Software/Hardware Integration

Localization for Wireless Sensor Networks

Rafael Pereira Pires

rafaelpp@lisha.ufsc.br

http://www.lisha.ufsc.br/~rafaelpp

April 10, 2007

Software/Hardware Integration

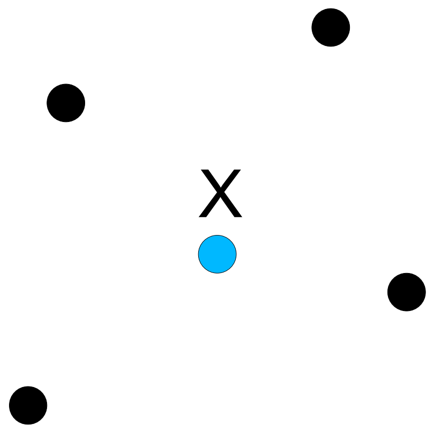
Localization Techniques

- Range-free Methods
  - Depends on a good geographical position of the anchor nodes
- Anchor Nodes
  - Nodes that are aware of their positions
- Range Measurements
  - Method to estimate the distance or angle between a pair of nodes

Software/Hardware Integration

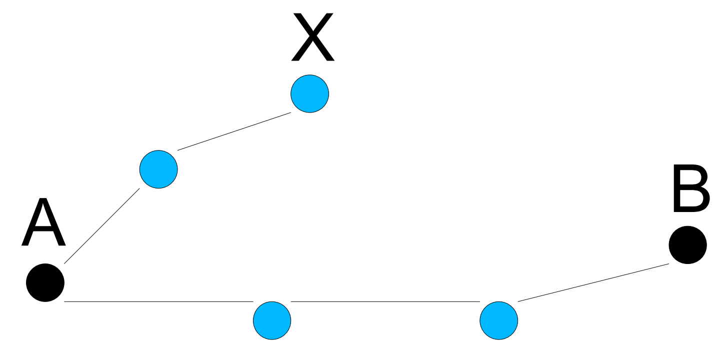
Range-free Methods

- Centroid



$$(x,y) = \left(\frac{x1+...+xn}{n}, \frac{y1+...+yn}{n}\right)$$

- DV-HOP



$d_{AB} = 60m \Rightarrow d_{AX} = 40m$

Range Measurements

- ToA (Time of Arrival)
  - Requires accurate clocks
- TDoA (Time Difference of Arrival)
  - Requires additional hardware (ultra-sound device)
- AoA (Angle of Arrival)
  - Requires additional hardware (for angle estimation)
- RSSI (Received Signal Strength Indication)
  - No additional hardware
  - Imprecise

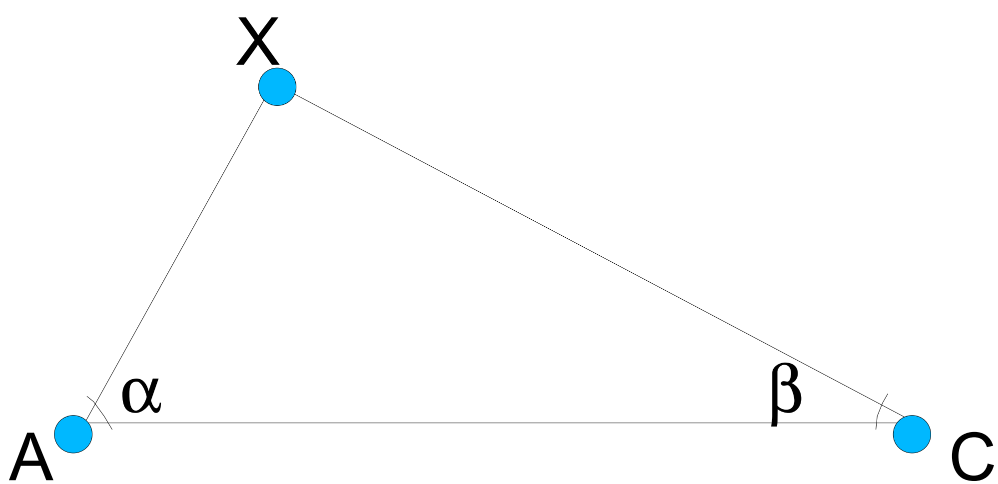
Software/Hardware Integration

Position Calculation

- Triangulation
- Trilateration
- Multilateration

Triangulation

- Angle information is provided.
  - Coordinates can be defined through trigonometry rules (law of sines, law of cosines, trigonometric functions).

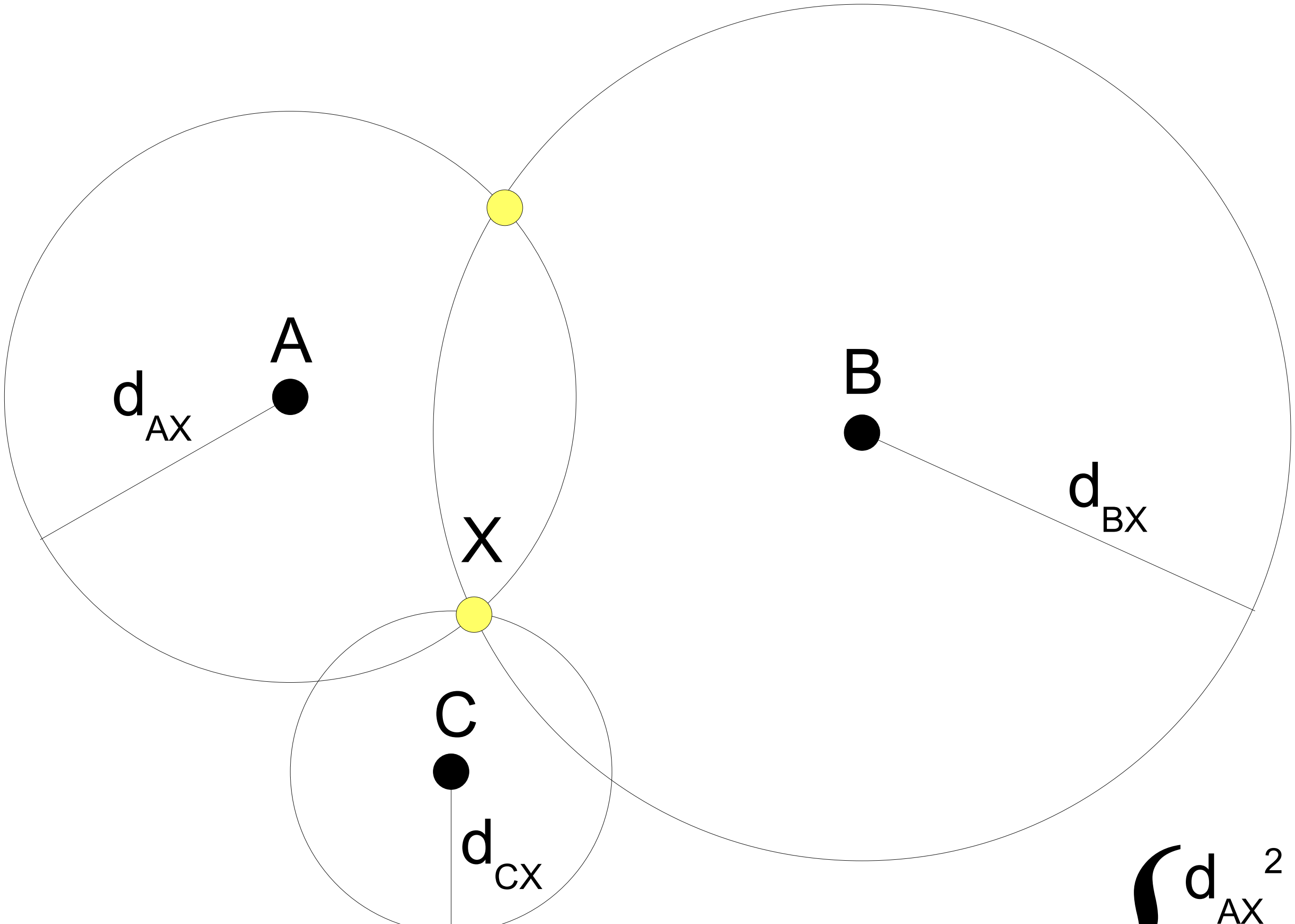


$m = tg\ \alpha \qquad n = tg\ \beta$

$$\begin{cases} y = m\,(x - x_A) + y_A \\ y = n\,(x - x_B) + y_B \end{cases}$$

Software/Hardware Integration

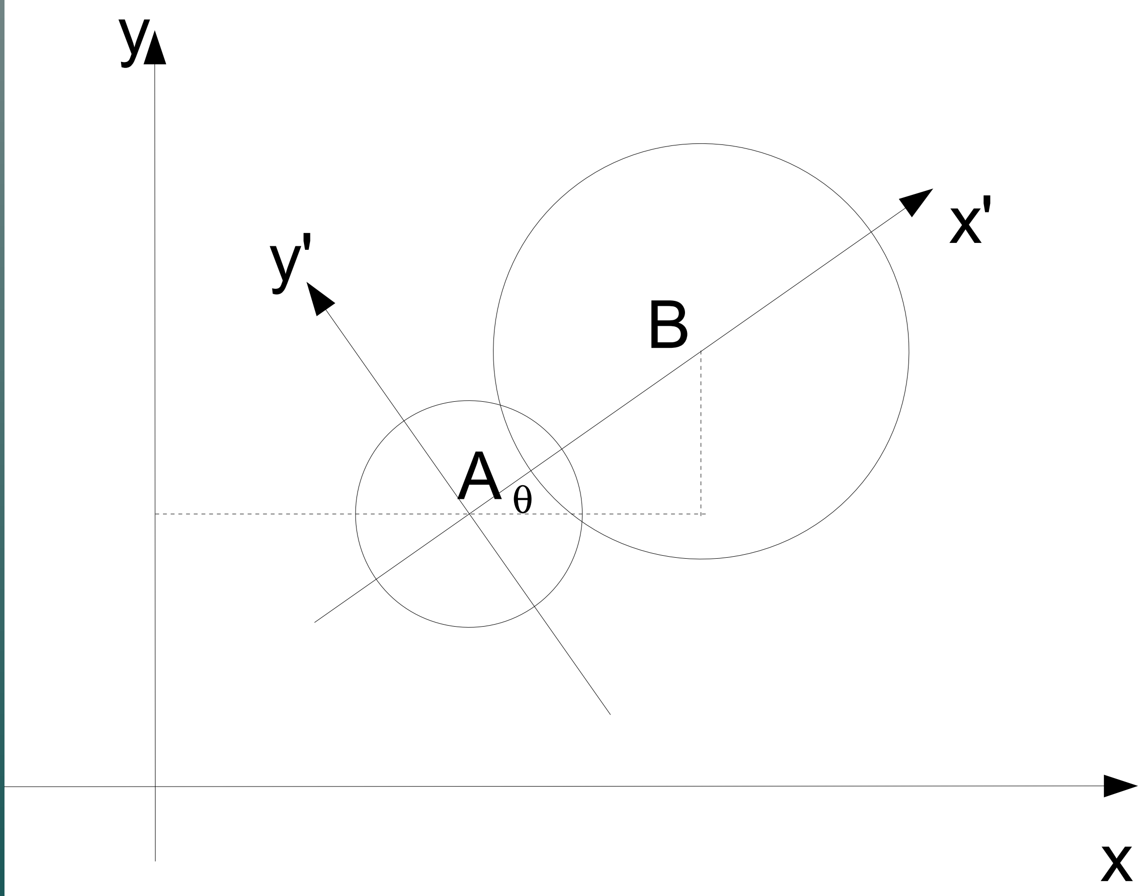
Trilateration



$$\begin{cases} d_{AX}^{2} = (x-x_{A})^{2} + (y-y_{A})^{2} \\ d_{BX}^{2} = (x-x_{B})^{2} + (y-y_{B})^{2} \\ d_{CX}^{2} = (x-x_{C})^{2} + (y-y_{C})^{2} \end{cases}$$

Software/Hardware Integration

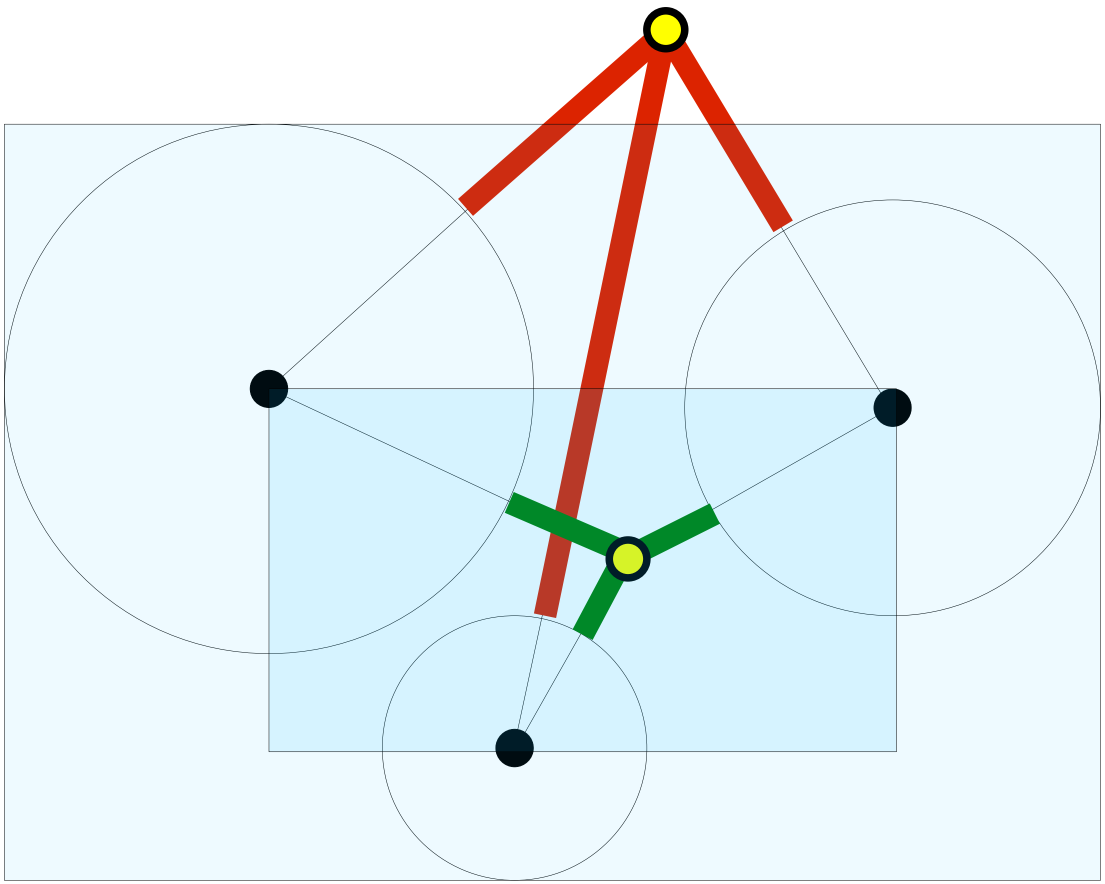
Translation and Rotation



$$x' = x \cos\theta - y \sin\theta$$
$$y' = x \sin\theta + y \cos\theta$$

$$\sin\theta = (y_B - y_A) / d_{AB}$$
$$\cos\theta = (x_B - x_A) / d_{AB}$$

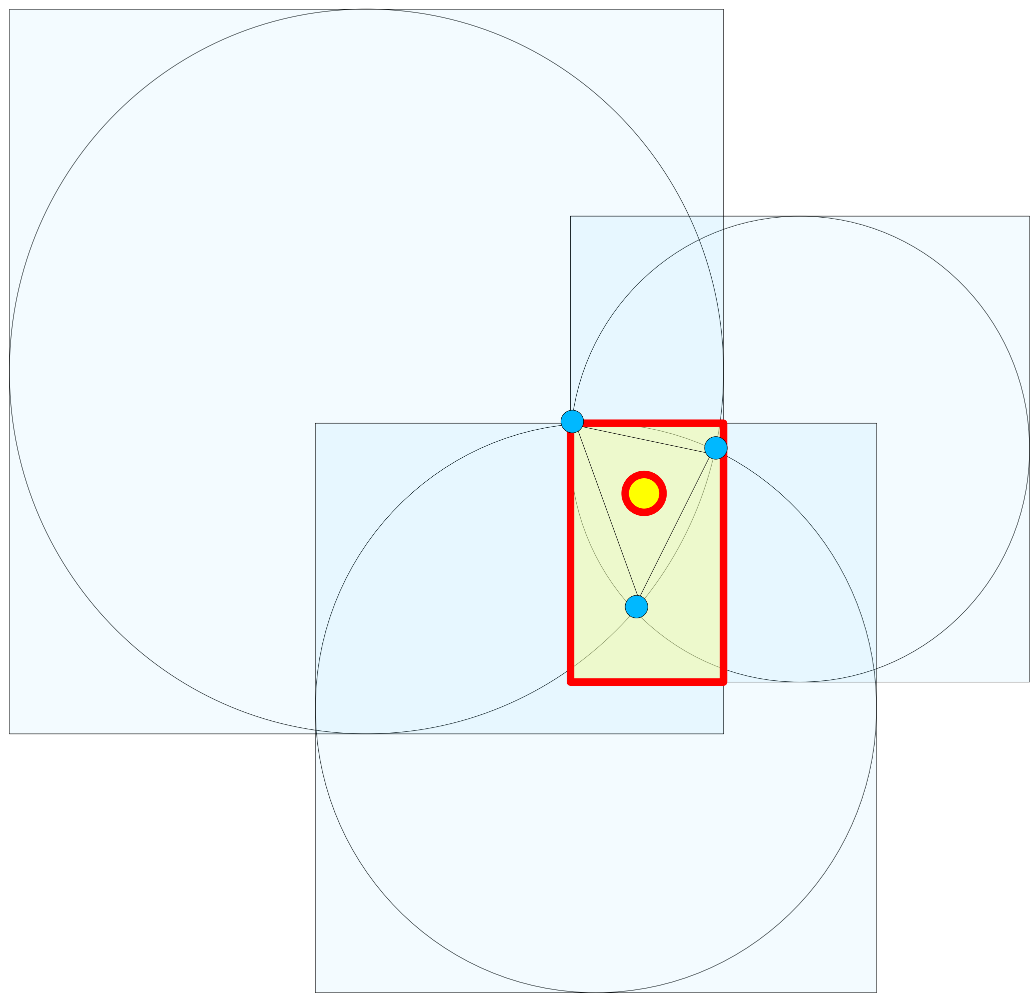
Approximation

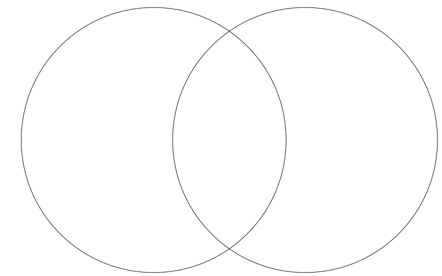


$$\text{Residue} = \sum_{i=1}^{n} |d_{iX} - d_X|$$

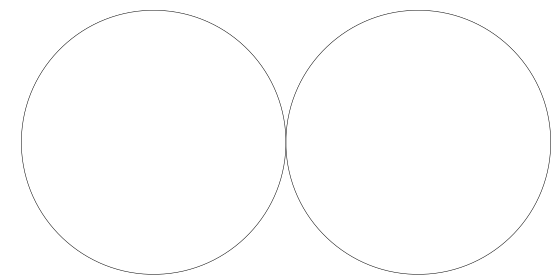
Where i iterates through anchors (and n is the total number)

Min-Max

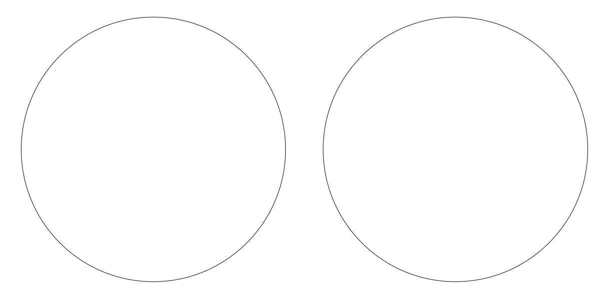




$$d_{AB} < r_A + r_B$$



$$d_{AB} == r_A + r_B$$



$$d_{AB} > r_A + r_B$$

Multilateration

$$\begin{cases} (x_1 - x)^2 + (y_1 - y)^2 = d_1^2 \\ \vdots \\ (x_n - x)^2 + (y_n - y)^2 = d_n^2 \end{cases}$$

The system can be linearized by subtracting the last equation from the first *n-1* equations. Reordering the terms gives a proper system of linear equations in the form of *Ax* = *b*, where:

$$A = \begin{bmatrix} 2(x_1 - x_n) & 2(y_1 - y_n) \\ \vdots & \vdots \\ 2(x_{n-1} - x_n) & 2(y_{n-1} - y_n) \end{bmatrix}$$

$$b = \begin{bmatrix} x_1^2 - x_n^2 + y_1^2 - y_n^2 + d_n^2 - d_1^2 \\ \vdots \\ x_{n-1}^2 - x_n^2 + y_{n-1}^2 - y_n^2 + d_n^2 - d_{n-1}^2 \end{bmatrix}$$

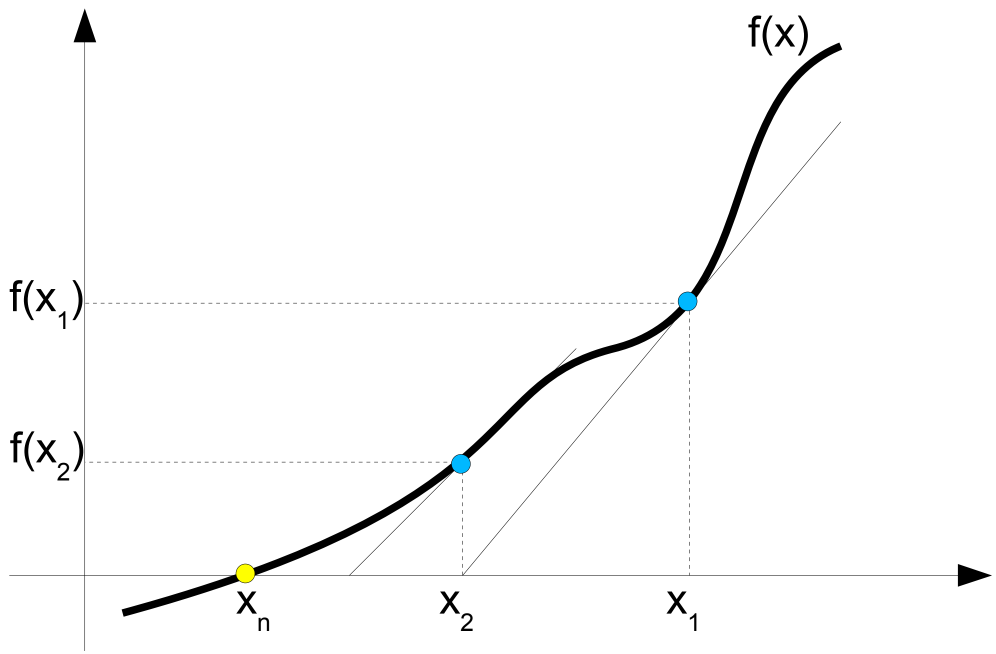
Using a standard least-squares approach:

$$\hat{x} = (A^T A)^{-1} A^T b$$

Software/Hardware Integration

Square-root Approximation

- Newton-Raphson method to find roots of functions

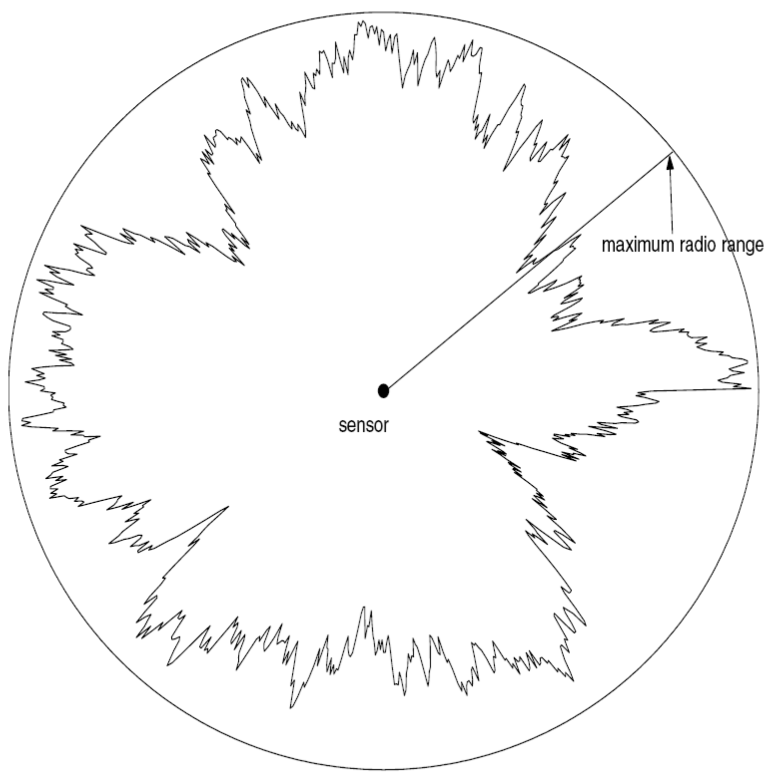


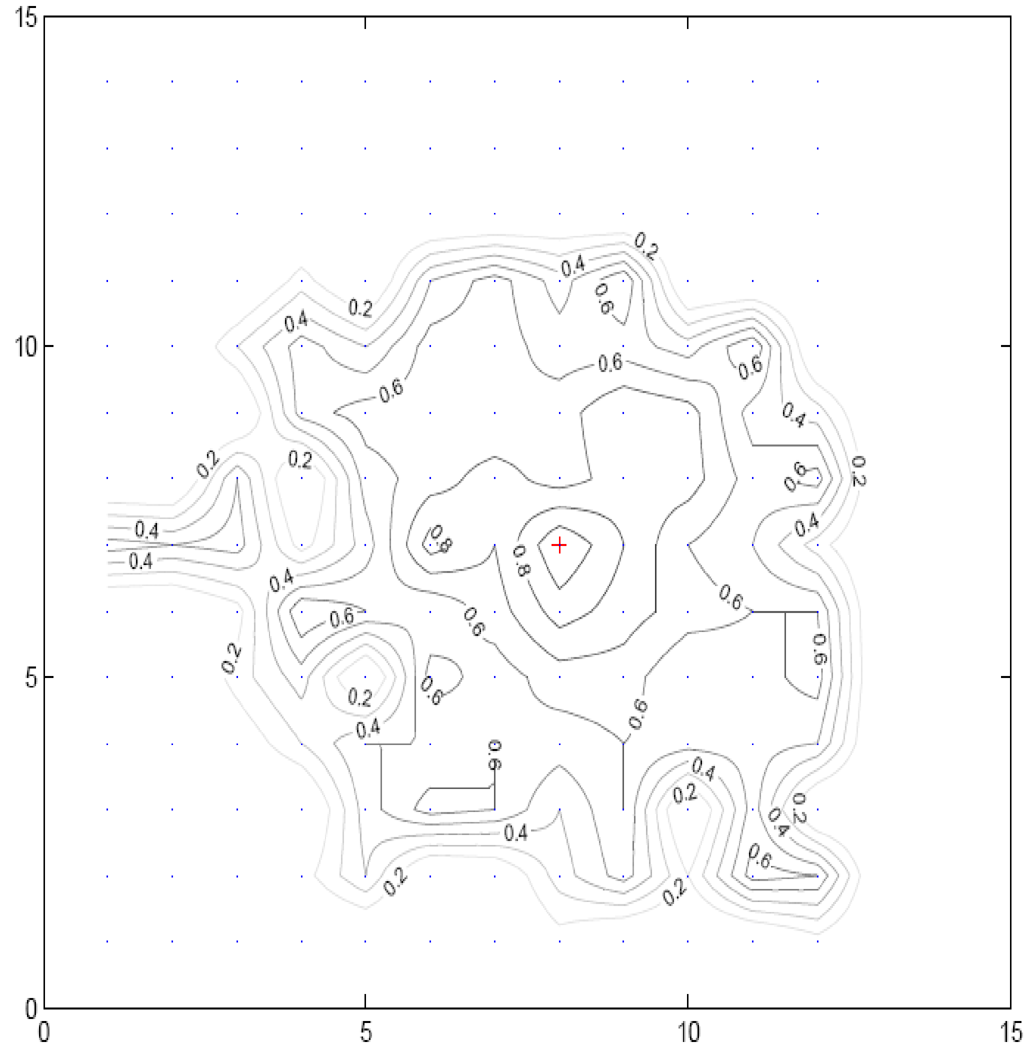
$$f'(x_n) = f(x_n) / (x_n - x_{n+1})$$

$$x_{n+1} = x_n - f(x_n) / f'(x_n)$$

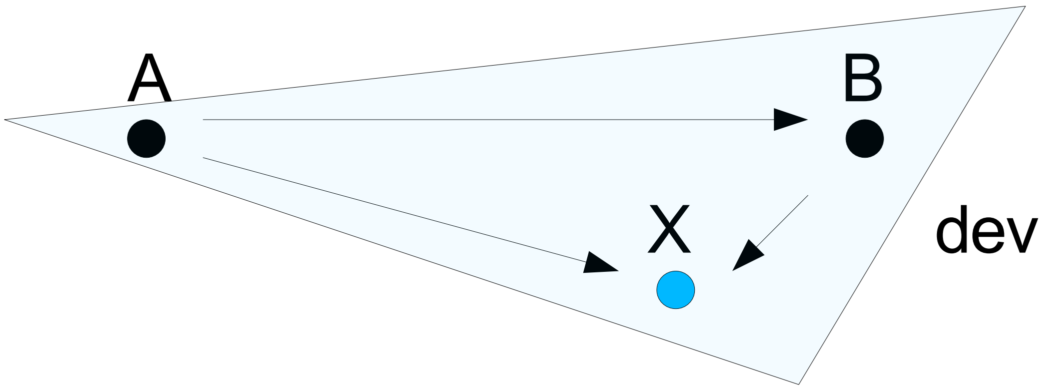
- Which function gives the square-root of a number?

RSSI Fluctuations





HECOPS Calibration



$$deviation_{AB} = d_{AB} \;/\; RSSI_{AB}$$

$$d_{AX} = deviation_{AB} \,.\, RSSI_{AX}$$