we refer to any such drawing as **embedding**, to emphasize that edges do not cross.[2]

A **homotopy** between two (not necessarily injective) drawings Γ_0 and Γ_1 of the same graph G is a continuous function $H: [0,1] \times G \to \mathbb{T}$ where $H(0,) = \Gamma_0$ and $H(1,) = \Gamma_1$. A cycle on $\mathbb{T}$ is **contractible** if it is homotopic to a single point and **non-contractible** otherwise. A homotopy is an **isotopy** if each intermediate function $H(t,)$ is injective. In other words, an isotopy is a continuous family of embeddings $(\Gamma_t)_{t \in [0,1]}$ that interpolates between Γ_0 and Γ_1. (Edges in these intermediate embeddings Γ_t are not required to be geodesics.)

Two toroidal embeddings of the same graph need not be isotopic, even if they have the same rotation system; see Figure 2.1. A recent algorithm of É. Colin de Verdière and de Mesmay [29] can decide whether two toroidal drawings of the same graph are isotopic in linear time; we describe an arguably simpler linear-time algorithm in Appendix A. However, neither of these algorithms actually construct an isotopy if one exists; rather, they check whether the

---

[1] Identifying opposite sides of any other parallelogram yields a different flat torus. All flat tori are related by homeomorphisms induced by linear transformations that map geodesics to geodesics, and therefore map morphs to morphs. Thus, our results automatically apply to embeddings on any flat torus. On the other hand, our results do not apply to geodesic embeddings on the standard torus of revolution in $\mathbb{R}^3$; geodesics on that surface have radically different behavior.

[2] Formally, an embedding is a continuous injective map from the graph (as a topological space) to the torus $\mathbb{T}$. We note that this usage differs from standard terminology in many other graph drawing papers, where "embedding" refers to either a homeomorphism class of (not necessarily injective) drawings or a rotation system.


Figure 2.1. Two combinatorially equivalent but non-isotopic geodesic toroidal triangulations with parallel edges and loops. Opposite edges of the square are identified.

two embeddings satisfy certain topological properties that characterize isotopy [57–59].

We explicitly consider embeddings of graphs with parallel edges and loops. In every geodesic toroidal embedding, every loop is non-contractible (since otherwise it would be a single point), and no two parallel edges are homotopic (since otherwise they would coincide). In this paper, we consider *only* geodesic embeddings; we occasionally omit the word "geodesic" when it is clear from context.

The **universal cover** $\widetilde{\Gamma}$ of a geodesic toroidal embedding Γ is the unique infinite straight-line plane graph whose projection to $\mathbb{T}$ is Γ; that is, the projection of any vertex, edge, or face of $\widetilde{\Gamma}$ is a vertex, edge, or face of Γ, respectively. A **lift** of any vertex u in Γ is any vertex in the preimage $\pi^{-1}(u) \subset V(\widetilde{\Gamma})$. Similarly, each edge of Γ lifts to an infinite lattice of parallel line segments in $\mathbb{R}^2$, and each face lifts to an infinite lattice of congruent polygons.
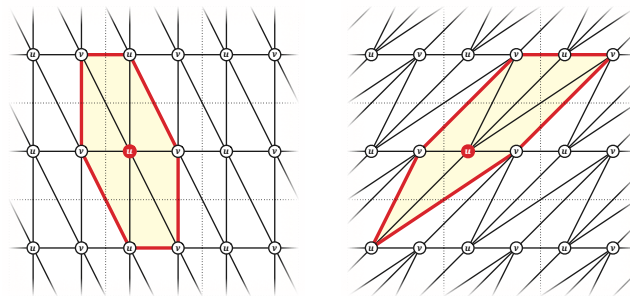
Figure 2.2. Universal covers of the geodesic embeddings from Figure 2.1, with the link of one vertex emphasized in each.

The **link** of a vertex $\widetilde{u}$ in the universal cover $\widetilde{\Gamma}$ is the simple polygon formed by the boundary of the union of the (closed) faces incident to $\widetilde{u}$; the vertices of the link are the neighbors of $\widetilde{u}$. We emphasize that when projecting a link down to the flat torus, the vertices and edges of the link need not remain distinct; see Figure 2.1 for an example. For a vertex u in Γ, we informally write "link of u" to refer to the link of an arbitrary lift $\widetilde{u}$ of u. Because the links of

any two lifts are congruent, any property proven about one lift applies to all of the others.

Geometric properties of geodesics, polygons, and embeddings on the flat torus are defined by projection from the universal cover. For example, the angle between two edges (or geodesics) e and e' at a common vertex u is equal to the angle between lifts $\tilde{e}$ and $\tilde{e}'$ at a common lift $\tilde{u}$. Similarly, the cyclic order of edges around a vertex u of Γ is the cyclic order of the corresponding edges around an arbitrary lift $\tilde{u}$. In particular, if u is incident to a loop, that loop appears twice in cyclic order around u, and each lift $\tilde{u}$ of u is incident to two different lifts of that loop. Finally, angles in the link of a vertex in Γ are projections of angles in the link of an arbitrary lift $\tilde{u}$.

A toroidal embedding Γ is a **triangulation** if every face of Γ is bounded by three edges, or equivalently, if its universal cover $\widetilde{\Gamma}$ is a planar triangulation. In particular, we do not insist that triangulations are simplicial complexes. Every *geodesic* toroidal embedding Γ is **essentially simple**, meaning its universal cover $\widetilde{\Gamma}$ is a planar embedding of a simple (albeit infinite) graph. A geodesic toroidal drawing Γ is **essentially 3-connected** if its universal cover $\widetilde{\Gamma}$ is 3-connected [45, 67–70]; every geodesic triangulation is essentially 3-connected.

2.3 Coordinates and Crossing Vectors.
To represent an arbitrary straight-line embedding of a graph in the plane, it suffices to record the coordinates of each vertex; each edge in the embedding is the unique line segment between its endpoints. However, vertex coordinates alone are not sufficient to specify a toroidal embedding; intuitively, we must also specify how the edges of the graph wrap around the surface.

Formally, we regard each edge of the graph G as a pair of opposing *half-edges* or **darts**, each directed from one endpoint, called the **tail**, toward the other endpoint, called the **head**. We write $rev(d)$ to denote the reversal of any dart d; thus, for example, $head(rev(d)) = tail(d)$ and $rev(rev(d)) = d$ for every dart d.

We can represent any geodesic embedding of any graph G onto the torus by associating a **coordinate** vector $p(v) \in [0,1)^2$ with every vertex v of G and a **crossing** vector $x(d) \in \mathbb{Z}^2$ with every dart d of G. The coordinates of a vertex specify its position in the unit square; to remove any ambiguity, we assign points on the boundary of the unit square coordinates on the bottom and/or left edges. The crossing vector of a dart records how that dart crosses the boundaries of the unit square. Specifically, the first coordinate of $x(d)$ is the number of times d crosses the vertical boundary to the rightward (with negative numbers counting leftward crossings), and the second coordinate of $x(d)$ is the number of times d crosses the horizontal boundary upward (with negative numbers

counting downward crossings). Crossing vectors are anti-symmetric: $x(rev(d)) = -x(d)$ for every dart d. See Figure 2.3.
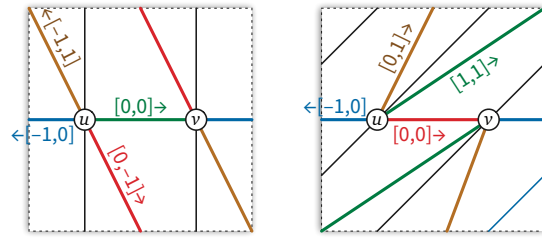
Figure 2.3. The geodesic embeddings from Figure 2.1, showing the crossing vectors of all four darts from u to v.

Crossing vectors and their generalizations have been used in several previous algorithms for surface graphs [23–25, 37, 39, 40] and simplicial complexes [19, 34, 35] to encode the homology classes of cycles. Crossing vectors are also equivalent to the *translation vectors* traditionally used to model periodic (or "dynamic") graphs [27, 31, 48–50, 52, 56, 73, 74, 85] and more recently used to model periodic bar-and-joint frameworks [16, 17, 33, 53, 64, 72, 77, 78].

In principle, our morphing algorithm can be modified to update the coordinates of any vertex v and the crossing vectors of darts incident to v whenever v crosses the boundary of the unit square, with only a small penalty in the running time. But in fact, this maintenance is not necessary; it suffices to modify *only* the vertex coordinates, keeping all crossing vectors fixed throughout the entire morph, even when vertices cross the boundary of the unit square. We describe how to interpret toroidal embeddings with these more relaxed coordinates in Appendix A.

2.4 Equilibrium Embeddings.
We make frequent use of the following natural generalization of Tutte's "spring embedding" theorem for 3-connected planar graphs [84], first proved by Y. Colin de Verdière [28] and later independently reproved by several others [32, 46, 62]:

THEOREM 2.1. *Let Γ be any essentially 3-connected geodesic toroidal drawing, where each edge e has an associated weight $\lambda(e) > 0$. Then Γ is isotopic to a geodesic embedding Γ_* in $\mathbb{T}$ such that every face is convex and each vertex is the weighted center of mass of its neighbors; moreover, this equilibrium embedding is unique up to translation.*

The equilibrium embedding Γ_* can be computed by solving a linear system for the vertex coordinates $p_*(v)$, treating the crossing vectors $x(d)$ and weights $\lambda(d)$ as constants [28, 38, 46, 79]. For each vertex v, the system contains the constraint

$$(\star) \quad \sum_{tail(d)=v} \lambda(d) \cdot \big(p_*(head(d)) + x(d) - p_*(v)\big) = (0,0),$$

where $\lambda(d) = \lambda(rev(d))$ is the weight of the edge containing dart d. This linear system has a two-dimensional set of solutions, which differ by translation [28, 46]; we can remove this ambiguity by fixing $p_*(r) = (0, 0)$ for some arbitrary *root* vertex r [79]. Some vertex coordinates $p_*(v)$ in the solution to this system may lie outside the unit square; as we explain in Appendix A, it is possible to move all coordinates back into the unit square by appropriately adjusting the crossing vectors, but in fact no such adjustment is necessary.

The support of this linear system is the toroidal graph G, which has balanced separators of size $O(\sqrt{n})$ [3]. Thus, the linear system can be solved in $O(n^{\omega/2})$ time using the generalized nested dissection technique of Lipton et al. [61], where $\omega < 2.37286$ is the matrix multiplication exponent [4, 60].

2.5 Morphs and Pseudomorphs.
A *morph* between two isotopic geodesic toroidal drawings Γ_0 and Γ_1 is a continuous family of geodesic drawings $(\Gamma_t)_{t \in [0,1]}$ from Γ_0 to Γ_1; in other words, a morph is a geodesic isotopy between Γ_0 and Γ_1. Any morph is completely determined by the continuous motions on the vertices; geodesic edges update in the obvious way.

A morph is *linear* if every vertex moves along a geodesic from its initial position to its final position at a uniform speed, and a morph is *parallel* if all vertices move along parallel geodesics, that is, along projections of parallel line segments.[3] In this paper, we construct morphs that consist of a sequence of $O(n)$ parallel linear morphs. Every morph of this type can be specified by a sequence of isotopic geodesic toroidal embeddings $\Gamma_0, \ldots, \Gamma_k$ and for each index i, a set of parallel geodesics connecting the vertices of Γ_i to corresponding vertices of Γ_{i+1}.

Like many previous planar morphing algorithms, our morphing algorithm first constructs a *pseudomorph*, which is a continuous family of drawings in which edges remain geodesics, vertices may become coincident, but edges never cross. The most basic ingredient in our pseudomorph is an *edge collapse*, which moves the endpoints of one edge together until they coincide.[4] Collapsing an edge also collapses the faces on either side of that edge to single edges; edge collapses also preserve essential 3-connectivity. Our final morph is obtained by carefully

---

[3] Alamdari et al. [1] and their predecessors [7, 10] call these "unidirectional" morphs; however, this term suggests incorrectly that vertices cannot move in *opposite* directions. Our usage differs from other papers, which use "parallel morph" to describe a morph that keeps each edge parallel to its original embedding [12–15].

[4] This procedure is often called edge *contraction* in planar graph morphing literature; we use the term "collapse" to avoid any confusion with the topological notion of contractible cycles.

perturbing a pseudomorph consisting of edge collapses and their reversals.

3 Technical Overview

In this section, we give a technical overview of our results: at a high level, first we develop an algorithm to compute a *pseudomorph* between geodesic toroidal *triangulations*, and then we describe how to use this algorithm to compute a morph between essentially 3-connected geodesic toroidal embeddings. Here we give only a brief overview of several necessary tools; each of these components is developed in detail in a later section of the paper.

3.1 List of Ingredients. We begin with some essential subroutines and structural results.

3.1.1 Direct collapses. Following Cairns' approach [20] and its later derivatives [1, 2, 6, 7, 10], a *direct collapse* consists of moving a vertex u along some edge to another vertex v, at uniform speed, until u and v coincide, keeping all other vertices fixed. To simplify our presentation, we require that the moving vertex u is not incident to a loop. We informally call a vertex *good* if it is not incident to a loop and it can be directly collapsed to one of its neighbors without introducing any edge crossings, and *bad* otherwise; see Section 4 for a simple geometric characterization of good vertices.

As noted by Cairns [20], every vertex of degree at most 5 not incident to a loop is good; indeed, this fact, along with the fact that every planar graph has a vertex of degree at most 5, forms the basis of Cairns' approach and its derivatives for computing (pseudo)morphs between straight-line planar drawings. We prove in Lemma 6.1 that in any geodesic toroidal triangulation, a vertex of degree at most 5 cannot be incident to a loop, so in fact all vertices of degree at most 5 are good.

On the other hand, Euler's formula implies that the average degree of a toroidal graph is exactly 6, and there are simple examples of degree-6 vertices that are bad. Morphing between torus graphs thus requires new techniques to handle the special case of 6-regular triangulations.

3.1.2 6-regular triangulations. We then prove in Lemma 6.2 that if a 6-regular toroidal triangulation contains a loop, then in fact *every* vertex is incident to a loop; we call this special type of triangulation a *zipper*. Because all the loops in any non-trivial zipper are parallel, morphing between isotopic zippers Z_0 and Z_1 turns out be straightforward. If the zippers have only one vertex, they differ by a single translation. Otherwise, two parallel linear morphs suffice, first sliding the loops of Z_0 to coincide with the corresponding loops in Z_1, and then rotating the loops to
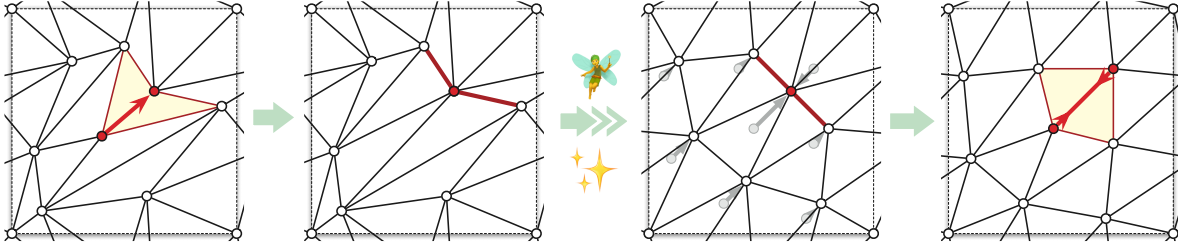
Figure 3.1. Our pseudomorph consists of a direct collapse, a recursive pseudomorph, and a reversed spring collapse.

move vertices and non-loop edges to their locations in Z_1. We describe and analyze zippers in detail in Section 6.

We analyze 6-regular triangulations without loops in detail in Section 4; we regard this analysis as the main technical contribution of the paper. Averaging arguments imply that in any 6-regular triangulation where every vertex is bad, in short a *bad triangulation*, every vertex link is one of two specific non-convex hexagons that we call *cats* and *dogs*, illustrated in Figure 4.4. Analysis of how cats and dogs can overlap implies that any bad triangulation must contain a non-contractible cycle of vertices (each of whose link is a cat) that consistently turns in the same direction at every vertex, and therefore has non-zero turning angle, contradicting the fact that the total turning angle of every non-contractible cycle on the flat torus is zero [75]. We conclude that bad triangulations do not exist; *every* geodesic toroidal triangulation that is not a zipper contains at least one good vertex.

3.1.3 Equilibria and spring collapses. Suppose we are given two isotopic geodesic triangulations Γ_0 and Γ_1 that are not zippers. Our analysis implies that Γ_0 and Γ_1 each contain at least one good vertex. However, it is possible that no vertex is good in both Γ_0 and Γ_1. More subtly, even if some vertex u is good in both triangulations, that vertex may be collapsible along a unique edge e_0 in Γ_0 but along a different unique edge e_1 in Γ_1.

The second problem also occurs for straight-line planar embeddings. Cairns' solution to this problem was to introduce an intermediate triangulation $\Gamma_{1/2}$ in which u can be collapsed along both e_0 and e_1. Recursively constructing pseudomorphs from Γ_0 to $\Gamma_{1/2}$ and from $\Gamma_{1/2}$ to Γ_1 yields a pseudomorph from Γ_0 to Γ_1 with exponentially many steps. Subsequent refinements of Cairns' approach, culminating in the work of Alamdari et al. [1] and later improvement by Kleist et al. [54], obtained a pseudomorph with only polynomial complexity by finding clever ways to avoid this intermediate triangulation.

Our algorithm does introduce one intermediate triangulation, but still avoids the exponential blowup of Cairns' algorithm. Specifically, we use an equilibrium triangulation Γ_* isotopic to Γ_0 and Γ_1, as given by Theorem 2.1.

A vertex that is good in Γ_0 might still be bad in Γ_*, so instead of applying a direct collapse to Γ_*, we introduce a novel method for collapsing edges in an equilibrium embedding in Section 5. Intuitively, we continuously increase the weight of an arbitrary edge e to infinity, while maintaining the equilibrium triangulation given by Theorem 2.1. This **spring collapse** moves the endpoints of e together, just like a direct collapse. By analyzing the solutions to the equilibrium linear system ($\star$), we show in Section 5.1 that a spring collapse is a parallel pseudomorph, and in fact can be simulated by an equivalent parallel *linear* pseudomorph. Moreover, this parallel linear pseudomorph can be computed by solving a single instance of system ($\star$).

3.2 Recursive Pseudomorph Between Triangulations. We are now ready to describe our recursive algorithm to compute a pseudomorph between two isotopic geodesic triangulations Γ_0 and Γ_1. We actually explain how to compute a pseudomorph Ψ_0 from Γ_0 to an isotopic equilibrium triangulation Γ_*. The same algorithm gives a pseudomorph Ψ_1 from Γ_1 to Γ_*, and concatenating Ψ_0 with the reversal of Ψ_1 yields the desired pseudomorph from Γ_0 to Γ_1.

If Γ_0 is a zipper, we morph directly between Γ_0 and the equilibrium zipper Γ_* using at most two parallel linear morphs. This is the base case of our recursive algorithm.

If Γ_0 is not a zipper, then it contains a good vertex u. By definition, u can be directly collapsed along some edge e to another vertex v, without introducing edge crossings. This direct collapse gives us a parallel linear pseudomorph from Γ_0 to Γ_0', a geodesic toroidal triangulation whose underlying graph G' has $n-1$ vertices. On the other hand, performing a spring collapse in Γ_* by increasing the weight of the same edge e to ∞ leads to a drawing where u and v coincide, that is, an equilibrium triangulation Γ_*' of G' that is isotopic to Γ_0'. Finally, because Γ_0' and Γ_*' are isotopic embeddings of the same graph G', we can compute a pseudomorph from Γ_0' to Γ_*' recursively.

Our full pseudomorph from Γ_0 to Γ_* thus consists of the direct collapse from Γ_0 to Γ_0', followed by the recursive pseudomorph from Γ_0' to Γ_*', followed by the reverse of the spring collapse from Γ_* to Γ_*'. See Figure 3.1.

Altogether our pseudomorph consists of $O(n)$ parallel linear pseudomorphs: at most n direct collapses to reach a zipper, $O(1)$ parallel linear pseudomorphs to reach an equilibrium zipper, and finally at most n reversed spring collapses. The time to compute the overall pseudomorph is dominated by the time needed to compute the $O(n)$ equilibrium embeddings; the overall running time is $O(n^{1+\omega/2})$.

3.3 Morphing Between Embeddings. In Section 7, we show that a technique introduced by Alamdari et al. [1] for perturbing planar pseudomorphs into morphs can be applied with only minor modifications to our toroidal pseudomorphs, to produce morphs between geodesic toroidal triangulations. This perturbation step adds only $O(n^2)$ overhead to the running time of our algorithm, and the resulting morphs consist of $O(n)$ parallel linear morphs.

Finally, in Section 8, we describe a simple reduction from morphing arbitrary essentially 3-connected embeddings to the special case of morphing triangulations. In brief, given any triangulation of any essentially 3-connected geodesic toroidal embedding Γ, the isotopic equilibrium embedding Γ_* given by Theorem 2.1 can be triangulated in the same way, and any morph between the two triangulations induces a morph between Γ and Γ_*. This reduction preserves the number of parallel linear morphs and adds only $O(n^{\omega/2})$ overhead to the running time.

In summary, given any two isotopic essentially 3-connected geodesic toroidal embeddings, we can construct a morph between them, consisting of $O(n)$ parallel linear morphs, in $O(n^{1+\omega/2})$ time.

4 Cats and Dogs

In this section, we present the core of our technical analysis: the proof that every geodesic toroidal triangulation without loops has a (directly) collapsible edge.

The **visibility kernel** of a simple polygon P is the set of all points in P that can "see" all of P; more formally, the visibility kernel is $\{p \in P \mid pq \subseteq P \text{ for all } q \in P\}$. The visibility kernel is always convex. If P is the link of a vertex v in a geodesic triangulation, then v must lie in the visibility kernel of P. We call a simple polygon **good** if its visibility kernel contains a vertex of the polygon, and **bad** otherwise. All triangles, quadrilaterals, and pentagons are good [20], but some hexagons are bad; Figure 4.1 shows several examples.

A vertex u of a geodesic toroidal triangulation is **good** if it is not incident to a loop, and the link of any (and thus every) lift $\tilde{u}$ of u in $\widetilde{\Gamma}$ is good, and **bad** otherwise. A good vertex can be safely collapsed to any neighbor in the visibility kernel of its link. Finally, a geodesic toroidal triangulation Γ with no loops is **good** if it contains at least one good vertex, and **bad** otherwise. The main result of
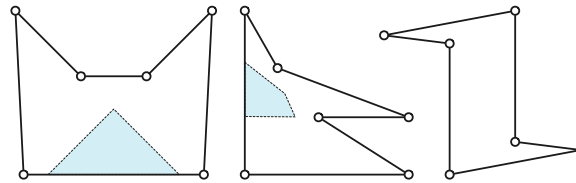
Figure 4.1. Three bad hexagons. Visibility kernels are shaded in blue; the third visibility kernel is empty.

this section is that bad triangulations do not exist; that is, every geodesic triangulation without loops is good.

LEMMA 4.1. *Every bad triangulation is 6-regular.*

Proof. Every vertex in a bad triangulation must have degree at least 6, because every vertex with degree at most 5 is good [20]. On the other hand, Euler's formula for the torus implies that the average degree is exactly 6. □

Like Alamdari et al. [1], the following analysis of 6-regular triangulations without loops implicitly assumes that the vertices are in general position. Specifically, for every vertex $\tilde{v}$ in the universal cover, we assume that (1) no pair of edges incident to $\tilde{v}$ are collinear, and (2) no edge of the link of $\tilde{v}$ is collinear with another vertex of that link. Because the underlying triangulation has no loops, each nondegeneracy condition involves at least two distinct vertices of the torus triangulation, so like Alamdari et al. [1], we can enforce these conditions if necessary by perturbing the vertices.

LEMMA 4.2. *In every bad triangulation, the link of each vertex has exactly two reflex vertices.*

Proof. If a simple polygon is convex or has exactly one reflex vertex, then it is good: the visibility kernel of a convex polygon is the polygon itself, and if there is exactly one reflex vertex, then the reflex vertex is in the visibility kernel. So each link in a bad triangulation has *at least* two reflex vertices. We argue next that the average number of reflex vertices per link is *at most* two, which implies that every link has exactly two reflex vertices.

Let Γ be any (not necessarily bad) 6-regular geodesic triangulation. A *corner* of a vertex v in Γ is the angle between two edges that are adjacent in cyclic order around v. If v is a vertex of the link of another vertex x, then the link of x contains exactly two adjacent corners of v. Moreover, if v is a *reflex* vertex of the link of x, those two corners sum to more than half a circle.

Thus, if v is reflex in the link of two neighbors x and y, the links of x and y must share a corner of v, which implies that edges vx and vy are adjacent in cyclic order around v; see Figure 4.2. If v were reflex in the links of

three neighbors x, y, and z, then all three edge pairs vx, vy and vx, vz and vy, vz would be adjacent around v, which is impossible because v has degree 6.
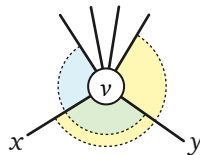

Figure 4.2. Each vertex in a 6-regular triangulation is reflex in at most two links

We conclude that each vertex in Γ is a reflex vertex of the links of at most two other vertices. It follows that the average number of reflex vertices in a link is at most two, which completes the proof. □

LEMMA 4.3. *A bad hexagon with two reflex vertices separated by two convex vertices has an empty visibility kernel, and thus is not the link of any vertex.*

Proof. Consider a bad hexagon P whose reflex vertices are separated by two convex vertices. Label the vertices a through f in cyclic order such that c and f are the reflex vertices, as shown in Figure 4.3.
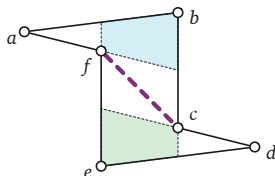
Figure 4.3. A hexagon with two reflex vertices that are separated by two convex vertices. The dashed diagonal splits the hexagon into two non-convex quadrilaterals with disjoint visibility kernels.

Because P is bad, neither c nor f lies in its visibility kernel. Vertex c can see vertices b, d, and f, so it cannot see both a and e. Without loss of generality, suppose c cannot see a. Then f is a reflex vertex of the quadrilateral $abcf$. Every quadrilateral has at most one reflex vertex, and that reflex vertex is in the visibility kernel, so f can see b. It follows that f cannot see d, and c is a reflex vertex of the quadrilateral $cdef$.

The visibility kernels of quadrilaterals $abcf$ and $cdef$ are disjoint, which implies that the visibility kernel of P is empty. □

For the remainder of the proof, we annotate the edges of any triangulation as follows. The *star* of an edge in a triangulation is the union of the faces incident to that edge. An edge is *flippable* if its star is convex, and non-flippable otherwise. Every non-flippable edge is incident

to the unique reflex vertex of its star; we direct each non-flippable edge away from this reflex vertex.

LEMMA 4.4. *In any bad triangulation, every vertex is incident to exactly two incoming directed edges, exactly two outgoing directed edges, and exactly two flippable edges.*

Proof. Fix a bad triangulation Γ. If w is a reflex vertex in the link of some vertex v, then the edge vw is directed toward v. So Lemma 4.2 implies that each vertex of Γ is incident to two incoming edges.

We argued in the proof of Lemma 4.2 that each vertex is reflex in the links of exactly two of its neighbors. Thus, each vertex of Γ is also incident to two outgoing edges.

Finally, because every vertex of Γ has degree 6, each vertex of Γ is incident to two flippable edges. □

Lemmas 4.3 and 4.4 imply that the links in every bad triangulation fall into two categories, which we call *cats* and *dogs*. A **cat** is a bad hexagon whose reflex vertices are adjacent; a **dog** is a bad hexagon whose reflex vertices are separated by one convex vertex. Cats are (combinatorially) symmetric; however, there are two species of dogs, which are reflections of each other.
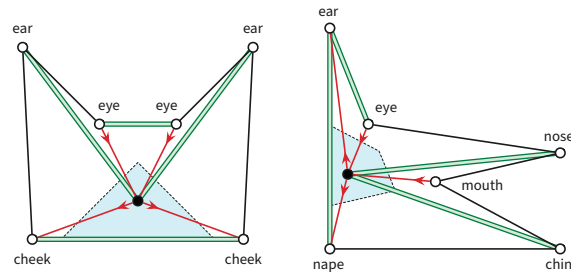
Figure 4.4. Cat and (right-facing) dog anatomy. Flippable edges are shown in double-thick green.

We label the vertices of each cat and dog mnemonically, as shown in Figure 4.4. In particular, if the link of vertex v is a dog, then $nose(v)$ is the unique convex vertex of the dog whose neighbors are reflex, and the vertex opposite $nose(v)$ in the link is $nape(v)$. Because $nape(v)$ is incident to two convex vertices, it can see every vertex of the dog except $nose(v)$. Because neither reflex vertex lies in the visibility kernel of the dog, $nape(v)$ can see both reflex vertices. The visibility between $nape(v)$ and $nose(v)$ is blocked by one of the reflex vertices, which we call $mouth(v)$. The edge from v to $nose(v)$ is flippable, because otherwise v could not see either $ear(v)$ or $chin(v)$. A dog is **right-facing** if $mouth(v)$ immediately follows $nose(v)$ in clockwise order, and **left-facing** otherwise.

The following lemma implies that two boundary edges of each cat and dog are flippable, as shown in Figure 4.4.

LEMMA 4.5. *In any bad triangulation, every triangle has exactly one flippable edge.*

Proof. Fix a bad triangulation Γ with n vertices. Euler's formula implies that Γ has $3n$ edges and $2n$ triangular faces. Lemma 4.4 implies that Γ has exactly n flippable edges, so the average number of flippable edges per triangle is exactly 1.

The flippable edges incident to any vertex v are separated in cyclic order around v by two non-flippable edges if the link of v is a cat, or by one non-flippable edge if the link of v is a dog. Thus, two flippable edges never appear consecutively around a common vertex. It follows that every triangle in Γ is incident to at most one flippable edge. $\square$

LEMMA 4.6. *Every bad triangulation contains a cat.*

Proof. Let Γ be a triangulation, and let u be any vertex in Γ whose link is a dog. We argue that the link of the nose of u must be a cat. (Mnemonically, dogs only sniff cats.)

Without loss of generality, assume that the link of u is facing right, so the triple $chin(u), nape(u), ear(u)$ is oriented clockwise, as shown in Figure 4.5. The fact that the link of u is a *bad* hexagon implies several orientation constraints on its vertices:

- The triple $ear(u), eye(u), mouth(u)$ is oriented counterclockwise; otherwise, $mouth(u)$ could see the entire dog. It follows that the triple $eye(u), mouth(u), nape(u)$ is oriented clockwise, and that $nape(u)$ and $eye(u)$ can see each other.

- The triple $eye(u), mouth(u), chin(u)$ is oriented counterclockwise; otherwise, $eye(u)$ could see the entire dog.

- Finally, the triple $nose(u), mouth(u), nape(u)$ is oriented counterclockwise; otherwise, $nape(u)$ could see the entire dog.
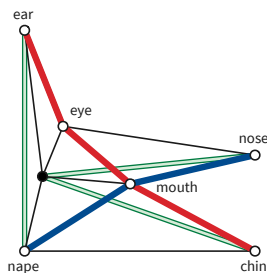

Figure 4.5. Concave chains inside any dog.

Now suppose for the sake of argument that the link of $v = nose(u)$ is a dog. We label the other vertices of the link

of u as shown on the top row of Figure 4.6; in particular, $x = eye(u)$ and $y = mouth(u)$. Because the flippable edge uv is inside the link of v, either $u = nose(v)$ or $u = chin(v)$; each of these cases admits two subcases. The four cases are illustrated schematically in the rows of Figure 4.6.
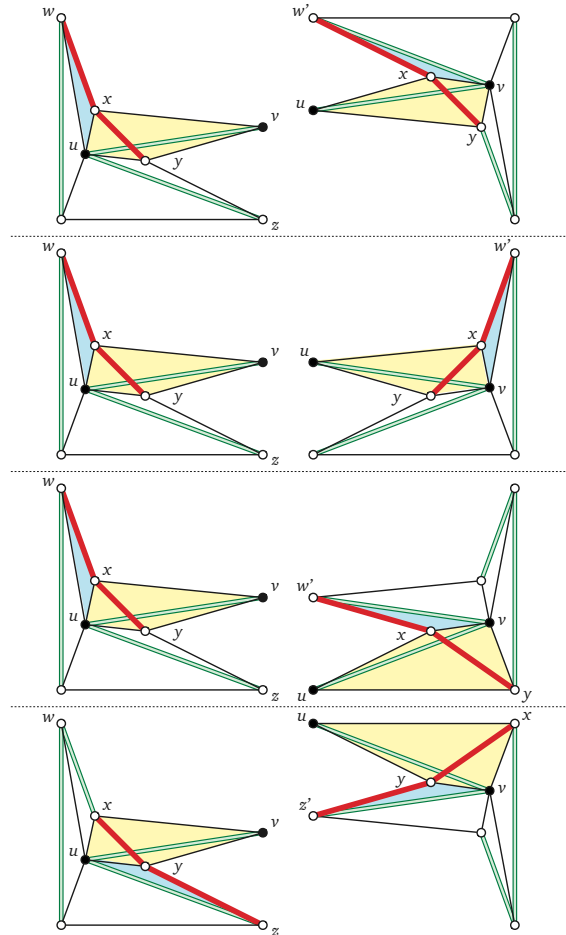

**Figure 4.6.** Four cases in the proof of Lemma 4.5: One dog's nose (v) cannot be the center of another dog.

- Suppose $u = nose(v)$ and $x = mouth(u)$, and therefore $y = eye(u)$. Let $w' = chin(v)$. The triple w', x, y must be oriented clockwise; our earlier analysis implies that w, x, y is oriented counterclockwise. It follows that triangles $w'xv$ and wxu overlap, which is impossible.

- Suppose $u = nose(v)$ and $x = eye(u)$, and therefore $y = mouth(u)$. Let $w' = ear(v)$. The triple w', x, y must be oriented clockwise; our earlier analysis implies that w, x, y is oriented counterclockwise. It follows that triangles $w'xv$ and wxu overlap, which is impossible.

- Suppose $u = chin(v)$ and $y = nape(v)$. Let $w' = nose(v)$. The triple w', x, y is oriented clockwise;

our earlier analysis implies that w, y, z is oriented counterclockwise. It follows that triangles $w'xv$ and wxu overlap, which is impossible.

- Finally, suppose $u = chin(v)$ and $y = mouth(v)$. Let $z' = nose(v)$. The triple x, y, z' is oriented clockwise; our earlier analysis implies that x, y, z is oriented counterclockwise. It follows that triangles $z'yv$ and zyu overlap, which is impossible.

In all four cases, we derive a contradiction. We conclude that the link of $nose(u)$ is actually a cat. ☐

Every 6-regular triangulation of the torus is isotopic to the quotient of the regular equilateral-triangle tiling of the plane by a lattice of translations [5, 18, 71]. We analyze the patterns of cats and dogs in bad triangulations by analyzing their images in this reference triangulation, and in particular, by studying the induced annotations of edges, as illustrated in Figure 4.7.
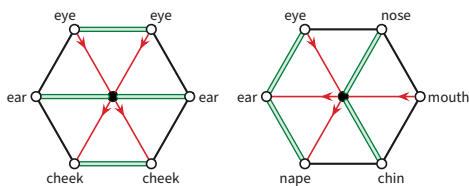
Figure 4.7. Cat and (right-facing) dog reference anatomy; compare with Figure 4.4.

A cycle in a 6-regular toroidal triangulation is **straight** if it corresponds to a closed geodesic in the corresponding lattice triangulation. Every straight cycle is non-contractible.

Let C be a simple piecewise geodesic closed curved on the torus (for example, a cycle in Γ), arbitrarily directed, and let u, v, w be three successive vertices of C. Let $\tilde{v}$ be any lift of v, and let $\tilde{u}\tilde{v}$ and $\tilde{v}\tilde{w}$ be lifts of the edges of C incident to v. The **turning angle** of C at u is the signed counterclockwise angle, strictly between $-\pi$ and π, between the vectors $\tilde{u}{\to}\tilde{v}$ and $\tilde{v}{\to}\tilde{w}$. In other words, when walking along the cycle C in the indicated direction, the turning angle is the angle one turns left at v (or right if the angle is negative). The **total turning angle** of C is the sum of the turning angles of the vertices of C.

LEMMA 4.7. *Every simple non-contractible closed curve on the flat torus has total turning angle zero.*

Lemma 4.7 follows immediately from classical results of Reinhart [75]. In short, the total turning angle (which Reinhart calls the "winding number") is an isotopy invariant of closed curves, and every simple non-contractible closed

curve on the flat torus is isotopic to a closed geodesic. Lemma 4.7 implies in particular that every *straight* cycle has total turning angle zero.

LEMMA 4.8. *If a bad triangulation contains one dog, it contains a straight cycle of vertices whose links are dogs.*

Proof. Let Γ be a bad triangulation and let v_1 be any vertex of Γ whose link is a dog. Label the vertices of v's link as shown in Figure 4.8; for example, $u_1 = nose(v_1)$, $v_2 = mouth(v_1)$, and $w_1 = chin(v_1)$.

Edges u_1v_1 and v_1w_1 are flippable, and therefore edges v_1v_2, u_1v_2, and w_1v_2 are not flippable. Thus, no opposite pair of edges incident to v_2 are both flippable. It follows that the link of v_2 is a dog, whose ear vertex is v_1.

Edges u_0v_1, v_0v_1, and v_1w_0 are not flippable, so edges u_0v_0 and v_0w_0 must be flippable. Thus, the link of v_0 is also a dog, whose mouth vertex is v_1.
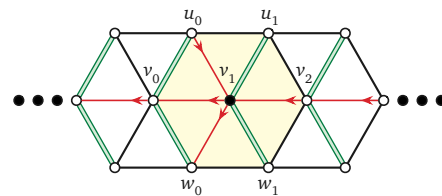
Figure 4.8. One dog induces a straight cycle of dogs.

Continuing by induction in both directions, we find a bidirectional sequence $\ldots, v_{-1}, v_0, v_1, v_2, \ldots$ of vertices whose links are dogs, where the center v_i of each dog is the mouth of the previous dog and the ear of the next dog. Because Γ is finite, this sequence must eventually repeat, forming a straight cycle. ☐

We can now finally prove the main result of this section.

THEOREM 4.1. *Bad triangulations do not exist.*

Proof. Let Γ be a bad triangulation. We derive a contradiction by showing that Γ contains a non-contractible cycle whose vertices all have cat links, whose edges are all non-flippable, and finally whose turning angle is non-zero, contradicting Lemma 4.7.

First, suppose Γ contains at least one dog. Following the proof of Lemma 4.6, suppose v_1 is a vertex whose vertex is a dog and $u_1 = nose(v_1)$. Then $v_2 = mouth(v_1)$ is also a neighbor of u_1. The previous lemma now implies a straight cycle of dogs $D = \ldots, v_0, v_1, v_2, v_3, \ldots$, where $v_i = mouth(v_{i-1}) = ear(v_{i+1})$ for every index i. For each index i, the link of $u_i = nose(v_i)$ is a cat. Thus, we find a straight cycle $C = \ldots, u_0, u_1, u_2, u_3, \ldots$ of vertices, parallel
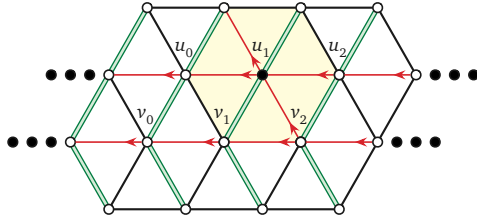
Figure 4.9. One cat induces a straight cycle of cats.

to the cycle D of dog vertices, all of whose links are cats, as shown in Figure 4.9. Every edge $u_i u_{i+1}$ in C is unflippable.

On the other hand, if Γ contains no dogs, then we can construct a straight cycle $C = \ldots, u_0, u_1, u_2, \ldots$ of cat vertices starting with any unflippable edge $u_0 u_1$. Again, every edge $u_i u_{i+1}$ in C is unflippable.

Without loss of generality, suppose u_2 is the left eye of u_1, and thus u_0 is the right cheek of u_1. (Symmetric arguments apply if u_2 is the right eye, right cheek, or left cheek of u_1.) The triple $u_0 u_1 u_2$ is oriented counterclockwise, as shown in Figure 4.10; otherwise, the right cheek of u_1 would lie in the visibility kernel of the link of u_1. It follows by induction that for every index i, vertex u_{i+1} is the left eye of u_i, and thus every triple $u_{i-1} u_i u_{i+1}$ is oriented counterclockwise. We conclude that the total turning angle of C is positive, contradicting Lemma 4.7. □
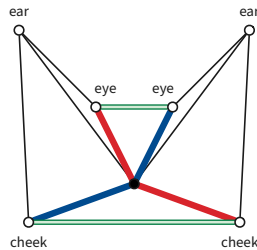

Figure 4.10. Concave chains inside any cat.

5 Equilibrium Pseudomorphs

As pointed out in Section 3, when computing a recursive pseudomorph between two triangulations Γ_0 and Γ_1, we must contend with the fact that while each triangulation must contain a good vertex, there may be no way to safely collapse the same vertex along the same edge in both triangulations. It is relatively straightforward to adapt Cairns' approach for planar graphs to the torus, by recursively computing pseudomorphs from both Γ_0 and Γ_1 to a suitable intermediate triangulation; unfortunately, the resulting pseudomorph from Γ_0 to Γ_1 consists of an exponential number of steps.

Instead, our algorithm introduces *one* intermediate triangulation isotopic to both Γ_0 and Γ_1, namely an equilibrium triangulation Γ_* guaranteed by Y. Colin de Verdière's generalization of Tutte's spring embedding theorem (Theorem 2.1). In this section, we describe how to collapse an *arbitrary* edge e in an equilibrium triangulation to obtain a simpler equilibrium triangulation. This operation allows us to recursively compute a pseudomorph from either Γ_0 or Γ_1 to Γ_* using only a linear number of steps, as described in Section 3.2.

Intuitively, we continuously increase the weight of e to ∞ and maintain the equilibrium triangulation. We show that the resulting "spring collapse" moves all vertices of the equilibrium embedding along geodesics parallel to e, as shown in Figure 5.1; in particular, edge e collapses to a single vertex. It follows that straightforward linear interpolation from one equilibrium triangulation to the other is a parallel linear pseudomorph.
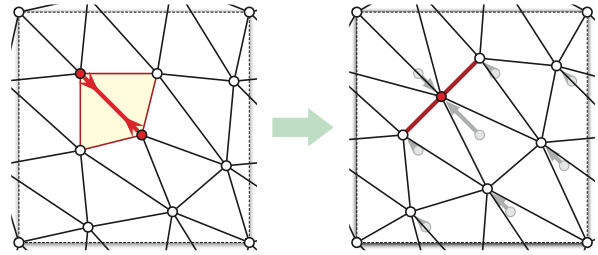

Figure 5.1. Increasing the weight of any edge e to infinity collapses e and moves every vertex parallel to e.

For any triangulation Γ and any assignment of positive weights $\lambda(e)$ to the edges of Γ, let $\boldsymbol{Eq(\Gamma, \lambda)}$ denote the geodesic embedding isotopic to Γ that is in equilibrium with respect to the weight vector λ, as guaranteed by Theorem 2.1. (At the top level of recursion, we can safely assume that $\lambda(e) = 1$ for every edge e, but as we point out in Section 5.2, we cannot maintain that assumption in deeper recursive calls.) For each vertex v of Γ, let $\boldsymbol{p_*(v, \lambda)}$ denote the coordinates of v in $Eq(\Gamma, \lambda)$, obtained by solving linear system ($\star$). To define the embedding $Eq(\Gamma, \lambda)$ and its coordinates $p_*(v, \lambda)$ uniquely, fix $p_*(o, \lambda) = (0, 0)$ for some arbitrary vertex o.

5.1 Parallel Motion. The following lemma states intuitively that changing a single edge weight $\lambda(e)$ moves each vertex of the equilibrium embedding $Eq(\Gamma, \lambda)$ along a geodesic parallel to e.

LEMMA 5.1. *Let λ and λ' be arbitrary positive edge weights such that $\lambda(e) \neq \lambda'(e)$ for some edge e, and $\lambda(e') = \lambda'(e')$ for all edges $e' \neq e$. Let d_e be a dart of e with tail u and head v. Then for every vertex w, the vector $p_*(w, \lambda') - p_*(w, \lambda)$ is parallel to the embedding of edge e in $Eq(\Gamma, \lambda)$.*

Proof. Arbitrarily index the vertices of Γ from 1 to n. By the definition of crossing vectors, edge e appears in $Eq(\Gamma, \lambda)$ as the projection of an edge in the universal cover between $p_*(u, \lambda)$ and $p_*(v, \lambda) + x(d_e)$. Fix a non-zero vector $\sigma \in \mathbb{R}^2$ orthogonal to the vector $p_*(v, \lambda) - p_*(u, \lambda) + x(d_e)$ and thus orthogonal to edge e in $Eq(\Gamma, \lambda)$. For each vertex i, let $z_i = p_*(i, \lambda) \cdot \sigma$ and $z_i' = p_*(i, \lambda') \cdot \sigma$, and for each dart d, let $\chi(d) = x(d) \cdot \sigma$. Our choice of σ implies that $z_u - z_v = \chi(d_e)$. We need to prove that $z_i' = z_i$ for every vertex i.

The real vector $Z = (z_i)_i$ is a solution to the linear system $LZ = H$, where L is the $n \times n$ weighted Laplacian matrix

$$L_{ij} = \begin{cases} \sum_{tail(d)=i} -\lambda(d) & \text{if } i = j \\ \sum_{\substack{tail(d)=i \\ head(d)=j}} \lambda(d) & \text{if } i \neq j \end{cases}$$

and $H \in \mathbb{R}^n$ is a vector whose ith entry is

$$H_i = \sum_{tail(d)=i} -\lambda(d)\chi(d).$$

(In fact, Z is the *unique* solution such that $z_o = 0$.) Similarly, $Z' = (z_i')_i$ is the unique solution to an analogous equation $L'Z' = H'$ with $z_o' = 0$, where L' and H' are defined *mutatis mutandis* in terms of λ' instead of λ.

We prove that $Z' = Z$ as follows. Let $\delta = \lambda'(d_e) - \lambda(d_e)$. The Laplacian matrices L and L' differ in only four locations:

$$L'_{ij} - L_{ij} = \begin{cases} -\delta & \text{if } i = j = u \text{ or } i = j = v \\ \delta & \text{if } \{i, j\} = \{u, v\} \\ 0 & \text{otherwise} \end{cases}$$

More concisely, we have $L' = L - \delta(\mathbf{e}_v - \mathbf{e}_u)(\mathbf{e}_v - \mathbf{e}_u)^T$, where $\mathbf{e}_i$ denotes the ith standard coordinate vector. Similarly, we have $H' = H + \delta \cdot \chi(d_e) \cdot (\mathbf{e}_v - \mathbf{e}_u)$. It follows that

$$
\begin{aligned}
L'Z &= LZ - \delta(\mathbf{e}_v - \mathbf{e}_u)(\mathbf{e}_v - \mathbf{e}_u)^T Z \\
&= H - \delta(\mathbf{e}_v - \mathbf{e}_u)(z_v - z_u) \\
&= H + \delta(\mathbf{e}_v - \mathbf{e}_u)\chi(d_e) \\
&= H',
\end{aligned}
$$

which completes the proof. $\qquad\square$

We note in passing that a nearly identical lemma applies to internally 3-connected plane graphs; changing the weight of one edge e moves the vertices of any Tutte embedding along lines parallel to e. Surprisingly, this observation appears to be new.

5.2 Spring Collapses. Now fix a toroidal triangulation Γ and edge weights λ. Let e_{uv} be an arbitrary edge of Γ with endpoints u and v, and let Γ' be the result of collapsing u to v along e_{uv}. Let a, b, c, d be the edges in the star of uv in Γ, as shown in Figure 5.2. Edges a and b collapse to a single edge ab in Γ', and edges c and d collapse to a single edge cd in Γ'. Now define weights for the edges of Γ' as follows:

$$\lambda'(e) := \begin{cases} \lambda(a) + \lambda(b) & \text{if } e = ab \\ \lambda(c) + \lambda(d) & \text{if } e = cd \\ \lambda(e) & \text{otherwise} \end{cases}$$
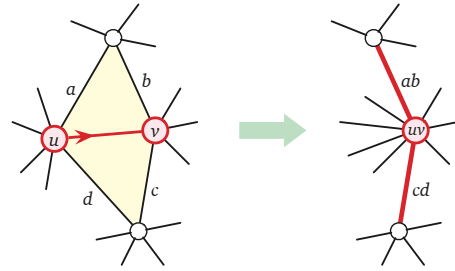
Figure 5.2. Collapsing u to v.

The new equilibrium embedding $Eq(\Gamma', \lambda')$ has the same image as the limit of $Eq(\Gamma, \lambda)$ as we increase $\lambda(e_{uv})$ to infinity and keep all other edge weights fixed. Lemma 5.1 implies that as we continuously increase $\lambda(e_{uv})$, the vertices of $Eq(\Gamma, \lambda)$ move continuously along geodesics parallel to e.

5.3 Geodesic Interpolation. This continuous deformation from $Eq(\Gamma, \lambda)$ to $Eq(\Gamma', \lambda')$ is not a parallel linear pseudomorph, because the vertices do not necessarily move at fixed speeds. To define a parallel linear pseudomorph, we simply move each vertex at constant speed along its corresponding geodesic. That is, for every vertex w and all real numbers $0 \le t \le 1$, let

$$p_t(w) = (1 - t) \cdot p_*(w, \lambda) + t \cdot p_*(w, \lambda'),$$

and let Γ_t denote the geodesic drawing of G with vertex coordinates $p_t(v)$ and the same crossing vectors as Γ. The following lemma implies that Γ_t is actually an embedding for all $0 < t < 1$. (Alamdari et al. used a similar observation [1, Corollary 7.2].) It follows that the continuous family of drawings Γ_t is a parallel linear pseudomorph from $\Gamma_0 = Eq(\Gamma, \lambda)$ to $\Gamma_1 = Eq(\Gamma', \lambda')$.

LEMMA 5.2. *Let p_0p_1, q_0q_1, and r_0r_1 be arbitrary parallel segments in the plane. For all real $0 \le t \le 1$, define $p_t = (1 - t)p_0 + tp_1$ and $q_t = (1 - t)q_0 + tq_1$ and $r_t = (1 - t)r_0 + tr_1$. If the triples p_0, q_0, r_0 and p_1, q_1, r_1 are oriented counterclockwise, then for all $0 \le t \le 1$, the triple p_t, q_t, r_t is also oriented counterclockwise.*

Proof. Without loss of generality, assume that segments $p_0 p_1$, $q_0 q_1$, and $r_0 r_1$ are horizontal. Thus, we can write $p_t = (px_t, py)$, and similarly for q_t and r_t. The triple p_t, q_t, r_t is oriented counterclockwise if and only if the following determinant is positive:

$$\Delta(t) := \begin{vmatrix} 1 & px_t & py \\ 1 & qx_t & qy \\ 1 & rx_t & ry \end{vmatrix}$$

Routine calculation implies

$$\Delta(t) = \begin{vmatrix} 1 & (1-t)px_0 + t\,px_1 & py \\ 1 & (1-t)qx_0 + t\,qx_1 & qy \\ 1 & (1-t)rx_0 + t\,rx_1 & ry \end{vmatrix}$$

$$= (1-t) \begin{vmatrix} 1 & px_0 & py \\ 1 & qx_0 & qy \\ 1 & rx_0 & ry \end{vmatrix} + t \begin{vmatrix} 1 & px_1 & py \\ 1 & qx_1 & qy \\ 1 & rx_1 & ry \end{vmatrix}$$

$$= (1-t) \cdot \Delta(0) + t \cdot \Delta(1)$$

Thus, the function $\Delta(t)$ has exactly one real root. It follows that if $\Delta(0) > 0$ and $\Delta(1) > 0$, then $\Delta(t) > 0$ for all $0 \le t \le 1$. $\square$

6 Zippers

Even if the original input triangulations Γ_0 and Γ_1 are simple, collapsing edges eventually reduces them to triangulations with parallel edges and loops. Every loop in a geodesic toroidal triangulation is a closed geodesic. The base case of our recursive algorithm is a special type of geodesic toroidal triangulation that we call a *zipper*, in which *every* vertex is incident to a loop. (This class of graphs were previously considered by Gonçalves and Lévêque [45, Fig. 44].)

Consider any zipper Z with n vertices, for some positive integer n. If $n = 1$, then Z consists of a single vertex, three loop edges, and two triangular faces. Otherwise, the loops in Z are disjoint closed geodesics, so they must be parallel. It follows that each vertex of Z is incident to exactly one loop. In either case, the n loops in Z decompose the torus into n annuli, each of which is decomposed into two triangles by two boundary-to-boundary edges. Figure 2.1 shows three two-vertex zippers, and Figure 6.1 shows a zipper with five vertices.
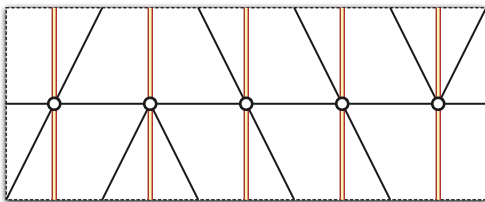

Figure 6.1. A five-vertex zipper. Doubled red edges are loops.

6.1 Structure.
The following results motivate our choice of zippers as a base case of our recursive algorithm.

LEMMA 6.1. *In every geodesic toroidal triangulation, every vertex incident to a loop has degree at least 6.*

Proof. Let Γ be a geodesic toroidal triangulation, let v be a vertex of Γ incident to a loop, let d_0 be either of the darts of that loop.

Let ∂f denote the clockwise facial walk around the face f to the right of d_0. Because the interior of f is an open disk, ∂f is contractible. Because Γ is a triangulation, ∂f consists of exactly three darts d_0, d_1, and d_2, where $head(d_i) = tail(d_{i+1 \bmod 3})$ for each index i. Because Γ is a *geodesic* triangulation, every face is incident to three *distinct* edges. It follows that $d_2 \ne rev(d_1)$.

Symmetrically, the counterclockwise walk around the face to the left of d_0 consists of three darts d_0, d_1', d_2', where $d_2' \ne rev(d_1')$. Thus, at least six distinct darts head into v: in counterclockwise cyclic order, d_0, $rev(d_1)$, d_2, $rev(d_0)$, d_2', $rev(d_1')$.[5] $\square$
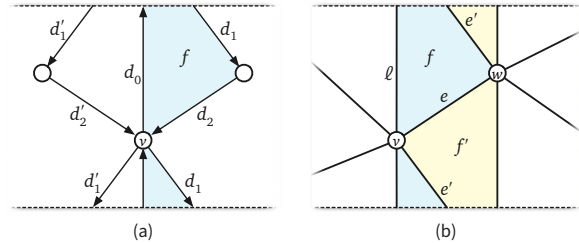

Figure 6.2. (a) Proof of Lemma 6.1. (b) Proof of Lemma 6.2

LEMMA 6.2. *Every 6-regular triangulation that contains a loop is a zipper.*

Proof. Again, let Γ be a 6-regular triangulation, and let v be a vertex of Γ incident to a loop ℓ. The previous proof implies that v is incident to two edges on either side of ℓ. Let e and e' be the edges incident to v on one side of ℓ; the edges ℓ, e, and e' enclose a triangular face f. Thus, e and e' share another common endpoint w. (Except in the trivial case where Γ has only one vertex, v and w are distinct.) Because e and e' are adjacent in cyclic order around v, there is another triangular face f' with e and e' on its boundary; the third edge of f' is a loop through w. The lemma now follows by induction. $\square$

COROLLARY 6.1. *Every triangulation that contains a loop but is not a zipper contains a vertex of degree at most 5 that is not incident to a loop.*

---

[5]If v is the only vertex of Γ, then $d_2' = d_1$ and $d_1' = d_2$, and thus v is incident to only three distinct *edges*, which are all loops. Otherwise, v is incident to at least five distinct edges, only one of which is a loop.
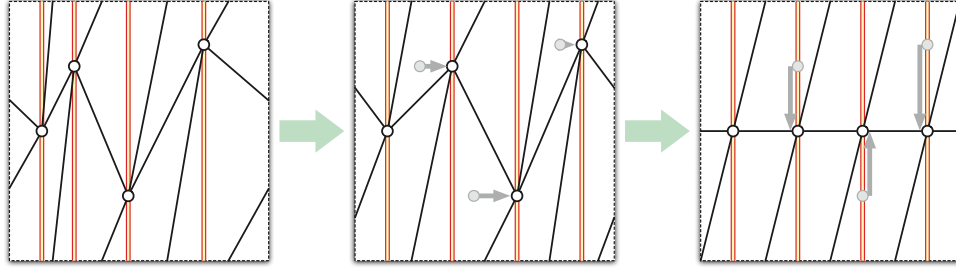
Figure 6.3. Morphing one zipper into another. Doubled red edges are loops.

6.2 Morphing Zippers. We next describe a straightforward approach to morphing between arbitrary isotopic zippers, which requires at most two parallel linear morphing steps.

Let Z and Z' be arbitrary isotopic zippers with n vertices. If $n = 1$, then Z and Z' differ only by translation, so assume otherwise. Ladegaillerie [57–59] proved that two embeddings of the same graph G on the same surface are isotopic if and only if the images of any cycle in G in both embeddings are homotopic. Two geodesic loops on the flat torus are homotopic if and only if they are parallel. Thus, Ladegaillerie's theorem implies that the loops in Z and Z' are all parallel to a common vector σ.

In the first parallel morphing step, we translate all vertices in Z along geodesics orthogonal to σ until each loop has the same image as the corresponding loop in Z'. Then in the second parallel morphing step, we translate all vertices along their respective loops to move all vertices and edges to their proper positions in Z'. See Figure 6.3 for an example. In both stages, Lemma 5.2 implies that linear interpolation between the old and new vertex coordinates yields an isotopy.

7 Converting Pseudomorphs to Morphs

In this section, we adapt a perturbation strategy of Alamdari et al. [1], which transforms their pseudomorphs between planar triangulations into morphs, to geodesic triangulations on the flat torus.

To explain our adaptation, we must first give a brief sketch of their algorithm. Let Γ_0 be the initial planar input triangulation. The input to their perturbation algorithm is a pseudomorph consisting of a (direct) collapse of a good vertex u to a neighbor v, a morph (not a pseudomorph) consisting of k parallel linear steps $\Gamma_0' \rightsquigarrow \Gamma_1' \rightsquigarrow \cdots \rightsquigarrow \Gamma_k'$, and finally the reverse of a collapse of u to v resulting in the final triangulation Γ_k. The output is a proper morph from Γ_0 to Γ_k, consisting of $k + 2$ parallel linear steps.

Alamdari et al. proceed as follows. Let P be the link of u in the initial triangulation Γ_0. For each index i, let v_i and P_i respectively denote the images of v and P in the intermediate triangulation Γ_i'. For each index i, a position u_i is found within the visibility kernel of P_i so that for all i, the vector $u_i \rightarrow u_{i+1}$ is parallel to $v_i \rightarrow v_{i+1}$ (the direction of the parallel linear morph $\Gamma_i' \rightsquigarrow \Gamma_{i+1}'$). For vertices of degree 3 and 4, it is simple to place u as a certain convex combination of the vertices in P. The strategy for vertices of degree 5 is more complicated. First, a value ε is computed such that for each index i, the intersection of the disk of radius ε centered at v_i and the visibility kernel of P_i consists of a full sector of the disk; call this intersection S_i. A specific position u_i is then chosen within each region S_i.

A close examination of their paper reveals that the strategy for vertices of degree 5 generalizes to vertices u of arbitrary degree (greater than 2). In particular, the definition of u_i depends solely on ε and the positions of the edges in P_i incident to v_i.

The radius ε is computed as follows. For each index i, we need a positive distance $\varepsilon_i > 0$ smaller than the minimum distance from v to any edge of the kernel of P_i that is not incident to v, at any time during the morphing step $\Gamma_i' \rightsquigarrow \Gamma_{i+1}'$. It suffices to compute the minimum distance from v to the lines supporting edges of P_i not incident to v. The squared distance to each of these lines at any time t can be expressed as the ratio $f(t)/g(t)$ of two quadratic polynomials f and g. Alamdari et al. argue that a lower bound $0 < \delta \le \min_t \sqrt{f(t)/g(t)}$ can be computed in constant time in an appropriate real RAM model [1]. Then ε_i is the minimum of these lower bounds δ. Altogether computing ε_i takes $O(\deg(u))$ time. Finally, $\varepsilon = \min_{1 \le i \le k} \varepsilon_i$.

Once the radius ε is known, computing each sector S_i in $O(\deg(u))$ time is straightforward.

The point u_0 can be chosen arbitrarily within S_0. For each index i in increasing order, Alamdari et al. describe how to choose a point $u_{i+1} \in S_{i+1}$ in $O(1)$ time, such that the vector $u_i \rightarrow u_{i+1}$ is parallel to the vector $v_i \rightarrow v_{i+1}$. This part of the algorithm makes no reference to the rest of the triangulation; it works entirely within the sectors S_i and S_{i+1}. Moreover, no part of this algorithm relies on u being *directly* collapsed to v, only that vertices u and v have the same image in the triangulations Γ_i' and Γ_{i+1}'.

We apply this perturbation technique to the toroidal pseudomorphs computed in Section 3.2 as follows. Recall that our pseudomorph consists of a direct collapse, a recursively computed pseudomorph, and a reversed spring collapse. First we (recursively) perturb the recursive pseudomorph into a proper morph $\Gamma'_0 \leadsto \cdots \leadsto \Gamma'_k$ consisting of k parallel linear morphs. We then compute the sectors S_i and the radius ε exactly as described above. To perturb the initial direct collapse from u to v, we move u to an arbitrary point in the intersection of S_0 and the edge e along which u is collapsed. We compute the intermediate positions u_i for u exactly as described above, working entirely within the local coordinates of the sectors S_i. Finally, to perturb the reversed spring collapse, we first move u from u_k to a new point u'_k in the visibility kernel so that the image of the collapsing edge e becomes parallel to the direction of the original spring collapse, after which we simply interpolate to the final triangulation, as described in Section 5.3. Because the vertices move along parallel geodesics, Lemma 5.2 implies that this final interpolation is a parallel linear morph. Altogether, we obtain a morph consisting of $k + 3$ parallel linear morphs. We emphasize that the additional step moving u_k to u'_k is the only significant difference from the algorithm presented by Alamdari et al.

Unrolling the recursion, we can perturb our pseudomorph between two n-vertex toroidal triangulations into a proper morph consisting of $O(n)$ parallel linear morphing steps in $O(n^2)$ time. The overall time to compute this morph is still dominated by the time needed to compute $O(n)$ equilibrium triangulations for the spring collapses.

THEOREM 7.1. *Given any two isotopic geodesic toroidal triangulations Γ_0 and Γ_1 with n vertices, we can compute a morph from Γ_0 to Γ_1 consisting of $O(n)$ parallel linear morphs in $O(n^{1+\omega/2})$ time.*

8 Not Just Triangulations

Finally, it remains to describe how to morph between embeddings that are not triangulations. Following existing work in the planar setting, we extend the given embeddings Γ_0 and Γ_1 to triangulations, and then invoke our earlier triangulation-morphing algorithm. The main difficulty is that it may not be possible to triangulate both Γ_0 and Γ_1 using the same diagonals, because corresponding faces, while combinatorially identical, may have different shapes.

Two different techniques have been proposed to overcome this hurdle in the planar setting. The first method subdivides each pair of corresponding faces into a compatible triangulation, introducing additional vertices if necessary [2, 47, 80–82]; however, this technique increases the complexity of the graph to $O(n^2)$ [8]. The second technique uses additional morphing steps to convexify faces to that

they can be compatibly triangulated without additional vertices [1, 7]. While the subdivision technique generalizes to toroidal embeddings (at least when all faces are disks), it is unclear how to generalize existing morphing techniques.

We introduce a third technique, which avoids both subdivision and additional morphing steps by exploiting Theorem 2.1. We emphasize that our method can also be applied to 3-connected straight-line plane graphs, giving a new and arguably simpler approach for the planar case as well.

THEOREM 8.1. *Given any two isotopic essentially 3-connected geodesic toroidal embeddings Γ_0 and Γ_1 with n vertices, we can compute a morph from Γ_0 to Γ_1 consisting of $O(n)$ parallel linear morphs in $O(n^{1+\omega/2})$ time.*

Proof. Let Γ_* be an equilibrium embedding isotopic to Γ_0 and Γ_1 as given by Theorem 2.1. It suffices to describe how to morph from Γ_0 to Γ_*; to morph from Γ_0 to Γ_1 one can simply first morph from Γ_0 to Γ_* and then from Γ_* to Γ_1.

Arbitrarily triangulate the faces of Γ_0; this can be done in $O(n)$ time using Chazelle's algorithm [26], or in $O(n \log n)$ time in practice. Because each face of Γ_* is convex, we can triangulate Γ_* in the exact same manner. The result is two isotopic geodesic toroidal triangulations T_0 and T_*. Given a morph between T_0 and T_* as promised by Theorem 7.1, we obtain a morph between Γ_0 and Γ_* by simply ignoring the edges added when triangulating. In particular, the morph is specified by a sequence of geodesic triangulations $T_0, T_1, \ldots, T_k = T_*$, and dropping the additional edges from each triangulation T_i results in a geodesic embedding Γ_i isotopic to Γ_0.

The number of parallel morphing steps remains $O(n)$, and the running time is dominated by the computation of the morph between T_0 and T_*, which is $O(n^{1+\omega/2})$ by Theorem 7.1. □

Finally, Theorem 8.1 immediately yields the first proof of the following corollary:

COROLLARY 8.1. *Two essentially 3-connected geodesic embeddings on the flat torus are isotopic if and only if they are isotopic through geodesic embeddings.*

9 Conclusions and Open Questions

We have described the first algorithm to construct a morph between two isotopic geodesic graphs on the flat torus. Key tools in our algorithm include a geometric analysis of 6-regular triangulations on the torus, as well as repeated use of a generalization of Tutte's spring embedding theorem by Y. Colin de Verdière [28] (Theorem 2.1). Several of our applications of spring embeddings also apply to

planar morphs and therefore provide a new approach to compute linear-complexity morphs between straight-line plane graphs.

Because it relies heavily on Theorem 2.1, our algorithm requires that the input embeddings are essentially 3-connected. If a given toroidal embedding Γ is *not* essentially 3-connected, an isotopic equilibrium drawing Γ_* still exists, but it may not be an embedding; nontrivial subgraphs can collapse to geodesics or even points in Γ_*. Morphing less connected toroidal embeddings remains an open problem.

It is natural to ask whether any of our results can be extended to higher-genus surfaces. Y. Colin de Verdière actually generalized Tutte's theorem to graphs on arbitrary Riemannian 2-manifolds without positive curvature [28]. However, the resulting equilibrium embedding is the solution a certain convex optimization problem that, in general, cannot be formulated as solving a linear system. At a more basic level, our analysis of cats and dogs in Section 4 relies on the average vertex degree being exactly 6, a property that holds *only* for graphs on the torus or the Klein bottle. Even the existence of "parallel" morphs requires an orientable locally Euclidean metric.

Our results share two closely related limitations with existing planar morphing algorithms. First, planar morphs involving either Cairns-style edge collapses or spring embeddings require high numerical precision to represent exactly. Second, while edge collapses yield planar morphs with low combinatorial complexity, the resulting morphs are not good for practical visualization applications [63]. Because our algorithm uses both edge collapses and spring embeddings, it suffers from the same numerical precision issues and (we expect) the same practical limitations. Floater, Gotsman, and Surazhsky's barycentric interpolation technique [42, 47, 80, 81] yields better results in practice for planar morphs, but as we discussed in Section 1.1, their technique does not immediately generalize to the torus.

Finally, there are several variants and special cases of planar morphing for which generalization to the flat torus would be interesting, including morphing with bent edges [63], morphing orthogonal embeddings [12, 15, 43, 44], and morphing weighted Schnyder embeddings [11].

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. arXiv:1606.00425.

[2] Soroush Alamdari, Patrizio Angelini, Timothy M. Chan, Giuseppe Di Battista, Fabrizio Frati, Anna Lubiw, Maurizio Patrignani, Vincenzo Roselli, Sahil Singla, and BryanT. Wilkinson. Morphing planar graph drawings with a polynomial number of steps. *Proc. 24th Ann. ACM-SIAM Symp. Discrete Algorithms*, 1656–1667, 2013.

[3] Lyudmil Aleksandrov and Hristo Djidjev. Linear algorithms for partitioning embedded graphs of bounded genus. *SIAM J. Discrete Math.* 9(1):129–150, 1996.

[4] 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.

[5] Amos Altshuler. Construction and enumeration of regular maps on the torus. *Discrete Math.* 4:201–217, 1973.

[6] Patrizio Angelini, Fabrizio Frati, Maurizio Patrignani, and Vincenzo Roselli. Morphing planar graph drawings efficiently. *Proc. 21st Int. Symp. Graph Drawing*, 49–60, 2013. Lecture Notes Comput. Sci. 8242, Springer.

[7] Patrizio Angelini, Giordano Da Lozzo, Giuseppe Di Battista, Fabrizio Frati, Maurizio Patrignani, and Vincenzo Roselli. Morphing planar graph drawings optimally. *Proc. 41st Int. Colloq. Automata Lang. Prog.*, 126–137, 2014. Lecture Notes Comput. Sci. 8572, Springer.

[8] Boris Aronov, Raimund Seidel, and Diane Souvaine. On compatible triangulations of simple polygons. *Comput. Geom. Theory Appl.* 3(1):27–35, 1993.

[9] Fidel Barrera-Cruz. *Morphing Planar Triangulations*. Ph.D. thesis, Univ. Waterloo, 2014. ⟨http://hdl.handle.net/10012/8518⟩.

[10] Fidel Barrera-Cruz, Penny Haxell, and Anna Lubiw. Morphing planar graph drawings with unidirectional moves. Preprint, November 2014.

[11] Fidel Barrera-Cruz, Penny Haxell, and Anna Lubiw. Morphing Schnyder drawings of planar triangulations. *Discrete Comput. Geom.* 61(1):161–184, 2019.

[12] Therese Biedl, Anna Lubiw, and Michael J. Spriggs. Morphing planar graphs while preserving edge directions. *Proc. 13th Int. Symp. Graph Drawing*, 13–24, 2005. Lecture Notes Comput. Sci. 3843, Springer.

[13] Therese Biedl, Anna Lubiw, and Michael J. Spriggs. Morphing planar graphs while preserving edge directions. *Proc. 14th Int. Symp. Graph Drawing*, 13–24, 2006. Lecture Notes Comput. Sci. 4372, Springer.

[14] Therese Biedl, Anna Lubiw, and Michael J. Spriggs. Morphing polyhedra with parallel faces: Counterexamples. *Comput. Geom. Theory Appl.* 42(5):395–402, 2009. ⟨http://www.sciencedirect.com/science/article/pii/S0925772108001120⟩.

[15] Therese C. Biedl, Anna Lubiw, Mark Petrick, and Michael J. Spriggs. Morphing orthogonal planar graph drawings. *ACM Trans. Algorithms* 9(4):29:1–29:24, 2013.

[16] Ciprian Borcea and Ileana Streinu. Periodic frameworks and flexibility. *Proc. Royal Soc. A* 466(2121):2633–2649, 2010.

[17] Ciprian Borcea and Ileana Streinu. Liftings and stresses for planar periodic frameworks. *Discrete Comput. Geom.* 53(4):747–782, 2015.

[18] Ulrich Brehm and Wolfgang Kühnel. Equivelar maps on the torus. *Europ. J. Combin.* 29(8):1843–1861, 2008.

[19] Oleksiy Busaryev, Sergio Cabello, Chao Chen, Tamal K. Dey, and Yusu Wang. Annotating simplices with a homology basis and its applications. *Proc. 13th Scand. Workshop Algorithm Theory*, 189–200, 2012. Lecture Notes Comput. Sci. 7357, Springer.

[20] Steward S. Cairns. Deformations of plane rectilinear complexes. *Amer. Math. Monthly* 51(5):247–252, 1944.

[21] Steward S. Cairns. Isotopic deformations of geodesic complexes on the 2-sphere and on the plane. *Ann. Math.* 45(2):207–217, 1944.

[22] Luca Castelli Aleardi, Éric Fusy, and Thomas Lewiner. Schnyder woods for higher genus triangulated surfaces, with applications to encoding. *Discrete Comput. Geom.* 42(3):489–516, 2009.

[23] Erin W. Chambers, Éric Colin de Verdière, Jeff Erickson, Francis Lazarus, and Kim Whittlesey. Splitting (complicated) surfaces is hard. *Comput. Geom. Theory Appl.* 41(1–2):94–110, 2008.

[24] Erin W. Chambers, Jeff Erickson, and Amir Nayyeri. Minimum cuts and shortest homologous cycles. *Proc. 25th Ann. Symp. Comput. Geom.*, 377–385, 2009.

[25] Erin W. Chambers, Kyle Fox, and Amir Nayyeri. Counting and sampling minimum cuts in genus *g* graphs. *Discrete Comput. Geom.* 52(3):450–475, 2014.

[26] Bernard Chazelle. Triangulating a simple polygon in linear time. *Discrete Comput. Geom.* 6(5):485–524, 1991.

[27] Edith Cohen and Nimrod Megiddo. Strongly polynomial-time and NC algorithms for detecting cycles in periodic graphs. *J. Assoc. Comput. Mach.* 40(4):791–830, 1993.

[28] Yves Colin de Verdière. Comment rendre géodésique une triangulation d'une surface? *L'Enseignment Mathématique* 37:201–212, 1991.

[29] Éric Colin de Verdière and Arnaud de Mesmay. Testing graph isotopy on surfaces. *Discrete Comput. Geom.* 51(1):171–206, 2014.

[30] 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.

[31] Lothar Collatz. Spektren periodischer Graphen. *Resultate Math.* 1(1–2):42–53, 1978.

[32] Olaf Delgado-Friedrichs. Equilibrium placement of periodic graphs and convexity of plane tilings. *Discrete Comput. Geom.* 33(1):67–81, 2004.

[33] Sean Dewar. Flexible placements of periodic graphs in the plane. Preprint, November 2019. arXiv:1911.05634.

[34] Tamal K. Dey, Tianqi Li, and Yusu Wang. Efficient algorithms for computing a minimal homology basis. *Proc. 13th Latin American Symp. Theoretical Informatics*, 376–398, 2018. Lecture Notes Comput. Sci. 10807, Springer.

[35] Tamal K. Dey, Jian Sun, and Yusu Wang. Approximating loops in a shortest homology basis from point data. *Proc. 26th Ann. Symp. Comput. Geom.*, 166–175, 2010.

[36] Jack Edmonds and Richard M. Karp. Theoretical improvements in algorithmic efficiency of network flow problems. *J. Assoc. Comput. Mach.* 19(2):248–264, 1972.

[37] Jeff Erickson, Kyle Fox, and Luvsandondov Lkhamsuren. Holiest minimum-cost paths and flows in surface graphs. *Proc. 50th Ann. ACM Symp. Theory Comput.*, 1319–1332, 2018.

[38] Jeff Erickson and Patrick Lin. A toroidal Maxwell-Cremona-Delaunay correspondence. *Proc. 36rd Int. Symp. Comput. Geom.*, 40:1–40:17, 2020. Leibniz Int. Proc. Informatics, Schloss Dagstuhl–Leibniz-Zentrum für Informatik.

[39] Jeff Erickson and Amir Nayyeri. Minimum cuts and shortest non-separating cycles via homology covers. *Proc. 22nd Ann. ACM-SIAM Symp. Discrete Algorithms*, 1166–1176, 2011.

[40] Jeff Erickson and Kim Whittlesey. Greedy optimal homotopy and homology generators. *Proc. 16th Ann. ACM-SIAM Symp. Discrete Algorithms*, 1038–1046, 2005.

[41] Stefan Felsner. Lattice structures from planar graphs. *Elec. J. Combin.* 11(1):#R15, 2004.

[42] Michael S. Floater and Craig Gotsman. How to morph tilings injectively. *J. Comput. Appl. Math.* 101(1–2):117–129, 1999.

[43] Arthur van Goethem, Bettina Speckmann, and Kevin Verbeek. Optimal morphs of planar orthogonal drawings II. *Proc. 27th Int. Symp. Graph Drawing Network Vis.*, 33–45, 2019. Lecture Notes Comput. Sci. 1904, Springer.

[44] Arthur van Goethem and Kevin Verbeek. Optimal morphs of planar orthogonal drawings. *Proc. 34th Int. Symp. Comput. Geom.*, 42:1–42:14, 2018. Leibniz Int. Proc. Informatics 99, Schloss Dagstuhl–Leibniz-Zentrum für Informatik.

[45] Daniel Gonçalves and Benjamin Lévêque. Toroidal maps: Schnyder woods, orthogonal surfaces and straight-line representations. *Discrete Comput. Geom.* 51(1):67–131, 2014.

[46] 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.

[47] Craig Gotsman and Vitaly Surazhsky. Guaranteed intersection-free polygon morphing. *Computers & Graphics* 25(1):67–75, 2001.

[48] Kazuo Iwano. *Two-Dimensional Dynamic Graphs and their VLSI Applications*. Ph.D. thesis, Princeton Univ., October 1987. ⟨https://search.proquest.com/docview/303485362⟩.

[49] Kazuo Iwano and Kenneth Steiglitz. Testing for cycles in infinite graphs with periodic structure. *Proc. 19th Ann. ACM Symp. Theory Comput.*, 46–55, 1987.

[50] Kazuo Iwano and Kenneth Steiglitz. A semiring on convex polygons and zero-sum cycle problems. *SIAM J. Comput.* 19(5):883–901, 1990.

[51] Donald B. Johnson. Efficient algorithms for shortest paths in sparse networks. *J. Assoc. Comput. Mach.* 24(1):1–13, 1977.

[52] Richard M. Karp, Raymond E. Miller, and Shmuel Winograd. The organization of computations for uniform recurrence equations. *J. Assoc. Comput. Mach.* 14(3):563–590, 1967.

[53] Viktória E. Kaszanitzky, Bernd Schulze, and Shin-ichi Tanigawa. Global rigidity of periodic graphs under fixed-lattice representations. *J. Comb. Theory Ser. B* 146:176–218, 2021.

[54] 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.

[55] William Kocay, Daniel Neilson, and Ryan Szypowski. Drawing graphs on the torus. *Ars Comb.* 59:259–277, 2001. ⟨http://www.combinatorialmath.ca/G&G/articles/Torus.pdf⟩.

[56] S. Rao Kosaraju and Gregory F. Sullivan. Detecting cycles in dynamic graphs in polynomial time (preliminary version). *Proc. 20th ACM Symp. Theory. Comput.*, 398–406, 1988.

[57] Yves Ladegaillerie. Classification topologique des plongements des 1-complexes compacts dans les surfaces. *C. R. Acad. Sci. Paris A* 278:1401–1403, 1974. ⟨https://gallica.bnf.fr/ark:/12148/bpt6k6236784g/f179⟩.

[58] Yves Ladegaillerie. Classification topologique des plongements des 1-complexes compacts dans les surfaces. *C. R. Acad. Sci. Paris A* 279:129–132, 1974. ⟨https://gallica.bnf.fr/ark:/12148/bpt6k6238171d/f143⟩.

[59] Yves Ladegaillerie. Classes d'isotopie de plongements de 1-complexes dans les surfaces. *Topology* 23(3):303–311, 1984.

[60] François Le Gall. Powers of tensors and fast matrix multiplication. *Proc. 25th int. Symb. Alg. Comput.*, 296–303, 2014.

[61] Richard J. Lipton, Donald J. Rose, and Robert Endre Tarjan. Generalized nested dissection. *SIAM J. Numer. Anal.* 16:346–358, 1979.

[62] László Lovász. Discrete analytic functions: An exposition. *Eigenvalues of Laplacians and other geometric operators*, 241–273, 2004. Surveys in Differential Geometry 9, Int. Press.

[63] Anna Lubiw and Mark Petrick. Morphing planar graph drawings with bent edges. *J. Graph Algorithms Appl.* 15(2):205–227, 2011.

[64] Justin Malestein and Louis Theran. Generic combinatorial rigidity of periodic frameworks. *Advances in Mathematics* 233(1):291–331, 2013. arXiv:1008.1837.

[65] Peter Mani-Levitska, Bruno Guigas, and Wilfrid E. Klee. Rectifiable n-periodic maps. *Geom. Dedicata* 8:127–137, 1979.

[66] Bojan Mohar. Straight-line representations of maps on the torus and other flat surfaces. *Discrete Math.* 155(1–3):173–181, 1996.

[67] Bojan Mohar. Circle packings of maps—The Euclidean case. *Rend. Sem. Mat. Fis. Milano* 67(1):191–206, 1997.

[68] Bojan Mohar. Circle packings of maps in polynomial time. *Europ. J. Combin.* 18(7):785–805, 1997.

[69] Bojan Mohar and Pierre Rosenstiehl. Tessellation and visibility representations of maps on the torus. *Discrete Comput. Geom.* 19(2):249–263, 1998.

[70] Bojan Mohar and Alexander Schrijver. Blocking nonorientability of a surface. *J. Comb. Theory Ser. B* 87(1):2–16, 2003.

[71] Seiya Negami. Uniqueness and faithfulness of embedding of toroidal graphs. *Discrete Math.* 44(2):161–180, 1983.

[72] Anthony Nixon and Elissa Ross. Periodic rigidity on a variable torus using inductive constructions. *Elec. J. Combin.* 22(1):P1.1, 2015. arXiv:1204.1349.

[73] James B. Orlin. Some problems on dynamic/periodic graphs. *Progress in Combinatorial Optimization*, 273–294, 1984. Academic Press. ⟨https://dspace.mit.edu/handle/1721.1/2033⟩.

[74] Sailesh K. Rao and Thomas Kailath. Regular iterative algorithms and their implementation on processor arrays. *Proc. IEEE* 76(3):259–269, 1988.

[75] Bruce L. Reinhart. Line elements on the torus. *Amer. J. Math.* 81(3):617–631, 1959.

[76] Vincenzo Roselli. *Morphing and visiting drawings of graphs*. Ph.D. thesis, Univ. Roma Tre, June 2014. ⟨http://hdl.handle.net/2307/4428⟩.

[77] Elissa Ross. The rigidity of periodic frameworks as graphs on a fixed torus. *Contrib. Discrete Math.* 9(1):11–45, 2014. arXiv:1202.6652.

[78] Elissa Ross. Inductive constructions for frameworks on a two-dimensional fixed torus. *Discrete Comput. Geom.* 54(1):78–109, 2015. arXiv:1203.6561.

[79] Dvir Steiner and Anath Fischer. Planar parameterization for closed 2-manifold genus-1 meshes. *Proc. 9th ACM Symp. Solid Modeling Appl.*, 83–91, 2004.

[80] Vitaly Surazhsky and Craig Gotsman. Controllable morphing of compatible planar triangulations. *ACM Trans. Graphics* 20(4):203–231, 2001.

[81] Vitaly Surazhsky and Craig Gotsman. Morphing stick figures using optimal compatible triangulations. *Proc. 9th Pacific Conf. Comput. Graphics Appl.*, 40–49, 2001.

[82] Carsten Thomassen. Deformations of plane graphs. *J. Comb. Theory Ser. B* 34(3):244–257, 1983.

[83] Nobuaki Tomizawa. On some techniques useful for solution of transportation network problems. *Networks* 1:173–194, 1971.

[84] William T. Tutte. How to draw a graph. *Proc. London Math. Soc.* 13(3):743–768, 1963.

[85] William M. Waite. Path detection in multidimensional iterative arrays. *J. Assoc. Comput. Mach.* 14(2):300–310, 1967.

A   Relaxed Coordinate Representations

The coordinate representation described in Section 2.3, while intuitive, is much more constrained than necessary. Here we describe a more relaxed representation that allows morphs to be described entirely in terms of changing vertex coordinates, while still encoding the non-trivial topology of the embedding. Similar representations have been traditionally used to model periodic (or "dynamic") graphs [27, 31, 48–50, 52, 56, 73, 74, 85], and more recently to model periodic bar-and-joint frameworks [16, 17, 33, 53, 64, 72, 77, 78].

We also observe that this representation leads naturally to a linear-time algorithm to test whether two given toroidal embeddings are isotopic. Our algorithm is arguably simpler than the more general linear-time isotopy algorithm of É. Colin de Verdière and de Mesmay [29]. We emphasize that both of these isotopy algorithms are non-constructive; they do not construct an isotopy if one exists. Rather, both algorithms test topological conditions that characterize isotopy [57–59].

A.1 Translation Vectors and Equivalence. To represent a geodesic embedding Γ of a graph G on the flat torus $\mathbb{T}$, we associate a ***coordinate vector*** $p(v) \in \mathbb{R}^2$ with every vertex v of G and a ***translation vector*** $\tau(d) \in \mathbb{Z}^2$ with every dart d of G. We do *not* require vertex coordinates to lie in the unit square; instead, each coordinate vector $p(v)$ records the coordinates of an arbitrary lift $\widetilde{v}$ of v to the universal cover $\widetilde{\Gamma}$. The translation vector of each dart encodes which lifts of its endpoints are connected in $\widetilde{\Gamma}$. Specifically, for each dart d in G, the universal cover $\widetilde{\Gamma}$ contains an edge between $p(tail(d))$ and $p(head(d)) + \tau(d)$, and therefore also contains and edge between $p(tail(d)) + (i, j)$ and $p(head(d)) + \tau(d) + (i, j)$ for all integers i and j. Translation vectors are antisymmetric: $\tau(d) = -\tau(rev(d))$.

Coordinate representations are not unique; in fact, each toroidal embedding has an infinite family of equivalent representations. Two coordinate representations (p, τ) and (p', τ') with the same underlying graph are *equivalent*, meaning they represent the same geodesic embedding (up to translation), if and only if

$$\Delta(d) := p(head(d)) + \tau(d) - p(tail(d))$$
$$= p'(head(d)) + \tau'(d) - p'(tail(d))$$

for every dart d. The vector $\Delta(d)$, which we call the ***displacement vector*** of d, is the difference between the head and tail of any lift of d to $\widetilde{\Gamma}$.

Let (p, τ) be any coordinate representation of Γ. Given *arbitrary* integer vector $\pi(v) \in \mathbb{Z}^2$ for each vertex of G, we can define a new coordinate representation (p^π, τ^π) as follows, for every vertex v and dart d:

$$p^\pi(v) = p(v) + \pi(v)$$
$$\tau^\pi(d) = \tau(d) + \pi(tail(d)) - \pi(head(d))$$

Easy calculation implies that the representations (p, τ) and (p^π, τ^π) are equivalent. This transformation is a multidimensional generalization of the *reweighting* or *repricing* strategy proposed by Tomizawa [83] and Edmonds and Karp [36] for minimum-cost flows, and later used by Johnson to compute shortest paths [51].

Every geodesic toroidal embedding has a unique ***canonical*** coordinate representation (p, τ), where $p(v) \in$ $[0, 1)^2$ for every vertex v. In this canonical coordinate representation, each translation vector $\tau(d)$ encodes how dart d crosses the boundaries of the fundamental square; in other words, canonical translation vectors are crossing vectors, exactly as described in Section 2.3.

A.2 Normalization and Isotopy Testing. Let Γ_0 and Γ_1 be two isotopic geodesic toroidal embeddings of the same graph G, given by coordinate representations (p_0, τ_0) and (p_1, τ_1) respectively. To simplify the presentation of our morphing algorithm, we implicitly assume that the translation vectors in both representations are identical: $\tau_0(d) = \tau_1(d)$ for every dart d. This assumption allows us to describe, reason about, and ultimately compute a morph from Γ_0 to Γ_1 entirely in terms of changes to the vertex coordinates; all translation vectors remain fixed throughout the morph.

If necessary, we can enforce this assumption in $O(n)$ time using the following *normalization* algorithm. Let (p_0, τ_0) and (p_1, τ_1) be the given coordinate representations of Γ_0 and Γ_1, respectively. First, construct an arbitrary spanning tree T of the underlying graph G, directed away from an arbitrary root vertex r. For every vertex v, let $P(v)$ denote the unique directed path in T from r to v. For each vertex v, let

$$\pi(v) = \sum_{d \in P(v)} (\tau_1(d) - \tau_0(d)).$$

We can compute the vectors $\pi(v)$ for all vertices in $O(n)$ time by preorder traversal of T. Finally, we replace the target representation (p_1, τ_1) with the equivalent representation (p_1^π, τ_1^π).

LEMMA A.1. *For all darts d in T, we have $\tau_1^\pi(d) = \tau_0(d)$.*

Proof. Let d be any dart in T directed from some vertex u to one of its children v in T. Straightforward calculation implies

$$\pi(u) - \pi(v) = \sum_{d' \in P(u)} (\tau_1(d') - \tau_0(d'))$$
$$- \sum_{d' \in P(v)} (\tau_1(d') - \tau_0(d'))$$
$$= \tau_0(d) - \tau_1(d)$$

and therefore $\tau_1^\pi(d) = \tau_1(d) + \pi(u) - \pi(v) = \tau_0(d)$. A similar calculation (or antisymmetry) implies that $\tau_1^\pi(d) = \tau_0(d)$ for every dart d directed from a vertex to its parent in T. □

THEOREM A.1. *Γ_0 and Γ_1 are isotopic if and only if $\tau_1^\pi(d) = \tau_0(d)$ for every dart d.*

Proof. We exploit a theorem of Ladegaillerie [57–59], which states that two embeddings are isotopic if and only if every cycle in one embedding is homotopic to the corresponding cycle in the other embedding.

For any dart d in the underlying graph G, let $\Delta_0(d)$ and $\Delta_1(d)$ denote the displacement vectors of d in Γ_0 and Γ_1, respectively. (We emphasize that displacement vectors are independent of the coordinate representation.) For any directed cycle C in G, let $\Delta_0(C)$ and $\Delta_1(C)$ denote the sum of the displacement vectors of its darts:

$$\Delta_0(C) := \sum_{d \in C} \Delta_0(d) \qquad \Delta_1(C) := \sum_{d \in C} \Delta_1(d)$$

The vector $\Delta_0(C)$ is the *integer homology class* of C in Γ_0. Two cycles on the torus are homotopic if and only if they have the same integer homology class; in particular, the image of C in Γ_0 is contractible if and only if $\Delta_0(C) = (0, 0)$. Ladegaillerie's theorem implies that Γ_0 and Γ_1 are isotopic if and only if $\Delta_0(C) = \Delta_1(C)$ for every cycle C.

The spanning tree T defines a set of *fundamental cycles* that span the cycle space of G. Specifically, for each dart d that is not in T, the fundamental directed cycle $C_T(d)$ consists of d and the unique directed path in T from $head(d)$ to $tail(d)$. Every directed cycle in G (indeed every *circulation* in G) can be expressed as a linear combination of fundamental cycles. It follows by linearity that Γ_0 and Γ_1 are isotopic if and only if every *fundamental* cycle has the same integer homology class in both embeddings, or in other words, if and only if $\Delta_0(C_T(d)) = \Delta_1(C_T(d))$ for every dart $d \in G \setminus T$.

Straightforward calculation implies that the homology class of any cycle is also equal to the sum of the *translation* vectors of its darts with respect to *any* coordinate representation:

$$\Delta_0(C) = \sum_{d \in C} \tau_0(d)$$
$$\Delta_1(C) = \sum_{d \in C} \tau_1(d) = \sum_{d \in C} \tau_1^\pi(d).$$

In particular, for any non-tree dart $d \notin T$, we immediately have

$$\Delta_0(C_T(d)) - \Delta_1(C_T(d)) = \sum_{d' \in C_T(d)} \left(\tau_0(d') - \tau_1^\pi(d') \right)$$
$$= \tau_0(d) - \tau_1^\pi(d)$$

Thus, $\tau_1^\pi(d) = \tau_0(d)$ for every dart d if and only if $\Delta_0(C) = \Delta_1(C)$ for every fundamental cycle C, which completes the proof of the theorem. □

Theorem A.1 and our normalization algorithm immediately imply an $O(n)$-time algorithm to test whether two

given coordinate representations (p_0, τ_0) and (p_1, τ_1) represent isotopic toroidal embeddings of the same graph G. Our algorithm is arguably simpler than the isotopy algorithm of É. Colin de Verdière and de Mesmay [29], which is also based on Ladegaillerie's theorem [57–59]. On the other hand, our isotopy algorithm is specific to geodesic embeddings on the flat torus; whereas, É. Colin de Verdière and de Mesmay's algorithm works for arbitrary combinatorial embeddings of graphs on arbitrary 2-manifolds.