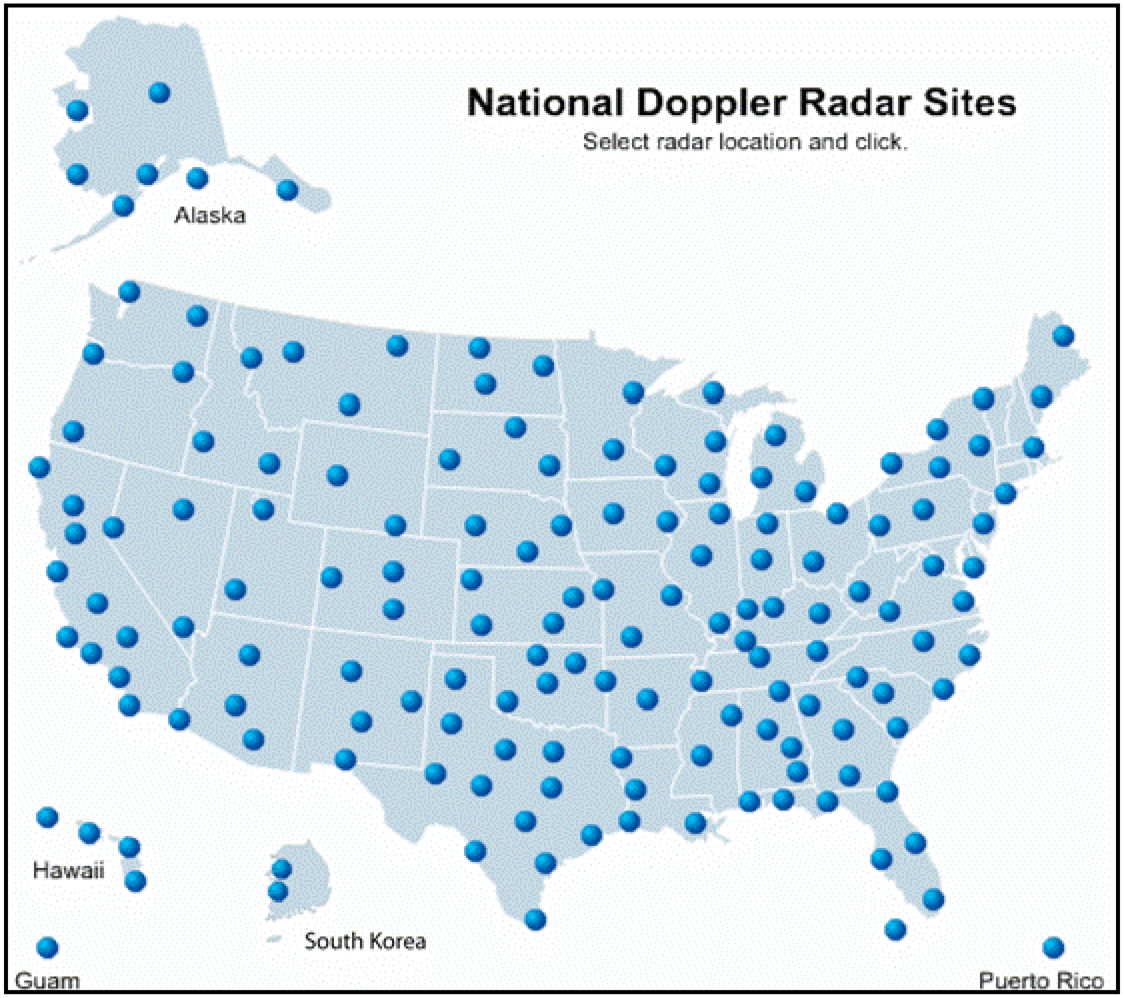
Resampling Scattered Data into a Regular Grid

Mike Bailey

mjb@cs.oregonstate.edu

Oregon State University

The Problem



http://www.ncdc.noaa.gov/nexradinv/

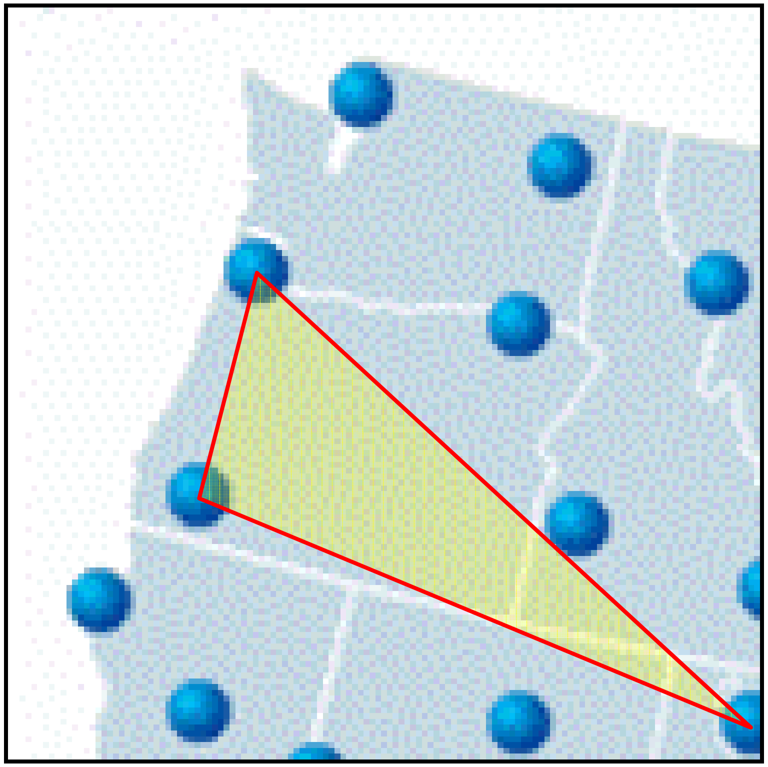
Oftentimes data points are located irregularly, that is, they are not in nice rectilinear grids. This is called ***Scattered Data***.

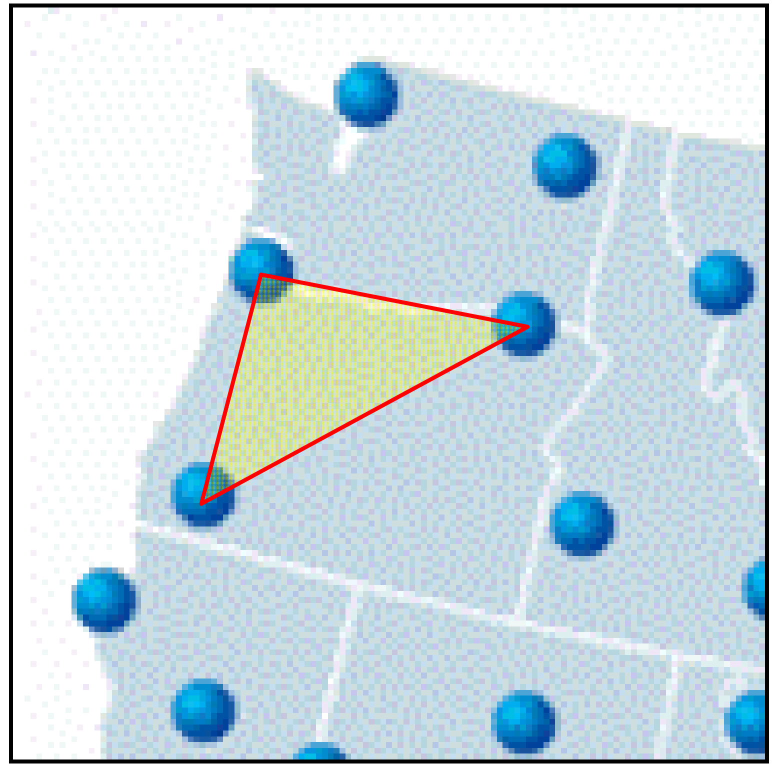
To use the Interpolated Color method (like in Project 4), we need to triangulate the data so we can draw color-interpolated triangles.

To use Contours, we need to triangulate the data and re-sample it into a rectangular grid.

How do we do this?

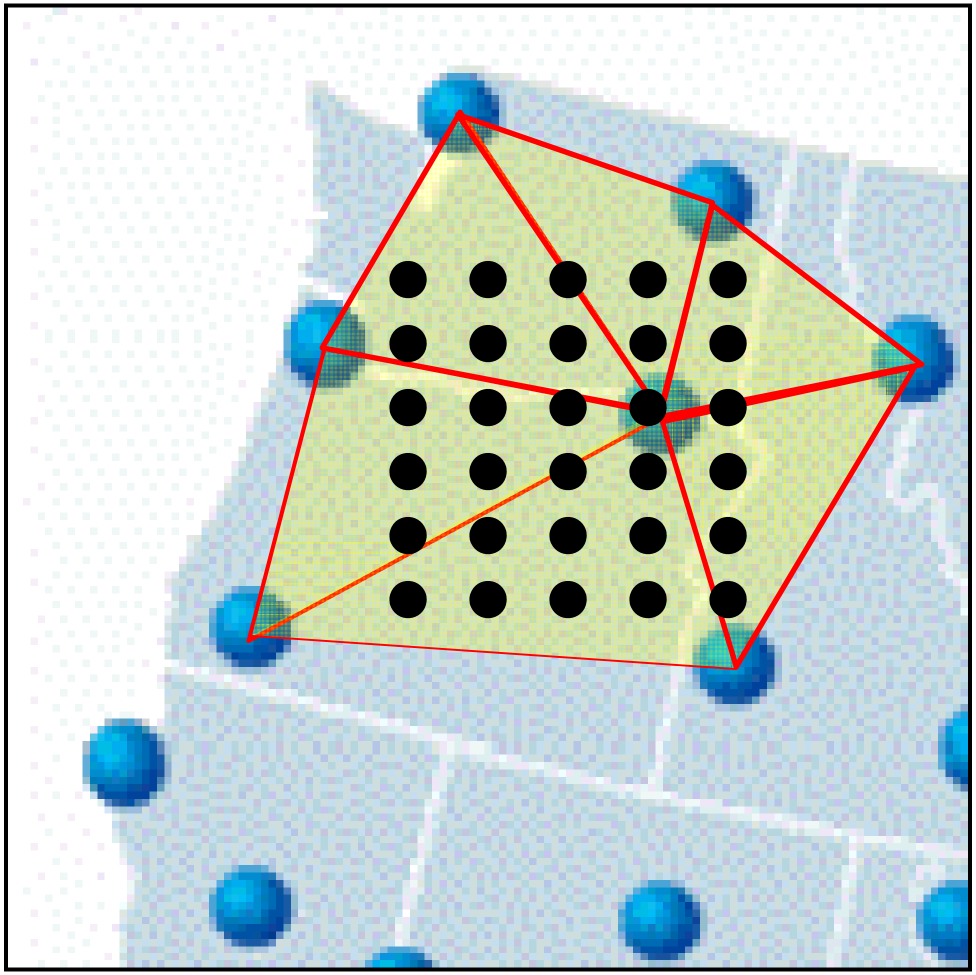
Not all Triangularizations are Created Equal: Which of These is Better, and Why?





http://www.ncdc.noaa.gov/nexradinv/

Once you have a Good Triangularization, You Can Superimpose any Data Grid You Want, and Re-sample the Data Values There



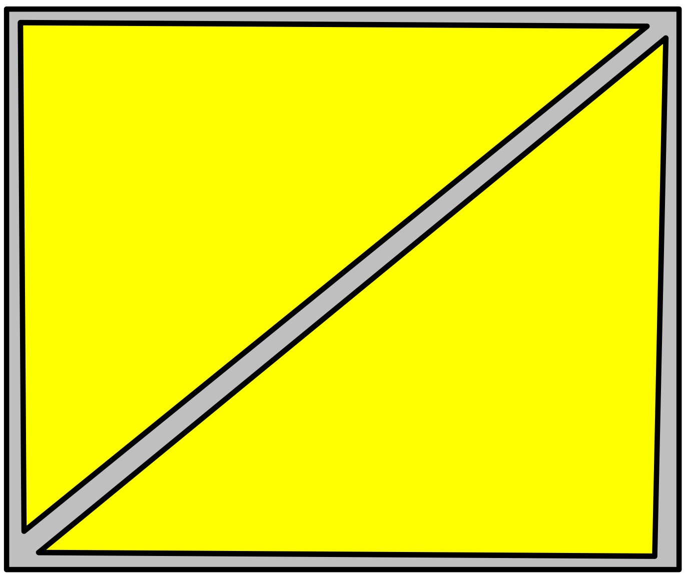
http://www.ncdc.noaa.gov/nexradinv/

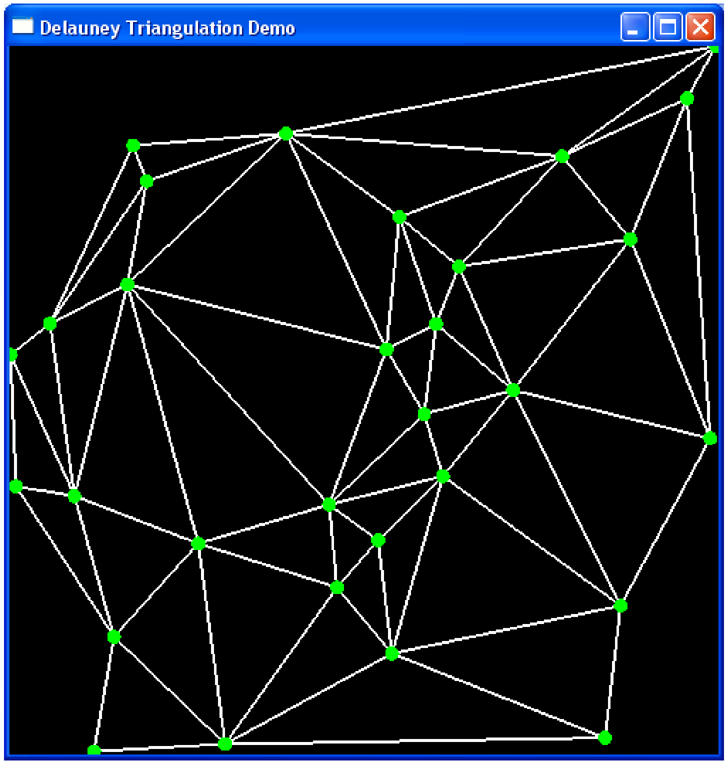
Three Steps

1. Fit a good set of triangles through the scattered points
2. Find out which triangle each new point is in
3. Interpolate within those triangles

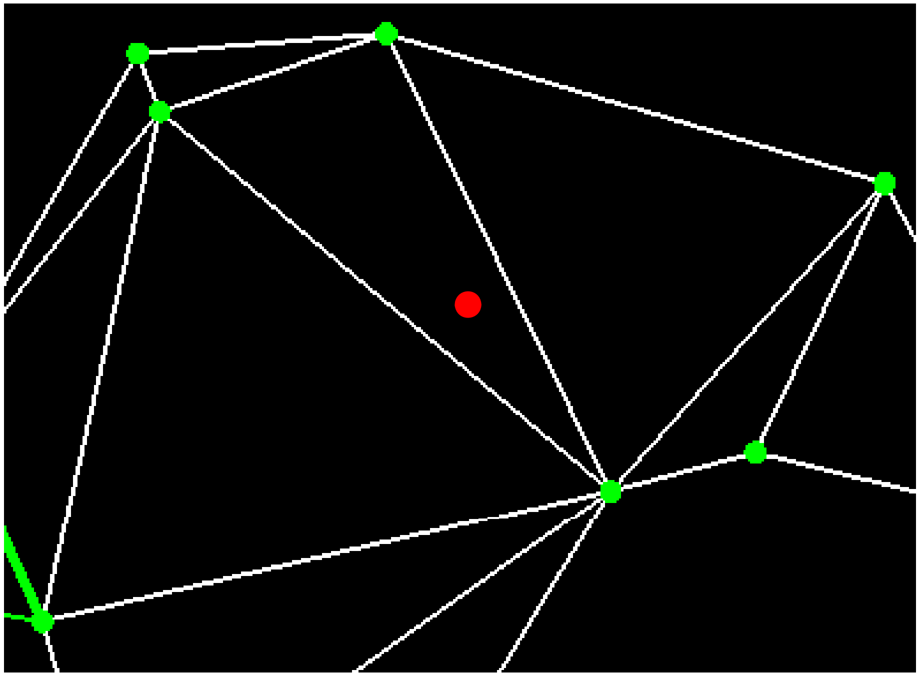
Delauney Triangulation

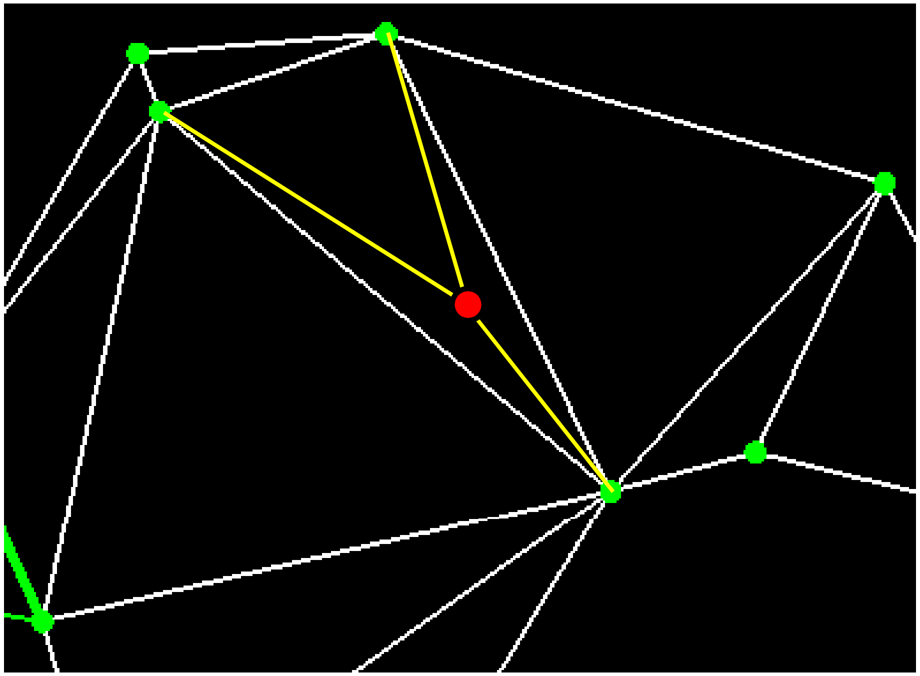
This is an incremental algorithm, that is, you start with a "frame triangularization", and add a point at a time, adjusting the triangularization each time.





Adding a Point



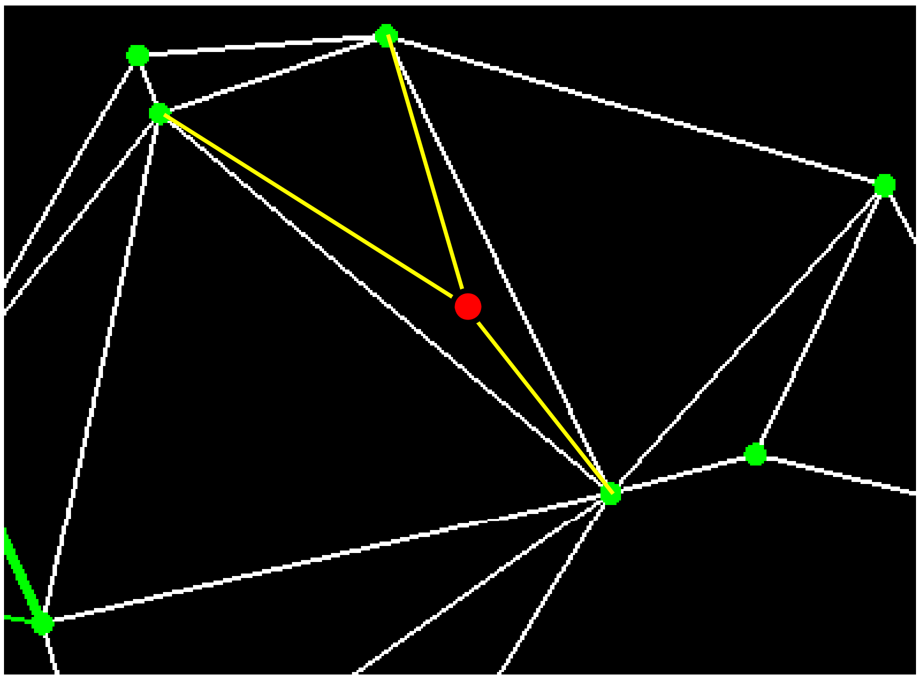


1. Add a new point

2. Figure out which triangle it is in.

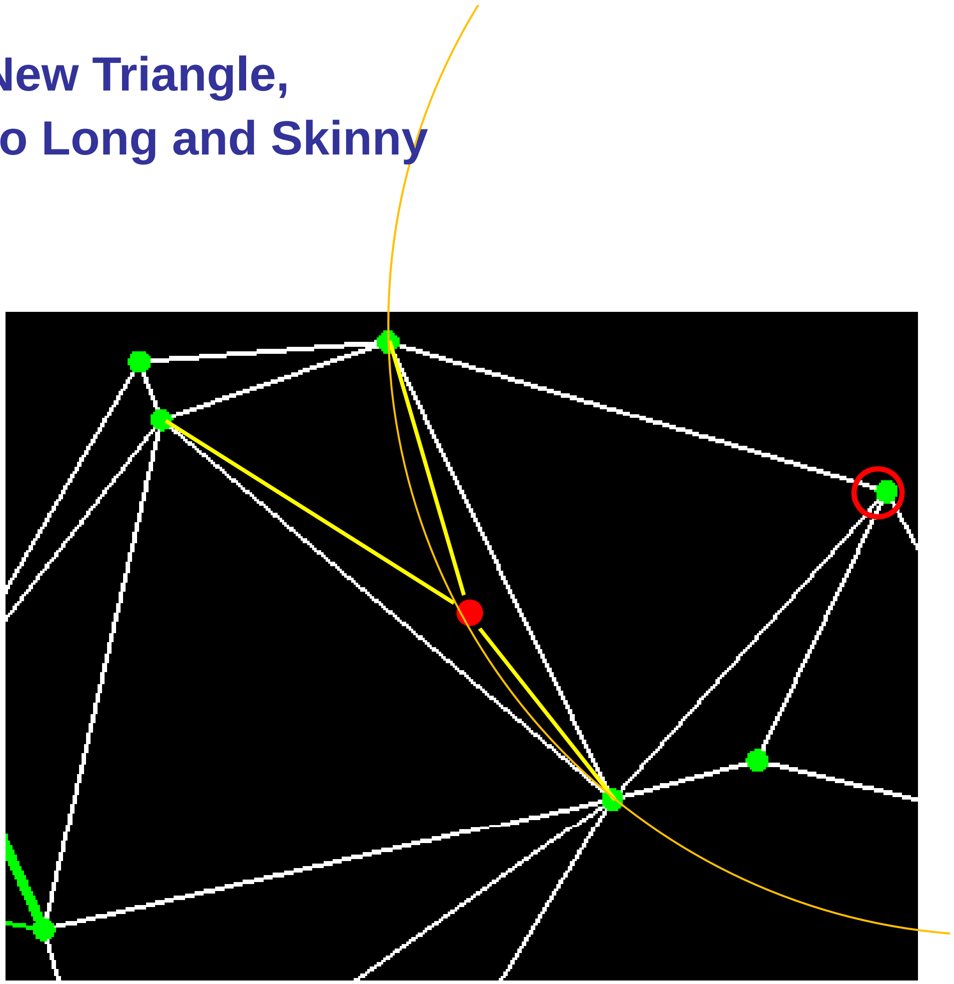
3. Create 3 new triangles by drawing lines from the new point to the 3 vertices of the bounding triangle

Look at each New Triangle, and Decide if it is Too Long and Skinny



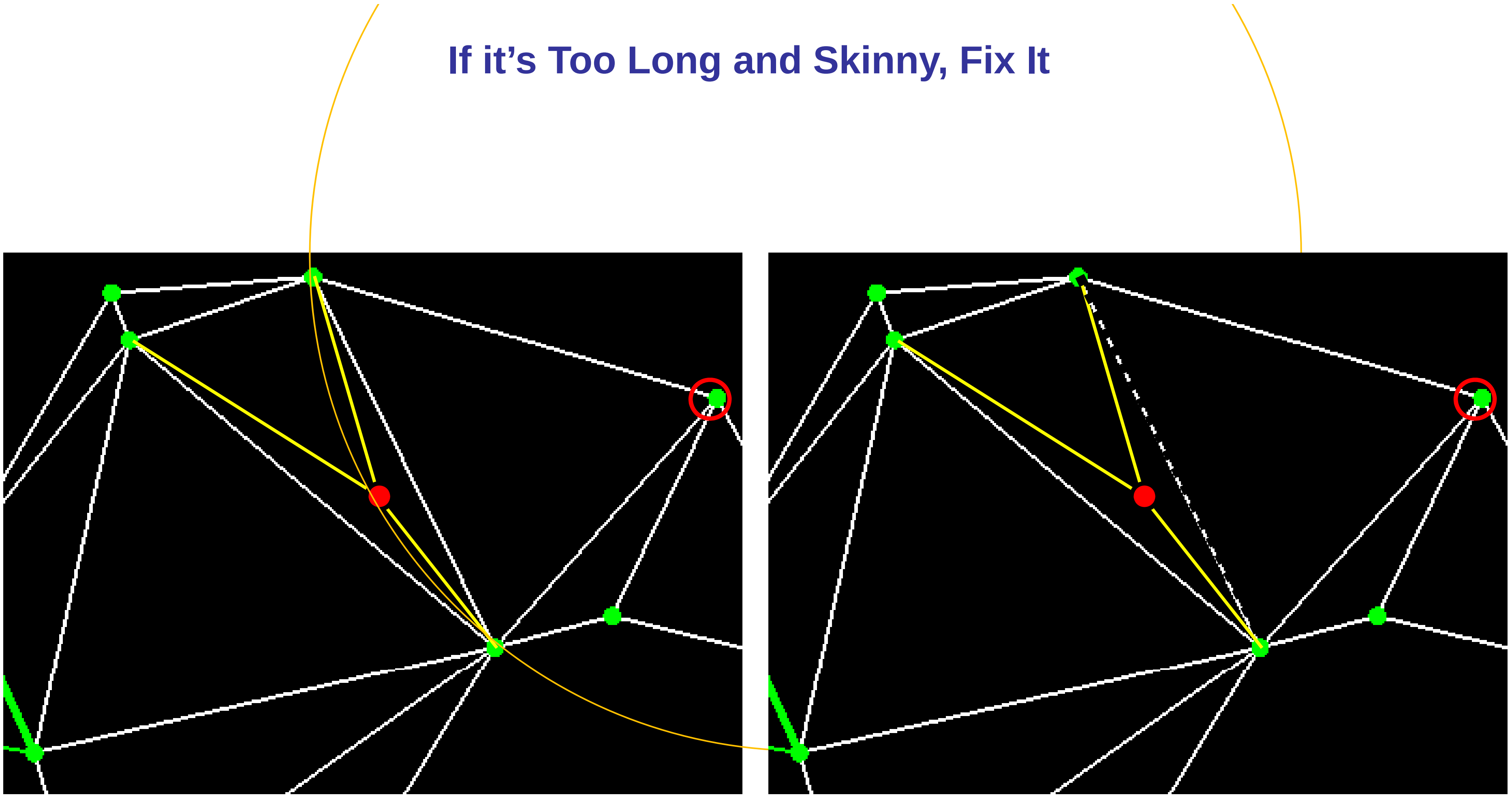
2. Figure out which triangle it is in.

3. Create 3 new triangles by drawing lines from the new point to the 3 vertices of the bounding triangle



4. For each of the 3 new triangles, fit a circle through the 3 vertices (the new point, and the two existing points).

If it’s Too Long and Skinny, Fix It

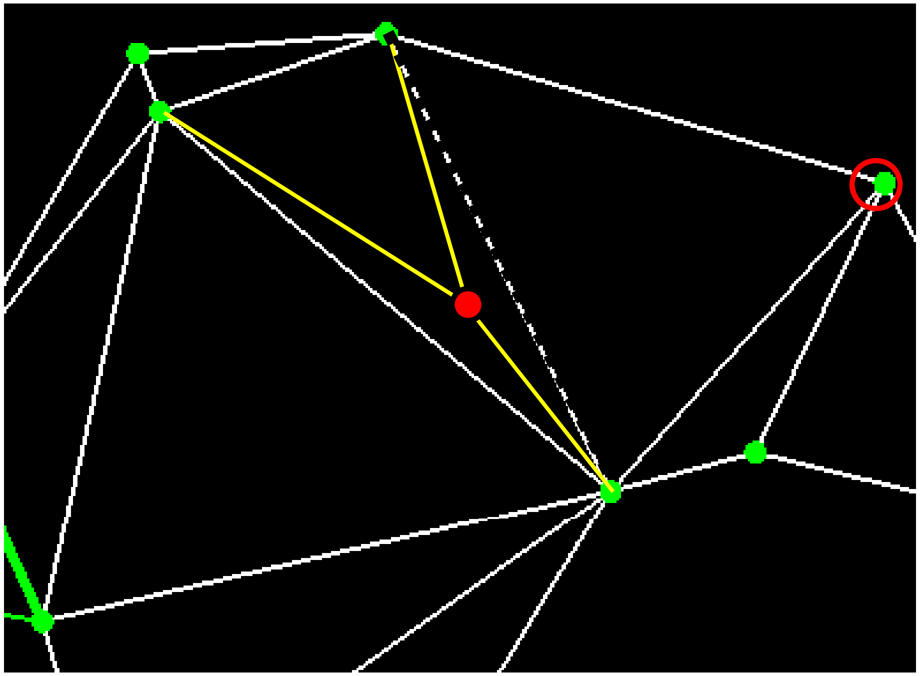


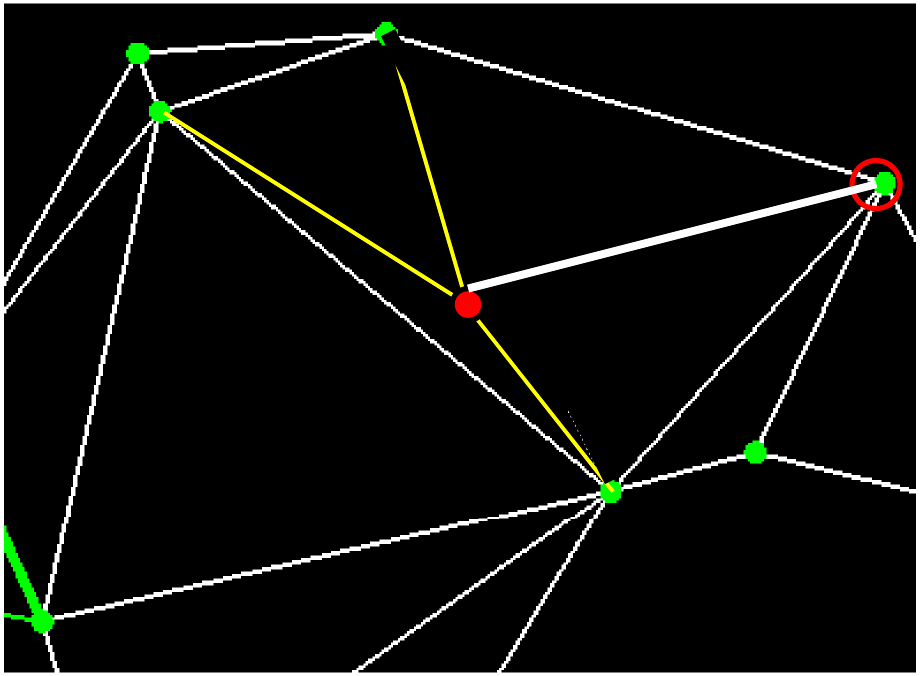
4. For each of the 3 new triangles, fit a circle through the 3 vertices (the new point, and the two existing points).

5. If the opposite point is inside the circle, then the circle is “too big”, indicating that this created triangle is too long and skinny.

6. Delete the existing bounding edge , thus deleting two triangles.

If it’s Too Long and Skinny, Fix It



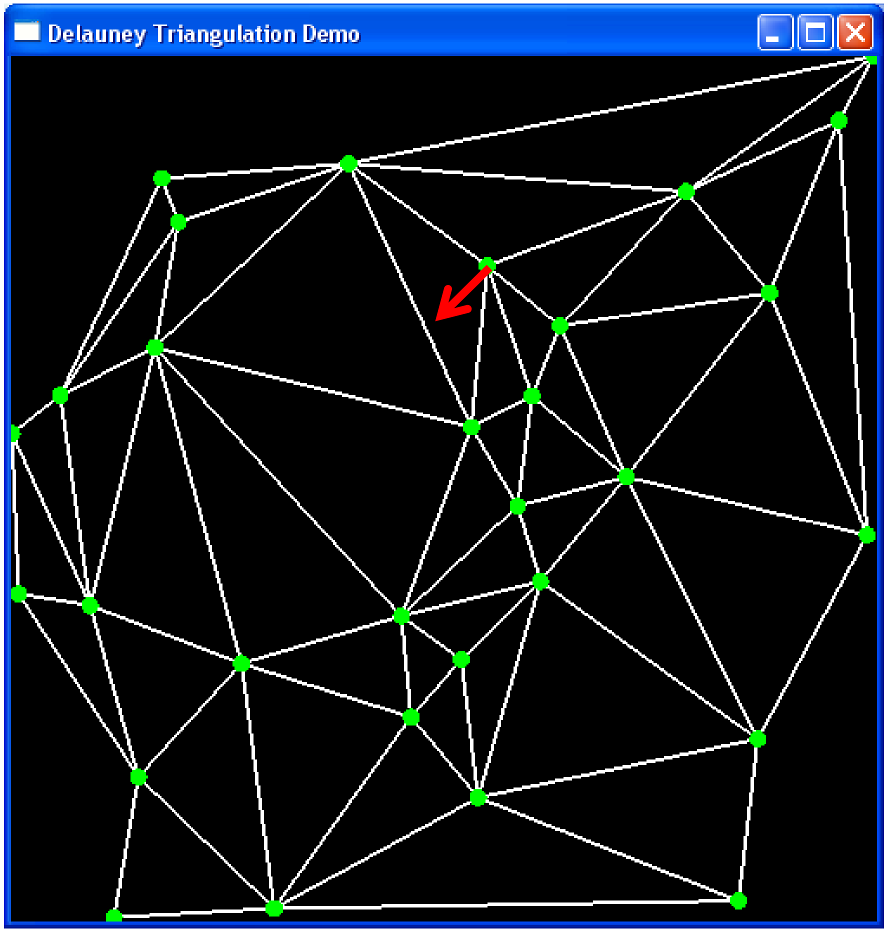


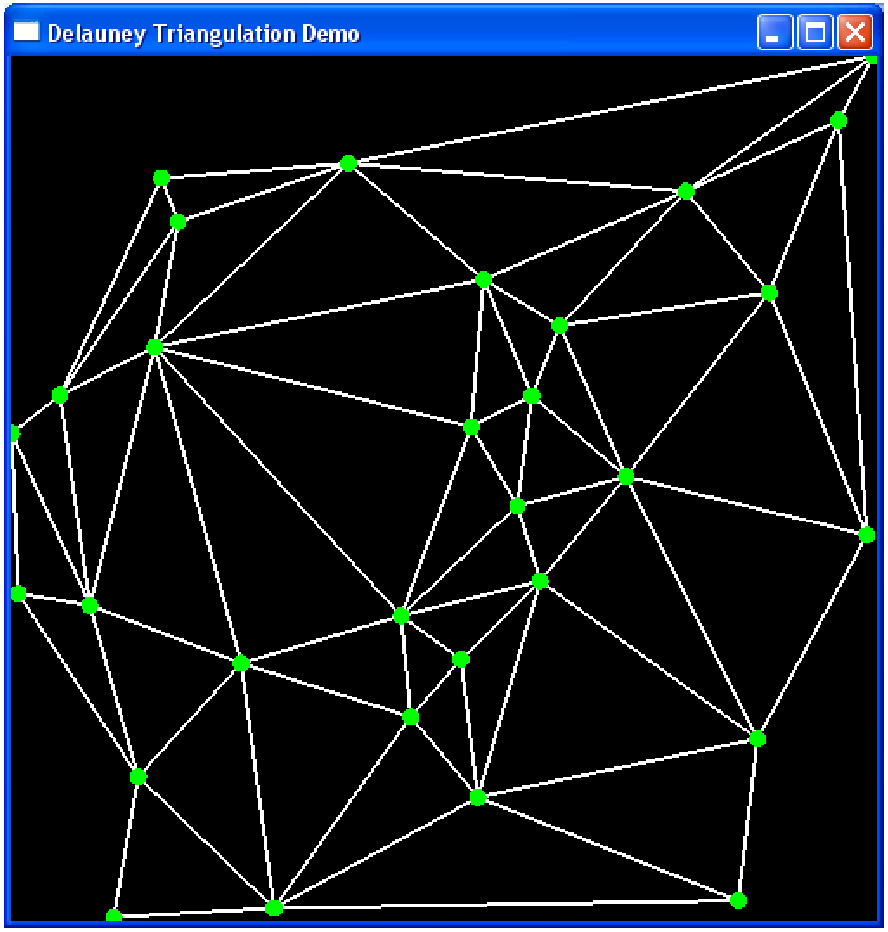
5. If the opposite point is inside the circle, then the circle is “too big”, indicating that this created triangle is too long and skinny.

6. Delete the existing bounding edge, thus deleting two triangles.

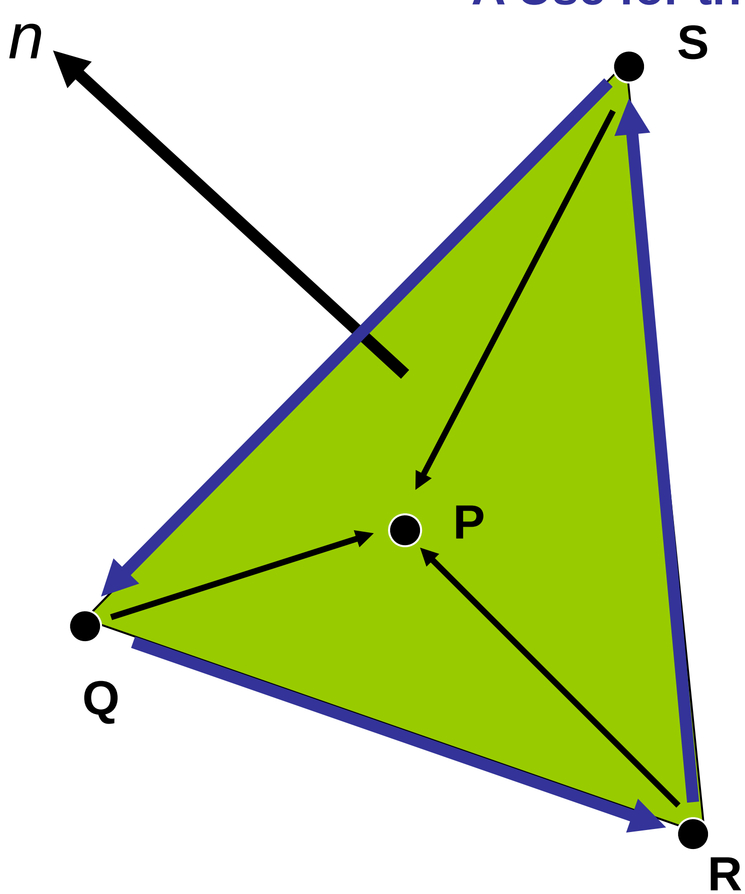
7. Add a cross edge to make 2 new triangles

A Very Slight Change in Point Location will affect the Triangularization





Is a Point inside a Triangle?
A Use for the Cross and Dot Products



Let:

$$n = (R - Q) \times (S - Q)$$

$$n_Q = (R - Q) \times (P - Q)$$

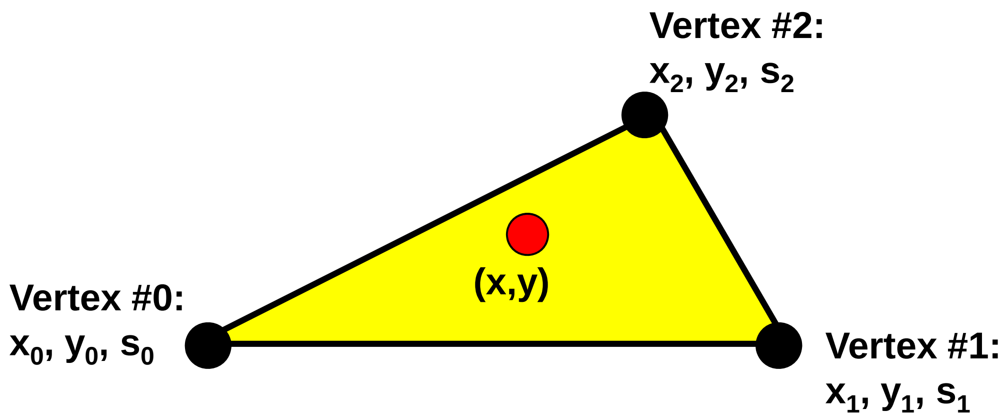
$$n_R = (S - R) \times (P - R)$$

$$n_s = (Q - S) \times (P - S)$$

If $(n \bullet n_q), (n \bullet n_r), and (n \bullet n_s)$
are all positive, then P is inside the triangle QRS

Finding if a point is inside a triangle is used both in the Delauney triangularization algorithm and in re-sampling to a new grid

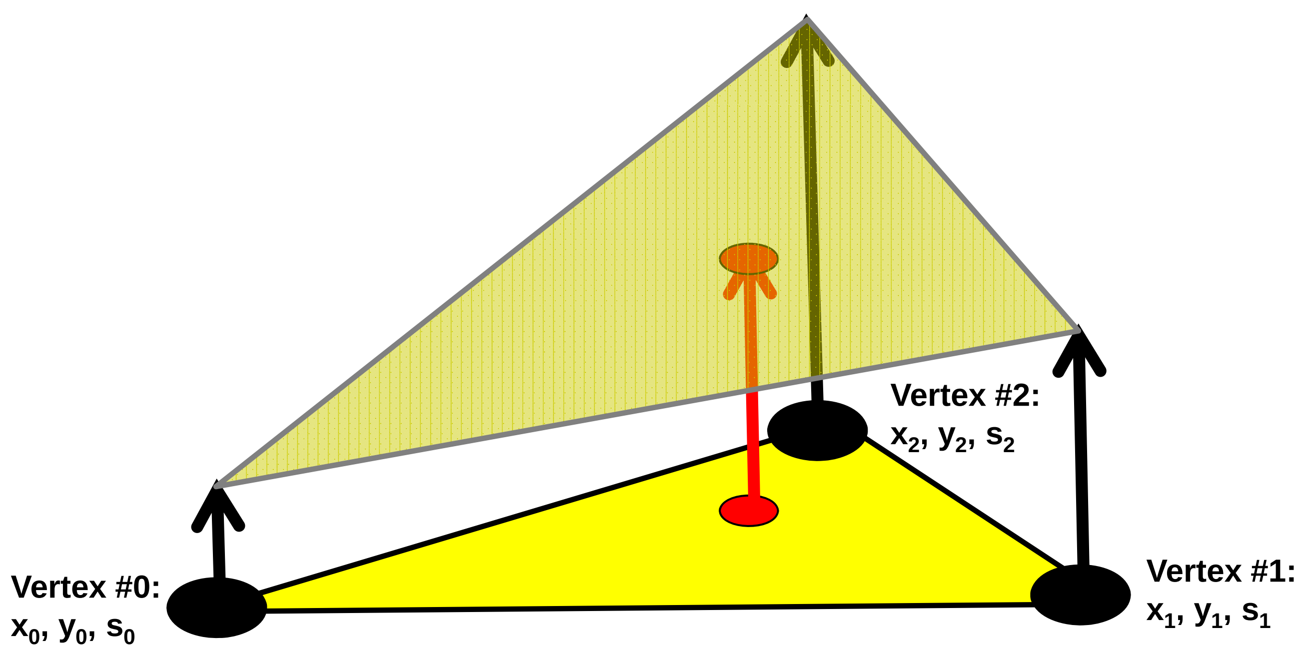
Interpolating Data Values within a Triangle



Once we know the point is within a particular triangle, we need to interpolate within that triangle. Use a linear function:

$$S = Ax + By + C$$

Think of the Scalar Function as Elevations and Think of the Triangle Linear Interpolation Function as a Plane Being Fitted on top of the Data Values



Since, at Vertices 0, 1, and 2, we know x, y, and s, we can write 3 Equations with 3 Unknowns

$$s_0 = Ax_0 + By_0 + C$$

$$s_1 = Ax_1 + By_1 + C$$

$$s_2 = Ax_2 + By_2 + C$$

or, in matrix form:

$$\begin{bmatrix} x_0 & y_0 & 1 \\ x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \end{bmatrix} \begin{Bmatrix} A \\ B \\ C \end{Bmatrix} = \begin{Bmatrix} s_0 \\ s_1 \\ s_2 \end{Bmatrix}$$

You can actually simplify it to 2 Equations with 2 Unknowns

$$s_0 = Ax_0 + By_0 + C$$
$$s_1 = Ax_1 + By_1 + C$$
$$s_2 = Ax_2 + By_2 + C$$

$$\Rightarrow$$

$$s_1 - s_0 = A(x_1 - x_0) + B(y_1 - y_0)$$
$$s_2 - s_0 = A(x_2 - x_0) + B(y_2 - y_0)$$

or, in matrix form:

$$\begin{bmatrix} x_1 - x_0 & y_1 - y_0 \\ x_2 - x_0 & y_2 - y_0 \end{bmatrix} \begin{Bmatrix} A \\ B \end{Bmatrix} = \begin{Bmatrix} s_1 - s_0 \\ s_2 - s_0 \end{Bmatrix}$$

Solve this 2x2 System in your Favorite Way – Cramers Rule Works Well

$$\begin{bmatrix} x_1 - x_0 & y_1 - y_0 \\ x_2 - x_0 & y_2 - y_0 \end{bmatrix} \begin{Bmatrix} A \\ B \end{Bmatrix} = \begin{Bmatrix} s_1 - s_0 \\ s_2 - s_0 \end{Bmatrix}$$

$$A = \frac{(s_1 - s_0)(y_2 - y_0) - (s_2 - s_0)(y_1 - y_0)}{(x_1 - x_0)(y_2 - y_0) - (x_2 - x_0)(y_1 - y_0)}$$

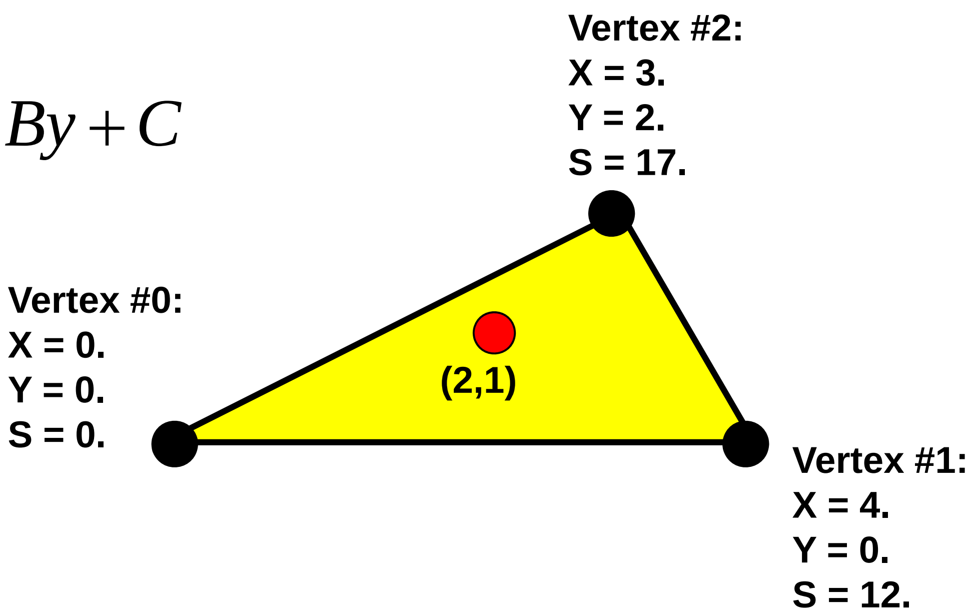
$$B = \frac{(x_1 - x_0)(s_2 - s_0) - (x_2 - x_0)(s_1 - s_0)}{(x_1 - x_0)(y_2 - y_0) - (x_2 - x_0)(y_1 - y_0)}$$

C is then computed by:

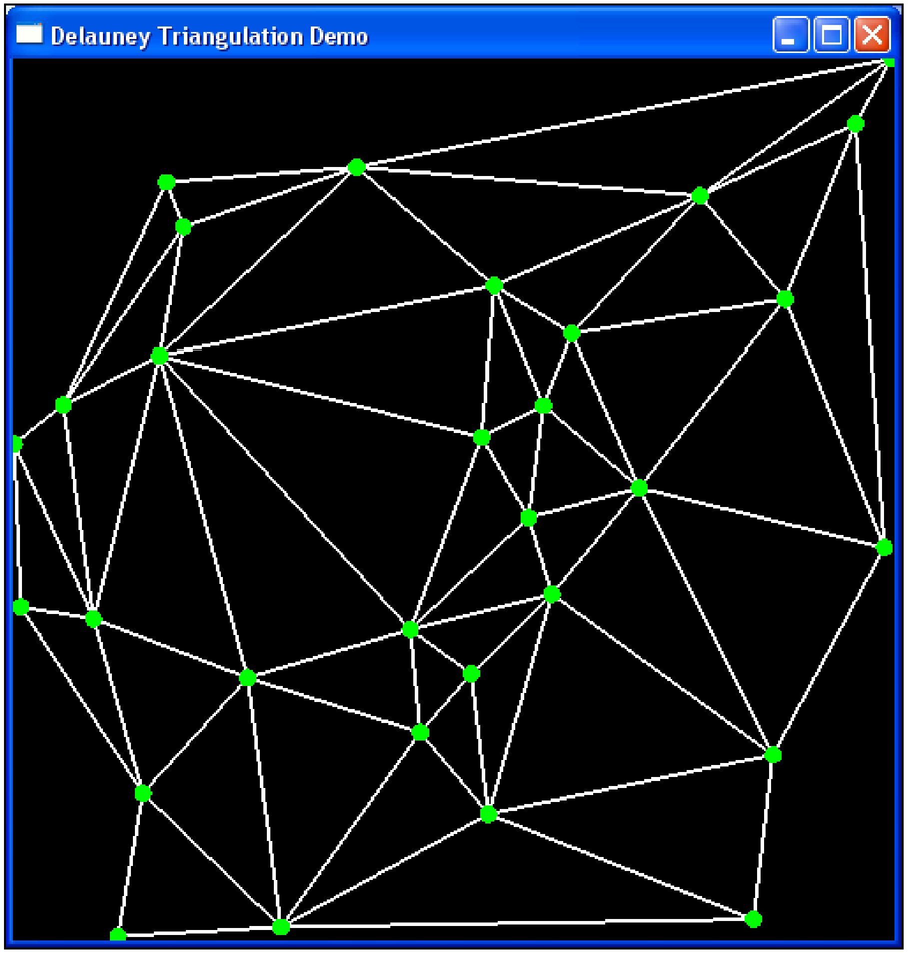
$$C = s_0 - Ax_0 - By_0$$

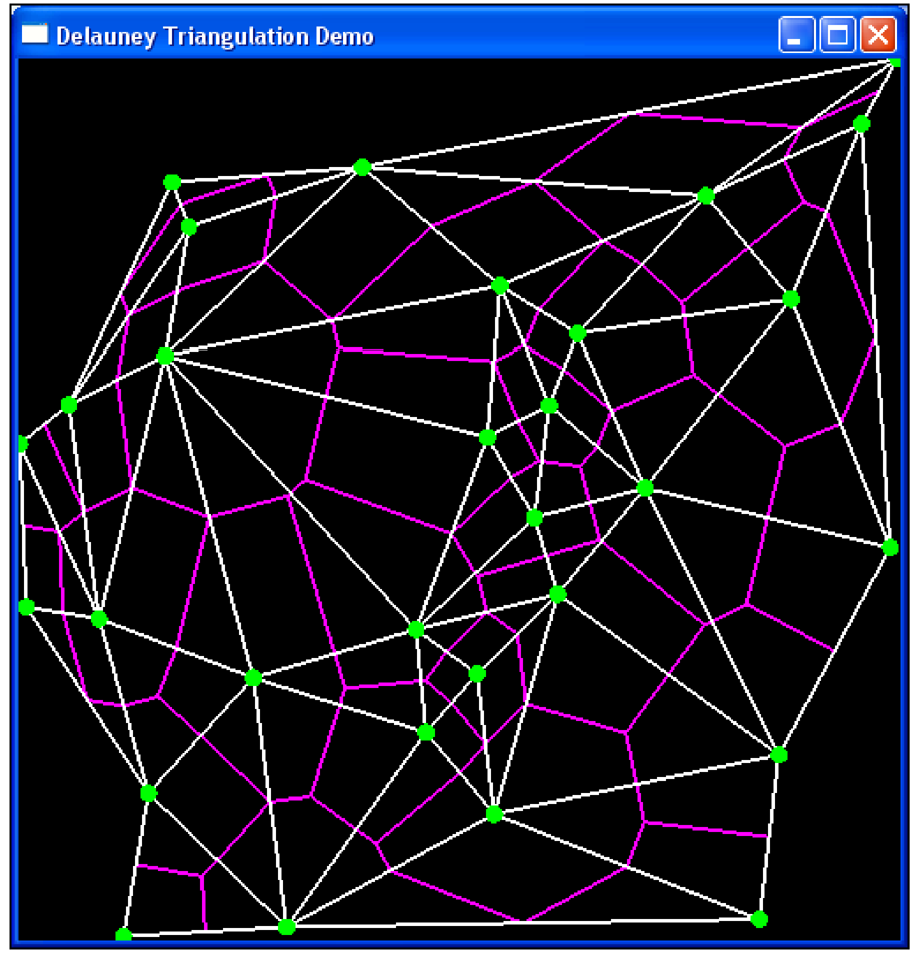
Interpolating Data Values within a Triangle: An Example

$$S = Ax + By + C$$



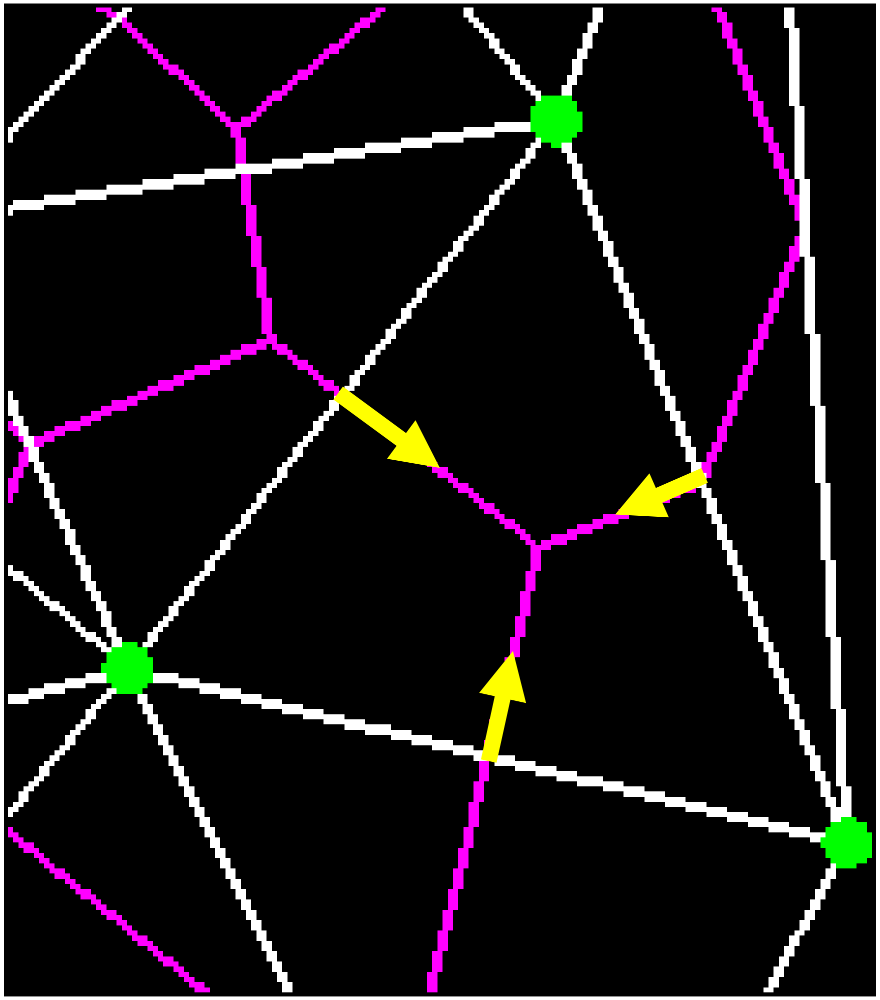
The Delauney Triangles can be used to Derive a Voronoi Diagram





Think of this as showing “Regions of Influence” around a Data Point

Voronoi Diagram:
Most of the Time, the Lines are the Perpendicular Bisectors of the Triangle Edges



Voronoi Regions of Influence