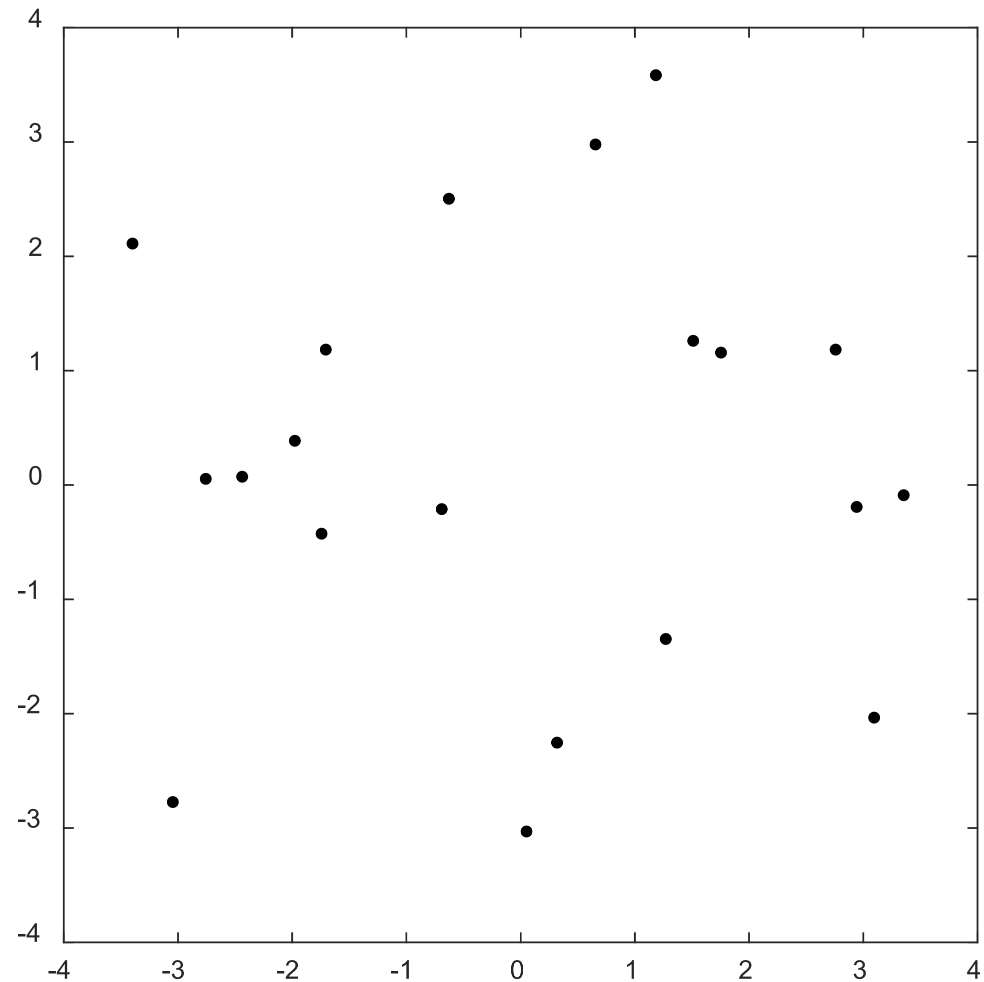
Computational Geometry

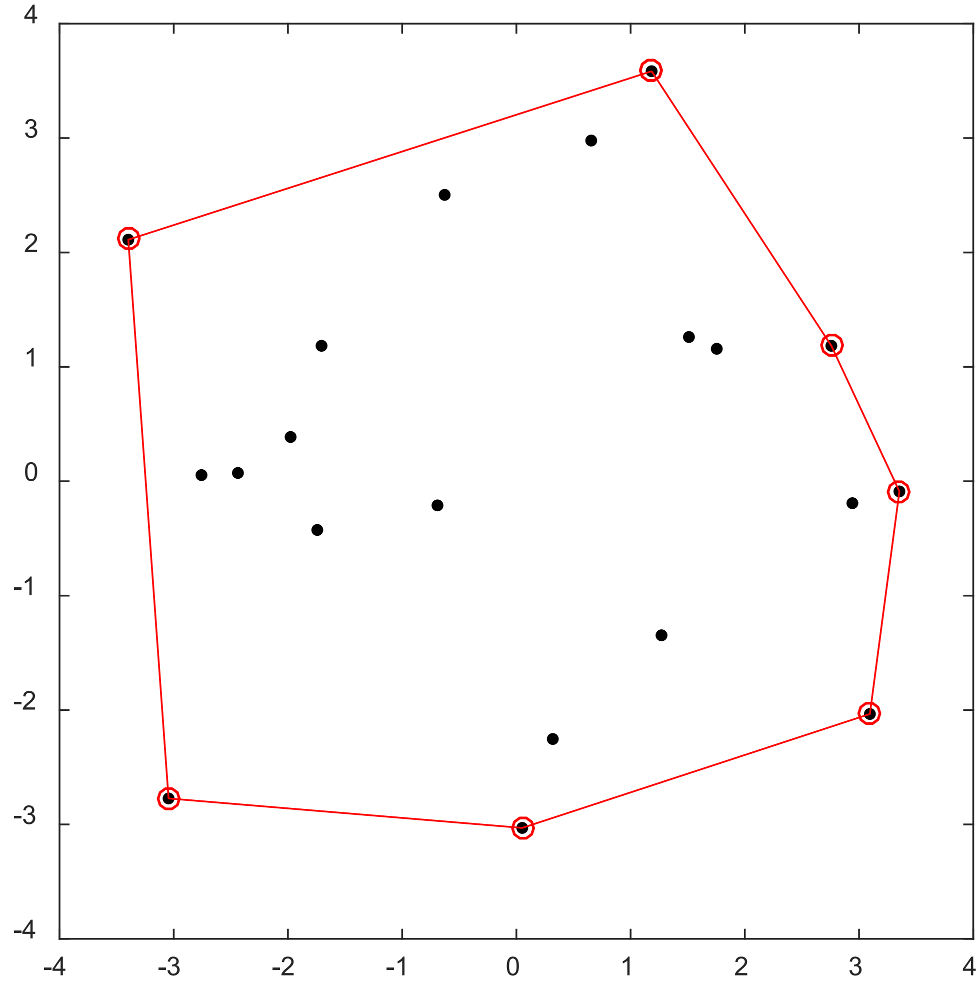
- Design and analysis of algorithms for solving geometric problems
  - Modern study started with Michael Shamos in 1975
- Facility location:
  - geometric data structures used to "simplify" solution procedures



Convex Hull

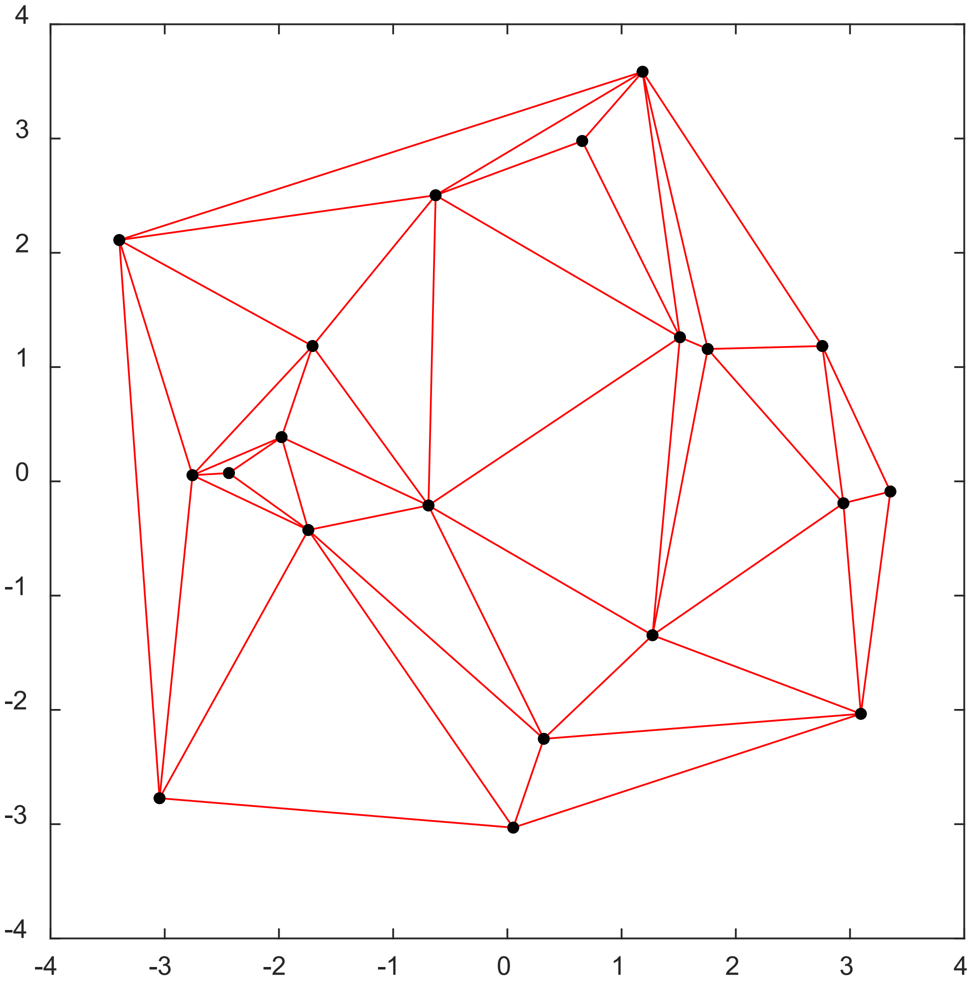
- Find the points that enclose all points
  - Most important data structure
  - Calculated, via Graham's scan in

    $O(n \log n), \quad n$ points



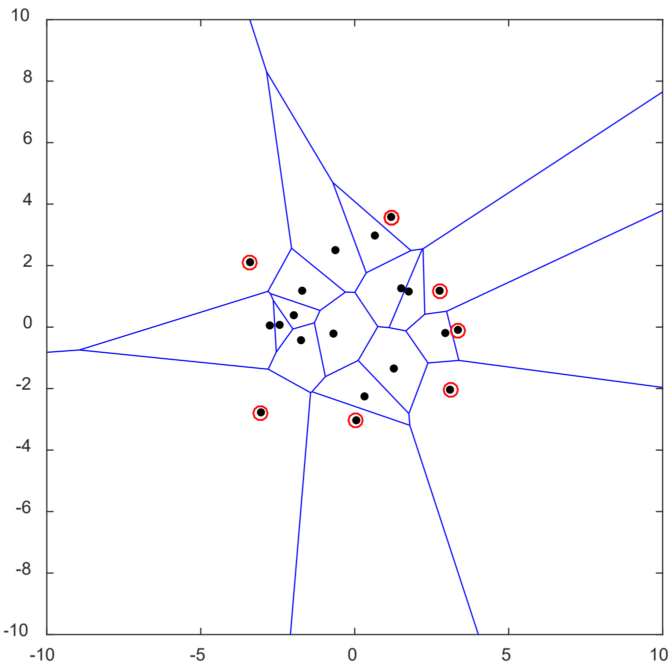
Delaunay Triangulation

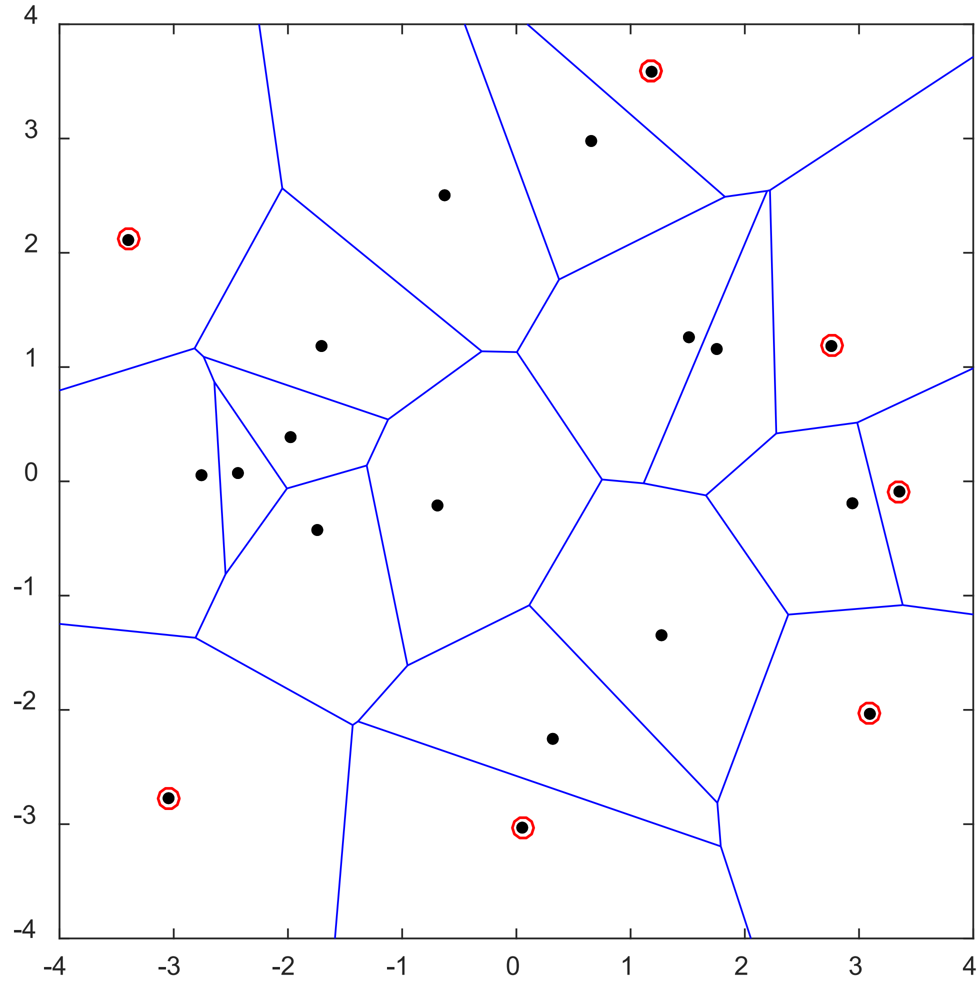
- Find the triangulation of points that maximizes the minimum angle of any triangle
  - Captures proximity relationships
  - Used in 3-D animation
  - Calculated, via divide and conquer, in $O(n \log n)$, n points



Voronoi Diagram

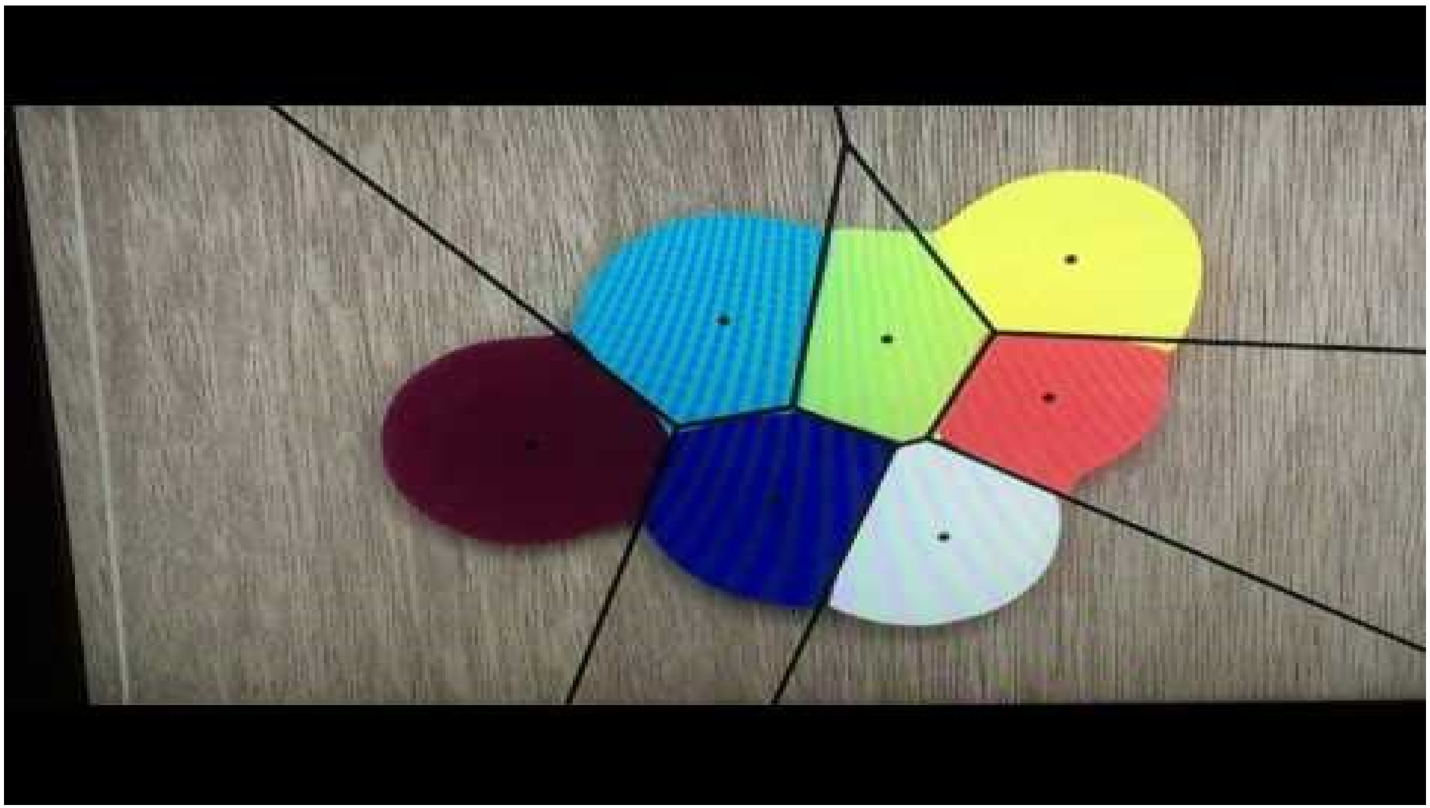
- Each region defines area closest to a point
  - Open face regions indicate points in convex hull





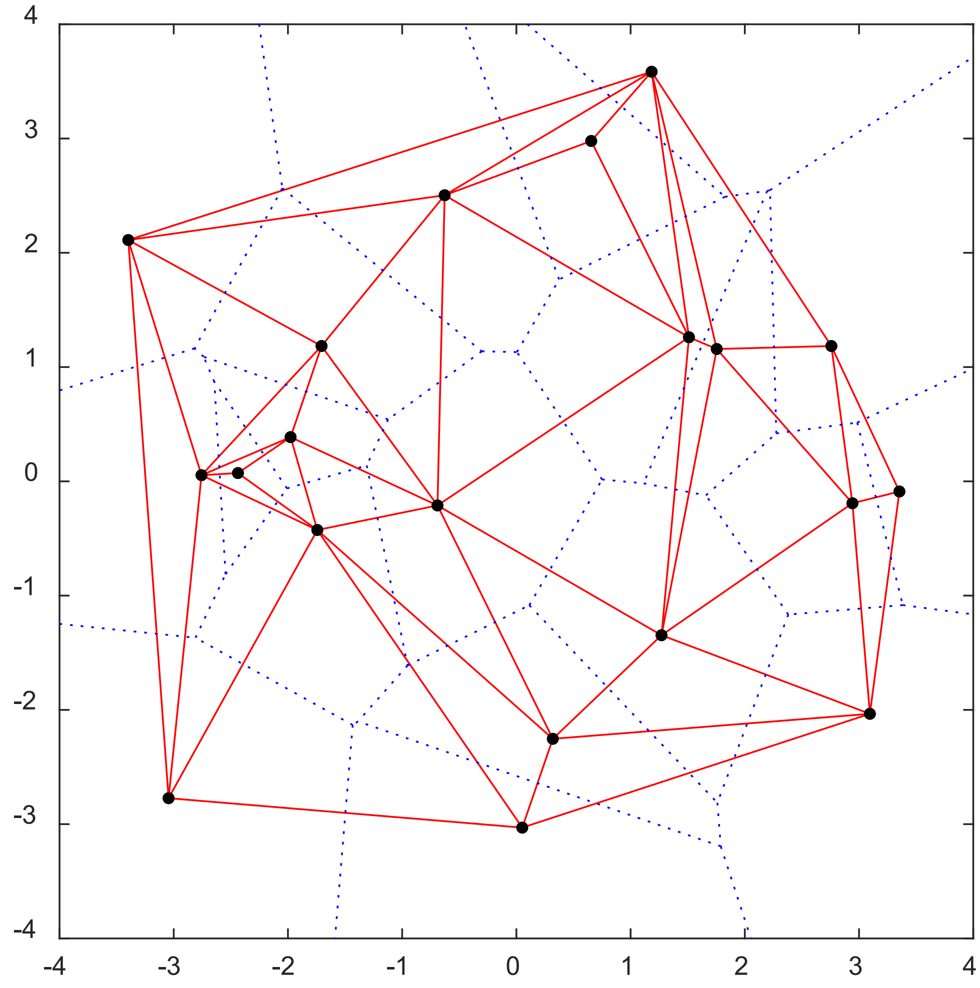
Voronoi Diagram

- Voronoi diagram from smooshing paint between glass
  - https://youtu.be/yDMtGT0b_kg



Delaunay-Voronoi

- Delaunay triangulation is straight-line dual of Voronoi diagram
  - Can easily convert from one to another

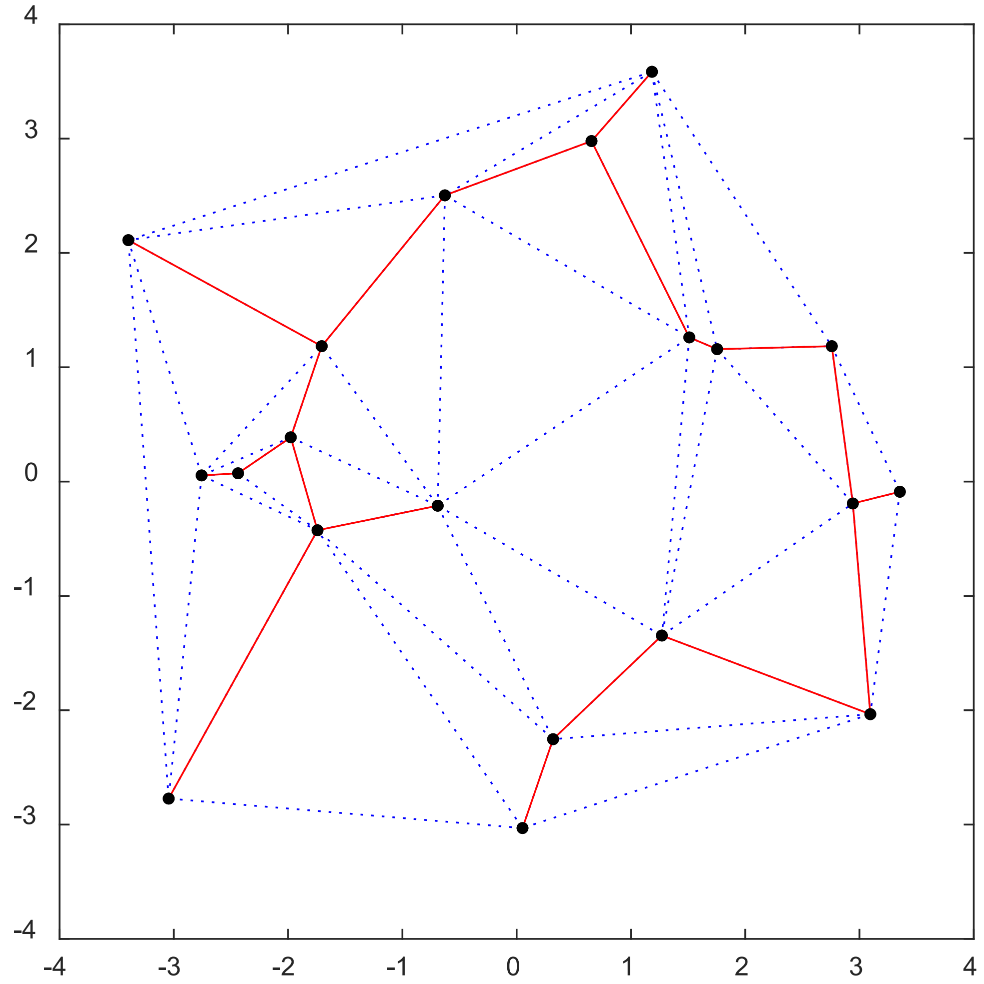


Minimum Spanning Tree

- Find the minimum weight set of arcs that connect all nodes in a graph
  - *Undirected* arcs: calculated, via Kruskal's algorithm,

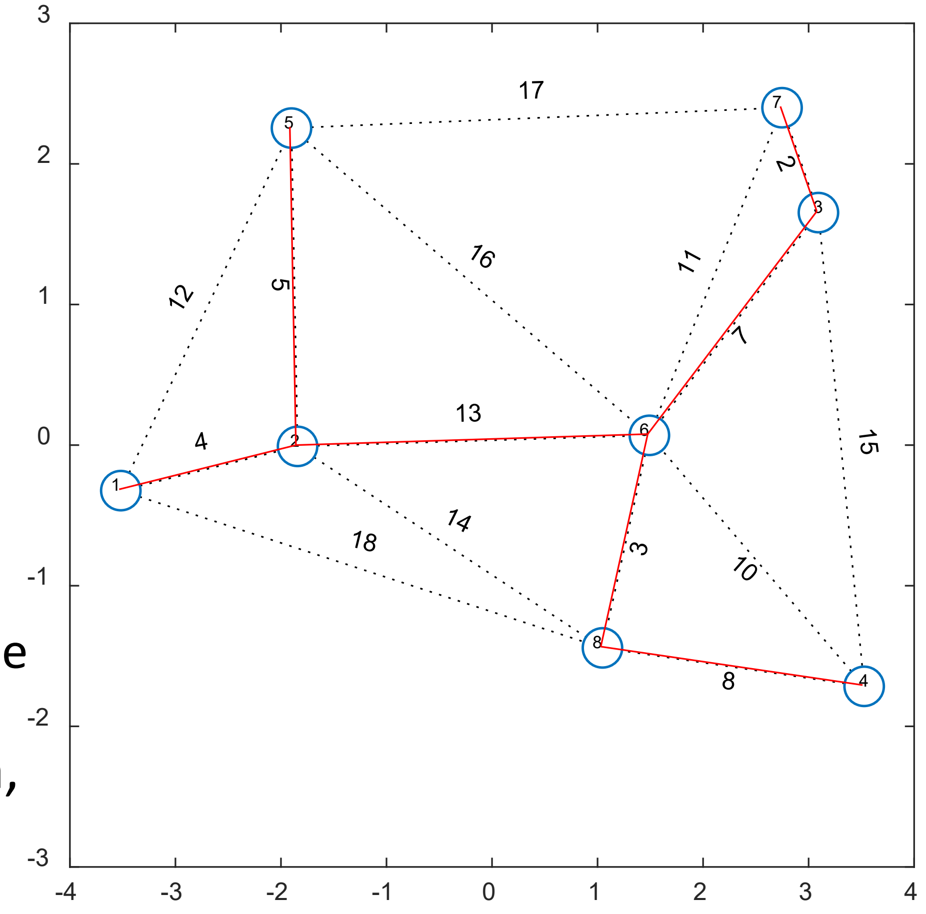
    $O(m \log n)$, m arcs, n nodes
  - *Directed* arcs: calculated, via Edmond's branching algorithm, in

    $O(mn)$, m arcs, n nodes



Kruskal's Algorithm for MST

- **Algorithm:**
  1. Create set *F* of single node trees
  2. Create set *S* of all arcs
  3. While *S* nonempty and *F* is not yet spanning
     4. Remove min arc from *S*
     5. If removed arc connects two different trees, then add to *F*, combining two trees into single tree
  6. If graph connected, *F* forms single MST; otherwise, forms multi-tree min spanning forest
- Optimal "greedy" algorithm, runs in $O(m \log n)$
- If directed arcs, $O(m\,n)$
  - useful in VRP to min vehicles
  - harder to code



m = 15 arcs, *n* = 8 nodes

Min Spanning vs Steiner Trees

- Steiner point added to reduce distance connecting three existing points compared to min spanning tree

$$\frac{b}{2} = \frac{1}{2}\sqrt{3}a \Rightarrow b = \sqrt{3}a, \quad 30-60-90 \text{ triangle}$$

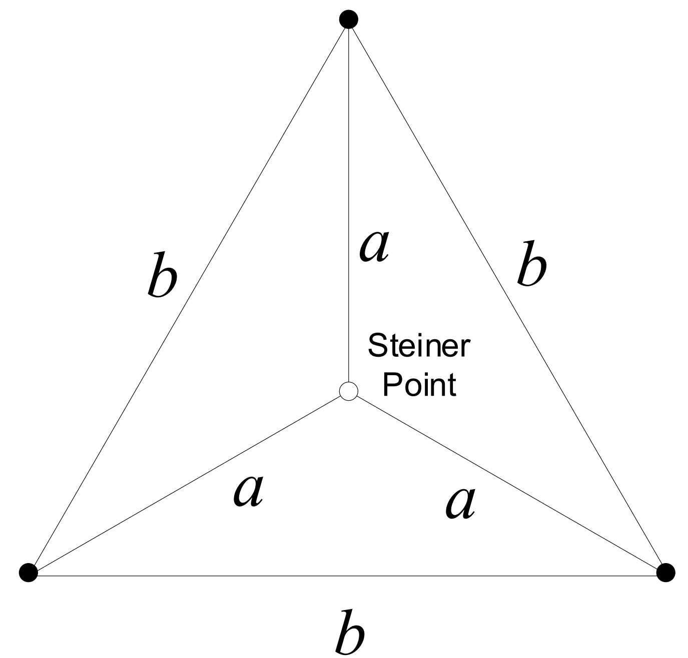
Min spanning tree distance > Steiner tree distance

$$2b > 3a$$

$$2\sqrt{3}a > 3a$$

$$2 > \sqrt{3}$$

$$\sqrt{4} > \sqrt{3}$$



Steiner Network

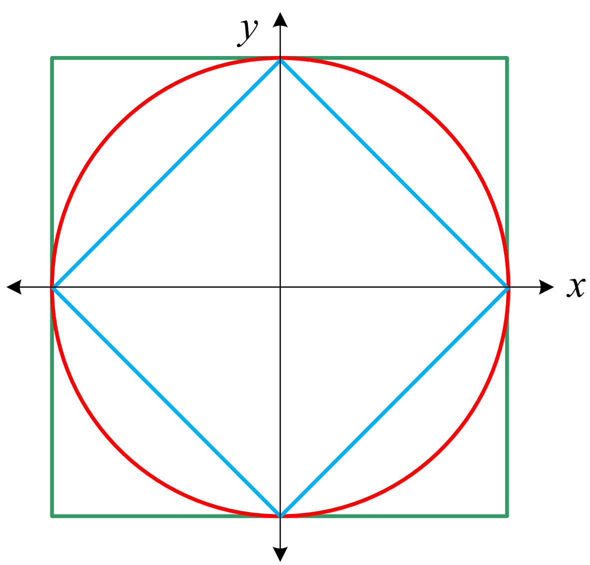
Metric Distances

General l_p: $$d_p(P_1, P_2) = \left[|x_1 - x_2|^p + |y_1 - y_2|^p \right]^{\frac{1}{p}}, \quad p \geq 1$$

Rectilinear $(p=1)$: $$d_1(P_1, P_2) = |x_1 - x_2| + |y_1 - y_2|$$

Euclidean $(p=2)$: $$d_2(P_1, P_2) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

Chebychev $(p \to \infty)$: $$d_\infty(P_1, P_2) = \max\{|x_1 - x_2|, |y_1 - y_2|\}$$



Chebychev Distances

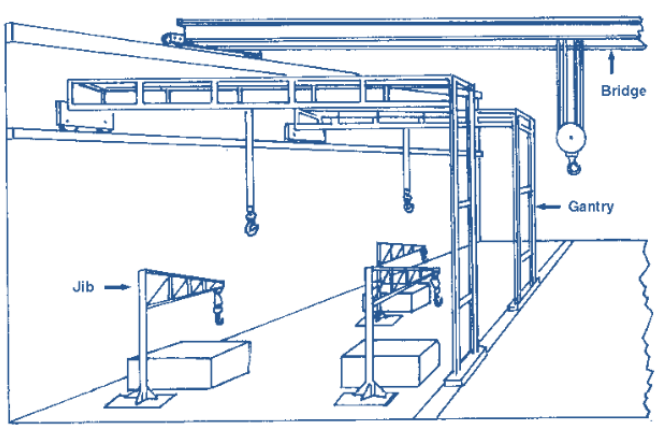
Proof

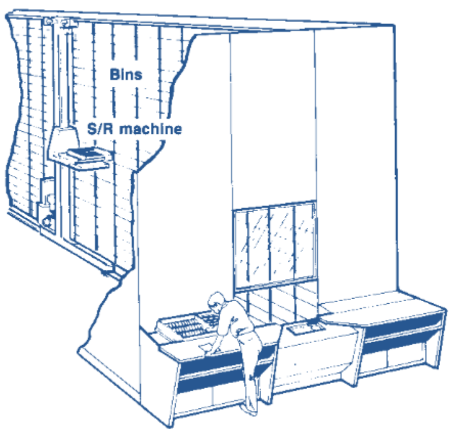
Without loss of generality, let $P_1 = (x, y)$, for $x, y \geq 0$, and $P_2 = (0, 0)$. Then $d_\infty(P_1, P_2) = \max\{x, y\}$ and $d_p(P_1, P_2) = \left[x^p + y^p\right]^{1/p}$.

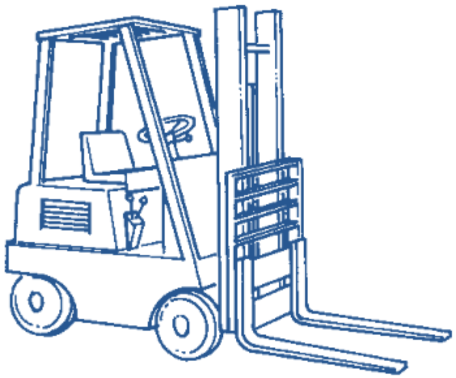
If $x = y$, then $\lim_{p\to\infty}\left[x^p + y^p\right]^{1/p} = \lim_{p\to\infty}\left[2x^p\right]^{1/p} = \lim_{p\to\infty}\left[2^{1/p}x\right] = x$.

If $x < y$, then $\lim_{p\to\infty}\left[x^p + y^p\right]^{1/p} = \lim_{p\to\infty}\left[\left((x/y)^p + 1\right)y^p\right]^{1/p} = \lim_{p\to\infty}\left((x/y)^p + 1\right)^{1/p} y = 1 \cdot y = y$.

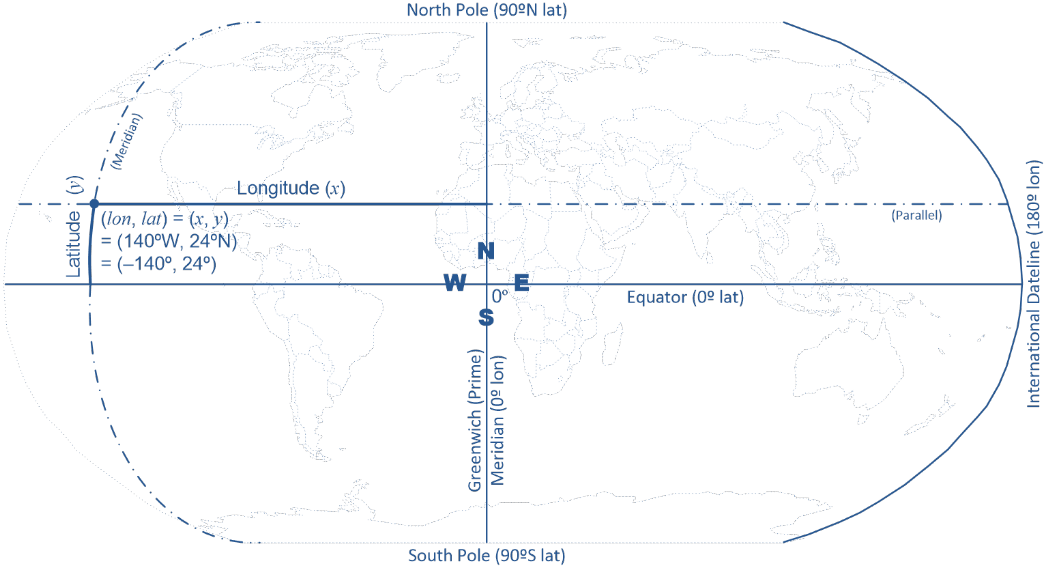
A similar argument can be made if $x > y$. ■

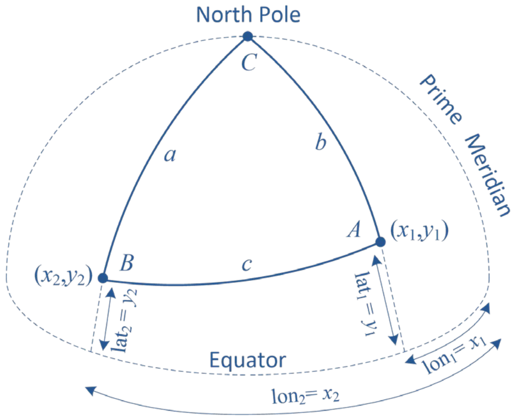


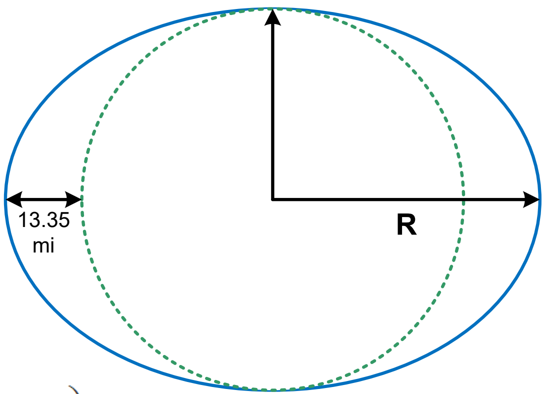




Great Circle Distances







$$\left(lon_1, lat_1\right) = \left(x_1, y_1\right), \quad \left(lon_2, lat_2\right) = \left(x_2, y_2\right)$$

$$d_{rad} = \left(\text{great circle distance in radians of a sphere}\right)$$

$$= \cos^{-1}\left[\sin y_2 \sin y_1 + \cos y_2 \cos y_1 \cos\left(x_1 - x_2\right)\right]$$

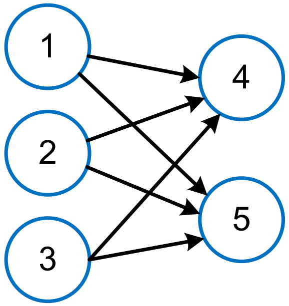
$$R = (\text{radius of earth at equator}) - (\text{bulge from north pole to equator})$$

$$= 3{,}963.34 - 13.35 \sin\left(\frac{y_1 + y_2}{2}\right) \text{ mi}, \quad = 6{,}378.388 - 21.476 \sin\left(\frac{y_1 + y_2}{2}\right) \text{ km}$$

$$d_{\text{GC}} = \text{distance } \left(x_1, y_1\right) \text{ to } \left(x_2, y_2\right) = \boxed{d_{rad} \cdot R}$$

$$x_{\deg} = \begin{cases} DD + \dfrac{MM}{60} + \dfrac{SS}{3{,}600}, & \text{if } E \text{ or } N \\ -DD - \dfrac{MM}{60} - \dfrac{SS}{3{,}600}, & \text{if } W \text{ or } S \end{cases}$$

Metric Distances using `dists`



$$\mathbf{D} = \begin{array}{c|cc} & 4 & 5 \\ \hline 1 & \bullet & \bullet \\ 2 & \bullet & \bullet \\ 3 & \bullet & \bullet \end{array}_{\substack{3\times 2 \\ n\times m}} = \text{dists}(\underset{\substack{3\times 2\\ n\times d}}{\mathbf{X1}}, \underset{\substack{2\times 2\\ m\times d}}{\mathbf{X2}}, p), \quad p = \begin{cases} \text{'mi'} & \text{'km'} \\ 1 & 2 \quad \text{Inf} \end{cases}$$

_d_ = 2

$$\mathbf{X1} = \begin{bmatrix} \bullet & \bullet \end{bmatrix}, \mathbf{X2} = \begin{bmatrix} \bullet & \bullet \end{bmatrix} \Rightarrow \mathbf{d} = \begin{bmatrix} \bullet \end{bmatrix}$$

$$\mathbf{X1} = \begin{bmatrix} \bullet & \bullet \end{bmatrix}, \mathbf{X2} = \begin{bmatrix} \bullet & \bullet \\ \bullet & \bullet \\ \bullet & \bullet \end{bmatrix} \Rightarrow \mathbf{d} = \begin{bmatrix} \bullet & \bullet & \bullet \end{bmatrix}$$

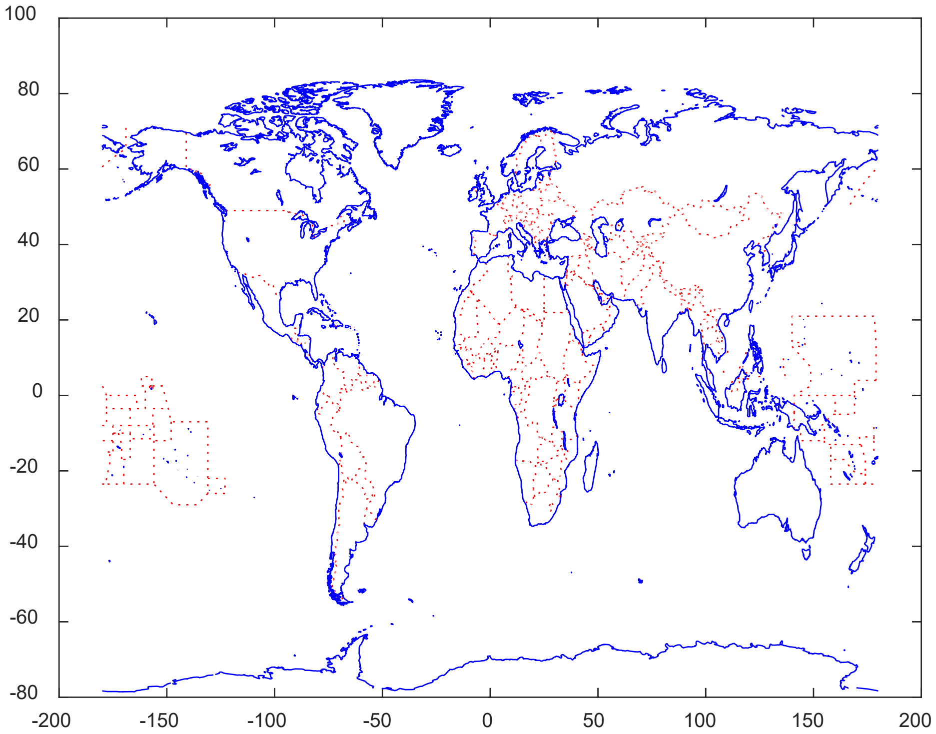
$$\mathbf{X1} = \begin{bmatrix} \bullet & \bullet \\ \bullet & \bullet \\ \bullet & \bullet \end{bmatrix}, \mathbf{X2} = \begin{bmatrix} \bullet & \bullet \\ \bullet & \bullet \\ \bullet & \bullet \end{bmatrix} \Rightarrow \mathbf{D} = \begin{bmatrix} \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet \end{bmatrix}$$

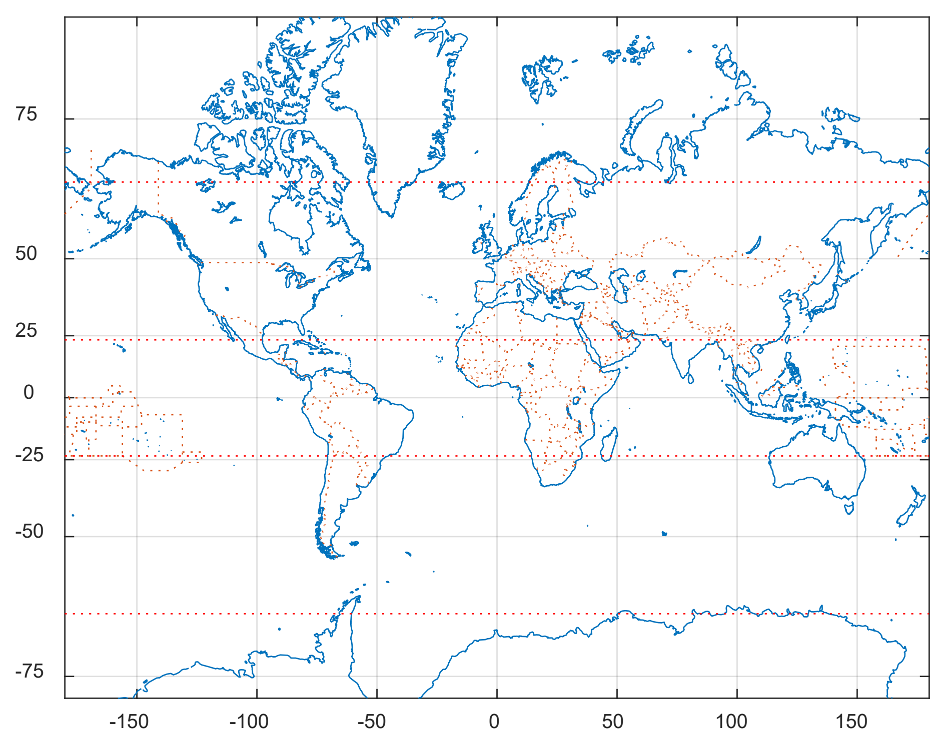
_d_ = 1

$$\mathbf{X1} = \begin{bmatrix} \bullet \\ \bullet \end{bmatrix}, \mathbf{X2} = \begin{bmatrix} \bullet \\ \bullet \end{bmatrix} \Rightarrow \mathbf{D} = \begin{bmatrix} \bullet & \bullet \\ \bullet & \bullet \end{bmatrix}$$

$$\mathbf{X1} = \begin{bmatrix} \bullet \\ \bullet \end{bmatrix}, \mathbf{X2} = \begin{bmatrix} \bullet & \bullet \end{bmatrix} \Rightarrow \textit{Error}$$

Mercator Projection





$$x_{\text{rad}} = \frac{x_{\text{deg}}}{180}\pi \quad \text{and} \quad x_{\text{deg}} = \frac{x_{\text{rad}} \cdot 180}{\pi}$$

$$x_{\text{proj}} = x$$

$$y_{\text{proj}} = \sinh^{-1}\left(\tan y\right)$$

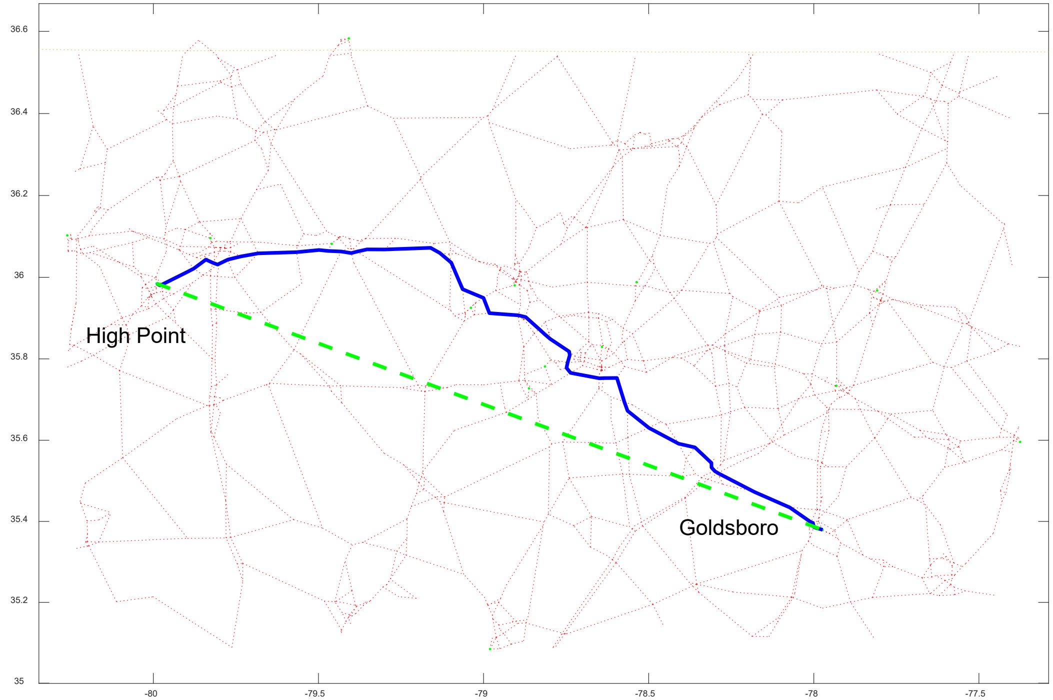
$$y = \tan^{-1}\left(\sinh y_{\text{proj}}\right)$$

Circuity Factor

$$\textit{Circuity Factor}: \quad g = \sum v_i \frac{d_{\text{road}_i}}{d_{GC_i}}, \quad \text{where usually } 1.15 \le g \le 1.5,\ v_i \text{ weight of sample } i$$

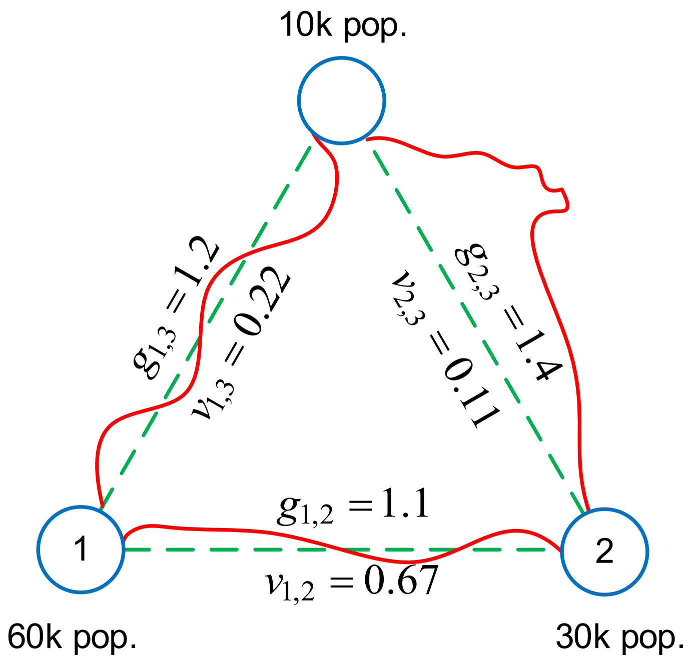
$$d_{\text{road}} \approx g \cdot d_{GC}(P_1, P_2), \quad \text{estimated road distance from } P_1 \text{ to } P_2$$

From High Point to Goldsboro: Road = 143 mi, Great Circle = 121 mi, Circuity = 1.19



Estimating Circuity Factors

- Circuity factor depends on both the trip density and directness of travel network
  - Circuity of high trip density areas should be given more weight when estimating overall factor for a region
  - Obstacles (water, mountains) limit direct road travel




```
v = [.6 .3 .1];
V = v'*v
  = 0.3600    0.1800    0.0600
    0.1800    0.0900    0.0300
    0.0600    0.0300    0.0100
V = triu(V,1)
  =      0    0.1800    0.0600
         0         0    0.0300
         0         0         0
V = V/sum(sum(V))
  =      0    0.6667    0.2222
         0         0    0.1111
         0         0         0
```