Compatible Geometries by Convex Representations with Applications to Computer Graphics

Ilya Eckstein Vitaly Surazhsky Craig Gotsman

Computer Science Department
Technion – Israel Institute of Technology

Introduction

Finding compatible triangulations for two corresponding sets of n planar points is a well-studied problem in computational geometry. The decision problem (whether such a triangulation exists) is believed to be NP-Complete, and to find the triangulation is believed to be NP-Hard [5]. The only algorithms which come close to solving this problem on triangulations and polygons require the addition of $\theta(n^2)$ Steiner points [1, 2, 6] to guarantee a solution. The dual problem to this is that of *compatible geometries*: Given two n-vertex topologies, embed these vertices in the plane, such that both the straight-line graphs induced by the two topologies are planar. See Fig. 1 for an example. This problem is just one of a class of many in which the topology is given and fixed, and the objective is to find a geometry satisfying a variety of conditions.

The compatible geometry problem has applications in texture-mapping for 3D polygon graphics. Assume the given topologies are those of two 3D meshes with the same number of vertices – n. The (u,v) coordinates of the planar embeddings are texture coordinates in 2D image space which serve both models simultaneously. The problem becomes more interesting in a multiresolution (or polygon reduction) setting, where a textured 3D triangle mesh is incrementally simplified by vertex removal. Each vertex is removed along with its associated properties, including the texture coordinates, and the resulting "hole" in the mesh is retriangulated. We would like the entire sequence of simplified meshes to still be textured in a consistent manner, using the texture coordinates inherited from the more detailed meshes. In general the simplification history, including the topologies, is given, and cannot be modified. Here a generalization of the compatible geometry problem arises: Given a corresponding sequence of graphs $\{T_i: i=1,..,n\}$ which describe a simplification process - find a common embedding of the vertices in the plane such that all the topologies of the graphs induce planar straight-line graphs. It is easy to find examples where the mesh topologies are such that the 3D mesh is valid (no self-intersections), but there is no solution to the planar embedding problem.

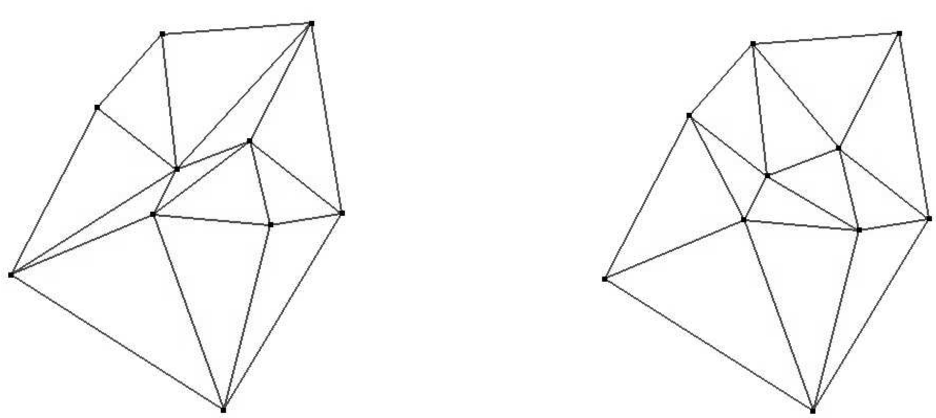

Figure 1: Two different topologies with compatible geometries.

Convex Representations

Simple convex representations were first used by Tutte [8] to draw straight-line planar graphs of given topologies with fixed convex boundaries by solving a linear set of equations, or, alternatively, iteratively computing their geometries. Using a straight-line representation involves expressing the coordinates of each vertex as a convex combination of the coordinates of its neighbors. This yields a sparse matrix with some redundancy, whose rows sum to unity. The matrix induces a geometry on the vertices – the eigenvector corresponding to the unit eigenvalue, and it may be shown that this geometry always guarantees a planar straight-line graph. Hence working in the space of these matrices has the advantage of preserving planarity (or non-self-intersections). The simple convex combination approach of Tutte was extended by Floater [3] to generate planar parametrizations of 3D meshes with more natural geometric properties. Floater and Gotsman [4] used these representations to generate morphs between compatible planar triangulations which preserve the compatibility throughout the morph. This, in turn, was improved by Surazhsky and Gotsman [7] to provide more precise local control over the morph.

In this paper, we show how the convex combination approach may be used to solve the compatible geometry problem. Our algorithms use the iterative approach advocated by [7], with local modifications to the convex representation matrices. This yields an iterative algorithm which is shown to find a solution if it exists.

As a side-benefit, variants of this basic algorithm may be used to generate a planar straight-line graph for a topology in which a subset of the vertices have given geometries, and the rest of the vertices are either totally or partially unconstrained. This is a generalization of the basic problem solved by Tutte [8]. The algorithm may then be used to generate non-self-intersecting morphs of planar triangulations when the trajectories of a subset of the points are given.

References

[1] B. Aronov, R. Seidel and D. Souvaine. On compatible triangulations of simple polygons. Computational Geometry: Theory and Applications, 3:27—35, 1993.

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

[3] M. S. Floater. Parameterization and smooth approximation of surface triangulation. Computer Aided Geometric Design,14:231—250,1997.

[4] M. S. Floater and C. Gotsman. How to morph tilings injectively. Journal of Computational and Applied Mathematics, 1999.

[5] A. Saalfeld. Joint triangulations and triangulation maps. Proceedings of 3rd Annual ACM Symposium on Computational Geometry, pp. 195-204, 1987.

[6] L. Souvaine and R. Wenger. Constructing piecewise linear homeomorphisms, DIMACS Technical Report 94-52,1994.

[7] V. Surazhsky and C. Gotsman. Morphing planar triangulations. Preprint, 1999.

[8] W. T. Tutte. How to draw a graph. Proc. London Math. Soc., 13:743-768, 1963.