Obtaining High-Quality Untangled Meshes Through Force-Directed Graph Embedding

Sanjukta Bhowmick, Suzanne Shontz
The Pennsylvania State University, University Park, PA
{bhowmick, shontz}@cse.psu.edu

Numerical solutions of partial differential equations require discretization of the governing equations. This discretization is achieved by decomposing the underlying domain into a tesselation or mesh of simple geometric shapes (such as triangles, quadrilaterals, tetrahedra, or hexahedra). Unstructured meshes have an irregular tesselation pattern and are used in physical simulations on domains involving complex geometries.

The accuracy of the numerical PDE solutions depends upon the mesh quality which can be measured according to various mesh quality metrics involving the shape, size, and orientation of mesh elements. Tangled meshes contain inverted elements, i.e., elements with non-positive volume (or negative orientation). Such meshes can result from node movement in large deformation problems, e.g, traumatic brain injuries, heart arrhthymias, foam-rubber rollers, and metal forming; or in fluid flow problems. Alternatively, they may be the result of the mesh generation process itself. Tangled meshes are invalid, and, therefore, it is imperative to untangle such meshes for use with PDE solvers.

Existing mesh untangling methods employ geometric- and/or optimization-based techniques in order to untangle the mesh. For example, Mesquite, the Mesh Quality Improvement Toolkit [1], uses a global objective function for mesh untangling which is based upon the elemental Jacobian determinants and the volume of the geometry to be meshed [2]. An important limitation of such methods is that the untangled mesh quality is often not very good since the mesh quality itself is not optimized.

In this talk, we present a physics-based, force-directed method for mesh untangling which seeks to address this issue. Our algorithm is based upon the Fruchterman-Reingold(FR) [3] graph embedding algorithm which aims to lay out an aesthetically-pleasing graph on a two- (or three)-dimensional plane. The FR algorithm calculates forces on the vertices based on models of n-body problems, such as planetary simulation. The algorithm considers two forces, namely an attractive force (between connected vertices) and a repulsive force (between each pair of vertices), based on Hooke's and Coulomb's Laws, respectively.

At each iteration of the algorithm, the forces acting on individual vertices are calculated, and the vertices are moved accordingly. To ensure convergence, the algorithm is associated with a cooling schedule, and the vertex displacement is limited by the corresponding temperature value. As the layout approaches a better configuration, the temperature decreases, thus restricting the displacement.

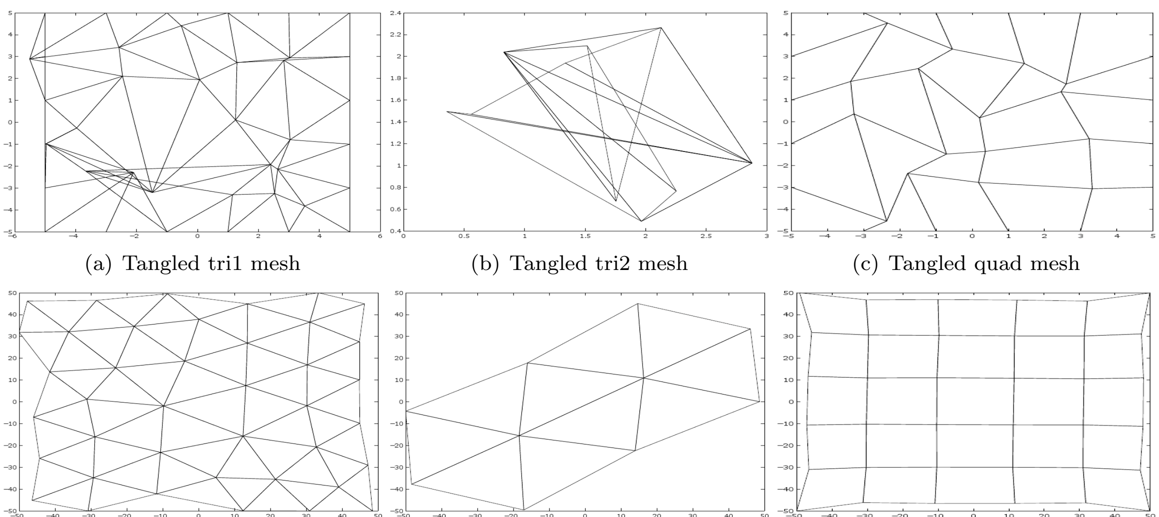
We observe that given the initial coordinates of a tangled mesh as input to the graph embedding algorithm, we can achieve untangling within 50 iterations. In addition, the quality of the elements as measured using the element condition number is near unity. We compare our results against those obtained using the untangling algorithm implemented in Mesquite.

Table 1 presents our preliminary results on three 2D unstructured meshes. It can be seen that the meshes obtained using our graph embedding technique are of higher quality than those obtained using Mesquite. Figure 1 shows the meshes before and after untangling via the graph embedding method.

For future work, we plan to extend our graph embedding mesh untangling algorithm to 3D and to apply our technique to untangle meshes resulting from computational mechanics applications involving large deformations. We also plan to improve the performance of the embedding code and to reduce the number of untangling iterations by applying the algorithm only on the tangled parts of the mesh. Finally, we plan to investigate the application of graph embedding code in other areas of meshing such as mesh warping.

Table 1: Untangling mesh quality results based on the condition number. The second and third columns represent the Mesquite and graph embedding results, respectively. An X denotes failure to untangle the mesh.

	Initial Mesh Quality			Final Mesh Quality Mesquite			Final Mesh Quality Graph Embedding		
Mesh	**Min**	**Avg**	**Max**	**Min**	**Avg**	**Max**	**Min**	**Avg**	**Max**
tri1	1.018	86K	1e06	1.04	5.44e07	3.15e09	1.00	1.03	1.28
tri2	1.03	300K	1e06	X	X	X	1.00	1.03	1.08
quad	1.01	800K	1e06	1.01	7.35	52.11	1.00	1.02	1.04



(a) Tangled tri1 mesh (b) Tangled tri2 mesh (c) Tangled quad mesh

(d) Untangled tri1 mesh (e) Untangled tri2 mesh (f) Untangled quad mesh

Figure 1: Top row: Tangled meshes. (a) tri1 mesh; (b) tri2 mesh; (c) quad mesh; Bottom row: Untangled meshes using graph embedding. (d) tri1 mesh; (e) tri2 mesh; (f) quad mesh.

References

[1] Brewer, M., Freitag Diachin, L., Knupp, P., Leurent, T., Melander, D: The Mesquite Mesh Quality Improvement Toolkit. In: 12th International Meshing Roundtable, pp. 239–250. Sandia National Laboratories, Albuquerque, NM (2003)

[2] Knupp, P.: Hexahedral mesh untangling and algebraic mesh quality metrics. Eng. with Comput. 17, pp. 261–268 (2001)

[3] Fruchterman T. M. J., Reingold E. M.: Graph Drawing by Force-directed Placement, Software - Practice and Experience, 21, pp. 1129-1164 (1991)