# Range based Localization Scheme for 3D Wireless Sensor Network using Joint Distance and Angle Information:A Brief Review

Mr. Santosh P. Salgar
Electronics Engineering Dept
DKTE's TEI, Ichalkaranji

Prof. Veeresh M Metigoudar
Electronics Engineering Dept
DKTE's TEI, Ichalkaranji

Prof. B N Sachidanand
Electronics Engineering Dept
DKTE's TEI, Ichalkaranji

Mr. Prashant P Zirmite
Electronics Engineering Dept
DKTE's TEI, Ichalkaranji

***Abstract* - This review paper describes range based 3D localization algorithm for wireless sensor network to satisfy the practical needs. The mentioned algorithm is anchor free, scalable and provides accurate physical position. To estimate ranges between neighbors both distance and direction measurement technique is used. Based on this information a global network with wide coordinate system is developed using local coordinate system leads to absolute position of nodes. Simulation results have shown that proposed algorithm achieves good tradeoff between localization percentage and precision when node degree equals 12 or around.**

***Keywords - Localization, Wireless Sensor Network, adjacency transformation matrix***



## I. INTRODUCTION

Many localization schemes have been proposed for more precise localization of senor node in wireless sensor network. While working with these schemes a little attention is given towards their application to the practical 3D environment. The existing 2D schemes not even found any deterministic algorithm to verify that uniquely localizable in 3D [1].

Among the existing range-based 3D localization mechanisms [2] tried to reduce the complexity and transform the 3D localization process into its 2D counterpart by employing sensor depth information and additional hardware-upgradable modules. Many 3D localization schemes use trilateration method to calculate the desired position [3-5]. The basic idea is to use at least four anchor nodes to implement the trilateration. This approach normally experiences accelerated error accumulation as more nodes are positioned iteratively, making it difficult to scale. Also assumption is made that the sensor node density to be high in order to attain good localization coverage [6].

This review paper proposes the Three Dimensional Anchor free Localization (3DAFL) algorithm that tries to achieve accurate physical node positions for large-scale WSN in a3D scenario. This algorithm makes use of both distance and angle of arrival (AOA) information [7]. There are two phases in this algorithms. Initially, local coordinate system (LCS) is constructed at each individual node and relative neighbor positions can be calculated accordingly. Later, the LCSs efficiently converge to form a global coordinate system by means of homogeneous coordinate transformation. This algorithm provides high accuracy, low communication overhead, and is robust to node failure.

## II. THREE DIMENTIONAL ANCHOR FREE LOCALIZATION(3DAFL)

The sensor nodes are randomly deployed in 3D scenarios rather than on pure 2D planes. From application (cost, power usage) point of view, minimizing the number of anchors in the network is highly desirable.

### *A. Local position computation*

At the system initialization stage, every node starts exchanging beacon frames to detect its 1-hop neighbors and to build the LCS. The beacon response frame of node *O* contains sequence number and a list of *O*'s neighbors maintained by its *neighbor list table*. For a node *O* to build $LCS_O$, it must have at least two non-collinear neighbors *A* and *B*. See Fig. 1. For the purpose of global coordinate transformation, *O* needs another neighbor, *Q*, such that all the four nodes (*O*, *A*, *B* and *Q*) are non-collinear 1-hop neighbors of each other.

Without loss of generality, we choose *A* and *B* to setup $LCS_O$. +X-axis is temporarily set as along $\overrightarrow{OA}$ with *A* lying on the +x-axis. Then XOY plane can be defined with node *B* lying in the direction of +y-axis. For homogeneity the positive direction of the z-axis is set to be extending out and conforms to the right handed rule.

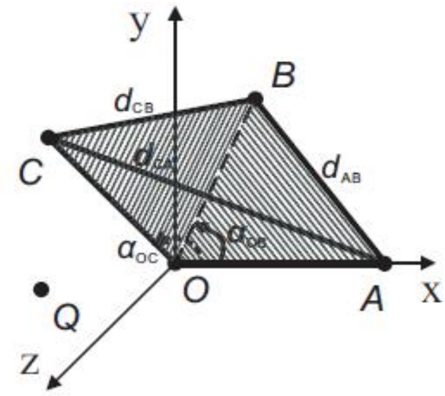


Fig. 1 Local coordinate system and localposition computation

To avoid the flip ambiguity [8] while computing the node positions relative to *LCSO*, node *O* utilizes the bearing information $\alpha_{OB}$ to establish final +x-axis. If $\alpha_{OB} < \pi$, +xaxis $=\overrightarrow{OA}$. Or else, +x-axis=$\overrightarrow{OB}$. This procedure guarantees that LCS at any node is always constructed such that +y-axis is $\pi/2$ counterclockwise to +x-axis with +z axis pulling out when viewed from the top. Now node *O* can use trilateration to compute the positions of its neighbors with respect to *LCSO*. The local position of node *B*, $_{O}P_{B} = (B_x, B_y, B_z)$, is

$$B_x = d_{OB}.cos \propto_{OB}, B_y = d_{OB}.sin \propto_{OB}, B_z = 0 \qquad (1)$$

Where,

$$\alpha_{OB} = across \frac{d_{OA}^2 + d_{OB}^2 - d_{AB}^2}{2d_{OA}d_{OB}}, \alpha_{OB} \epsilon\, (0, \pi)$$

As shown in Fig. 1, the local position of another node *C*, *OPC* = (*Cx*, *Cy*, *Cz*), can be computed if it is the neighbor of both *A* and *B* such that *dOC*, *dCA* and *dCB* are explicitly known by ranging techniques.

$$\begin{cases} C_x = \dfrac{d_{OC}^2 - d_{CA}^2 + A_x^2}{2A_x} \\ C_y = \dfrac{d_{OC}^2 - d_{CB}^2 + B_x^2 + B_y^2 - 2B_xC_x}{2B_y} \\ C_z = \begin{cases} \sqrt{d_{OC}^2 - C_x^2 - C_y^2}\alpha_{OC} \le \pi \\ -\sqrt{d_{OC}^2 - C_x^2 - C_y^2}\alpha_{OC} < \pi \end{cases} \end{cases} \qquad (2)$$

Since the *z* coordinate results from a square root calculation, it is possible to have one or two solutions for the trilateration problem. One solution predicates node *C* being in the same XOY plane while the two alternative solutions can be determined on the AOA basis. Once a node estimates its position it becomes a beacon and assists other unknown nodes in estimating their positions.

Similarly the local position of any node *Q*, $_{O}P_{Q}$, can be fixed if it is neighbors of at least three known nodes *L*, *M* and *N* such that they are all non-collinear neighbors of *O* [5]. In this way sensor nodes are described in a local coordinate reference frame, then they are repositioned into a global coordinate scene as described in the next subsection.

*B. Global position transformation*

In this part we compute the position of a node to a fixed coordinate system (FCS) which acts as a physical location reference to all the nodes in the WSN. To transform position descriptions from LCSs to the FCS, we need to develop an *adjacency transformation matrix* (ATM) that brings two LCSs into alignment. This process continues till all the nodes know their positions relative to the FCS.

We illustrate global position computation by considering the example in Fig. 2. Node *R* can convert its

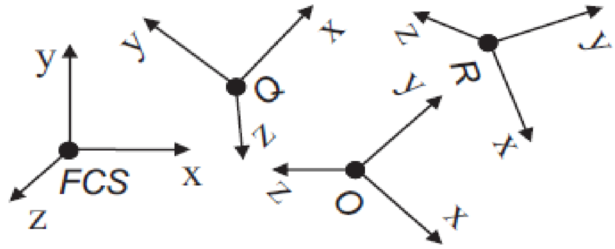


Fig.2 Global position computation

local position $_{O}P_{R}$ to a position relative to *O*'s neighbor *Q*, and thus to a position relative to the FCS, $_{FCS}P_{R}$, as:

$$^{FCS}P_R = {}^{FCS}[T]_Q \,.\, {}^{Q}[T]_O \,.\, {}^{O}P_R \qquad (3)$$

This process continues until all the nodes know their positions relative to the FCS. Such positions, which are consentaneous within the whole network, can easily be absolute positions once the FCS knows its physical position by means such as GPS.

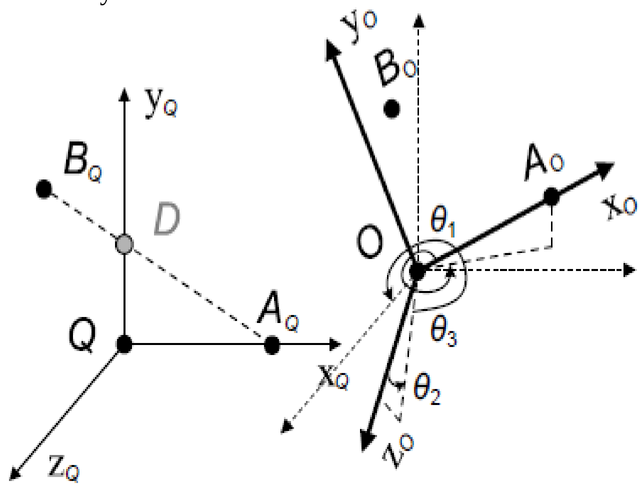


Fig.3 Adjacency transformation matrix development

So the crucial problem is to acquire the ATM at any node. We can generate the *ATMo*, $Q_{[T]_O}$, by performing the following sequence of operations. See Fig. 3.

i. Rotate $\vec{x_o}$ about the $z_o$ axis by the positive angle $\Theta_1$ to bring into the plane that is parallel to XQZ plane.

ii. Rotate $\vec{z_o}$ about the new $x_o$ axis (after step 1) by $\Theta_2$ to bring $z_o$ into the plane that is parallel to XQZ plane.

iii. Rotate $\vec{z_o}$ (after step 2) about the new $y_o$ ” axis (after step 2) by the positive angle $\Theta_3$ to bring $+\vec{z_o}$coincide with the axis that is parallel to + $\vec{z_Q}$ axis.

iv. Translate the rotated *LCSo*”’ (after step 3), which is already aligned with *LCSQ*, such that *O* moves to *Q*.

Accordingly, $Q_{[T]_O}$ can be expressed in the homogeneous coordinate form [8] as a composite transformation involving combination of the four matrix multiplications.

$$Q_{[T]_O} = \begin{bmatrix} 1 & 0 & 0 & {}^{O}Q_x \\ 0 & 1 & 0 & {}^{O}Q_y \\ 0 & 0 & 1 & {}^{O}Q_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} cos\theta_3 & 0 & sin\theta_3 & 0 \\ 0 & 1 & 0 & 0 \\ -sin\theta_3 & 0 & cos\theta_3 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & cos\theta_2 & -sin\theta_2 & 0 \\ 0 & sin\theta_2 & cos\theta_2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} cos\theta_1 & -sin\theta_1 & 0 & 0 \\ sin\theta_3 & cos\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4)$$

Where,

$$\begin{aligned} sin\theta_1 &= -\cos(\vec{\imath_O}, \vec{QD}) \\ cos\theta_1 &= \pm\sqrt{1 - sin^2\theta_1} \\ sin\theta_2 &= -\cos(\vec{k_O}, \vec{QD}) \\ cos\theta_2 &= \pm\sqrt{1 - sin^2\theta_2} \\ cos\theta_3 &= \left| \frac{\cos(\vec{\imath_O}, \vec{\imath_Q})}{cos\theta_1} \right| \\ sin\theta_3 &= \pm\sqrt{1 - cos^2\theta_3} \end{aligned}$$

The plus or minus sign of $\theta_1$ , $\theta_2$ and $\theta_3$ ($\in (0, 2\pi)$) is determined with the aid of AOA technique, respectively. Note that *D* is a virtual node whose relative position $O_{P_D}$ = ($O_{D_x}$ , $O_{D_y}$ , $O_{D_z}$ ) can be calculated by $O_{P_{A_Q}}$, $O_{P_{B_Q}}$ , $O_{P_Q}$ and ∠*DQAQ* = *π*/2. Thus one necessary condition of obtaining $Q_{[T]_O}$ is that the two nodes (*AQ*, *BQ*) determining XQY plane are also *O*’s neighbors.

Iteratively all the nodes can know their positions relative to the FCS, which acts as a physical location reference. Due to the presence of the fixed data-sink in a WSN, it makes sense to select it for defining the FCS.

## III. SIMULATION

To evaluate through simulation the performance of 3DAFL, we deploy *n* sensors in a cubic region *R* such that the node positions are generated using a random uniform distribution.The values of *n* and *R* are selected such that with communication range *r* of 10m we obtain a node degree *d* varying from 6 to 16. The ranging error is Gaussian with a fixed standard deviation, for both distance and bearing estimation.We assume a ranging error of 1% for all our simulations.

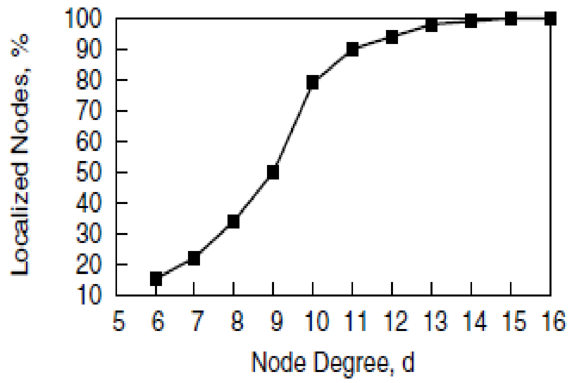


Fig.4 Localization percentage Vs Node degree

Figure 4 shows the ratio of deployed nodes that are able to localize themselves relative to the FCS at a given node degree. We observe that with *d* ≤9 the localization percentage is low. First in a sparse WSN some nodes may not be able to find enough beaconing neighbors to calculate their positions relative to a LCS. When the node density gets higher, this happens mostly at the network borders. Second, as mentioned in section 2.2 a node *O* may not be able to develop an ATM $Q_{[T]_O}$ due to the absence of *AQ* or *BQ*. This problem is greatly alleviated as the node degree increases to 11 and more, which provides us with a valid parameter in real WSN implementation.

Localization error is the offset of the estimated node position from the actual node position. We express this metric relative to the communication range *r*, in terms of percentage. The average localization error for different node density *d* is illustrated in Fig. 5. When the node degree is low (*d* ≤9), the localization error is high due to the reason that enough neighbors are unavailable to reduce the size of the estimation region. Increasing *d* brings more rigid frameworks and more positive conectivity constraints to

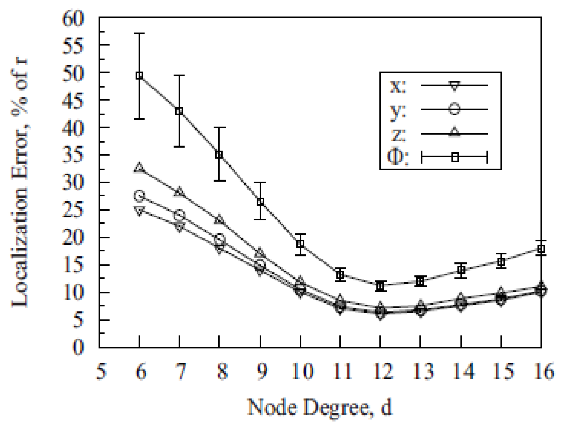


Fig. 5 Mean errors of the x- y- and z- component sorted by the mean

position errors $\Phi = \sqrt{(\Delta_x)^2 + (\Delta_y)^2 + (\Delta_z)^2}$ of sensor nodes.

increase the accurate iterative calculation. One observation is that the errors of the z-component are always higher than those of the horizontal components. This is, however, an acceptable deviation derived from the positioning formula where the z-component of $Q_{P_R}$ depends more on the two values of $\cos\theta_3$ and $\sin\theta_3$ .

## IV. Conclusion

In this paper we proposed a 3D distributed, distance and AOA information oriented, and anchor-free localization algorithm for practical WSN applications. Simulation results demonstrated the effectiveness of our algorithm. For node degree equals 12 or around, the deployment wined good tradeoff between localization percentage and precision. One of our ongoing work is to verify 3DAFL through quantitative comparisons to other 3D range-based localization algorithms.

## REFERENCES

[1] R. Connelly, T. Jordan, and W. Whiteley, "Generic global rigidity of body-bar frameworks," ERGCO Technical Report, TR-2009-13, 2009.

[2] A. Y. Teymorian, W. Cheng, L. Ma, X. Cheng, X. Lu, and Z. Lu, "3D underwater sensor network localization," *IEEE Trans. Mobile Comput.*, vol. 8, no. 12, pp. 1610–1621, Dec. 2009.

[3] F. Thomas and L. Ros, "Revisiting trilateration for robot localization,"*IEEE Trans. Robotics*, vol. 21, no. 1, pp. 93–101, Feb. 2005.

[4] E. Doukhnitch, M. Salamah, and E. Ozen, "An efficient approach for trilateration in 3D positioning," *Computer Commun.*, vol. 31, no. 17, pp. 4124–4129, Nov. 2008.

[5] G. S. Kuruoglu, M. Erol, and S. Oktug, "Three dimensional localization in wireless sensor networks using the adapted multi-lateration technique considering range measurement errors," in *Proc. IEEE GLOBECOM 2009*, pp. 1–5.

[6] D. Moore, J. Leonard, D. Rus, and S. Teller, "Robust distributed network localization with noisy range measurements," in *Proc. SenSys 2004*, pp. 50–61.

[7] G. D. Stefano and A. Petricola, "A distributed AOA based localization algorithm for wireless sensor networks," *J. Computers*, vol. 3, no. 4, pp. 1–8, Apr. 2008.

[8] D. Hearn and M. P. Baker, *Computer Graphics: C Version*, 2nd edition. Pearson Education, 2004.