Game Theoretic Secure Localization in Wireless Sensor Networks

Susmit Jha
Strategic CAD Labs, Intel
Email: susmit.jha@intel.com

Stavros Tripakis, Sanjit A. Seshia
EECS, UC Berkeley
Email: stavros,sseshia@eecs.berkeley.edu

Krishnendu Chatterjee
IST, Austria
Email: Krishnendu.Chatterjee@ist.ac.at

***Abstract*—Wireless sensor networks (WSNs) composed of low-power, low-cost sensor nodes are expected to form the backbone of future intelligent networks for a broad range of civil, industrial and military applications. These sensor nodes are often deployed through random spreading, and function in dynamic environments. Many applications of WSNs such as pollution tracking, forest fire detection, and military surveillance require knowledge of the location of constituent nodes. But the use of technologies such as GPS on all nodes is prohibitive due to power and cost constraints. So, the sensor nodes need to autonomously determine their locations. Most localization techniques use *anchor* nodes with known locations to determine the position of remaining nodes. Localization techniques have two conflicting requirements. On one hand, an ideal localization technique should be computationally simple and on the other hand, it must be resistant to attacks that compromise anchor nodes. In this paper, we propose a computationally light-weight game theoretic secure localization technique and demonstrate its effectiveness in comparison to existing techniques.**

I. Introduction

Recent technological advances in microelectromechanical (MEMS) devices, low-cost sensors, low-power system-on chips and wireless technologies have enabled development and deployment of sensor nodes for a variety of civil, industrial and military applications such as water surveillance [5], military surveillance, traffic monitoring [12], habitat monitoring [25], forest-fire detection and tracking [31] and flood monitoring [2]. These sensor nodes are capable of collecting data of interest from environment, processing it, and transmitting it to the base stations over wireless. Sensor nodes are often used in large numbers in many of these applications. Multiple sensors are deployed in a given area and they form a wireless sensor network (WSN) that communicates to the base station.

WSNs are often deployed through random in-mass spreading. For example, WSNs used to detect and track forest fire or flood are built by spreading sensors through aircrafts or ground vehicles. Apart from sensor data, these applications also need to know the location of the sensor nodes transmitting the data. Location information is often important for correctly interpreting the sensor measurements. Further, all sensor nodes in WSNs may not be in direct communication with the base station or with each other, and communication among two nodes often requires routing through intermediate nodes. Power constraints prohibit energy inefficient broadcasting and necessitate intelligent routing and cooperative communication which, in turn, rely on localization [1], [17], [23]. Many robotic swarm applications that require distributed formation and coordination also rely on localization to ensure connectivity and coordination between the swarm nodes [21], [24]. Equipping each sensor node with GPS for direct localization is not practical due to power and cost constraints. Only a small fraction of the sensor nodes in a WSN can be equipped with GPS or have a known fixed location. As a result, the location of sensor nodes are often not known a priori and need to be determined after deployment. Localization of sensor nodes is critical to the functioning of WSNs.

Most sensor localization techniques rely on *anchor* nodes (also called beacons) with known locations to identify the location information of the remaining nodes. Anchor nodes transmit beacon signals containing their locations using which other nodes can estimate their distance from the anchors and deduce their location. Estimation of distance can be done by measuring physical metrics such as received signal strength (RSS) [8], time of arrival [28], time difference of arrival [13] and hop count [29].

WSNs often operate in hostile environments where an adversary can try to introduce error in sensor localization of the nodes and consequently, prevent the functioning of the WSNs [18], [32]. The adversary can compromise sensor nodes, steal secret keys, impersonate anchor nodes and provide misleading information to constituent nodes in the WSNs. This necessitates techniques to filter out and neutralize the effect of incorrect measurements reported by the compromised *malicious* anchor nodes in order to ensure accurate localization of sensor nodes. Secured localization techniques [3], [9], [18], [32] are needed to enable accurate determination of sensor positions in presence of malicious anchor nodes transmitting misleading information. Further, the sensors have limited memory, computation and energy resources, and hence, secured localization must be energy-efficient and computationally light-weight.

In this paper, we propose a novel light-weight game theoretic secure localization technique which can be used for accurate localization of sensor nodes in presence of malicious anchor nodes transmitting misleading information. The novel contributions made in this paper are as follows:

- We present a game theoretic technique to learn reputation weights of the anchor nodes being used for sensor localization. Low weights mean that the anchor nodes are compromised and malicious.
- We use the above learnt weights to enable accurate sensor localization by filtering the misleading information from malicious nodes.
- We experimentally compare our approach to existing secure localization techniques and illustrate its effectiveness.

II. Preliminaries

WSNs are often formed by nondeterministic dispersal of sensors throughout an area of interest to detect and possi-

bly track events of interest in the area. Each sensor node is equipped with wireless communication and elementary computing capabilities. Sensor nodes are often grouped into clusters and each cluster is assigned one or more gateway nodes which are capable of long haul communication. These gateway nodes communicate to command nodes located at much longer distances. They collect data from sensor nodes and transmit it to command nodes. They also relay commands back to sensor nodes. A typical WSN is illustrated in Figure 1.

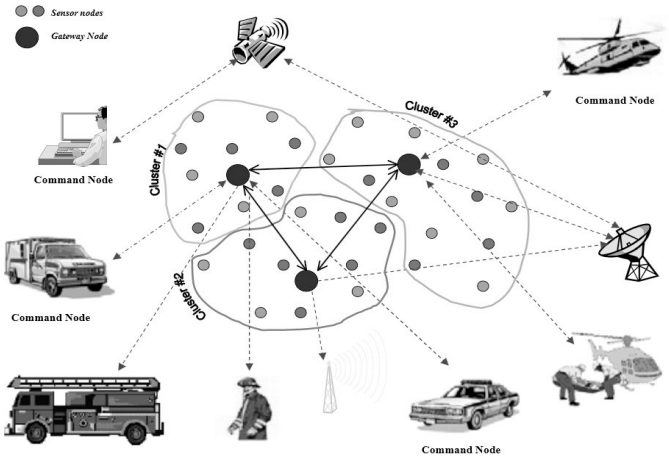


Fig. 1: A typical Wireless Sensor Network [30]

Sensor nodes in WSNs are of two types: anchor nodes for which the location can be directly known, and other normal sensor nodes for which localization needs to be performed. Each sensor node can estimate the distance to the neighboring nodes that are within its transmission range. This can be done using metrics such as received signal strength (RSS) [8], time of arrival (ToA) [28], time difference of arrival (TDoA) [13].

Localization of sensor nodes is done using a number of anchor nodes. Each anchor node i (or a small set of them) provides distance estimates $\mathtt{ed}_i$ for the actual distance $\mathtt{d}_i$ from the sensor node using RSS, ToA, TDoA or a combination of these. The estimate $\mathtt{ed}_i$ is approximate, that is, $\mathtt{ed}_i = \mathtt{d}_i + \eta_i$ where η_i is Gaussian noise. The positions of anchor nodes are fixed and known a priori, that is, $\mathbf{P}_i$ for anchor node i. After receiving the distance estimates, the *most likelihood estimate (MLE)* of the sensor node's position $\mathbf{P}$ is given by

$$\arg\min_{\mathbf{P}}(\sum_i((\mathbf{P}-\mathbf{P}_i)-\mathtt{ed}_i)^2)$$

where $(\mathbf{P}-\mathbf{P}_i)$ denotes the Euclidian distance between the two positions in 2D or 3D space. Thus, the localization of sensor nodes is done in two steps: transmission and receipt of information about distance estimates from anchor nodes; followed by computation of sensor position from the estimates.

III. Problem Formulation

Let N be the number of sensor nodes in a WSN and let M be the number of anchor nodes. The position of the anchor node i is denoted by $\mathbf{P}_i$. The position of the sensor node being localized is denoted by $\mathbf{P}$. Each sensor node performs its localization independently. The measurements available to a sensor node for localization are $\{(\mathbf{P}_1,\mathtt{ed}_1),(\mathbf{P}_2,\mathtt{ed}_2)\ldots(\mathbf{P}_M,\mathtt{ed}_M)\}$ where $\mathtt{ed}_i$ is the approximate estimated distance between the sensor node and the anchor i with position $\mathbf{P}_i$. The distance estimate is noisy, that is, $(\mathtt{ed}_i = \mathtt{d}_i + \eta_i)$. We get the following system of equations:

$$||\mathbf{P}_i - \mathbf{P}|| + \eta_i = \mathtt{ed}_i, \;\; i = 1, 2, \ldots, M$$

As mentioned in Section II, the most likelihood estimate (MLE) of $\mathbf{P}$ is $\arg\min_{\mathbf{P}}(\sum_i((\mathbf{P}-\mathbf{P}_i)-\mathtt{ed}_i)^2)$. If an anchor node i is compromised by adversarial attack, it can intentionally send wrong information $(\mathbf{P}_i,\mathtt{ed}_i)$ to force the MLE position estimate far from the correct position.

Adversary can choose to attack localization in WSNs using one of several ways [7], [10], [15]. The attack could involve compromise of sensor nodes [4]. Normal sensor nodes could be physically captured and reprogrammed, or attacker could employ larger computing resources such as laptops to attack normal nodes, recover network keys, circumvent any security mechanism on the nodes, and reprogram them. Reprogrammed nodes can provide incorrect localization information. The attacker could also replicate nodes [16], [27] if the nodes have been compromised and their authentication/encryption keys have been extracted. It can produce many replicas with the same identify as the captured sensor node and integrate these replica with the WSN. Flooding the WSN with replicas of captured sensor node allows easy injection of false localization information into the WSN. Using nodes with larger computational resource, the attacker can launch Sybil attacks [20] where same node replicates more than one node. Another common impersonation attack is wormhole attack [14] in which the attacker can capture information transmitted by a legitimate anchor node and replay it elsewhere in the WSNs to attack sensor localization. Consequently, all these attacks will eventually result in sensor nodes receiving wrong information $(\mathbf{P}_i,\mathtt{ed}_i)$ corresponding to anchor i. We refer to the anchor nodes transmitting incorrect information as *malicious* anchor nodes.

We can classify attacks on sensor localization into two classes: *uncoordinated attacks* where malicious nodes act independently, and *coordinated attacks* where malicious anchor nodes cooperate.

- Uncoordinated attacks: Each malicious anchor node acts independently and modifies $(\mathbf{P}_i,\mathtt{ed}_i)$ from its correct value. Any modification in position can be easily transformed into equivalent modification in distance, and hence, we only need to consider modification in $\mathtt{ed}_i$. So, the reported distance from an anchor node is
$$\mathtt{ed}_i = \begin{cases} ||\mathbf{P}_i - \mathbf{P}|| + \eta_i + u_i & \text{if node } i \text{ is malicious} \\ ||\mathbf{P}_i - \mathbf{P}|| + \eta_i & \text{otherwise} \end{cases}$$
where u_i is the perturbation added by malicious node in the distance reported to the sensor node.
- Coordinated attacks: Another kind of attack is a coordinated attack launched by multiple malicious nodes acting together to make the localizing sensor node estimate its position as $\mathbf{P}_{bad}$ instead of its true location $\mathbf{P}$. So, the reported distance from the anchor node is
$$\mathtt{ed}_i = \begin{cases} ||\mathbf{P}_i - \mathbf{P}_{bad}|| + \eta_i & \text{if node } i \text{ is malicious} \\ ||\mathbf{P}_i - \mathbf{P}|| + \eta_i & \text{otherwise} \end{cases}$$

The secure sensor localization problem takes as input the position and estimated distances from all the M anchor nodes:

$\{(\mathbf{P}_1, \mathtt{ed}_1), (\mathbf{P}_2, \mathtt{ed}_2) \ldots (\mathbf{P}_M, \mathtt{ed}_M)\}$. Some of the anchor nodes (say m) might be malicious and reporting incorrect values deliberately. As described earlier, the incorrect values reported by malicious nodes could be coordinated (pointing to a common wrong position $\mathbf{P}_{bad}$) or uncoordinated. The number of malicious nodes must be less than the number of honest secure nodes, that is, $m < M/2$. The output is the most likelihood estimate (MLE) of the correct sensor position $\mathbf{P}$.

IV. Secure Sensor Localization

In this section, we describe our secure sensor localization approach. In order to accurately estimate the correct sensor position in presence of malicious anchor nodes, we need to be able to identify malicious sensor nodes and filter out their misleading inputs. We accomplish this using a combination of least trimmed square algorithm (LTS) [22] and game theoretic aggregation (GTA) algorithm [26]. Before describing our approach in further detail, we briefly summarize these two key components: LTS and GTA. Least trimmed squares regression (LTS) [22] is a statistical approach used in regression to identify and remove regression factors which are anomalous. The learnt model only includes non-anomalous factors and hence, provides a more accurate prediction. LTS works in two phases. In the first phase, LTS performs gradient search for least square regression model. This continues till the cumulative gradient has reached below some user-defined threshold. After that, all the factors which have high gradient components are pruned out iteratively and the gradient search continues with the remaining factors. The intuition behind the approach is that the anomalous factors would have high gradients in comparison to the majority factors as the regression converges. Thus, pruning the factors with high gradient eliminates the anomalous factors. A key weakness of LTS is its reliance on the threshold parameter to trigger the phase of pruning anomalous factors. To improve LTS, a single phase weight-based combination of factors which does not need any threshold specification can be used by combining GTA with LTS. Game theoretic aggregation (GTA) considers the problem of combining outputs from a number of predictors to construct a more accurate predictive model [26]. In each iteration, the learner makes a prediction based on weighted combination of the predictions from each predictor expert. The environment or nature then chooses an outcome for the parameter of interest. The difference between the learner's prediction and the nature's true outcome determines the penalty incurred by the learner. Based on the incurred penalty and the predictions made by each predictor, multiplicative update is made to each weight. The goal of the learner is to minimize the cumulative penalty over iterations. In order to minimize the cumulative penalty, the learner can choose a strategy of combining the output of individual predictors (by picking the most suitable weights for combination) such that its prediction is closer to the true outcome. Our approach on secure sensor localization exploits ideas from both techniques: LTS and GTA. LTS is used to compute the least square MLE of sensor location using gradient search. Each individual anchor node is viewed as a predictor in GTA technique, and the problem of identifying malicious nodes is similar to identifying poor predictors. Hence, secure sensor localization becomes the problem of finding best GTA strategy of combining gradients from individual anchor nodes such that the combined gradients can be used to compute accurate sensor location.

The algorithm for secure sensor localization is presented in Algorithm 1. It computes least square MLE of the sensor location over k iterations similar to [6], [22]. But in contrast to [6], the MLE estimation takes into account the reputation of each sensor node by reducing the influence of malicious nodes using weights learnt through a game theoretic online algorithm [26]. We need to find $\mathbf{P}$ that minimizes $(\sum_j((\mathbf{P}_j - \mathbf{P}) - \mathtt{ed}_j)^2)$. Taking the partial derivative of the above expression and setting them to zero for minimizing the expression, we obtain the following set of equations:

$$||\mathbf{P}_j - \mathbf{P}|| - \mathtt{ed}_j = 0 \ \mathtt{for}\ j = 1\ to\ M$$

So, we need to find $\mathbf{P}$ that is the solution of above over-constrained set of equations. But since, the measurements from anchor nodes are noisy and some anchor nodes are malicious, we would only have an approximate solution.

Data: $(\mathbf{P}_j, \mathtt{ed}_j)$ from M anchor nodes j with less than $M/2$ malicious nodes, Number of iterations k, Step size function $\delta(i)$, Convergence parameter τ

Result: Sensor location MLE estimate $\mathbf{P}$

initialization: $i = 0$; $\hat{\mathbf{P}}(0) =$ a random point $\mathbf{P}_0$;
$w_j(1) = 1$ for $j = 1$ to M;
for $i = 1$ ***to*** k **do**
  for $j = 1$ ***to*** M **do**
    $f_j(i) = (||\mathbf{P}_j - \hat{\mathbf{P}}(i-1)|| - \mathtt{ed}_j) \times \frac{\mathbf{P}_j - \hat{\mathbf{P}}(i-1)}{||\mathbf{P}_j - \hat{\mathbf{P}}(i-1)||}$;
  end
  $maj =$ set of $M/2$ smallest derivatives $f_j(i)$;

$$f(i) = \frac{\sum_{j=1}^{M} f_j(i) w_j(i)}{\sum_{j=1}^{M} w_j(i)}; f^*(i) = \frac{\sum_{j \in maj} f_j(i) w_j(i)}{\sum_{j \in maj} w_j(i)} ;$$

  $\hat{\mathbf{P}}(i) = \hat{\mathbf{P}}(i-1) + \delta(i) \times \frac{f(i)}{||f(i)||}$;
  $W(i) = \sum_{j=1}^{M} w_j(i) \exp(-\tau(||f_j(i) - f^*(i)||))$;
  for $j = 1$ ***to*** M **do**
    $w_j(i+1) = \frac{w_j(i) \exp(-\tau(||f_j(i) - f^*(i)||))}{W(i)}$;
  end
end
$\mathbf{P} = \hat{\mathbf{P}}(k)$

Algorithm 1: Game Theoretic Secure Sensor Localization

Starting from a random initial position $\mathbf{P}_0$ in the deployment area, Algorithm 1 uses gradient descent to find the least square MLE of sensor location $\mathbf{P}$. The weights denoting reputation of the anchor node are uniformly initialized to 1. The estimated sensor location in i-th iteration of Algorithm 1 is denoted by $\hat{\mathbf{P}}(i)$. Over iterations, $\hat{\mathbf{P}}(i)$ will converge to the MLE estimate. The weights $w_j(i)$ of malicious nodes are expected to go down while the weight of secure nodes would go up over iterations. At the i-th iteration, we compute $f_j(i)$ as the partial derivative of the least square most likelihood estimate with respect to the anchor node j. If this derivative is high, it means that the anchor node is indicating that the sensor position is far from the estimate obtained in previous iteration $\hat{\mathbf{P}}(i-1)$. Most anchor nodes would have high derivatives in the first few iterations. But after a few iterations, the malicious anchor nodes will have higher derivatives than other

honest anchor nodes. The overall derivative $f(i)$ is computed as the weighted sum of the partial derivatives with respect to each anchor node. This is used to determine the shift in estimated sensor location $\hat{\mathbf{P}}(i)$ in each iteration using an adaptive descent function $\delta(i)$. The function $\delta(i)$ can also be set to a constant value for non-adaptive gradient descent. We employ adaptive gradient descent and use the function in [6]: $\delta(i) = 15 - 15(i-1)/k$. We also compute the sum $f_j^*(i)$ of $M/2$ smallest derivatives. This set can not be composed of only malicious sensors since the number of malicious sensors $m < M/2$. Over iterations, as the derivatives of malicious nodes become higher, the difference between the partial derivatives of malicious anchor nodes and $f_j^*(i)$ would also become higher. Hence, we use this difference to revise the reputation weight $w_j(i)$ of the anchor nodes in each iteration. If the difference is high, the reputation weight is decreased and if the difference is small, the reputation weight is increased. The revision of the reputation weights is controlled by the convergence parameter τ. Over iterations, the weights of malicious nodes will become very small and hence, their contribution to the gradient $f(i)$ and hence, the contribution to the shift in position $\hat{\mathbf{P}}(i)$ will also become small. Thus, Algorithm 1 can successfully identify the malicious nodes (when the number of malicious nodes $m < M/2$), and perform secure sensor localization which ignores the misleading input from malicious nodes. The identification of malicious nodes and secure localization can be refined by increasing the number of iterations k. As k becomes larger and larger, the impact of each sensor is analyzed using smaller changes in sensor location estimation. Hence, more accurate identification and filtering of the malicious nodes can be done with larger number of iterations k.

In Algorithm 1, $f^*(i)$ is computed using the anchor nodes with smallest gradients. If the weights learnt by the algorithm correctly determine the reputation of the anchor nodes, the weights of malicious anchor nodes (having high gradients) would be small and the gradient $f(i)$ used in the algorithm should be close to $f^*(i)$. We formally prove in Theorem 1 that there is a choice of τ such that the average difference between $f(i)$ and $f^*(i)$ is bounded and this difference goes to 0 as the number of iterations k are increased.

Theorem 1: For the choice of convergence parameter $\tau = \sqrt{8\frac{\ln M}{k}}$, the average error in gradients, $\sum_i ||f(i) - f^*(i)||/k$, computed by Algorithm 1 over k iterations is $\sqrt{\frac{\ln M}{2k}}$. As k becomes larger and larger, the average gradient error converges to 0.

Proof: Let $\Delta_j(i) = ||f_j(i) - f^*(i)||$.

So, $\frac{W(i)}{W(i-1)} = \frac{\sum_{j=1}^{M} w_j(i)\exp(-\tau(\Delta_j(i)))}{\sum_{j=1}^{M} w_j(i-1)\exp(-\tau(\Delta_j(i-1)))}$

We can rewrite the above compactly as $\frac{W(i)}{W(i-1)} = \sum_{j=1}^{M} \hat{w}_j(i)\exp(-\tau(\Delta_j(i)))$. Now we define a new random variable l with probability distribution, $P(l=j) = \hat{w}_j(i)$.

$$\frac{W(i)}{W(i-1)} = \mathbf{E}_l[\exp(-\tau(\Delta_j(i)))]$$

Using Hoeffding inequality [11],

$$\mathbf{E}_l[\exp(-\tau(\Delta_j(i)))] \leq \exp(-\tau\mathbf{E}_l[(\Delta_j(i))] + \tau^2/8)$$

From Jensen's inequaility [11],

$\mathbf{E}_l[(\Delta_j(i))] \geq (||||E_l[f_j(i)] - f^*(i)|| = ||f(i) - f^*(i)|| = \Delta_i$

So, $\frac{W(i)}{W(i-1)} \leq \exp(-\tau\Delta_i + \tau^2/8)$,

that is, $\frac{W(k)}{W(k-1)} \cdots \frac{W(2)}{W(1)} \leq \exp(-\tau\sum_{i=1}^{k}\Delta_i + k\tau^2/8)$,

that is, $W(k) \leq M\exp(-\tau\sum_{i=1}^{k}\Delta_i + k\tau^2/8)$. Also, $W(k) \geq 1$ since there is at least one anchor node that is not malicious and has a weight of 1. Hence,

$$\log M - \tau\sum_{i=1}^{k}\Delta_i + k\tau^2/8 \geq 0 \text{ i.e. } \sum_{i=1}^{k}\Delta_i \leq \log M/\tau + k\tau/8$$

Choosing $\tau = \sqrt{8\frac{\ln M}{k}}$ and denoting the average gradient error $\Delta_{avg} = \sum_{i=1}^{k}\Delta_i/k$, $\Delta_{avg} \leq \sqrt{\frac{\ln M}{2k}}$. Hence, as $k \to \infty$, $\Delta_{avg} \to 0$, that is, the average gradient error goes to 0. ■

Thus, the weights computed by Algorithm 1 can identify malicious nodes and neutralize the impact of their misinformation on localization. The error in localization now depends mainly on the error in $f^*(i)$ which is dependent on the noise η in the distance estimates from the honest nodes.

V. Related Work

Secure sensor localization in WSNs has received a lot of attention with the rise of internet of things. In this section, we describe a representative set of related work and compare those with our proposed approach. Greedy sensor localization using a voting scheme has been proposed in [19]. The localization area is partitioned into a grid. Voting is done for each grid point by the anchor nodes. If the distance reported by anchor nodes is the same as the distance between a grid point and the anchor node, the anchor node votes for that grid point as candidate sensor location. At the end, the grid point with the maximum number of votes is selected as the estimated sensor location. This approach also assumes that the number of malicious nodes is less than half the total number of anchor nodes. The voting scheme is less robust to noises in the distance reported by honest anchor nodes. Each anchor node either votes for a grid point or does not vote for it. So, a small number of votes would be cast even for a grid point that has many anchor nodes reporting distances close to it but not sufficiently close to trigger a vote for the grid point. Another technique [18] used for localization uses random sample consensus algorithm. It uses several subsets of anchor nodes to identify candidate sensor locations, and then chooses the solution from one of these sets that minimizes the median of the residues with respect to all anchor nodes. This subset sampling approach proposed in [18] also requires that the number of malicious anchor nodes is less than half the total number of anchor nodes. This technique relies on sampling a number of subsets of anchor nodes and is computationally very expensive. In presence of a large number of malicious nodes, the number of subsets to be considered for accurate localization would also be large. Sensor localization through gradient descent has also

Approach	Complexity	Runtime (ms)	Relative Experimental Complexity
Voting Scheme	$O(g^2M)$	12.4	7.29
Subset Sampling	$O(sM)$	26.8	15.76
Gradient Descent	$O(kM)$	4.8	2.82
Proposed Approach	$O(kM)$	1.7	1

TABLE I: Computation Complexity

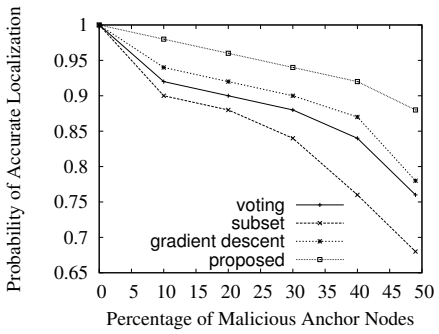
been proposed in literature [6]. The existing gradient descent technique relies on two phases of descent. The first takes into account input from all sensor nodes. After reaching a particular user-specified threshold of convergence, it switches to second mode in which the higher gradients are pruned out iteratively. In contrast, we use weights to capture reputation of the anchor nodes and use a game theoretic approach to automatically learn these reputation weights to identify malicious nodes and ignore their information in sensor localization.

We summarize the computational complexity of localizing a sensor node in each of the techniques described above in Table I. As used in the rest of the paper, M is the number of anchor nodes, k is the number of iterations in gradient descent, g is the width and length of the grid in the voting scheme, and s is the number of subsets in the subset sampling scheme which grows with the number of malicious nodes m. So, our technique has the same computational complexity as the gradient descent technique, and unlike the subset sampling scheme, it is independent of the number of malicious nodes in the WSNs. Along with the computational complexity, we also provide the average runtime for different techniques from our experiments described in Section VI. The last column of Table I shows the relative experimental computational complexity of the techniques normalized to the proposed approach. The computational complexity and relative speed-up in runtime clearly indicate that the proposed approach is computationally light-weight and hence, it would be more energy-efficient.

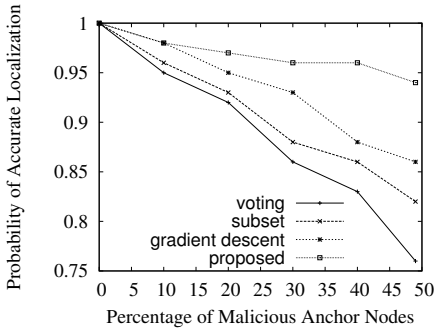
VI. Experimental Evaluation

The proposed algorithm is compared with existing secure localization techniques: voting scheme (VS) [19], subset sampling scheme (SS) [18] and gradient descent (GD) [6]. We use the same simulation parameters as those used in [19] and [6]. A set of 30 anchor nodes are randomly deployed in an area of $60m \times 60m$. The standard deviation in distance measurement reported by anchor nodes is $2m$. For voting based techniques, we use a grid with unit size of $1m \times 1m$, that is, $g = 60$. For the subset sampling technique, we use 20 subsets each consisting of 4 nodes. For gradient descent techniques, we use $k = 200$ iterations, and the threshold to switch is 0.9. The results are obtained by averaging over 1500 simulation runs. We consider both types of attacks: uncoordinated attacked and coordinated attacks. We conduct two sets of experiments - the first experiment compares the robustness of the localization technique to increase in the number of malicious nodes, and the second experiment compares the accuracy of the localization technique with varying the strength of attack of the malicious nodes.

In the first experiment, we vary the number of malicious nodes from 0% to 49%. For uncoordinated attacks, we use s.d. of $u_i = 4m$. For coordinated attacks, we use $\mathbf{P}_{bad}$ such that the *distance of attack* $||\mathbf{P}_{bad} - \mathbf{P}|| = 22m$. We measure the probability of the computed sensor location being within s.d. of u_i i.e. $4m$ from its actual location over 1500 runs.

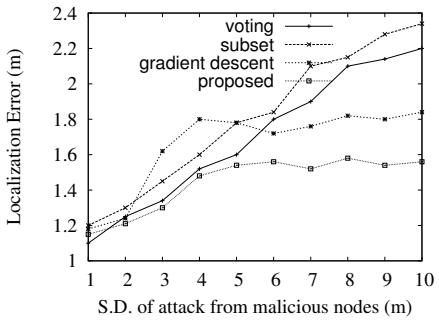


(a) Uncoordinated Attacks

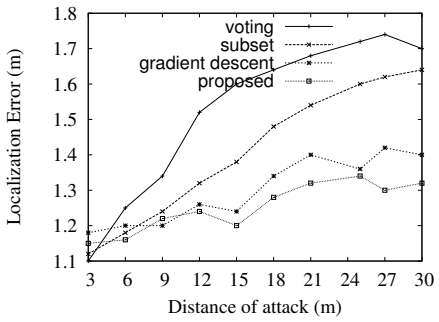


(b) Coordinated Attacks

Fig. 2: Probability of Accurate Localization



(a) Uncoordinated Attacks



(b) Coordinated Attacks

Fig. 3: Accuracy with 30% Malicious Anchor Nodes

The results from this experiment are presented in Figure 2. The proposed technique is more robust to increase in number of malicious anchor nodes in the network. Even with 49% malicious anchor nodes in the case of uncoordinated attacks, the proposed approach is able to localize correctly with a probability of 88% while GD, VS and SS localize correctly only with a probability of 78%, 76% and 68% respectively. In case of coordinated attacks, the attacking nodes are more easily identifiable since they point to a common misleading location and their gradients are equally high as the algorithm converges to the sensor location suggested by majority honest nodes. The proposed algorithm localizes correctly with a probability of 94% in contrast to 86%, 82% and 76% for GD, SS and VS. VS is less robust to coordinated attacks since it relies on voting.

In the second experiment, we vary the error being introduced by the malicious anchor nodes but keep the percentage of malicious nodes fixed to 30%. We vary the standard deviation of error u_i being introduced by uncoordinated malicious nodes from $1m$ to $10m$. In case of coordinated attacks, we vary $err = ||\mathbf{P}_{bad} - \mathbf{P}||$ from $1m$ to $30m$. We measure the accuracy of the sensor localization using different techniques. The results from this experiment are presented in Figure 3. The proposed approach saturates at a much lower error compared to GD, VS and SS in both cases: 1.55 for uncoordinated attacks and 1.3 for coordinated attacks. Thus, the proposed technique has better accuracy compared to the existing approaches.

VII. Conclusion

We presented a novel approach to secure sensor localization. It combines game theoretic reputation determination and gradient descent search for secure sensor localization. We showed that the proposed technique is computationally lightweight. We also experimentally illustrated that the proposed approach computes sensor location more accurately and is more robust to increase in the number of malicious anchor nodes. In future work, we plan to investigate the application of our technique to a mobile setting where sensor nodes can move over time. We also plan to investigate the use of localization history to better detect compromised malicious nodes.

References

[1] Young bae Ko and Nitin H. Vaidya. Location-aided routing (lar) in mobile ad hoc networks. In *MobiCom98*, 1998.

[2] Elizabeth A. Basha, Sai Ravela, and Daniela Rus. Model-based monitoring for early warning flood detection. In *SenSys '08*, pages 295–308, New York, NY, USA, 2008. ACM.

[3] Honglong Chen, Wei Lou, and Zhi Wang. A novel secure localization approach in wireless sensor networks. *EURASIP J. Wireless Comm. and Networking*, 2010.

[4] Xiangqian Chen. *Defense Against Node Compromise in Sensor Network Security: Node Compromise Modeling in Sensor Network Security*. LAP Lambert Academic Publishing, Germany, 2009.

[5] Y. Dogan E. Cayirci, H. Tzcan and V.Coskun. Wireless sensor networks for underwater surviellance systems. 4:431–446, 2006.

[6] R. Garg, A.L. Varna, and Min Wu. An efficient gradient descent approach to secure localization in resource constrained wireless sensor networks. *Information Forensics and Security, IEEE Transactions on*, 7(2):717–730, April 2012.

[7] Ravi Garg, Avinash L. Varna, and Min Wu. An efficient gradient descent approach to secure localization in resource constrained wireless sensor networks. *IEEE Transactions on Information Forensics and Security*, 7(2):717–730, 2012.

[8] Fredrik Gustafsson, Fredrik Gunnarsson, and David Lindgren. Sensor models and localization algorithms for sensor networks based on received signal strength. *EURASIP Journal on Wireless Communications and Networking*, 2012(1):1–13, 2012.

[9] Guangjie Han, Jinfang Jiang, Lei Shu, Mohsen Guizani, and Shojiro Nishio. A two-step secure localization for wireless sensor networks. *Comput. J.*, 56(10):1154–1166, 2013.

[10] Guangjie Han, Huihui Xu, Trung Q. Duong, Jinfang Jiang, and Takahiro Hara. Localization algorithms of wireless sensor networks: a survey. *Telecommunication Systems*, 52(4):2419–2436, 2013.

[11] Sariel Har-Peled. Geometric Approximation Algorithms.

[12] Tian He, Sudha Krishnamurthy, John A. Stankovic, Tarek Abdelzaher, Liqian Luo, Radu Stoleru, Ting Yan, and Lin Gu. Energy-efficient surveillance system using wireless sensor networks. In *In Mobisys*, pages 270–283. ACM Press, 2004.

[13] J.T. Isaacs, D.J. Klein, and J.P. Hespanha. Optimal sensor placement for time difference of arrival localization. In *CDC/CCC 2009*, pages 7878–7884.

[14] Mohit Jain and Himanshu Kandwal. A survey on complex wormhole attack in wireless ad hoc networks. In *ACT '09*, pages 555–558.

[15] Jinfang Jiang, Guangjie Han, Chuan Zhu, Yuhui Dong, and Na Zhang. Secure localization in wireless sensor networks: A survey (invited paper). *JCM*, 6(6):460–470, 2011.

[16] D. Vinoth Kannan and S. Bala Murugan. Article: Energy efficient detection of replica node in mobile sensor networks. *IJCA Special Issue on International Conference on Electronics, Communication and Information systems*, ICECI(1):34–37, 2012.

[17] AmirE. Khandani, Eytan Modiano, Jinane Abounadi, and Lizhong Zheng. Cooperative routing in wireless networks. In BoleslawK. Szymanski and Blent Yener, editors, *Advances in Pervasive Computing and Networking*, pages 97–117. Springer US, 2005.

[18] Zang Li, W. Trappe, Y. Zhang, and B. Nath. Robust statistical methods for securing wireless localization in sensor networks. In *IPSN 2005*, pages 91–98, April 2005.

[19] Donggang Liu, Peng Ning, An Liu, Cliff Wang, and Wenliang Kevin Du. Attack-resistant location estimation in wireless sensor networks. *ACM Trans. Inf. Syst. Secur.*, 11(4):22:1–22:39, July 2008.

[20] James Newsome, Elaine Shi, Dawn Song, and Adrian Perrig. The sybil attack in sensor networks: Analysis & defenses. In *IPSN '04*, pages 259–268, New York, NY, USA, 2004. ACM.

[21] S.I. Roumeliotis and George A. Bekey. Distributed multirobot localization. *Robotics and Automation, IEEE Transactions on*, 18(5):781–795, Oct 2002.

[22] Peterj Rousseeuw and Katrien Driessen. Computing lts regression for large data sets. *Data Mining and Knowledge Discovery*, 12(1):29–45, 2006.

[23] V. Savic and S. Zazo. Reducing communication overhead for cooperative localization using nonparametric belief propagation. *Wireless Communications Letters, IEEE*, 1(4):308–311, August 2012.

[24] Andrea Simonetto, T. Keviczky, and R. Babuska. Distributed nonlinear estimation for robot localization using weighted consensus. In *ICRA Robotics and Automation, 2010*, pages 3026–3031, May 2010.

[25] Robert Szewczyk, Eric Osterweil, Joseph Polastre, Michael Hamilton, Alan Mainwaring, and Deborah Estrin. Habitat monitoring with sensor networks. *Commun. ACM*, 47(6):34–40, June 2004.

[26] V. Vovk and V. Vovk. A game of prediction with expert advice. *Journal of Computer and System Sciences*, 56:153–173, 1997.

[27] Kai Xing, Fang Liu, Xiuzhen Cheng, and David Hung-Chang Du. Real-time detection of clone attacks in wireless sensor networks. In *ICDCS*, pages 3–10, 2008.

[28] Enyang Xu, Zhi Ding, and S. Dasgupta. Source localization in wireless sensor networks from signal time-of-arrival measurements. *Signal Processing, IEEE Transactions on*, 59(6):2887–2897, June 2011.

[29] Sungwon Yang, Jiyoung Yi, and Hojung Cha. Hcrl: A hop-count-ratio based localization in wireless sensor networks. In *SECON '07*, pages 31–40.

[30] A. Youssef, A. Agrawala, and M. Younis. Accurate anchor-free node localization in wireless sensor networks. In *IPCCC 2005*, pages 465–470, April 2005.

[31] Liyang Yu, Neng Wang, and Xiaoqiao Meng. Real-time forest fire detection with wireless sensor networks. In *International Conference on Wireless Communications, Networking and Mobile Computing*, volume 2, pages 1214–1217, Sept 2005.

[32] Yingpei Zeng, Jiannong Cao, Jue Hong, Shigeng Zhang, and Li Xie. Secure localization and location verification in wireless sensor networks: a survey. *The Journal of Supercomputing*, 64(3):685–701, 2013.