Editors: Isabel Beichl, isabel.beichl@nist.gov
Julian V. Noble, jvn@virginia.edu

DEALING WITH DEGENERACY IN TRIANGULATION

By Isabel Beichl

HERE IS A METHOD FOR DEALING WITH A TYPE OF DEGENERACY THAT OFTEN ARISES IN TRIANGULATION WHEN THE INPUT CONTAINS MUCH REGULARITY, SUCH AS FOUR OR more points that lie on the same circle in 2D computations or five or more on the same sphere in 3D. Figure 1 shows a dramatic application of Francis Sullivan's and my degeneracy method[1] by Alexander Agathos (http://users.forthnet.gr/ath/agalex), who used it for surface reconstruction and simplification.

Triangulation

Triangulations are used for representing objects computationally. Consider a cloud of points in the plane. A triangulation of these vertices is a set of triangles that must fit together exactly. That is, the triangles may not overlap except along edges or vertices. In this article we do not allow the addition of extra points. We fill out the convex set spanned by the input points with triangles. Figure 2 illustrates a triangulation in the plane. In three dimensions, we fill out the convex set spanned by the points with tetrahedra. Although we will not specifically discuss the case of three and higher dimensions in this article, all the ideas extend, and we have used them in 3D.

There is more than one way to triangulate a given set of points. The *Delaunay* triangulation is popular because it tends to produce triangles with larger areas than other triangulations and connects all pairs of vertices of minimum separation. The Delaunay triangulation satisfies the criterion that for every triangle made from input vertices $\bar{x}_1$, $\bar{x}_2$, and $\bar{x}_3$, the circle determined by those three vertices contains no other vertex in its interior. ($\bar{x}_1$, $\bar{x}_2$, and $\bar{x}_3$ are 2D vectors.) This criterion determines a triangulation uniquely, in the absence of degeneracy. This article is about what to do when there *is* degeneracy. First, we discuss triangulation itself.

One way to approach the idea behind the Delaunay triangulation in 2D is to lift the points on the plane onto a paraboloid in 3D; that is,

$$(x, y) \rightarrow (x, y, x^2 + y^2).$$

Then, the boundary of the lower convex hull in 3D will be a set of triangles. A convex hull is the smallest convex set containing the input vertices. (Occasionally, people say "convex hull" when they mean the convex hull's boundary.) The *lower* convex hull's boundary is the triangles that have a normal that points in the negative third z direction.

Figure 3 shows a set of 2D points projected onto a paraboloid and triangulated. When projected back down to the plane, the facets are a triangulation of the original point set.

The Delaunay triangulation has some other interesting properties. The plane determined by a facet of the convex hull on the paraboloid intersects the paraboloid in an ellipse. When this ellipse is projected back to the plane, it becomes a circle. No points of the data on the ellipse are in front of the intersecting plane, so none of the original points in the plane are in the circle. So, in the Delaunay triangulation, circles determined by the three vertices of each triangle contain no other vertices in their interior.

I don't intend here to describe in detail how to construct a Delaunay triangulation, but just to set the stage, I'll give a brief sketch. Naturally, you can find just the boundary of the lower convex hull of the points lifted to the paraboloid. However, this begs the question, because I haven't told you how to find the convex hull's boundary.

One method to get the triangulation is to use the "empty circles" property repeatedly. Start by finding the two points $\bar{x}_1$, $\bar{x}_2$ in the given set that are closest to each other. Because the circle having the line connecting $\bar{x}_1$ and $\bar{x}_2$ as its diameter is empty, it must be an edge of a triangle in the Delaunay triangulation. Next, pick any other point $\bar{x}_3$, and, by solving three equations, determine the radius r and center (c_x, c_y) of the circle determined by the three vertices of the triangle $\bar{x}_1$, $\bar{x}_2$, $\bar{x}_3$. We determine r and (c_x, c_y) by solving these equations:

1521-9615/02/$17.00 © 2002 IEEE

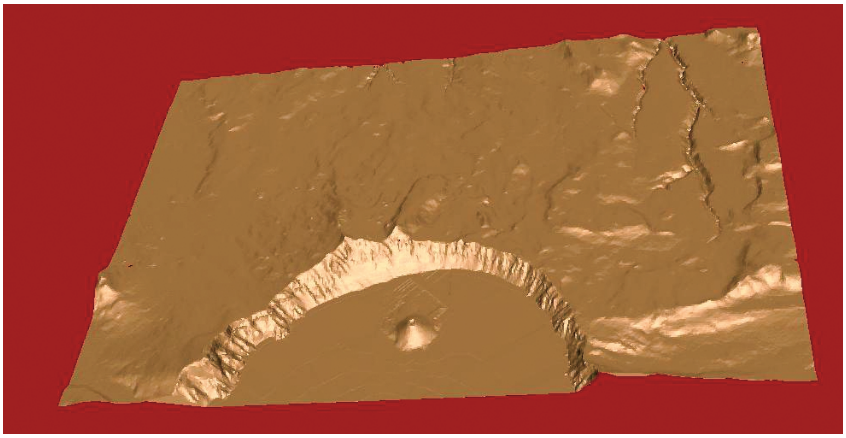

Figure 1. A surface reconstruction. (Figure courtesy of Alexander Agathos.)

$$(x_{1,1}-c_x)^2+(x_{1,2}-c_y)^2=r^2$$
$$(x_{2,1}-c_x)^2+(x_{2,2}-c_y)^2=r^2$$
$$(x_{3,1}-c_x)^2+(x_{3,2}-c_y)^2=r^2.$$

In vector notation, this is

$$\begin{bmatrix}1 & \bar{x}_1^T\\ 1 & \bar{x}_2^T\\ 1 & \bar{x}_3^T\end{bmatrix}\begin{bmatrix}-\mu\\ -2c_x\\ -2c_y\end{bmatrix}=-\begin{bmatrix}\|\bar{x}_1\|^2\\ \|\bar{x}_2\|^2\\ \|\bar{x}_3\|^2\end{bmatrix},$$

where $\mu = r^2 - c_x^2 - c_y^2$ and $\bar{x}_i^T$ is the vector $\bar{x}_i$ written as a row vector.

If this circle is empty, we have one of the Delaunay triangles. If not, pick any vertex in the circle and make a new circle to examine. Because each iteration excludes at least one point, the process ends with a Delaunay triangle. It's an amusing exercise to picture what this iteration looks like when it is lifted to the paraboloid. For more about triangulation, read *Computational Geometry in C*.[2]

Dealing with degeneracy

Figure 4 shows four vertices on a circle. Conceivably, you can make two triangles from this data; to

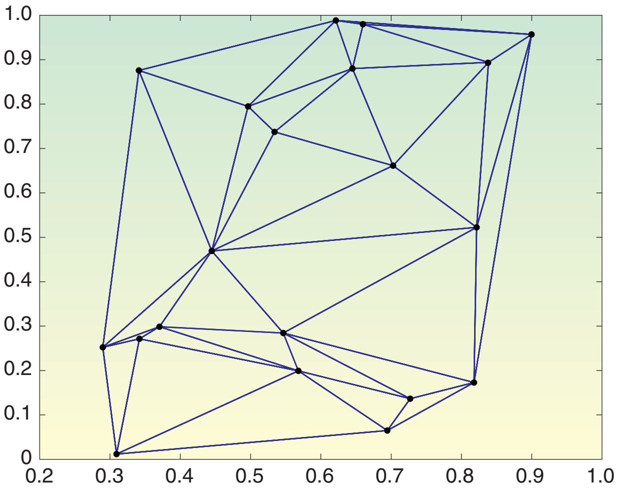

Figure 2. A triangulation in two dimensions.

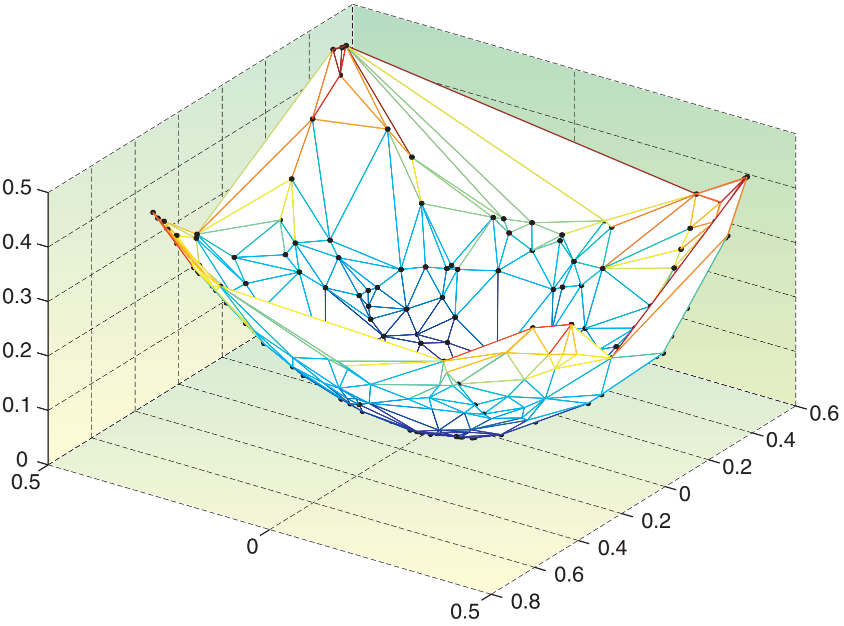

Figure 3. A paraboloid.

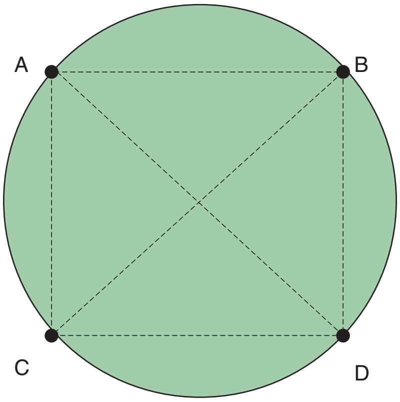

Figure 4. Four degenerate points. Which triangle do you choose, ABC or ABD? It doesn't matter as long as you don't pick both.

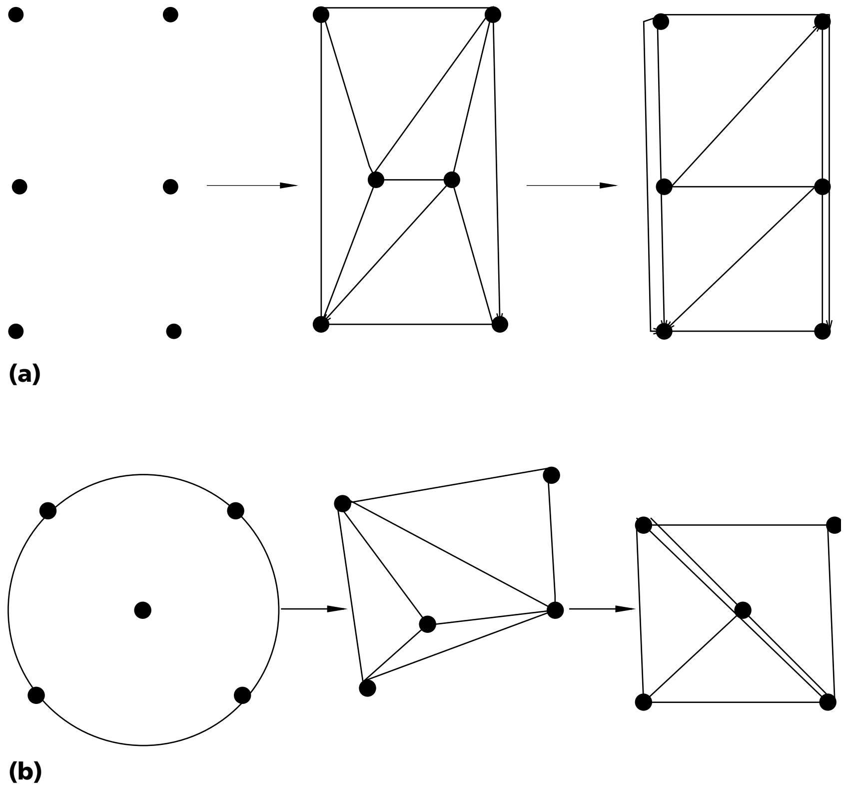

Figure 5. Perturbing points at random before triangulation sometimes produces (a) zero-volume triangles and (b) an incorrect triangulation.

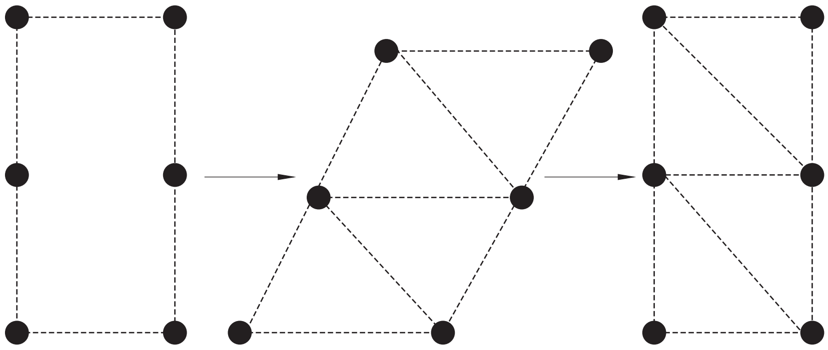

Figure 6. The linear transformation causes one direction to be preferred.

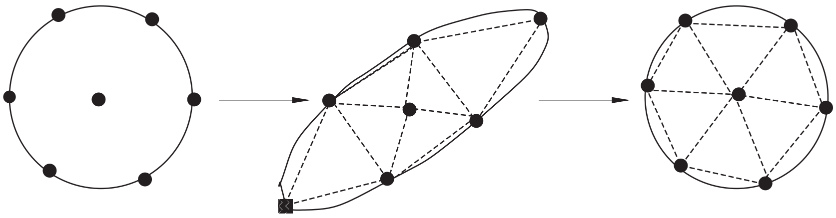

Figure 7. Another example of the linear transformation for dealing with degeneracy.

many people, either would be fine as long as you don't put both into the triangulation.

You might think of perturbing all the points a little and then triangulating. This often produces triangles with zero area (see Figure 5a), which are incorrect. It also produces a triangulation that you might not want (see Figure 5b). You would need at least to go through all the triangles and delete those with zero area. So, simply adding noise is not an optimal way to deal with degeneracy.

Methods that work better but are more complicated are Simulation of Simplicity[3] and using integer and exact arithmetic. The latter methods add considerably to the running time.

Here's a simpler and efficient way to deal with degeneracy that works in most cases. Using a random linear transformation close to the identity, transform the input points. In 2D, the matrix for such a transformation would be

$$\begin{bmatrix} 1+\eta * r_{11} & \eta * r_{12} \\ \eta * r_{21} & 1+\eta * r_{22} \end{bmatrix},$$

and in 3D, it would be

$$\begin{bmatrix} 1+\eta * r_{11} & \eta * r_{12} & \eta * r_{13} \\ \eta * r_{21} & 1+\eta * r_{22} & \eta * r_{23} \\ \eta * r_{31} & \eta * r_{32} & 1+\eta * r_{33} \end{bmatrix},$$

where η is a small positive number and the $r_{i,j}$ are random numbers between 0 and 1. In other words, multiply the input points by a matrix $A = I + \eta R$ instead of replacing the original points $\{\bar{x}_k\}$ by randomly perturbed points. For more about this method, read "Coping with Degeneracies in Delaunay Triangulation."[1]

Why this works

Figures 6 and 7 give an intuitive idea

of why this method works. When degeneracies exist, the linear transformation preserves linearity and still causes one direction to be preferred.

If η is small and R is a matrix of random numbers, then $A = I + \eta R$ is a nonsingular matrix:

$$\bar{v}_k = A\bar{x}_k = \bar{x}_k + \eta R\bar{x}_k .$$

In this notation, a degeneracy is four points such that

$$\begin{bmatrix} 1 & (A\bar{x}_1)^T \\ 1 & (A\bar{x}_2)^T \\ 1 & (A\bar{x}_3)^T \\ 1 & (A\bar{x}_4)^T \end{bmatrix} \begin{bmatrix} -\mu \\ -2c_x \\ -2c_y \end{bmatrix} = -\begin{bmatrix} \|A\bar{x}_1\|^2 \\ \|A\bar{x}_2\|^2 \\ \|A\bar{x}_3\|^2 \\ \|A\bar{x}_4\|^2 \end{bmatrix} .$$

Here (c_x, c_y) is the center of the empty circle determined by the four degenerate points and $\mu = r^2 - c_x^2 - c_y^2$, where r is the empty circle's radius. $(A\bar{x})^T$ is the vector $A * \bar{x}$ transposed—that is, written as a row vector.

Written differently, this is equivalent to saying that a certain determinant is zero. Using the determinant's properties, we get

$$\begin{vmatrix} 1 & \bar{v}_1^T & \|\bar{v}_1\|^2 \\ 1 & \bar{v}_2^T & \|\bar{v}_2\|^2 \\ 1 & \bar{v}_3^T & \|\bar{v}_3\|^2 \\ 1 & \bar{v}_4^T & \|\bar{v}_4\|^2 \end{vmatrix} = \det(A) \begin{vmatrix} 1 & \bar{x}_1^T & \|A\bar{x}_1\|^2 \\ 1 & \bar{x}_2^T & \|A\bar{x}_2\|^2 \\ 1 & \bar{x}_3^T & \|A\bar{x}_3\|^2 \\ 1 & \bar{x}_4^T & \|A\bar{x}_4\|^2 \end{vmatrix} = 0 .$$

where $\bar{v} = A\bar{x}$. (For an explanation of why this equation works, see the sidebar.)

Because η is small, A is nonsingular, so $\det(A) \neq 0$ This means that the right-most determinant must be zero. So, for any x,

Why the Determinant Formula Is True

We will show that for the 2 × 2 case,

$$\begin{vmatrix} 1 & \bar{v}_1^T & \|\bar{v}_1\|^2 \\ 1 & \bar{v}_2^T & \|\bar{v}_2\|^2 \\ 1 & \bar{v}_3^T & \|\bar{v}_3\|^2 \\ 1 & \bar{v}_4^T & \|\bar{v}_4\|^2 \end{vmatrix} = \det(A) \begin{vmatrix} 1 & \bar{x}_1^T & \|A\bar{x}_1\|^2 \\ 1 & \bar{x}_2^T & \|A\bar{x}_2\|^2 \\ 1 & \bar{x}_3^T & \|A\bar{x}_3\|^2 \\ 1 & \bar{x}_4^T & \|A\bar{x}_4\|^2 \end{vmatrix} = 0 ,$$

where $\bar{v}_i = A\bar{x}_i$ ($i = 1 \ldots 4$). Similar proofs hold in higher dimensions.

Let

$$\bar{x}_i = \begin{pmatrix} x_{i,1} \\ x_{i,2} \end{pmatrix}$$

and

$$\bar{v}_i = A * \bar{x}_i = \begin{pmatrix} a_{1,1} & a_{1,2} \\ a_{2,1} & a_{2,2} \end{pmatrix} \begin{pmatrix} x_{i,1} \\ x_{i,2} \end{pmatrix} = \begin{pmatrix} a_{1,1}x_{i,1} + a_{1,2}x_{i,2} \\ a_{2,1}x_{i,1} + a_{2,2}x_{i,2} \end{pmatrix} .$$

So,

$$\begin{vmatrix} 1 & \bar{v}_1^T & \|\bar{v}_1\|^2 \\ 1 & \bar{v}_2^T & \|\bar{v}_2\|^2 \\ 1 & \bar{v}_3^T & \|\bar{v}_3\|^2 \\ 1 & \bar{v}_4^T & \|\bar{v}_4\|^2 \end{vmatrix} = \begin{vmatrix} 1 & (A\bar{x}_1)^T & \|\bar{v}_1\|^2 \\ 1 & (A\bar{x}_2)^T & \|\bar{v}_2\|^2 \\ 1 & (A\bar{x}_3)^T & \|\bar{v}_3\|^2 \\ 1 & (A\bar{x}_4)^T & \|\bar{v}_4\|^2 \end{vmatrix}$$

$$= \begin{vmatrix} 1 & a_{1,1}x_{1,1} + a_{1,2}x_{1,2} & a_{2,1}x_{1,1} + a_{2,2}x_{1,2} & \|A\bar{x}_1\|^2 \\ 1 & a_{1,1}x_{2,1} + a_{1,2}x_{2,2} & a_{2,1}x_{2,1} + a_{2,2}x_{2,2} & \|A\bar{x}_2\|^2 \\ 1 & a_{1,1}x_{3,1} + a_{1,2}x_{3,2} & a_{2,1}x_{3,1} + a_{2,2}x_{3,2} & \|A\bar{x}_3\|^2 \\ 1 & a_{1,1}x_{4,1} + a_{1,2}x_{4,2} & a_{2,1}x_{4,1} + a_{2,2}x_{4,2} & \|A\bar{x}_4\|^2 \end{vmatrix}$$

$$= \begin{vmatrix} 1 & x_{1,1} & x_{1,2} & \|A\bar{x}_1\|^2 \\ 1 & x_{2,1} & x_{2,2} & \|A\bar{x}_2\|^2 \\ 1 & x_{3,1} & x_{3,2} & \|A\bar{x}_3\|^2 \\ 1 & x_{4,1} & x_{4,2} & \|A\bar{x}_4\|^2 \end{vmatrix} * \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & a_{1,1} & a_{2,1} & 0 \\ 0 & a_{1,2} & a_{2,2} & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix} .$$

Because the determinant of a product is the product of the determinants, and because $\det(A^T) = \det(A)$, we get

$$\begin{vmatrix} 1 & x_{1,1} & x_{1,2} & \|A\bar{x}_1\|^2 \\ 1 & x_{2,1} & x_{2,2} & \|A\bar{x}_2\|^2 \\ 1 & x_{3,1} & x_{3,2} & \|A\bar{x}_3\|^2 \\ 1 & x_{4,1} & x_{4,2} & \|A\bar{x}_4\|^2 \end{vmatrix} * \det(A^T) = \det(A) \begin{vmatrix} 1 & \bar{x}_1^T & \|A\bar{x}_1\|^2 \\ 1 & \bar{x}_2^T & \|A\bar{x}_2\|^2 \\ 1 & \bar{x}_3^T & \|A\bar{x}_3\|^2 \\ 1 & \bar{x}_4^T & \|A\bar{x}_4\|^2 \end{vmatrix} .$$

$$\|A\bar{x}\|^2 = \|\bar{x} + \eta R\bar{x}\|^2 = \|\bar{x}\|^2 + 2\eta\langle \bar{x}, R\bar{x}\rangle + \eta^2\|R\bar{x}\|^2,$$

where $\langle ., .\rangle$ is the inner product. Because a determinant is a real-valued, linear function of its columns, the previous equation plus the fact that the right-most determinant is zero tells us that we have a quadratic in η with determinants as coefficients that is equal to zero. (In case you don't recall, when two matrices are equal except for one column, then the determinant of their sum is the sum of their determinants.) That is,

$$D_0 + 2D_1\eta + D_2\eta^2 = 0,$$

where D_0, D_1, and D_2 are the determinants.

Because η is a root,

$$\eta = \frac{-2D_1 \pm \left(4D_1^2 - 4D_2D_0\right)^{1/2}}{2D_2}.$$

However, η is a *single fixed-input quantity*, merely the perturbation's size. But D_1 and D_2 vary continuously with R. So, for the degeneracy to remain after you use this method would be a measure-zero event—that is, virtually impossible.

In a practical sense, this method often works well and has a low cost. It's certainly worth trying if degeneracy is a problem and you don't want to spend a lot of programmer and computer time on it. Finding the appropriate η would depend on the problem—mainly on the smallest distance between pairs of nearest neighbors. This method is certainly faster than alternative methods using exact and integer arithmetic, which is what people sometimes resort to.

References

1. I. Beichl and F. Sullivan, "Coping with Degeneracies in Delaunay Triangulation," *Modeling, Mesh Generation and Adaptive Numerical Methods for Partial Differential Equations*, J.E. Flaherty et al., eds., Springer-Verlag, New York, 1995, pp. 23–30.
2. J. O'Rourke, *Computational Geometry in C*, Cambridge Univ. Press, Cambridge, UK, 1994.
3. H. Edelsbrunner and E. Muecke, "Simulation of Simplicity: A Technique to Cope with Degenerate Cases in Geometric Algorithms," *ACM Trans. Graphics*, vol. 9, no. 1, Jan. 1990, pp. 66–104.

Isabel Beichl is the department editor for Computing Prescriptions. She is also a mathematician at the National Institute of Standards and Technology. Contact her at isabel.beichl@nist.gov.