# Locating Emergency Responders using Mobile Wireless Sensor Networks

**Imane Benkhelifa**
CERIST Research Center, Algiers
i.benkhelifa@cerist.dz

**Samira Moussaoui**
USTHB- Department of Informatics, Algiers
moussaoui_samira@yahoo.fr

**Nadia Nouali-Taboudjemat**
CERIST Research Center, Algiers
nnouali@cerist.dz

## ABSTRACT

Emergency response in disaster management using wireless sensor networks has recently become an interest of many researchers in the world. This interest comes from the growing number of disasters and crisis (natural or man-made) affecting millions of lives and the easy-use of new and cheap technologies. This paper details another application of WSN in the post disaster scenario and comes up with an algorithm for localization of sensors attached to mobile responders (firefighters, policemen, first aid agents, emergency nurses, etc) while assisted by a mobile vehicle (fire truck, police car, or aerial vehicle like helicopters) called mobile anchor, sent to supervise the rescue operation. This solution is very efficient and rapidly deployable since no pre-installed infrastructure is needed. Also, there is no need to equip each sensor with a GPS receiver which is very costly and may increase the sensor volume. The proposed technique is based on the prediction of the rescuers velocities and directions considering previous position estimations. The evaluation of our solution shows that our technique takes benefit from prediction in a more effective manner than previous solutions. The simulation results show that our algorithm outperforms conventional Monte Carlo localization schemes by decreasing estimation errors with more than 50%.

### Keywords

Emergency Response; Disaster Management, Localization; Mobility; Prediction; Speed; Direction; Mobile Anchor; Mobile Wireless Sensor Networks.

## INTRODUCTION

Emergency response demands fast and effective reaction, often in life-threatening situations. It requires collaboration between numerous people and groups: the personnel at an incident site, in the fire trucks, at the command and dispatch centers, at hospitals, etc. In addition, major incidents—like earthquakes, train accidents, industrial accidents and chemical spills—are characterized by having too few resources for the amount of work to be carried out. The dynamic changes in the situation, including the position of victims, professionals, vehicles and other resources, makes it extremely difficult for anyone to obtain and maintain a situational overview, both on superior and specific levels (Romano, Marabissi and Tarchi, 2009).

Location awareness is an important aspect of disaster response: incident command operations depend heavily on tracking rescuers and victims locations. The first hand responders play a major role in effective and efficient disaster management (Kunnath, Madhusoodanan and Ramesh, 2012; Fischer and Gellersen, 2010). Locating and tracking the first hand responders are necessary to organize and manage real-time delivery of medical and food supplies for disaster hit people. This requires effective communication and information processing between various groups of emergency responders in harsh and remote environments.

The technology has made possible to monitor otherwise remote and inaccessible areas such as active volcanoes, avalanches and so on. Wireless Sensor Networks have recently being used in various areas, such as, environmental monitoring, medical care, and disaster prevention and mitigation (Kunnath et al., 2012). This paper details yet another application of WSN in the post disaster scenario and comes up with an algorithm for localization of sensors attached to mobile responders (firefighters, policemen, first aid agents, emergency nurses, etc) while assisted by a mobile vehicle (fire truck, police car, or aerial vehicle) called mobile anchor, sent to supervise the rescue operation. The mobile anchor helps sensors attached to rescuers to localize themselves by broadcasting periodically location beacons and can also perform other tasks necessary for disaster management or for sensor network operations. For instance, taking useful photos, also, it could be used to reconfigure or recalibrate sensors, synchronize the clocks, collect data from sensors (in an efficient manner), deploy new sensors etc. In our days, aerial vehicles and helicopters have become part of the rescue team even in developing countries. As an example, the Algerian civil protection has more than 12 helicopters and aircrafts available for emergency situations (www.protectioncivile.dz). Some solutions were proposed to optimize the anchor trajectory to cover the maximum of sensors can be found in (Benkhelifa and Moussaoui, 2011; Koutsonikolas, Das and Hu, 2007).

When a disaster occurs, the command center sends rescue teams to the affected zone equipped with sensors, and supervised by aid vehicles to facilitate the communication between rescuers, as well as, to organize and to manage real-time delivery of medicine and food to disaster hit people. Sensors attached to rescuers have the role, in addition to locate rescuers, to send information about the disaster situation in the hit area, such as approximated number of victims, area temperature, water level in case of flood, needed aids for victims…etc.

The solution presented in this paper is very efficient and rapidly deployable since no pre-installed infrastructure is needed. Also, there is no need to equip each sensor with a Global Positioning System (GPS). Using GPS proves to be too costly in energy due the excessive energy consumption by GPS receivers and its aggressive usage can cause complete drain of the battery within a few hours (Zhuang, Kim and Singh, 2010). More, to give precise locations, GPS receivers need high time synchronization (Zhuang et al., 2010). Sometimes, this solution is inadequate due to natural obstacles or buildings or others that make connections to satellites impossible due to the NLoS (Non Line of Sight). In addition, price speaking, an integrated GPS chip could cost between 50€ and 90€ depending on the manufacturer. As a consequence, if a rescue operation (such as evacuating earthquake victims) needs 60 rescuers, this costs between 3000€ and 5400€ only to equip sensors with GPS receivers which is very expensive for developing countries. On the other hand, radio antennas are available in all the sensors, by construction, so no additional hardware is needed; thus, why not use these radios to measure the distance between the anchor and the sensor?

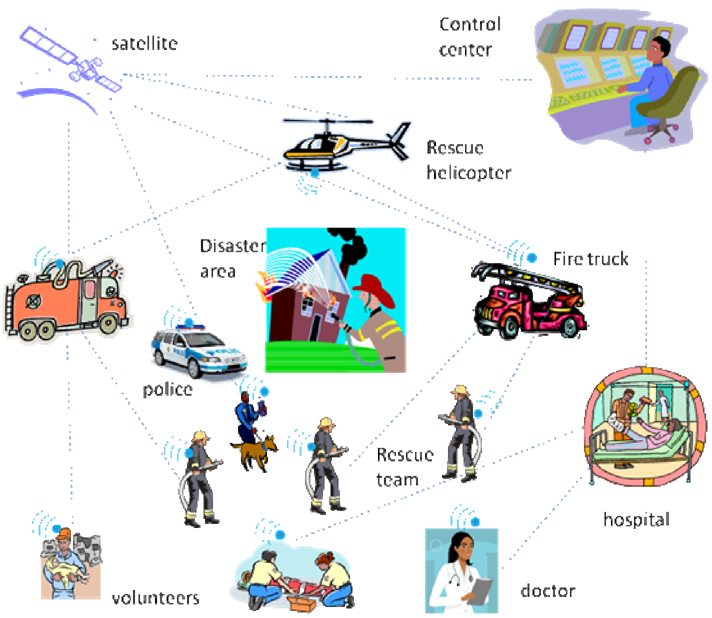


**Figure 1: Emergency Response Scenario.**

As emergency responders can move with random mobility and different directions and speeds, positions of their attached sensors need to be renewed frequently. For this reason, we propose a localization method for mobile sensors based on the prediction of the rescuers movement called Speed and Direction Prediction-based Localization (SDPL). SDPL tries to predict rescuer speed and direction considering its previous position estimations. This provides more positioning accuracy compared with other methods that do not consider any direction information and use only the maximum speed to predict positions.

## RELATED WORK

In (Kunnath et al., 2012), authors present a localization algorithm to localize first responders using the well known Trilateration algorithm. Instead of considering the received signal RSSI, they propose to use Time Difference of Arrival (TDoA) of each packet to estimate coordinates based on distance measurement. This method needs at least three satellites for a 2-dimension disaster area and four satellites in case of 3-dimensions. Authors however did not discuss the mobility of first responders.

The majority of prior research related to localization problems in sensor networks has primarily focused on static sensor (Pal, 2010). Recently, however, more attention has been paid to mobile environments since many applications need sensors to be mobile.

In (Evans and Hu, 2007), authors present an anchor-based localization algorithm for mobile sensors based on the Sequential Monte Carlo method which has been extensively used in robotics (Burgard, Dellaert, Fox and Thrun, 2001). They assume that a sensor has little control and knowledge over its movement, in contrast to a robot. The only assumption that is made is that the sensors or anchors move with a known maximum speed and that the radio range is common to the sensors and the anchors.

The localization process happens in two steps. First, the prediction step leads to choosing a set of samples representing the belief of the node regarding its location. During the prediction step, a node picks random locations within the deployment area, possibly constrained by its maximum speed and the previous location samples. Second, the filtering step aims at removing the impossible locations from the set of samples. The filtering is done using information obtained from the environment, such as the location of the anchors in the case of a sensor anchor or the detection of landmarks in the case of a mobile robot. The process repeats so that sensors can be able to update their position estimations.

An improvement of MCL can be found in (Baggio and Langendoen, 2008). This new version is called Monte Carlo localization Boxed (MCB). It uses steps similar to those of MCL. The major differences lie in the way, anchor information and the drawing method for new samples, are used. The method used for constraining the area from which MCB draws samples is as follows. A node that has heard anchors – one-hop or two-hop anchors – builds a box that covers the region where the anchors' radio ranges overlap. In other words, this box is the region of the deployment area where the node is localized. This box is called "anchor box". Once the anchor box is built, a node simply has to draw samples within the region it covers. Building an anchor box as described above is used in the case where the sample set is empty. In the case where samples already exist, a bounding box is built with an additional constraint, namely, for each old sample from the sample set, an additional square of size 2 * vmax centered at the old sample is built, called "sample box" where vmax is the maximum speed of all the nodes. This updated box delimits per old sample the area a node can move in one time interval at maximum. Whenever a node has an initialized sample set but heard no anchor, the sample box is built solely based on the maximum speed and the old samples. MCB is supposed to support mobility. However, in the evaluation, authors simulate two time units. In the first time units, the nodes move without localizing. To localize nodes, the time freezes and the whole network is localized using a snapshot. There is no movement while localizing nodes.

Although the probabilistic methods described above take benefit from mobility to improve the localization instead of considering mobility as a tedious issue but we have noticed some drawbacks in MCL and MCB:

- More anchors (samples) gains improved accuracy at cost of additional material (memory).
- None of the methods takes into account the direction and the real speed of the nodes.
- Building boxes is based on maximum values. This may increase the box size, more particularly the transmission range for the anchor box and the maximum speed for the sample box. In a real environment, however, the transmission range varies by residual battery, geometric characteristics, and many other factors. If the expected radio range is smaller than the actual radio range, MCB fails to construct a sampling box and picks random values for localization. When the knowledge of the radio range is longer than the actual range, the estimation error also increases since the large transmission range constructs too large sampling box. In addition, to build sampling boxes, MCB uses the maximum speed while in real environment; a node varies its speeds according to its needs and it doesn't travel always with the maximum speed.

While the proposed solution shares the prediction idea presented in related work, to the best of our knowledge, no other localization system based on the prediction of the unknown nodes' speed and direction has been investigated.

## LOCALIZATION ALGORITHM

In this section, we describe the proposed localization algorithm for mobile emergency responders. First, we believe mobility should be taken into account directly when designing new localization algorithms. A wireless sensor should benefit from mobility and exploit it to improve the efficiency of the localization algorithm or to get a better accuracy of its position estimates.

### Assumptions

- For simplicity, we suppose that nodes are deployed in a two dimensional area. Note that the proposed technique could be used equivalently for three dimensions or other location representation. In 3D localization, circles are replaced with spheres.
- Anchor speed is greater than node 'speeds. This assumption is inspired from our emergency response application where the anchor is a vehicle or an aircraft travelling over the affected area and having the task of collecting sensed data and sending them to the rescue teams and to the control center.
- The sensor area is known, in advance, by all the sensors. This assumption was only suggested so that sensors that have never heard an anchor beacon be able to have a global idea about their locations (see case 1 of the location technique).
- Nodes can convert the Received Signal Strength Indicator (RSSI) of the anchor beacon to a distance.

### Location Techniques

Only the anchor can obtain its coordinates at any time (e.g., equipped with GPS since the anchor has no energy constraints and in general is in Light of Site with satellite). At the beginning, unknown nodes (attached to rescuers) have no knowledge about their positions. The anchor (rescue vehicle) broadcasts periodically location packets while travelling through the affected area. The localization process is repeated periodically. At each period of location invocation $\Delta P$, a node can be found in one of the three cases:

- Case 1: The node has never received any anchor message. In this case, the node draws random samples from the whole deployment area and takes the mean as its estimated position.
- Case 2: The node has received anchor messages:

For better understanding, we divide the set of the received anchor positions in $\Delta P$ into *k* sub-sets $E_i$ where each $E_i$ contains only interconnected circles formed by these positions. The circles have the anchor positions as centers and as radius, the distance between the anchor and the unknown node. Let's be $E_k$ the last sub-set.

*1)* If there is no $E_i$ before $E_k$ that contains three or more circles, then, the location estimation is, according only to $E_k$, as follows:

*a)* If $E_k$ has one element, the estimated position is the mean of the *N* samples drawn from the circle centered at the anchor position and has, as radius, the distance between the node and the anchor converted from the Received Signal Strength with a certain probability of noise.

*b)* If $E_k$ has two elements, the node takes as estimation, the gravity center of the intersection zone of the two circles formed by the two heard positions. In reality, the node position is one of the intersection points, but a node cannot decide in which side it belongs without third information.

*c)* If $E_k$ has three elements or more, the node treats the formed circles three by three. The shortest distance between the four intersection points determines two consecutive positions of the node. The node considers, then, the last point as its estimated position (see Figure 2).

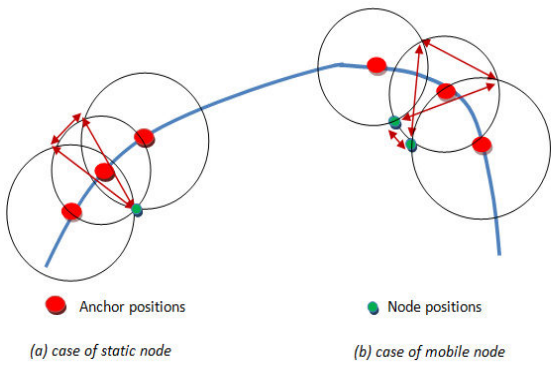


**Figure 2: Estimation with three circles.**

We have taken the shortest distance to determine node' positions because, as supposed, the anchor speed is greater than any node speed. This case is verified by simulation where in 91% of the cases, this technique has given successful results.

The case of receiving three messages or more is very important because it allows nodes to calculate an estimation of their velocities and directions for future utilizations (see the section about the prediction), especially, when a node cannot receive anchor messages anymore whether because of the anchor remoteness or because of obstacle presence. For example, when the rescuer enters into a building or under a tunnel, the node uses its estimated speed and direction to predict its current location.

*2)* If there is a subset $E_i$ ($i < k$) which contains three or more elements, then, the estimation is as follows: The node estimates the positions in $E_i$ similarly to the case 1).c, then, it draws a line $T_i$ through these positions with the *linear regression*.

*a)* If $T_i$ goes across all the circles of $E_k$, then, the node concludes that it has approximately not changed its direction. If $|E_k| \leq 2$, then, the estimated position will be predicted from $T_i$. Else, the node executes 1).c for $E_k$ and uses the new estimated position to refine the previous line of regression.

*b)* If there is no connection between $T_i$ and $E_k$, the node deduces that its direction has been changed. The node, then, calculates its position only according to $E_k$ and deletes all previous estimations because they will be no more useful and may falsify the estimation process.

- Case 3: The node has not received anchor messages in $\Delta P$: if the node has already calculated an estimation of its velocity and its direction, then the node uses them to predict its new position as will be explained in the next section. Else, the node keeps its last estimation.

#### Speed and Direction Prediction

SDPL is mainly based on the prediction of the velocity and the direction of the unknown nodes. To do so, we suppose that nodes follow a rectilinear movement for small moments (see Figure 3) where nodes have a constant velocity and direction during certain time periods ($\Delta t$).

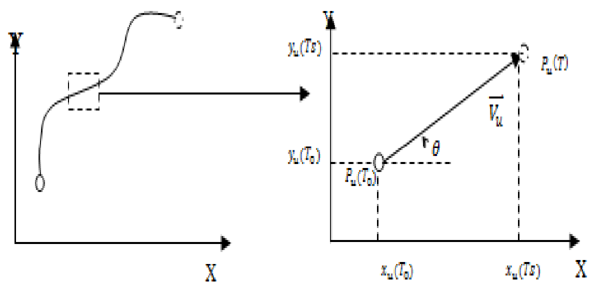


**Figure 3: Example of a node movement.**

The supposition of the rectilinear movement reflects the reality where nodes (e.g., ambulances, fire trucks, firefighters) keep their speed and direction, at least, for moments. This allows nodes to predict positions at time $T = T_0 + \Delta T$, according to figure 3, with the following equations:

$$P_u(T) = \overrightarrow{V_u} \times \Delta T + P_u(T_0) \rightarrow \left\{ \begin{array}{l} x_u = V_u \times cos\theta \times \Delta T + x_u(T_0) \\ y_u = V_u \times sin\theta \times \Delta T + y_u(T_0) \end{array} \right. \quad (1)$$

Where $\theta$ is the angle between the abscissa axe and the speed vector $\vec{V}$.
The speed is calculated as follows:

$$V_u = \frac{\sqrt{\left(x_u(T_0) - x_u(T_s)\right)^2 + \left(y_u(T_0) - y_u(T_s)\right)^2}}{T_s - T_0} \quad (2)$$

Where $T_0$ and $T_s$ are times corresponding to two positions already estimated. If the calculated speed is equal to zero, the node deduces that it is static during $\Delta t$.

The angle $\theta$ defines the node direction. $\theta$ is calculated as follows:

$$\tan\theta = \frac{y_u(T_0) - y_u(T_s)}{x_u(T_0) - x_u(T_s)} \quad (3)$$

In case where a node has already estimated many positions, it calculates the speed between each couple of positions and takes the mean as its predicted speed. In addition, to predict the direction angle, the node makes a linear regression with these positions to deduce, then, the line that provides a best fit for the data points using the *least squares* approach (see Figure 4).

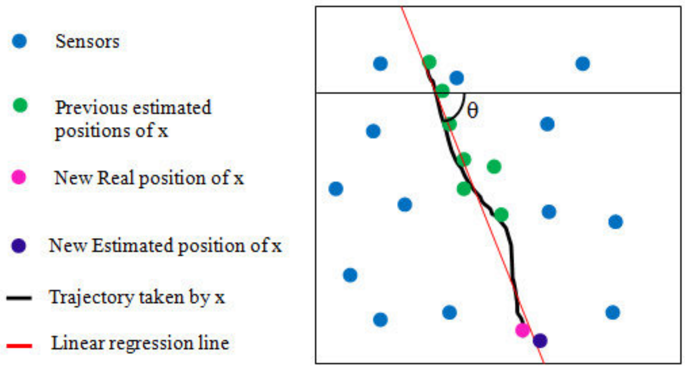


**Figure 4: Direction Prediction of node $x$ using Least Squares Line.**

After the prediction of the speed and the direction, a node estimates its coordinates (x, y) as follows:

$$\begin{cases} x = x_{prev} + \cos\theta \times V \times T_{dif} \\ y = y_{prev} + \sin\theta \times V \times T_{dif} \end{cases} \quad (4)$$

Where $(x_{prev}, y_{prev})$ is the last estimated position; $(V, \theta)$ are respectively the predicted speed and direction angle; $T_{dif}$ is the time between the current time and time of the last estimation.

## EVALUATION

To evaluate our proposed algorithm, we opted for simulation. Our simulations were performed using the Network Simulator (NS2) version ns-allinone-2.34. NS2 is widely used in academic network researches.

To analyze the simulation results, the main metric is the localization error. As done in (Evans and Hu, 2007; Baggio and Langendoen, 2008), the localization error is calculated by measuring the distance between the real and the estimated position of a node as follows:

$$e = \sqrt{(x_{es} - x_{ac})^2 + (y_{es} - y_{ac})^2} \quad (5)$$

Where $(x_{es}, y_{es})$ is the estimated position and $(x_{ac}, y_{ac})$ is the actual position.

We consider the average localization error over all sensors.

$$Mean\ Error\ = Sum\ (e) /\ number\ \ of\ nodes \tag{6}$$

We compare the mean error of our method with the one gives by MCB, since it outperforms MCL (Baggio and Langendoen, 2008). To compare only the localization algorithm, we consider for both methods, one single mobile anchor with the same mobility scenario and we believe that if SDPL uses more mobile anchors (eg. the fire trucks and police cars at the same time, this can improve the efficiency of the technique). For unknown nodes mobility, we have chosen to use the random waypoint mobility model (Boleng, Davies and Camp, 2002) where each node can vary its speed and direction at each own time step. Note that any mobility model can be applied since the proposed solution does not depend on the mobility model of nodes. Finally, as suggested in (Evans and Hu, 2007; Baggio and Langendoen, 2008), we use a sample set of 50 location estimations for MCB.

For each method, we consider a deployment area of 200×200 $m^2$, the transmission range (r) is set to 30 m as an example. Obviously, when the transmission range is bigger, more nodes hear the anchor beacons and thus can localize themselves. This gives more localization coverage. In addition, nodes get more beacons and thus can predict their mobility following the SDPL prediction technique but with the price of big boxes which may increase the location error. The probability of noise is chosen to be set to 0.3 as an example. This probability is taken into account by nodes when converting the anchor RSSI into a distance. As in (Evans and Hu, 2007; Baggio and Langendoen, 2008), the mean error is referenced to the transmission range (r).

We note that the localization error does not depend on the nodes' density; since nodes only receive beacons directly from the mobile anchor. No neighboring interaction is needed. Thus, sensor energy is saved. The number of sensor nodes is 100. The speed of the anchor is constant and equal to 20 m/s same to 72 km/h. $\Delta P$, the positioning invocation period is set to 5s. That means that each 5 seconds, nodes estimate their positions. The mean error in the tests was calculated when the anchor covers the whole area for the first time, let us call it *first round localization*. Obviously, when the anchor covers and travels the area many times, the mean error of sensors decreases because nodes have the chance to receive more beacons from the anchor whenever it passes close to them and so, they can improve their estimations. The simulation results are averaged over 50 runs and taken while nodes are moving.

**Variation of Nodes' Speed**

For the first test, we set the broadcasting interval of the mobile anchor to 1s, it means, each 1s, the anchor diffuses its current location to its nearby sensors. Figure 6 shows that, for both methods, when the maximum speed of unknown nodes increases, the mean error increases too. Indeed, the high speed (up to 54 km/h in the graph) allows the node to move quickly away from the anchor. This deprives the node of receiving anchor messages that help it localizing. On the other hand, when the speed is small (the smallest speed in the graph is 3.6 km/h), the node moves slowly and the anchor messages remain valid for certain time. This allows the node to minimize its localization error. In addition, in the case where the node cannot receive anchor messages, MCB uses always the maximum speed to build the sample box in order to predict the new position and has no knowledge about its direction; while SDPL tries to predict the speed and the direction of the node movement. This enables the node to minimize the estimation error and thus outperforming MCB. Let us explain these results in real world: if a firefighter walks with a speed up to 5 m/s then with SDPL method, the position error is averaged by 27 m during the first round localization. Considering outdoor area, it's largely sufficient to get information about the fire position, and even when considering indoor area, it is still in the neighborhood (same apartment, same building). While with MCB, the mean error is 60 m. Obviously, when the anchor travels more in the area, we can get more precise results.

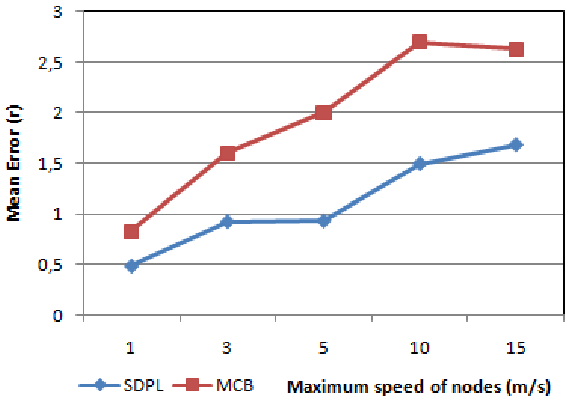


**Figure 6: Mean Error VS Speed of nodes**

### Variation of the Broadcasting Interval

Now, we set the maximum speed of unknown nodes to 5m/s and we study the impact of the broadcasting interval on the localization error. The broadcasting interval is the time between two consecutive messages sent by the anchor. Varying this interval has a direct influence on the number of the diffused beacon messages.

In Figure 7, we notice that when decreasing the anchor broadcasting interval, the localization error decreases. In fact, if this interval is small, that means the anchor diffuses more location messages in each positioning invocation period (let's remind, is set to 5s). Thus, nodes receive more messages that help them to determine their positions. In the case of MCB, the high number of messages enables building smaller anchor and sample boxes which allows drawing samples closer to the real position. As for SDPL, the high number of messages enables getting more speed and direction-prediction cases. This makes estimated positions closer to the real positions. Figure 8 presents in details the occurrence ratio of each calculation case with SDPL. When the broadcasting interval is small, the probability of being in case n° 4 and n° 5 is high. This explains the decrease of the localization error in Figure 7 for SDPL method.

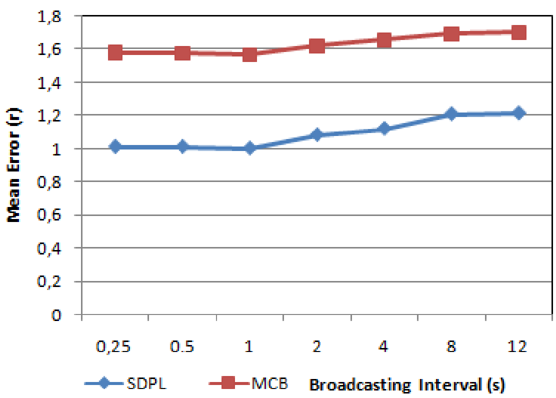


**Figure 7: Mean Error VS Broadcasting interval**

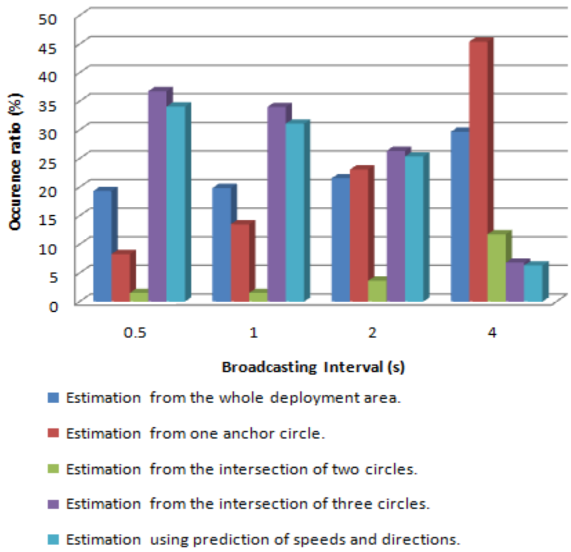


**Figure 8: Estimation cases with SDPL**

## CONCLUSION

In this paper, we proposed a new method designed for locating emergency responders using mobile sensor networks called Speed and Direction Prediction -based Localization that deals with mobility issues to allow responders localizing themselves with efficiency by making better use of information a sensor gathers from a mobile anchor. The proposed method is mainly based on the prediction of the speed and the direction that a node moves with. Its main contribution lies in case when sensors cannot get location signals from the anchor, they still can estimate their positions independently thanks to an efficient prediction which makes it a decisive choice if compared with GPS that depends heavily on connections with satellites.

Our mechanism requires only that the mobile anchor broadcasts some beacon messages to assist sensors getting initial information about their locations. SDPL is based on simple calculation of speed and direction, thus, does not need great processing time to calculate complicated formulas, this make it very suitable for disaster management applications that need quick information. Sensors with SDPL technique estimate their positions by themselves; this makes our solution scalable and useful for applications such as fire forests that need huge number of sensors and other applications with mobile sensors. The results of simulations of our algorithm show that it allows a node to get an improved accuracy more than 55% over Monte Carlo localization Boxed which is an improved derivative of the basic Monte Carlo localization. Most importantly, SDPL ensures to a node not being able to receive information from the anchor to be localized thanks to an efficient prediction.

We plan to test our method under real experiments with MICA-Z motes available in our center; the mobile anchor will be a car for first tests and then a helicopter for further tests. The technique can be more evaluated by varying the anchor's trajectory and speed as well as considering different parameters of noise and interference since these parameters may influence the localization accuracy.

## ACKNOWLEDGMENTS

This work is part of SAGESSE project supported by CERIST Research Center, Algiers, Algeria and funding from the Algerian General Direction of Scientific Research and Technological Development fund under the grant 5/CR0167/177.

## REFERENCES

1. Romano F., Marabissi, D. and Tarchi, D., (2009) A novel Communication infrastructure for emergency management: The In.Sy.Eme. vision, *Wirel. Commun. Mob. Comput.* **10**: pp. 1672–1681.
2. Kunnath, A.T., Madhusoodanan, A. and Ramesh, M. V., (2012) Locating Emergency Responders in Disaster Area Using Wireless Sensor Network, *Springer Wireless Communications and Applications.* Volume 72, Part 1, pp. 1-10.
3. Pal, A. (2010) Localization Algorithms in Wireless Sensor Networks: Current Approaches and Future Challenges, *In Journal of Network Protocols and Algorithms*, Vol. 2, No.1.

4. Evans, D. and Hu, L. (2007) Localization for Mobile Sensor Networks, In: *Proceedings of IEEE FGCN 2007 workshop chairs.*
5. Fischer C. and Gellersen H., (2010) Location and Navigation Support for Emergency Responders: A Survey, In Pervasive Computing, IEEE. Volume 9, Issue 1. pp. 38-74
7. Baggio A. and Langendoen, K., (2008) Monte Carlo Localization for Mobile Wireless Sensor Networks, *In ELSEVIER Ad Hoc Networks 6*, pp. 718-733.
9. Burgard, W., Dellaert, F., Fox, D. and Thrun, S. (2001) Robust Monte Carlo Localization for Mobile Robots, *Artificial Intelligence Journal*.
10. NS-2 Simulator, http://www.isi.edu/nsnam/ns/, checked on 02/13/2013
11. Boleng, J., Davies, V. and Camp, T., (2002) A Survey of Mobility Models for Ad Hoc Networks Research, *Wireless Communications and Mobile Computing*. Volume 2.
12. Benkhelifa, I. and Moussaoui, S., (2011) Anchor Path Planning-based Localization for wireless sensor networks, In: *Proceedings of MIC-CCA*. pp. 48-53
13. Koutsonikolas, D., Das, S. M. and Hu, Y. C. (2007) Path planning of mobile landmarks for localization in wireless sensor networks, *ELSEVIER Computer Communications*. pp. 2577–2592
14. Zhuang, Z., Kim, K. H. and Singh, J. P., (2010) Improving Energy Efficiency of Location Sensing on Smartphones, In: *Proceedings of MobiSys*.