Int. Computer Symposium, Dec. 15-17, 2004, Taipei, Taiwan.

A Range-free Localization Scheme Using Mobile Reference Nodes for Wireless Sensors

Chia-Ho Ou Kuo-Feng Ssu

Department of Electrical Engineering
National Cheng Kung University
Tainan, Taiwan 701

Abstract*-Localization is one of the substantial issues in wireless sensor networks. Several approaches, including range-based and range-free, have been proposed to calculate positions for randomly deployed sensor nodes. With specific hardware, the range-based schemes typically achieve high accuracy based on either node-to-node distances or angles. On the other hand, the range-free mechanisms support coarse positioning accuracy with the less expense. This paper describes a range-free localization scheme using mobile reference nodes. Each reference node equipped with the GPS moves in the sensing field and broadcasts its current position periodically. The sensor nodes obtaining the information are thus able to compute their locations. With the scheme, no extra hardware or communication is needed for the sensor nodes. The localization mechanism has been implemented in the network simulator ns-2. The experimental results show that the location error for our scheme was less than 1 meter on average.*

Keywords: *Wireless sensor networks, localization, range-free, mobile reference nodes, geometry.*

1 Introduction

With advances in hardware and wireless technology, sensor networks can be used for various application areas, such as home, health, military, and industry [1]. A sensor network is composed of a large number of sensor nodes that are densely deployed in a field. Each sensor performs a sensing task for detecting specific events. A particular node, the sink, is responsible for collecting sensing data reported from all the sensors. The sink finally transmits the data to a task manager. When the task manager needs to perform another operation, the new assignment will be disseminated through the sensor network. Communication in the sensor network is based on the wireless ad hoc networking technology [2, 3]. If the sensor nodes cannot directly communicate with the sink, some intermediate sensors have to forward the data.

Several schemes, broadly classified into two categories, have been proposed for dealing with the localization. First, the *range-based* schemes need either node-to-node distances or angles for estimating locations [4–6]. The information can be obtained using Time of Arrival (TOA), Time Difference of Arrival (TDOA), Angle of Arrival (AOA), and Received Signal Strength Indicator (RSSI) technologies. The range-based schemes typically provide high accuracy, less than 5 meters [7] in location error, but they require more hardware on sensor nodes. The *range-free* schemes do not need the distance or angle information for localization [8–10]. The above approaches typically need a large amount of stationary reference points for achieving higher accuracy (10% of the radio range in the best case [10]). Extensive communication among neighboring sensor nodes were also required for some schemes [9, 10]. Accuracy and communication overhead are thus the most critical issues for the range-free approaches. Although the range-free schemes cannot accomplish as high precision as the range-based, they provide an economic approach. Due to the inherent characteristics (low power and cost) of wireless sensor networks, the range-free mechanism could be a better choice to localize a sensor's position.

This paper develops a localization mechanism using the geometry conjecture, (*Perpendicular Bisector of a Chord*). The conjecture states that *a perpendicular bisector of a chord passes through the center of the circle*. Consider that the transmission range of a sensor node is a circle and the center of the circle indicates the position of the sensor node. If any two chords are available, the location of the sensor node can be easily computed based on the conjecture. Our mechanism utilizes mobile reference nodes that move around in the sensing area and periodically broadcast beacon messages, including their current location information. After sensor nodes receive

This research was supported by the Taiwan National Science Council (NSC) under contracts NSC 92-2213-E-006-042 and 93-2213-E-006-019.

the beacon messages with appropriate filtration, the valid beacon points and chords will be determined. Therefore, the center of the circle (the sensor node's location) can be discovered. Several refinements, including randomized beacon scheduling and chord selection, are introduced for performance improvement. The localization mechanism only requires that mobile reference nodes broadcast beacon messages. The ordinary sensor nodes do not spend energy on neighboring interaction for localization.
The mechanism has been evaluated using the network simulator ns-2. With our scheme, location error was less than 1 meter (or 5% of the radio range) on average. The needed execution time for localization can be reduced efficiently by increasing the moving speed, transmission range, or number of the reference nodes.

2 Mobile Reference Nodes

2.1 System Environments and Assumptions

The system environment is a sensor network consists of sensor nodes and mobile reference nodes. The sensor nodes are distributed randomly in the sensing field. Once the nodes are deployed, they will stay at their locations for sensing tasks. The sensor nodes can receive messages from both other nodes and reference nodes. The mobile reference nodes are able to traverse for assisting sensor nodes to determine their locations.
Two main assumptions are required in the paper. First, each mobile reference node has a Global Positioning System (GPS) [11] receiver and sufficient energy during the localization process. Second, the mobile reference nodes are able to move by themselves or other carriers such as robots or vehicles.

2.2 Localization Scheme

The localization scheme was inspired by the *Perpendicular Bisector of a Chord Conjecture*. The conjecture describes that the perpendicular bisector of any chord passes through the center of the circle. As shown in Figure 1, the chord of a circle ($\overline{AB}$) is a segment whose endpoints are on the circle. With two chords of the same circle, the intersection point of two perpendicular bisectors of the chords will be the center of the circle. The localization problem can be transformed based on the conjecture. The center of the circle is the location of the sensor node; the radius of the circle is the largest distance where the sensor node can communicate with the mobile anchors. The endpoint of the chord is the position where the mobile reference node passes through the circle.

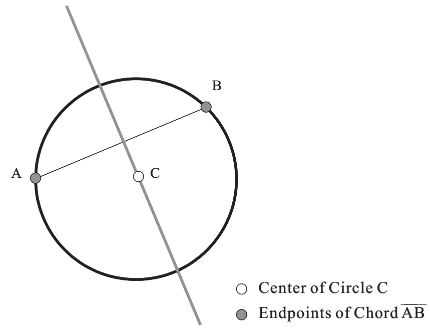

Figure 1. Perpendicular bisector of a chord conjecture.

2.2.1 Beacon Point Selection

In the mechanism, at least three endpoints on the circle should be collected for establishing two chords. Each mobile reference node periodically broadcasts beacon messages when it moves in the sensor network. The beacon message contains the anchor node's id, location, and timestamp. Every sensor node maintains a set of *beacon points* and a *Visitor List*. The beacon point is considered as an approximate endpoint on the sensor node's communication circle. The Visitor List stores both the mobile anchors whose messages have been received by the sensor node and their associated lifetime. The *i-th* beacon point in the sensor is represented as $(id_i, location_i, timestamp_i)$ and the *j-th* entry in the Visitor List can be recorded as $(id_j, lifetime_j)$. When a sensor node receives a beacon message from a mobile reference node, the node will check whether the reference node is in its Visitor List. If not, a beacon point will be added and the reference node with a predefined lifetime will be inserted in the Visitor List. Otherwise, the beacon message will be ignored and the lifetime of the mobile reference node will be extended. When the lifetime of the reference node is expired, the corresponding entry in Visitor List will be removed and the last beacon message of the reference node will be recorded as a beacon point.

Figure 2 demonstrates an example for beacon point selection. A mobile reference node (M) moves and broadcasts beacon messages with an interval t ($T_{i+1} - T_i, i = 0, \ldots, 15$). The M's movement trajectory is shown as the broken line from (x, y) via (x', y') to (x'', y''). The beacon point at T_1 is $(M, (x_1, y_1), T_1)$ and the entry for (M) in Visitor List is $(M, T_1 + \delta)$. The δ is the predefined lifetime for mobile reference nodes and the value of δ should be larger than the beacon interval t ($\delta = \alpha t, \alpha \geq 1$). When the M arrives at T_2, T_3, T_4, and T_5, the M's lifetime will be renewed ($T_i + \delta, i = 2, \ldots, 5$). Af-

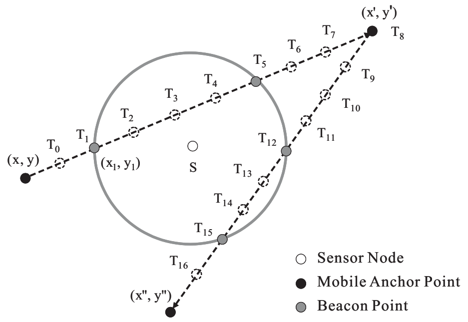

Figure 2. Beacon point selection.

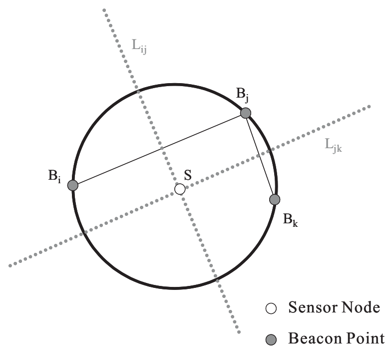

Figure 3. Location calculation.

ter M leaves the transmission range of the sensor S and M's lifetime expires, the entry for M's in the Visitor List will be deleted. In the meantime, the beacon at T_5, the last beacon of M will be picked as a beacon point. Similarly, both beacons at T_{12} and T_{15} are also chosen as beacon points for organizing chords to estimate the location of S.

2.2.2 Location Calculation

After three beacon points are obtained, two different chords can be generated. As shown in Figure 3, the set of selected beacon points is $\{B_i, B_j, B_k\}$ and their locations are (x_i, y_i), (x_j, y_j), and (x_k, y_k). Two chords randomly chosen, $\overline{B_iB_j}$ and $\overline{B_jB_k}$, are formed based on the beacon points. Consider that lines L_{ij} and L_{jk} are the corresponding perpendicular bisectors of the chords $\overline{B_iB_j}$ and $\overline{B_jB_k}$, respectively. The gradients of the chords $\overline{B_iB_j}$ and $\overline{B_jB_k}$ are $g(\overline{B_iB_j}) = \frac{y_j - y_i}{x_j - x_i}$ and $g(\overline{B_jB_k}) = \frac{y_k - y_j}{x_k - x_j}$. Due to the perpendicular property of straight line, the gradients of lines L_{ij} and L_{jk} are $g(L_{ij}) = -\frac{x_j - x_i}{y_j - y_i}$ and $g(L_{jk}) = -\frac{x_k - x_j}{y_k - y_j}$. Therefore, the equations of two lines L_{ij} and L_{jk} can be presented as follows ($a_{ij} = x_j - x_i,\ b_{ij} = y_j - y_i,\ a_{jk} = x_k - x_j,\ and\ b_{jk} = y_k - y_j$).

$$L_{ij} : a_{ij}x + b_{ij}y = c_{ij}$$

$$L_{jk} : a_{jk}x + b_{jk}y = c_{jk}$$

The values for c_{ij} and c_{jk} can be calculated using the midpoint of $\overline{B_iB_j}$ ($x_{ij} = \frac{x_i + x_j}{2}$, $y_{ij} = \frac{y_i + y_j}{2}$) and the midpoint of $\overline{B_jB_k}$ ($x_{jk} = \frac{x_j + x_k}{2}$, $y_{jk} = \frac{y_j + y_k}{2}$), respectively. Therefore, $c_{ij} = a_{ij} * x_{ij} + b_{ij} * y_{ij}$ and $c_{jk} = a_{jk} * x_{jk} + b_{jk} * y_{jk}$. Based on the Cramer's Rule, the intersection point of L_{ij} and L_{jk} ,the estimated location of the sensor node, is ($x = \frac{|c_{ij}*b_{jk} - c_{jk}*b_{ij}|}{|a_{ij}*b_{jk} - a_{jk}*b_{ij}|}$, $y = \frac{|a_{ij}*c_{jk} - a_{jk}*c_{ij}|}{|a_{ij}*b_{jk} - a_{jk}*b_{ij}|}$).

3 Enhancements

3.1 Beacon Scheduling

Broadcasting in wireless ad hoc networks may cause destructive bandwidth congestion, contention, and collision [12]. The collision at sensor nodes could occur due to beacon messages in the mechanism. To handle the problem, the scheduling for broadcasting beacon messages is jittered. The jitter time is randomly selected from a uniform distribution between 0 and ($0.01*$beacon interval). So the jittered beacon interval = beacon interval + jitter time.

The randomized scheduling prevents the beacon collision at the sensor nodes so each node can efficiently obtains beacon messages from different mobile reference nodes.

3.2 Chord Selection

The localization will be accurate if selected beacon points are exact on the communication circle. However, in practical environments, incorrect beacon points could be chosen due to collision or inappropriate beacon intervals. The chords generated using the beacon points thus fails to estimate the position of the sensor. Figure 4 displays large localization errors because of the badly chosen chords. According to our observation, when the length of the chord is too short, the probability of unsuccessful localization will increase rapidly. A threshold (λ) for the length of a chord is used to solve the problem. The relationship between the threshold and communication circle's radius (R) is $0 < \lambda \leq 2R$.

The length of a chord must surpass the threshold for reducing the localization error. Appropriate values for the threshold will be investigated in the performance evaluation.

4 Performance Evaluations

Our simulations were built using the ns-2 network simulator with the Monarch Project wireless and mobile extensions [13]. The radio model was based on the Lucent WaveLAN IEEE 802.11 product with 2 Mbps bandwidth. The IEEE 802.11 wireless LAN

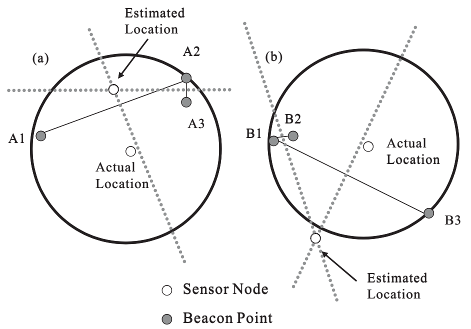

Figure 4. Unsuccessful localization.

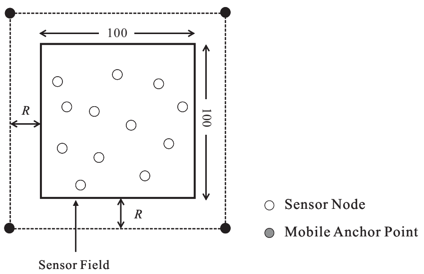

Figure 5. Simulation environments.

Distributed Coordination Function (DCF) was used for media access control model.

4.1 Scenarios

4.1.1 Environment

The sensor field in each simulation was a square of $100 * 100$ meters. 319 sensor nodes were randomly deployed in the sensor field. The radio propagation model was based on *Free Space* model. Each mobile reference node was randomly placed in a corner of the simulated field at the beginning, as illustrated in Figure 5. The reference nodes cannot be placed within the sensing area. Otherwise, the sensor node will take the first beacon as a beacon point that is, however, not located on the communication circle. The transmission range of mobile reference nodes is *R*. The mobile reference nodes moved with the *Random Waypoint* model [14]. Each mobile reference node selected a random destination in the sensor field and then moved to the location with a speed that was uniform distributed from zero to the maximum speed. When the mobile reference node arrived at the destined position, the mobile reference node immediately travelled to the next destination (pause time was zero) with a different velocity. Varying maximum speeds were used in the simulations. When all sensor nodes obtained their locations, the simulation was terminated.

Table 1. Simulation Parameters

Parameter	Value(s)
Beacon interval	(0.1), 0.3, 0.5, 0.7, 0.9 sec
Visitor list lifetime	(3 * beacon interval)
Beacon scheduling	(randomized), periodical
Threshold	0%, 10%, 20%, (30%), 40%, 50% of R
Movement speed	(10), 20, 30, 40, 50 m/sec
Radio range	5, 10, 15, (20), 25 m
# of reference nodes	1%, (2%), 3%, 4%, 5% of sensor nodes

4.1.2 Parameters and Simulations

Detailed parameter settings for the simulations were shown in Table 1. The values in the parentheses were default during the simulations. There were three sets of simulations for evaluation. (1) Beacon scheduling: The evaluation compared the localization accuracy between the randomized scheduling and periodical scheduling. Each simulation utilized five different beacon intervals. (2) Threshold for the length of a chord: There were six varying thresholds. This simulation can verify suitable thresholds for the length of chords with the localization mechanism. (3) Execution time: The experiment can examine the relationship between the needed execution time and the three parameters, including moving speed, radio range, and the number of reference nodes.

4.2 Metrics

Three metrics were utilized to evaluate the performance for our localization mechanism:

- *Average location error*: the average distance between the estimated location (Xe_i, Ye_i) and the actual location (X_i, Y_i) of all sensor nodes. The formula is $\frac{\sum\sqrt{(Xe_i-X_i)^2+(Ye_i-Y_i)^2}}{\#\ of\ sensor\ nodes}$.
- *Average execution time*: the average needed time for sensor nodes to compute their locations. The formula is $\frac{\sum Exec\_time_i}{\#\ of\ sensor\ nodes}$.
- *Beacon overhead*: the average number of beacon messages broadcast by the mobile reference nodes.

4.3 Simulation Results

4.3.1 Beacon Scheduling

Figure 6 compares average location error for the randomized and periodical broadcasting schemes with varying beacon intervals. Reducing the beacon interval can improve localization accuracy for both schemes. The location error of periodical scheduling remained about 9 meters after the beacon interval was less than 0.5 second. On the contrary, the location error decreased dramatically with the randomized scheme. Using 0.1-second beacon interval, the error was less than 1 meter. The results

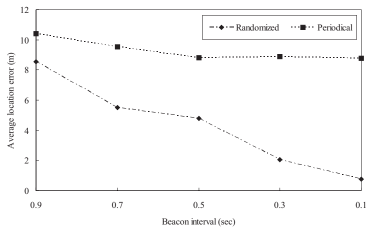

Figure 6. Average location error vs. beacon scheduling.

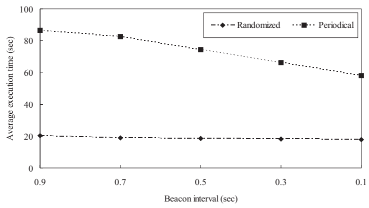

Figure 7. Average execution time vs. beacon scheduling.

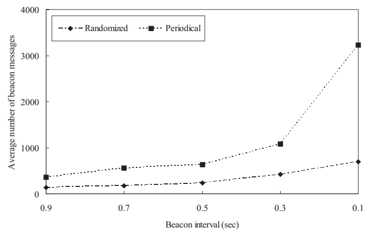

Figure 8. Beacon overhead vs. beacon scheduling.

Table 2. Threshold for Chords

Threshold (% of R)	Average location error (m)	Average execution time (sec)	Average number of beacon messages (packet)
0%	2.79 (0%)	15.25 (0 %)	435.42 (0 %)
10%	1.30 (-54%)	16.18 (+6%)	540.85 (+24%)
20%	0.87 (-69%)	16.96 (+11%)	696.71 (+60%)
30%	0.74 (-74%)	17.86 (+17%)	702.57 (+61%)
40%	0.63 (-78%)	19.03 (+24%)	770.00 (+76%)
50%	0.54 (-81%)	21.42 (+40%)	1255.42 (+188%)

demonstrate that the randomized beacon scheduling promoted the localization accuracy by improving the beacon collision problem.

Avoiding beacon collision also decreased the needed execution time for localization. The fewer number of collisions produced less message loss so the time for collecting adequate beacon points was shorten. As shown in Figure 7, the randomized method reduced 60 seconds in average execution time compared to the periodical approach.

Communication overhead in our mechanism only included the beacon broadcasts. In Figure 8, the beacon overhead for both schemes grew as the beacon interval declined. The periodical scheme required more beacon messages typically. When the beacon interval was 0.1 second, the periodical scheme broadcast about 4 times more beacons than the randomized approach.

4.3.2 Chord Selection

Five different thresholds for the length of chords were evaluated in the simulation (from 0% to 50% of radio range). The 0% represents no limitation for the length of a chord. For a valid chord, its length must exceed the threshold. Table 2 depicts that the location error fell down rapidly with the chord selection. When the threshold was over 20% of R, the error was kept less than 1 meter (about 5% of R). The location error could be improved further to about 0.5 meter if the threshold was 50% of R. The threshold for the chords did enhance the performance of the localization scheme.

The average execution time increased slightly following the growth of the threshold (see Table 2). The time for localization was prolonged because the mechanism needed to collect more beacon points for constructing qualified chords. The beacon overhead was related to the execution time. As displayed in Table 2, the setting of higher thresholds also introduced more number of the needed beacon packets. Based on the above results, the appropriate threshold was about 20% to 30% of the radio range. The remainder of the simulations used 30% of the transmission range as the chord threshold.

4.3.3 Reducing Execution Time

In order to speed up the time for localization, accelerating movement speed, extending radio range, and increasing the number of reference nodes were experimented. Table 3 indicates the relationship between movement speed and average execution time. If the moving speed was faster, the number of beacon messages that sensor nodes received increased in a fixed time period. Therefore, the average execution time for localizing all sensor nodes was reduced. To maintain the original localization accuracy with faster reference nodes, it is necessary to lower the beacon interval correspondingly (see Table 3). The objective of the modification is to ask the mobile anchors to send a beacon every fixed distance of movement.

Table 4 shows that the average execution time was lowered about 66 seconds when the radio range was extended from 5 to 25 meters. With a larger transmission range, the beacon messages sent by the reference node can be listened by more sensor nodes.

Table 3. Moving Speed for Reference Nodes

Moving speed (m/sec)	Beacon interval (sec)	Average location error (m)	Average execution time (sec)	Average number of beacon messages (packet)
10	0.1	0.74 (0%)	17.86 (0%)	696.71 (0%)
20	0.05	0.75 (+1%)	8.99 (-50%)	780.14 (+11%)
30	0.033	0.74 (0%)	6.85 (-62%)	886.71 (+27%)
40	0.025	0.73 (-1%)	4.81 (-74%)	1097.57 (+57%)
50	0.02	0.74 (0%)	4.35 (-76%)	1482.57 (+112%)

Table 4. Radio Range for Reference Nodes

Radio range (m)	Average location error (m)	Average execution time (sec)	Average number of beacon messages (packet)
5	0.96 (0%)	80.60 (0%)	4551.00 (0%)
10	0.83 (-14%)	41.42 (-49%)	2529.14 (-45%)
15	0.74 (-23%)	27.02 (-67%)	1597.57 (-65%)
20	0.74 (-23%)	17.86 (-78%)	696.71 (-85%)
25	0.71 (-27%)	14.21 (-83%)	690.71 (-85%)

Table 5. Number of Reference Nodes

# of reference nodes (% of sensor nodes)	Average location error (m)	Average execution time (sec)	Average number of beacon messages (packet)
1%	0.74 (0%)	41.76 (0%)	1180.85 (0%)
2%	0.74 (0%)	17.86 (-58%)	696.71 (-41%)
3%	0.73 (-1%)	15.66 (-63%)	645.57 (-46%)
4%	0.74 (0%)	11.29 (-73%)	733.14 (-38%)
5%	0.74 (0%)	10.53 (-75%)	846.71 (-29%)

The sensor nodes thus spent less time on collecting beacon points. The needed number of beacons was also decreased due to the reduction of the time for localization. The location error was improved slightly with the large transmission range due to the longer chords (discussed in Section 4.3.2).
Using more reference nodes can reduce the execution time as well. More beacon messages are broadcast in a fixed time interval so the sensors are easier to obtain sufficient beacon points for generating valid chords. Table 5 represents the average time for localization diminished with the increasing numbers of mobile reference nodes. Both the average execution time and beacon overhead were improved apparently when the number of reference nodes was increased from 1% to 2% of the total sensor nodes. With 3% or the higher percentages, the performance enhancement was limited.

5 Conclusion

This paper demonstrated that the range-free localization mechanism without using distance or angle information was also able to achieve fine-grained accuracy. Based on the location information from mobile anchors and the principles of elementary geometry, the sensor nodes can calculate their positions without additional interactions. All computation is performed locally and beacon overhead only occurs on mobile anchors so the mechanism is distributed, scalable, effective, and power efficient. Several enhancements, including randomized beacon scheduling and chord selection, were also introduced. The execution time for the localization mechanism can be shorten if the moving speed, the radio range, or the number of mobile reference nodes is increased. The mechanism was successfully implemented and evaluated using ns-2. The results showed that our mechanism outperformed previous range-free localization schemes. The average location error (less than 1 meter) was also competitive to other range-based approaches that typically require additional hardware on each sensor node.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A Survey on Sensor Networks," *IEEE Communications Magazine*, vol. 40, no. 8, pp. 102–114, Aug. 2002.

[2] E. M. Royer and C.-K. Toh, "A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks," *IEEE Personal Communications*, vol. 6, no. 2, pp. 46–55, Apr. 1999.

[3] C. H. Ou, K. F. Ssu, and H. C. Jiau, "Connecting Network Partitions with Location-Assisted Forwarding Nodes in Mobile Ad Hoc Environments," *Proceedings of IEEE Pacific Rim International Symposium on Dependable Computing (PRDC)*, pp. 239–247, Mar. 2004.

[4] P. Bahl and V. N. Padmanabhan, "RADAR: An In-Building RF-Based User Location and Tracking System," *Proceedings of IEEE Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, pp. 775–784, Mar. 2000.

[5] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan, "The Cricket Location-Support System," *Proceedings of ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM)*, pp. 32–43, Aug. 2000.

[6] D. Niculescu and B. Nath, "Ad Hoc Positioning System (APS) Using AoA," *Proceedings of IEEE Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, pp. 1734–1743, Mar. 2003.

[7] J. Hightower and G. Borriello, "Location Systems for Ubiquitous Computing," *IEEE Computer*, vol. 34, no. 8, pp. 57–66, Aug. 2001.

[8] 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, Oct. 2000.

[9] D. Niculescu and B. Nath, "DV Based Positioning in Ad Hoc Networks," *Kluwer journal of Telecommunication Systems*, vol. 22, no. 1, pp. 267–280, Jan. 2003.

[10] T. He, C. Huang, B. M. Blum, J. A. Stankovic, and T. Abdelzaher, "Range-Free Localization Schemes for Large Scale Sensor Networks," *Proceedings of ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM)*, pp. 81–95, Sept. 2003.

[11] B. Hofmann-Wellenhof, H. Lichtenegger, and J. Collins, *Global Positioning System: Theory and Pratice*. Springer-Verlag, 1997.

[12] S.-Y. Ni, Y.-C. Tseng, Y.-S. Chen, and J.-P. Sheu, "The Broadcast Storm Problem in A Mobile Ad Hoc Network," *Proceedings of ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM)*, pp. 151–162, Aug. 1999.

[13] The Network Simulator - ns-2. http://www.isi.edu/nsnam/ns/.

[14] J. Broch, D. A. Maltz, D. B. Johnson, Y.-C. Hu, and J. G. Jetcheva, "A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols," *Proceedings of ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM)*, pp. 85–97, Oct. 1998.