Hybrid Soft Computing Distance Based Localization in Wireless Sensor Networks

1P.M.Kalyani, 2Dr.Tirumala Ramashri

1Research Scholar

Rayalaseema University, Kurnool AP

SV university, Tirupathi, AP

Abstract-- **To improve the localization of nodes in Wireless sensor networks (WSNs), this paper performance analysis of the distance based localization algorithm (LSL) in WSNs, and then proposed to improve soft computing algorithms in Wireless sensor network (WSN). A WSN is a wireless decentralized structure network comprised of nodes, which autonomously set up a network. The node localization that is to be aware of position of the node in the network is an essential part of many sensor network operations and applications. The existing localization algorithms can be classified into two categories: range-based and range-free. The range-based localization algorithm has requirements on hardware, thus is expensive to be implemented in practice. The range-free localization algorithm reduces the hardware cost. Because of the hardware limitations of WSN devices, solutions in range-free localization are being pursued as a cost-effective alternative to more expensive range-based approaches. However, these techniques usually have higher localization error compared to the range-based algorithms. Soft computing algorithms are a typical range-free localization algorithm based on distance estimation. In this paper, propose an improved soft computing algorithm based on distance between nodes. an adaptive hybrid GA–PSO approach is developed to identify the optimal solutions and improve computation efficiency for these localization in WSN. Simulation results show that our proposed algorithm improves the localization accuracy compared with existing algorithms.**

Keywords: **Wireless sensor network, GA, GAPSO, Localization algorithm;**

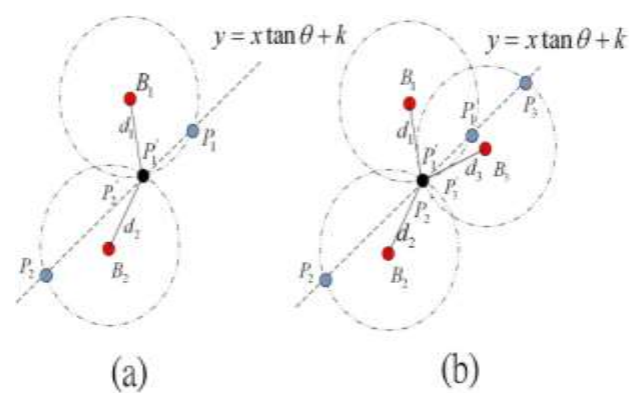
I. Introduction

In WSN, sensor node localization problem is an important issue in many location dependent applications, such as object tracking, traffic management and location based routing. When an abnormal event occurs, the sensor node detecting the event needs the position information to locate the abnormal event and report to the base station. The location of the sensor node is the important information that must be included in the report messages. It is meaningless for the report messages without position information. Global Positioning System (GPS) is the most accurate and most perfect positioning technology to this problem but due to its large equipment and high cost it is not feasible. Because of the high-cost, only a few anchor nodes are equipped with GPS. The other nodes, called unknown nodes, don't know their own position. The anchor nodes can assist the unknown nodes to locate themselves. The problem of obtaining location information of unknown sensor nodes has become a hot topic in WSN.

A. Computing the minimum hop-count value

In the first phase, each anchor node broadcasts a beacon packet with the location of the anchor node and a hop-count value initialized to one, to its neighbor nodes. The format of the packet is {id, xi, yi, Hi}, including the identifier id, coordinate of anchor node i, (xi, yi) and the minimum hop-count value Hi from anchor node i, where the initial value of Hi is 0. After neighbor nodes receive the beacon packet with lesser hop-count value to a particular anchor node, they save the location of the anchor node, and increase the hop-count value by one before transmitting it to other neighbor nodes. Beacon packets with higher hop-count values to a particular anchor node are defined as stale information and will be ignored. Through this mechanism, each node in the network gets the minimum hop-count value to every anchor node.

B. Estimating the distance among nodes



Location determination of normal node: **(a)** with help of two beacon nodes, **(b)** with help of three beacon nodes.

In the second step, we estimate the distance between each node. Firstly, each anchor node calculates the average size for one hop, HopSize, using Eq. (1) which is given as:

$$HopSize_i = \frac{\sum_{i\neq j}^{m}\sqrt{(x_i-x_j)^2+(y_i-y_j)^2}}{\sum_{i\neq j}^{m}H_{ij}} \quad (1)$$

Where (xi, yi) and (xj, yj) are the coordinate of anchor node i and j, and Hij is the minimum hop-count value between anchor nodes i and j, m is the number of anchor nodes.

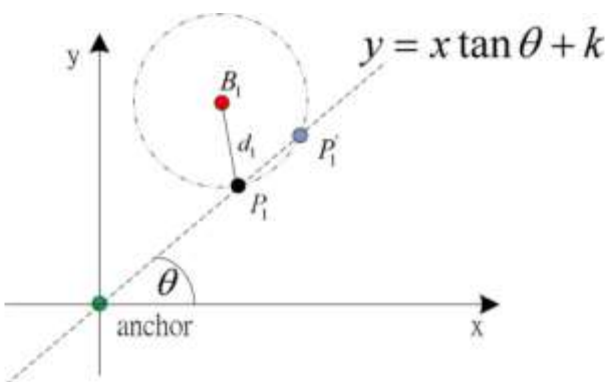
After calculating HopSizei, each anchor node broadcasts its HopSizei in the network by using controlled flooding. When an unknown node j receives the HopSizei information from an anchor node, it calculates the distance between itself and the anchor node using Eq. (2) which is given as:

$$di = HopSizeXH_{ij} \quad (2)$$

where Hij is the minimum hop-count value between anchor node i and unknown node j

C. Computing the location

In the third phase, we calculate the location of each unknown nodes. According to the coordinates of the anchor nodes and the distance to them that has been obtained in the second phase, the unknown node calculates its coordinate by using the multilateration method.



Let (x,y) be the coordinate of unknown node P and (xi, yi) the location of the anchor node i. Therefore, distance between the unknown node P and m anchor nodes is given by Equation (3) can be expanded into

$$\begin{cases} (x_1-x)^2+(y_1-y)^2 = d_1^2 \\ (x_2-x)^2+(y_2-y)^2 = d_2^2 \\ \cdots \\ (x_m-x)^2+(y_m-y)^2 = d_m^2 \end{cases} \quad (3)$$

$$\begin{cases} x_1^2-x_m^2-2(x_1-x_m)x+y_1^2-y_m^2-2(y_1-y_m)y = d_1^2-d_m^2 \\ x_2^2-x_m^2-2(x_2-x_m)x+y_2^2-y_m^2-2(y_2-y_m)y = d_2^2-d_m^2 \\ \cdots \\ x_{m-1}^2-x_m^2-2(x_{m-1}-x_m)x+y_{m-1}^2-y_m^2-2(y_{m-1}-y_m)y = d_{m-1}^2-d_m^2 \end{cases}$$

$$A = \begin{bmatrix} 2(x_1-x_m) & 2(y_1-y_m) \\ 2(x_2-x_m) & 2(y_2-y_m) \\ \cdots & \\ 2(x_{m-1}-x_m) & 2(y_{m-1}-y_m) \end{bmatrix} \quad (5)$$

$$b = \begin{bmatrix} x_1^2-x_m^2+y_1^2-y_m^2+d_m^2-d_1^2 \\ x_2^2-x_m^2+y_2^2-y_m^2+d_m^2-d_2^2 \\ \cdots \\ x_{m-1}^2-x_m^2+y_{m-1}^2-y_m^2+d_m^2-d_{m-1}^2 \end{bmatrix} \quad (6)$$

$$X = \begin{bmatrix} x \\ y \end{bmatrix} \quad (7)$$

Coordinate of the unknown node (x, y) is computed as follows: X = (ATA)-1ATb

II. WSN localization problem

It is supposed that there are m anchor nodes and n unknown nodes in a two-dimensional network. Vector θ = [Z1, Z2 …….Zm+n] represent the initial coordinate of sensor nodes and Zi = [xi , yi]T . The coordinates of m anchor nodes are (x_1, y_1) (x_2, y_2) …. (x_m, y_m) respectively. The essence of the localization problem is to calculate the coordinates of unknown nodes (x_{m+1}, y_{m+1}) (x_{m+2}, y_{m+2}) …. (x_{m+n}, y_{m+n}) based on the given coordinates of m anchor nodes and die distance to the anchor nodes. Therefore, the WSN localization problem can be described as follows:

$$(x, y) = F_{i=1\ldots m}(x_i, y_i, d_i)$$

Where (x, y) is the position of the unknown node, (xi, yi) is the position of the anchor node i and di is the distance between the unknown node and anchor node i.

III. Hybrid PSO with GA

The drawback of PSO is that the swarm may prematurely converge. The underlying principle behind this problem is that, for the global best PSO, particles converge to a single point, which is on the line between the global best and the personal best positions. This point is not guaranteed for a local optimum. A other reason for this problem is the fast rate of information flow between particles, resulting in the creation of similar particles with a loss in diversity that increases the possibility of being trapped in local optima

A further drawback is that stochastic approaches have problem-dependent performance. This dependency usually results from the parameter settings in each algorithm. The different parameter settings for a stochastic search algorithm result in high performance variances. In general, no single parameter setting can be applied to all problems. Increasing the inertia weight (w) will increase the speed of the particles resulting in more exploration (global search) and less exploitation (local search) or on the other hand, reducing the inertia weight will decrease the speed of the particles resulting in more exploitation and less exploration. Thus finding the best value for the parameter is not an easy task and it may differ from one problem to another. Therefore, from the above, it can be concluded that the PSO performance is problem-dependent. The problem- dependent performance can be addressed through hybrid mechanism. It combines different approaches to be benefited from the advantages of each approach. To overcome the limitations of PSO, hybrid algorithms with GA are proposed. The basis behind this is that such a hybrid approach is expected to have merits of PSO with those of GA. One advantage of PSO over GA is its algorithmic simplicity. Another clear difference between PSO and GA is the ability to control convergence. Crossover and mutation rates can subtly affect the convergence of GA, but these cannot be analogous to the level of control achieved through manipulating of the inertia weight. In fact, the decrease of inertia weight dramatically increases the swarm’s convergence. The main problem with PSO is that it prematurely converges to stable point, which is not necessarily maximum. To prevent the occurrence, position update of the global best particles is changed. The posit ion update is done through some hybrid mechanism of GA. The idea behind GA is due to its genetic operator’s crossover and mutation. By applying crossover operation, information can be swapped between two particles to have the ability to fly to the new search area. The purpose of applying mutation to PSO is to increase the diversity of the population and the ability to have the PSO to avoid the local maxima.

Hybrid PSO and GA for Global Maximization A further drawback is that stochastic approaches have problem-dependent performance. This dependency usually results from the parameter settings in each algorithm. The different parameter settings for a stochastic search algorithm result in high performance variances. In general, no single parameter setting can be applied to all problems. Increasing the inertia weight (w) will increase the speed of the particles resulting in more exploration (global search) and less exploitation (local search) or on the other hand, reducing the inertia weight will decrease the speed of the particles resulting in more exploitation and less exploration. Thus finding the best value for the parameter is not an easy task and it may differ from one problem to another. Therefore, from the above, it can be concluded that the PSO performance is problem-dependent. The problem-dependent performance can be addressed through hybrid mechanism. It combines different approaches to be benefited from the advantages of each approach.

To overcome the limitations of PSO, hybrid algorithms with GA are proposed.

The basis behind this is that such a hybrid approach is expected to have merits of PSO with those of GA. One advantage of PSO over GA is its algorithmic simplicity. Another clear difference between PSO and GA is the ability to control convergence. Crossover and mutation rates can subtly affect the convergence of GA, but these cannot be analogous to the level of control achieved through manipulating of the inertia weight. In fact, the decrease of inertia weight dramatically increases the swarm’s convergence. The main problem with PSO is that it prematurely converges to stable point, which is not necessarily maximum. To prevent the occurrence, position update of the global best particles is changed. The position update is done through some hybrid mechanism of GA. The idea behind GA is due to its genetic operator’s crossover and mutation. By applying crossover operation, information can be swapped between two particles to have the ability to fly to the new search area. The purpose of applying mutation to PSO is to increase the diversity of the population and the ability to have the PSO to avoid the local maxima.

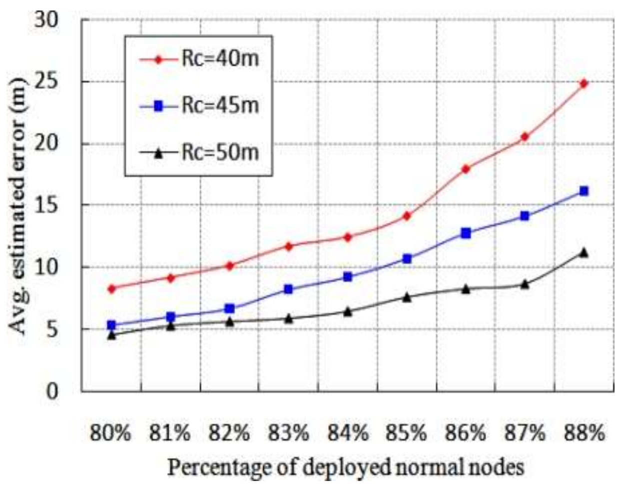
There are three different hybrid approaches are proposed

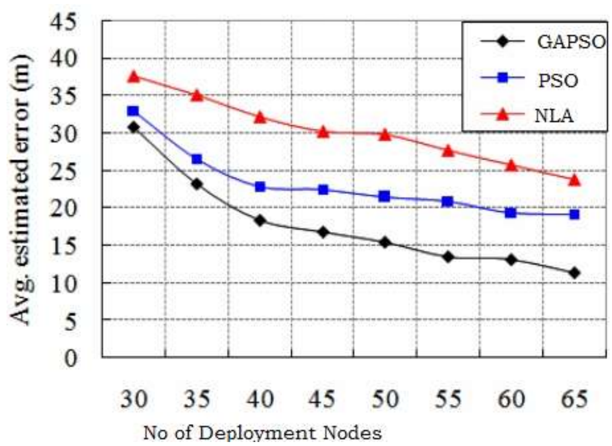
PSO-GA (Type 1): The gbest particle position does not change its position over some designated time steps, the crossover operation is performed on gbest particle with chromosome of GA. In this model both PSO and GA are run in parallel.

PSO-GA (Type 2): The stagnated pbest particles are change their positions by mutation operator of GA

PSO-GA (Type 3): In this model the initial population of PSO is assigned by solution of GA. The total numbers of iterations are equally shared by GA and PSO. First half of the iterations are run by GA and the solutions are given as initial population of PSO. Remaining iterations are run by PSO

IV. Simulation Results





Conclusion

In this paper, a novel distributed localization algorithm is proposed to find location of the normal nodes using only two or three beacon nodes. The localization error determination and error correction methods are proposed GAPSO to given best simulation results. The advantage of our algorithm is that it can work even if only one beacon node provides location information to a normal node. From the performance evaluation of our algorithm, it is observed that our algorithms outperform over similar protocols. Besides, using the proposed method, location of the nodes can be calculated with the simplest ways with less time complexity, which is quite suitable for the memory and energy constraint sensors.

References

1. Li WW, Zhou WN (2011) Genetic algorithm-base localization algorithm for wireless sensor networks. In: Proceedings of seventh international conference on natural computation. IEEE, pp 2096–2099
2. Peng R, Sichitiu ML. Angle of Arrival localization for wireless sensor networks. Proceedings of IEEE Communications Society on Sensor and Ad Hoc Communications and Networks; Reston, VA, USA. 28 September 2006; pp. 374–382.
3. He T, Huang C, Blum BM, Stankovic JA, Abdelzaher T. Range free localization schemes for large scale sensor networks. Proceedings of ACM International Conference on Mobile Computing and Networking; San Diego, CA, USA. 14–19 September 2003; pp. 81–95.
4. Niculescu D, Nath B. *Ad-hoc* positioning system (APS). Proceedings of IEEE Global Telecommunications Conference; San Antonio, TX, USA. 25–29 November 2001; pp. 2926–2931.
5. Savarese C, Langendoen K, Rabaey J. Robust positioning algorithms for distributed *ad-hoc* wireless sensor networks. Proceedings of USENIX Technical Annual Conference; Monterey, CA, USA. 10–15 June 2002; pp. 317–328.
6. Sheu JP, Hsu CS, Li JM. A distributed location estimating algorithm for wireless sensor networks. Proceedings of IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing; Taichung, Taiwan. 5–7 June 2006; pp. 218–225.
7. Fanimokun A, Frolik J. Effects of natural propagation environments on wireless sensor network coverage area. Proceedings of the 35th Southeastern Symposium on System Theory; Morgantown, WV, USA. 16–18 March 2003; pp. 16–20.
8. Lee Y-K, Kwon EH, Lim JS. Self location estimation scheme using ROA in wireless sensor networks. LNCS. 2005;3823:1169–1177.
9. Xu J, Liu W, Lang F, Zhang Y, Wang C. Distance measurement model based on RSSI in WSN. Sci. Res. Wirel. Sens. Netw. 2010;2:606–611.