Scalable and Distributed GPS free Positioning for Sensor Networks[1]

Rajagopal Iyengar and Biplab Sikdar
Department of ECSE, Rensselaer Polytechnic Institute Troy, NY 12180
Email: {iyengr,sikdab}@rpi.edu

***Abstract*— Accurate positioning mechanisms are important in large scale sensor networks to achieve a number of functionalities like location aware routing, efficient coordination of resources and other application specific requirements. This paper proposes a distributed and scalable GPS free positioning algorithm for wireless sensor networks. This approach is an effort in the direction of finding a solution to the positioning problem which minimizes the number of messages exchanged and the coordinate setup time. We use a clustering based approach for the coordinate formation wherein a small subset of the nodes can successfully establish the coordinate system for the whole network. We also compare the performance of this system against existing mechanisms and show that our system scales linearly as the number of nodes in the network increases in contrast to the exponential increase in current mechanisms. Additionally, our mechanism takes considerably lower convergence times. The proposed mechanism is scalable, distributed and able to support the ad hoc deployment of large scale sensor networks quickly and efficiently.**

I. Introduction

Networks of sensors and actuators are characterized by large size, need for distributed coordination and ubiquitous connectivity, power constraints and the ability to be ad hoc deployable. For efficient coordination of the distributed functionality, it is critical to determine the positions of the nodes. This paper presents a performance study of GPS free and beaconless positioning mechanisms and introduces a scalable solution for distributed positioning.

Most of the current literature on location discovery in wireless and sensor networks assumes the availability of GPS receivers at some nodes [8] or beacon nodes with known position [2], [9], [1]. Having a GPS receiver at sensor nodes may not be feasible due to the limitations of satellite coverage or obstructions in the path of satellite signals. For ad hoc deployment of nodes, it unreasonable to assume the presence of beacon nodes with prior position information as assumed in [1], [2], [9]. Also, the solutions proposed in [6], [4] either need centralized systems to solve large optimization problems or are expensive in terms of the number of messages to be exchanged before positions are established.

In this paper, we assume that there are no beacon nodes with known location. A major motivation for our approach is to reduce penalty incurred in sensor networks due to communication overhead. The cost of local computation is lower than that of communication in a power constrained scenario [7]. We show that our system performs better than the one proposed in [4] (which to our knowledge is the only other GPS free and beaconless solution) in terms of communication costs as well as the convergence time. Our solution is scalable as the number of the nodes in the network increases.

The rest of the paper is organized as follows: In the following section we present some background information and related work. Section III presents our positioning mechanism while Section IV evaluates our methodology and compares its performance against existing mechanisms. Finally, Section V presents the concluding remarks.

II. Background and Related Work

The basic requirement of any positioning system is the ability to measure the distance between any two nodes. Various proposals which have been made to determine the distance between nodes in wireless systems include the Signal Strength method, the Angle of Arrival (AOA) method, the Time of Arrival (TOA) and Time Difference of Arrival (TDOA) methods [3]. The effectiveness of the positioning system depends greatly on the accuracy of the distance measurements. The main cause of errors are Non-Line of Sight errors and measuring errors. Methods for detecting and correcting NLOS errors are presented in [5], [12]. In this paper, we assume that the TOA method is used to compute the distances between nodes.

Various approaches have been proposed for positioning in wireless and sensor networks. In [6], a centralized scheme is proposed which collects the entire topology in a server and then solves a large system to minimize the positioning errors. A location system based on an uniform grid of beacon nodes is proposed in [2] while ultrasound signals are used in [9] to measure the distances of nodes from well known beacons. Localization in ad hoc networks based on the known position of a few nodes in the network are proposed in [10], [11]. In [4] the authors propose a distributed mechanism for GPS free positioning in mobile ad hoc networks. However, this procedure is expensive in terms of the number of messages to be exchanged since each node individually re-orients its coordinates to the reference node's coordinates.

III. Scalable GPS Free Positioning

In this section we describe our proposed positioning system. We follow a cluster based approach. The coordinate establishment phase is split into two phases: the local coordinate

[1]This work was supported by the DARPA under contract number F30602-00-2-0537.

establishment at a subset of the nodes and the convergence of the individual coordinate systems to form a global coordinate system.

A. Local Coordinate System

The formation of the local coordinate system is based on triangulation as proposed in [4]. However, to keep the system scalable as the number of nodes increases, we require the formation of local coordinates at only a small subset of the total nodes (which we call *master nodes*). In the following discussion, we assume that a number of nodes are deployed randomly over a geographical region with a given average density.

Once the nodes are deployed, each node starts to decrement a random timer. If the timer of node i expires before it is contacted by any other node, node i becomes a master node and broadcasts a message establishing itself as a master. All nodes in the range of node i who receive this message stop decrementing their timers and become *slave nodes*. We refer to this set of nodes as the *domain* of master node i. Also, some nodes in the domain of i hear from other master nodes. These nodes are called *border nodes* and are central to the formation of the global coordinate system.

To establish the coordinate system and obtain the distance estimates from nodes, the protocol uses various messages for inter-node communication. Each message consists of: (1) the sending node's ID (2) the master node id (3) message type (4) message body. A node can send any of the four following types of messages:

M1: [NodeID, MasterID, M1, Body] enables neighbours to establish distances from the sender.

M2: [NodeID, MasterID, M2, Body] is sent by the slave nodes to masters with distances to the slave's neighbours.

M3: [NodeID, MasterID, M3, Body] are sent out by the master nodes providing the {nodeID, coordinates} tuples to slave nodes.

M4: [NodeID, MasterID, M4, Body]. contains information about transformations to be made by the master nodes.

The overall procedure and sequence of events carried out at each node for obtaining the coordinate system is outlined in Algorithm 1. Once a node's timer expires before anyone else's in its domain, the node assumes the role of a master and broadcasts an M1 message. Nodes hearing this message become part of the master's domain and they also broadcast M1 messages announcing their existence. These messages are also used by the nodes to compute their relative distances from each other. Once a node obtains distance estimates to a prespecified number of nodes (say 2 or 3) it sends these estimates to its master using M2 messages. The master node collects all the distance estimates and then uses triangulation to establish a coordinate system. This forms the local coordinate system at each master node (Details in Appendix 1).

B. Global Coordinate System

Once local coordinate systems have been established at the master nodes, all but one of the master nodes need to reorient their systems in order for the network to converge to a single coordinate system. Let us consider two master nodes i and k which share a border node j. The decision on whether node i should change to the coordinate system of k or vice versa depends on their respective node IDs. In our scheme, the master node with the higher ID changes to the system of the master node with the lower ID. Also, if a master node shares border nodes with more than one master node, it changes its coordinates to those of the master node with the smallest ID. In the following discussion, we assume, without loss of generality, that node k changes over to the system of node i.

While [4] presents a mechanism for determining the translation parameters for a system of three nodes which are aware of their mutual distances, this cannot be applied directly here. This is because, by definition, two master nodes cannot be within the range of each other and thus do not know the distance between them. Thus we need another border node (say l), in addition to j, common to the master nodes i and k in order to obtain the angles and distances necessary for computing the translation parameters. The exact procedure for obtaining these is given in Section III-C.

All master nodes wait for updates from their border nodes with information to compute the rotational and translational changes to be applied. From its set of master nodes, the border node chooses the master node with the lowest ID and forwards its coordinate information to the remaining master nodes, who

Algorithm 1 Algorithm for co-ordinate establishment

```
timer ⇐ init()
decrement(timer)
if timer = 0 AND no M1 received then
  status = master
  broadcast(M1)
  wait(M2s) and form coordinate
  broadcast(M3)
  while master nodeid ≠ minimum do
    wait(updates from border nodes)
    if nodeid < nodeid of update then
      recalculate coordinates and broadcast(M3)
    end if
  end while
else
  status = slave
  broadcast(M1)
  wait(3 distinct M1s are received)
  transmit(M2) to master
  while master nodeid ≠ minimum do
    wait(coordinates from master)
    update coordinates
    if Number of coordinates > 1 then
      type = border
      transmit(M4) to masters with higher nodeid
    end if
  end while
end if
```

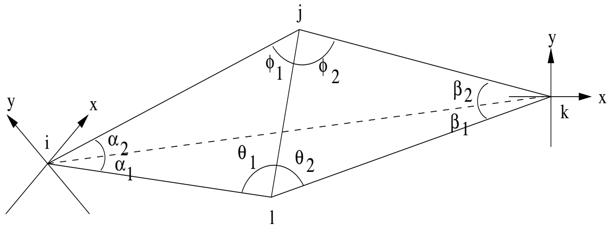


Fig. 1. Estimation of d_{ik} through triangulation.

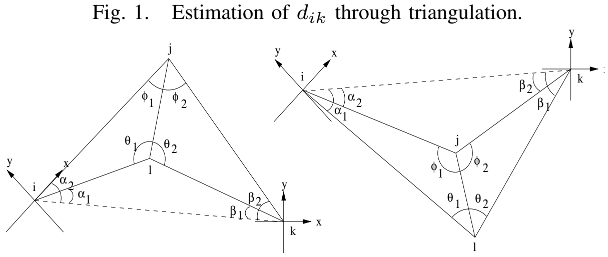


Fig. 2. Special cases of the quadrilateral formation.

use this information and the steps outlined in Section III-C to orient and recompute their positions. Thus at each step, each master node changes its coordinate system to that of the master node with the lowest ID amongst its neighbors. The new coordinates for the domain are then broadcast using M3 messages so that the slaves can update their positions. This procedure continues till the system converges to the coordinate system of the master node with the minimum ID. Detailed results on the cost and convergence times are presented in Section IV.

C. Coordinate Translation and Position Computing

Nodes i,j,k,l now form a quadrilateral as shown in Figure 1 and now the distance d_{ik} between nodes i and k can be obtained through triangulation, as described below.

Using triangles $\triangle(i,j,l)$ and $\triangle(k,j,l)$, the angles θ_1, θ_2, ϕ_1, ϕ_2 can be obtained as

$$\theta_1 = \cos^{-1}\frac{d_{il}^2 + d_{jl}^2 - d_{ij}^2}{2d_{il}d_{jl}} \quad \theta_2 = \cos^{-1}\frac{d_{kl}^2 + d_{jl}^2 - d_{kj}^2}{2d_{kl}d_{jl}}$$

$$\phi_1 = \cos^{-1}\frac{d_{ij}^2 + d_{jl}^2 - d_{il}^2}{2d_{ij}d_{jl}} \quad \phi_2 = \cos^{-1}\frac{d_{kj}^2 + d_{jl}^2 - d_{kl}^2}{2d_{kj}d_{jl}}$$

Then, using $\theta = \theta_1 + \theta_2$, the distance d_{ik} can be calculated as

$$d_{ik}^2 = d_{il}^2 + d_{lk}^2 - 2d_{il}d_{lk}\cos\theta \tag{1}$$

Note however that for the special case shown in right hand side (RHS) of Figure 2 which occurs if $\theta_1 + \theta_2 < \pi$, $\theta = 2\pi - \theta_1 - \theta_2$. In order to find the orientation of node k with respect to node i, we now need to calculate the angles α_1 and α_2. These are given by

$$\alpha_1 = \cos^{-1}\frac{d_{il}^2 + d_{ik}^2 - d_{kl}^2}{2d_{il}d_{ik}} \quad \alpha_2 = \cos^{-1}\frac{d_{ij}^2 + d_{ik}^2 - d_{kj}^2}{2d_{ij}d_{ik}}$$

The final calculations for the rotation and mirroring of the coordinate system at node k depends on the angle ω_{ik} of the vector $\overrightarrow{ik}$ at node i (please see Figure 3). ω_{ik} can now be obtained by adding or subtracting α_1 from ω_{il} depending on the orientation of the quadrilateral $ijkl$. If node l forms the

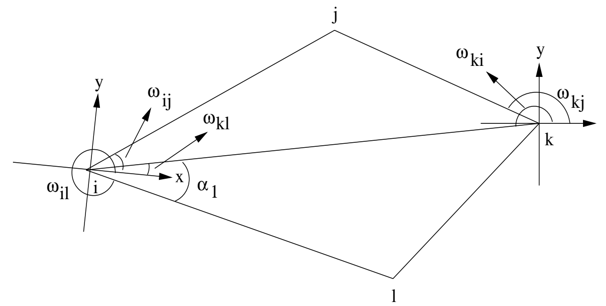


Fig. 3. Angles of the nodes j, k and l from node i (ω_{ij}, ω_{ik} and ω_{il} respectively) and angles of j and i from node k (ω_{kj} and ω_{ki} respectively).

lower two sides of the quadrilateral, the for the quadrilateral shapes in the RHS of Figure 2 and Figure 1, we add α_1 while for the quadrilateral in the LHS of Figure 2, we subtract α_1. For the general case, the node which forms the lower two sides is determined by the following conditions

$$\text{Lower node} = \begin{cases} l & \text{if } \omega_{il} < \omega_{ij} \text{ AND } \omega_{ij} - \omega_{il} < \pi \\ l & \text{if } \omega_{il} > \omega_{ij} \text{ AND } \omega_{il} - \omega_{ij} > \pi \\ j & \text{if } \omega_{ij} < \omega_{il} \text{ AND } \omega_{il} - \omega_{ij} < \pi \\ j & \text{if } \omega_{ij} > \omega_{il} \text{ AND } \omega_{ij} - \omega_{il} > \pi \end{cases} \tag{2}$$

We also note that the quadrilateral on the RHS of Figure 2 is characterized by the fact that $\phi_2 > \pi/2$. Then, ω_{ik} is given by

$$\omega_{ik} = \begin{cases} (\omega_{il} + \alpha_1)\mathrm{mod}(2\pi) & \text{if } \phi_2 < \pi/2 \\ (\omega_{il} - \alpha_1)\mathrm{mod}(2\pi) & \text{otherwise} \end{cases} \tag{3}$$

The equation above assumes that node l is the lower node. However, if j is the lower node, we replace ω_{il} by ω_{ij} and ϕ_2 by θ_2 in the above equation. Similar calculations can also be carried out to determine ω_{ki}. Now, following the arguments of [4], for node k to orient itself to node i's coordinates, the following translations and mirroring are necessary:

if $\omega_{ij} - \omega_{ik} < \pi$ AND $\omega_{kj} - \omega_{ki} < \pi$
or $\omega_{ij} - \omega_{ik} > \pi$ AND $\omega_{kj} - \omega_{ki} > \pi$
$\Rightarrow$ mirroring is necessary
$\Rightarrow$ the correction angle $= \omega_{ki} + \omega_{ik}$
if $\omega_{ij} - \omega_{ik} < \pi$ AND $\omega_{kj} - \omega_{ki} > \pi$
or $\omega_{ij} - \omega_{ik} > \pi$ AND $\omega_{kj} - \omega_{ki} < \pi$
$\Rightarrow$ mirroring is not necessary
$\Rightarrow$ the correction angle $= \omega_{ki} - \omega_{ik} + \pi$

Since the coordinate systems of master nodes i and k now have the same direction and the direction and magnitude of the vector $\overrightarrow{ik}$ is known, the coordinates of any node m in the domain of k can simply be calculated as

$$\overrightarrow{im} = \overrightarrow{ik} + \overrightarrow{km} \tag{4}$$

IV. Results

We now present the results comparing the performance our proposed positioning system and the method outlined in [4]. In the simulations conducted, we consider a rectangular region

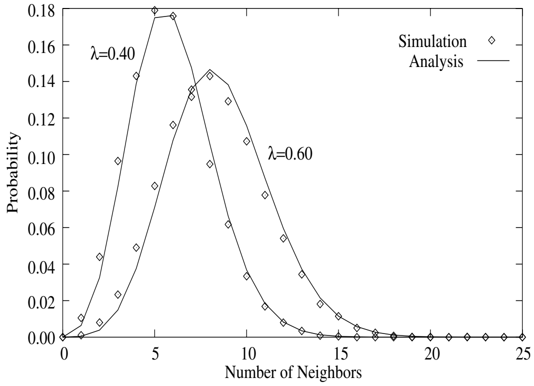


Fig. 4. Probability of number of neighbors.

of length L units and breadth B units. We assume that the nodes are distributed uniformly over this region with density λ nodes/unit^2. Each sensor has a range of r units and the simulations assume a flat topology without any obstructions.

Our first results revolve around the requirements which must be met before any GPS free positioning system become applicable. Since these mechanisms are based on triangulation, we need at least two neighbors for each node for obtaining its location. Additionally, for our scheme of orienting the coordinates of master node to be successful, each border node effectively needs at least three other neighbors (in case the third neighbor is not another border node, we need more nodes). With the nodes being distributed randomly, the probability that an arbitrary node has n neighbors can be approximated by the Poisson distribution with parameter λa:

$$p(n) = \frac{(\lambda a)^n \exp^{-\lambda a}}{\lambda a!} \tag{5}$$

In Figure 4 we compare this probability with those obtained from simulations and note the close match. These results were for an area of 20×20 units with $r = 2$. To find the critical node density where the overwhelming majority of the nodes have more than m neighbors, we note that

$$\text{Prob}(n \geq m) = \sum_{m}^{\infty} p(m) = \frac{\Gamma(m) - \Gamma(m, \lambda a)}{\Gamma(m)} \tag{6}$$

where $\Gamma()$ represents the well known Gamma function. In Figure 5 we show the probability that a node has 2 or more and 3 or more neighbors as a function of the node density. We note that for node densities for more than 0.5 nodes per unit area, more than 90% of the node have 3 or more neighbors. This shows the effectiveness of our proposed positioning policy to form coordinate systems encompassing the overwhelming majority of the nodes, even with relatively low node densities.

In Figure 6 we compare the volume of traffic exchanged by our proposed system with the one from [4]. We see that while the volume of message exchanged by the system of [4] increases exponentially with the node density, our positioning mechanism scales linearly. This can be explained as follows: All nodes converge individually to the chosen coordinate axis according to [4] and then provide information required for convergence to their immediate neighbors. This convergence

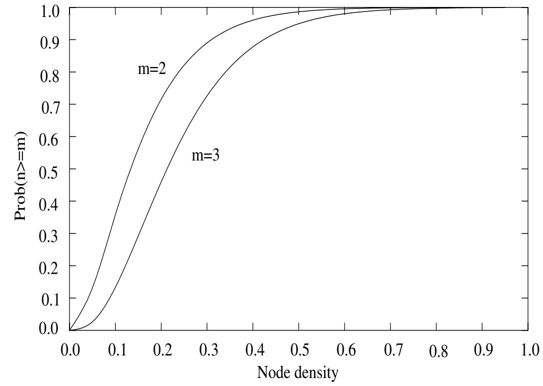
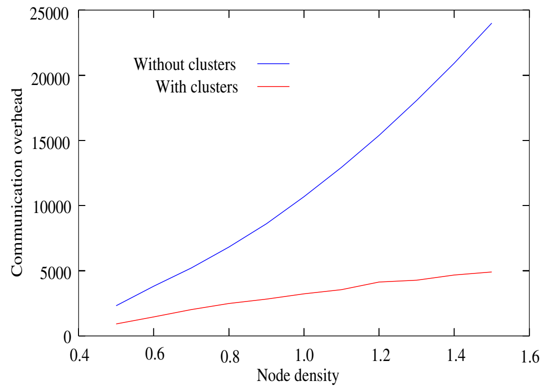
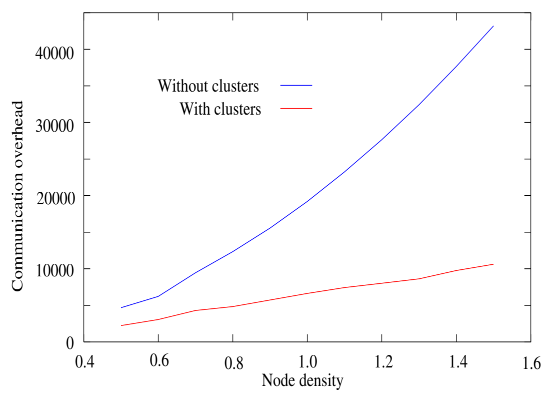


Fig. 5. Probability of having 2 or more and 3 or more neighbors.



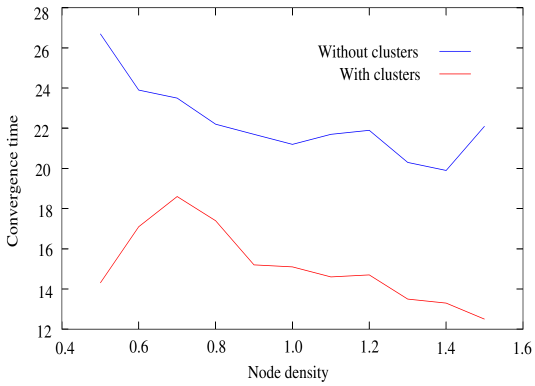
Area: 30×30, $r = 2.0$



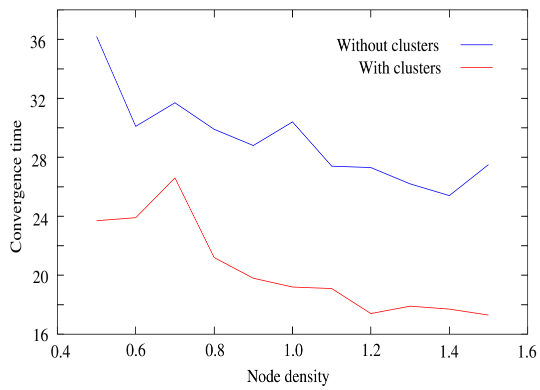
Area: 40×40, $r = 2.0$

Fig. 6. Communication overhead of convergence for the proposed methodology (with clusters) and the one proposed by Capkun et al (without clusters).

information consists of the angle measures to all its neighboring nodes. The size of each message transmitted is then proportional to the number of nodes that it has. We consider the case where the node broadcasts the angle information about all its master node neighbors thereby resulting in lower overheads. Now, the average number of neighbors for each node is given by $\pi r^2 \lambda$. For a given area A, the average number of nodes is given by $A\lambda$. Thus, the overall volume of traffic exchanged is proportional to λ^2. In contrast, our proposal scales all communication by the number of domains rather than the number of nodes. Secondly, each message has information about only two angles making the overall volume of traffic much smaller. These results underline the the fact



Area: 30×30, $r = 2.0$



Area: 40×40, $r = 2.0$

Fig. 7. Convergence time for the proposed methodology (with clusters) and the one proposed by Capkun et al (without clusters).

that our proposal is better suited for deployment in large scale networks.

In Figure 7 we plot the number of rounds required by the two positioning systems for various node densities. We note that our proposed mechanism requires lesser time to converge as compared to the method of [4]. Thus the methodology proposed in this paper is a better candidate to extension to mobile systems.

V. Conclusions

In this paper, we have proposed a distributed and scalable GPS free positioning mechanism. Our proposal provides considerable improvement over current efforts by reducing communication overheads and convergence times. The mechanism is cluster based and allows a distributed framework for establishing a global coordinate system. This allows nodes to be ad hoc deployable in a region without prior location information and ensures a fast setup of the coordinate system. Due to its scalability and fast convergence times, the proposed mechanism is also an ideal candidate for extension to mobile systems.

References

[1] P. Bahl and V. N. Padmanabhan, "RADAR: An in-building RF based user location and tracking system," *Proceedings of IEEE INFOCOM,* pp. 775-784, Tel Aviv, Israel, March, 2000.

[2] N. Bulusu, J. Heidemann and D. Estrin, "GPS-less low cost outdoor localization for very small devices," *IEEE Personal Communications,* vol. 7, no. 5, pp. 28-34, October 2000.

[3] J. Caffery and G. L. Stuber, "Overview of radiolocation CDMA cellular systems," *IEEE Communications Magazine,* vol. 36, no. 4, pp. 38-45, April 1998.

[4] S. Capkun, M. Hamdi and J.-P. Hubaux, "GPS-free positioning in mobile ad-hoc networks," *Proceedings of Hawaii International Conference on System Sciences,* pp. 3481-3490, Maui, HW, January 2001.

[5] P.-C. Chen, "A non-line-of-sight error mitigation algorithm", *Proceedings of IEEE WCNC,* pp. 316 -320, New Orleans, LA, September 1999.

[6] L. Doherty, "Algorithms for positioning and data recovery in wireless sensor networks," Master's Report, University of California, Berkeley, June 2000.

[7] D. Estrin, L. Girod, G. Pottie and M. Srivastava, "Instrumenting the world with wireless sensor networks," *Proceedings of ICASSP,* pp. 2033-2036, Salt Lake City, UT, May 2001.

[8] D. Niculescu and B. Nath, "Ad hoc positioning system (APS)," *Proceedings of IEEE GLOBECOM,* pp. 2926-2931, San Antonio, TX, November 2001.

[9] N. B. Priyantha, A. Chakraborty and H. Balakrishnan, "The cricket location-support system," *Proceedings of ACM MOBICOM,* pp 32-43, Boston, MA, August 2000.

[10] C. Savarese, J. M. Rabaey and J. Beutel, "Locationing in distributed ad-hoc wireless sensor networks," *Proceedings of ICASSP,* pp. 2037-2040, Salt Lake City, UT, May 2001.

[11] A. Savvides, C.-C. Han and M. Srivastava, "Dynamic finegrained localization in ad-hoc networks of sensors," *Proceedings of ACM MOBICOM,* pp. 166-179, Rome, Italy, July 2001.

[12] M. P. Wylie and J. Holtzman, "The non-line of sight problem in mobile location estimation," *Proceedings of the International Conference on Universal Personal Communications,* pp 827-831, Cambridge, MA, September 1996.

Appendix

Let the domain of a master node i be denoted by K_i (i is the origin). Denote the distance between any two nodes, p and q by d_{pq}. A coordinate system can then be established, if there exist two nodes $p, q \in K_i$ such that d_{pq} is known at node i. With the master node i being the origin of the coordinate system, either node p (or node q) can be defined to lie on the positive x axis. Node q (or node p) is now assumed to have a positive y component to define the y axis and the coordinates of the nodes i, p and q are given by

$$\begin{array}{c} p_x = d_{ip}; q_x = d_{iq}\cos\gamma \\ p_y = 0; q_y = d_{iq}\sin\gamma \end{array} \quad (7)$$

where γ is the angle $\angle(p,i,q)$ in the triangle $\triangle(p,i,q)$. For any other node j, $j \in K_i$, $j \neq p,q$ for which d_{ij}, d_{jq}, d_{jp} are known;

$$j_x = d_{ij}\cos\alpha_j \quad (8)$$

$$j_y = \begin{cases} d_{ij}\sin\alpha_j & \text{if } \beta_j = \mid \alpha_j - \gamma \mid \\ -d_{ij}\sin\alpha_j & \text{else} \end{cases} \quad (9)$$

The angles α_j and β_j are obtained through triangulation as

$$\alpha_j = \cos^{-1}\frac{d_{ij}^2 + d_{ip}^2 - d_{jp}^2}{2d_{ij}d_{ip}} \quad (10)$$

$$\beta_j = \cos^{-1}\frac{d_{ij}^2 + d_{iq}^2 - d_{jq}^2}{2d_{ij}d_{iq}} \quad (11)$$

For nodes $k \in K_i$, $k \neq p,q$ which are not neighbors of p and q, the positions can be calculated using its distances from two other nodes for whom the positions are established.