Localization with Snap-Inducing Shaped Residuals (SISR) - Coping with Errors in Measurement

H.T. Kung, Chit-Kwan Lin, Tsung-Han Lin, Dario Vlah

TR-03-09

Computer Science Group
Harvard University
Cambridge, Massachusetts

Localization with Snap-Inducing Shaped Residuals (SISR) - Coping with Errors in Measurement

H.T. Kung, Chit-Kwan Lin, Tsung-Han Lin, Dario Vlah
School of Engineering and Applied Sciences
Harvard University
Cambridge, MA 02138, USA
{htk, cklin, thlin, dario}@eecs.harvard.edu

Abstract

We consider the problem of localizing wireless nodes in an outdoor, open-space environment, using ad-hoc radio ranging measurements, e.g., 802.11. As in other range-based methods, we cast ranging measurements as a set of distance constraints, thus forming an over-determined system of equations suitable for non-linear least squares optimization. However, ranging measurements are often subject to errors, induced by multipath signals and variations in path loss, or even faulty hardware or antenna connectors. Including such potentially large and non-Gaussian errors in the measurement data ultimately produces inaccurate localization solutions. We propose a new method, called *snap-inducing shaped residuals* (SISR), to automatically identify "bad nodes" and "bad links" arising from these errors, so that they receive less weight in the localization process. In particular, SISR snaps "good nodes" to their accurate locations and gives less emphasis to other nodes. While the mathematical techniques used by SISR are similar to those in robust statistics, SISR's exploitation of the snap-in effect in localization appears to be novel. We provide analysis on the principle of SISR, and demonstrate a working SISR implementation in field experiments on a testbed of 20 wireless nodes, as well as the superior performance of SISR in simulation with a larger number of nodes.

Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design—*Wireless communication*

General Terms

Algorithms, Analysis, Experimentation, Measurement, Performance

Keywords

Localization, Least Squares Optimization, Wireless Ranging, 802.11, Robust Statistics

1 Introduction

Power-based localization, which relies on range estimation via received signal strength of RF signals, is convenient in ad hoc networks of wireless nodes. For instance, 802.11 hardware enjoys widespread deployment and tends to operate in an always-on manner with at least a minimal amount of protocol transmissions. Furthermore, power-based ranging does not require precise time synchronization between network nodes. Compared to audio-based ranging, RF ranging can cover a longer distance and, relative to GPS, it is often faster and can be a useful alternative in environments where GPS signals are blocked or jammed.

As a general technique, localization is useful in many ad-hoc networking applications. For example, it can improve various network services such as routing and security provisioning, or help applications such as acoustic beamforming and speaker tracking to find good reference nodes. Additionally, self-clustering can be performed on localization information to uncover cluster structures in an ad-hoc network, enabling cluster-driven protocols and applications.

In this paper, we consider the problem of using 802.11 radio ranging to localize ad-hoc nodes in an outdoor, open-space environment. When the terrain is flat and nodes can all hear each other well, solving the problem ought to be relatively easy. However, this may not be the case in real-word applications. For example, a node in a pit, with a faulty antenna, or at the edge of radio range to other nodes may incur large errors in its distance measurements due to weak signal from the node or received by the node. In addition, heterogeneous radio equipment may also introduce inconsistency in measurement errors. Such errors can create asymmetric ranging measurements between nodes and, often, are non-Gaussian in nature. Just a few such instances can drastically affect a localization solution.

We propose a technique to cope with the presence of a relatively small number of possibly large, non-Gaussian errors in ranging measurements. The method, called "least squares with snap-inducing shaped residuals," or *snap-inducing shaped residuals* (SISR) for short, emphasizes those computed node locations which match well with ranging measurements, while de-emphasizing others. That is, localization based on SISR will favor those computed locations which represent good matches for a majority of measurement data, even if these locations may mean a large deviation from a minority.

We perform analysis, simulation and field experiments to provide insights and validation on how SISR works. The two major contribution of this paper are:

1. The SISR technique to provide accurate localization for those nodes which have sufficient, good mea-

surement data, even in the presence of large errors in measurement data for other nodes. SISR's effectiveness is validated via simulation and outdoor field experimentation.

2. RSSI-distance modeling and field validation of the model of a particular 802.11 radio in an open, outdoor environment.

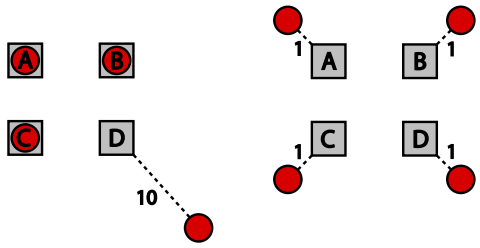
2 Related Work

Our goal of extracting geometry information from a network of wireless nodes is similar to that of self-localization in wireless sensor networks. Existing solutions fall into two major categories: range-free and range-based localization. Typically, range-free localization methods [12, 8, 17, 15] utilize node connectivity and hop-count information along with geometric constraints to determine node locations. Aiming at providing more accurate localization results, range-based localization methods [14, 13, 4] acquire pairwise distance measurements between wireless nodes and use this information to derive a localization solution. Typical ranging techniques include measuring time-of-arrival (TOA) and received signal strength indications (RSSIs).

A major issue for range-based localization schemes is error in distance (ranging) measurements. Whitehouse et al. [19] have indicated that the errors of RSSI-based RF ranging do not follow a Gaussian distribution, implying that conventional least squares optimization schemes are particularly susceptible to such errors. Without properly taking this into account, localization results can be drastically altered or biased by only a few bad measurements.

Error mitigation in range-based wireless localization has recently received some attention in the literature, with several groups proposing iterative and incremental distributed localization algorithms. Liu et al. [10] use an explicit error management approach to prevent error propagation during incremental localization, and propose a modified least-squares objective function that includes a perturbation term such that the difference in measurement data can be minimized in an average sense, resulting in less sensitivity to errors. Other research efforts have taken a different tack, focusing on identifying and exploiting rigid network topologies to defend against flip and discontinuous flex ambiguities in a localization solution [6, 11]. In this paper, we assume a dense network with a large number constraints, where flip and flexing ambiguities are less likely to happen.

Like SISR, other methods such as semidefinite programming (SDP) [4] and multidimensional scaling (MDS) [16] have taken a centralized approach to the localization problem. Centralized approaches can be adopted in ad-hoc networks with an additional step of collecting data to a master node, but, in comparison with distributed methods, have advantages such as fewer information exchanges between individual nodes and more efficient computation. Reducing inter-node communication messages is especially significant for a wireless ad-hoc network, since wireless medium is shared. SDP maps the localization problem to a convex optimization problem and is able to locate nodes in the presence of errors, but does not consider discounting the outliers. Similarly, MDS-based techniques are robust against some forms of measurement errors. By replacing all-pairs ranging measurements with a multi-hop shortest path distance, some of overestimated distances are excluded or reduced in the localization process. However, this strategy does not perform well for non-convex topologies, where ends of a curved topology can be deformed due to overestimation by the shortest path distance. We will revisit this issue later in the paper.



(a) Uneven (b) Even

Figure 1: Two possible localization solutions of nodes A, B, C, and D, where D has large measurement errors. Squares indicate the ground truth location; circles, the computed localization solution. (a) Solutions for A, B, C are very accurate, but that for D is very inaccurate (magnitude "10"), since D's measurement error is not amortized over the other nodes. (b) The measurement error from D is amortized over A, B, and C, causing them to be localized with decreased accuracy (magnitude "1").

SISR is a non-linear least squares-based optimization method, with a twist. To cope with errors, SISR exploits geometric information to improve the robustness of estimation algorithms. Given that each node can have only one location, SISR dampens the effect of inconsistent distance measurements while rewarding and amplifying that of consistent distance estimates. With a sufficient amount of consistent data (i.e., good range estimates), SISR is able to determine node locations accurately by ignoring the polluting effects of bad measurements. The SISR estimator is similar to those found in "robust statistics" [7, 9], the study of outlier rejection problems and techniques.

Although similar, the SISR estimator differs from those previously proposed in robust statistics in some important details, as described in Section 4. These differences make SISR localization results more accurate for snapped-in nodes and also make the convergence of the method more robust. More importantly, to our knowledge, SISR appears to be the first work in applying robust statistics in localization and in using "snap-in" for good nodes as a localization objective, as described in Section 3. Robust statistics has been used in image processing and pattern recognition, for such applications as object velocity estimation from a sequence of images [5], anisotropic diffusion [**?**], and bilateral filtering [**?**].

3 Motivation

Consider a four node scenario where nodes A, B and C incur small measurement errors on their links, while node D incurs large errors. Thus, A, B and C are "good nodes", and D is a "bad node." A localization method could lead to one of the following two solutions, as shown in Figure 1:

- **Solution *UNEVEN***, which localizes A, B and C accurately, but misses D by a large margin.
- **Solution *EVEN***, which localizes D with the same order of accuracy as A, B, and C, but achieves this at the expense of accurately locating A, B and C.

We prefer solution *UNEVEN* to *EVEN*, since D cannot be localized accurately in any case, given that its links have large measurement errors. Moreover, since A, B, and C could potentially be localized with great accuracy, a localization method that returns solution *UNEVEN* ought to de-emphasize D's measurements in order to avoid polluting the localization solutions of A, B, and C. A conventional least squares method would find solution *EVEN*, since it does not differentiate between good and bad ranging measurements.

Developing a localization method that can locate "good nodes" accurately, by discerning "bad nodes," is the motivation behind our work. An assumption we make throughout is that good nodes will represent a majority of nodes. In Section 4.3, we will show a break-down analysis on the maximum percentage of "bad nodes" allowed. SISR makes a key modification to the conventional least squares method: the residual function is *shaped*. Our resulting *snap-inducing shaped residuals (SISR) method* is capable of differentiating between good and bad nodes automatically and, consequently, is able to find the desired solution *UNEVEN*. Furthermore, by applying the method iteratively, SISR can achieve greater accuracy in localization for good nodes.

The failure mode that we consider in this work—that of the "bad node"—differs from the typical failure mode found in the literature, where individual directional links are assumed to experience ranging errors determined by an error model that is independent and identical across links. In practice, physical effects such as multi-path phenomena, shadowing or even faulty hardware can produce gross error that is insufficiently captured by such a single, link-scope error model. In addition, such effects can give rise to correlated ranging errors (e.g., a node that has both faulty transmit and receive antennas) which simple, unidirectional link error models cannot capture. Additionally, concentrating ranging errors at a particular node increases the difficulty of discerning good measurements from bad, making the ultimate task of accurately localizing the node even harder. The effects of these types of errors is not yet well-understood and, to our knowledge, no localization methods prior to SISR have been developed to specifically defend against such errors.

4 Snap-Inducing Shaped Residuals (SISR)

In this section we present a novel kind of residual function for use in optimization-based localization. We will say that the residual function is *snap-inducing* due to its tendency to preserve residuals smaller than some threshold, while diminishing the effect of the larger ones. We give a theoretical analysis of the SISR estimator and show that the properties of SISR can be explained by the theory of robust statistics.

4.1 The SISR Estimator

The localization problem is typically framed as a nonlinear least squares optimization, where ranging measurements are used as constraints, and a best fit is sought to minimize squared residuals. In the context of localization, the residual is defined as the difference between pairwise ranging measurements and the distances estimated by the estimator. Eq (1) shows the mathematical formulation of the residual $r(i,j)$ from a ranging measurement between nodes i and j.

$$r(i,j) \quad = \quad \widehat{d}_{ij} - d_{ij} \tag{1}$$

where $\widehat{d}_{ij}$ is the distance estimated by the least-squares estimator and d_{ij} is the ranging measurement between the two nodes.

Conventional least squares works by minimizing an objective function, which is the sum of squared residuals over all node pairs (i,j):

$$F \quad = \quad \sum_{i,j} r(i,j)^2 \tag{2}$$

The squared residual function is shown in Fig 2. The least-squares estimator is not robust to extremely noisy measurements because the squared residual grows quadratically. To solve the problem as well as produce the snapped-in effect for good nodes, we propose a new *snap-inducing* shaped residual (SISR) estimator, which has a shaping function that deemphasizes the influence of bad nodes while emphasizing the good ones. The function is sketched in Figure 2 and has the following two properties.

1. The shaping function increases with a smaller slope when the residual is large. In particular, the function dampens the impact of residuals larger than a threshold τ. We call this the *wing-shaped section*.
2. The shaping function has a narrow and deep well for residuals close to 0. The potential good measurements can therefore be emphasized by growing the shaped residuals or the "cost function" more rapidly. This is called the *U-shaped section*.

For the solutions $EVEN$ and $UNEVEN$ described in Section 3, Property 1 punishes the former by amplifying the the residuals in A, B and C, while Property 2 forgives the latter by diminishing the effect of the increased residual in D. This means SISR will favor the desired solution $UNEVEN$.

The general form of the SISR shaping function is shown in Eq (3):

$$s(i,j) = \begin{cases} \alpha r(i,j)^2 & \text{if } |r(i,j)| < \tau \\ \ln(|r(i,j)| - u) - v & \text{otherwise} \end{cases} \quad (3)$$

where α, τ, u, and v are parameters to be configured.

Tuning the shape of the SISR function controls its sensitivity to errors. The parameter α is introduced to control the height of the U-shaped section, while τ controls its width. (Note that the "U" portion has its x-axis values in the range [-τ, +τ].) The wing-shaped section is created by taking logarithms of the residuals larger than τ. Note that in robust statistics literature, a completely flat function is often used to produce similar robust behaviors [9]. However, a flat objective function with zero slope is usually difficult to handle numerically. Numerical algorithms such as Levenberg-Marquardt require a non-zero gradient of the objective function, and therefore we choose a slowly increasing log function instead. Another common requirement for numerical methods is that the objective function has to be continuous and differentiable. To make the SISR function piecewise-continuous and piecewise-differentiable at τ, the other two parameters u and v should have the form shown in Eq (4) and (5), respectively.

$$u = \tau - \frac{1}{2\alpha\tau} \quad (4)$$

$$v = \ln\left(\frac{1}{2\alpha\tau}\right) - \alpha\tau^2 \quad (5)$$

In short, the SISR function is controlled by two tunable parameters α and τ. The SISR estimator operates by minimizing the objective function as the sum of the shaped residuals over all node pairs (i,j).

$$F = \sum_{i,j} s(i,j) \quad (6)$$

Note that the function for residuals greater than τ need not be logarithmic; for example, we have found that radical functions (e.g., $x^{1/2}$) exhibit the above desired characteristics as well. The trade-off is in running time performance—flatter wings will generally increase the time a numerical method takes to converge to a solution but has a larger domain of convergence.

4.2 Effect of Error on SISR: A Simple Illustrative Simulation Result

To illustrate how SISR manages measurement errors, we give a basic example. Consider k nodes randomly deployed on a square field. Assuming all of the nodes are within radio range of each other, we can obtain $k(k-1)$ pairwise ranging measurements. Suppose only one of these ranging measurements has a different magnitude of error. Figure 3 shows, in simulation, how such errors affect conventional least-squares localization and SISR localization.

In the simulation, we placed 20 nodes on a 100×100 unit field. The performance of the two localization methods are evaluated using the average pairwise distance error for the $k-2$ nodes that the bad link is not incident on. We can see that the error obtained by least-squares continues to grow with the increasing error in the bad link. This is the standard least-squares behavior whereby large residuals are reduced by increasing any other smaller residuals. In effect, the bad link is causing the topology to flex, and so we call this a *flexed localization.* The error bars are not shown for clarity; however, the least-sqares result does have a large variance, because not every randomly chosen bad link has the same ability to flex the rest of the topology.

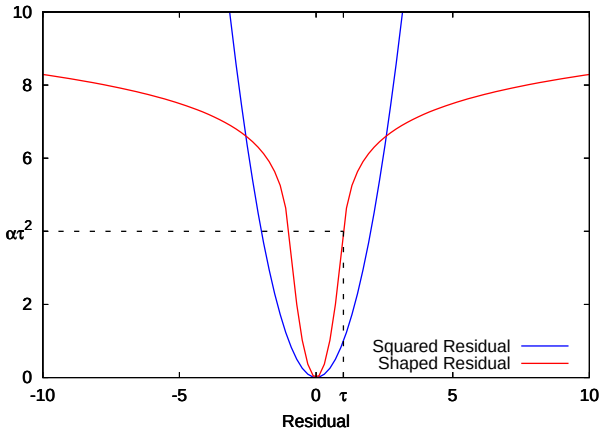


Figure 2: A comparison between the standard squared residual used in conventional least squares and our shaped residual, as in Eq (3) with $\alpha = 4, \tau = 1$ and with u and v as defined in (4) and (5). Note that the shaped residual function is shaped like a "U" near 0 and like "wings" for values larger than τ.

The SISR error behaves similarly up to a point. Specifically, as long as the distortion in any given residual is smaller than the threshold τ, according to (3), the residual's contribution will be same as least-squares. However, any distortions larger than τ will contribute diminishing residuals, leaving the topology "snapped into" its original shape; we can see this effect in Figure 3 as the SISR error departs from the least-squares error. In fact, as the bad link's error grows, its effect on the rest of the nodes *decreases*— this is because the slope of the bad link's increased residual is smaller, and so reducing the error on the bad link at the expense of other, low-error links becomes costlier than vice-versa.

4.3 Analysis of SISR and Breakdown Points

Here, we give a theoretical analysis of the SISR estimator and show how the snap-in effect emerges in SISR. Recall parameter τ is introduced to control the estimator's error sensitivity. τ is the residual value at which the "U" portion of the shaping function $s(r)$ attains its maximum. The shaped residual at τ is denoted as $s(\tau)$. To facilitate analysis, we introduce parameter η, defined as the maximum shaped residual we expect to encounter in a given localization scenario. By denoting the maximum residual

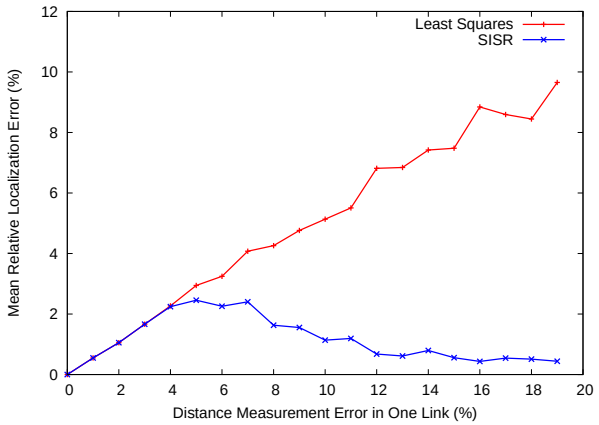


Figure 3: A simple illustrative experiment comparing localization performance of least-squares vs SISR under exactly one erroneous link. Each point is an average of 100 runs. The least-squares mean error grows along with the input error, while the SISR result remains "snapped" into place. (Note that localization error starts descending when measurement error is beyond a certain level, due to the decreasing slope of the "wings" portion of the SISR shaping function.)

as r_{upper}, then $\eta = s(r_{upper})$. Since the wing portion of the SISR shaping function is slowly increasing, the ratio of η to $s(\tau)$ is a small positive constant value $c > 1$. That is, $\eta = c \cdot s(\tau)$. Generally, the value of c is 2 to 3. In our simulations, where $\tau = 0.25$ and $\alpha = 80$ and the largest expected residual is 25m, $c = 2.38$. Even if we increase the largest expected residual to 100m, $c = 2.65$.

We use the same scenario as in Section 4.2 where k nodes are deployed on a field. Suppose that the ranging measurements for all the $k(k-1)$ links are accurate, and that all k nodes have been correctly localized without any error. That is, all the $k(k-1)$ links have zero residuals.

Now suppose that a node n_k changes its ranging measurements with the other $k-1$ nodes, with the change in each ranging measurement being no more than r_{upper}. We compare the estimation of the two different estimators: the SISR estimator, and the least-squares estimator. We refer to the localization solution using SISR as *snapped-in localization*, and that using least-squares as *flexed localization*.

(a) **Snapped-in localization.** All k nodes are snapped into their original locations, i.e., there are no changes in their locations. This means that distances between node n_k and the other $k-1$ nodes no longer match the corresponding changed ranging measurements. Thus, there is an increase in each of the $(k-1)$ residuals by an amount no greater than r_{upper}. In terms of the shaped residual for each link, each increases is no more than $s(r_{upper})$, i.e., η. Thus, the total increase in shaped residuals, $\Delta r_{snapped}$, must satisfy (7).

$$\Delta r_{snapped} \leq (k-1)\eta \tag{7}$$

(b) **Flexed localization.** In this solution, we assume here that all the k nodes change their locations by at least τ in distance, in order to adjust to the changed ranging measurements involving node n_k. Furthermore, we assume that the location changes are not synchronized in their directions, so each pair of the k nodes will observe a change in their distance by at least τ. Recall that only node n_k has changed its ranging measurements, and the ranging measurements for pairs in all the other nodes remain unchanged. Thus, each of these node pairs incurs an increase in its residual by at least τ. In terms of shaped residuals, the increase is at least $s(\tau)$ or η/c. Since there are $(k-1)(k-1)$ ranging measurements subject to change, the total increase in shaped residuals satisfies (8).

$$\Delta r_{flexed} \geq (k-1)^2 \frac{\eta}{c} \tag{8}$$

Note that the condition for $\Delta r_{flexed} > \Delta r_{snapped}$ is

$$k > c+1 \tag{9}$$

Thus, when k is sufficiently large, snapped-in localization will incur a smaller cost than flexed localization. This means that least squares with snap-inducing shaped residuals (SISR) has the desired property that it will find the snapped-in localization as a local minimum.

Following the same argument, we can also show that the snap-in effect occurs when there are multiple nodes changing their ranging measurements. In the case where p nodes change their ranging measurements rather than only one node n_k, Δr_{flexed} and $\Delta r_{snapped}$ can be shown to be

$$\Delta r_{snapped} \leq p(k-1)\eta \tag{10}$$

$$\Delta r_{flexed} \geq (k-p)(k-1)\frac{\eta}{c} \tag{11}$$

Suppose that b is the value of p at which Δr_{flexed} equals to $\Delta r_{snapped}$. Then b is the *breakdown point* of SISR. If there are more than b nodes that change their ranging measurements and the changes can be arbitrarily large, then SISR can no longer snap into the correct location. Solving p in the equation yields $p = b$ where

$$b = \frac{k}{1+c}. \tag{12}$$

Eq (12) indicates that the portion of bad nodes that SISR can tolerate is a function of the growth rate in the wing section, c. When $c = 1$, i.e., the shaping function has a completely flat wing portion, SISR can tolerate up to 50% bad nodes. In our formulation with a logarithmic wing portion, c is typically 2 to 3, and the breakdown point therefore is around 25% to 33%. Table 1 gives the breakdown points for different c values.

Table 1: Breakdown Points of SISR

c	1	1.5	2	2.5	3
Breakdown pt.	50%	40%	33%	29%	25%

Above, we assume that in flexed localization, all nodes slightly flex to minimize the total residuals. This assumption can be further relaxed. For example, our argument still holds by requiring only a sub-linear number $f(k)$ of nodes to flex rather than all k nodes, as long as $f(k)$ is a monotonically increasing function, e.g., $f(k) = \log k$ or $\sqrt{k}$. Further, we note that for irregular node placements, location changes under least squares optimization are generally not synchronized in their directions, as we assumed in the analysis (Figure 3).

4.4 Robust Statistics

The SISR estimator can be viewed under the lens of robust statistics, a theoretical framework concerned with the outlier rejection problem in statistical analysis. Outliers can bias the final estimation of an estimator and lead to incorrect conclusions and are usually hard to filter out, especially when different outliers of different magnitudes are mixed together in the data. Estimators which suffer from the presence of outliers are mean and least-squares; they are not robust to outliers in the sense that the estimation can be altered without bound by an extremely noisy outlier. In contrast, the median estimator is not as susceptible to such polluting data, and is considered a robust estimator.

To assess the robustness of an estimator, an associated *influence function* is useful in characterizing the importance of different data samples [7]. A smaller absolute value in the influence function means the data receives less weight in the estimation. The influence function is typically proportional to the derivative of the estimator. For example, Figures 4 (a) and (b) show the least-squares estimator and its influence function. The influence function grows linearly with the residual of a sample and, as a result, a single noisy data sample can have a huge effect on the estimator. On the other hand, the *Lorentzian* estimator, depicted in Figure 4 (c) and (d), has a *redescending influence function*. The function approaches 0 for large residuals, and thus data with large residuals have diminishing effects on the estimator. Such estimators are less sensitive to gross errors and are more robust.

The SISR estimator and its influence function are illustrated in Figures 4 (e) and (f). The influence function has a redescending shape similar to that of the Lorentzian, and hence we know SISR is also a robust estimator. The major difference between SISR and the Lorentzian is that SISR has a narrower center well, and flatter wings at the two ends. This creates better opportunity for SISR to ignore the bad data and stick with the good measurements. Our experimental results in Section 7 show that SISR indeed gives better localization estimates than the Lorentzian estimator.

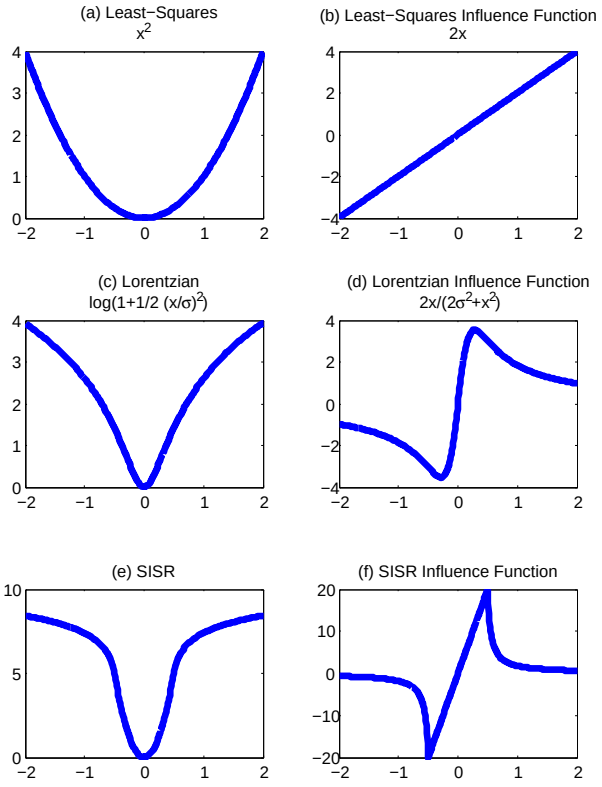


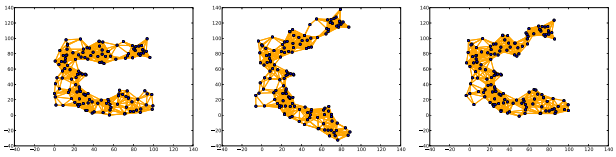
Figure 4: Robust estimators and their influence functions.

5 Evaluation Methodology

We compare SISR against an optimized variant of multidimensional scaling—MDS-MAP(C,R) [16]. A key insight in MDS-MAP is the replacement of the ranging measurement matrix with a "proximity matrix" P, where p_{ij} is the shortest path distance between nodes i and j. This essentially mitigates (the common) ranging error where the distance between two nodes is overestimated and, in multi-hop topologies, provides a proxy ranging estimate for node pairs that are not within communications range. MDS-MAP(C,R) further develops this insight by performing a weighted least squares refinement on a classic MDS-MAP solution, where constraint weights are inversely proportional to the number of hops in the shortest path. This strategy dampens the effect of long-distance estimates, which are more susceptible to errors, but makes the blanket assumption that long-distance estimates are always less trustworthy. In contrast, SISR can discern between good and bad ranging measurements and automatically use only those that are good.

In general, MDS-MAP is an effective method for multi-hop localization. However, for certain non-convex multi-hop topologies the shortest path distance can give rise to errors in localization, a recognized limitation of such approaches [18, 2]. For example, in the "C"-shaped topology in Figure 5a, the true distance between nodes at the ends of the "C" is much smaller than the shortest path distance would indicate. On such a topology, MDS-MAP tends to force apart the ends of the "C" (Figure 5b). The presence of ranging errors tends to exacerbate the problem. For this reason, we focus on these difficult topologies in our simulations.

Given a reasonably good initial approximation, SISR can arrive at a very accurate localization solution for nodes that have good ranging measurements. MDS-



(a) (b) (c)

Figure 5: (a) A non-convex "C"-shaped topology with 147 randomly-placed nodes within a 100m $\times$ 100m bounding area. Topologies such as these were used as ground truth in simulations described in Section 6. (b) The corresponding MDS-MAP(C,R) solution. Deformation in the ends of the "C" are clearly visible. (c) The corresponding SISR solution, which corrects for the deformation, even when ranging errors are present.

MAP(C,R) results are good candidates for initial approximations, and as such, we use them in our simulations.

The most natural way to evaluate the performance of these algorithms is through an *absolute localization error metric*, i.e., the discrepancy between a node's computed location and its ground truth location. However, the formulation of SISR we have presented thus far is an anchor-free, relative localization scheme; it depends only on pairwise distance measurements and does not return its results in the same coordinate system as the ground truth. The core MDS-MAP(C,R) algorithm is also such a scheme.

In order to report absolute localization error, we modified both algorithms slightly to employ the ground truth locations of four anchor nodes. As in [16], the anchors are randomly selected. For MDS-MAP(C,R), the solution returned by classic MDS-MAP is first transformed (via rotation, translation and reflection) into the ground truth coordinate system by fitting the anchors to their ground truth locations. This provides a better-conditioned initial solution to the subsequent refinement stage, where the anchor locations are further used as additional, fixed-constraint equations passed to the weighted least squares process. This forces the least squares operation to respect the ground truth coordinate system. The same technique is employed in providing constraint equations to SISR. Note that the way we use anchor information is different from the method in [16], where the final localization solution is fitted to the ground truth via anchors. Since we postulate the availability of anchor information, it makes sense to take advantage of it during the localization procedure.

In 2D localization, three anchors are sufficient, but their random selection can cause ambiguities during coordinate system fitting (e.g., if the three anchors are almost collinear). In [16], a fourth anchor is used to reduce the ambiguity, but there is no guarantee that this will always work. Thus, a question arises: what criteria describe a good set of anchors? First, they must be well-separated. Second, the angles between any pair must not be near zero. We note that, in its anchor-free, relative localization formulation, SISR can be used to bootstrap an absolute localization by providing hints as to which nodes satisfy these two criteria. First, SISR's snapped-in effect allows accurately localized nodes to be distinguished by their low residuals. Second, from the relative localization solution, we can distinguish which among the low-residual nodes are well-separated. However, for fairness in comparison with MDS-MAP(C,R), we do not employ this method in our evaluations below.

6 Simulation Results

In our first set of simulations, we want to determine how much SISR can improve upon MDS-MAP(C,R) in non-convex, "C"-shaped topologies, under various error conditions. We first outline the experimental procedure and follow with simulation details. For a given topology and set of error conditions, we compute the MDS-MAP(C,R) localization solution as described in Section 5. This is then used as an initial approximation for SISR localization. We then compare the accuracy of the resulting solutions from the two methods.

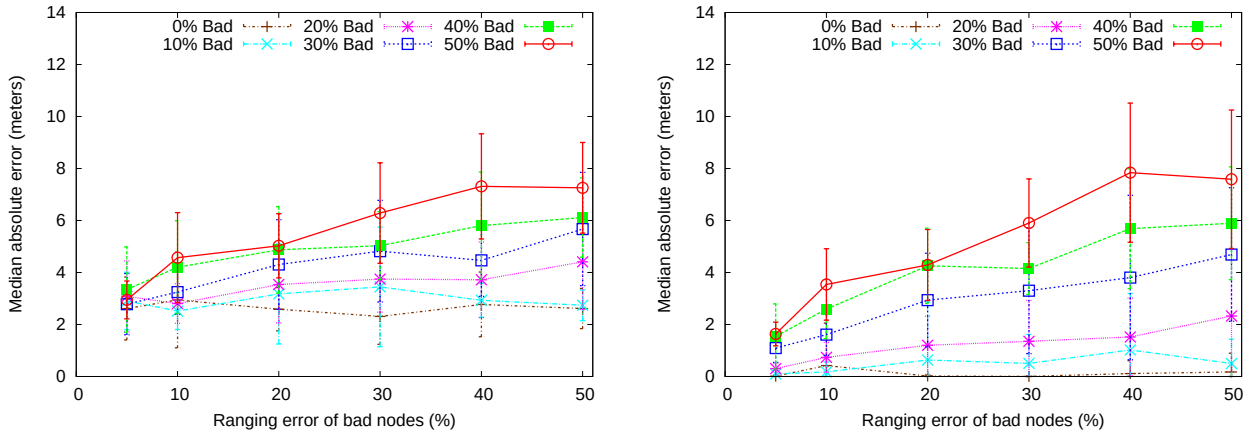
For each trial, we generate a topology where 147 nodes[1] are placed uniformly at random within a "C"-shaped region in a bounding area of 100m $\times$ 100m. An example of such a topology is shown in Figure 5a. Each node has a simulated radio range of 16.5m, which results in a multi-hop topology.

Each trial is also described by two experiment conditions governing error. The first experiment condition is the percentage of "bad nodes" within the population (0%, 10%, 20%, 30%, 40%, 50%). A node is assigned to either a "good" population, in which case it has no error in ranging, or a "bad" population, in which case it experiences ranging error $\varepsilon \sim N(\mu_{bad}, 0.2\mu_{bad})$. The second experiment condition is μ_{bad} (5%, 10%, 20%, 30%, 40%, 50%). Note that a "bad node" has ranging error only from itself to other nodes and not vice versa; this results in asymmetric range measurements, a more realistic scenario.

We have 36 combinations across the two experiment conditions, for a total of 36 simulations. Each simulation consisted of 20 trials on distinct topologies. The average median absolute localization errors are shown Figure 6. First, consider the results in Figure 6a, where MDS-MAP(C,R) is shown to be relatively insensitive to all values of ranging error, when the percentage of "bad nodes" is low ($<$30%), but never gives particularly accurate results. All MDS-MAP(C,R) solutions had at least 2m error, even when there are no "bad nodes" in the population.

In contrast to MDS-MAP(C,R), SISR is capable of providing much more accurate results—less than 2m error—when the percentage of "bad nodes" is less than 30%, as shown in Figure 6b. For these conditions, due to (1) the "snap-in" behavior of SISR, (2) the fact that "good nodes" have no measurement error, and (3) a good initial

[1] We chose 147 nodes to maintain the same node density as 196 nodes placed in a regular grid pattern in the same area.



(a) MDS-MAP(C,R) (b) SISR

Figure 6: A comparison of the accuracy of (a) MDS-MAP(C,R) and (b) SISR. Topologies were generated with 147 nodes placed uniformly at random in a "C"-shape within a 100m $\times$ 100m bounding area. For each experiment, a different percentage of nodes were simulated as "bad", i.e., experiencing ranging errors. We vary the magnitude of the ranging error and compare the average median absolute localization error returned by the different methods (20 trials per data point). Compared to MDS-MAP(C,R), SISR gives less error in the solution when the percentage of "bad nodes" is $< 30\%$, due to the "snapped-in" effect.

approximation afforded by MDS-MAP(C,R), the resulting localization solution is not significantly affected by the magnitude of the ranging error. Of course, in a real setting, we cannot expect "good nodes" to be absolutely correct in their range estimates; we will explore one such real setting in Section 7, where we describe our field experiments.

The simulation results above are consistent with our analysis of the SISR estimator's breakdown point in Section 4.3. For $\tau = 0.25$ and $\alpha = 80$, as used in our simulation, the breakdown point for a fully-connected network should be approximately 30%. However, our simulated "C"-topologies were not fully-connected but multihop. Thus, the calculated 30% breakdown point is an upper bound for our simulation scenarios, which agrees with the data presented in Figure 6b. Note that modifying τ can have a large effect on the SISR estimator's behavior. This leads us to explore ways of tuning τ in the following section.

6.1 Iterative Refinement with SISR

Tuning the SISR estimator by setting a very small τ can lead to more accurate localization results, but will also increase the likelihood of falling into an incorrect local minimum. On the other hand, a more permissive τ reduces this likelihood at the expense of localization accuracy. We can naturally exploit this trade-off by performing multiple rounds of SISR where τ is successively reduced in each round and the localization solution of one round is used as the initial approximation for the next.

In this simulation, 38 nodes are randomly placed within a C-shaped topology with a bounding area of 50m $\times$ 50m region (radio range is 16.5m), giving the same node density as in our previous simulations. Bad nodes are assigned ranging measurement error $\varepsilon \sim N(0.5, 0.1)$, with a different percentage of bad nodes across three experiments (10%, 30%, 50%). τ is initially set at 5m ($\alpha = 1/\tau^2$, so that the height of the "U"-shaped region does not change). SISR is then run for 10 iterations, with τ decreased by 25% with each iteration. As before, the first iteration is given the result of MDS-MAP(C,R) as its initial approximation. Note that with fewer nodes (as compared to the previous simulations), the optimization problem is actually more difficult in some sense, since there are fewer constraints and more degrees of freedom.

Figure 7 shows the result of this simulation. Each data point is the average median localization error in meters over 10 trials. Note that as we successively decrease τ, the accuracy of the localization improves. For 10% bad nodes, SISR already localizes with high accuracy, even when $\tau = 5$. The improvement is greatest for 30% bad nodes, where there is room for improvement by SISR and the percentage of bad nodes is within SISR's breakdown point. The improvement is not as marked for 50% bad nodes, as the high number of bad nodes places a fundamental limit on the accuracy of the localization.

In this simulation, we arbitrarily chose to scale down τ by a factor $\delta = 3/4$ per iteration for simplicity. In practice, one can tune δ by keeping track of the number of low-residual nodes from iteration to iteration (i.e., nodes that have been snapped-in). If, from one iteration to the next, a large number of these nodes suddenly exhibit high residuals then it is an indication that δ was too aggressive and that it should be scaled back.

Similarly, in this simulation, we arbitrarily terminated the τ refinement after 10 iterations. In theory, we could have further reduced τ to produce even better re-

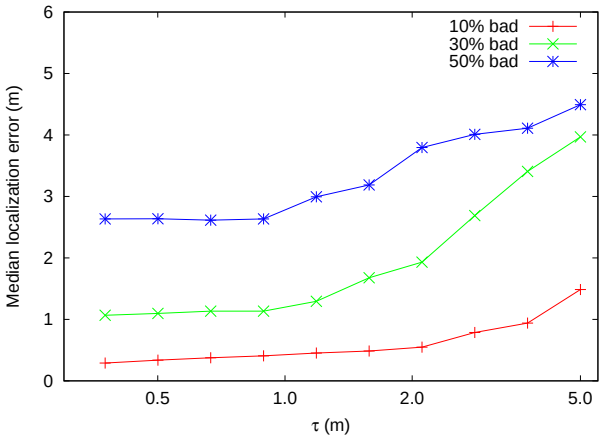


Figure 7: Iterative refinement of localization solutions by gradually decreasing τ gives improved accuracy. Each curve is a separate experimental condition, differing in the percentage of nodes in the population with ranging errors of 50% (10%, 30%, 50% bad nodes).

sults, since the "good nodes" experience no ranging error. However, in a real-world setting, the accuracy of range measurements on "good nodes" establishes a lower bound on τ (i.e., we cannot expect better performance than our best measurements). Thus, the iterative approach also provides a way of tuning τ to the optimal level—when the accuracy of the resulting localization solution no longer improves (as measured by the residuals of the nodes), we know we have reached an optimal value for τ.

7 Field Experiment and Results

In this section we describe a preliminary localization experiment using actual 802.11 radios in an outdoor environment. The goal of this experiment was to validate SISR localization using real systems and radios on real-world data and especially in the "bad node" scenario. We first describe our experimental setup and explain how we collected and processed the RSSI measurements. We then compare the localization results returned by MDS-MAP(C,R), the Lorentzian estimator, and SISR. In comparing the Lorentzian with SISR, we tuned the former such that it closely matched the shape of the SISR function at $\tau = 0.25$m. As we will see, SISR effectively mitigates the effect of the bad node. However, since the topology in this experiment is convex and nodes reside in a single broadcast domain, we do not expect to observe significant performance gains over MDS-MAP(C,R). Nevertheless, we do want to verify that our SISR implementation is correct in the sense that it should perform comparably to MDS-MAP(C,R). As future work, we will conduct field experiments with non-convex topologies to demonstrate SISR's performance gain, as predicted by our simulation results in Section 6.

7.1 Experimental Setup

To convert RSSI into range measurements, we have developed a path-loss model, and validated the model via field measurements. Details on our path-loss model can be found in Appendix A.

We chose a flat grass field near a football stadium as the experiment site. There were no major buildings or other obstacles nearby, and only faint ambient beaconing traffic from 802.11 base stations. The nodes we used were the One Laptop Per Child (OLPC) computers, which contain an x86 compatible AMD processor and run Linux. The radio in the OLPCs is a Marvell 8388 802.11 b/g internal USB module. We placed the radios into ad-hoc mode, and took steps to disable 802.11 beaconing and IBSS formation which could cause unexpected network splitting or otherwise interfere with experiments [1] [3]. Lastly, we modified the Linux kernel to provide Received Signal Strength Indicator (RSSI) values as metadata alongside any received UDP packets in order to facilitate link measurements.

We structured our experiments as a series of basic one-way signal strength measurement operations between pairs of nodes, performed as follows. One node of the pair would transmit for a relatively long period of time, broadcasting a stream of 64-byte UDP probe packets at the 2Mbps modulation and on 802.11 channel 4. The receiver node would listen for these probe packets, logging each packet's transmitter address and RSSI. Note that it is convenient to perform these basic measurement operations in parallel, by having multiple receivers capture a single transmitter's probes.

Due to environmental effects, RSSI measurements may be subject to fading fluctuations. At the same time, the RSSI reporting mechanism inside our radios' firmware is an undocumented black box, and may report values which deviate from the actual received power. Therefore, in order to obtain a more reliable RSS estimate, we use a simple windowing heuristic to filter the RSSI data. For node i receiving from node j, we take n total RSSI samples, divide these into w consecutive windows, take the maximum RSSI value in the window and then take the median of these values to be the "representative" RSSI of transmissions from node j. Of course, the more samples taken, the more accurate the representative RSSI. From our field data, we have found that the RSSI heuristic converges in as few as 1000 packets ($\sim$ 300ms transmission time), suggesting that the wireless ranging approach we have adopted can be done quickly.

7.2 Pairwise Measurements in the Field

We arranged 20 testbed nodes within a 25m $\times$ 25m square region, which we found to be the approximate area of a broadcast domain for nodes that are placed on the ground (the diagonal, at $\sim$ 36m, is the limit of transmission range). Multi-path propagation—in particular, ground reflection—gives rise to null regions, where the RSSI-to-distance function is aliased or measurement is unavailable. To guard against this during experimentation, we use physical receiver diversity by deploying pairs of nodes within 1m of each other. As shown in Fig-

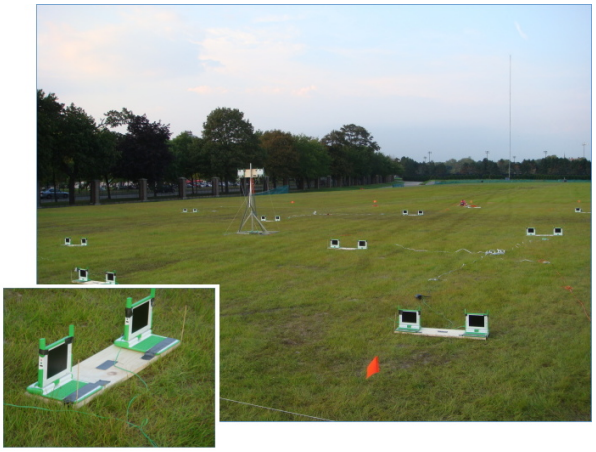


Figure 8: Photo of outdoor OLPC testbed (20 nodes comprising ten two-node planks) in a 25m $\times$ 25m open field.

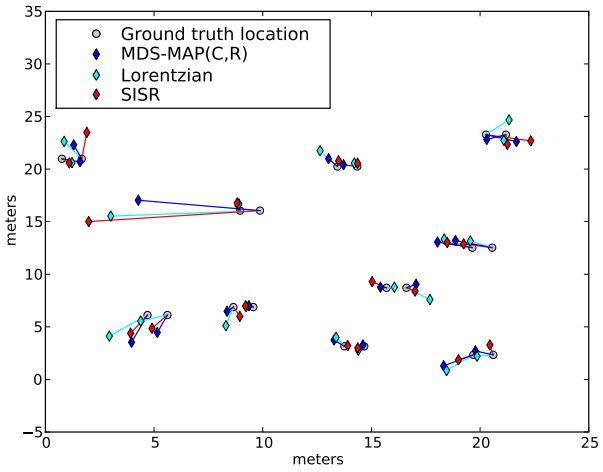


Figure 9: SISR (red), Lorentzian (cyan) and MDS-MAP(C,R) (blue) localization solutions, relative to ground truth (grey). The median absolute localization errors are: SISR = 0.90m (τ = 0.25m), Lorentzian = 1.27m ($\sigma = 0.025$), MDS-MAP(C,R) = 0.89m. As noted in Section 7, the comparable performance of SISR and MDS-MAP(C,R) is expected due to the fact that almost all nodes can hear from each other directly.

ure 8, pairs of nodes were fastened onto 1m-long wooden planks. These pairs were then arbitrarily placed within the square region, and their pairwise distances physically measured.

We measured the signal strength of all 380 transmitter-receiver combinations (190 pairs measured in both directions). Using this data, each node uses the path loss model as shown in Figure 12 to estimate its distance to all the nodes from which it has received packets.

7.3 Localization Experiment

Among the nodes in our testbed, a single node X proved to have an interesting failure mode. Specifically, this node functioned normally when transmitting, but would consistently report abnormally low RSSI values for any received packets (e.g., $\sim$30dB lower than other nodes for a transmitter only 1m away). Likely due to faulty hardware, this behavior is ideal as a test case for fault-tolerant localization methods such as SISR or MDS-MAP.

Figure 9 shows the localization solution given by SISR (red diamonds) and MDS-MAP(C,R) (black diamonds). The ground truth locations are indicated by grey circles and the lines connecting the localized positions indicate node identity. Figure 9 visually verifies that most nodes are accurately localized by SISR, with the obvious exception of one node, which has been localized to an impossibly distant position. Naturally, this is node X, whose ranging measurements were faulty across the board. This is the hallmark behavior of the SISR method– relatively less effort is made to minimize the error due to bad measurements at the expense of the good ones.

SISR gives a median absolute localization error of 0.90m and MDS-MAP(C,R) gives 0.89m; in contrast, the Lorentzian estimator gives 1.27m. The CDF of absolute localization error across nodes is shown in Figure 10. The SISR curve (red) tracks very closely to that of MDS-MAP(C,R) (blue). The two methods performed comparably, due to the relatively uniform, convex topology and lack of potentially distorting multihop links. As a comparison, the Lorentzian tracks under the MDS-MAP(C,R) curve, indicating that it performs worst overall. This shows that not all robust estimators solve the localization problem equally well and that SISR is particularly suited for the task.

8 Applying SISR to Other Localization Approaches

The SISR approach is applicable to other localization approaches beyond the range-based one described above. This is because the principle of placing emphasis on computed locations which match well with measurement data is a universal notion applicable to any localization method. To provide a validation of this point, we consider briefly the use of SISR in TDOA (time-difference-of-arrival) based localization. In TDOA, a speaker is localized by measuring the difference in propagation time of acoustic signals. The location can be solved with a set of non-linear equations describing the distance relationships between the speaker and multiple microphone sensors. The measurement of the difference in arrival time is usually achieved by applying cross-correlation on recorded signals. However, this often suffers from environmental noises that corrupt the waveforms of the recorded signals. In our simple test scenario with five sensor nodes, one of them gives erroneous measurements because of its low signal-to-noise ratio. Simulations show that the SISR estimator can still successfully localize the speaker within 2 meters when the errors vary between 0 to 50%. Figure 11 shows a simulation result of a scenario with 40% error.

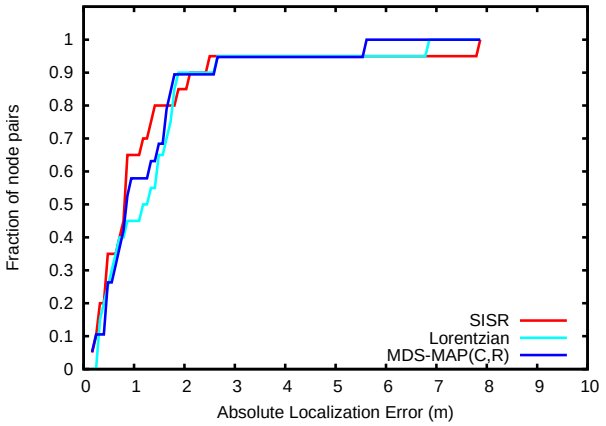


Figure 10: CDFs of the absolute localization error across all nodes as given by SISR (red), the Lorentzian estimator (cyan) and MDS-MAP(C,R) (blue).

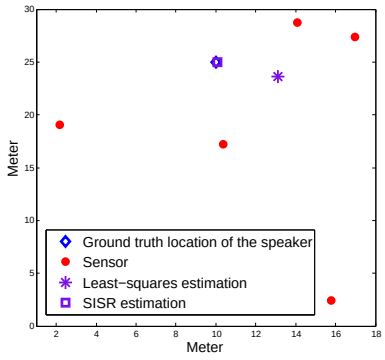


Figure 11: TDOA localization with SISR. Circles denote the five sensors. The square is the location estimated by SISR while the star is the location estimated by least-squares. The ground truth of the speaker is denoted as a diamond, which is almost overlapped with the square.

9 Conclusions

In this paper, we present SISR, a localization method for wireless nodes that employ RF ranging. Via simulation and field experiments, we have validated that our SISR localization method can tolerate errors in ranging measurements by de-emphasizing bad data while accentuating more accurate readings. In particular, SISR can accurately localize difficult non-convex topologies, even in the presence of non-Gaussian errors in ranging. Moreover, SISR is tunable, in that it can be made effective for different environments with different error conditions. We note that SISR is a general method and can be applied towards any ranging technique that experiences gross error, e.g., acoustic time-difference-of-arrival (TDOA). In the future, we plan to extend SISR to distributed and cluster-aware localization, where local, reliable clusters would first be localized, and later stitched together to form a global topology by using SISR on inter-cluster ranging measurements. SISR also suggests a natural way of pruning bad nodes from the network, which we plan to implement: by comparing the unshaped residual (conventional least squares) residual of nodes, outliers can be rejected. Such node discrimination may be critical to the performance of higher-level network applications.

10 Acknowledgments

This research was supported in part by the Air Force Research Laboratory Grants FA8750-08-1-0220 and FA8750-08-1-0191.

11 References

[1] D. Aguayo, J. Bicket, S. Biswas, G. Judd, and R. Morris. Link-level measurements from an 802.11b mesh network. In *SIGCOMM 2004*, Aug. 2004.

[2] J. Bachrach and C. Taylor. Localization in sensor networks. *Handbook of Sensor Networks: Algorithms and Architectures, 1st ed.*, 1, 2005.

[3] G. Bianchi, A. Di Stefano, C. Giaconia, L. Scalia, G. Terrazzino, and I. Tinnirello. Experimental assessment of the backoff behavior of commercial ieee 802.11b network cards. In *INFOCOM 2007*, 2007.

[4] P. Biswas, T. C. Liang, K. C. Toh, Y. Ye, and T. C. Wang. Semidefinite programming approaches for sensor network localization with noisy distance measurements. *Automation Science and Engineering, IEEE Transactions on*, 3(4):360–371, 2006.

[5] M. J. Black and P. Anandan. The robust estimation of multiple motions: parametric and piecewise-smooth flow fields. *Comput. Vis. Image Underst.*, 63(1):75–104, 1996.

[6] D. K. Goldenberg, P. Bihler, R. Y. Yang, M. Cao, J. Fang, S. A. Morse, and B. D. O. Anderson. Localization in sparse networks using sweeps. In *MobiCom 2006*, 2006.

[7] F. Hampel, E. Ronchetti, P. Rousseeuw, and W. Stahel. Robust Statistics: The Approach Based on Influence Functions. *New York*, 1986.

[8] T. He, C. Huang, B. Blum, J. Stankovic, and T. Abdelzaher. Range-free localization schemes for large scale sensor networks. In *MobiCom 2003*, 2003.

[9] P. Huber. *Robust statistics*. Wiley New York, 1981.

[10] J. Liu, Y. Zhang, and F. Zhao. Robust distributed node localization with error management. In *MobiHoc 2006*, 2006.

[11] D. Moore, J. Leonard, D. Rus, and S. Teller. In *SenSys 2004*, New York, NY, USA, 2004.

[12] D. Nicolescu and B. Nath. Ad-Hoc Positioning Systems (APS). In *Proceedings of IEEE GLOBECOM*, volume 1, pages 25–29, 2001.

[13] C. Savarese, J. M. Rabaey, and K. Langendoen. Robust positioning algorithms for distributed ad-hoc wireless sensor networks. In *ATEC '02: Proceed-*

ings of the General Track of the annual conference on USENIX Annual Technical Conference, 2002.

[14] A. Savvides, C.-C. Han, and M. B. Strivastava. Dynamic fine-grained localization in ad-hoc networks of sensors. In *MobiCom 2001*, 2001.

[15] A. Savvides, H. Park, and M. B. Srivastava. The n-hop multilateration primitive for node localization problems. *Mob. Netw. Appl.*, 8(4):443–451, 2003.

[16] Y. Shang, W. Rumi, Y. Zhang, and M. Fromherz. Localization from connectivity in sensor networks. *Parallel and Distributed Systems, IEEE Transactions on*, 15(11):961–974, 2004.

[17] Y. Shang, W. Ruml, Y. Zhang, and M. P. J. Fromherz. Localization from mere connectivity. In *MobiHoc 2003*, 2003.

[18] C. Whitehouse. The design of calamari: an ad-hoc localization system for sensor networks. Master's thesis, UC Berkeley, 2002.

[19] K. Whitehouse, C. Karlof, A. Woo, F. Jiang, and D. Culler. The effects of ranging noise on multihop localization: an empirical study. 2005.

A Path Loss Modeling Using 1D Measurements

In order to build a path loss model, we measured the signal strenths at a fine grained set of distances. In particular, we placed a set of transmitting nodes in a fixed, origin location, and moved a set of receiving nodes across 180 points betwen 1 and 64 meters away from the origin. There were transmitting nodes at 3 different heights, namely, ground height, 1m and 2m, while we only placed the receiving nodes on the ground. For diversity purposes, we used 2 transmitters and 2 receivers for each measurement.

Propagation over flat ground is usually modeled as a combination of two rays–one direct ray, and a secondary ray reflected off the ground. Each ray in turn experiences free space attenuation, where the power density falls off proportionally to the inverse of the square of the distance. We express this using a slightly modified standard model as a function of the distance between two nodes d, and their respective heights, h_1 and h_2 above ground. We omit the derivation of the model due to space constraints, but we include the resulting model and show it in Figure 12 plotted against the data obtained from ground-height nodes.

The model is shown in the following equation, with P_T, ρ, $\alpha(\cdot)$ and $\beta(\cdot)$ being model parameters, and $\lambda = 12.3$cm the wavelength of the center frequency of the communication channel used:

$$RSSI(d,h_1,h_2) = P_T \cdot \cdot \left(\frac{\alpha(d,h_1,h_2)}{d^2+(h_2-h_1)^2} + \frac{\beta(d,h_1,h_2)}{d^2+(h_2+h_1)^2} - \right.$$

$$\left. - \frac{2\sqrt{\alpha(d,h_1,h_2)\beta(d,h_1,h_2)}}{(1/\rho)\sqrt{d^2+(h_2-h_1)^2}\sqrt{d^2+(h_2+h_1)^2}} \cdot \right.$$

$$\left. \cdot \frac{1}{\cos(\frac{2\pi}{\lambda}(\sqrt{d^2+(h_2+h_1)^2} - \sqrt{d^2+(h_2-h_1)^2}))} \right)$$

Our modification consists of adding factors $\alpha(\cdot)$ and $\beta(\cdot)$ to model the effects of antenna radiation patterns. When $\alpha(\cdot) \equiv 1$ and $\beta(\cdot) \equiv 1$ we obtain the standard 2-ray model, while in our model we used the following functions of d and h:

$$\alpha(d,h) = \left(d/\sqrt{d^2+h^2} \right)^a$$

$$\beta(d,h) = \left(d/\sqrt{d^2+h^2} \right)^b$$

where the parameters a and b are model parameters.

We trained the above model by finding parameters P_T, ρ, a and b using a least squares minimization; the values of the parameters were $P_T = -29.7dBm$, $\rho = 0.99$, $a = 4.0$, and $b = 54.8$. The data points and the modeled signal strength curves are shown in Figure 12. Note that the null regions typically seen in 2-ray propagation models were quite shallow in the measured data, indicating a lack of fully destructive interference, possibly due to a weak reflected path. Thus, in the model we attenuated the reflected path using the $\beta(\cdot)$ factor with a large b parameter.

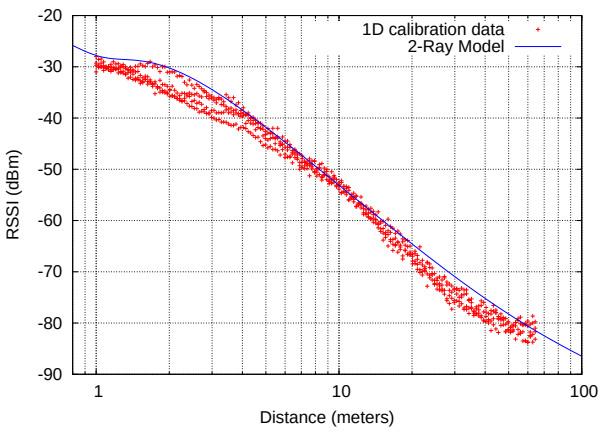


Figure 12: Path loss model derived from field calibration data.

There are several reasons that the above simple model might not work. The two-ray formulation is not appropriate for terrains which are not flat, or contain obstacles. Furthermore, the reflective properties of ground in different environments may differ, necessitating a retraining of the model. Finally, the model has dependencies on specific antenna radiation patterns, and may not work for antennas whose patterns do not match. We discuss these points in turn.

There are plenty of interesting scenarios with flat, or nearly flat terrain where the model can still work. The effect of increasing terrain irregularities would be an increasing error, and so the performance of methods based on the model would gracefuly degrade. The model will

also have errors in environments with obstacles, commensurate to the amount of clutter, because obstacles introduce additional reflection paths which no longer match the model assumptions. Similar to the non-flat terrain case, the model will likely encounter increasing error. In both cases, the model may be used up to some error tolerance.

Different ground materials have different reflection coefficients. However, at the small grazing angles that our model is targeted to, most terrain types behave similarly. For example, Table 2 lists the reflection coefficients for some common ground types, calculated for a case where nodes are 10 meters apart and positioned at 20cm above ground.

Table 2: Reflection coefficients of vertically polarized waves over some common ground types.

Ground type	Reflection coefficient
Poor (dry) ground	0.92
Typical ground	0.87
Good (wet) ground	0.84
Sea water	0.72
Fresh water	0.73
Brick	0.92
Limestone	0.90

The model assumes that antennas are vertically polarized, and have vertical dipole-like radiation patterns; that is, omnidirectional in the horizontal plane, and with the main lobe oriented horizontally. However, the vertical dipole is a widely used antenna choice, and in those cases the model would likely be appropriate—especially if the host device construction only has a minor effect on the radiation pattern.