On the Impact of Localization Data in Wireless Sensor Networks with Malicious Nodes

Mattia Monga
Dip. di Informatica e Comunicazione
Università degli Studi di Milano
Via Comelico 39, I–20135 Milano, Italy
mattia.monga@unimi.it

Sabrina Sicari
Dip. di Informatica e Comunicazione
Università degli Studi dell'Insubria
Via Mazzini 5, I–21100 Varese, Italy
sabrina.sicari@uninsubria.it

ABSTRACT

The knowledge of node positions is a core concept in any Wireless Sensor Network context. Several localization algorithms were devised, but secure localization of sensor nodes is still a challenging task to achieve with a high level of performance. In fact, location information might be the target of different kinds of malicious attacks and several secure localization approaches were proposed. In this paper we analyze the impact of false data in a secure localization algorithm, known as *Verifiable Multilateration*. We found that the strategy used to compute the positions of nodes might have an impact both on the computational effort needed to achieve acceptable measures and the precision of the detection of malicious nodes.

Categories and Subject Descriptors

K.6.5 [**MANAGEMENT OF COMPUTING AND INFORMATION SYSTEMS**]: Security and Protection

General Terms

Wireless Sensor Networks, security

Keywords

WSN, security, localization

1. INTRODUCTION

Several researchers are proposing information systems based Wireless Sensor Networks (WSNs), that provide a flexible and effective means to monitor large and diverse geographical areas. However, WSNs are composed by individual nodes with very limited capabilities and energy consumption is a major concern, thus unorthodox solutions are required for many situations, especially aimed at minimizing the communication overload. Moreover, the monitoring activity greatly relies on data about the positions of nodes, which are often deployed randomly, thus a great challenge is represented by localization at time of operations [12].

Various location services have been proposed. The Global Positioning System (GPS) is the most well-known location service in use today, but it is unsuitable for low-cost, ad-hoc sensor networks since GPS is based on an extensive infrastructure (i.e. satellites) that requires frequent transmissions and devices are still quite expensive and heavy. Likewise the solutions developed in the area of robotics [1, 13, 24] and ubiquitous computing [10] are generally not applicable for sensor networks as they require too much processing power and energy. Recently a number of localization systems have been proposed specifically for sensor networks [3, 4, 7, 9, 23, 16, 19, 22]. Ideally, these approaches aim at large-scale ad-hoc sensor networks (100+ nodes) and their design goals are:

- to be as much as possible self-organizing, thus that communication happens mostly locally, without the need of a globally accessible infrastructure;
- to be tolerant to node failures and range errors;
- to require little computation and, especially, communication effort.

Unfortunately, most of the current approaches omit to consider that WSNs could be deployed in an adversarial setting, where hostile nodes under the control of an attacker coexist with faithful ones. In fact, from a security point of view, the wireless communications and the deployment in uncontrolled environments rise several issues: the confidentiality, the integrity, and the availability of data might be put at risk by malicious tampering of sensors and/or traffic.

Node position is a really critical information due to the strict relation with the quality of the provided services. In fact, the location information is sometimes target of different kinds of malicious attacks, classified in internal and external attacks. So the trustworthiness of node position information is a challenging task for wireless sensor networks since classical solutions based on access control and strong authentication, are problematic to implement with limited resources and short battery life. Also, nodes are prone to physical attacks and is pretty easy to clone a sensor device and its on-board keys: thus cryptography provides only a partial protection and should be used with care.

In this paper we analyzed an approach to the secure localization of nodes known as *Verifiable Multilateration* (VM) [5]. VM potentially uses untrusted information to derive the positions of nodes, together with a measure of their trustworthiness. VM relies on *lateration* to compute positions, a

ACM SPRINGL'09 November 3, 2009. Seattle, WA, USA

generalization of triangulation to multiple nodes: several techniques to do lateration are known and VM is largely independent from the choice of one of them. We found, however, that the choice is not neutral. Lateration algorithms can be computationally heavy in order to get very precise results or very trivial if only gross data are needed. One could legitimately ask if and when the additional efforts are needed and at what level, since computation in WSNs is a scarce resource. In this paper we try to answer to these questions by analyzing three lateration algorithms and assessing their impact on VM. We will show that in general the precision is worth the computational price, but, under some hypotheses, even a trivial solution can be acceptable.

The paper is organized as follows: Section 2 provides a short state of art about the sensor node localization solutions; Section 3 describes the reference scenario in which we performed our analysis; Section 4 introduces the Verifiable Multilateration for secure localization; Section 5 analyzes three different approaches to lateration and their impact on secure localization and finally, Section 6 draws some conclusions and provides hints for future works.

2. RELATED WORK

All the proposed localization algorithms for wireless sensor networks have to face the particular context in which sensor nodes are deployed. More specifically, there is in general no fine control over the placement of the sensor nodes when the network is deployed (e.g., when nodes are dropped from an airplane) and some self organization of the communication overlay is needed. Moreover, the connectivity of the nodes in the network (i.e., the average number of neighbors) is another important parameter that has a strong impact on the accuracy of most localization algorithms. In fact, the main node position approach is based on node density and radio range, and in some cases it can be dynamically adjusted by changing the transmit power of the RF radio. So taking into account the domain constraints, any localization algorithm has to address three main requirements: self organizing, robustness and energy efficiency.

Existing localization schemes may be classified in *range-based methods*, which use exact measurements of distances, and *range-free methods*, which only need beacon signals. Typical techniques to obtain the measurements between two nodes include Received Signal Strength Indicator (RSSI), Time of Arrive (ToA), Time Difference of Arrive (TDoA), and Angle of Arrive (AoA). Range-based localization schemes in sensor networks include those in [21, 22, 17, 15, 9]. Savvides et al. developed an ad-hoc localization system localization protocol based on TDoA [21]. Extension of this work can be found in [22]. Doherty et al. presented a localization scheme based on connectivity induced constraints and the relative angle between neighbors [9]. AoA is also used to develop localization schemes in [17] and [15]. Range-free based schemes are proposed to provide location estimation services for those applications with less required location precision [3, 14, 23]. Estrin et al. proposed a simple range-free, coarse grained localization scheme where each sensor estimated its location by centering the locations contained in the received signals [3]. All of the current localization schemes become vulnerable when there are malicious attacks. In all these schemes, the accuracy of location estimation depends on the accuracy of the origins of the beacon signals and certain measurements obtained from the beacon signals, including distances and/or angles in range-based schemes, and the existence of beacon signals in range-free schemes. Though the above measurements are directly obtained from the physical signals, the locations of the beacon signals' origins can be easily forged. As a result, a malicious attacker may introduce large errors when a node estimates its location. More specifically, an attacker can introduce arbitrarily large errors by declaring false locations in beacon packets, arbitrarily introducing large errors into a non-beacon node's location estimation. Such attacks cannot be simply prevented by cryptographic techniques due to the threat of compromised nodes and replay attacks. In order to overcome such a limit localization algorithms adopt some techniques able to reveal malicious behavior.

We focus our attention on the secure localization algorithm, named Verifiable Multilateration (VM) [5] that is a range based approach using MMSE as one criterium for revealing malicious behavior. Our choice of MMSE is due to the robustness towards attacks and the capability to reveal malicious behavior and then the accuracy of the obtained results as we show in details in the following sections. We aim at analyzing in some depth the computational effort of minimization, by comparing MMSE with other available solutions. Since a weak assessment of localization information may damage service performance, our goal is to understand the trade-off between the computational cost and the overall trustworthiness of the obtained results.

3. REFERENCE SCENARIO

We consider a dense network composed of nodes n_i, where $n_i \in N, 0 < i \leq |N|$ and a base station b in which all the collected data sink. We consider two subsets of N:

- S, composed by nodes $s_i, 0 < i \leq |S|$, which perform sensing functions;
- V, composed by nodes $v_i, 0 < i \leq |V|$, which work as verifiers in the secure localization protocol.

$N = S \cup V$ and V may overlap S (in principle every node whose position can be taken for granted might be used as a verifier).

Each s_i node senses a given type of data (*e.g.,* temperature, pressure, brightness, position and so on). Each node (sensing, and verifier) directly communicates with its closer neighbours (at one hop distance).

4. SECURE LOCALIZATION

The node positions can be evaluated by using a multilateration technique, which determines the node coordinates by exploiting a set of landmark nodes, called *anchor* nodes, whose positions are known. The position of the unknown node u is computed by using an estimation of the distances between the anchor nodes and the node itself. The distance is not measured directly; instead, it can be computed by knowing the speed of the signal in the medium used in the transmission, and by measuring the time needed to get an answer to a beacon message sent to u. If the computation is carried on without any precaution, u might fool the anchors by delaying the beacon message. However, since a malicious node can delay the answer beacon, but not speed it up, under some conditions it is possible to spot malicious behaviors. *Verifiable Multilateration* (VM) [5] uses three or

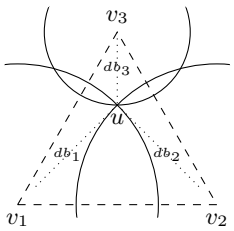


Figure 1: Verifiable multilateration

more anchor nodes to detect misbehaving nodes. In VM the anchor nodes work as *verifiers* of the localization data and they send to the sink b the information needed to evaluate the consistency of the coordinates computed for u. The basic idea of VM is shown in Figure 1: each verifier v_i computes its *distance bound* [2] to u; any point $u' \neq u$ inside the triangle formed by v_1, v_2, v_3 has necessarily at least one of the distance to the v_i enlarged. This enlargement, however, cannot be masked by u by sending a faster message to the corresponding verifier. Therefore, if the verifiers are trusted and they can securely communicate with b, the following algorithm can be used to check the localization data:

1. Each verifier v_i sends a beacon message to u and records the time τ_i needed to get an answer;
2. Each verifier v_i (whose coordinates $< x_i, y_i >$ are known) sends to b a message with its τ_i;
3. From τ_i, b derives the corresponding distance bound db_i (that can be easily computed if the speed of the signal is known) and it estimates u's coordinates by minimizing the mean square error

   $$\epsilon = \sum_i (db_i - \sqrt{(x_u - x_i)^2 + (y_u - y_i)^2})^2$$

   where $< x_u, y_u >$ are the (unknown) coordinates to be estimated[1].;
4. b can now check if $< x_u, y_u >$ are feasible in the given setting by two incremental tests:
   (a) *δ-test:* For all verifiers v_i, compute the distance between the estimated u and v_i: if it differs from the measured distance bound by more than the expected distance measurement error, the estimation is affected by malicious tampering;
   (b) *Point in the triangle test:* Distance bounds are reliable only if the estimated u is within at least one verification triangle formed by a triplet of verifiers, otherwise the estimation is considered unverified.

If both the δ and the point-in-the-triangle tests are positive, the distance bounds are consistent with the estimated node position, which moreover falls in at least one verification triangle. Thus, the sink can consider the estimated position of the node as `Robust`; else, the information at hands is not sufficient to support the reliability of the data. An estimation that does not pass the δ test is considered `Malicious`. A sensible value of δ depends on the expected error in time measurement and the number of available verifiers. The simulation reported below should clarify the considerations involved in the choice of δ. If the δ test is passed, but the point-in-the-triangle one fails, the sink marks the estimation as `Unknown`, meaning there is no sufficient information for evaluating the trustworthiness of node position. Thus, the localization phase ends up, for each unlocalized node u_i, with an estimation of the position of u_i and a quality $W_i \in Robust, Unknown, Malicious$.

[1]In an ideal situation where there are no measurement errors and/or malicious delays this is equivalent to finding the (unique) intersection of the circles defined by the distance bounds and centered in the v_i (see Figure 1) and $\epsilon = 0$

5. THE IMPACT OF LOCALIZATION INFORMATION

Summing up, VM aims at assessing the trustworthiness of a node position by checking the consistency of the data received by the sink:

- the δ test establishes a threshold incompatible with highly deceptive data;
- the point-in-the-triangle test rules out geometrically infeasible deceptions.

As stated above, the original VM approach requires (step 3) the minimization of the mean square error ϵ. This function, however, is not linear and minimization is far from trivial. In fact, no exact solution is possible and some approximation is needed. In our experiments, we found that most of the computational effort of the approach was in the minimization. Thus, we considered three alternatives:

1. use a probabilistic heuristic to approximate the search for the minimum ϵ (MMSE approach)
2. use a function easier to deal with (exact lateration approach)
3. use a trivial estimate of the position (min-max approach)

In order to analyze the feasibility of these simplifications in an adversarial context, we used OMNeT++ (ver. 3.3p1, [8, 18]) to set up a simulation of the secure localization algorithm. A claimant node u to be localized resides at the center of a 100m×100m field, *i.e.,* at point $< 50, 50 >$. Since the best approach to lay out three verifiers is on the vertexes of an equilateral triangle [5], we fixed their coordinates to be the points $< 1, 1 >, < 99, 1 >, < 50, 85 >$. If u is *faithful,* it answers to verifiers' beacons without any delay. Otherwise, if u is *malicious* it adds a variable delay to the answers, in order to dissimulate a fake position u': *i.e.,* for each v_i, if the distance $v_i\bar{}u'$ is greater than $v_i\bar{}u$ a proper delay is added by u to the answer beacon to v_i. We assumed that signals travel at the speed of light and that time can be measured with an error whose standard deviation is 2ns. As described above, the timing information collected by verifiers v_i can be used by the base station to classify the claimant as `Malicious`, `Unknown`, or `Robust`.

In a preliminary study, we discovered that the error introduced by the localization heuristic is indeed critical, since it could cause an unexpected behavior in the algorithm. Figure 2 shows a number of anchors v_i and the distance bounds they estimate in color. The actual position of the malicious

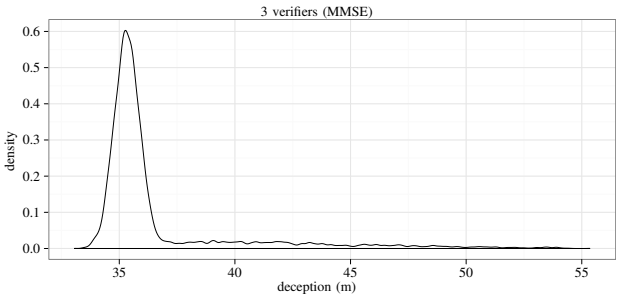


Figure 4: Deception when a node is classified as `Unknown` ($\delta = 35$)

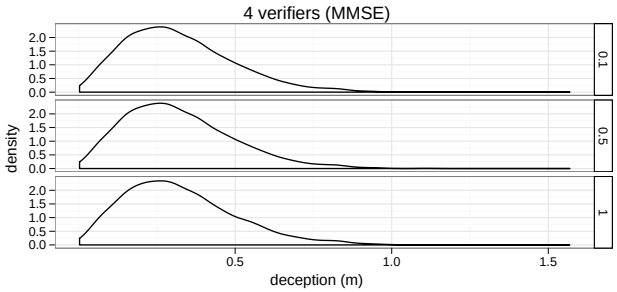


Figure 5: Deception when a malicious node is classified as `Robust`

claimant node is u and the estimated position u'. In Figure 2(a), with three verifiers, the node results as `Unknown` since it is outside the unique verification triangle. However, when a fourth verifier is added to the system (see Figure 2(b)) the estimation u' falls inside at least one of the four verification triangles and the node ends up to be `Robust`.

Thus, we decide to analyze the sensitivity of VM to the localization heuristic used. In particular, we considered as a quality metric the *deception* that can be induced by an attacker, *i.e.,* the distance between the actual node position and the estimated one, when node are classified as `Robust` or `Unknown`.

5.1 The MMSE approach

The original proposal of VM, relies on the minimization of the mean square error (MMSE). In our simulation, we used *simulated annealing* [11, 6] heuristic to approximate a solution.

Figure 3(a) shows the effect of the choice of the δ_{max} in the δ-test on 10000 runs with 3 verifiers: the only sensible value is about 35, since lower levels have an overwhelming rate of false positives (*i.e.,* faithful nodes classified as `Malicious`), and a higher δ gives too much false negatives (*i.e.,* malicious nodes classified as `Robust`) and unknowns. About 50% of malicious claimants and 90% of faithful ones were classified as `Unknown`: the error in taking the estimated position instead of the real one is pretty high, as one can see from Figure 4 that plots the density of deception: most of the time by accepting an estimation classified as `Unknown` one has to deal with a deception of about 35 m. The situation is clearly improved when a fourth verifier is added (see Figure 3(b)): the setting is now with a verifier at each corner of the field and all the values less than 2.5 give acceptable results; there are no `Unknowns`. It is worth noting that the range of δ considered is different, since by increasing the number of verifiers, the maximum acceptable error δ_{max} should decrease. There are still some false negatives, but the deception induced by a malicious node taken as `Robust` is always less than 1m with $\delta \leq 1$. Figure 5 plots the density distribution of the deception — *i.e.,* the distance between the real position and the estimated one — at different values of δ. Adding a fifth verifier randomly deployed significantly decreases the rate of false negatives, as shown in Figure 3(c).

5.2 The exact lateration approach

An easier estimation to compute is *exact lateration* (used for example by [16, 20]) that considers the system of equations

$$\forall i, 1 \leq i \leq |V| : \quad (x_i - x_u)^2 + (y_i - y_u)^2 = db_i \qquad (1)$$

The system (1) can be linearized by subtracting the last equation (the one corresponding to verifier $|V|$) from the other $|V| - 1$ ones.

$$\forall i, i \neq |V| :$$
$$x_i^2 - x_{|V|}^2 - 2(x_i - x_{|V|})x_u + y_i^2 - y_{|V|}^2 - 2(y_i - y_{|V|})y_u = db_i^2 - d_{|V|}^2$$

The system above can be expressed in the matrix form

$$A_{(|V|-1,2)}x_{(2,1)} = b_{(|V|-1,1)}$$

where

$$A_{(|V|-1,2)} = [2(x_i - x_{|V|}) \quad 2(y_i - y_{|V|})]$$
$$b_{(|V|-1,1)} = [x_i^2 - x_{|V|}^2 + y_i^2 - y_{|V|}^2 + db_{|V|}^2 - d_i^2]$$

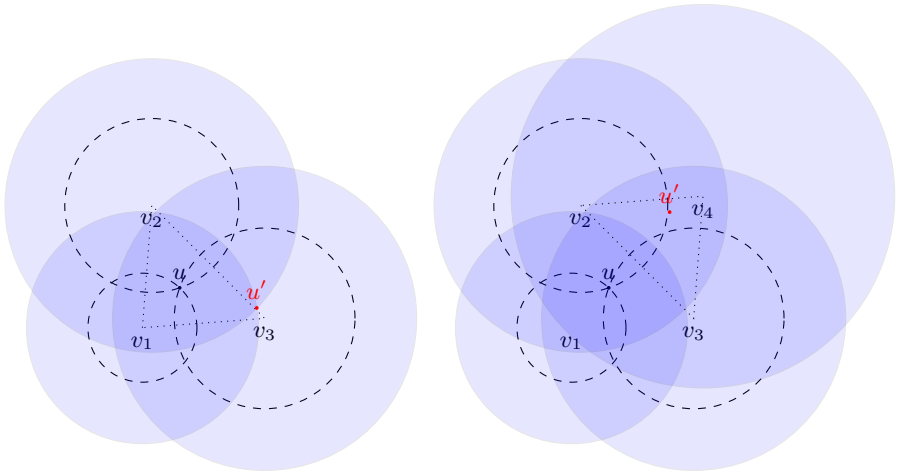
The system can be solved by using a standard linear algebra least-squares approach: $x = (A^T \cdot A)^{-1} \cdot A^T \cdot b$. A measure of the quality of the solution is then given by

$$r_{lat} = \frac{\sum_i (db_i - \sqrt{(x_u - x_i)^2 + (y_u - y_i)^2})^2}{|V|}$$

In order to evaluate this approach to localization with respect to the MMSE one described in Section 5.1, we considered the relation between the residue r_{lat} (analogous to ϵ in the MMSE case) and the deception induced by assuming $< x_u, y_u >$ as the position. Figure 6 shows the correlation between the quality of the estimation and deception for both MMSE (Figure 6(a)) and exact lateration (Figure 6(b)): the latter is more spread, thus indicating that MMSE ϵ is a better proxy indicator for deception. In fact, deception by a malicious claimant evaluated by an exact lateration approach gives results fairly uncorrelated with the ones obtained with MMSE (see Figure 7).

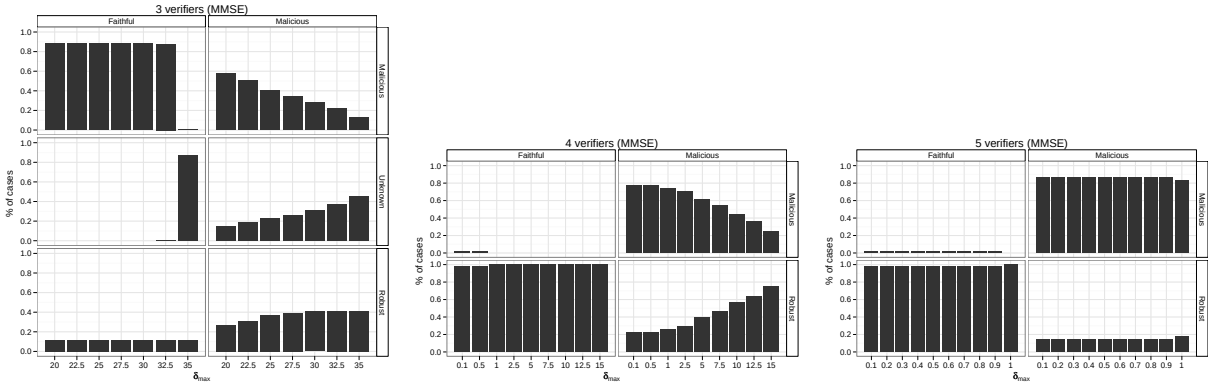
5.3 The min-max approach

Sometimes an even easier estimation used is the *min-max* method ([21], the name is coined in [12]). Its computation is almost trivial: for each verifier one considers the bounding box defined by $< x_i - db_i, y_i - db_i > -< x_i + db_i, y_i +$



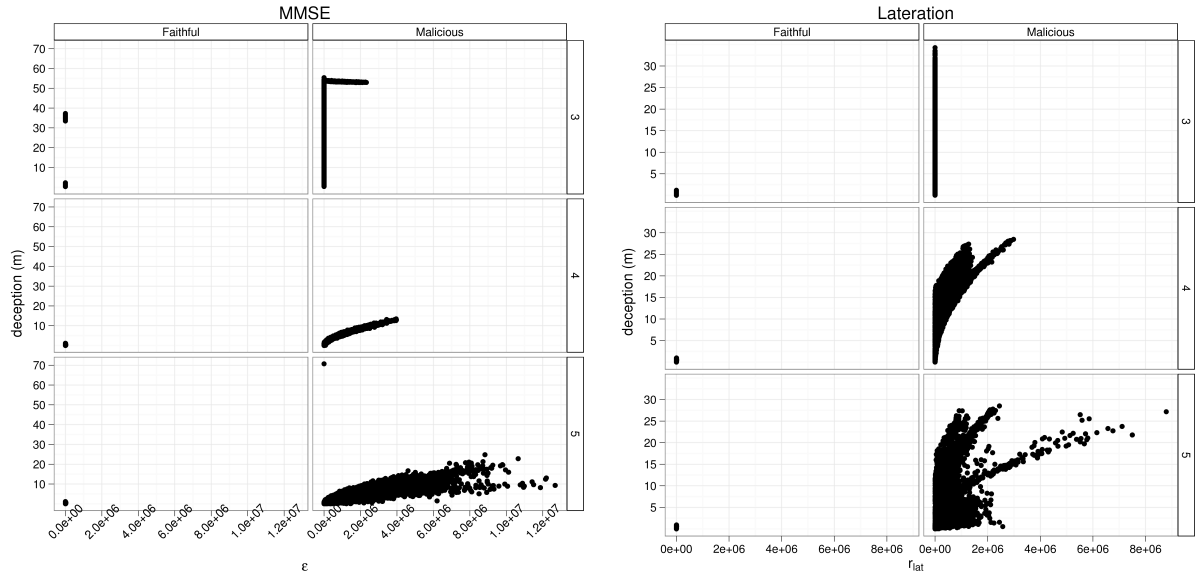
(a) (b)

Figure 2: The quality of the estimated position u' depends on the number of verifiers



(a) (b) (c)

Figure 3: Classification by secure localization



(a) (b)

Figure 6: Correlation between estimation quality and deception with 3,4, and 5 verifiers

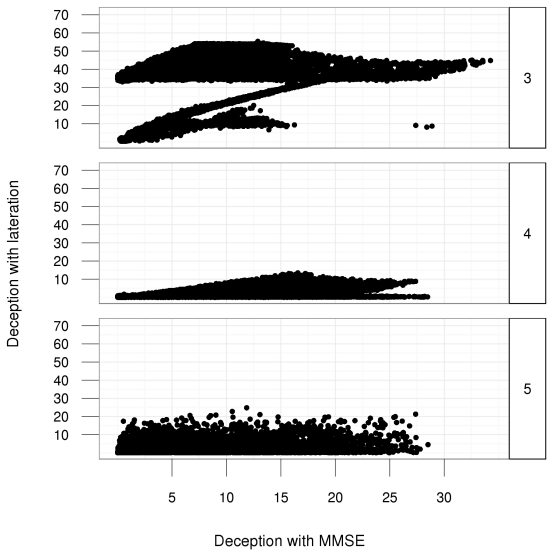


Figure 7: Deception by a malicious node evaluated by exact lateration with 3,4, and 5 verifiers

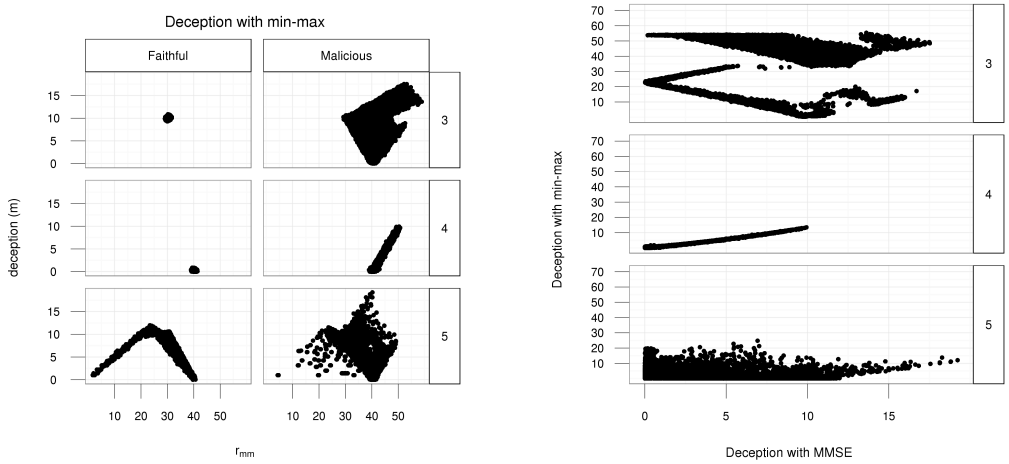
$db_i >$. The intersection of all bounding boxes is then computed as $< \max(x_i - db_i), \max(y_i - db_i) > - < \min(x_i + db_i), \min(y_i + db_i) >$ and the final position estimated is $\frac{<\max(x_i - db_i) + <\min(x_i + db_i)}{2}, \frac{<\max(y_i - db_i) + <\min(y_i + db_i)}{2} >$. We measured the quality of the estimation as

$$r_{mm} = |\frac{\max(x_i - db_i) - \min(x_i + db_i) + \max(y_i - db_i) - \min(y_i + db_i)}{2}|$$

Again, we found that MMSE ϵ is a much better proxy for deception in an adversarial setting (see Figure 8(a)). However, with four verifiers (posed on the vertexes of the rectangular field) the results would be consistent with the ones obtained via MMSE, but with a considerable saving in computation (see Figure 8(b)). However, this result is not confirmed in the 5-verifiers case: in fact, the fifth verifier — randomly deployed — destroys the symmetry of bounding boxes, and it has an unexpected detrimental effect. The setting with four verifiers, instead, could be a good alternative to the MMSE corresponding solution since it can give proportionally equivalent result with a much reduced computational effort.

6. CONCLUSIONS

Reliability of node positions is a core requirement in most Wireless Sensor Networks. Verifiable Multilateration uses potentially untrusted information to derive the positions of nodes, together with a measure of their trustworthiness. However, VM itself relies on node positions deduced by lateration. We analyzed different approaches to lateration in order to understand when the computational effort needed by the most sophisticated algorithms is really needed. Our results show that in general the precision provided by the most onerous algorithm (MMSE) is indeed needed. However, if a careful position of verifiers is possible, the much simpler min-max method could be useful. The aim of secure



(a) Correlation between estimation quality and deception

(b) Deception with a malicious node via MMSE and min-max

Figure 8: Estimation quality and deception in the min-max approach with 3,4, and 5 verifiers

localization algorithms is to define some criteria in order to identify and remove malicious positions. False positive may always occur, however, and one has to spend in verifiers and communication to increase the quality of the collected information. We are currently investigating the use of cross-layer information to assess the overall quality of the monitoring performed by the WSN and a game theoretical approach to model malicious behavior, in order to reason about the rational strategies open to the system designers.

Acknowledgment

This research has been partially funded by the European Commission, Programme IDEAS-ERC, Project 227977-SMScom.

7. REFERENCES

[1] S. Atiya and G. Hager. Real-time vision-based robot localization. *IEEE Trans. on Robotics and Automation*, 9(6):785–800, 1993.

[2] S. Brands and D. Chaum. Distance-bounding protocols. In *Proc. of Workshop on the Theory and Application of Cryptographic Techniques*, 1994.

[3] N. Bulusu, J. Heidemann, and D. Estrin. Gps-less low-cost outdoor localization for very small devices. *IEEE Personal Communications*, 7(5):28–34, Oct. 2000.

[4] S. Čapkun, M. Hamdi, and J.-P. Hubaux. Gps-free positioning in mobile ad-hoc networks. *Cluster Computing*, 5(2):157–167, April 2002.

[5] S. Čapkun and J. Hubaux. Secure positioning in wireless networks. *IEEE Journal On Selected Areas In Communications*, 24(2):221–232, Feb. 2006.

[6] V. Cerny. A thermodynamical approach to the travelling salesman problem: an efficient simulation algorithm. *Journal of Optimization Theory and Applications*, 45:41–51, 1985.

[7] J. Chen, K. Yao, and R. Hudson. Source localization and beamforming. *IEEE Signal Processing Magazine*, 19(2):30–39, 2002.

[8] O. Community. `http://www.omnetpp.org/`.

[9] L. Doherty, K. Pister, and L. E. Ghaoui. Convex position estimation in wireless sensor networks. In *Proc. of IEEE Infocom 2001*, 2001.

[10] J. Hightower and G. Borriello. Location systems for ubiquitous computing. *IEEE Computer*, 34(8):57–66, 2001.

[11] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *Science, New Series*, 220(4598):671–680, May 1983.

[12] K. Langendoen and N. Reijers. Distributed localization in wireless sensor networks: a quantitative comparison. *Elsevier Computer Networks*, 43:499–518, 2003.

[13] J. Leonard and H. Durrant-Whyte. Mobile robot localization by tracking geometric beacons. *IEEE Trans. on Robotics and Automation*, 7(3):376–382, 1991.

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

[15] A. Nasipuri and K. Li. A directionality based location discovery scheme for wireless sensor networks. In *Proc. of ACM WSNA'02*, 2002.

[16] D. Niculescu and B. Nath. Ad-hoc positioning system. In *Proc. of GlobeCom*, 2001.

[17] D. Niculescu and B. Nath. Ad hoc positioning system (aps) using aoa. In *Proc. of IEEE INFOCOM 2003*, 2003.

[18] G. Pongor. OMNeT: objective modular network testbed. In *MASCOTS'93 Proceedings of the International Workshop on Modeling, Analysis, and Simulation On Computer and Telecommunication Systems*, pages 323–326, San Diego, CA, USA, 1993. The Society for Computer Simulation, International.

[19] V. Ramadurai and M. Sichitiu. Localization in wireless sensor networks: A probabilistic approach. In *Proc. of Int. Conf. on Wireless Networks (ICWN)*, 2003.

[20] C. Savarese, K. Langendoen, and J. Rabaey. Robust positioning algorithms for distributed ad-hoc wireless sensor networks. In *Proc. of USENIX technical annual conference*, 2002.

[21] A. Savvides, C. Han, and M. Srivastava. Dynamic fine-grained localization in ad-hoc networks of sensors. In *Proc. of ACM Mobicom'01*, 2001.

[22] A. Savvides, H. Park, and M. Srivastava. The bits and flops of the n-hop multilateration primitive for node localization problems. In *Proc. of First ACM Int. Workshop on Wireless Sensor Networks and Application (WSNA)*, 2002.

[23] T.He, C. Huang, B. M. Blum, J. A. Stankovic, and T. Abdelzaher. Range-free localization schemes for large scale sensor networks. In *Proc. of ACM International Conference on Mobile Computing and Networking (Mobicom)*, 2003.

[24] R. Tinos, L. Navarro-Serment, and C. Paredis. Fault tolerant localization for teams of distributed robots. In *Proc. of IEEE Int. Conf. on Intelligent Robots and Systems*, 2001.