A Localization Game in Wireless Sensor Networks

Nicola Gatti[1], Mattia Monga[2], and Sabrina Sicari[3]

[1] Politecnico di Milano
Dip. di Elettronica e Informazione
Piazza Leonardo da Vinci, 32
I–20133 Milan, Italy
ngatti@elet.polimi.it
[2] Università degli Studi di Milano
Dip. di Informatica e Comunicazione
Via Comelico, 39
I–20135 Milan, Italy
mattia.monga@unimi.it
[3] Università degli Studi dell'Insubria
Dip. di Informatica e Comunicazione
Via Mazzini, 5
I–21100 Varese, Italy
sabrina.sicari@uninsubria.it

Abstract. Wireless Sensor Networks (WSNs) support data collection and distributed data processing by means of very small sensing devices that are easy to tamper and clone: therefore classical security solutions based on access control and strong authentication are difficult to deploy. In this paper we look at the problem of assessing the reliability of node localization data from a game theoretical viewpoint. In particular, we analyze the scenario in which Verifiable Multilateration (VM) is used to localize nodes and a malicious node (*i.e.,* the adversary) try to masquerade as non-malicious. We resort to non-cooperative game theory and we model this scenario as a two-player game. Thus, we were able to compute an upper bound to the error that an attacker can induce in localization data, given the number of available verifiers. We focused on the *maximum deception,* that is the distance between the inferred position of an unknown node and the actual one: we found that if the verifiers are placed opportunely, the deception is at most 25% of the power range, and can be halved by triplicating the number of the verifiers.

1 Introduction

Wireless sensor networks (WSNs) [1, 2] become increasingly popular in many application domains: indoor/outdoor surveillance systems, traffic monitoring and control systems for urban and sub-urban areas, systems supporting telemedicine, attendance to disable or elderly people, environment monitoring, localization and recognition of services and users, monitoring and control of manufacturing processes in industry, etc. Most of these activities greatly rely on data

about the positions of sensor nodes, which is not necessarily known before hand. In fact, nodes are often deployed randomly or they even move, and one of the challenges is computing localization at time of operations. Several localization approaches have been proposed (for example, [4, 5, 7, 8, 15, 11, 13, 14]), but 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, wireless communications are easy to tamper and nodes are prone to physical attacks and cloning: thus classical solutions, based on access control and strong authentication, are difficult to deploy.

A well defined approach to localize nodes even when some of them are compromised was proposed in [6] by Čapkun *et al.* and it is known as *Verifiable Multilateration* (VM). VM computes an unknown location by leveraging on a set of trusted landmark nodes, named *verifiers*. Although VM is able to recognize reliable localization measures (known as *robust* computations) and sure malicious behaviors, it allows for undecided positions (*unknown nodes*), *i.e.,* cases in which localization data do not provide enough information to support a certain marking as robust or malicious. A conservative approach could be to discard every undecided measure, but this could be unfeasible in some scenarios. This weakness could be exploited by a malicious node to masquerade as an unknown one, pretending to be in a position that is still compatible with all verifiers' information. To the best of our knowledge, the analysis of this scenario, in terms of how a malicious, on the one side, could act and, on the other side, could be faced, has not been explored so far in the literature. This constitutes the original contribution of our work.

To study the properties of a system based on VM deployed in an adversarial setting, we resort to non-cooperative game theory. More precisely, we model it as a two-player game, where the first player employs a number of *verifiers* to do VM computations and the second player is a *malicious node*. The verifiers act to securely localize the malicious node, while the malicious node acts to masquerade as unknown, since when it is recognized as malicious its influence on the system is ruled out by VM. As is customary in game theory, the players are considered rational (*i.e.,* maximizers). This amounts to say that the malicious node is modeled as the strongest adversary. Thanks to game theory model the potentialities of VM are analyzed in depth showing interesting information that should improve the defender's strategy. In [9] we studied the game wherein the verifiers and the malicious node act simultaneously characterizing the players' equilibrium strategies. In this paper, we model the game in extensive form assuming that the malicious node, at first, observes the verifiers' actions and, then, takes its action. This model captures more satisfactorily real world situations. We show how the verifiers should be placed to put a bound on the error the attacker might induce if the defender accepted also unknown positions. Moreover, the study of VM by a game theoretical approach improved our insight in VM properties giving us a tool to quantify the overall robustness of the localization protocol.

The paper is organized as follows: Section 2 provides a short overview about Verifiable Multilateration; Section 3 shortly describes secure localization game, providing some basic concepts; Section 4 analyzes the game in its extensive form and discusses the impact of multiple verifiers in an ad-hoc topology. Section 5 draws some conclusions and provides hints for future works.

2 Verifiable Multilateration

Multilateration is a technique used in WSNs to estimate the coordinates of the unknown nodes, given the positions of some given landmark nodes, sometimes called anchor nodes, whose positions are known. The position of the unknown node U is computed by geometric inference based on the distances between the anchor nodes and the node itself. However, the distance is not measured directly; instead, it is derived 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.

Unfortunately, if this computation is carried on without any precaution, U might fool the anchors by delaying the beacon message. However, since in most settings a malicious node can delay the answer beacon, but not speed it up, under some conditions it is possible to spot malicious behaviors. VM uses three or 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 node 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* [3] to U; any point $P \neq U$ inside the triangle formed by $V_1V_2V_3$ 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.

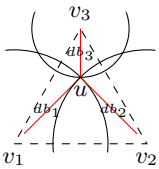


Fig. 1. Verifiable multilateration.

Under the hypothesis that verifiers are trusted and they can securely communicate with B, the following verification process can be used to check the localization data in a setting in which signals cannot be accelerated:

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 $\langle x_i, y_i \rangle$ 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 sum of squared errors

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

where $\langle x, y \rangle$ are the (unknown) coordinates to be estimated[1];
4. B can now check if $\langle x, y \rangle$ 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. This means that none of the distance bounds were enlarged. 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. In all the other cases, the sink marks the estimation as UNKNOWN. In an ideal situation where there are no measurement errors, there are neither malevolent nodes marked as ROBUST, nor benevolent ones marked as MALICIOUS. Even in this ideal setting, however, there are UNKNOWN nodes, that could be malevolent or not. In other words there are no sufficient information for evaluating the trustworthiness of node position. In fact, U could pretend, by an opportune manipulation of delays, to be in a position P that is credible enough to be taken into account. No such points exist inside the triangles formed by the verifiers (this is exactly the idea behind verifiable multilateration), but outside them some regions are still compatible with all the information verifiers have.

Consider N verifiers that are able to send signals in a range R. Let x_0 and y_0 the *real* coordinates of U. They are unknown to the verifiers, but nevertheless they put a constraint on plausible fake positions, since the forged distance bound to V_i must be greater than the length of $\overline{UV_i}$.

Thus, any point $P = \langle x, y \rangle$ that is a plausible falsification of U has to agree to the following constraints, for each $1 \leq i \leq N$:

[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 $\epsilon = 0$. In general the above computation in presence of errors is not trivial: this has several consequences on the trust model; see [10].

$$\begin{cases} (y-y_i)^2 + (x-x_i)^2 < R^2 \\ (y-y_i)^2 + (x-x_i)^2 > (y_0-y_i)^2 + (x_0-x_i)^2 \end{cases} \quad (1)$$

The constraints in (1) can be understood better by looking at Figure 2, where three verifiers are depicted: the green area around each verifier denotes its power range, and the red area is the bound on the distance that U can put forward credibly. Thus, any plausible P must lay outside every red region and inside every green one (and, of course, outside every triangle of verifiers).

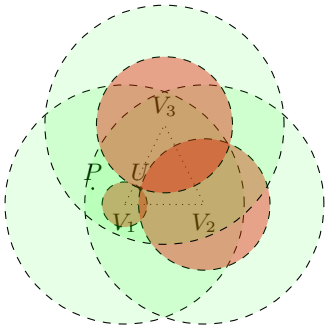


Fig. 2. Plausible falsification region: P is a plausible fake position for U since lays outside every red region and inside every green one whose radius is R (moreover it is outside the triangle of verifiers).

3 Secure Localization Game

Our aim is the study of the behavior of a possible malicious node that acts to masquerade as an unknown node and, at the same time, how the malicious node can be faced at best by the verifiers. This is a typical non-cooperative setting that can be analyzed by leveraging on game theoretical models. A *game* is described by a couple: *mechanism* and *strategies*. The mechanism defines the rules of the game in terms of number of players and actions available to the players. When the mechanism prescribes that the players act simultaneously (*e.g.,* rock-paper-scissors game), the game is said to be in *strategic form*. Instead, when the mechanism prescribes that the players act sequentially (*e.g.,* chess) the game is said to be in *extensive form*. The strategies describe the behaviors of the players during the game in terms of played actions. Strategies can be pure, when a player acts one action with a probability of one, or they can be mixed, when a player randomizes over a set of actions. The players' strategies define an outcome (if the strategies are pure) or a randomization over the outcomes (if mixed). Players have preferences over the outcomes expressed by utility functions and each player is *rational*, acting to maximize its own utility. Solving a game

means to find a profile of strategies (*i.e.,* a set specifying one strategy for each player) such that the players' strategies are somehow in equilibrium. The most known equilibrium concept is *Nash* where each player cannot improve its own utility by deviating unilaterally (a detailed treatment of Nash equilibrium can be found in [12]): a fundamental result in the study of equilibria is that every game admits at least one Nash equilibrium in mixed strategies, while pure strategy equilibrium might not exist.

We now formally state our secure localization game as a two-step extensive-form game where the first player to act is the defender (*i.e.,* the verifiers) and then the attacker (*i.e.,* the malicious node) acts. This capture real-world settings where, usually, at first the verifiers are placed and subsequently nodes whose position has to be determined appear. The game is a tuple $\langle Q, A, u \rangle$. Set Q contains the players and is defined as $Q = \{\mathbf{v}, \mathbf{m}\}$ ($\mathbf{v}$ denotes the verifiers and $\mathbf{m}$ denotes the malicious node). Set A contains the players actions. More precisely, given a surface $S \subseteq \mathbb{R}^2$, the actions available to $\mathbf{v}$ are all the possible tuples of positions $\langle V_1, \ldots, V_n \rangle$ of the n verifiers with $V_1, \ldots, V_n \in S$, while the actions available to $\mathbf{m}$ are all the possible couples of positions $\langle U, P \rangle$ with $U, P \in S$ (where U and P are the same as defined in Section 2). We denote by $\sigma_{\mathbf{v}}$ the strategy of $\mathbf{v}$ and by $\sigma_{\mathbf{m}}$ the strategy of $\mathbf{m}$. Given a strategy profile $\sigma = (\sigma_{\mathbf{v}}, \sigma_{\mathbf{m}})$ in pure strategy, it is possible to check whether or not constraints (1) are satisfied. The outcomes of the game can be $\{\text{MALICIOUS}, \text{ROBUST}, \text{UNKNOWN}\}$; we denote respectively with $\sigma_M, \sigma_R, \sigma_U$ any strategy profile that has one of the stated outcome. Set u contains the players' utility functions, denoted by $u_{\mathbf{v}}(\cdot)$ and $u_{\mathbf{m}}(\cdot)$ respectively, that define their preferences over the strategy profiles. We define $u_i(\sigma_M) = u_i(\sigma_R) = 0$ for $i \in \{\mathbf{v}, \mathbf{m}\}$; since 0 will be the maximum for $\mathbf{v}$ and the minimum for $\mathbf{m}$, this captures the fact that an outcome in $\{\text{MALICIOUS}, \text{ROBUST}\}$ impedes the malicious node from influencing the knowledge of the verifier. $u_i(\sigma_U)$ can be defined differently according to different criteria. A simple criterion could be to assign $u_{\mathbf{v}}(\sigma_U) = -1$ and $u_{\mathbf{m}}(\sigma_U) = 1$. However, our intuition is that the UNKNOWN outcomes are not the same for the players, because $\mathbf{m}$ could prefer those in which the distance between U and P is maximum. In particular we propose three main criteria to characterize UNKNOWN outcomes:

1. *maximum deception*, $u_{\mathbf{m}}$ is defined as the distance between U and P, while $u_{\mathbf{v}}$ is defined as the additive inverse;
2. *deception area*, $u_{\mathbf{m}}$ is defined as the size of the region $S' \subseteq S$ such that $P \in S'$ is marked as UNKNOWN, while $u_{\mathbf{v}}$ is defined as the opposite;
3. *deception shape*, $u_{\mathbf{m}}$ is defined as the number of disconnected regions $S' \subseteq S$ such that $P \in S'$ is marked as UNKNOWN, while $u_{\mathbf{v}}$ is defined as the opposite.

Players could even use different criteria, *e.g.,* $\mathbf{v}$ and $\mathbf{m}$ could adopt the maximum deception criterion and the deception shape respectively. However, when players adopt the same criterion, the game is *zero-sum*, the sum of the players' utilities being zero. This class of games is easy and has the property that the maxmin, minmax, and Nash strategies are the same. In this case calculations are simplified by the property that $u_{\mathbf{v}} = -u_{\mathbf{m}}$; in the following we shall adopt this assumption.

4 Game Analysis

For the sake of simplicity, we focus on the case in which both players adopt the maximum deception criterion (a reasoning on the same lines can be applied to the other possibilities). In this section we build upon our previous work [9] to analyze the game its extensive form and we finally draw some conclusions valid in the multiple verifier case.

4.1 Maxmin Solution with Three Verifiers

We focus on the case with three verifiers. In our analysis of the game, we consider only the case in which

$$\forall i, j \, \overline{V_i V_j} \leq R \tag{2}$$

In fact, if we allowed $\overline{V_i V_j} > R$, then there could be several unreasonable equilibria. For instance, an optimal verifiers' strategy would prescribe that the verifiers were positioned such that only one point satisfied constraints (1). This strategy would assures the verifiers the largest utility (*i.e.,* zero), no UNKNOWN positions being possible. However, it is not reasonable because the area monitored by the verifiers has a null measure (in the sense of Lebesgue).

At first, we can show that for each action of the verifiers (under the assumption (2)), there exists an action of the malicious node such that this is marked as UNKNOWN. Therefore, there is no verifiers' strategy such that, for all the malicious node's actions, the malicious node is marked as ROBUST or MALICIOUS.

Theorem 1. *For each tuple* $\langle V_1, V_2, V_3 \rangle$ *such that* $\overline{V_i V_j} \leq R$ *for all* i, j*, there exists at least a couple* $\langle U, P \rangle$ *such that* $u_{\mathbf{m}} > 0$*.*

Proof. Given V_1, V_2, V_3 such that $\overline{V_i V_j} \leq R$ for all i, j, choose a V_i and call X the point on the line $\overline{V_k V_j}$ $(k, j \neq i)$ closest to V_i. Assign $U = X$. Consider the line connecting V_i to X, assign P to be any point X' on this line such that $\overline{V_i X} \leq \overline{V_i X'} \leq R$. Then, by construction $u_{\mathbf{m}} > 0$. □

We discuss what is the configuration of the three verifiers, such that the maximum deception is minimized.

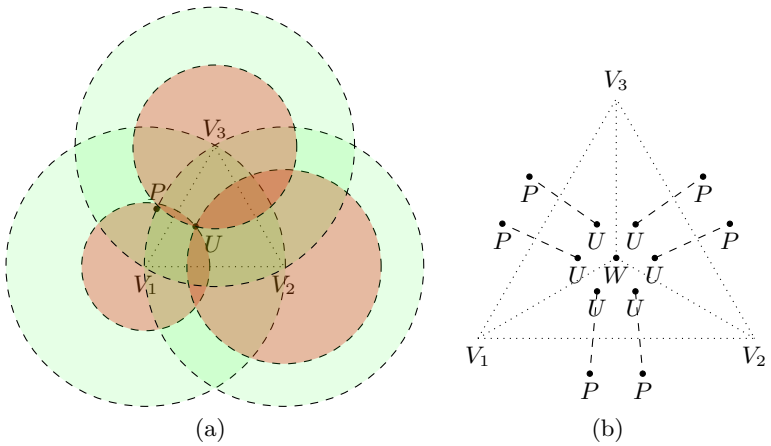
Theorem 2. *Any tuple* $\langle V_1, V_2, V_3 \rangle$ *such that* $\overline{V_i V_j} = R$ *for all* i, j *minimizes the maximum deception.*

Proof. Since we need to minimize the maximum distance between two points, by symmetry, the triangle whose vertexes are V_1, V_2, V_3 must have all the edges with the same length. We show that $\overline{V_i V_j} = R$. It can easily seen, by geometric construction, that U must be necessarily inside the triangle. As shown in Section 2, P must be necessarily outside the triangle and, by definition, the optimal P will be on the boundary constituted by some circle with center in a V_i and range equal to R (otherwise P could be moved farther and P would not be optimal). As $\overline{V_i V_j}$ decreases, the size of the triangle reduces, while the boundary is unchanged, and therefore $\overline{UP}$ does not decrease. □

We are now in the position to find the maxmin value (in pure strategies) of the verifiers, *i.e.,* the action that maximizes the verifiers' utility given that the malicious node will minimize it. The problem of finding the maxmin strategy can be formulated as the following non-linear optimization problem, given V_1, V_2, V_3 such that $\overline{V_i V_j} = R$ for all i, j:

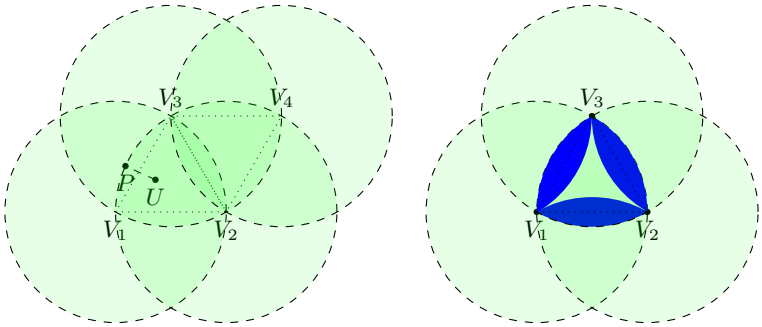
$$\max_{U,P \in S} \overline{UP}$$
$$\text{constraints } (1) \wedge P \text{ outside } V_1 V_2 V_3$$

We normalized the problem assigning $R = 1$ and we solved it by using conjugated subgradients. We report the solution. Let W be the orthocenter of the triangle, U and P can be expressed more easily with polar coordinates with origin in W. We assume that $\theta = 0$ corresponds to a line connecting W to a V_i. We have, $U = (\rho = 0.1394R, \theta = \frac{\pi}{6})$ and $P = (\rho = 0.4286R, \theta = \frac{\pi}{6} + 0.2952)$, and, for symmetry, $U = (\rho = 0.1394R, \theta = -\frac{\pi}{6})$ and $P = (\rho = 0.4286R, \theta = -\frac{\pi}{6} - 0.2952)$. Therefore, there are six optimal couples $\langle U, P \rangle$s. In Figure 3(a) depicts one malicious node's best action and Figure 3(b) shows all the other symmetrical positions. The value of $u_{\mathbf{m}}$ (*i.e.,* the maximum deception) is $0.2516R$. In other words, when the verifiers compose an equilateral triangle, a malicious node can masquerade as unknown and the maximum deception is about 25% of the verifiers' range R.



(a) (b)

Fig. 3. Malicious node's best responses (maximum deception is $\overline{UP} = 0.2516R$).



(a) Maximum deception with four verifiers is still $\overline{UP} = 0.2516R$

(b) U positions that allows for plausible Ps

Fig. 4. Impact of verifiers on U ability to fake positions.

4.2 Maximum Deception with Multiple Verifiers

The result exposed in Section 4.1 are the basis to study situations with multiple verifiers. Our main result is the derivation of a bound between the maximum deception and the number of multiple verifiers.

Initially consider the simple situation in which we have four verifiers and they constitute two adjacent equilateral triangles as shown in Figure 4(a). The maximum deception does not change with respect to the case with three verifiers, since some of the best responses depicted in Figure 3(b) are still available. In fact, the fourth verifier is useful to rule out only the two positions that are on the edge V_4 faces: on this side any fake P would surely marked as MALICIOUS (or even ROBUST if $P \equiv U$) since it would be inside the triangle $V_2V_3V_4$. The proof is straightforward. Consider (without loss of generality) the triangle $V_1V_2V_3$ in Figure 4(a). In order for a node not to be marked as MALICIOUS, U must be in the areas depicted in Figure 4(b). Moreover, any plausible P cannot be neither inside the triangle $V_1V_2V_3$ nor inside the triangle $V_2V_3V_4$, otherwise the node would be marked as MALICIOUS. In fact, any plausible fake P, given a U in the blue area between V_2 and V_3 (see Figure 4(b)), cannot be in regions that are outside both the triangles $V_1V_2V_3$ and $V_2V_3V_4$.

The above observation can be leveraged to give a bound over the maximum deception with a given number of verifiers opportunely placed and tuned such that the shape of the area they monitored is a triangle.

Theorem 3. *Given a triangular area, in order to have a maximum deception not larger than* $\frac{0.2516R}{2^k}$ *we need at least* $2 + \sum_{i=0}^{k} 3^i$ *verifiers.*

Proof. Consider the basic case with three verifiers (composing an equilateral triangle) with range R and $\overline{V_iV_j} = R$. As shown in Section 4.1 the maximum deception is then $0.2516R$. Introduce now more three verifiers such that we have four equilateral triangles with edge $\frac{R}{2}$ as shown in Figure 5. The range of all

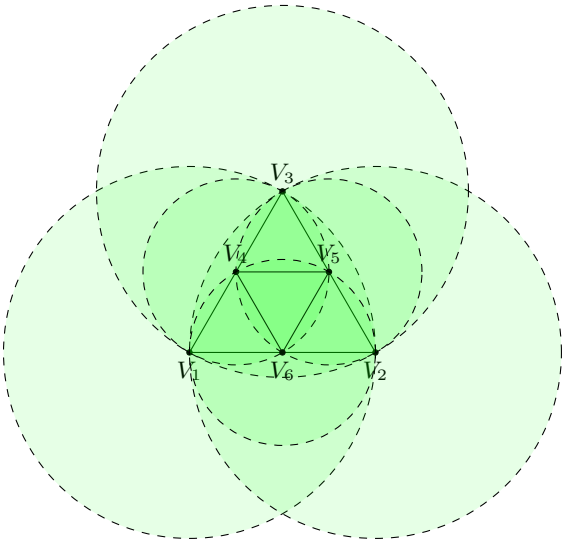


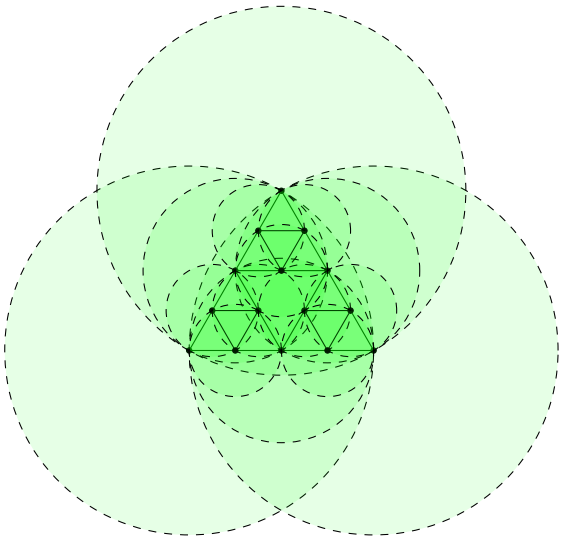
Fig. 5. Maximum deception with six verifiers is $\overline{UP} = \frac{0.2516R}{2}$.

the verifiers is set equal to $\frac{R}{2}$ (*i.e.,* they could just ignore any beacon message that takes longer than needed to cover the distance $\frac{R}{2}$). Since the edge of the small triangles is now $\frac{R}{2}$, the maximum deception here is $\frac{0.2516R}{2}$ and no U positions are possible in the central triangle $V_4V_5V_6$: indeed all the edges of the central triangle are adjacent to the edge of other triangles. This last result allows us not to consider the central triangle when we want to reduce the maximal deception, the malicious node never positioning itself within it. The basic idea is that if we want to halve the maximum deception we need to decompose all the triangles vulnerable by the malicious node by introducing three verifiers. By introducing three new verifiers per triangle we obtain four sub-triangles with an edge that is the half of the original triangle and therefore the maximum deception is halved. In general, in order to have a maximum deception of $\frac{0.2516R}{2^k}$, the number of required verifiers[2] is $\frac{3}{2}(1+3^k)$, as shown in Table 6(b). In Figure 6(a) we report an example with $k = 2$ and 15 verifiers. Notice that when we introduce new verifiers we need to halve the range. In general, we will have verifiers with multiple different ranges. □

The number of verifiers increases according to the formula $n(k) = n(k-1) + 3^k$. Asymptotically $\lim_{k\to\infty} \frac{n(k+1)}{n(k)} = 3$, thus we need to multiply by three the number of verifiers to divide by two the maximum deception. Notice that, as shown in the proof of Theorem 3, the verifiers are required to have different ranges. Increasing the number of verifiers require to add new verifiers with range smaller than those already present in the network.

[2] That is the number of vertices in Sierpinski triangle of order k; see [16].

(a) 15 verifiers ($k = 2$) give a maximum deception $\overline{UP} = \frac{0.2516R}{4} = 0.0629\ R$



(b) Maximum deception

k	# *ver.*	*max. deception*
0	3	$0.2516\ R$
1	6	$0.1258\ R$
2	15	$0.0629\ R$
3	42	$0.02145\ R$
4	123	$0.015725\ R$
5	366	$7.8625 \cdot 10^{-3} R$

Fig. 6. Maximum deception is reduced by increasing the number of verifiers.

5 Conclusion

The trust we put on wireless sensor node localization information is the fundamental base to provide trust to context aware applications and data. Verifiable Multilaterion is a secure localization algorithm, which is able to deal with nodes that falsify their data. VM defines two tests for evaluating node behavior as malicious, or robust, or in the worst case as unknown. Unknown nodes could be simply ignored since VM has not enough information for evaluating the trustworthiness of the node. But unknown nodes could also be faithful, thus by ignoring that source of information, the system loses some opportunities. However, by considering it, we give to a potential attacker the chance of introducing false data into the system. In this paper, by modelling the localization behavior of VM as non-cooperative game we were able to compute an upper bound to the error that an attacker can induce in localization data, given the number of available verifiers. We focused on the maximum deception, that is the distance between the inferred position of an unknown node and the actual one: we found that if the verifiers are placed opportunely, the deception is at most 25% of the power range, and can be halved by triplicating the number of the verifiers. Currently, our results are valid only with a single attacker, since this is key to the assumption that signals cannot be accelerated. In future, we shall consider situations where a malicious attacker can manipulate more nodes.

Acknowledgment

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

References

1. Akyildiz, I.F., Su, W., Sankarasubramaniam, Y., Cayirci, E.: A survey on wireless sensor network. IEEE Wireless Communications 40(8), 102 – 114 (2002)
2. Baronti, P., Pillai, P., Chook, V.W.C., Chessa, S., Gotta, A., Hu, Y.F.: Wireless sensor networks: A survey on the state of the art and the 802.15.4 and zigbee standards. Computer Communications 30(7), 1655–1695 (2007)
3. Brands, S., Chaum, D.: Distance-bounding protocols. In: Proceedings of the Workshop on the Theory and Application of Cryptographic Techniques (EUROCRYPT) (1994)
4. Bulusu, N., Heidemann, J., Estrin, D.: Gps-less low-cost outdoor localization for very small devices. IEEE Personal Communications 7(5), 28–34 (2000)
5. Čapkun, S., Hamdi, M., Hubaux, J.P.: Gps-free positioning in mobile ad-hoc networks. Cluster Computing 5(2), 157–167 (2002)
6. Čapkun, S., Hubaux, J.: Secure positioning in wireless networks. IEEE Journal on Selected Areas in Communications 24(2), 221–232 (2006)
7. Chen, J., Yao, K., Hudson, R.: Source localization and beamforming. IEEE Signal Processing Magazine 19(2), 30–39 (2002)
8. Doherty, L., Pister, K., Ghaoui, L.E.: Convex position estimation in wireless sensor networks. In: Proceedings of the IEEE Conference on Computer Communications (INFOCOM) (2001)
9. Gatti, N., Monga, M., Sicari, S.: A game theoretical analysis of localization security in wireless sensor networks with adversaries, submitted to IEEE GLOBECOM 2010
10. Monga, M., Sicari, S.: On the impact of localization data in wireless sensor networks with malicious nodes. In: Kamra, A. (ed.) Proceedings of the 2nd SIGSPATIAL ACM GIS International Workshop on Security and Privacy in GIS and LBS (SPRINGL'09). pp. 63–70. ACM SIGSPATIAL, ACM, New York, NY, USA (Nov 2009)
11. Niculescu, D., Nath, B.: Ad-hoc positioning system. In: Proceedings of the IEEE Global Communication Conference (GLOBECOM) (2001)
12. Osborne, M., Rubinstein, A.: A course in game theory. MIT Press (1994)
13. Ramadurai, V., Sichitiu, M.: Localization in wireless sensor networks: A probabilistic approach. In: Proceedings of the International Conference on Wireless Networks (ICWN) (2003)
14. Savvides, A., Park, H., Srivastava, M.: The bits and flops of the n-hop multilateration primitive for node localization problems. In: Proceedings of the ACM International Workshop on Wireless Sensor Networks and Application (WSNA) (2002)
15. T.He, Huang, C., Blum, B.M., Stankovic, J.A., Abdelzaher, T.: Range-free localization schemes for large scale sensor networks. In: Proceedings of the ACM International Conference on Mobile Computing and Networking (MOBICOM) (2003)
16. Wessendorf, M.: The on-line encyclopedia of integer sequences. `http://www.research.att.com/~njas/sequences/A067771` (2002), aT&T Labs Research