CS634 : Group 5

Localization of Sensors

Rahul Ajmera Gaurav Sharma
Y9227451 Y9229

April 11, 2013

INTRODUCTION

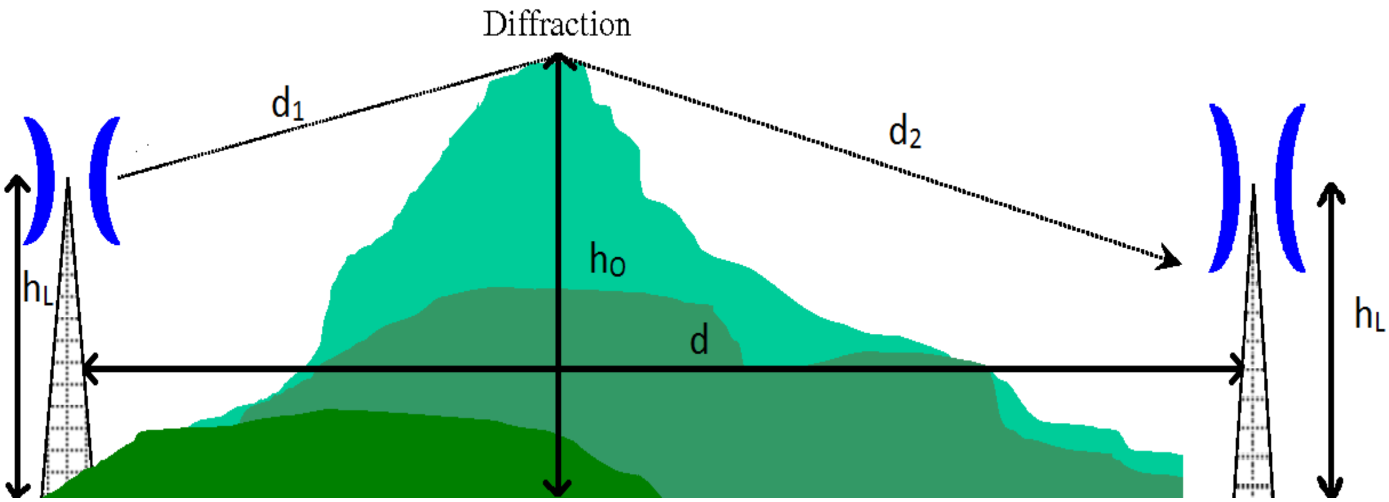
With advancement in technology we need sensor networks to be working cost efficiently and accurately. One of the major requirements of any application using the sensors in the network is to accurately locate a node. Since we want to make our systems cheap and reliable we can't use expensive GPS like systems in built in the node, rather we would want to make a coordinative system which would behave as a self-organizable sensor network. In past years we have developed many efficient and cheaper algorithms to locate sensors in a network. Our aim is to study, simulate and compare such algorithms.

PROBLEM STATEMENT

Given a sensor network consisting of N sensors at locations S = $\{S_1, S_2, \ldots\ldots, S_N\}$. Let Sx_i refer to the x-coordinate of the location of sensor i and let Sy_i refer to the y coordinate. Determining these locations constitutes the localization problem. Some sensor nodes are aware of their own positions, these nodes are known as **anchors** or **beacons**. All the other nodes localize themselves with the help of location references received from the anchors. So, mathematically the localization problem can be formulated as follows: given a multihop network, represented by a graph G = (V, E), and a set of beacon nodes B, their positions $\{x_b, y_b\}$ for all b ε B, we want to find the position $\{x_u, y_u\}$ for all unknown nodes u ε U.

RADIO PROPAGATION MODEL

Radio Propagation Model is a mathematical formulation for signal strength of radio wave as a function of frequency, distance, obstacles and other conditions.The ITU Terrain Model is a radio propagation model that provides a method to predict the median path loss for a telecommunication link. This method is developed on the basis of diffraction theory, where diffraction is caused by obstacles in the surroundings. This model predicts the path loss as a function of the height of the obstacle and the First Fresnel zone for the transmission link, which creates signal with in-phase strength. This model was used due to its genericity and its ability to work for all frequency and distances. Hence, this model for all terrains, cities or open fields.



Mathematical Formula:

$A=10-20C_N$

$C_N=h/F_1$

$h=h_L-h_O$

$F_1=17.3\sqrt{\frac{d1*d2}{fd}}$ where, f is frequency and A is additional loss due to diffraction from one obstacle.The losses due to all obstacles are summed with Friis transmission equation loss to get overall signal strength.

ALGORITHMS

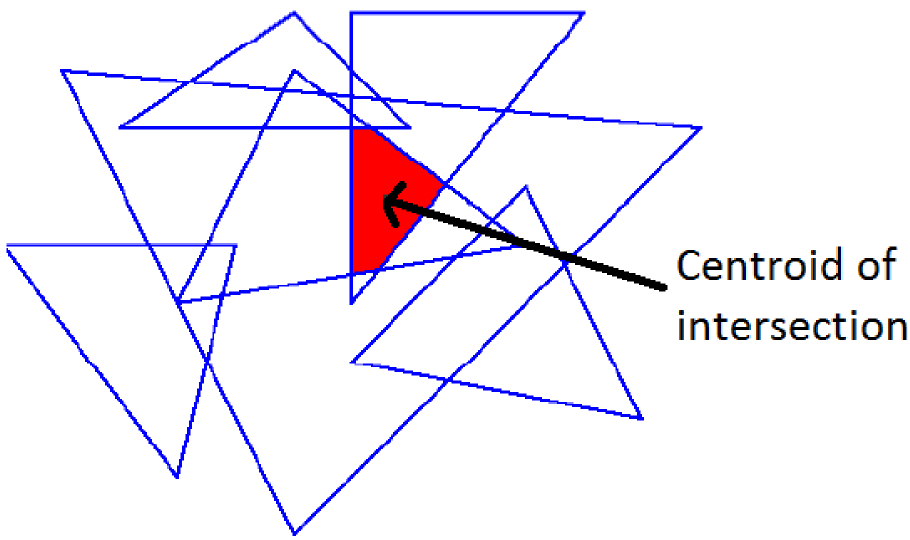
There are seven algorithms taken to be simulated:-

1. Diffusion: The basic concept used here is a node is expected to be located at the centroid of its one-hop neighbors. It is a very simple algorithm in which location of a node is estimated repeatedly by averaging the position of its peers until it converges to a steady state or a predefined number of iterations have occurred. This algorithm works perfectly for high dense node network and uniformly dense networks, but the algorithm is less accurate and precise for low dense networks because of the simplicity of the algorithm.

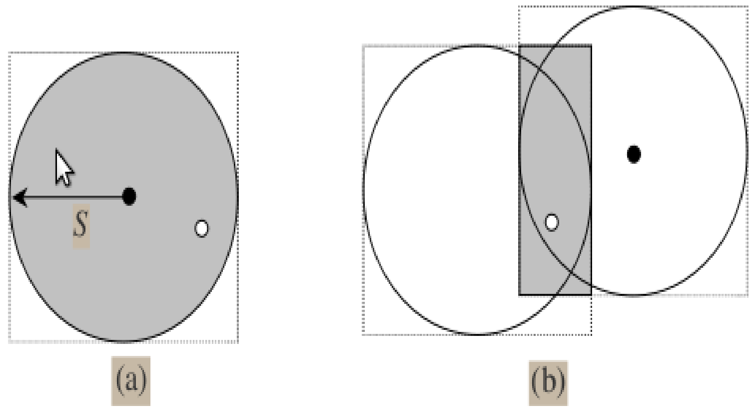
2. Gradient: This algorithm works on the principle of Multilateration (navigation technique which is based on differences in distance to 2+ nodes at known locations), which helps in finding the most approximate value of the location of unknown node. For all beacons a bi-connected graph is created in which there is an edge between every node and it's one-hop neighbor. Dijkstra's Algorithm is run to get the minimum number of hops required to reach a node from every beacon. One-hop distance of each beacon is calculated by averaging the actual distance between other beacons and number of hops required to reach those beacons. Finally, for every normal node, distance is calculated

from each beacon by multiplying the number of hops and the one-hop distance. Now, for each normal node the error function $|\Sigma(x-x_i)^2+\Sigma(y-y_i)^2-\Sigma d_i^2|$ (where, (x_i,y_i) is the location of $beacon_i$ and d_i is the calculated distance of the node from $beacon_i$), is minimized for two variables x,y to get the approximated location .This algorithm is most effective for homogeneous networks with high node density.

3. APIT(Approximate Point-In-Triangulation): This algorithm works on the principle of variation of received signal strength from beacons over varied distances. The node is assumed to be listening signal from all beacons and forming beacon triangles. Then based on the signal strength, it determines the triangles in which the node is located and reports the centroid of the intersection of all such possible beacon triangles as shown in figure below. This algorithm is more accurate as compared to other beacon based algorithms.



4. Moving Sensor: A beacon is randomly walked in the grid, and is broadcasting it’s new coordinates continuously to nodes, in it’s node range.Every time a node senses the beacon, it generates a new quadratic constraint that it uses to further reduce the uncertainty in its position. As shown in the figure below-
(a) A node, sensing a beacon for the first time and constraining itself in the circular shaded region.
(b) Beacon moves to another location, node senses again and reduces its reg

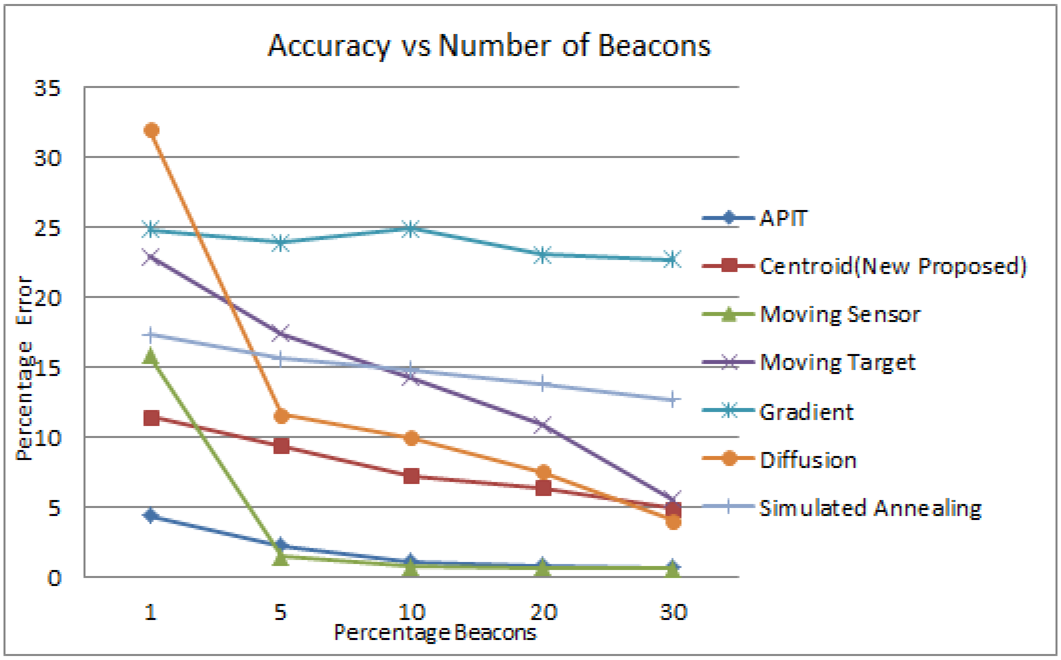


5. Moving Target: All the targets are randomly moved inside the grid. whenever a targets comes in the range of beacon, it makes a bound on its position by superimposing a rectangular box at position shifted by(-x,-y), where (x,y) is the displacement of target in the grid. At last intersection of all the bounds is taken and centroid of that region is taken as the target position.

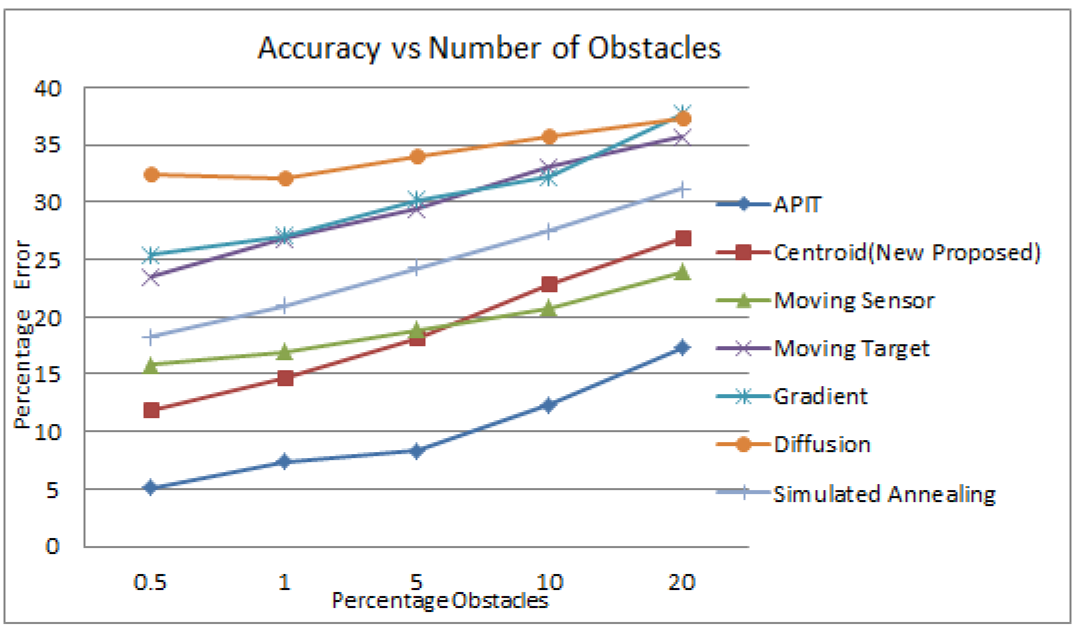
6. Simulated Annealing: System initially starts from random position(i.e. all the normal nodes are assigned random position). A small perturbation d is given to node and Cost Function(CF) is calculated for both transition.CF is the summation of square of calculated distance in random state and measured distance between one hop neighbours. if Δ CF is negative new state is accepted, else new state is accepted with a probability P(Δ CF)=exp$^{(-\Delta CF/T)}$, Here T is Temperature, which gradually decreases along with d after p*q*n perturbation so as to reach equilibrium. n is total number of nodes, p perturbations are given to each node q times in a round robin way. Because we are accepting positive deviation in CF with some probability hence, we are not trapped in local minima.

7. Centroid(New Proposed Algorithm): Since, APIT is a very slow algorithm, a new algorithm is proposed with almost the same accuracy but which is very fast. Nodes are made to listen from all beacons and triangles containing the node are added to a set. Now, based on the area of the triangles a $1/Area$ weight is given to each triangle, because the larger the area the least it should contribute in determining the location of the node. Then the location is calculated by giving these weights to the centroids of the triangle. It works fine with uniformly distributed beacon nodes.

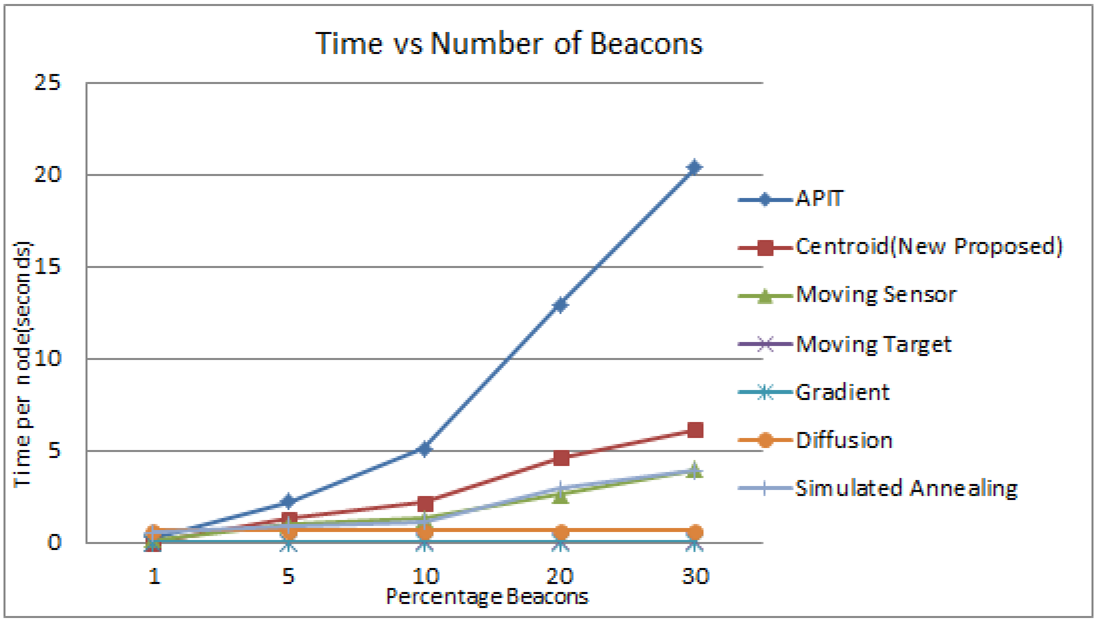
RESULTS



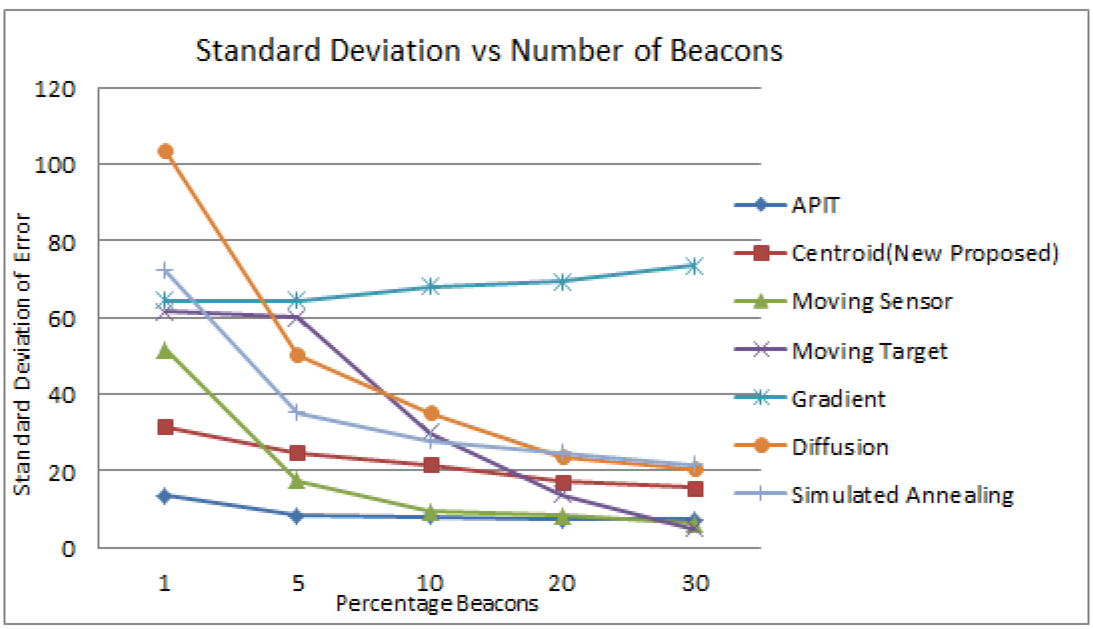
APIT, Centroid(New Proposed), Moving Sensor, Diffusion works better in case of accuracy.



APIT, Centroid(New Proposed), Moving Sensor, Simulated Annealing works better for large number of obstacles.



Gradient, Diffusion, Simulated Annealing, Moving Sensor works better in case of time required per node.



APIT, Moving Sensor, Moving Target, Centroid(New Proposed) works better in case of standard deviation in error.

FUTURE WORK

Based on the qualities required by the system, a cost function can be assigned to the system which is a function of accuracy, tolerance due to obstacles, time required for computation and variation in error. Based on the obtained cost function an appropriate algorithm could be used which minimizes the cost. For example if time is an important parameter then maybe Centroid or Moving Sensor could be used instead of APIT to get minimum cost.