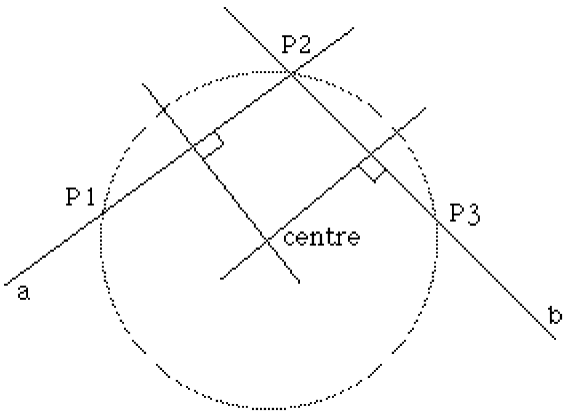
Equation of a Circle from 3 Points (2 dimensions)

Written by Paul Bourke
January 1990

See also: Sphere from 4 points

This note describes a technique for determining the attributes of a circle (centre and radius) given three points $\mathbf{P_1}$, $\mathbf{P_2}$, and $\mathbf{P_3}$ on a plane.



Calculating Centre

Two lines can be formed through 2 pairs of the three points, the first passes through the first two points $\mathbf{P_1}$ and $\mathbf{P_2}$. Line b passes through the next two points $\mathbf{P_2}$ and $\mathbf{P_3}$.
The equation of these two lines is

$$y_a = m_a (x - x_1) + y_1 \quad \text{and} \quad y_b = m_b (x - x_2) + y_2$$

where m is the slope of the line given by

$$m_a = \frac{y_2 - y_1}{x_2 - x_1} \quad \text{and} \quad m_b = \frac{y_3 - y_2}{x_3 - x_2}$$

The centre of the circle is the intersection of the two lines perpendicular to and passing through the midpoints of the lines $\mathbf{P_1P_2}$ and $\mathbf{P_2}$ $\mathbf{P_3}$. The perpendicular of a line with slope m has slope -1/m, thus equations of the lines perpendicular to lines a and b and passing through the midpoints of $\mathbf{P_1P_2}$ and $\mathbf{P_2P_3}$ are

$$y'_a = -\frac{1}{m_a}\left(x - \frac{x_1 + x_2}{2}\right) + \frac{y_1 + y_2}{2}$$
$$y'_b = -\frac{1}{m_b}\left(x - \frac{x_2 + x_3}{2}\right) + \frac{y_2 + y_3}{2}$$

These two lines intersect at the centre, solving for x gives

$$x = \frac{m_a m_b (y_1 - y_3) + m_b (x_1 + x_2) - m_a (x_2 + x_3)}{2(m_b - m_a)}$$

Calculate the y value of the centre by substituting the x value into one of the equations of the perpendiculars. Alternatively one can also rearrange the equations of the perpendiculars and solve for y.

Radius

The radius is easy, for example the point $\mathbf{P_1}$ lies on the circle and we know the centre....

Notes:

- The denominator (mb - ma) is only zero when the lines are parallel in which case they must be coincident and thus no circle results.
- If either line is vertical then the corresponding slope is infinite. This can be solved by simply rearranging the order of the points so that vertical lines do not occur.

Source Code

C++ code implemented as MFC (MS Foundation Class) supplied by Jae Hun Ryu. Circle.cpp, Circle.h.