Cluster-Based Minimum Mean Square Estimation for Secure and Resilient Localization in Wireless Sensor Networks*

Cliff Wang
U.S. Army Research Office
Email: cliff.wang@us.army.mil

An Liu, Peng Ning
Dept. of Computer Science, NC State University
Email: aliu3@ncsu.edu, pning@ncsu.edu

Abstract

To support a wide variety of applications ranging from military surveillance to health care clinic monitoring, a wireless sensor network must obtain accurate location for each sensor. A number of localization schemes have been developed to allow each sensor node to acquire its location. However, most of these techniques assume benign environments, and thus cannot survive malicious attacks in hostile environments where external and/or compromised nodes may launch attacks. This paper proposes a new computationally efficient and resilient localization scheme based on the clustering of benign location reference anchors. Moreover, this paper reports both simulation and field experiments using a test-bed of MICAz motes performed to compare the proposed approach with several recent secure localization schemes. The experimental results demonstrate that the proposed scheme has the fastest execution time among all resilient localization schemes that can be used for the current generation of sensor platforms (e.g., MICA series of motes).

Keywords: Localization, wireless sensor networks, robustness and attack resiliency

1 Introduction

Wireless sensor networks (WSN) have found a great variety of applications ranging from health monitoring in the civilian world to military surveillance and reconnaissance. Location information is the key to many networking protocols such as geographical routing protocols or geographic data-centric storage, to applications such as target tracking and environmental monitoring. To support these protocols and applications, nodes in WSN need to acquire and maintain accurate location information. Although Global Positioning System (GPS) is a popular outdoor localization system for mobile devices, it is highly undesirable to have a GPS receiver on every sensor node due to the cost reasons. Moreover, in some situations such as indoor sensor network applications, GPS may not be used. This creates a demand for efficient and cost-effective location discovery algorithms in WSN.

There has been intensive investigation of localization techniques that do not entirely depend on GPS in recent years. Such techniques use some special nodes called *anchors*, which know their own locations, to help the other nodes discover locations. Existing localization schemes can be classified into *range-based* (e.g., [4,16,19]) or *range-free* techniques (e.g., [1, 5, 14, 17]). Range-based localization uses Received Signal Strength Indicator (RSSI), Time of Arrival (TOA), or Time Difference of Arrival (TDOA) to estimate the distance between the node that needs to discover its location and each reachable anchor, and estimates the node's location based on these distances and the anchors' locations. To reduce the demand on sophisticated hardware, computing power, and energy, range-free localization does not rely on physical distance measurement, but uses other means (e.g., centroid of all reachable anchors [1], overlap of triangles formed by reachable anchors [5], or hop counts from anchors [17]) for location estimation.

WSN may be deployed in hostile environments, and localization may become the target of attacks due to its importance. The threats to WSN localization in a hostile environment mandates the development of *secure and resilient* localization algorithms. Further, due to the resource constraints of a typical sensor node, the secure and resilient algorithm has to be *efficient* in terms of computation and memory requirement.

Among recent developments of secure and resilient localization schemes for WSN [2, 8–12], attack-Resistant Minimum Mean Square Estimation (ARMMSE) was first published in [12], and later an improved algorithm called Enhanced ARMMSE (EARMMSE) [11]. It is based on the observation that a location reference introduced by a malicious attack is usually "different" from benign ones, since it is aimed at misleading location estimation. Thus, AR-

*This work is supported by United States Army Research Office grant W911NF-04-D-003. Ning's work is supported by the National Science Foundation under grants CAREER-0447761 and CNS-0430223.

MMSE uses the "inconsistency" among the location references provided by anchors to identify the malicious injection, and discard them before making final location estimation. The voting-based location estimation [11, 12] has each location reference "vote" on the locations at which the node of concern may reside, on a grid of cells. It then selects the cell(s) with the highest vote and use the "center" of the cell(s) as the estimated location. Li et al. [10] studied and developed an attack resilient location estimator based on Least Median of Squares (LMS). The idea is to draw random subsets of data from the original data pool for individual subset estimation and then combine these estimates based on estimation quality. There are several other secure and resilient localization techniques, including SeRLoc [9], SPINE [2], and ROPE [8]. However, SeRLoc requires directional antenna on sensor nodes, SPINE requires nano-second scale time synchronization among sensor nodes, and ROPE, which is an integration of SeRLoc and SPINE, requires both directional antenna and nano-second scale time synchronization. These requirements cannot be met on the current generation of sensor platforms such as MICA series of motes.

Given the severe resource constraints (e.g., limited computing power, depletable battery power) of the current generation of sensor platforms, a usable resilient localization scheme must be lightweight and computationally efficient. In this paper, we develop a novel computationally efficient and resilient localization scheme, which is more efficient than all the previous approaches with bounded location estimation error similar to the previous approaches. The computation efficiency and attack resiliency make the proposed approach suitable for resource constrained sensor nodes. This is our first contribution in this paper.

With the exception of [11, 12], all the other approaches have not been validated on real sensor platforms. To evaluate and compare the new algorithm with existing schemes, we implemented all the resilient localization schemes that can be used for the current generation of sensor platforms in TinyOS, and perform extensive evaluation and comparison of all these approaches through both simulation and experiments in a test-bed of MICAz motes. In particular, our implementation offers a readily available code base for integration into location-aware WSN applications. This is our second contribution in this paper. Our experimental results demonstrate that the new approach proposed in this paper is fastest among all the ones evaluated. For complete simulation and experimental results, please refer to full version of this paper [20].

The rest of this paper is organized as follows. The next section describes our assumptions and threat model. Section 3 discusses the cluster-based minimum mean square estimation (CMMSE) algorithm for secure and resilient localization in wireless sensor networks. Section 4 presents the implementation and the experiments of the proposed schemes as well as several other resilient localization schemes through both simulation and field experiments. Section 5 concludes this paper and points out some future research directions.

2 Assumptions and Threat Model

We first clarify our assumptions and threat model to facilitate the discussion. We assume that a WSN consists of a large number of regular sensor nodes (e.g., MICA motes) that need to estimate their locations and a small fraction of special anchors that are location aware (through, e.g., GPS receivers or manual configuration). We assume that the anchors are roughly uniformly distributed in the network, and each regular sensor node can obtain localization information from a sufficient number of anchors. For simplicity, we assume that a WSN operates on a 2-dimension plane, though the algorithms investigated in this paper can all be used for 3-dimension space with slight modification.

We focus on range-based localization in this paper. A regular node gets two pieces of information from each anchor that it communicates with for localization purpose: the *location* of the anchor and the *distance* between them. While the location of an anchor is usually provided by the anchor directly in a localization packet, there are multiple ways to obtain the distance between them, for example, using RSSI, ToA, or TDoA. We assume a sensor network may use any method to obtain these two pieces of information. Following [11, 12], we refer the localization information obtained from an anchor as a *location reference*, represented as a triple (x, y, d), where (x, y) is the coordinate of the anchor, and d is the estimated distance to the anchor node. We assume there may be errors in the estimated distances. However, when there is no malicious attacks, all distance measurement errors are bounded by ϵ_{max}, i.e., $-\epsilon_{max} < \epsilon < \epsilon_{max}$, where ϵ is any distance measurement error obtained in attack-free environments. Based on the location references received from multiple anchors, a regular sensor node runs a localization algorithm to estimate its own location.

We assume that all localization related packets are authenticated (e.g., using TinySec [7])and that each anchor can be uniquely identified. This can be achieved using key management schemes to provide unique pairwise keys for different pairs of nodes (e.g., TinyKeyMan [13], random pairwise keys scheme [3]). Moreover, each regular node uses at most one location reference from each anchor.

We consider both external and insider attacks. When launching *external attacks*, the adversary does not control any valid node in the network. Though message authentication is effective in preventing the adversary from forging localization related packets, it cannot stop all external attacks. For example, the adversary may replay previously

intercepted localization related packets captured at different locations. Moreover, the adversary may launch wormhole attacks by creating low latency and high bandwidth communication channels between different locations in the network. The adversary may compromise anchors and launch *insider attacks*, such as reporting incorrect locations, or manipulating the transmission of localization related packets (e.g., by using overly high or low transmission power if RSSI is used for distance measurement). The adversary may jam the communication channel to launch Denial of Service (DoS) attacks. However, we assume that the adversary cannot constantly jam the communication channel without being detected and removed. Under both external and insider attacks, the adversary may convince regular sensor nodes to accept malicious location references. However, a regular node will accept at most one location reference from each compromised anchor or benign anchor whose localization packets are manipulated by the adversary, regardless how the adversary launches the attacks. Finally, we assume there are more benign anchors than colluding malicious anchors.

The primary objective of any secure and resilient localization scheme is to ensure that adversaries cannot introduce arbitrary localization errors under the above assumptions. Moreover, we would like to establish a good resiliency with the least computation overhead. In other words, we would like to develop light weight location estimation algorithms that can provide reasonably good location estimate with the presence of injected errors from malicious anchors.

3 Cluster-Based Minimum Mean Square Estimation (CMMSE)

In this section, we present a new secure and resilient localization scheme, called *Cluster-Based Minimum Mean Square Estimation (CMMSE)*. This approach achieves higher efficiency than existing approaches while providing comparable resilience against malicious location references. Similar to a previous work, ARMMSE [12], CMMSE is based on the Minimum Mean Square Estimation (MMSE) method proposed in [19].

3.1 CMMSE: Achieving High Efficiency

A malicious anchor can provide an arbitrary location reference by either changing its declared location (x, y) or manipulate the distance measurement (e.g., by changing the transmission power if RSSI is used). When both benign and malicious location references co-exist and the malicious nodes inject arbitrary location reference errors, the minimum MSE obtained with an MMSE method will exceed the normal MSE bound, giving us an opportunity to discover and discard malicious location references.

Obviously the basic MMSE method cannot deal with malicious location references, since it can't distinguish and discard faulty location references from malicious nodes. A resilient algorithm needs to filter out malicious location references and uses the good location references to perform localization.

Based on the assumption of majority benign anchor nodes, the ARMMSE approaches developed in [11, 12] run multiple rounds of basic MMSE operations to search for the largest consistent benign location reference set. The ARMMSE approaches start from the whole set of location references, and runs MMSE based consistency check to filter out malicious ones iteratively. This is inefficient since MMSE calculation involves many matrix operations. The larger the set of location references, the more costly the MMSE calculation.

To achieve a higher computational efficiency, our scheme takes an opposite approach by growing the largest consistent set from two randomly chosen location references (as the seed). If the two seeds selected are benign, we can grow the largest consistent location reference set within one round by go through each remaining location reference, using the MMSE threshold. Thus efficiency of this scheme depends on how quickly two benign location references can be selected. When the percentage of benign ones is high, two benign seeds can be chosen quickly. Another important feature of this algorithm is that any MMSE calculation is only performed on three location references. As a result, the calculation can be performed very quickly. Furthermore, we develop a variation of the cluster-based algorithm, which can provide a location estimation once more than half of benign location references are found. This provides further speedup of the algorithm. Section 3.2 provides a detailed analysis on the performance of this new approach.

3.1.1 Algorithm

Our CMMSE algorithm is based on the examination of location reference triplets. Given k location references received, there are a total of $\binom{k}{3}$ possible triplets (of location references) that can be formed. If there are no malicious location references, any one of the $\binom{k}{3}$ triplets may provide a good location estimate. However, due to malicious attacks, a triplet may have 0 to 3 malicious location references. It is obvious to see that with a total of *k* location references and *m* malicious ones, $\binom{k-m}{3}$ triplets are free of malicious location references, and $\binom{m}{3}$ triplets contain malicious location references only. The rest of the $\binom{k}{3} - \binom{m}{3} - \binom{k-m}{3}$ triplets may contain one or two malicious location references each.

Each triplet of location references provides a location estimate $(\hat{x}, \hat{y})$ and an associated minimum MSE. In the case of a benign triplet, the minimum MSE is bounded by measurement error ϵ^2_{max}. (Note that when an approximated MMSE method such as [19] is used, a set of three approxi-

mately co-linear location references will lead to large minimum MSE. We will address this problem with an expansion phase, as discussed later.) If all three location references in a triplet are malicious and colluding with each other, the minimum MSE may also be bounded because of the consistency of the three location references. However, when a triplet has both benign and malicious location references, or non-colluding malicious location references, the minimum MSE is no longer bounded, and can be arbitrarily large due to the injected error.

Based on the above intuition, *Cluster-Based MMSE (CMMSE)* uses a simple MMSE threshold test to identify consistent triplets, and to form a consistent reference set for the final location estimation.

The algorithm runs in rounds. In each round, a *triplet formation and examination* phase is used to identify the consistent set of location references. The basic MMSE method [19] is used to perform location estimation. As discussed earlier, the minimum MSE of a benign triplet may be greater than the MSE bound due to the algorithm's approximation error when the three reference anchors are close to be collinear. To address this issue, a second *expansion* phase may be used to examine and include those benign location references removed incorrectly in phase I.

For the sake of presentation, we denote the set of all input location references as S, and the set of location references to use for the final location estimation as C.

Phase I: Partitioning S via Triplet Formation and Examination.

1. Set two sets C and L both as empty sets.

2. We randomly select two location references r_1 and r_2 as the seeds, and perform a *proximity check* as follows: Assuming that d_i and d_j are the measured distances from the node to be localized to the two anchors and that the maximum radio signal range is d_{max}, if $d_i + d_j < 2*(d_{max}+\epsilon_{max})$, r_1 and r_2 are accepted as seeds and put into C. Otherwise, reject these two seeds and repeat this step.

   The rationale for this check is that two benign anchors from which a regular node receives localization references cannot be more than $2d_{max}$ away from each other.

3. Using the two seeds r_1 and r_2, we examine each of the remaining location references in S one by one. Specifically, for each remaining location reference r in S, it forms a triplet along with the two seeds r_1 and r_2. Following the basic MMSE method [19], we calculate the location estimation and its corresponding minimum MSE. If the minimum MSE is smaller than ϵ^2_{max}, the three anchors are consistent, and we put r into the consistent set C. Otherwise, it is not consistent with the two seeds and is placed in the set L.

After completing the above steps in Phase I, we split the original set S of location references into the consistent set C containing the two seeds and all nodes consistent with them, and the leftover set $L = S$ inconsistent with the seeds.

Phase II: Expansion of Consistent Set. During phase I, a location reference is placed into the leftover set L when the minimum MSE is greater than the bound ϵ^2_{max}. As mentioned earlier, the basic MMSE method may generate large calculation error when the three corresponding anchors are approximately collinear. In the expansion phase, each location reference in the leftover set L is checked against the whole reference set C to see if these $|C| + 1$ location references will generate a minimum MSE lower than ϵ^2_{max}. If yes, it is added into C. Any benign location reference mistakenly rejected in phase I due to the approximate error can be recovered.

After the execution of these two phases in one round, there are two possible outcomes:

1. $|C| > \frac{k}{2}$. In this case, there are C has more than $\frac{k}{2}$ consistent location references. Under the assumption that there are more benign location references than the malicious ones, we have obtained the benign set and can use it to perform the final location estimate.

2. $|C| \leq \frac{k}{2}$. In this case, we fail to grow a consistent set larger than $\frac{k}{2}$. We then remove the two seeds r_1 and r_2 from S, since they cannot be used to form a consistent majority from the input location references. If $|S| > \frac{k}{2}$, we start a new round by repeating Phase I and Phase II. Otherwise, there is no chance to form a consistent majority based on the input, and thus the algorithm terminates.

Quick CMMSE (QCMMSE): A Variation. To reduce execution time and reduce computation load, we can stop examining the remaining location references once we have obtained more than $\frac{k}{2}$ consistent location references. This leads to the QCMMSE variation of the proposed scheme. Specifically, we keep track of the size of C in both Phase I and Phase II. We terminate the algorithm whenever C has $max\{3, \lfloor \frac{k}{2} + 1 \rfloor\}$ consistent location references, and then use C for the final location estimation. This variation further speeds up CMMSE, with a trade-off of a slightly increased location estimation error.

3.2 Complexity

Each round of CMMSE is of complexity $O(k)$, given k location references. We only need to run one round if the first two seeds selected are benign. Otherwise, multiple rounds are needed until two benign seeds are selected. Ob-

viously, the number of rounds required depends on the number of malicious location references. A malicious location reference is in general not consistent with a benign one, and will not be selected as a seed along with a benign one. Thus, for *m* malicious location references, at most $\lfloor \frac{m}{2}+1 \rfloor$ rounds are needed in the worst case when each round selects two colluding malicious location reference as the seeds. This is possible probabilistically, but highly unlikely.

We can derive the probability of correctly selecting two benign seeds at round *i* as follows:

$$p(1) = \frac{(k-m) \times (k-m-1)}{k \times (k-1)},$$

and

$$p(i) = \frac{(k-m-(i-1)) \times (k-m-(i-1)-1)}{(k-2\times(i-1)) \times (k-2\times(i-1)-1)} \times \prod_{j=1}^{i-1} \left(1 - \frac{(k-m-(j-1)) \times (k-m-(j-1)-1)}{(k-2\times(j-1)) \times (k-2\times(j-1)-1)}\right),$$

when $i = 2, 3, ..., \lfloor \frac{m}{2}+1 \rfloor$. In this equation, the first part represents the probability of picking two benign location references in round i, and the second part represents the probability of not being able to pick two benign seeds in the previous rounds. The average number of rounds required to find the complete set can be found as $\sum_{i=1}^{\lfloor \frac{m}{2}+1 \rfloor} p(i) \times i$.

The average round required depends on the number of malicious location references. For example, when $k = 19$, $m = 3$, the average number of rounds is 1.16. When m increases to 9, the average number of rounds is 1.91. With no or a few malicious location references, CMMSE can finish fairly quickly.

3.3 Security Analysis

This section provides the security analysis of our localization scheme. We show that our scheme can tolerate arbitrary false messages introduced by compromised anchors as long as the benign anchors constitute the majority of all reachable anchors.

Based on our assumption that the location reference messages are cryptographically protected, an adversary cannot launch attacks simply by forging or modifying location reference messages without the knowledge of correct cryptographic keys. The adversary only has two ways to launch attacks: replay location reference messages captured in other places, and send malicious location reference messages through compromised anchors. The first case is indeed equivalent to a compromised anchor claiming to be in the same location as the original transmission location of the captured location reference message. Thus, we will focus our attention on attacks launched from compromised anchors. In this case, once the adversary has access to the keys on a compromised anchor, arbitrary but legitimate messages (that can be verified cryptographically) can be sent. Nevertheless, because of the unique pairwise key shared between any two nodes, each node will accept at most one location reference from each (potentially compromised) anchor.

Similar to all previous work [2, 8–12], our scheme relies on the assumption that the majority of the location references are benign. If two colluding location references are selected as the seeds, they will eventually be removed since they cannot grow a majority consistent set. The adversary has to either increase the number of colluding malicious nodes or reduce the number of benign nodes in order to sabotage our localization scheme. For example, the adversary may jam the message sent by benign anchors. This can effectively reduce the set of benign location references. When the malicious colluding set has more nodes than the benign set, our algorithm fails. Dealing with physical layer attacks such as jamming or MAC layer denial of service attacks is outside the scope of this paper. There are techniques such as spread spectrum communication, special coding, and frequency hopping that can provide an efficient mechanism to shield the physical layer against jamming attacks.

Other possible attacks include wormhole attack [6] and Sybil attack [15]. We discuss them next.

Wormhole Attack: A wormhole is a direct tunnel between two points in the network established by the adversary [6]. Under normal network operations, there is no direct link between these two points due to the communication range or other constraints. The direct wormhole link is established by the adversary with the intention of eavesdropping and recording messages at one end (origin) of the wormhole link and replaying them at the other end (destination).

The wormhole attack can be launched against our localization scheme, but it will not be effective. When a remote location reference beacon is replayed at local neighborhood, the location coordinates (*x*,*y*) in the replayed message reflects the original anchor location. The attacker may arbitrarily set the beacon signal strength to manipulate the distance measurement. But the distance manipulation is limited by the maximum anchor transmission range, and thus may not be consistent with the original anchor coordinates. (Indeed, as pointed out in [12], if the manipulated location reference is consistent with the other benign ones, it will not introduce localization error.) In addition, the replayed anchor coordinates also conflict with the local anchor coordinates (since they are not supposed to be within the same communication range to the node to be localized). As a result, the minimum MSE will exceed the measurement error bound, and the malicious location reference will be discarded.

For multiple remote location references to form a consistent set, the adversary not only has to replay these reference messages, but also has to replay them at different local lo-

cations mirroring the same geographic layout at the origin site. Otherwise, the replayed remote messages cannot form a complete consistent set and will be detected. Thus, a simple wormhole link can not effectively disrupt our localization scheme.

Sybil Attack: The adversary can launch a Sybil attack when it has compromised anchor nodes and cloned the victim anchors at different sites of a wireless sensor network [15]. In a Sybil attack, the attacker has access to the cryptographic keys on the compromised anchors and can use the same key in the cloned nodes. As a result, the attacker can distribute arbitrary information using valid node ID and keys and will not be detected by authentication check. The Sybil attack poses a great threat to our scheme since cloned nodes may invalidate our assumption of majority benign anchors.

To defeat Sybil attack, we need to detect and identify nodes that are cloned. A solution to detect the Sybil attack was recently proposed in [18], which relies on a third party to witness duplicated identity among the cloned nodes. In our scheme, assuming that that each anchor node shares a pairwise key with the base station, we can rely on the base station to detect cloning. Each anchor sends back to the base station a cryptographically protected message reporting all heard location references from other anchor nodes. Since the base station has the knowledge of each anchor node's deployment location, it can detect Sybil attack when location conflicts among the reports are discovered. The base station can then flood the whole network to revoke the cloned anchors.

4 Implementation and Evaluation

In this section, we report the implementation of the proposed schemes, as well as the experimental evaluation performed to compare the proposed schemes with all the secure and resilient localization techniques that can be used on the current generation of sensor platforms. In our evaluation, we first perform outdoor field experiments to understand the performance of the proposed schemes in a particular deployment, and then perform a large number of simulation experiments to obtain the performance results in general cases.

4.1 Implementation

We implemented the proposed CMMSE and QCMMSE schemes and all recent secure and resilient localization techniques that can be used on the current generation of sensor platforms (e.g., MICA series of motes) running TinyOS, including EARMMSE [11], voting-based scheme [11], and LMS-based scheme [10]. For the EARMMSE scheme, we set the mean square error threshold $\tau = 0.8\epsilon_{max}$ as discussed in [11], where ϵ_{max} is the maximum distance measurement error. Based on our measurement in the field experiments, we set $\epsilon_{max} = 7.4feet$ and $\tau = 0.8\epsilon = 5.92feet$. We set the number of cells M for the voting-based scheme as $M = 100$ and $M = 225$. For the LMS scheme [10] we set the subset size $n = 4$.

4.2 Field Experiments

We perform a series of outdoor field experiments using MICAz motes to compare the proposed schemes with the other alternatives under investigation. These field experiments offer an opportunity to observe their performance in a realistic setting. We use the RSSI method to measure the distance, since this is the only option for MICAz motes.

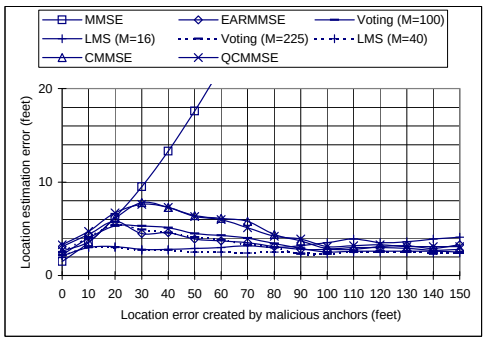
In the outdoor field experiments, we deploy 15 MICAz motes in a 60feet $\times$ 60feet target field. We use 14 motes as anchors to replicate a dense deployment. The anchors broadcast location reference messages periodically. The sensor node with ID 0 (in the middle of the field) is a regular node that needs to estimate its own location.

With this deployment setup, we perform experiments under three attack scenarios. In the first scenario, we randomly select four malicious anchors. Each malicious anchor adds a random location offset of x feet from its true location. The second scenario mimics node collusion. Four randomly selected anchors collude with each other and send out false but consistent location references. In this case, all malicious anchors report a falsified position shifted x feet from its true location in the same direction. In the third scenario, we experiment with a varying number of colluding anchors ranging from 1 to 8 (out of 14 anchors) to examine the impact on the estimated location.

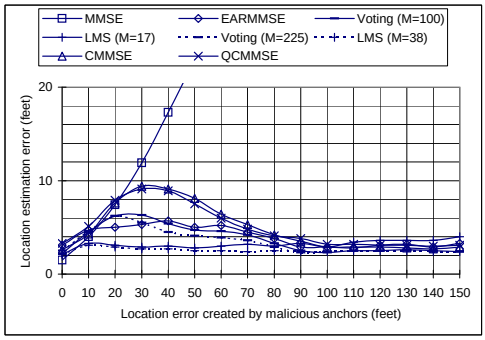
Under each attack scenario, we investigate the resiliency of CMMSE and QCMMSE in terms of localization error and malicious location reference detection rate, and the algorithm efficiency in terms of execution time. In the experiment we vary the error injected from 10 to 150 feet with a 10 feet increment. In all scenarios, we run each scheme 10 rounds for each random placement of malicious anchors. We then compute the average location estimation error and execution time for each scheme.

From Figures 1(a), 1(b), and 1(c), all schemes except for the basic MMSE method have bounded location estimation error, and they can tolerate not only non-colluding malicious anchors but also non-majority colluding malicious anchors. The proposed CMMSE and QCMMSE schemes have slightly higher location estimation errors than EARMMSE, voting-based and LMS schemes; however, the location estimation errors in CMMSE and QCMMSE schemes are in general comparable with the other alternatives. As shown in Figure 1(c), both CMMSE and QCMMSE in fact has smaller location estimation errors than the LMS scheme when the number of colluding malicious anchors is large.

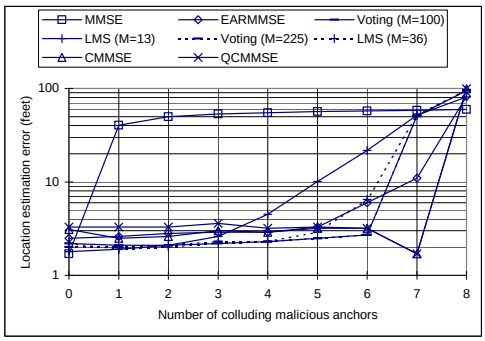
We also investigate the effectiveness of each scheme to filter out malicious location references under different



(a) 4 non-colluding malicious anchors



(b) 4 colluding malicious anchors

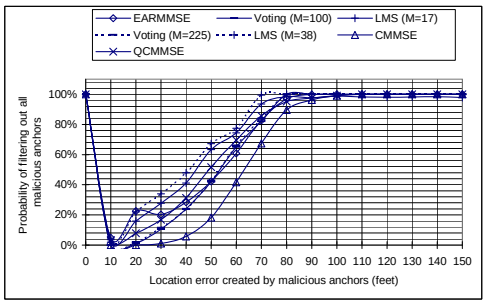


(c) varying number of colluding malicious anchors (error created by malicious anchors is 100 feet)

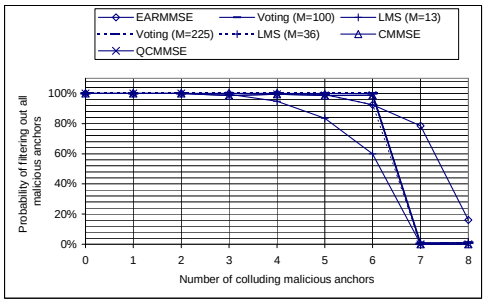
Figure 1: Location estimation error under different scenarios in field experiments

amounts of error injection. For each scheme in each attack scenario, we capture the number of malicious location references that have been successfully identified in each round and calculate the average detection rate over 10 rounds. Figures 2(a), and 2(b) show the success rate of removing malicious location references in our experiments. As we can see, all schemes, including the proposed CMMSE and QCMMSE, have similar results in these figures.

All the schemes under investigation fail to identify and remove all malicious location references when the injected errors are small ($<$70 feet). When the injected error is at 10 feet, no scheme is able to identify and remove the malicious location references. This is because the malicious anchors behave in a way very similar to benign anchors,



(a) 4 colluding malicious anchors



(b) varying number of colluding malicious anchors (error created by malicious anchors is 100 feet)

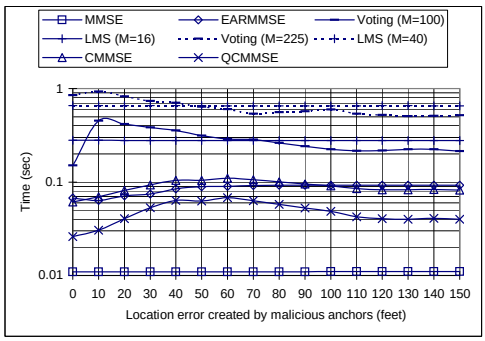
Figure 2: Success rate of removing malicious location references in field experiments

and the injected errors are indistinguishable from normal measurement errors. In such cases, the errors introduced by malicious anchors do not introduce significant error into location estimation.

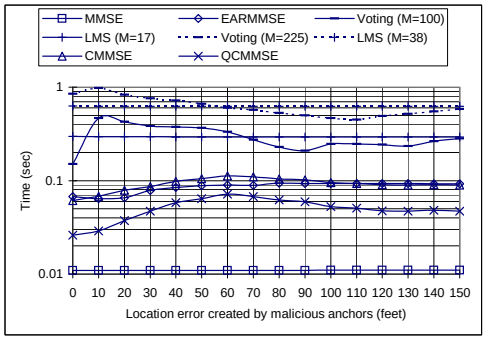
Figure 2(b) provides the results on the malicious location reference detection probability when we have 1$\sim$8 malicious location references. Since the injected error is 100 feet, all schemes are able to identify and remove the malicious location references when the number of colluding ones is small (1$\sim$6). The figure shows that the LMS scheme is the first scheme to break down, and the EARMMSE method provides the best detection rate. All the schemes have very similar results.

The field experiment results indicate that the proposed CMMSE and QCMMSE schemes have slightly worse but comparable performance in terms of location estimation errors compared to the EARMMSE, the voting-based, and the LMS schemes. Next we focus on the efficiency of the schemes under investigation.

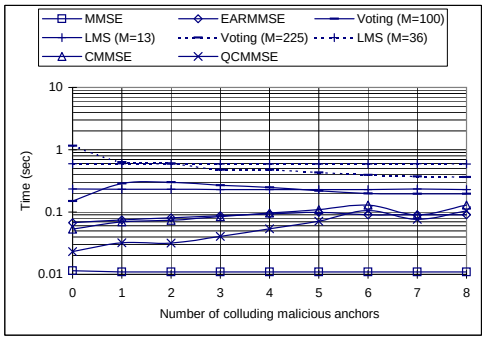
Figure 3 shows the execution time of the schemes under investigation on MICAz motes. Under three evaluation scenarios, we can see the execution time of the voting-based and the LMS schemes are much longer than that of other schemes. EARMMSE and CMMSE have similar execution time, and are much quicker than the voting-based and the LMS schemes, but not as good as QCMMSE. In most cases, QCMMSE is at least twice as fast as EARMMSE. The basic MMSE scheme is obviously the quickest, but it is not



(a) 4 non-colluding malicious anchors



(b) 4 colluding malicious anchors



(c) varying number of colluding malicious anchors (error created by malicious anchors is 100 feet)

Figure 3: Execution time in different scenarios in field experiments

resilient to malicious anchors at all.

Combining the results obtained in the field experiments, we see that the proposed CMMSE and QCMMSE schemes have slightly worse but comparable location estimation errors compared with the alternative schemes, but are in general much more efficient in terms of computation. In particular, QCMMSE requires the least computation among all the resilient schemes while maintaining a similar level of resiliency against malicious anchors.

4.3 Simulation

A limitation of field experiments is that we cannot obtain comprehensive evaluation results through a large number of random deployments. To get better understanding of the performance results, we also perform simulations aimed at verifying and confirming the conclusion drawn from the field experiments. Since the simulation is executed in PC rather than motes, we focus on our evaluation on location estimation errors.

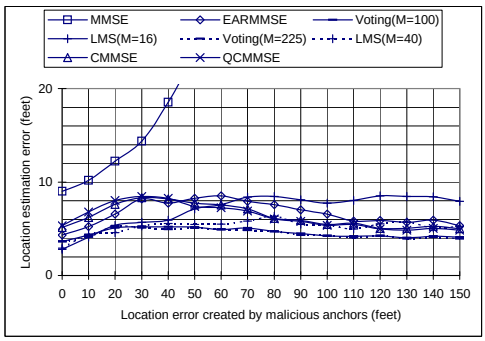
In all simulations, we use the same parameters for the target field, signal range, and maximum measurement error as in the field experiments. For simplicity, we simulate the distance measurement error using a uniform distribution between $-\epsilon_{max}$ and ϵ_{max}. In each simulation, a set of 14 anchors, including both benign and malicious anchors, are deployed in the target field. The non-anchor sensor node, which needs to localize itself, is located at the center of this target field. So we know the true location of this node. We use the same attack scenarios as in the field experiments. In our evaluation, we run 1,000 rounds of simulation in TOSSIM (the simulator of TinyOS) for each data point to calculate the average location estimation error. In each round we randomize the location of each anchor.

We use the same deployment of anchor nodes as in the field experiments to verify results. Figure 4 shows the location estimation errors from simulation and they are very similar to the results obtained in the field experiments. The subtle difference is due to the simulated measurement errors.

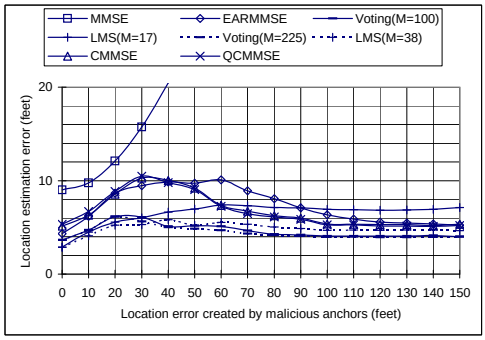
5 Conclusion

In this paper, we developed CMMSE and its variation QCMMSE to efficiently tolerate attacks against localization in WSN. Through random seed selection and triplet examination, CMMSE and QCMMSE use the basic MMSE method to filter out malicious location references efficiently. The algorithm is well suited for the current generation of low end wireless sensor nodes. To evaluate the proposed schemes, we implemented CMMSE, QCMMSE, and all recent secure and resilient localization schemes that can be used on the current generation of sensor platforms, including EARMMSE [11], the voting-based scheme [12], and the LMS-based scheme [10]. We have performed thorough experimental evaluation through both outdoor field experiments and simulation. Our results conclude that the newly proposed schemes, particularly QCMMSE, has the fastest execution among all the resilient localization schemes, and at the same time provide a comparable degree of resiliency against malicious attacks.

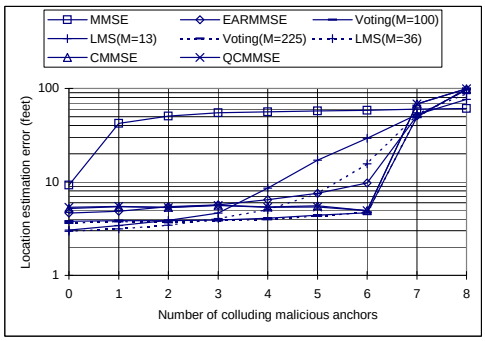
All the existing secure and resilient localization schemes, including CMMSE and QCMMSE, require a majority of benign location references. In the future, we will investigate resilient and/or detection techniques complementary to resilient estimation, so that we can provide resilient location estimation even when the colluding malicious anchors form the majority.



(a) 4 non-colluding malicious anchors



(b) 4 colluding malicious anchors



(c) varying number of colluding malicious anchors (error created by malicious anchors is 100 feet)

Figure 4: Location estimation error in simulation (same anchor location as field experiments)

References

[1] N. Bulusu, J. Heidemann, and D. Estrin. GPS-less low cost outdoor localization for very small devices. In *IEEE Personal Communications Magazine*, pages 28–34, October 2000.

[2] S. Capkun and J. Hubaux. Secure positioning of wireless devices with application to sensor networks. In *Proceedings of IEEE InfoCom'05 (to appear)*, 2005.

[3] H. Chan, A. Perrig, and D. Song. Random key predistribution schemes for sensor networks. In *IEEE Symposium on Research in Security and Privacy*, pages 197–213, 2003.

[4] L. Doherty, K. S. Pister, and L. E. Ghaoui. Convex optimization methods for sensor node position estimation. In *Proceedings of INFOCOM'01*, 2001.

[5] T. He, C. Huang, B. M. Blum, J. A. Stankovic, and T. F. Abdelzaher. Range-free localization schemes in large scale sensor networks. In *Proceedings of ACM MobiCom 2003*, 2003.

[6] Y. Hu, A. Perrig, and D. B. Johnson. Wormhole detection in wireless ad hoc networks. Technical Report TR01-384, Department of Computer Science, Rice University, Dec 2001.

[7] C. Karlof, N. Sastry, and D. Wagner. TinySec: A link layer security architecture for wireless sensor networks. In *Proceedings of the 2nd ACM Conference on Embedded Networked Sensor Systems (SensSys 2004)*, November 2004.

[8] L. Lazos, S. Capkun, and R. Poovendran. Rope: Robust position estimation in wireless sensor networks. In *Proceedings of the Fourth International Conference on Information Processing in Sensor Networks (IPSN '05)*, April 2005.

[9] L. Lazos and R. Poovendran. Serloc: Robust localization for wireless sensor networks. *ACM Transactions on Sensor Networks*, 1(1):73–100, August 2005.

[10] Z. Li, W. Trappe, Y. Zhang, and B. Nath. Robust statistical methods for securing wireless localization in sensor networks. In *Proceedings of the Fourth International Conference on Information Processing in Sensor Networks (IPSN '05)*, April 2005.

[11] D. Liu, P. Ning, and W. Du. Attack-resistant location estimation in wireless sensor networks. Technical Report TR-2004-29, North Carolina State University, Department of Computer Science, 2004. Revised August 2005.

[12] D. Liu, P. Ning, and W. Du. Attack-resistant location estimation in wireless sensor networks. In *Proceedings of the Fourth International Conference on Information Processing in Sensor Networks (IPSN '05)*, April 2005.

[13] D. Liu, P. Ning, and R. Li. TinyKeyMan: Key management for sensor networks. `http://discovery.csc.ncsu.edu/software/TinyKeyMan/`.

[14] R. Nagpal, H. Shrobe, and J. Bachrach. Organizing a global coordinate system from local information on an ad hoc sensor network. In *IPSN'03*, 2003.

[15] J. Newsome, R. Shi, D. Song, and A. Perrig. The sybil attack in sensor networks: Analysis and defenses. In *Proceedings of IEEE International Conference on Information Processing in Sensor Networks (IPSN 2004)*, April 2004.

[16] D. Niculescu and B. Nath. Ad hoc positioning system (APS) using AoA. In *Proceedings of IEEE INFOCOM 2003*, pages 1734–1743, April 2003.

[17] D. Niculescu and B. Nath. DV based positioning in ad hoc networks. In *Journal of Telecommunication Systems*, 2003.

[18] B. Parno, A. Perrig, and V. Gligor. Distributed detection of node replication attacks in sensor networks. In *IEEE Symposium on Security and Privacy*, May 2005.

[19] A. Savvides, C. Han, and M. Srivastava. Dynamic fine-grained localization in ad-hoc networks of sensors. In *Proceedings of ACM MobiCom '01*, pages 166–179, July 2001.

[20] C. Wang, A. Liu, and P. Ning. Cluster-based minimum mean square estimation for secure and resilient localization in wireless sensor networks. Technical Report TR-2007-15, North Carolina State University, 2007.