IOSR Journal of Electronics and Communication Engineering (IOSR-JECE)
ISSN : 2278-2834, ISBN : 2278-8735, PP : 13-18
www.iosrjournals.org

Advance Scheme for Localization using Curvilinear Component Analysis

Sachin T. Matale[1], Dr. Smt. R. S. Patil[2]
[1]*(E&TC Department, D. Y. Patil College of Engg. & Tech. Kolhapur Maharashtra, India)*
[2]*(Prof. E&TC Department, D. Y. Patil College of Engg. & Tech. Kolhapur Maharashtra, India)*

ABSTRACT: *The performance of applications of wireless sensor network depend upon the precise location of the nodes. The localization protocol patches together the relative coordinate, local maps into a global coordinate map. These protocols require some nodes that know their absolute coordinates, called anchor nodes. While many factors influence the node position errors, in this class of protocols, using Procrustes Analysis, the placement of the anchor nodes can significantly impact the error. Through simulation, using the Curvilinear Component Analysis (CCA-MAP) protocol, we show the impact of anchor node placement and propose a set of guidelines to ensure the best possible outcome, while using the smallest number of anchor nodes possible.*

Keywords *-Anchor node placement, Curvilinear Component Analysis, Connectivity Level, Localization, Mean localization error, Range based and range free localization*

1. INTRODUCTION

Many protocols have been proposed [1–3] to calculate relative positions amongst the nodes of a network. They vary in the required network functionality in terms of radio ranging or range-free. Radio ranging involves specialized hardware to measure the distance between nodes based on physical data like signal strength or transmission delays. Procrustes analysis [4] is a common method to convert from relative to global coordinates, requiring some of the nodes to have a local source of global coordinates like GPS receiver or some other source. We call these enhanced nodes anchors. We have CCA-MAP [3, 4] as the algorithm to provide simulation results.

The special hardware and power requirement to perform the ranging techniques is counter to the goal of low-cost, low-power nodes, and thus we exclude ranging protocols from our study. Range-free protocols do not rely on any specialized hardware for additional information. Rather, they rely solely on network connectivity, specifically knowledge of their direct neighbors. Often, a node will collect information about their direct neighbors' neighbors as well, known as two-hop information. Knowledge of each further node requires more information to be shared and therefore transmitted between nodes, thus requiring more power for radio transmission. It is for this reason that only one-hop or possibly two-hop knowledge is preferred.

2. NONLINEAR MAPPING USING CURVILINEAR COMPONENT ANALYSIS

Before going directly to the CCA MAP algorithm first we will briefly go through the non-linear data mapping method of CCA [4][14] and then we will move towards the localization scheme using Curvilinear Component Analysis (CCA).

We can state the localization problem as follows: Given a distance matrix $D_{(NxN)}$ of N nodes, find the coordinates of all the nodes to achieve:

$$\min \sum_{i,j} (dij - pij)sq \quad \text{for } i,j=1,2,\ldots\ldots N \qquad \ldots\ldots(1)$$

Where dij is the measured/known distance between node i and j, and p_{ij} is the distance between node i and j computed using the calculated position coordinates of i and j.

If dij is taken as the distance matrix of the input data set and pij the distance matrix of the output data set, CCA then pushes the above equation to a minimum as it minimizes the cost function. As the only known data of these N nodes assumed here is their distance matrix $D_{(NxN)}= (dij)_{(NxN)}$, we take the distance matrix $D_{(NxN)}$ as both the input data set, i.e., $x_{(NxN)}= D_{(NxN)}$ and the inter-vector distance matrix of the input data set, i.e., $Xij= D_{(NxN)}$. Even though $D_{(NxN)}$ is not the real distance between vectors (i.e., the row vectors) in $D_{(NxN)}$ the CCA algorithm, projects data points quite well given a defined distance matrix without requiring that it be the real Euclidean distance between the input data vectors.

In this way, we have now in the input space N vectors $\{x_i$, i=1, 2 ….N\} each of N dimensions (i.e., n=N). The output data set contains N vectors each reduced to a dimension of 2 (or 3). Without losing the generality of applicability to both 2D and 3D spaces, we use a dimension of 2 in this paper. Thus the output data set is denoted as $y_{(Nx2)}$ which is in fact the 2D coordinates matrix of the N nodes.

The CCA algorithm consists of the following two simple steps to project node coordinates given their distance matrix:

1. Set the initial output estimation of $y_{(Nx2)}$ the mean values of the first two columns of the input data set $x_{(NxN)}$ adjusted by a uniformly randomized standard deviation of the same column.

2. In each cycle, select node i and compute for each node j (j ≠i) the new y_j (t+1) from the current value of $y_j(t)$ using:

$$y_j(t+1) = y_j(t) + \propto(t)\, e^{-\frac{Yij}{\lambda(t)}} \left(\frac{Xij}{Yij} - 1\right) (y_j - y_i) \qquad .(2)$$

Here we selected $F(Yij,\lambda) = e^{-\frac{Yij}{\lambda(t)}}$, Both α (t) and λ (t) decrease with time (i.e., along each computing cycle). Here we have used the function

$$V(t) = v(o) \times \left\{\frac{v(c)}{v(o)}\right\}^{\frac{t}{c-1}} \qquad .(3)$$

Where c is the number of total computing cycles which is often also called the training length in CCA We choose the following values in this paper: α (0) = 0.5, α (c) =α (0) /100 , λ (0)= max{std1,std2, …..stdN}x 3 and λ (c) = 0.01, where stdi is the standard deviation for the i column of the input data set $D_{(NxN)}$

The number of training cycles required is mainly related to the size of the input data set and the accuracy of the distance matrix. The bigger the input data set, i.e., the larger the N, the fewer the cycles are required projecting the final output data. A more accurate distance matrix would also result in fewer cycles, as the cost function will decrease much faster towards the minimum. A maximum number (e.g., 100 is used in this paper for computing local maps) may be assigned to the total cycles.

3. THE CCA-MAP ALGORITHM

We have chosen the CCA-MAP [5,6] algorithm as localization algorithm since it is among the best performing algorithms, with the following characteristics [6]:

· High localization accuracy, even when using only the minimal number of required anchor nodes.
· Where available, supports the use of ranging information, resulting in even better localization performance
· Flexibility whether to run it as a distributed or a centralized algorithm.
· Has a variant, iCCA-MAP [7,8], that can localize nodes in a mobile WSN.

CCA-MAP is a cooperative node localization algorithm, which applies an efficient non-linear data mapping technique called Curvilinear Component Analysis (CCA) [9], to localize nodes in a WSN.

The CCA-MAP scheme builds local maps at each node in the network and then patches them together to form a global map. CCA is employed in computing the node coordinates in the local map. Each node computes its local map using only the local information. If accurate ranging capability is available in the network, local distance between each pair of neighboring nodes is measured and known. Otherwise, only connectivity information is used, assigning a value of 1 to the edge between each neighboring pair of nodes. Then, a distance matrix for all the nodes in the R hop neighborhood of node x can be constructed using the shortest distance matrix as approximation. CCA's reduction technique can generate quite accurate results with a reasonably accurate distance matrix of a small size. In addition, a smaller matrix results in faster computation time. Therefore, R can be set to 1 if the network is dense; otherwise it is set to 2. In the ranging-based scenarios, a one-hop neighborhood distance matrix of at least 12 × 12 produced good results. In the range-free cases, the distance matrix is considerably more inaccurate using the hop count approximation. Thus a larger distance matrix assists in mapping to the node position coordinates using CCA. We set R = 1 when the one-hop neighborhood has a very large size (above 30), otherwise R = 2 is chosen. The steps of CCA-MAP are thus as follows:

1. For each node x, neighbors within R hops are included in building its local map. Compute the shortest distance matrix of the local map and take it as the approximate distance matrix LD.

2. Each node applies the CCA algorithm using the local distance matrix LD as both the input data set and the distance matrix of the input data set. This generates the relative coordinates for each node in the local map of node x of its R hop neighborhood.

3. Merge local maps.

4. Given sufficient anchor nodes (≥ 3 for 2-dimensional space and ≥ 4 for 3-dimensional), transform the merged map to an absolute map based on the absolute positions of anchors.

In step 3, a randomly selected node's local map is used as the starting point. Each time, the neighbor node whose local map shares the most nodes with the current map is selected to merge its local map into the current map. Two maps are merged/patched using the coordinates of their common nodes. To merge a new local map B into the current map A, a linear transformation (translation, reflection, orthogonal rotation, and scaling) is determined to ensure that the coordinates of the common nodes in map B after transformation best conform with those in current map A. This operation, called the Procrustes method, is the same whether we are merging maps or translating the relative coordinates into global coordinates, based on the known anchor locations.

Computing of local maps can be distributed to each local node, or can be carried out at more powerful gateway nodes of each cluster, should the sensor network have a hierarchical structure to relieve the resource-constrained sensor nodes from any of the computing and communication demands imposed by localization. The local maps can be merged in parallel in different parts of the network by selected nodes. The computations can also be done sequentially in a centralized fashion on a central computer. There is no need for anchor nodes in patching the maps. When at least three anchor nodes are found in the patched map of a subnetwork, an absolute map of the subnetwork can be computed using the coordinates of the anchor nodes to obtain the absolute coordinate values of all the nodes in the map of the subnetwork.

4. EXPERIMENTAL RESULTS

We can select rectangle, L-shaped, C-shaped, sine shaped, or irregular shaped network. The CCA MAP algorithm starts with the building the shortest distance matrix, shortest hop matrix, list of neighbors which are in the radio range with their index and the connectivity matrix. We have used the Floyd algorithm to construct the shortest distance and hop matrix.

Local map generation for each node is very important step in the CCA MAP algorithm. If the number of neighbors are greater than 1000 then we have to select hop 1 for the better performance of the algorithm else in general we can take hop as 2. Find the nodes which are in the 2 hop distance from the node and calculate the local distance matrix. The network connectivity level can be calculated by the following formula

Network Connectivity Level = Sum of (number of neighbors for each node -1) / number of nodes

While performing the simulation of CCA MAP algorithm we have observed different case studies by changing the performance parameters such as number of sensor nodes, number of anchor nodes, number of anchor sets, radio range, connectivity level, mean localization error computation time etc.

4.1 Case 1: Effect of number of sensor nodes on connectivity level

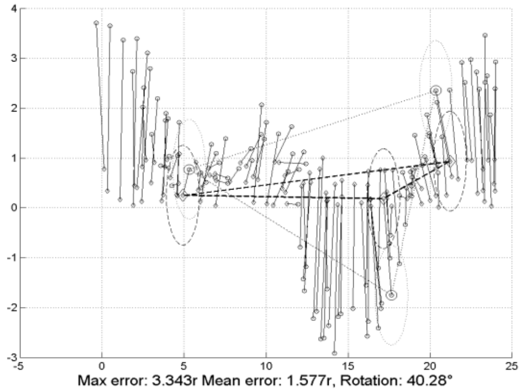
Table 1 Relation between N, CL & T

Sr. No.	Number of nodes	Connectivity Level	Total Time (min)
1	100	14.64	2.780
2	150	22.12	1.062
3	200	30.22	9.195
4	250	38.81	13.357
5	300	46.84	20.885
6	325	50.41	13.701

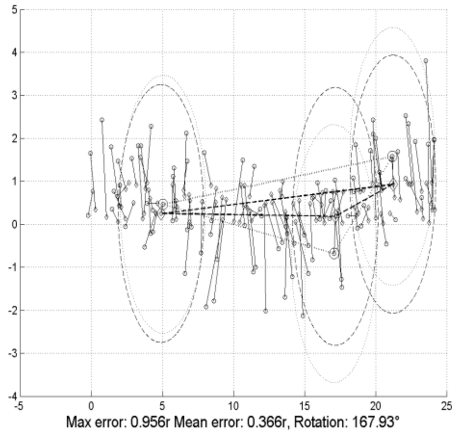
By keeping the network shape & size constant if we increase the number of nodes , the network connectivity level increases but the total time required for CCA Map execution also increases.

4.2 Case 2: Effect of Radio Range on mean localization error

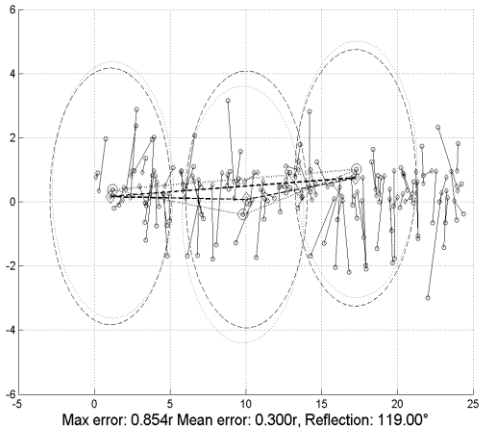
The CCA MAP simulation result we have as in Fig. 1 we state that by keeping the number of sensor nodes, anchor nodes constant if we increase the radio range of the sensor then mean localization error decreases. The Fig.1 shows that for radio range 1.0 mean error is 1.577 while for 3.0 radio range it is 0.366 and for range 6.0 the mean error is 0.286. But to increase the radio range the power consumption and cost of the sensor node increases.



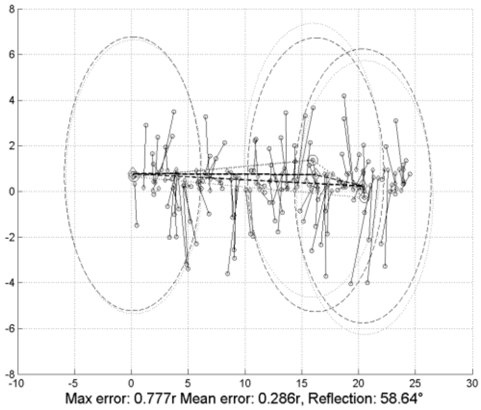
Radio Range 1.0



Radio Range 3.0



Radio Range 4.0



Radio Range 6.0

Fig, 1 Effect of radio range on mean error

4.3 Case 3: Effect of connectivity level on mean error

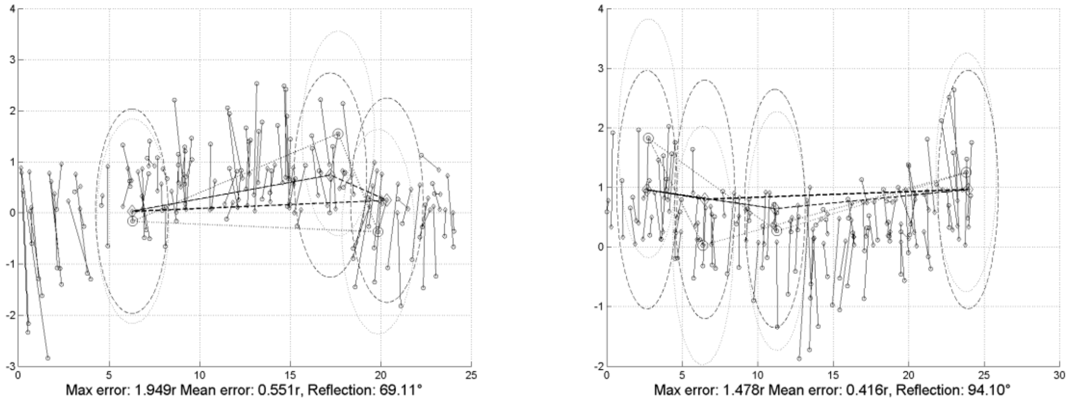
The simulation result shown in table 2 we state that by keeping the radio range constant as the connectivity level increases, mean error decreases.

Table 2 Relation between connectivity level and mean error

Sr. No.	No. of Nodes	Connectivity Level	Mean error
1	100	14.64	0.551
2	150	22.12	0.414
3	200	30.22	0.357

4.4 Case 4: Effect of number of Anchors per set

Fig.2 shows that for sensor nodes 100, radio range 2.0 and number of Anchor set 100, if the number of Anchors per set increases from 3 to 4 the mean error decreases from 0.551 to 0.416.



Anchors per set 3 Anchors per set 4

Fig.2 Effect of number of Anchors per set on mean error

4.5 Case 5: Mean error for different anchor sets

If we have range free localization of 100 sensor nodes having rectangle network shape, radio range 2.0, number of Anchor sets 100 and number of anchors per set is 3 then for this scenario if we observed the CCA MAP simulation results for different anchor sets we have different results depending on the position of the anchors. This is shown in Fig.3

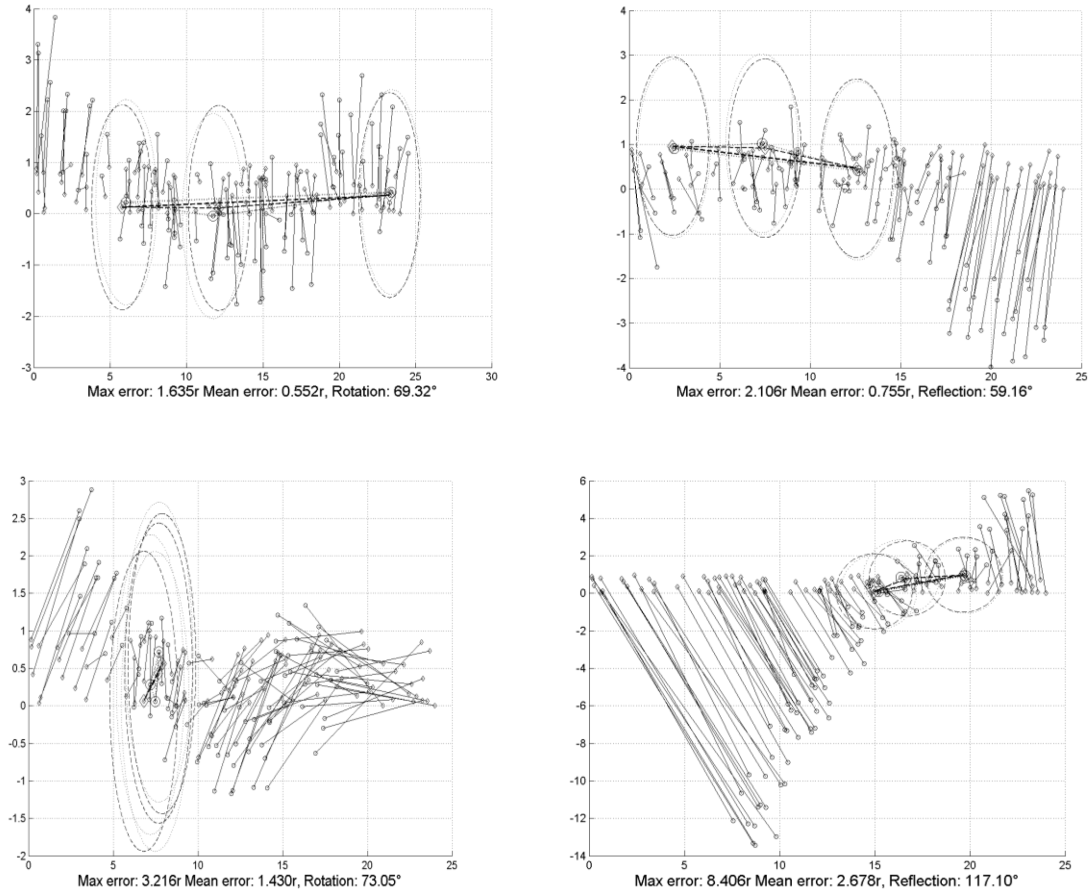


Fig.3 four anchor sets out of 100 anchor sets having 3 anchors per set

5. Conclusion

The simulations study of CCA-MAP algorithm can be used in the system where a local coordinate system is transformed into a global coordinate system using a set of anchor nodes with Procrustes analysis. While we have provided some results to minimize the localization error, there are a few key areas where future study could enhance the results. First, an examination of the other factors affecting localization performance would be beneficial, specifically, an analysis of the impact of the network connectivity levels. Second, expanding these results to three dimensions may benefit some network designers. For example, a sensor network may be deployed through a high-rise building or on a bridge.

REFERENCES

[1] D. Nicelescu and B. Nath, "Ad Hoc Positioning System (APS)," in Global Telecommunications Conference (GLOBECOM 2001), vol. 5, (San Antonio, Texas), pp. 2926–2931, November 2001.

[2] Y. Shang, W. Rumi, Y. Zhang, and M. Fromherz, "Localization from connectivity in sensor networks," Parallel and Distributed Systems, IEEE Transactions on, vol. 15, pp. 961–974, November 2004.

[3] L. Li and T. Kunz, "Localization applying an efficient neural network mapping," in Proceedings of the 1st international conference on Autonomic computing and communication systems (Autonomics 2007), vol. 302, (Rome, Italy), 2007.

[4] L. Li and T. Kunz, "Cooperative node localization using nonlinear data projection," ACM Transactions on Sensor Networks (TOSN), vol. 5, February 2009.

[5] Li, L.; Kunz, T. Cooperative node localization using non-linear data projection. ACM Trans. Sens. Netw. 2009, 5, 1–26.

[6] Li, L.; Kunz, T. Localization Applying an Efficient Neural Network Mapping. In Proceedings of the First International Conference on Autonomic Computing and Communication Systems, Rome, Italy, October 2007; Volume 1.

[7] Alikhani, S.; St-Hilaire, M.; Kunz, T. iCCA-MAP: A New Mobile Node Localization Algorithm. In Proceedings of the IEEE International Conference on Wireless and Mobile Computing, Networking and Communications, Marrakech, Morocco, October 2009; pp. 382–387.

[8] Alikhani, S.; Kunz, T.; St-Hilaire, M. iCCA-MAP versus MCL and Dual MCL: Comparison of Mobile Node Localization Algorithms. Lect. Notes Comput. Sci. 2010, 6288, 163–176.

[9] Demartines, P.; Hrault, J. Curvilinear component analysis: A self-organizing neural network for nonlinear mapping of data sets. IEEE Trans. Neural Netw. 1997, 8, 148–154.

[10] Li Li, Localization in Self-healing Autonomous Sensor Network (SASNet): Requirements, Design Considerations and Research Recommendations, Report of CRC/DRDC, CRC-RP- 2007-005, September 2007

[11] Yi Shang, Wheeler Ruml, Ying Zhang, Markus Fromherz. "Localization from Mere Connectivity", in Proceedings of MobiHoc 2003, Annapolis, MD, June 2003

[12] Yi Shang and Wheeler Ruml, "Improved MDS Based Localization", in Proceedings of IEEE INFOCOM 2004, Hong Kong, March, 2004

[13] Pierre Demartines and Jeanny Herault, Curvilinear Component Analysis: A Self- Organizing Neural Network for Nonlinear Mapping of Data Sets", IEEE Trans. on Neural Networks, Vol. 8, No.1, Jan. 1997, pp.148-154

[14] V. Vivekanandan and V.W.S. Wong, "Ordinal MDS-based Localization for Wireless Sensor Networks", International Journal of Sensor Networks, Vol.1, No. 3/4, 2006, pp.169- 178