# Range-Free Sensor Localization with Ring Overlapping Based on Comparison of Received Signal Strength Indicator

Chong Liu, *Student Member, IEEE,* Tereus Scott, Kui Wu, *Member, IEEE,* and Dan Hoffman, *Member, IEEE*

**Abstract**

Sensor localization has become an essential requirement for many applications over wireless sensor networks. Radio irregularity and stringent constraints on hardware cost and battery power, however, make localization very challenging. Range-based localization depends on special-purpose hardware for direct measurement of the distance to reference points. Range-free localization, in contrast, does not depend on absolute point-to-point distance measurement and thus does not require extra hardware. We present a range-free localization method using Ring-Overlapping based on Comparison of Received Signal Strength Indicator (ROCRSSI). We test its performance with a small network of MICA2 Motes and with simulation for large-scale networks. Compared to one of the best existing range-free localization algorithms, ROCRSSI exhibits better localization accuracy and other good features such as smaller communication overhead and resilience to radio irregularity.

**Index Terms**

Range-Free Localization, Ring-Overlapping based on Comparison of Received Signal Strength Indicator (ROCRSSI), Wireless Sensor Networks

This research was supported by Natural Sciences and Engineering Research Council of Canada (NSERC) and Canada Foundation for Innovation (CFI).

Chong Liu, Kui Wu, and Dan Hoffman are with Department of Computer Science, University of Victoria, Victoria, British Columbia, Canada V8W 3P6, E-mail:{chongliu,wkui,dhoffman}@cs.uvic.ca.

Tereus Scott is with GTES Inc., Vancouver, British Columbia, Canada V5K 5B8, E-mail: Tereus.Scott@gtesinc.com

## I. INTRODUCTION

With recent advances in wireless communication and Micro-Electro-Mechanical Systems (MEMS) technology, it has become practical to construct nodes which integrate sensors, processors, memory, wireless communication, and power supply within several cubic millimetres. Large numbers of sensor nodes can be simply dropped in place to monitor a wide variety of real-world phenomena. Through short-range wireless communication, sensor nodes can form Wireless Sensor Networks (WSNs) to coordinate their behaviour, collect and relay observed data, and process the data in a distributed fashion [1].

Location information plays a crucial role for many applications over WSNs. First, location information is needed to determine where sensing data originates, e.g., the location of a monitored vehicle or the point of entry of a burglar. Second, location information can assist in system functionalities such as geographical routing [13], location-based sensor scheduling [14], and location-based information querying [8]. Third, location information facilitates application services such as a location directory service that provides doctors with medical equipment and personnel information in a smart hospital. It is clear that, with the advance of WSN technologies, many protocols and applications will depend on location-aware devices.

For the above reasons, localization in WSNs has become an important research direction. Many algorithms for WSNs have been proposed to provide per-node location information. They can be divided into two categories: range-based methods and range-free methods. Range-based methods depend on absolute point-to-point distance (range) estimation or angle estimation, while range-free methods do not require any range information at all.

Range-based localization depends on the assumption that the absolute distance between a sender and a receiver can be estimated by received signal strength or by the time-of-flight of communication signal from the sender to the receiver. The accuracy of such estimation, however, varies with the transmission medium and surrounding environment, and usually relies on complex hardware [18]. As the miniaturization of sensor nodes continues, the hardware cost associated with range-based localization will make this method less and less attractive. In contrast, range-free localization does not estimate the absolute point-to-point distance based on received signal strength. As a result, the hardware can be greatly simplified, making range-free localization appealing for WSNs.

In this paper, we present a range-free localization algorithm which uses Ring-Overlapping

based on Comparison of Received Signal Strength Indicator (ROCRSSI). While many range-free localization approaches have been proposed [9], [15], [16], their performance has been evaluated only via simulation; we have not seen reports on real implementations of these methods. We have tested our ROCRSSI localization method based on a small-scale wireless sensor network with MICA2 Mote sensors. We have also performed an extensive simulation study with large-scale sensor networks. Compared to one of the best range-free localization methods, Approximate Point In Triangle (APIT), ROCRSSI is more accurate and has smaller communication overhead.

In Section II, we present the ROCRSSI localization method. In Section III, we discuss the impact of radio irregularity on the algorithm. In Section IV, we evaluate ROCRSSI experimentally using MICA2 Motes in two network configurations, and in Section V, we evaluate ROCRSSI using simulation and a wide variety of network configurations. In Section VI, we compare ROCRSSI to APIT. We discuss related work in Section VII.

## II. The ROCRSSI Algorithm

To calculate a position for sensor $S$, the ROCRSSI algorithm generates a set of rings each containing $S$, computes the intersection of the rings, and uses the center of gravity of the intersection as the estimated position of $S$.

The algorithm is based on the following assumptions, for $S$ and the nodes within its radio range:

1) There are two kinds of nodes: anchor nodes and sensor nodes. The positions of the anchor nodes are known to $S$. The distance between anchor nodes $X$ and $Y$ is denoted by $d(X, Y)$.
2) The Received Signal Strength Indicator (RSSI) values between each pair of anchor nodes and between each anchor node and $S$ are known to $S$. The received signal strength measured by node $Y$ for a signal transmitted from node $X$ is denoted by $\gamma(X, Y)$.
3) Signal strength decreases monotonically as distance increases, with some variation as noted in Section III.

Consider the ring shown in Figure 1(a). Using assumption 1, $S$ can calculate the ring centered at $A$. The inner radius is $d(A, B)$; the outer radius is $d(A, C)$. Suppose that $\gamma(A, B) > \gamma(A, S)$. Then $S$ can conclude that it lies outside the circle with center $A$ and radius $d(A, B)$. Suppose further that $\gamma(A, S) > \gamma(A, C)$. Then $S$ can conclude that it lies inside the circle with center $A$ and radius $d(A, C)$. Hence, $S$ lies in the shaded ring shown in Figure 1(a). With a fourth

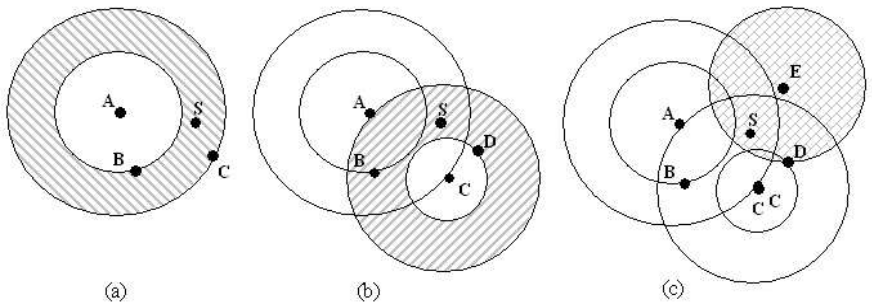


Fig. 1. An example of ROCRSSI: How $S$ decides its location

anchor node, $D$, another ring can be computed. If $\gamma(C, D) > \gamma(C, S) > \gamma(C, A)$, then $S$ can conclude that it lies in the shaded ring with center $C$, as shown in Figure 1(b). Hence, $S$ lies in the intersection of the two rings. Circles, as well as rings, are used to reduce the intersection area. For example, as shown in Figure 1(c), if $E$ is another anchor and if $\gamma(E, S) > \gamma(E, D)$, $S$ can conclude that it falls within the circle with center $E$ and radius $d(E, D)$. With more rings or circles, the intersection area of the rings and circles becomes smaller and smaller.

The ROCRSSI algorithm operates in three phases:

1) *RSSI measurement*: each node measures RSSI from each anchor node in range.
2) *Anchor node data distribution*: each anchor node transmits its location and the RSSI data from the previous phase.
3) *Sensor node location estimation*: each sensor node uses the data from the previous two phases to determine its position.

In the first phase, each anchor node transmits a fixed number of messages containing the anchor identifier. Both the anchor nodes and the sensor nodes monitor the RSSI value for these messages. After the last message has been received, each node computes the mean RSSI value for each node within range.

In the second phase, each anchor node $A$ transmits a message containing the anchor identifier and location, and a $\gamma(X, A)$/identifier pair for each anchor node $X$ within range of $A$, computed in the previous phase. At this point, each sensor node $S$ has acquired:

- The location and identifier of each anchor node within range.
- A value for $\gamma(X, S)$ for each anchor node $X$ in range, acquired during the first phase.
- For each anchor node $A$ within range of $S$, a value for $\gamma(B, A)$ for each anchor node $B$ in

range of both $A$ and $S$, acquired during the second phase.

For example, in Figure 1(a), suppose that the nodes shown are all within radio range of each other. Among the four nodes, there are 12 ordered pairs. At the end of the second phase, $S$ will have acquired an RSSI value for 9 of the 12 pairs. The three values $\gamma(S, A)$, $\gamma(S, B)$, and $\gamma(S, C)$ will be absent, because *sensor nodes do not transmit in the ROCRSSI algorithm.* Note that, during the second phase, sensor $S$ may receive a message from anchor node $A$ which contains information about an anchor node $X$ in range of $A$ but not in range of $S$. This information will be discarded; $S$ retains only information about anchor nodes in range of $S$ itself.

In the third phase, we use a grid-scan algorithm [9] to calculate the center of gravity of the intersection area. In this algorithm, we divide the terrain into a square grid. In general, finer grids result in better accuracy but also larger computational and memory overhead. A counter, initialized to 0, is associated with each square in the grid. Each time we generate a ring, we increment the counter value associated with the squares within the ring. Once all the ring regions have been processed, we scan the grid to find the area with the maximum counter value. We take this area as the final intersection area and calculate its center of gravity as the sensor's location.

## III. Impact of Radio Irregularity

According to measurement results over real sensor devices, radio propagation is usually not homogeneous in all directions [6], [20], [23], i.e., different directions have different radio attenuation rates. Factors such as heterogeneous transmission environment in different directions [23] and antenna position in sensor devices [20] may cause such phenomena. Figure 2 shows the horizontal radio propagation pattern of MICA2 Mote sensors measured with an experimental test bed [20].

Therefore, a good localization algorithm should accommodate radio irregularity. The ROCRSSI algorithm is resilient to radio irregularity for two reasons. First, the grid-scan algorithm helps reduce the influence of incorrect rings generated from radio irregularity. As mentioned before, the grid-scan algorithm takes the grid elements with maximum counter value as the final intersection area. Assume that *more than half* of the rings generated by RSSI comparison are correct (This assumption is validated in most cases in our experimental tests in Section IV). In Figure 3, suppose that the intersection area of all correct rings is the gray area labeled as $A$. The center of gravity of $A$ will be taken as the estimated location of the sensor, because even if all the

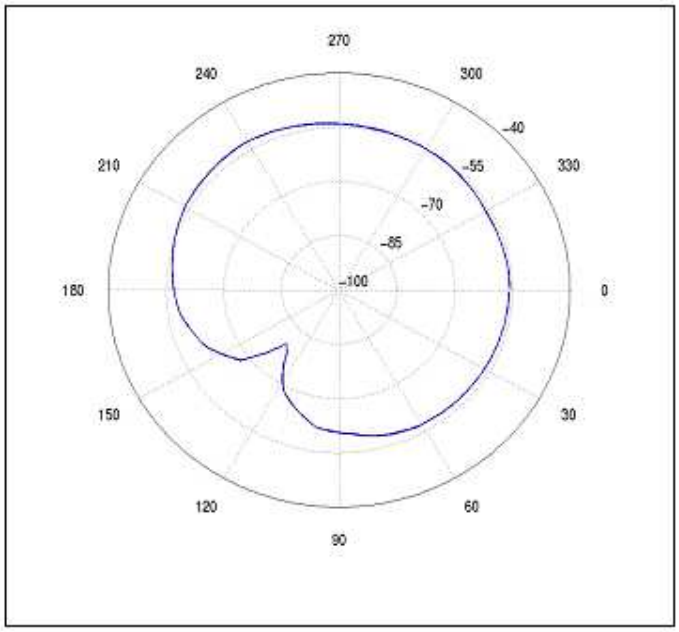


Fig. 2. Horizontal external antenna radiation pattern with MICA2 mote sensors measured in dBm with a distance of 8m from the transmitter

incorrect rings have no intersections with $A$ and all the incorrect rings overlap at one place, the counter values associated with the grid elements at that incorrect place must be smaller than that of the elements in $A$. As a result, incorrect rings will not impact the location calculation. If some incorrect rings overlap with $A$, such as ring $R_2$ in Figure 3, then the final intersection area may not be very accurate. Nevertheless, since the size of $A$ is usually small when the number of in-range anchors is large enough, the center of gravity of the final intersection area will be close to the sensor. As such, even if ROCRSSI generates some incorrect rings, it can still yield fairly accurate location estimation.

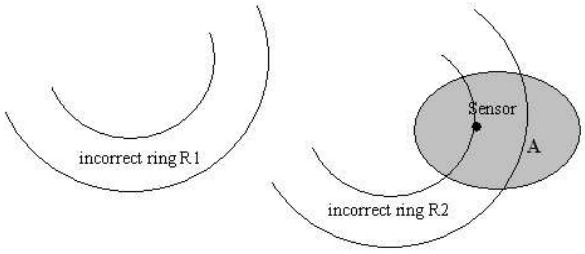


Fig. 3. Grid scan algorithm alleviates the influence of wrong rings

Second, the anchors that are used for sensor localization usually fall within a certain range. As shown in Figure 4, sensor node $S$ will only use RSSI information from in-range anchors $A$, $B$, and $C$, implying that $S$ must fall within the shadowed area. This is because in order to generate rings with $A$'s RSSI values, $S$ must fall within the transmission range of $A$, $B$ and $C$, and $B$ and $C$ must be able to hear $A$. The likely positions of anchors and sensors shown in Figure 4 make ROCRSSI work reasonably well since it is usually true that radio propagation is homogenous in a certain range of directions as shown in Figure 2.

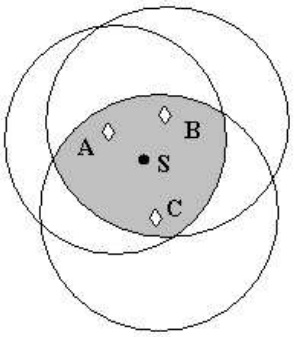


Fig. 4. Likely positions of anchors and sensors that can use ROCRSSI

In summary, while there may exist rare situations where ROCRSSI estimates are poor, the ROCRSSI algorithm is usually insensitive to radio irregularity and can achieve accurate estimates. This claim is supported by the experiments and simulation described in the sections to follow.

## IV. Experimental Study Based on MICA2 Motes

### A. Test Environment and Test Harness Setup

We tested the ROCRSSI localization method with MICA2 Motes in two different environments: a rooftop test range and an outdoor yard.

The rooftop range is an industrial RF emissions test range originally designed for measuring RF emissions from telecommunications equipment. This range was used in order to avoid some of the unwanted and unpredictable RF affects that can occur indoors such as movement of people and furniture. Although it is outdoors, this range exhibits some distinct multipath effects due to the presence of a copper floor, metal air-conditioners and a metal shed at one end of the range.

The outdoor yard is a grass yard on a small slope with large trees and shrubs on the perimeter and a one-story wood-frame house along the side. This environment was chosen because it

should have fewer multipath effects. Unlike the rooftop environment, there are no large metal surfaces to reflect the signal.

Each mote was placed on a wooden pillar, 63 cm from the ground. This distance was used to ensure that the antenna is close to the 0.6 Fresnel zone to reduce the impact of spectral reflections. The sensor mote was placed in the MIB510 programming cradle and a serial cable was connected to a laptop computer located outside of the range. This was done to ensure that there was no impact on the signal strength readings by having a human body close to the equipment under test.

The anchor motes were placed in the same configuration in both the rooftop test and the outdoor-yard test, as shown in Figure 5. The units of measure were different for both ranges; in the roof range the units were 0.5m, and in the yard range the units were 1m. There are three anchor motes, labeled $A$, $B$ and $C$. Each pair of anchor motes defines a circle with one mote as the center and the distance between the two as the radius. The sensor motes are labeled $S_1, S_2, \ldots, S_{11}$. Note that the test configuration, i.e., the number of anchors, the locations of anchors, and the locations of sensors, was chosen to thoroughly test potential areas that a sensor may be located in. Some locations, such as the locations of $S_2$ and $S_4$, were intentionally selected to challenge ROCRSSI since they either are outside of any ring or fall within a large intersection area.

In both test environments, each test run lasted approximately 30 minutes and resulted in 21 to 33 cycles. Each cycle consisted of 9 signal strength samples.

### *B. Design and Implementation*

MICA2 mote sensors run TinyOS, an event-driven operating system. The sensor begins by sending a *startBeacon* message to the anchors. Then the sensor listens for beacons and RSSI reports. The anchors start the beaconing process after they get the *startBeacon* message. Each anchor waits for a different length of time before it starts sending beacons to avoid collisions.

Each anchor then sends periodic beacons while at the same time listening for beacons from other anchors. Once an anchor has heard all the beacons from other anchors, it broadcasts an RSSI report to the sensors with the average RSSI value heard.

The message exchange between the motes is shown in Figure 6. Because we found that the sensors were not always receiving all the RSSI reports from the anchors, we implemented

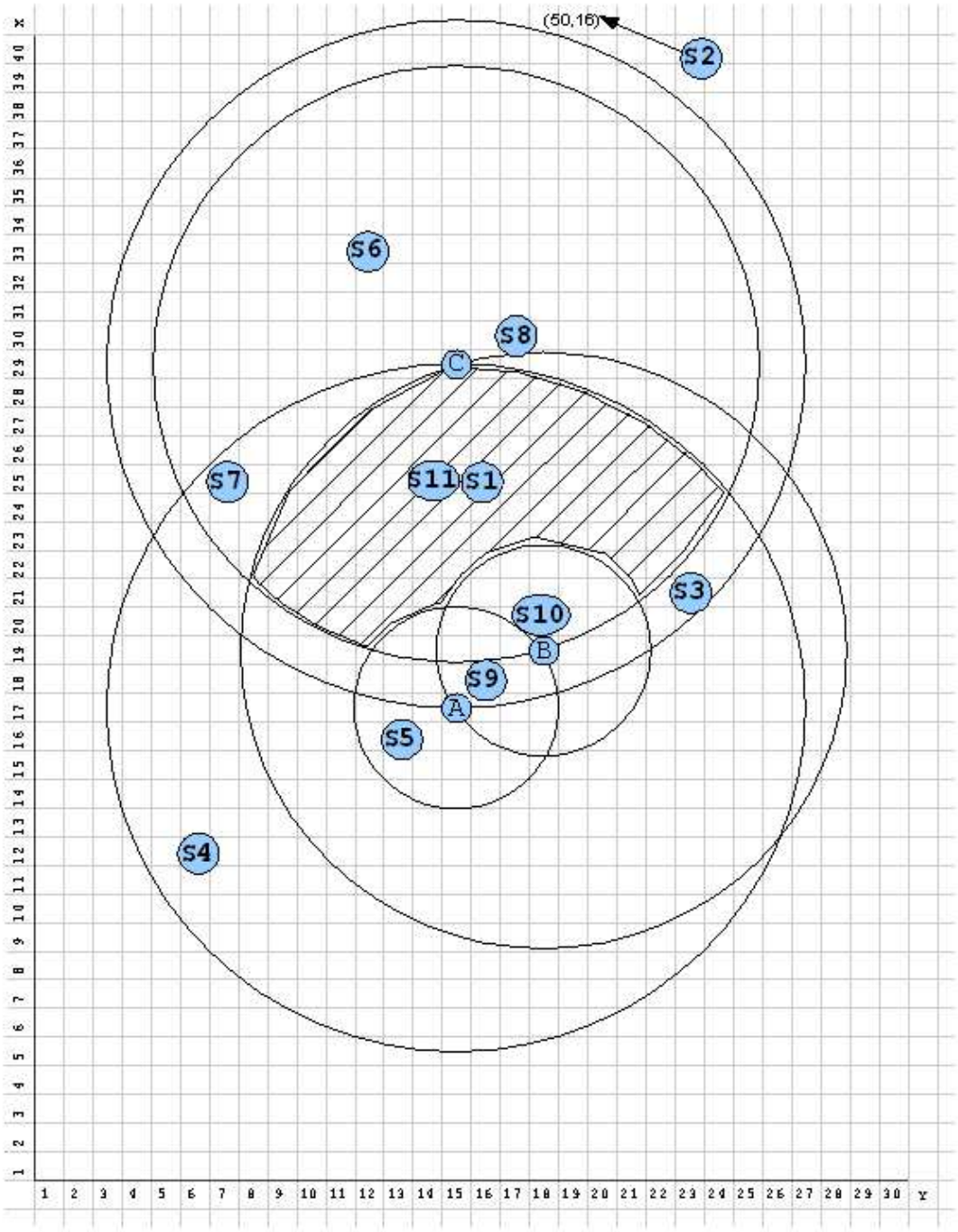


Fig. 5. Test range configuration, anchor and sensor locations

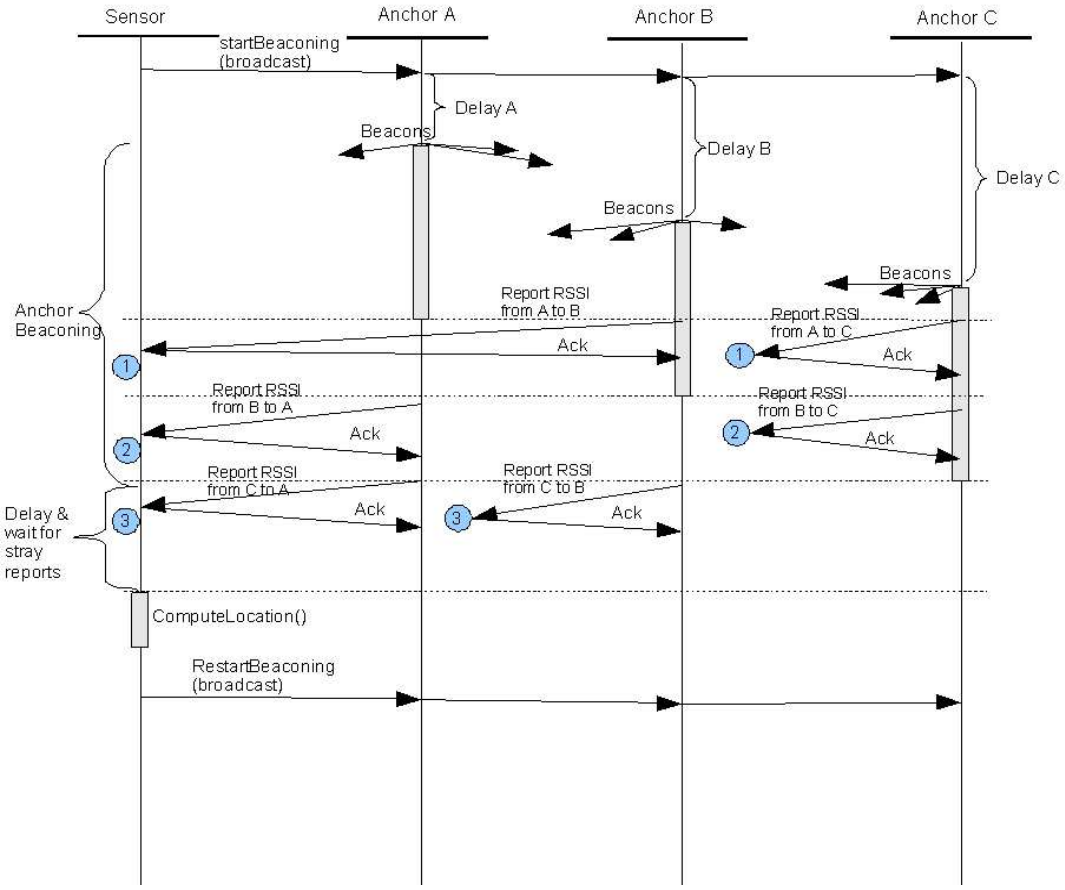


Fig. 6. Beaconing and RSSI report message sequence diagram

an acknowledgement scheme: the sensor sends an acknowledgement when it receives an RSSI report from an anchor. If the anchor does not receive an acknowledgement, it retransmits the RSSI report. Note that this acknowledgement mechanism is for test purposes only. In real applications, a sensor might omit acknowledgements to save energy. As long as a sensor collects enough RSSI reports from the anchors, it can estimate its location. While the loss of several RSSI reports may delay the estimation, it will not have a large impact on the estimation accuracy.

For testing purposes the beaconing process restarts after it is done. There is a short delay before the restart. When the anchors receive the restart message, they reinitialize and start the beaconing process again. Using this approach, a test can be started and restarted at any time. If the sensor mote is reset (using the hardware reset button) it will send out a start beaconing message that will cancel any beaconing that is currently in process at the anchors.

### C. Test Results and Analysis

*1) Verification:* The purpose of this test was to verify that the ROCRSSI algorithm works correctly given a small number of test cases. We kept the anchor locations the same and looked at a number of different regions that the sensor could be in, as shown in Figure 5. In these tests, we used the distances between the nodes to generate ideal RSSI values. These values had to be inverted so that we had a higher number for a closer distance. We selected RSSI equal to 200 minus the distance. With this method, we made up a set of synthetic test files to test ROCRSSI.

We use the synthetic test files to see if the algorithm could properly identify the area containing the sensor. The result was a predicted location for each sensor: the location that the ROCRSSI algorithm should choose if the difference between the signal strength measurements is consistent with our assumptions about signal propagation. Table I shows the results of the preliminary verification, comparing the predicted and actual locations. Note that $S_2$ is an "undetermined node" since it is outside any ring. The estimate of $S_2$ is based on the fact that it falls outside any ring but within the boundary of the test area. This table also compares the average error for all sensor positions as well as the average error with the undetermined node removed.

Since we used synthetic test files to generate ideal RSSI values, the results in Table I represent the best results ROCRSSI can achieve in the test configuration shown in Figure 5.

*2) Measurement Results:* Compared to the results in Table I, results with real measurement data have larger errors for some sensor nodes such as $S_2$, $S_6$, and $S_5$, which fall outside any ring, fall only within one or two circles, or are close to the boundary between two regions. In these situations, there is a chance that ROCRSSI will either use a large overlapping area or put the sensor in the wrong region. This indicates that anchors must be carefully placed in order to make ROCRSSI work well in practice. This test configuration was designed to include scenarios with a poor anchor deployment strategy and a moderately challenged RF environment for ROCRSSI. If we put anchors on the boundary of the testing area and place more anchors, its accuracy will be greatly improved. Nevertheless, even with this test configuration, our real measurement indicates that some sensors such as $S_1$ and $S_{11}$ have estimation accuracy close to the best results in Table I.

For each sensor, we tested the ROCRSSI algorithm at least 22 times. Note that we intentionally omit the statistical results of total sensors, e.g. average error and standard deviation. Such calculation is meaningless since one undetermined node such as $S_2$ will make the error extremely

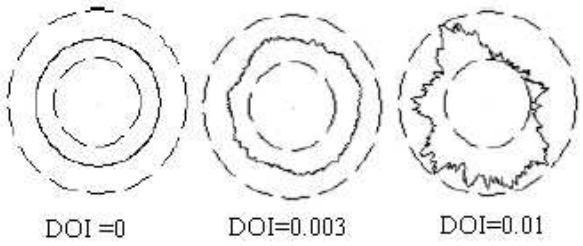


Fig. 7. The Degree of Irregularity (DOI) radio model [9]

large. Instead, we list the detailed measurement results. To save space, we only list part of our measurement results. In Table II, we compare the measurement results of $S_1$ with the rooftop range and the yard-range. Table III lists the measurement results of $S_8$, $S_9$, and $S_{10}$ with the yard-range.

## V. Simulation Study In Large-Scale Networks

### A. Radio Model

For cost reasons, we did not test the ROCRSSI algorithm in a large-scale sensor network. Instead, we used simulation to investigate the performance of ROCRSSI localization in large-scale networks with different settings, i.e., network density, radio irregularity, and anchor deployment strategy.

As demonstrated in [9], [20], [23], radio propagation exhibits the feature of non-isotropic path losses. Thus, to closely approximate a real network environment, we do not use the perfect circular radio model in our simulation. Instead, we adopt and extend the Degree Of Irregularity (DOI) radio model [9]. In this model, the DOI value is defined as the maximum range variation per unit degree change in the direction of radio propagation. Large DOI values represent large variation of radio irregularity as shown in Figure 7. Unlike the previous DOI model, we do not assume any lower bound on radio irregularity, i.e., even close to the sender, radio irregularity may still be present. This model makes localization challenging but produces more realistic results. The extended DOI model can calculate the possible received signal strength at any point within the maximum radio range of a sender.

In the extended model, the DOI value is still used to adjust the degree of radio irregularity. The signal strength is $C \times K(\theta)/d^2$ where $C$ is a constant, $d$ is the distance between the receiver and the sender and is smaller than the maximum radio range of the sender, and $K(\theta)$ is the

coefficient representing the radio propagation feature in the specific direction $\theta$. Thus the radio irregularity is expressed by different $K(\theta)$'s associated with different directions. For $\theta \in [0, 360)$:

$$K(\theta) = \begin{cases} 1 & \theta = 0 \\ K(\theta - 1) + Rand \times \text{DOI} & \theta \text{ is a positive integer} \\ K(t) + (\theta - s) \times (K(t) - K(s)) & \text{Otherwise} \end{cases}$$

In the formula, $Rand$ is a random number uniformly distributed between $(-1, 1)$, $s = \lfloor \theta \rfloor$, and $t = \lceil \theta \rceil$.

### *B. System Parameters and Performance Measurements*

In our simulation, we studied several system-wide parameters which directly affect the accuracy of location estimation of ROCRSSI:

- Radio transmission range (R): the maximum distance over which a sensor can successfully send messages. This value is used for normalization only. All distances including error estimation are normalized by R to ensure generally applicable results.
- Anchor to Node Range Ratio (ANR): the average distance an anchor beacon can travel divided by R. We assume that all anchors have the same ANR within each simulation.
- Anchor Number (AN): the number of anchors deployed. We use AN, rather than Anchor Percentage [9] because ROCRSSI does not depend on the density of sensors; with ROCRSSI sensor nodes do not exchange information.
- Anchor deployment strategies. We investigated four anchor deployment strategies within a square with edge length $10R$:
    1) *Even deployment at the edges of terrain*. The anchors are deployed evenly at the edges of terrain.
    2) *Random deployment at the edges of terrain*. The anchors are deployed randomly at the edges of the terrain, and the number of anchors on each edge is the same.
    3) *Uniform deployment inside terrain*. The terrain is partitioned with each grid element the same size, and the anchors evenly divided amongst these grid elements. The locations of anchors in each grid element are randomly selected.
    4) *Random deployment inside terrain*. The anchors are distributed randomly inside the $10R \times 10R$ square.

We define the location estimation error as the Euclidian distance between the real location of a node and its estimated location. For each simulation scenario, ten runs with different random seeds were executed and the results were averaged. We use average location error as the metric to evaluate the accuracy of location estimation. It is defined as the mean of location estimation errors collected over all determined sensor nodes in ten runs. A sensor node is determined if it falls within at least three rings (or circles). Otherwise, it is an undetermined node.

### *C. Results and Analysis*

*a) Varying Deployment Strategy:* Figure 8 indicates that, no matter how DOI and ANR vary, uniform deployment inside terrain always has the best performance given the same number of anchors. Also, the performance of random deployment inside terrain is very close to that of uniform deployment inside terrain. Therefore, the random deployment has close to the best performance. This is important because this deployment is the easiest to carry out in practice.

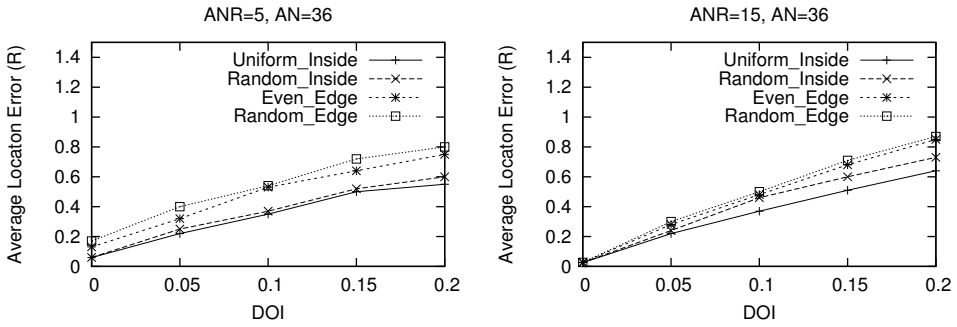


Fig. 8. Average location error under different deployment strategies

That deployment of anchors inside terrain has better performance than deployment of anchors at the edges of terrain can be explained as follows. With ROCRSSI, anchors on the edge reduce the coverage of rings since a ring will be cut by the edges and the usable range of a ring with ROCRSSI is only half of the ring at most. In contrast, placing anchors within the terrain can fully exploit a ring because the whole ring can be used potentially by several sensors.

Nevertheless, the differences of average location errors among different deployment strategies are not significant when ANR is large. When deployment strategies do impair the performance of ROCRSSI, the impact can be controlled effectively by increasing the radio range of the anchors. In the simulation, we used uniform deployment inside terrain to obtain the best accuracy in each scenario.

*b) Varying DOI:* Figure 9 shows that, with the increase of DOI, the average estimation error increases. This is because, with the increase of DOI, radio propagation has large variation along different directions. As a result, a ring generated by the comparison of signal strength may not represent the real area containing a sensor. Estimation based on incorrect rings is likely to have errors.

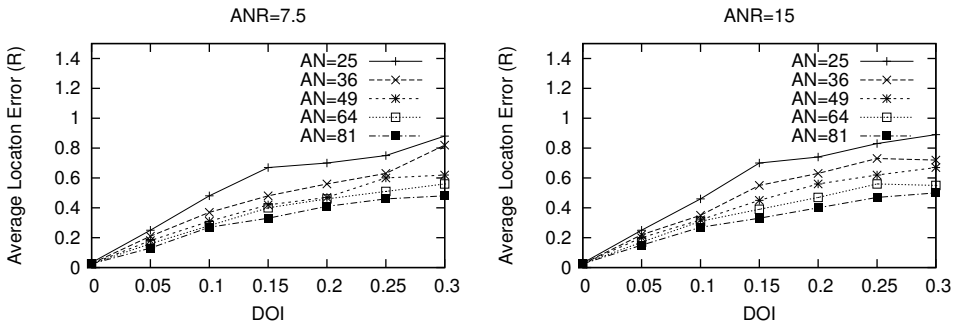


Fig. 9. Average location errors vs. DOI with anchors uniformly deployed inside terrain

Fortunately, as explained in Section III, the grid-scan algorithm used to calculate the center of gravity of the intersection area helps reduce the influence of incorrect rings. As a result, ROCRSSI can achieve accurate estimation, even with DOI as large as 0.2. For instance, from Figure 9, when AN= 81 and DOI= 0.2, ROCRSSI can achieve estimation with average error of 0.4R, which is acceptable for most applications [9].

As shown in Figure 9, the performance with ANR value of 15 is very similar to the performance with ANR value of 7.5, demonstrating that, when ANR value is large enough, increasing this value further provides little benefit. This phenomenon is also confirmed by later simulation as shown in Figure 11.

*c) Varying AN:* Figure 10 shows that increasing AN can reduce the average estimation error. With more deployed anchors, a sensor node can hear more anchors and generate more rings. As a result, the intersection area becomes smaller, resulting in more accurate estimation.

*d) Varying ANR:* Figure 11 illustrates that, with a perfectly circular radio propagation model (DOI= 0), increasing ANR reduces the average estimation error to the minimum possible with the grid granularity in use. The reason is that increasing ANR will increase the number of audible anchors per sensor node, and thus more rings will be generated. The circular radio propagation model guarantees that *each ring will certainly contain the sensor node* and can be

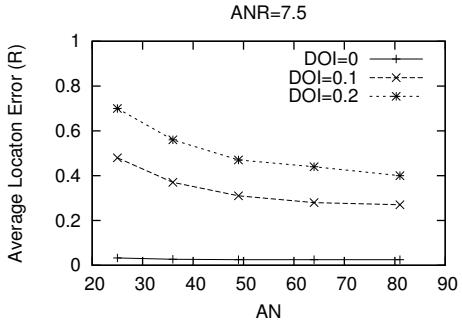


Fig. 10. Average location errors vs. AN with anchors uniformly deployed inside terrain

used to narrow down the final intersection area, resulting in very accurate estimation.

Figure 11 also shows that, when radio propagation is irregular, as ANR increases, the average estimation error first decreases and then oscillates around a fixed value. With an irregular radio propagation model, increasing ANR has two effects: (1) the number of audible anchors per sensor node increases and (2) more incorrect rings might be generated by possible anchor pairs in different directions. When ANR is small, the first effect dominates, and thus the average estimation error can be reduced. When ANR is large enough, however, both factors can influence the final outcome, causing the oscillation.

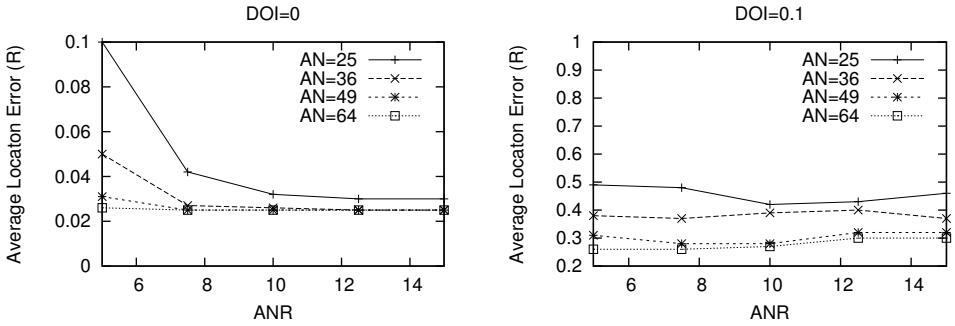


Fig. 11. Average location errors vs. ANR with anchor uniform deployment inside terrain

## VI. Comparison between APIT and ROCRSSI

We were unable to find implementations for other range-free localization algorithms [9], [15], [16]. Consequently, simulation is the best available means for performance comparison between ROCRSSI and other range-free algorithms. We compare the performance of ROCRSSI and

Approximate Point In Triangle (APIT) [9], because APIT [9] performs extremely well with randomly deployed wireless sensor networks and also because there are important similarities between APIT and ROCRSSI.

### *A. Introduction to APIT*

Like other range-free approaches, APIT depends on some nodes, whose locations are known in advance, serving as anchors. Because the number of anchor nodes is usually small, each node can use the Global Positioning System (GPS) to determine its location and can be equipped with a powerful radio transceiver. Using beacon messages from these anchors, APIT divides the neighbourhood area of a sensor node into many overlapped triangles. The vertexes of these triangles are anchors this sensor node can hear. The sensor node can narrow down the area in which it may reside by checking whether it falls within a triangle. Utilizing information from all anchors in range, the size of the intersection area of triangles can be reduced, and the center of gravity of this area is taken as the sensor's approximate location. As shown in Figure 12, the shadowed area is the intersection area of all perceived triangles and the center of gravity of the shadowed area is taken to be the sensor's location. A sensor node is considered undetermined if the node does not fall within any triangle.

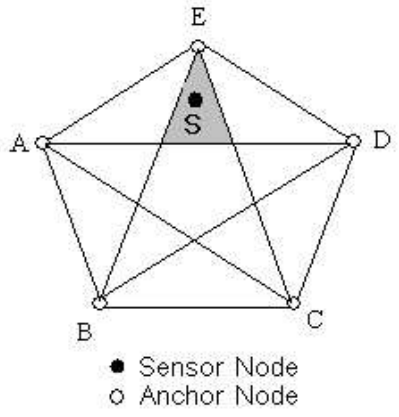


Fig. 12. An example of triangle overlapping in APIT

Therefore, a sensor must have the ability to check whether it falls inside or outside a triangle. As shown in Figure 13, a known theorem can be used for such a test:

> If there exists a direction such that a point adjacent to $M$ is further/closer to points $A$, $B$, and $C$ simultaneously, then $M$ is outside of $\triangle$ABC. Otherwise, $M$ is inside

△ABC [9].

Since a node cannot move around to carry out the above test, neighbouring sensor nodes exchange information to perform an approximate test, i.e., if a node has no neighbouring nodes further or closer to three anchors simultaneously, this node is assumed to fall within the triangle formed by the three anchors. Otherwise, it is assumed to lie outside the triangle.

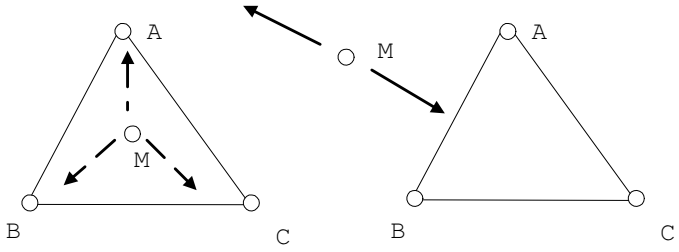


Fig. 13. Point In Triangle (PIT) test

It is easy to see that the accuracy of the above test depends greatly on how many neighbors a node has. Fortunately, in realistic wireless sensor networks, sensor nodes are typically densely deployed. APIT's accuracy also requires that two sensor nodes can decide their relative distance to an anchor by comparing the received signal strength from the anchor.

APIT is sensitive to node density and requires a fairly high node density for good performance [9]. In contrast, ROCRSSI does not perform message exchanges between neighboring sensors, making it insensitive to sensor radio transmission range and sensor density. While this is an advantage of ROCRSSI over APIT, it makes it difficult to perform an absolutely fair comparison between APIT and ROCRSSI.

Nevertheless, from the results in [9], when sensor density is above 15, i.e., a sensor node has 15 neighboring sensor nodes in average, APIT performs best and the estimation accuracy is relatively stable. To get a reasonably fair comparison, we set sensor density to approximately 15 by adjusting maximum sensor radio range in the simulation. We use the same extended DOI radio model to test APIT. Moreover, we adjust the anchor deployment strategy and ANR value for each method to make the two approaches achieve roughly their best performance with the same number of anchors.

### B. Average Location Error

Figure 14 demonstrates that ROCRSSI always outperforms APIT in terms of average estimation error, whether the radio propagation is regular or irregular. This is because the intersection of rings usually has smaller size than the intersection of triangles. A simple example can be found in Figure 15, where it is easy to see that the size of the shadowed ring intersection area is smaller than the size of $\triangle$ABC.

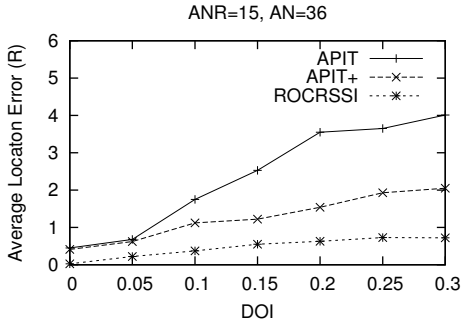


Fig. 14. The approximate comparison between APIT and ROCRSSI

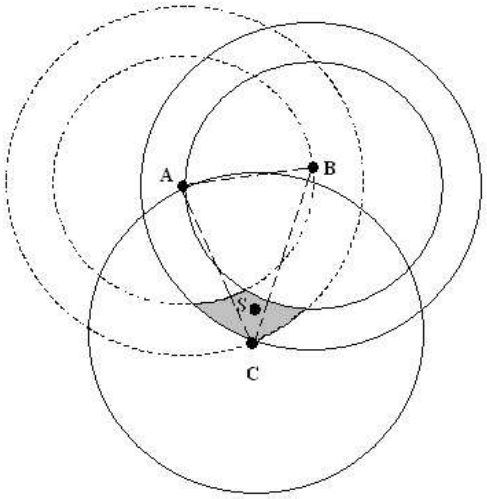


Fig. 15. Intersection of rings is smaller than intersection of triangles

The accuracy of APIT in our simulation is worse than that reported in [9] due to different radio models. In paper [9], the DOI radio model is used. In this model, radio irregularity influences only the number of immediate neighbors of each sensor node. If each pair of sensors can symmetrically

communicate correctly, APIT assumes that they can always make a correct decision on whether one is closer to or further from a certain anchor. In the extended DOI model, however, radio irregularity can influence not only the number of immediate neighbors of each sensor, but also the correctness of their decision on their relative distance to a certain anchor by the comparison of received signal strength. Therefore, in the extended DOI model, APIT will make more errors when deciding whether a node is inside or outside a certain triangle. This factor can significantly decrease the accuracy of APIT, resulting in worse performance than that reported in [9].

We also simulated an improved version of APIT, denoted as APIT+ in Figure 14. Using the grid-scan algorithm, when a triangle is added, only those grid elements within the maximum range of all anchors in range are added. This method improves the performance of APIT in the face of radio irregularity with extra checking for each grid element.

### *C. The Undetermined Nodes Problem*

Besides the improvement in accuracy, ROCRSSI can eliminate the inherent undetermined nodes problem present in APIT, regardless of the anchor deployment strategy. We verify this claim by considering the worst cases in ROCRSSI.

We have studied all four deployment strategies with ROCRSSI and observed that random deployment of anchors inside terrain can yield more undetermined nodes than uniform deployment of anchors inside terrain, and random deployment of anchors at the edges of terrain can yield more undetermined nodes than even deployment of anchors at the edges. Therefore, we only present the results for two deployment strategies: random deployment of anchors inside terrain and random deployment of anchors at the edges of terrain.

Figure 16 shows the ratio of undetermined nodes under the above deployment strategies with different ANR and DOI values. The figures show that, if ANR is larger than a threshold value (for example, $7.5$ in the simulation), there will be no undetermined nodes.

### *D. Communication Overhead*

Consider a sensor network with $N$ anchors and $M$ sensor nodes. Suppose that each broadcast from a sensor node costs 1 energy unit. Since the radio range of each anchor node is ANR times of that of a sensor node, each broadcast from anchors costs roughly $(\text{ANR})^2$ energy units if we assume a free-space radio propagation model. In APIT, all nodes broadcast at least once

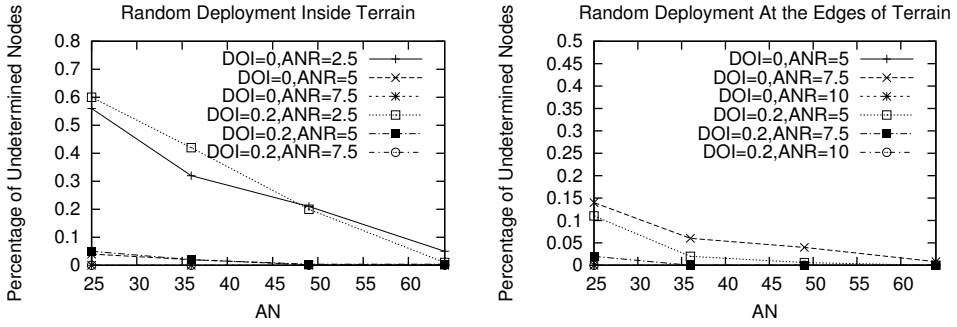


Fig. 16. Percentage of undetermined nodes (missing curves mean no undetermined nodes.)

under a collision-free situation, so the total number of broadcast messages from sensor nodes is $\Theta(M)$ and that from anchor nodes is $\Theta(N)$, where $\Theta$ means asymptotically tight bound. Therefore, the total energy consumption is $\Theta(M) + \Theta(N) \times (\text{ANR})^2$ energy units. In ROCRSSI, only anchors need to broadcast and sensor nodes do not need to exchange any information for localization. So the total number of broadcast messages is $\Theta(N)$ and the total energy consumption is $\Theta(N) \times (\text{ANR})^2$ energy units. In cases where $M \gg N$ and ANR is not very large, ROCRSSI consumes much less energy than APIT with the same number of anchors deployed.

## VII. Related Work

Due to the stringent constraints of energy consumption, cost, and bandwidth on wireless sensor devices, traditional localization approaches are usually not suitable. Many localization approaches have been proposed for wireless sensor networks to provide per-node location information. These approaches can be divided into two main categories: range-based methods, which depend on estimating point-to-point distance (range) and range-free methods, which do not depend on absolute range information. In both methods, a small number of nodes with known locations serve as anchor nodes.

Range-based approaches estimate the absolute distance between a sensor node and the anchors. In most cases, triangulation is used to compute the locations of sensor nodes. Some approaches map the received radio signal strength directly to the distance between the sender and the receiver [2], [10]. However, due to the effects of factors such as multi-path fading, reflection, diffraction, scattering, and background interference, radio signal propagation demonstrates great irregularity [6]. It is hard to get accurate distance estimation based purely on RSSI. To improve

estimation accuracy, range-based approaches usually take advantage of averaging, smoothing, and other hybrid techniques.

The Time Difference of Arrival (TDOA) technique [18] presents another means to get range estimation with better accuracy than pure RSSI-based techniques. The main idea is to calculate the distance based on two distinct communication modalities: ultrasound and radio. Ultrasound signals and radio signals travel at vastly different speeds. The radio signal is used for synchronization and the ultrasound signal for ranging. The TDOA method works well but requires extra hardware in sensor nodes.

In contrast to range-based approaches, range-free approaches do not depend on received signal strength from a sender to a receiver to estimate their absolute distance. Locations are estimated by utilizing relationships between sensor nodes and anchor nodes including *relative* distance, connectivity, and hop-count information. Range-free localization methods include the centroid algorithm [3], mere connectivity based localization [19], the DV-hop approach [16], and the amorphous positioning algorithm [15].

In the Centroid algorithm [3], a sensor node takes the centroid of all anchors in range as its estimated location. If all the anchors are deployed at the intersection points of a grid, the average location estimation error is about one-third of the distance between two anchors [3]. Peer-to-peer communication in the networks is modeled as a set of geometric constraints on node locations. Convex optimization is used to solve the constraints problem and the global solution for all constraints yields location estimation for sensor nodes.

In [19], mere connectivity information and a multi-dimensional scaling algorithm are used for location estimation. Multi-dimensional scaling is a type of data analysis technique which requires $O(n^3)$ calculation complexity for a network of $n$ nodes.

The DV-hop approach [16] estimates the distance between two nodes by translating the hop count between them. Initially, anchors flood their location information to all the nodes in the network. Each node, sensor or anchor, maintains a minimum hop count to each anchor in range. Then, an anchor infers the distance to another anchor, divides the distance by the hop count to that anchor, and distributes the information of the average distance per hop count to the network via controlled flooding. The average distance per hop count is called "correction information" and is used by nearby sensors to map hop counts to distances. A sensor node uses the latest correction information to estimate distances to anchors and performs multi-lateration to three or

more anchors to estimate its own location.

The amorphous positioning algorithm [15] is very similar to the DV-hop approach. The main difference concerns the way of computing average per-hop distance. Instead of using simple division to get average per-hop distance, the amorphous positioning algorithm calculates average per hop distance using the Kleinrock and Sliverster formula [12].

Unlike the above range-free approaches, APIT and ROCRSSI use a totally different way for location estimation: a sensor decides whether it falls within a sequence of regular geometric areas—triangles in APIT and rings in ROCRSSI—by comparing received signal strength, and shrinks the estimated area by intersection.

## VIII. Conclusions

Range-free localization has great potential in wireless sensor networks. In this paper, we present ROCRSSI, a new range-free localization method, based on generating overlapping rings and calculating their intersection. The rings are formed with the comparison of received signal strength from anchor nodes.

We implemented and tested the localization algorithm with MICA2 Motes. We also performed a simulation study using experimentally determined radio propagation model to investigate algorithm performance in large-scale networks. The results show that our range-free localization method is resilient to irregular radio propagation patterns and can achieve better performance with less communication overhead, compared to another well-known range-free method, APIT.

## References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A Survey on Sensor Networks," *IEEE Communications Magazine*, Vol. 40, No. 8, August 2002, pp. 102-114.

[2] P. Bahl and V. N. Padmanabhan, "RADAR: An In-Building RF-Based User Location and Tracking System," *Proceedings of the IEEE INFOCOM 2000*, Telaviv, Israel, March 2000.

[3] N. Bulusu, J. Heidemann and D. Estrin, "GPS-less Low Cost Outdoor Localization for Very Small Devices," *IEEE Personal Communications Magazine*, newblock Vol. 7, No. 5, October 2000, pp. 28-34.

[4] N. Bulusu, J. Heidemann and D. Estrin, "Density Adaptive Algorithms for Beacon Placement in Wireless Sensor Networks," *Proceedings of the IEEE ICDCS 2001* , newblock Phoenix, AZ, April 2001.

[5] L. Doherty, K. Pister, and L. E. Ghaoui, "Convex position estimation in wireless sensor networks," *Proceeding of IEEE Infocom 2001,* Anchorage, Alaska, April 2001.

[6] D. Ganesan, D. Estrin, A. Woo, D. Culler, "Complex Behaviour at Scale: An Experimental Study of Low-Power Wireless Sensor Networks, " *Technical Report UCLA/CSD-TR 02-0013*, 2002.

[7] L.Girod and D.Estrin, "Range Estimation using Acoustic and Multimodal Sensing," *Proceedings of IROS 2001,* Maui, Hawii, October 2001.

[8] H. Gupta, S. R. Das, and Q. Gu, "Connected Sensor Cover: Self-Organization of Sensor Networks for Efficient Query Execution," textitProceeding of MobiHoc 2003, Annapolis, Maryland, June 2003.

[9] T. He, C. Huang, B.M. Blum, J. A. Stankovic, and T. Abdelzaher, "Range-Free Localization Schemes for Large Scale Sensor Networks," *Proceedings of the ninth annual international conference on Mobile computing and networking (MobiCom 2003)*, San Diego, California, September 2003, pp. 81-95.

[10] J. Hightower, G. Boriello and R. Want, "SpotON: An Indoor 3D Location Sensing Technology Based on RF Signal Strength," *University of Washington CSE Report 2000-02-02*, February 2000.

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

[12] L.Kleinrock and J.Silvester, "Optimum Transmission Radii For Packet Radio Networks or Why Six Is a Magic Number," *Proceedings of National Telecomm Conference,* Pages 4.3.1-4.3.5, 1978.

[13] Y.B. Ko and N. H. Vaidya, "Location-Aided Routing (LAR) Mobile Ad Hoc Networks," *Proceedings of the fourth annual international conference on Mobile computing and networking (MobiCom 98)*, Dallas, Texas, October 1998, pp. 66-75.

[14] S. Meguerdichian, F. Koushanfar, M. Potkonjak, and M.B. Srivastava, "Coverage Problems in Wireless Ad-hoc Sensor Neworks," Proceedings of IEEE Infocom 2001, Ankorange, Alaska, April 2001.

[15] R. Nagpal, "Organizing a Global Coordinate System from Local Information on an Amorphous Computer," *A.I. Memo1666,* MIT A.I. Laboratory, August 1999.

[16] D. Niculescu and B. Nath, "DV Based Positioning in Ad hoc Networks," *Journal of Telecommunication Systems*, Vol. 1 2003.

[17] C.Savarese, J.Rabay and K.Langendoen, "Robust Positioning Algorithm for Distributed Ad-Hoc Wireless Sensor Networks," *USENIX Technical Annual Conference,* Monterey, California, June 2002.

[18] A. Savvides, C. Han, and M. B. Strivastava, "Dynamic Fine-Grained Localization in Ad-hoc Networks of Sensors," *Proceedings of the seventh annual international conference on Mobile computing and networking (MobiCom 01)*, Rome, Italy, July 2001.

[19] Y. Shang, W. Ruml, Y. Zhang, "Localization From Mere Connectivity," *Proceedings of MobiHoc 2003,* Annapolis, Maryland, June 2003.

[20] T. Scott, K. Wu, and D. Hoffman, "Radio Propagation Patterns in Wireless Sensor Networks: New Experimental Results," *Proceedings of International Wireless Communications and Mobile Computing Conference,* Vancouver, Canada, July.

[21] K.Whitehouse adn D. Culler, "Calibration as Parmeter Estimation in Sensor Networks," *First ACM International Workshop on Wireless Sensor Networks and Applications,* Atalanta, Georgia, September 2002.

[22] J. Zhao and R. Govindan, "Understanding Packet Delivery Performance in Dense Wireless Sensor Networks," *Proceedings of ACM SenSys 2003,* Los Angeles, California, November, 2003.

[23] G. Zhou, T. He, J. Stankovic, "Impact of Radio Asymmetry on Wireless Sensor Networks," *Proceedings of MobiSys 2004,* Boston,Massachusetts, June, 2004.

# APPENDIX A - TEST RESULTS

TABLE I

VERIFICATION RESULTS WITH IDEAL RSSI VALUES

| Sensor Position | Anchor Positions | | | Predicated Location | Actual Location | Error (m) |
|---|---|---|---|---|---|---|
| | $A$ | $B$ | $C$ | (X, Y) | (X,Y) | |
| $S_1$ | $(17, 15)$ | $(19, 18)$ | $(29, 15)$ | $(24, 16)$ | $(25, 16)$ | 1.00 |
| $S_2$ | $(17, 15)$ | $(19, 18)$ | $(29, 15)$ | $(24, 24)$ | $(50, 16)$ | 27.20 |
| $S_3$ | $(17, 15)$ | $(19, 18)$ | $(29, 15)$ | $(20, 16)$ | $(21, 23)$ | 7.07 |
| $S_4$ | $(17, 15)$ | $(19, 18)$ | $(29, 15)$ | $(13, 14)$ | $(12, 6)$ | 8.06 |
| $S_5$ | $(17, 15)$ | $(19, 18)$ | $(29, 15)$ | $(15, 15)$ | $(16, 13)$ | 2.24 |
| $S_6$ | $(17, 15)$ | $(19, 18)$ | $(29, 15)$ | $(32, 15)$ | $(33, 12)$ | 3.16 |
| $S_7$ | $(17, 15)$ | $(19, 18)$ | $(29, 15)$ | $(25, 10)$ | $(25, 7)$ | 3.00 |
| $S_8$ | $(17, 15)$ | $(19, 18)$ | $(29, 15)$ | $(32, 15)$ | $(30, 17)$ | 2.83 |
| $S_9$ | $(17, 15)$ | $(19, 18)$ | $(29, 15)$ | $(18, 16)$ | $(18, 16)$ | 0.00 |
| $S_{10}$ | $(17, 15)$ | $(19, 18)$ | $(29, 15)$ | $(20, 18)$ | $(20, 18)$ | 0.00 |
| $S_{11}$ | $(17, 15)$ | $(19, 18)$ | $(29, 15)$ | $(24, 16)$ | $(25, 14)$ | 2.24 |
| Average Error = 5.16 | | | | | | |
| Average Error with undetermined node $S_2$ removed = 3.18 | | | | | | |

TABLE II

COMPARISON OF TEST RESULTS OF $S_1$ WITH DIFFERENT TEST RANGES

| Test # | Estimated Position (X,Y) | | Error (m) | |
|---|---|---|---|---|
| | Rooftop Range | Yard Range | Rooftop Range | Yard Range |
| | (units = 0.5m) | (units = 1m) | | |
| 1 | (23, 16) | (23, 16) | 1 | 2 |
| 2 | (24, 16) | (23, 16) | 0.5 | 2 |
| 3 | (24, 16) | (23, 16) | 0.5 | 2 |
| 4 | (24, 16) | (23, 16) | 0.5 | 2 |
| 5 | (24, 16) | (23, 16) | 0.5 | 2 |
| 6 | (24, 16) | (23, 16) | 0.5 | 2 |
| 7 | (24, 16) | (23, 16) | 0.5 | 2 |
| 8 | (24, 16) | (23, 16) | 0.5 | 2 |
| 9 | (24, 16) | (23, 16) | 0.5 | 2 |
| 10 | (24, 16) | (23, 16) | 0.5 | 2 |
| 11 | (24, 16) | (23, 16) | 0.5 | 2 |
| 12 | (24, 16) | (23, 16) | 0.5 | 2 |
| 13 | (24, 16) | (23, 16) | 0.5 | 2 |
| 14 | (24, 16) | (23, 16) | 0.5 | 2 |
| 15 | (24, 16) | (23, 16) | 0.5 | 2 |
| 16 | (24, 16) | (23, 16) | 0.5 | 2 |
| 17 | (24, 16) | (24, 24) | 0.5 | 8.06 |
| 18 | (24, 16) | (23, 16) | 0.5 | 2 |
| 19 | (24, 16) | (23, 16) | 0.5 | 2 |
| 20 | (24, 16) | (23, 16) | 0.5 | 2 |
| 21 | (24, 16) | (23, 16) | 0.5 | 2 |
| 22 | (24, 16) | (23, 16) | 0.5 | 2 |
| | | | Average error with rooftop range = 0.52m | |
| | | | Average error with yard range = 2.25m | |

TABLE III

TEST RESULTS OF $S_8$, $S_9$, AND $S_{10}$ WITH THE YARD RANGE

| Actual Location of $S_8$ : (30,17), Average Error: 7.07m<br>Actual Location of $S_9$ : (18,16), Average Error: 3.2m<br>Actual Location of $S_{10}$ : (20,18), Average Error: 0.17m | | | | | | |
|---|---|---|---|---|---|---|
| Test # | Estimated Location (X, Y) | | | Error (m) | | |
| | $S_8$ | $S_9$ | $S_{10}$ | $S_8$ | $S_9$ | $S_{10}$ |
| 1 | (23, 16) | (19, 13) | (20, 18) | 7.07 | 3.16 | 0 |
| 2 | (24, 24) | (18, 13) | (20, 18) | 9.22 | 3.00 | 0 |
| 3 | (23, 16) | (18, 13) | (20, 18) | 7.07 | 3.00 | 0 |
| 4 | (23, 16) | (18, 13) | (20, 18) | 7.07 | 3.00 | 0 |
| 5 | (23, 16) | (18, 13) | (20, 18) | 7.07 | 3.00 | 0 |
| 6 | (23, 16) | (19, 13) | (20, 18) | 7.07 | 3.16 | 0 |
| 7 | (23, 16) | (18, 13) | (24, 16) | 7.07 | 3.00 | 4.47 |
| 8 | (23, 16) | (19, 13) | (20, 18) | 7.07 | 3.16 | 0 |
| 9 | (26, 20) | (18, 13) | (20, 18) | 5.00 | 3.00 | 0 |
| 10 | (23, 16) | (24, 16) | (20, 18) | 7.07 | 6.00 | 0 |
| 11 | (23, 16) | (19, 13) | (20, 18) | 7.07 | 3.16 | 0 |
| 12 | (23, 16) | (19, 13) | (20, 18) | 7.07 | 3.16 | 0 |
| 13 | (23, 16) | (18, 13) | (20, 18) | 7.07 | 3.00 | 0 |
| 14 | (23, 16) | (19, 13) | (20, 18) | 7.07 | 3.16 | 0 |
| 15 | (23, 16) | (18, 13) | (20, 18) | 7.07 | 3.00 | 0 |
| 16 | (23, 16) | (19, 13) | (20, 18) | 7.07 | 3.16 | 0 |
| 17 | (23, 16) | (18, 13) | (20, 18) | 7.07 | 3.00 | 0 |
| 18 | (23, 16) | (18, 13) | (20, 18) | 7.07 | 3.00 | 0 |
| 19 | (23, 16) | (19, 13) | (20, 18) | 7.07 | 3.16 | 0 |
| 20 | (23, 16) | (19, 13) | (20, 18) | 7.07 | 3.16 | 0 |
| 21 | (23, 16) | (19, 13) | (20, 18) | 7.07 | 3.16 | 0 |
| 22 | (23, 16) | (19, 13) | (20, 18) | 7.07 | 3.16 | 0 |
| 23 | (23, 16) | (19, 13) | (20, 18) | 7.07 | 3.16 | 0 |
| 24 | (23, 16) | (18, 13) | (20, 18) | 7.07 | 3.00 | 0 |
| 25 | (23, 16) | (19, 13) | (20, 18) | 7.07 | 3.16 | 0 |
| 26 | (23, 16) | (18, 13) | (20, 18) | 7.07 | 3.00 | 0 |
| 27 | (23, 16) | (19, 13) | (20, 18) | 7.07 | 3.16 | 0 |