6

A Pragmatic Approach to Cooperative Positioning in Wireless Sensor Networks

Albert Bel Pereira, José López Vicario and Gonzalo Seco-Granados
Department of Telecommunications and Systems, Universitat Autònoma de Barcelona, Barcelona, Spain

6.1 Introduction

Location estimation in wireless sensor networks has become an important field of interest from researchers [1–3]. This is due to the demand for knowledge of the node position by the majority of applications. In environmental monitoring, such as fire or agriculture control, a basic premise to give sense to all the data measured is to know their location; if not known, data could be considered as meaningless information.

The main purpose of a localization algorithm is to estimate those positions of nodes with unknown coordinates from the following information: a priori knowledge of some node positions and intersensor measurements. Hence, the majority of existing localization methods applied in a wireless sensor network (WSN) tries to achieve the best accuracy considering the restrictions that this kind of network imposes.

Although nowadays there are many methods of localization in wireless networks, such as the global positioning system (GPS), a localization method suitable to be used in a WSN must take into account the resource constraints imposed by the nodes, such as energy consumption and the costs of transmission and computing hardware. The increase in terms of size and cost of energy required by the GPS hardware makes this method unsuitable to be applied in WSN. It could only be used to obtain a priori knowledge of the positions of reference nodes that are present in the network.

Microwave and Millimeter Wave Circuits and Systems: Emerging Design, Technologies, and Applications, First Edition. Edited by Apostolos Georgiadis, Hendrik Rogier, Luca Roselli, and Paolo Arcioni.

The objective of this chapter is to provide a review on cooperative schemes for WSN. Among all of them, special emphasis is given to received signal strength (RSS)-based techniques as these provide suitable solutions for practical implementation and take care of the restrictions in terms of cost and complexity. Since the accuracy of RSS methods depends on the suitability of the propagation models, cooperative localization algorithms that dynamically estimate the path loss exponent are also described. The chapter is also interested in the cost, in terms of energy consumption, that a localization algorithm has. In that sense, in order to achieve a trade-off between energy consumption and accuracy, a node selection criterion is proposed. In addition, practical examples based on real WSN deployments are presented.

The rest of the chapter is organized as follows. In Section 6.2 the two main categories of internode measurements, range-free and range-based, are presented. Furthermore, a first classification of the localization algorithms between cooperative and noncooperative is also presented. In Section 6.3 several localizations algorithms, divided between centralized and distributed, are presented. In Section 6.4 the two parts that compose an RSS-based algorithm, measurement and location update phases, are presented. Later, in Section 6.5 the different node selection mechanisms existing in the literature are presented. Furthermore, two different methods are proposed and an energy model is presented. In addition, the localization algorithm with joint node selection and path loss exponent estimation is presented. Finally, Sections 6.6 and 6.7 present, respectively, the simulation and experimental results obtained from the algorithm proposed.

6.2 Localization in Wireless Sensor Networks

Nowadays, the existing algorithms are able to locate nodes inside a WSN. Choosing between the different approaches depends on requirements that the final application demands. In order to differentiate between the great number of existing methods different classifications have been created.

The first classification divides existing methods into two main categories: range-based and range-free approaches. These approaches differ in the way of obtaining internode distances.

In the context of a WSN, a second classification is cooperative versus noncooperative algorithms. These classifications will be presented in the following sections.

6.2.1 Range-Free Methods

Range-free methods are based on connectivity information. These are the simplest measurements that an algorithm can do. The basic idea is to decide if a node is connected to an adjacent node or not. The connectivity information is usually obtained from RSS measurements. Considering perfect circle radio coverage, any node that receives an RSS above a threshold is supposed to be connected to the receiving node. Also a node can be considered inside the coverage range of another node by measuring the number of received packets. The commonest algorithms based on connectivity measurements are centroid, DV-Hop or APIT [4–6].

6.2.1.1 Centroid

One of the simplest methods used is the centroid. The mean of the coordinates of all anchors (nodes with known location) becomes the position estimation of each listening node. A

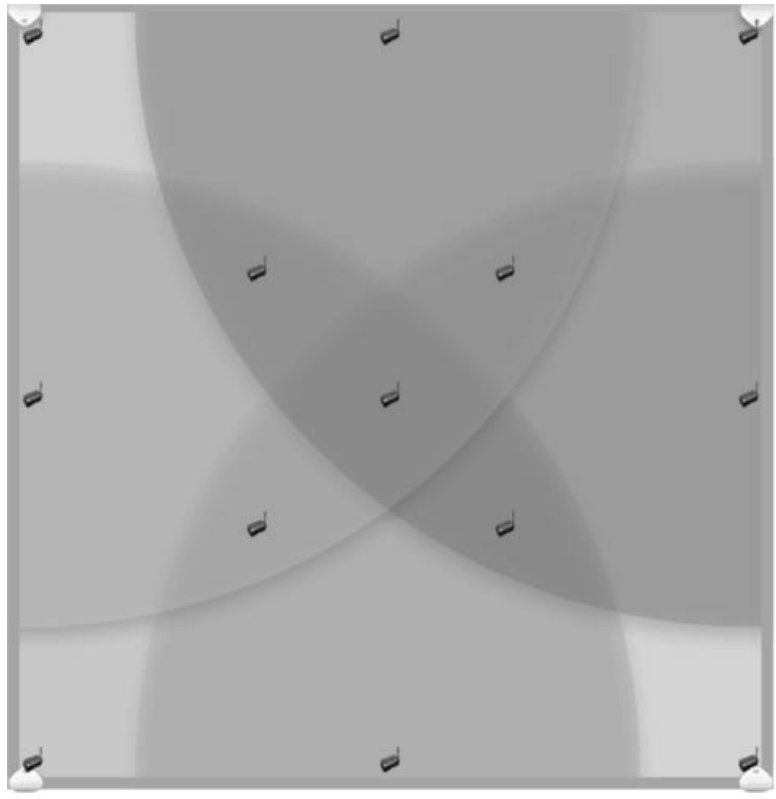

Figure 6.1 Centroid method with the different possibilities of positioning.

possible improvement could be the introduction of weights in order to achieve higher accuracy. An application of this method in a WSN is presented in Reference [4]. The authors considered an unconstrained outdoor environment. The results presented are for a network formed by four anchor nodes with an ideal spherical transmission range (see Figure 6.1). In that case an average localization error of 1.83 m is obtained. The major problem of this method, applied to a WSN, is that the accuracy is directly related to the number of anchor nodes. It could be increased if the number of overlapped reference nodes is increased. Nevertheless, the inclusion of more anchor nodes increases the cost of the network.

6.2.1.2 DV-Hop

The distance vector (DV)-Hop algorithm [6] is another method that is based on connectivity measurements. The DV-Hop is more complex than the centroid but more accurate estimations are achieved. The algorithm starts with a broadcast message sent by all the anchors. Nodes in the network count how many hops exist between them and all the anchor nodes inside the network. Each node selects the shortest path to every anchor node. Once having these hop distances, anchors broadcast the average distance per hop to neighbours (achieved when a message from an anchor is received by another anchor). With both measurements a trilateration or multilateration method is used to obtain the final location.

The DV-Hop algorithm is a distributed strategy that does not require any extra hardware and is capable of providing global coordinates to the nodes. The simplicity in terms of calculation and no increase in terms of cost due to hardware requirements makes this algorithm suitable to be used in a WSN.

Compared to the previous algorithm, the accuracy does not only depend on the connectivity with the anchors. Now the accuracy depends on the connectivity with any node (anchor or not anchor). In that sense, reducing the necessity of having anchors inside the network will reduce the cost of the network. At the end of the subsection a comparison between the three range-free algorithms will be made.

6.2.1.3 APIT

The approximate point in triangle (APIT) test [5] is an area-based approach. The APIT algorithm is basically divided into four steps: each node receives the location of as many anchors as possible; given all possible combinations of three anchors, each node has to form triangles; then the node has to determine whether or not it is within each triangle; and finally the position is obtained by calculating the centroid of the intersection of all triangles selected (see Figure 6.2).

As in the previous approaches, the APIT algorithm is capable of providing global coordinates (thanks to the anchor nodes) without increasing the cost of the hardware. However, in comparison with the DV-Hop, the APIT algorithm depends only on communications with the anchor nodes. The APIT and centroid algorithms are considered noncooperative methods, because they do not take advantage of communication between nonlocated nodes. Hence,

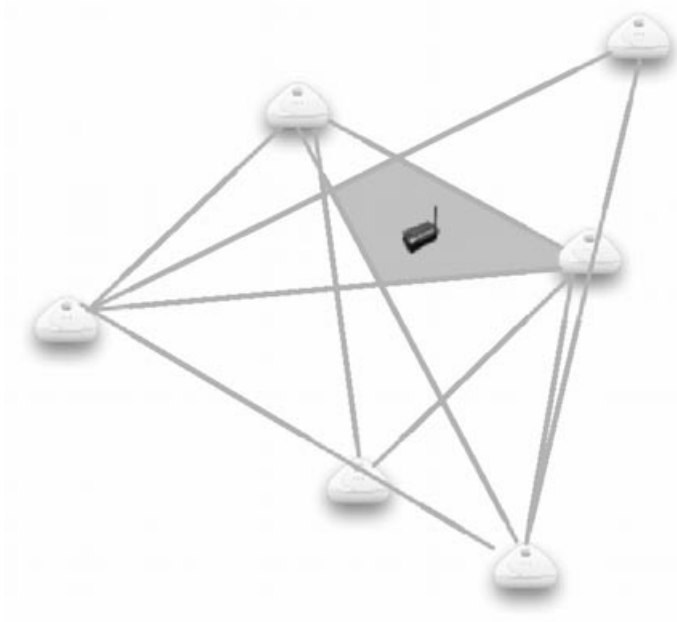

Figure 6.2 APIT estimation.

the accuracy of the solution will depend, in part, on the density of anchor nodes or having long-range transmission range anchor nodes able to be sensed at further distances.

6.2.1.4 Comparison of the Three Methods

A comparison of these three methods is presented in Reference [5]. Authors analyse the dependence of the accuracy of the three methods with different parameters such as anchor density, radio range ratio or nonlocated node density. The results reflect that no single algorithm can be used in all scenarios. The centroid algorithm has the largest localization error, but it is not dependent on features such as node density. Furthermore, this is the algorithm with less exchange of information and is simpler to implement. The DV-Hop algorithm requires a higher exchange of messages through the network and it is necessary to have a higher density of nodes in order to obtain good values of accuracy. The DV-Hop achieves a mean error equal to the radio range R. The APIT algorithm obtains similar results in terms of accuracy but nonlocated nodes have to have more anchor nodes inside their radio range. The parameter that badly affects all the algorithms is the anchor node range. As the transmission range of the anchors increases, the error also increases. The APIT algorithm achieves a mean error 0.75 times the radio range R.

6.2.2 Range-Based Methods

The range-based classification groups all the methods that estimate the internode distances or angles with the use of range information. Range information helps the algorithm to achieve a better accuracy in distance estimates than range-free approaches.

In this subsection the different signal metrics used to obtain distance estimates are presented. These distances are then used to determine node position.

6.2.2.1 Time of Arrival (TOA)

The first range-based measurement is the time of arrival (TOA) [7]. The TOA is based on measuring the difference between the sending time of a signal at the transmitter and the receiving time at the receiver. The major problem of this method is the possible lack of synchronization between nodes. Errors of about 2% are achieved over a communication range of 3–6 m [8]. Moreover, the node clocks resolution should be of the order of nanoseconds (in radio frequency (RF) 1 ns translates to 0.3 m [9]). Although it is a range measure usually used in wireless or satellite networks [10] (in which base stations and mobile nodes are synchronized), a recent trend uses time measurement approaches with ultra wideband signals (UWB) [11–13]. The UWB signal achieves a high accuracy because the transmitted pulses has a wide bandwidth and hence a very short pulse waveform. With the recovery of this transmitted pulse it is possible to estimate the distance between the receiver and transmitter.

6.2.2.2 Time Difference of Arrival (TDOA)

Another existing time measurement approach avoids the necessity of having an entire synchronized network. This method is known as the time difference of arrival (TDOA) and two different ideas are presented [14].

The first TDOA method [11] is based on the measurement of the difference between the arrival times of a signal sent by a transmitter at two receivers. This method assumes that the

receiver locations are known and the two receivers are perfectly synchronized. For that reason, it is mostly used in cellular networks where the complexity of base stations is considerably relaxed.

Another method also based on a difference of arrival times is presented in References [15] and [16]. The basic idea is to eliminate the necessity of having a synchronized network, neither the senders nor the transmitters. In order to achieve its purpose the method uses a combination of two kinds of signal, for example RF and ultrasonic signals, which have different velocities of transmission. One transmitter sends two kinds of signal to a unique receiver. The time difference between the first and the second signal is used as an estimate of the one-way acoustic propagation time. In this case nodes require extra [16] hardware in order to be able to transmit and receive different signals.

The first case is more appropriate in cellular networks due to the fact that base stations have fewer requirements in terms of complexity or cost. It is possible to synchronize them; hence they could act as the receivers at the time of estimating the TDOA. On the other hand, the second TDOA approach could be more suitable to be used in a WSN. The major con is the necessity of including extra hardware in the entire network.

Both approaches obtain a good accuracy. In Reference [16] the results show an average error of the distance estimates between 29 and 8 cm. However, both methods increase the complexity of the network and the cost of nodes.

6.2.2.3 Round-Trip Time of Arrival (RTOA)

The round-trip time (RTT) method [17] avoids the synchronization constraint that the TOA or first TDOA methods impose, nor the hardware requirements of the second TDOA method. The measurement starts when a node A sends a packet to a node B. When node B receives the packet, it retransmits it to node A. At the end, node A receives the packet; hence, it can calculate the propagation time because the difference between the sending time and the receiving time at node A is twice the propagation time plus the processing time at node B (obtained from specifications or estimated at the calibration time).

Numerical results based on different experimental setups are presented in Reference [17]. The results show that RTT measurements give a root mean square (RMS) error between a minimum of 75 cm and a maximum of 2.51 m. The difference in accuracy compared to that achieved by TOA or TDOA measurements is remarkable. Furthermore, this technique needs a higher exchange of packets in order to estimate the internode distance.

6.2.2.4 Received Signal Strength Indicator (RSSI)

In this case, the distance between two nodes is obtained by using the measured power of the received signal [18]. The RSS is measured during normal transmission. Hence, this technique does not impose any extra requirement in terms of complexity or hardware. Compared to the AOA- or TOA-based approach, this technique has become the most inexpensive. However, distance estimates obtained through RSS measurements suffer from errors induced by shadowing and multipath effects. Usually, RSS-based distance estimations are based on the well-known radio propagation path loss model. This model assumes that the power decays proportionally to the distance $1/d^{\alpha}$, where α is the path loss exponent. In order to include the shadowing effects the power received is modelled as a lognormal variable (Gaussian if it is

expressed in dB), resulting in

$$P_{R_x}(dBm) = P_0(dBm) - 10\alpha\log_{10}(d) - \upsilon_i \tag{6.1}$$

where P_0 is the power received at a reference distance (usually 1 metre) and v_i represents the shadowing effects modelled as a Gaussian with zero mean and variance $\sigma^2_{\upsilon_i}$ expressed in dB.

The adoption of lognormal modelling is usually motivated by experimental results such as those provided by References [19] and [20]. Some results in terms of accuracy are presented in Reference [21]. The authors carried out a measurement campaign by means of using TelosB motes. The average error achieved in the measured distances is 2.25 m, for a distance between 1 and 8 meters. The RSS-based estimates achieve worse accuracy compared to that achieved with time measurements. However, the major advantage is that RSS provides a lower complex solution. Moreover, distance estimates could achieve better accuracy if an accurate model is used.

6.2.2.5 Angle of Arrival (AOA)

Angle or direction of arrival, AOA and DOA [9], are those methods that rely on the measurements of the angle between senders and receivers. In Reference [9] two different ways of estimating distances are presented. The first method [22], which is the most common, estimates the angle of arrival by means of using a sensor array. Each sensor requires two or more sensors placed at a known location with respect to the centre of the node. The angle is estimated following the same approach as in a time-delay estimation. The measurement consists of two phases. At the first phase, anchors transmit their location and a short omnidirectional pulse. Then they transmit a beacon with a rotating radiation pattern. Taking advantage of beamforming techniques, the anchors are able to transmit directional pulses every T seconds and to change the direction of the signal by a constant angular step $\Delta\beta$. Sensors have to register the arrival time between the first omnidirectional signal and the time of arrival of the pulse with maximal beacon power. This difference in time (Δt) allows the sensor to estimate the angle of arrival as

$$\beta = \Delta\beta\frac{\Delta t}{T} \tag{6.2}$$

The accuracy achieved in Reference [22] is an average error of 2 m in a scenario with 6 anchors and 100 nonlocated nodes uniformly distributed in a 50 m × 50 m area. Increasing the anchors produces mean errors in the localization below 1.5 m.

In References [23] and [1], RSS measurements from directional antenna arrays on each node were also used to estimate arrival angles. Accuracy errors below 1 metre are achieved compared to those achieved with distance-based algorithms. However, the increase in cost and size of the nodes makes the AOA a more complex solution although it achieves good results in terms of localization accuracy.

6.2.2.6 Radio Interferometry

This technique [24] is based on exploiting interfering radio waves. Although it is also based in RSS measurements, the procedure of extracting the distance is more complex. The basis of

radio interferometry is to utilize two transmitters (two reference nodes with known location) to create an interference signal and compare the phase offset at two receivers. By measuring this relative phase offset at different carrier frequencies, it is possible to obtain a linear combination of the distances between both transmitters and receivers. The accuracy obtained is considerably increased. Results achieved in Reference [24] show that more than 50% of the range measurements achieve accuracy lower than a quarter of the wavelength. In Reference [25] a tracking algorithm based on radio interferometry measurements is presented. The mean absolute error achieved with the mobile experiment is between 0.94 and 1.96 m. The error achieved with a stationary experiment is between 0.54 and 0.83 m. A high accuracy is achieved with the measurement of RSS signals (no extra hardware is required). On the other hand, the method presents some requirements, such as synchronization of some nodes and signal processing units able to estimate the carrier offset.

6.2.3 Cooperative versus Noncooperative

Once distance estimates are obtained the next step is to estimate node locations inside the network. This second phase, usually called the location-update phase, could also be classified as cooperative or noncooperative.

Noncooperative approaches estimate the nonlocated node positions by only considering the anchor node positions. In this case, the accuracy of these approaches mainly depends on the density of the anchor nodes or the usage of long-range transmission anchors. Using this kind of technique in large-scale networks could be critical. On the other hand, in small-scale networks, where nodes have a high probability of having direct communication with anchor nodes, the use of a noncooperative technique could become a great option.

In cooperative algorithms, there are no restrictions in the communication between any nodes inside the network. Nodes are able to obtain information from more nodes than only anchors. With this strategy the number of anchor nodes in the network can be reduced. Hence, cooperative localization can offer increased accuracy and coverage.

6.3 Cooperative Positioning

The cooperative algorithms have become an accurate approach for localization algorithms in low-cost and low-complex sensor networks. Cooperating with the entire, and not only with the anchor node, network can increase the accuracy, in terms of error, of the final position estimate. An important purpose is to achieve a good trade-off in terms of accuracy versus network complexity and cost, and the cooperative approaches are a kind of algorithm that could achieve a good trade-off. Furthermore, a cooperative algorithm is a scalable solution because it does not only depend on the number of anchors nodes; it also takes advantage of all the nodes inside the network (anchor or nonlocated).

Cooperative approaches could be classified into two categories: centralized versus distributed. Although this classification could also be applied to noncooperative algorithms, the discussion is here presented as differences among such alternatives, emphasized in a cooperative system.

Centralized approaches involve those algorithms in which one node becomes a central unit. This central unit is the one that has to recollect all data and it is the only one that computes all the position estimates. On the other hand, in a distribution algorithm each node is responsible for estimating its own position, relative to its neighbours.

In the following subsection different alternatives of both approaches are presented, discussing which are their strong points and when they are the appropriate algorithms to be used.

6.3.1 Centralized Algorithms

In some applications it is appropriate to implement a centralized WSN due to the nature of the application, for example environmental monitoring, where the information should be controlled in a central point. In this kind of application, a unique central processor controls all the data. In this situation, the use of centralized algorithms in order to locate the nodes inside the network should be useful.

On the one hand, centralized algorithms provide good accuracy, because the central node has information of the entire network. Moreover, these nodes are less restricted in terms of complexity and hence a more complex algorithm could be implemented. On the other hand, all the information collected at the network must be transmitted to the central node; hence the traffic is increased and the scalability is reduced.

6.3.1.1 Multidimensional Scaling

Multidimensional scaling (MDS) [1] was originally developed for use in mathematical psychology and has many variations. Although MDS is normally developed in a centralized fashion, distributed-based algorithms exist in the literature. The most usual approach is the MDS-MAP (maximum a posteriori) [26], which is a direct application of the simplest kind of multidimensional scaling: classic metric MDS. The basic idea is to arrange objects in a space of a certain number of dimensions trying to reproduce the dissimilarities observed in the objects. Adapting to a localization algorithm, the objects are the nodes and the dissimilarities are the distance estimates. By means of using the law of cosines and linear algebra the MDS is able to reconstruct the relative positions of the points based on the internode distances. The last step of an MDS algorithm is transforming the relative map obtained to an absolute map based on the knowledge of the absolute position of some anchors.

Clearly, MDS has potential in the sensor localization domain. It is possible to construct a relative map without knowing any absolute position. Results that depend on different features are shown in Reference [26]. They show that the accuracy of the MDS-MAP is highly dependent on the connectivity. In order to reduce the error, the algorithm needs a high density of nodes (a minimum number of 12 cooperating nodes). Another important point is the high accuracy achieved when range information is used instead of connectivity information. In conclusion, it is possible to achieve an error less than half the range radio of the nodes with a number of 12 cooperating nodes and 4 anchors in order to change from a relative to an absolute map. A major drawback is the necessity of having all the information in a central node. This problem can be reduced by means of using map-stitching techniques [27].

6.3.1.2 Semidefinite Programming

Semidefinite programming (SDP) is a subfield of the convex optimization. SDP basically consists in minimizing a linear function subject to the constraint that an affine combination

of symmetric matrices is positive semidefinite. Such a constraint is nonlinear and nonsmooth, but convex, so semidefinite programs are convex optimization problems. The major problem of applying these techniques is that the localization problem is a nonconvex problem. Hence, the basic idea of an SDP algorithm is to convert the nonconvex quadratic distance constraints into linear constraints by introducing a relaxation to remove the quadratic term in the formulation. Three different approaches applying SDP algorithms are presented in Reference [28]. The best result achieved is an error of 5% of the radio range. This accuracy is highly affected by the noise factor, although it is maintained below a value of 20% of the radius range having a value equal to or higher than 0.3. Moreover, when the size of the network increases the solution of a large SDP becomes more complex. This problem can be solved by means of dividing the network into several clusters, reducing the complexity of the entire network.

6.3.1.3 Maximum-Likelihood Estimation

The maximum-likelihood estimation (MLE) is a centralized localization algorithm [19]. As it occurs with MDS it can be solved in a distributed fashion. MLE is a popular statistical method used for fitting a statistical model to data and providing estimates for the model's parameters. One of its advantages is its asymptotic efficiency. This method applied to a WSN is developed in Reference [19]. Simulations carried out in a scenario with 40 nonlocated nodes achieve a root mean square of 2.1 m. Although it is possible to achieve a good accuracy two major problems appear when this method is used. The first is that if the method is not initialized with a good starting point it is possible not to achieve the global maxima. The second one is that if data measurements deviate from the statistical model assumed, the results obtained may not be optimal.

6.3.2 Distributed Algorithms

In the distributed algorithm each node processes all the data that it collects from the network. They themselves are responsible for estimating their own coordinates. This is possible because nodes share their position information. A distributed algorithm is usually considered more efficient, in terms of computational cost, and scalable. On the other hand, distributed algorithms have lower accuracy compared to that achieved with a centralized algorithm due to the fact that the calculus is done at the nodes, which have less computational capacities.

6.3.2.1 Lateration

Instead of using angles to estimate the position, lateration methods use distances. These methods compute the nonlocated node positions by using the estimated distances to reference nodes nearby. The node position is obtained by means of calculating the intersection point of the circles with radius equal to the estimated distances and centred at reference node positions (see Figure 6.4). This technique is significantly affected by the errors on the distance measurement (ranging). Once a nonlocated node estimates its position, it becomes a new reference node that will help the rest of the nodes, giving to nonlocated nodes the possibility of exchanging information with other nonlocated nodes. Hence a lateration method can be considered as a cooperative localization algorithm.

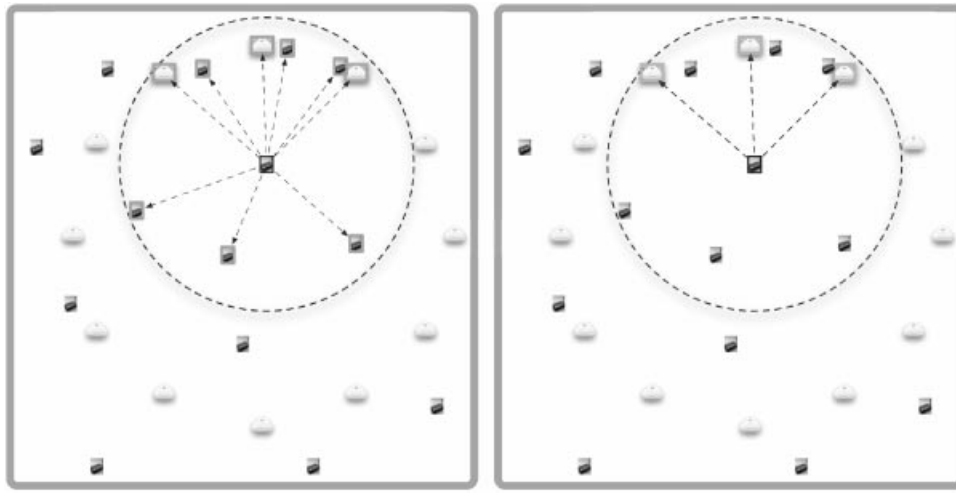

Figure 6.3 Cooperative and non-cooperative approaches

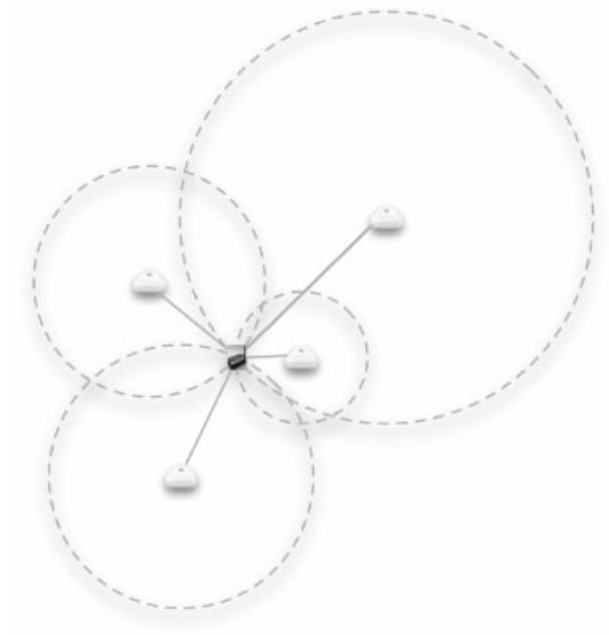

Figure 6.4 Lateration example.

6.3.2.2 Non-Bayesian Estimators

The non-Bayesian estimators [29] are one of the categories inside the distributed algorithms. The basic idea is the minimization of a cost function, such as LS (least squares) or ML (maximum likelihood), in a distributed way. Each node estimates its own position following a three-step algorithm. The procedure starts with the distribution of their coordinates to their neighbours. Then each node estimates the internode distance, by means of using range metrics, with those nodes from which the node has received their coordinates. Finally, nodes recalculate their position estimates by means of using both data. These steps are repeated until a convergence. The problem with these methods is the possibility of not converging to a global minimum. This fact is related to the use of a good starting point. If an initial point near to the final solution is not used the algorithm could not converge to a global solution.

The cost functions of both methods are

$$C_{LS}(x) = \sum_{i=1}^{N_2} \sum_{j \in S_i} \left\| Z_{j \to i} - f\left(\mathbf{x}_i - \mathbf{x}_j\right) \right\|^2 C_{ML}(x) = \sum_{i=1}^{N_2} \sum_{j \in S_i} \left(\ln\left(\frac{z_{j \to i}}{f(\mathbf{x}_i - \mathbf{x}_j)} \right) \right)^2 \qquad (6.3)$$

where $z_{j \to i}$ is the distance estimate between node i and node j and $f(\boldsymbol{x}_i - \boldsymbol{x}_j)$ is a function that estimates the distance between node i and j using the coordinates of them.

The major difference between both methods is that the ML approach takes advantage of the statistics of noise sources and the LS approach does not. A comparison between both methods is shown in Reference [30]; more concretely, the LS approach is a weighted one. All results presented show that the ML approach achieves better accuracy in terms of positioning errors. The mean error values obtained with the ML algorithm are between 1.2 and 1.5 m in comparison to that obtained with a WLS algorithm (values between 1.4 and 1.75 m).

6.3.2.3 Bayesian Estimators

Other methods that are used in the localization algorithms are those that are based on the Bayesian estimators. These methods have been firstly used in localization algorithms for robots [31]; nowadays many works have used them in sensor localization and tracking algorithms.

The basic idea is that, given sensors measurements $\boldsymbol{z}$, what is the probability of being at position $\boldsymbol{x}$ ($p(\boldsymbol{x}|\boldsymbol{z})$)? This posterior density over the random $\boldsymbol{x}$ conditioned to all received measurements $\boldsymbol{z}$ is usually called belief [25]. By means of computing the belief it is possible to obtain the position estimate. The authors present different algorithms of obtaining the position estimates using a Bayesian approach such as the Kalman filter or particle filter. Results show that both methods achieve good accuracy but in terms of complexity the particle filter implementation is the best option.

In Reference [29], the authors present a distributed approach called factor graphs (FGs), which is a successive refinement method used to estimate the probability density of sensor network parameters. Each sensor initializes their belief. Then, all nodes broadcast their beliefs to neighbours. Each node updates their belief with their own belief and the information extracted from the neighbour beliefs. Hence, nodes iteratively refine their position. Results obtained in Reference [29] show that the distributed approach achieves similar results obtained with a centralized approach. Results presented show that 90% of the nodes achieve an error below 1 m, as compared to the results achieved by the non-Bayesian LS algorithm in which only the 40% achieve error values below 1 m.

6.4 RSS-Based Cooperative Positioning

Several methods have been presented in previous sections. The first classification is between range-free and range-based methods. Both methods are suitable to be used in a localization algorithm for a WSN. However, the choice between one or another is based on the requirements of the final application. In order to achieve a good trade-off between accuracy and cost one of the most appropriate options is an RSS-based range measurement. RSS-based methods present less accurate measurement results but they are the simplest methods that could be applied in a WSN.

A cooperative approach is most suitable for large-scale networks, compared to a noncooperative approach. The accuracy of the noncooperative methods is highly dependent on the density of the anchor nodes. Also a high density of neighbour nodes, obtained with a cooperative approach, gives more robustness to the localization algorithm.

Finally, the last choice is between a centralized and a distributed algorithm. As discussed before, centralized approaches give a higher accuracy due to the possibility of developing a more complex algorithm. Distributed approaches need to be able to be computed in each node so they have to be as simple as possible. On the other hand, a centralized approach needs a higher traffic exchange, because all data has to be sent to the central node, which limits the capability of scaling the network.

In order to start the discussion of an RSS-based cooperative algorithm let us consider a wireless sensor network with N nodes. There are N_1 nodes, whose exact locations are known (anchor nodes). The rest of the nodes, $N_2 = N - N_1$, do not know their position (nonlocated nodes). Those algorithms are normally divided into two steps. The first one is the measurement phase in which the algorithm uses some range measurement in order to obtain distance estimates. The second one is the location update phase, in which by means of using the estimates obtained at the first phase and the node state information, the algorithm computes the position estimates.

6.4.1 Measurement Phase

As commented above, this chapter focuses on the RSS-based cooperative approach. In this kind of algorithm the first phase of the algorithm consists in obtaining internode distances, in this case by means of RSS measurements.

RSS-based measurements are a very attractive ranging method for practical implementation because they do not need extra hardware to be measured. The most common sources of error that affect RSS-based distance estimations are shadowing and multipath signals, which complicate the modelling of the channel that nodes need to know a priori. Usually, RSS measurements are modelled through the well-known radio-propagation path loss and shadowing model [19]. Received power is modelled as a lognormal distributed random variable with a distance-dependent mean. Hence, power received in node j from a signal transmitted by node i, P_{ij}, is expressed as

$$RSS_{ij} = P_{ij} = P_0 = 10\alpha_{ij} \log_{10} d_{ij} - \upsilon_{ij} \tag{6.4}$$

where P_0 is the power received in dBm at 1 m distance, d_{ij} is the distance between nodes i and j in metres, parameter α_{ij} is the path loss exponent, that is the rate at which the power

decreases with distance, and $\nu_{ij} \sim N(0, \sigma_v^2)$ represents lognormal shadow-fading effects, where the value of the standard deviation σ_v depends on the characteristics of the environment. The small-scale fading effects are diminished [19] by time averaging; hence they do not affect the distribution of ν_{ij}. Since static scenarios are considered, the major sources of error are shadowing and path loss.

In Reference [20], the authors discuss that the lognormal distribution is often used to explain the large-scale variations of the signal amplitudes in multipath fading environments. References inside Reference [20] present the validity of this model for modelling an indoor radio channel. Some results show that the lognormal fits better than the Rayleigh model. Furthermore, large-scale variations of data collected at 900 MHz, 1800 MHz and 2.3 GHz for transmission into and within buildings were found to be lognormal.

Given the received power RSS_{ij} in Equation (6.4), the density of P_{ij} is [32]

$$f_{P|\gamma}(P_{ij}|\gamma) = \frac{\frac{10}{\log 10}}{\sqrt{2\pi\sigma_{dB}^2}} \frac{1}{P_{ij}} \exp\left(-\frac{1}{8} \left(\frac{10\alpha}{\sigma_{dB} \log 10} \right)^2 \left(\log \left(\frac{d_{ij}^2}{d_0 \left(\frac{P_0}{P_{ij}} \right)^{\frac{2}{\alpha}}} \right) \right)^2 \right) \tag{6.5}$$

It is worth noting that P_0 and P_{ij} are not expressed in dBm. Hence, an ML estimate of the distance d_{ij} could be derived as [32]

$$\delta_{ij} = 10^{\frac{P_0 - RSS_{ij}}{10\alpha_{ij}}} \tag{6.6}$$

An important result of the lognormal model is that RSS-based distance estimates have variance proportional to their actual range [9]. The standard deviation in decibels is considered constant with range. This consideration means that the multiplicative factors are constant with range; hence, this explains the multiplicative error present in RSS-based distance estimates.

6.4.2 Location Update Phase

Once the relative distances between nodes are obtained, the main goal is to estimate the location of the nonlocated nodes with the help of anchor nodes and the rest of the nodes in the network.

The position estimates for each nonlocated node are obtained by means of the least squares (LS) criterion. The localization algorithm has to obtain the set of nonlocated node positions that minimize the difference between the estimated distances at the first phase and the distances computed using such position estimates. In particular, the problem consists in minimizing the following cost function:

$$C_{LS}(x) = \sum_{i=1}^{N_2} \sum_{j \in S_i} \left(\delta_{ij} - d_{ij}(\mathbf{x}_i - \mathbf{x}_j) \right)^2 \tag{6.7}$$

where $d_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) = ||\boldsymbol{x}_i - \boldsymbol{x}_j||$ is the distance between nodes i and j, calculated with the estimated position (or real coordinates if node j is an anchor) of nodes i and j, S_i is the group of

nodes (anchor and nonlocated) that cooperates in the position estimation of nonlocated node i and $\mathbf{x}$ are the coordinates of the nodes.

The cost function is minimized with the optimization of the node coordinates. The minimization will be obtained by means of calculating the derivative of Equation (6.7) with respect to $\mathbf{x}_i$:

$$\frac{\partial C_{LS}}{\partial x_i} = \sum_{j \in S_i} \frac{\left(\delta_{ij} - d_{ij}\right)^2}{\partial x_i} + \sum_{k \in S_i} \frac{\left(\delta_{ki} - d_{ki}\right)^2}{\partial x_i} \tag{6.8}$$

Wireless channels are usually not considered reciprocal. For that reason measurements of the second summation should be omitted. As a result, the cost function adopted by each node can be rewritten as

$$C_{DLS}(x_i) = \sum_{j \in S_i} \left(\delta_{ij} - d\left(\mathbf{x}_i - \mathbf{x}_j\right)\right)^2 \tag{6.9}$$

A distributed cost function is found, so each node is responsible for obtaining the minimization of this cost function. Many methods could be applied in order to obtain this minimization. A gradient descent is one of the simplest approaches that have a lower computational complexity. Hence, the distributed cost function in Equation (6.9) is iteratively minimized. These algorithms have the possibility of not reaching a global minimum when a good starting point is not used. Even so, it is a simple method with a low computational complexity.

The gradient of the cost function is

$$\nabla_{x_i} C_{DLS}(x_i) = \nabla_{x_i} \left(\sum_{j \in S_i} \left(\delta_{ij} - \left\| x_i - x_j \right\| \right)^2 \right) = \sum_{j \in S_i} \left(\delta_{ij} - \left\| x_i - x_j \right\| \right) e_{ij} \tag{6.10}$$

where $e_{ij} = (x_i - x_j / \| x_i - x_j \|)$ is the unit vector that takes the orientation between the node i and node j. So, the estimate of $\mathbf{x}_i$ can be iteratively computed by using the gradient descent algorithm as follows:

$$\hat{x}_i(t+1) = \hat{x}_i(t) + \gamma \sum_{j \in S_i} \left(\delta_{ij} - d_{ij}\right) e_{ij} \tag{6.11}$$

where γ is the step length factor.

This algorithm becomes a simple, low computational approach that obtains position estimates in a cooperative and distributed way by means of using RSS measurements.

In Figure 6.5, a WSN scenario of 50 m × 50 m with 24 anchor nodes and 30 nonlocated nodes is considered. The mean absolute error obtained in the position estimates as a function of the number of cooperating nodes is presented. As commented previously, the higher the number of cooperating nodes, the lower the error obtained. However, it is remarkable that the error is not monotonically decreasing and the error is saturated for high values of cooperating nodes. This is basically due to the effect commented previously: having a further node causes a higher error in a distance estimate. On the other hand, there also exists a relation with the presence of more anchor nodes cooperating in the estimation of coordinates. For that reason, in the following section a node selection mechanism is presented to limit the number of

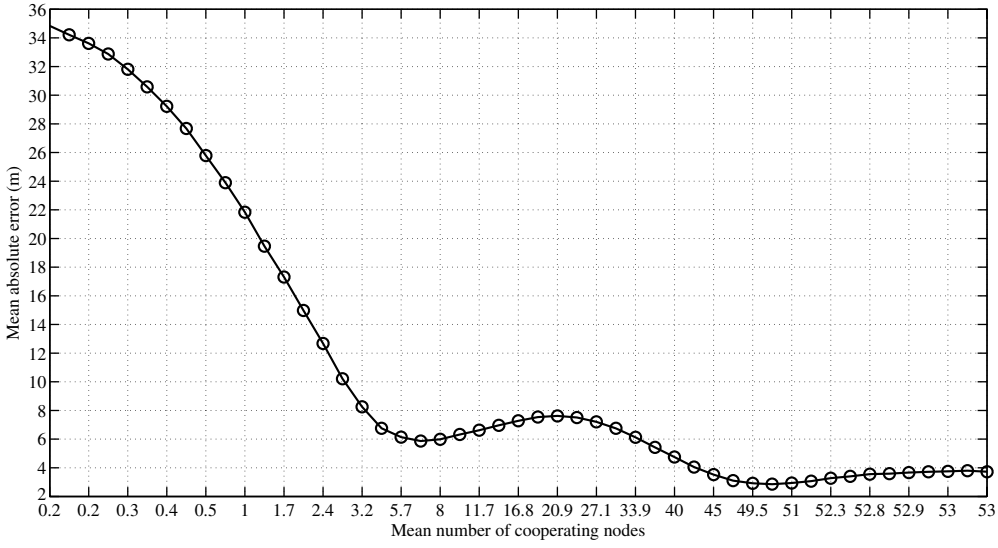

Figure 6.5 Mean absolute error versus mean number of cooperating nodes (24 anchor nodes).

cooperating nodes by selecting those nodes that provide an appropriate error value in terms of position accuracy. By doing so, the number of cooperating nodes, those within group S_i, is reduced and energy efficiency can be improved. For example, the difference in error obtained with a number of cooperating nodes equal to 6 and 50 is minimal but the number of cooperating nodes is considerably lower; that is energy consumption can thus be significantly reduced.

6.5 Node Selection

A cooperative approach allows the algorithm to achieve better accuracy results. On the other hand, the computational cost is increased because nodes receive more information from more cooperating nodes. Furthermore, a higher packet exchange in the network will be necessary; hence there exists a higher probability of losing a packet due to a collision.

Considering also that this chapter focuses on RSS-based distance estimates, one needs to take into account that the error that suffers these estimates is multiplicative to distance. Having a cooperative approach allows each node to cooperate with more nodes that could be further away. Hence, the distance estimates that a node would use should have a higher error. It is also shown in Figure 6.5 that reducing the number of cooperating nodes does not seriously affect the accuracy of the localization algorithm.

In order to reduce the use of a great number of cooperating nodes, that is having a lower number of nodes inside each group S_i, the node selection mechanism is presented. The major purpose behind the node selection is to reduce the packet exchange inside the network, and thus the reduction of computational effort done by each node and the energy consumption of them. Some works have presented different approaches in order to select those nodes that will cooperate with each nonlocated node.

In Reference [33], the authors present a noncooperative microgenetic algorithm (MGA) in order to select nodes and improve the localization in a WSN. The adaptation of the MGA presented is based on three steps: firstly, the construction of a small population of chromosomes based on the best values of position estimation; secondly, a genetic operator called a descend-based mutation is applied; and, finally, a second genetic operator called a crossover operator is used. The basic idea is to first select best nodes (done at the first step) and then apply both genetic operators to the chromosomes (aka nodes) in order to converge to a final solution. The results obtained are, in mean, 0.2 times the node range R.

In Reference [34] the authors propose to select nodes by means of using the Crámer–Rao bound (CRB) instead of using the closest nodes. The algorithm calculates the CRB of all the reference nodes that a node receives and selects those nodes with a lower CRB. Results obtained show that with a number of eight reference nodes the location error is, approximately, 7 m if RSS-based distance measurements are used. On the other hand, 0.8 m of error is achieved if TOA-based distance estimates are used.

In Reference [35], the authors present a censoring method based, also, on the CRB. The algorithm censors those nodes with an unreliable estimation. Based on the CRB calculation, the authors propose a criterion that will reflect the quality of the information that a node transmits as well as the geometry of the positions of the anchors and nonlocated nodes. Three methods of censoring are presented in the work: the first one is the one in which a node can censor itself, that is each node can decide not to broadcast its own information; the second one blocks the reception of information from the neighbours considered not reliable; and, finally, the last one is created to avoid an unnecessary transmission when a node is censored by all its neighbours, that is a node receives the order not to transmit because all its neighbours have censored it.

All these censoring methods are executed through the calculation of the CRB, and by comparing it with a threshold imposed by the algorithm. With the inclusion of all these censoring methods, the authors obtain a reduction in complexity and in network traffic, while the position accuracy is maintained.

All methods achieve a high reduction of messages exchanged in their results. A comparison between distance-based selection and CRB-based selection is done in Reference [34]. On the one hand, better results are achieved when the CRB is used to select cooperating nodes; for example differences between 2.5 m with three cooperating nodes and 0.3 m with 10 cooperating nodes are presented in the location error. On the other hand, the distance-based selection does not require any extra calculation at the time of deciding which nodes are the best to cooperate.

Finally, a node selection least squares (NS-LS) location algorithm is presented in Reference [36]. The idea of obtaining a good trade-off in terms of position accuracy versus energy consumption is maintained. As discussed in Reference [36], the derivation of the optimal selection criterion is not possible. For that reason, the authors presented a suboptimal scheme based on the received power threshold (RSS_{th}). In other words, only nodes with an RSS higher than RSS_{th} are allowed for cooperation. In a simple scheme this criterion becomes suitable for a hardware restricted WSN. In particular, the choice of the RSS_{th} value was designed to ensure a minimum number of anchor nodes inside the cooperating node group S_i. In accordance with this value N_m, different trade-off points of energy consumption versus accuracy can be achieved. Results showed that $N_m = 3$ allows the algorithm to achieve an excellent trade-off. Concerning the relation between RSS_{th}

and N_m, the authors in Reference [36] derived an analytical procedure to obtain the required threshold that assures the desired value of N_m. Considering a uniform distribution of the nonlocated nodes, the mean number of anchor nodes inside a circumference of radius r_{th} is obtained as

$$N_m \approx \sum_{j=1}^{N_1} \pi r_{th}^2 \frac{1}{A} = \frac{N_1 \pi r_{th}^2}{A} \tag{6.12}$$

where A is the total area of the considered scenario and N_1 is the total number of anchor nodes in the scenario. An appropriate received power threshold (RSS_{th}) must be selected in order to ensure that a number N_m of anchor nodes are inside the coverage range radius. Taking r_{th} as the coverage range radius, a relation between the RSS received at a distance r_{th} is obtained by means of using the path loss propagation model. Hence, a relation between RSS_{th} and N_m could be established by means of

$$RSS_{th} = \left(\frac{N_1 \pi P_0^{2/\alpha}}{N_m A} \right)^{\alpha/2} \tag{6.13}$$

With this relation it is possible to fix a threshold that assures a mean number of anchor nodes. With this threshold nodes are able to discard those nodes that have an RSS below this value; hence the algorithm is able to reduce the number of cooperating nodes.

The algorithm achieves worse results in terms of accuracy (a degradation between 0.23 and 0.44 m). On the other hand, the reduction achieved in terms of energy consumption is above 78% compared to a nonselection algorithm.

6.5.1 Energy Consumption Model

Wireless sensor network nodes rely on low data rates, a very long battery life (several months or even years) and very low computational complexity associated with the processing and communication of the collected information across the WSN. In order to maintain the battery life, the reduction of the energy consumption is an important point in a WSN.

The energy consumption of a node is directly related to the number of messages transmitted or received. The purpose of using node selection mechanisms is to reduce the packet exchange between nodes inside the network. The reduction of packet exchanges becomes an important function in the reduction of energy consumption. Nodes only interact with their group of cooperating nodes. Hence, the energy consumed will depend on the required number of transmissions. An energy consumption model is presented in order to reflect the effects produced in the network.

Each node i needs to create its own group of cooperating nodes S_i (see Figure 6.6). At the beginning, each node i sends a broadcast message with its coordinates $\mathbf{x}_i$. Only those nodes that receive this message answer with their node id and their location coordinates. With the received messages, each nonlocated node can create its own S_i group. Once these groups are created, the exchange of messages is only done between cooperating nodes. At this time, the total amount of energy consumed by the network follows the

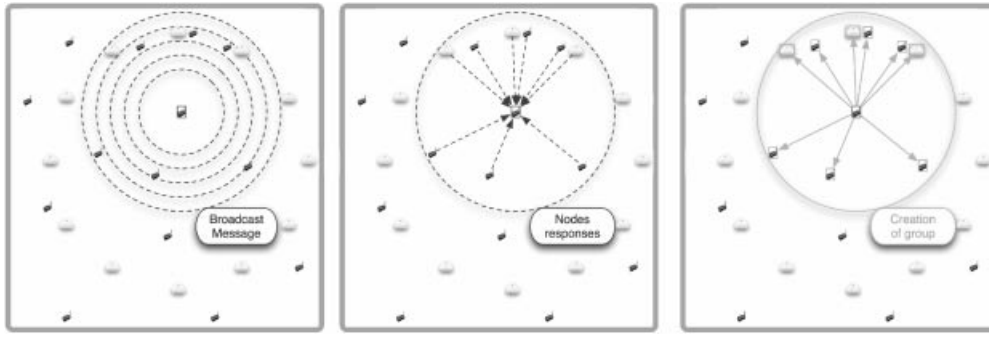

Figure 6.6 Creation of group S_i.

model presented in Reference [37]:

$$\varepsilon = \left(\mu_{R_x} - \mu_{T_x}\right)\left(\sum_{i=1}^{N_2} N_{S_i} - N_2\right)\kappa \tag{6.14}$$

where κ is the number of iterations of the algorithm, N_{S_i} is the number of nodes inside S_i, and μ_{T_x} and μ_{R_x} (e.g. a value equal to 400 nJ/s is used in Reference [37]) are the energy consumption values dedicated for peer to peer transmission and reception procedures, respectively. It is noticeable that this model presents some differences compared with the model presented in Reference [37]. It is supposed that the energy per transmission is always the same instead of having an energy consumption depending on time. The purpose behind a node selection mechanism is the reduction in the traffic. Hence, the energy consumption model presented only reflects the impact that selection mechanisms could produce. Furthermore, it has only taken care of the energy consumption at the transmission and reception times.

Energy consumption is an increasing function of the number of cooperating nodes (N_{S_i}). The introduction of a node selection mechanism reduces the number of cooperating nodes; hence the energy consumption will decrease.

In the following subsection two low complex mechanisms suitable to be used in a WSN localization algorithm are proposed. Later, in Section 6.5.3, the joint path loss and node selection localization algorithm is proposed. The different blocks that form the proposed online path loss and node selection localization algorithm are presented in detail.

6.5.2 Node Selection Mechanisms

Which is the best criterion to use in order to select the nodes that will cooperate in the location-update phase? With this question in mind, and taking a look at existing methods presented at the beginning of the section, two node selection criteria are presented.

The continuous idea that is maintained through the entire chapter has an important purpose: to present a low complex localization algorithm. For that reason, two node selection criteria are presented. The introduction of a node selection should not suppose an increase in the computational complexity of the algorithm. Both node selection mechanisms are related by the fact that a node can do without any extra measurement requirements.

In order to take the advantage of these measurements, both node selection mechanisms will depend on the path loss estimates obtained by the location algorithm. With those selection mechanisms a reduction in the number of nodes that cooperate in the location algorithm, as well as a reduction in energy consumption of the network, is achieved.

6.5.2.1 Low Path Loss Selection

The first selection criterion is the selection of those nodes that have the lowest values of the path loss exponent estimated by the localization algorithm. The idea behind this node selection mechanism is to select those nodes that have the best channel conditions. When the path loss exponent takes a high value the conditions of propagation are worse.

The method works as follows. Given all the estimates of $\hat{\alpha}_{ij}$, the first selection mechanism selects those nodes that have the lowest values for the path loss exponents of the nodes inside the coverage of node i:

$$\hat{\alpha}_{i1} \leq \hat{\alpha}_{i2} \leq \cdots \leq \hat{\alpha}_{in}, \qquad 1, \ldots, n \in S_i$$

where $\hat{\alpha}_{i1}$ and $\hat{\alpha}_{in}$ are the lowest and the highest exponents, respectively. Those nodes with the lowest values are selected:

$$S_i^{NS} = \{i1, i2, \ldots, in_\alpha\}$$

with n_α standing for the number of selected nodes.

6.5.2.2 Low Distance Selection

The second selection criterion selects those nodes with lower values of distance estimates, that is those nodes closer to the nonlocated nodes. RSS-based distance estimates have an error multiplicative to the distance; hence the selection of closer nodes will choose those nodes with a lower distance estimate. The higher the distance, the higher the error of the distance estimates. For that reason the mechanism tries to reduce this effect by selecting the closest nodes to the nonlocated node i.

Given now all the distance estimates of the nodes inside the coverage of node i:

$$\delta_{i1} \leq \delta_{i2} \leq \cdots \leq \delta_{in}, \qquad 1, \ldots, n \in S_i,$$

where in this case δ_{i1} and δ_{in} are the lowest and highest distance estimate, respectively. The new group of cooerating nodes becomes

$$S_i^{NS} = \{i1, i2, \ldots, in_\delta\}$$

with n_δ standing for the number of selected nodes.

6.5.2.3 Selection Mechanisms Performance

In Figure 6.7, an example of both node selection mechanisms is shown. Both nodes could not use nearer nodes; hence they have the possibility of having a higher error on distance

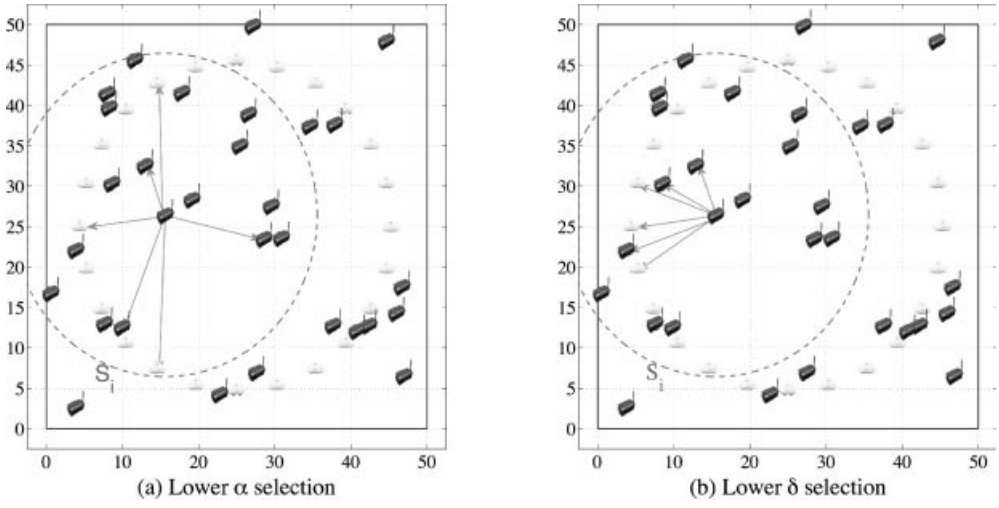

Figure 6.7 Example of both node selection methods.

estimates. The only mechanism that could have a higher probability of selecting closest nodes is the lower distance criterion. Only with perfect channel knowledge is it possible to achieve a perfect selection of the closest nodes.

Two important points now have to be discussed: which of the suitable methods is better to use and how many nodes has to be used in order to obtain the best results?

Selection of the Criterion

The first results of the performance are presented in Figure 6.8. Both methods are used in the same scenario and for different numbers of nodes for n_α and n_δ. Low path loss selection selects those nodes with the best-estimated channel conditions. The major problem is that these nodes could be further away from the node *i*. As RSS-based range measurements are

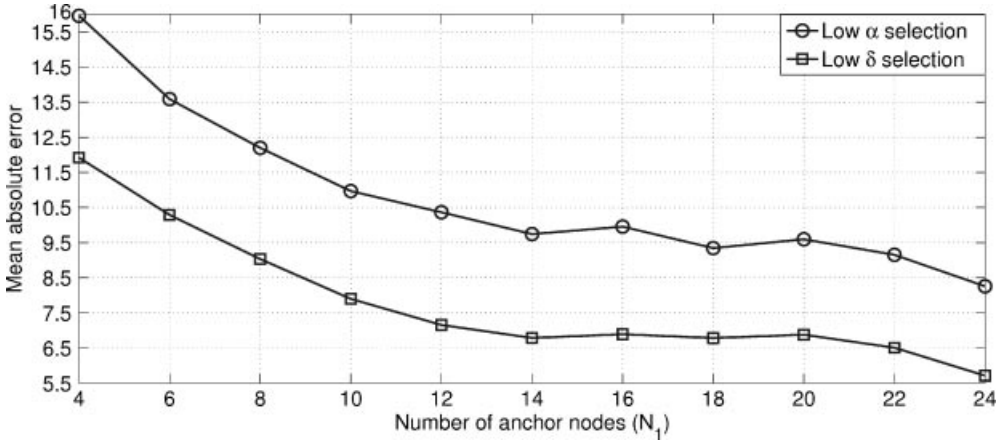

Figure 6.8 Mean absolute error versus $N_1(n_\alpha = n_\delta = 6)$.

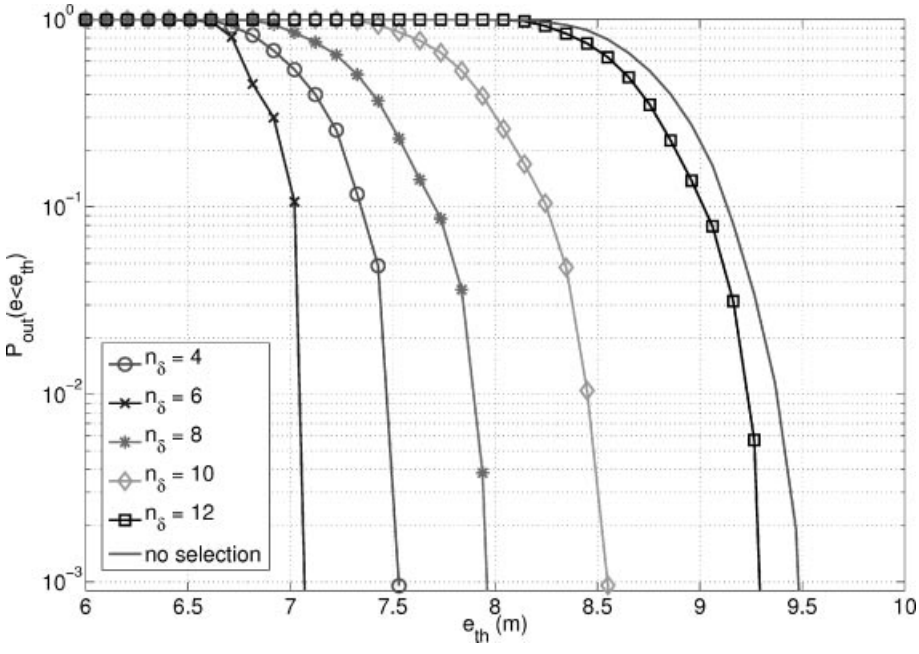

Figure 6.9 Outage probability ($N_1 = 18$).

used, the error in the distance estimates becomes higher. The probability of achieving a worse accuracy is increased, as reflected in the results shown. The low distance selection achieves better results in terms of accuracy.

Selection of the Number of Cooperating Nodes

This is the second important point when a node selection mechanism is included in the localization algorithm. In the previous presented works both methods has defined how a node collaborates or not in the location-update phase. In the first work they only allow cooperation to those nodes that are above a CRB threshold. In the second work, authors have analytically extracted a way in which they assure, in mean, that at least three cooperating nodes are anchors.

Simulation results are presented in order to decide which is the suitable number of cooperating nodes, because it is not straightforward to obtain the value of cooperating nodes that optimize the system behaviour. In principle, the optimum number of n_δ is scenario dependent. In the scenario considered here, results presented in Figure 6.9 show that a value equal to 6 is generally the best choice. This means that actually the optimum n_δ does not depend on the fine-grained distribution of the nodes, but rather on general parameters of the scenario.

6.5.3 Joint Node Selection and Path Loss Exponent Estimation

During the entire chapter the different blocks of a localization algorithm have been presented. A presentation of an RSS-based distributed algorithm has been done at the beginning, followed with the introduction of a node selection mechanism in order to reduce the cooperating nodes. Moreover, obtaining RSS-based distance estimations depends on the knowledge

of a transmission model. This model is usually obtained with a previous measurement campaign. However, the idea of repeating a measurement campaign in every possible scenario in which the algorithm has to work is not a good decision. For that reason, an on-line path loss estimation is introduced in order to dynamically estimate the transmission model and allow the algorithm to estimate the model that best fits the current scenario. In this subsection is presented all the steps that the localization has to do in order to obtain position estimates. Algorithm 6.1 presents the different stages of the node localization algorithm with an on-line path loss estimation. A more in-depth explanation of the different blocks of the algorithm present in Algorithm 6.1 is presented in the following subsections.

Algorithm 6.1 LS Localization Algorithm with On-Line Path Loss Estimation

Discovering Cooperating Group S_i:
node i transmits a broadcast message
nodes able to receive respond with their id and their coordinates
Previous Coordinate Estimation:

for $i = 1$ to N_2 **do**

$\hat{\mathbf{x}}(t=1) = \sum_{a=1}^{n_{anch}} \mathbf{x}_a \left| \frac{RSS_{ia}}{\sum_{a=1}^{n_{anch}} RSS_{ia}} \right|$

end for
On-Line Path Loss-Node Selection-Least Squares algorithm
for $i = 1$ to t_{iter1} **do**
Coordinate Estimation:
for $t(= 1$ to t_{iiCT2} **do**
for $i = 1$ to N_2 **do**

$\hat{\mathbf{x}}_i(t) = \hat{\mathbf{x}}_i(t-1) + \gamma_x \sum_{j \in S_i} \left(\delta_{ij} - \hat{d}_{ij} \right) \mathrm{e}_{ij}$

end for
end for
Path Loss Estimation:
for $t' = 1$ to t_{iter2} **do**
for $i = 1$ to N_2 **do**
for $j = 1$ to N_{Si} **do**

$\hat{\alpha}_{ij}(t) = \hat{\alpha}_{ij}(t-1) - \gamma_\alpha \log(10) \frac{P_0 - RSS_{ij}}{5} \frac{1}{\alpha_{ij}^2} \delta_{ij} \left(\delta_{ij} - \hat{d}_{ij} \right)$

$\delta_{ij} = 10^{\frac{P_0 - P_{ij}}{10\hat{\alpha}_{ij}}}$

end for
end for
Low Distance Node Selection
if $t == 1$ **then**

$S_i = \{i1, i2, \ldots, in_\delta\}$ with $\delta_{i1} < \delta_{i2} < \cdots < \delta_{in\delta} < \cdots < \delta_{i_{N_{S_i}}}$

$N_{S_i} = n_\delta$

end if
end for
end for

6.5.3.1 Discovering Cooperating group S_i

The first necessary step is to discover which nodes are inside the radio range of each nonlocated node i. In order to do that, each nonlocated node broadcasts a message with their node id. Nodes that are able to receive this message send an answer to the nonlocated sender, with their node id and their position. As a result, each node can form its own group and can know the number of anchor and nonlocated nodes that are inside its radio range.

6.5.3.2 Initial Coordinates Estimation

The next step is giving an initial value to the position estimates of the nonlocated nodes. As discussed before, a gradient descent approach could not converge to the global solution if a biased initial value is used. Hence, it is important to give a good starting point in order to converge to a global solution. In that sense many works present different options. The authors of Reference [38] present the idea of combining two approaches: obtaining a previous estimate by means of using an MDS algorithm and then implementing a refinement algorithm, such as an ML non-Bayesian approach. On the one hand, this solution obtains good results in terms of accuracy. On the other hand, this method is more complex compared to other solutions. Moreover, it is possible to give a random initialization to all nonlocated nodes. It is a low complex but inaccurate method. Following with the general purposes presented previously, the starting point procedure has to attain a good trade-off between accuracy and complexity.

A simple method to initialize each $\hat{\boldsymbol{x}}_i(0)$ is the use of a centroid method. Once each node forms its own cooperating group S_i, all the nonlocated nodes are able to compute its initial position with a weighted centroid algorithm based on the use of anchors inside S_i. The computation becomes

$$\hat{\boldsymbol{x}}_i(t=0) = \sum_{a=1}^{n_{anch}} \boldsymbol{x}_a \left| \frac{RSS_{ia}}{\sum_{a=1}^{n_{anch}} RSS_{ia}} \right| \tag{6.15}$$

where $\mathbf{x}_a$ are the coordinates of the anchor $a \in S_i$.

It is possible to achieve a closer initial point with a weighted centroid algorithm; hence a better performance of the gradient descent approach could be achieved. Although the centroid algorithm presents low accuracy it is a simple method and is only used to obtain the initial coordinates that will be later refined with a non-Bayesian LS method.

6.5.3.3 On-line Path Loss Estimation and Node Selection Least Squares Algorithm (OLPL-NS-LS)

The introduction of this estimation is motivated by the necessity of having a good propagation model in order to have accurate distance estimates. Usually, the propagation model is achieved by doing a previous measurement campaign. With the introduction of an on-line path loss estimation, the localization algorithm is responsible for obtaining the node coordinates and the path loss estimation; hence the necessity of previous modelling is avoided. The next step presented in Algorithm 6.1 is the minimization of the LS cost function:

$$C_{DLS}(\boldsymbol{x}, \alpha_{i1}, \alpha_{i2}, \ldots, \alpha_{in_\alpha}) = \sum\nolimits_{j \in S_i} \left(\delta_{ij}(\alpha_{ij}) - d_{ij}(\boldsymbol{x}_i - \boldsymbol{x}_j) \right)^2 \tag{6.16}$$

The objective is to minimize the difference between both distances and optimize the node coordinates and also the set of path loss exponents. The node coordinates $(\mathbf{x}_i)$ and the set of path loss exponents $\left(\alpha_{ij} \forall j \in S_i\right)$ affect the computation of both distances, d_{ij} and δ_{ij}, respectively. In order to solve the cost function of Equation (6.16), a Gauss–Seidel algorithm is adopted in Reference [39]. This nonlinear algorithm is based on a circular iterative optimization with respect to one set of variables while maintaining the rest of the variables fixed. Hence, the minimizations are carried out successively for each component.

Considering a generic cost function F that depends on a set of variables β, the desired minimization of F is formally defined as [39]

$$\beta(t+1) = \arg\min_{\beta_i} F\left(\beta_1(t+1), \ldots, \beta_{i-1}(t+1), \beta_i, \beta_{i+1}, \ldots, \beta_m(t)\right) \tag{6.17}$$

At the time instant $t+1$, the F function is minimized by means of optimizing the β_i component. Components between β_1 and β_{i-1} have already been optimized while components from β_{i+1} to β_m (m being the total number of components) have not yet been optimized. These last components must remain constant while the other components are being optimized. By using the Gauss–Seidel approach, it is possible to divide the optimization into two steps: firstly, a minimization of the cost function by means of optimizing the node coordinates (fixing the path loss exponents) could be carried out; secondly, another minimization is done by means of the optimization of the path loss exponents (fixing the nodes coordinates). As the convergence of the nonlinear Gauss–Seidel algorithm can be achieved using a descent approach (see Reference [9]), both minimizations could be carried out through a gradient descent mechanism. The basic idea is summarized in Figure 6.10.

Coordinates Estimation

The estimation is done following the algorithm described in the previous section. The algorithm minimizes, by means of a least squares criterion, the difference between the estimated distance and the distance calculated with node coordinates. With the use of a gradient descent approach the algorithm is able to converge to a minimum of the cost function presented before.

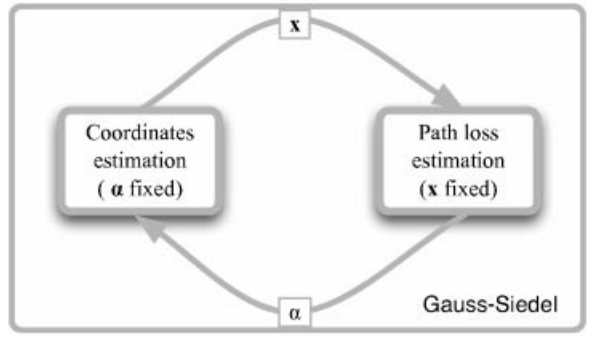

Figure 6.10 Optimization of the node coordinates and path loss exponents using a Gauss–Siedel algorithm.

Path Loss Estimation

This step is necessarily done after a previous position estimate. The algorithm has obtained the previous coordinates by means of using an arbitrary and equal value of the path loss exponent. At the second step, each nonlocated node estimates a path loss exponent for each link. Following the Gauss–Seidel approach, the cost function of Equation (6.16) is now minimized by means of optimizing the path loss exponents. Following the same methodology applied in Section 6.2.2, it is necessary to calculate the gradient of the cost function. The estimated distance

$$\delta_{ij} = f(\alpha_{ij}) = 10^{\frac{P_0 - RSS_{ij}}{10\alpha_{ij}}}$$

depends on the path loss exponent α_{ij}; hence, the cost function is minimized by means of calculating the derivate with respect to the path loss exponent of each individual link. In that case, the fixed variables are the coordinate estimates and the rest of the path loss exponents $(\alpha_{ik} \forall k \neq j)$. The gradient of the cost function is

$$\begin{aligned} \nabla_{\alpha_{ij}} C_{DLS}(x_i, \alpha_{S_i}) &= \nabla_{\alpha_{ij}} \left(\left(10^{\frac{P_0 - RSS_{ij}}{10\alpha_{ij}}} - d_{ij}(x_i, x_j) \right)^2 \right) \\ &= \log(10) \frac{P_0 - RSS_{ij}}{5} \frac{1}{\hat{\alpha}_{ij}^2} \delta_{ij} \left(\delta_{ij} - \hat{d}_{ij} \right) \end{aligned} \tag{6.18}$$

Each node estimates their own path loss exponents for all the links in an iteratively fashion as

$$\hat{\alpha}_{ij}(t+1) = \hat{a}_{ij}(t) - \gamma_\alpha \log_{10}(10) \frac{P_0 - RSS_{ij}}{5} \frac{1}{\hat{\alpha}_{ij}^2(t)} \delta_{ij} \left(\delta_{ij} - \hat{d}_{ij} \right) \tag{6.19}$$

It is a distributed method that minimizes the cost function through an iterative gradient descent strategy. The algorithm is able to estimate the path loss exponents by means of using RSS measurements. The presented algorithm maintains the philosophy of having a low complex and low cost localization algorithm.

Low Distance Node Selection

Finally the node selection mechanism is applied. With this node selection the localization algorithm modifies the cooperating group formed at the beginning of the algorithm. Hence, the algorithm reduces the traffic exchange among the network, allowing the network to decrease the total amount of energy consumed. Based on the results shown in Figure 6.8, the algorithm selects n_δ number of nodes that have the lowest distance estimates. With this selection mechanism each cooperating group S_i is reduced.

6.6 Numerical Results

This section presents the performance of the presented location algorithm with an on-line path loss estimation and node selection. In order to evaluate the accuracy of the algorithm different simulation results are presented. The simulated scenario and the assumed simulation parameters are presented in Table 6.1.

Table 6.1 Simulation parameters

Simulation parameters	Parameter value
Size of sensor field	50 m × 50 m
Number of nonlocated nodes (N_2)	30
Path loss exponent α_{ij}	2–5
Standard deviation σ_υ	1 dB
First-metre RSS P_0	−50 dBm
Anchor radius	20.4 m
Energy consumption to transmit or receive μ_{Tx} or μ_{Rz}	400 nJ

Path loss exponents take values between a maximum value of 5 and a minimum value of 2 (the uniform distribution of the path loss exponents between 2 and 5 are based on experimental results obtained in Reference [40]). Hence, path loss values are simulated with a uniform distribution ($\alpha \in u(2,5)$). An initial value of the path loss equal to 3.5, which is the middle value of the random values used in the uniform distribution, is assumed. The experimental parameters are shown in Table 6.1.

Concerning the anchor node placement, the approach presented in Reference [41] is initially adopted, where the authors contend that the best anchor placement is a centred circumference with the radius equal to the root-mean-square (rms) of the nonlocated node distances to the centre.

As previously discussed, the use of a gradient descent algorithm requires a good starting point in order to achieve better results. The algorithm calculates the starting coordinates using a centroid method. The weighted centroid method depends on the number of anchor nodes inside the network.

In order to reflect the effect of using a great number of anchor nodes n_{anch}, the mean absolute error achieved by the localization algorithm is shown in Figure 6.11 for different values of anchor nodes used in the weighted centroid.

Better results are obtained when a number of anchor nodes equal to 1 is used. At first sight, this result could seem strange because, normally, it is better to use as many nodes as possible. However, at the initial time instant of our algorithm the path loss exponent values have still not been estimated. Hence, the higher RSS_{ij}, the lower the distance estimate δ_{ij}. If the number of anchor nodes n_{anch} has a greater value, the probability of having a further anchor node is increased. Hence, an initial position far away from the real position is estimated. Selecting only the anchor node with the highest RSS is the best option.

Although the distribution of the anchors in a centred circumference was demonstrated to be the best one in Reference [41], it could not be the optimal one when a centroid algorithm is used as the initial position estimate. The centroid method depends on the distribution of the anchors inside the network. For that reason, a grid-based positioning is proposed as an alternative.

The results achieved are shown in Figure 6.12(b). On this occasion it is necessary to use more anchor nodes in order to obtain better results. The centroid method achieves better results when the anchors are placed in a grid-based distribution. Hence, the starting point used in the LS algorithm will be closer to the final solution. In this case, when the centroid

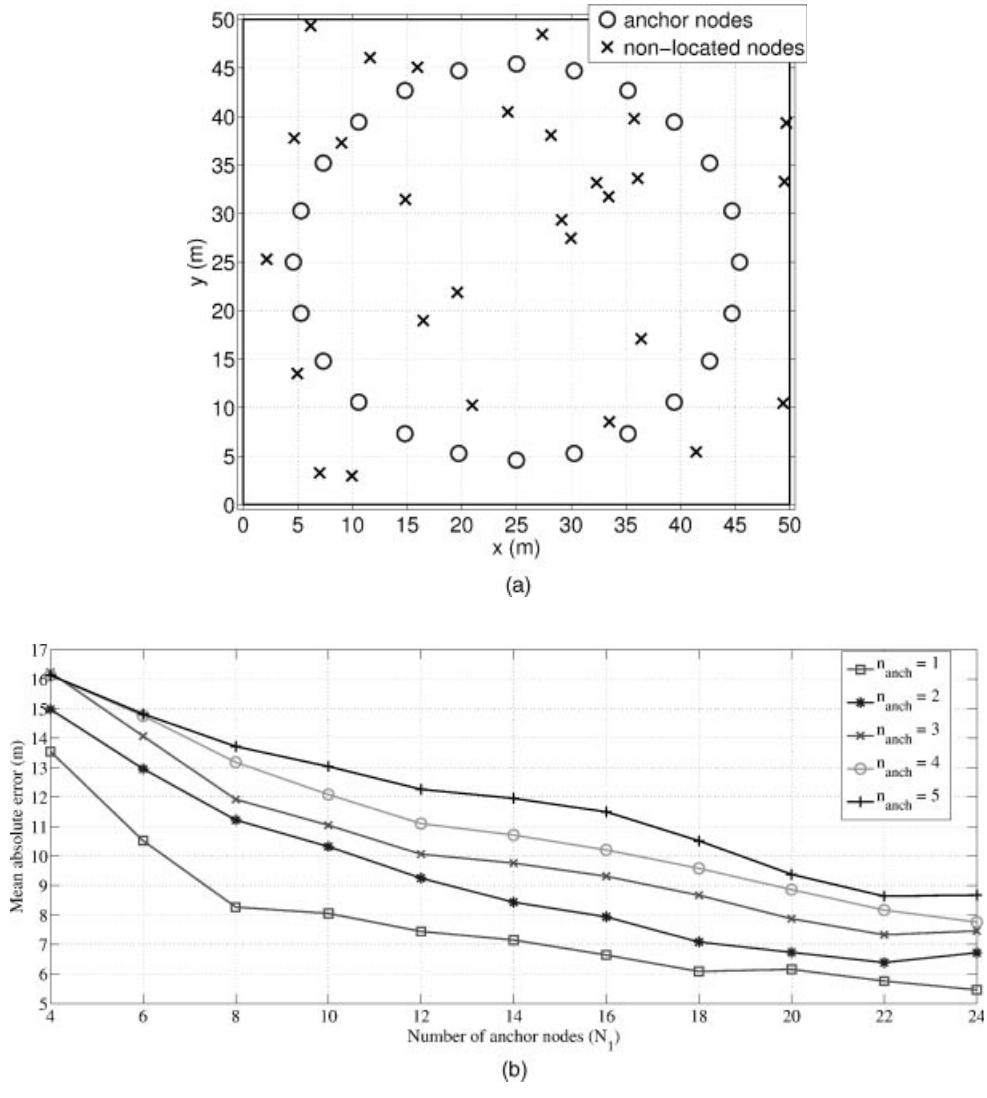

Figure 6.11 Circular-based anchor distribution: (a) simulation scenario, (b) mean absolute error versus N_1.

uses three anchor nodes, the localization algorithm achieves better results in terms of accuracy.

In the sequel, both scenarios will be simulated in order to compare the efficiency of the solution, comparing the accuracy when the algorithm estimates the path loss exponent and when it does not. Furthermore, a comparison between the presented algorithm and two other localization algorithms (a distributed and a centralized) will be presented.

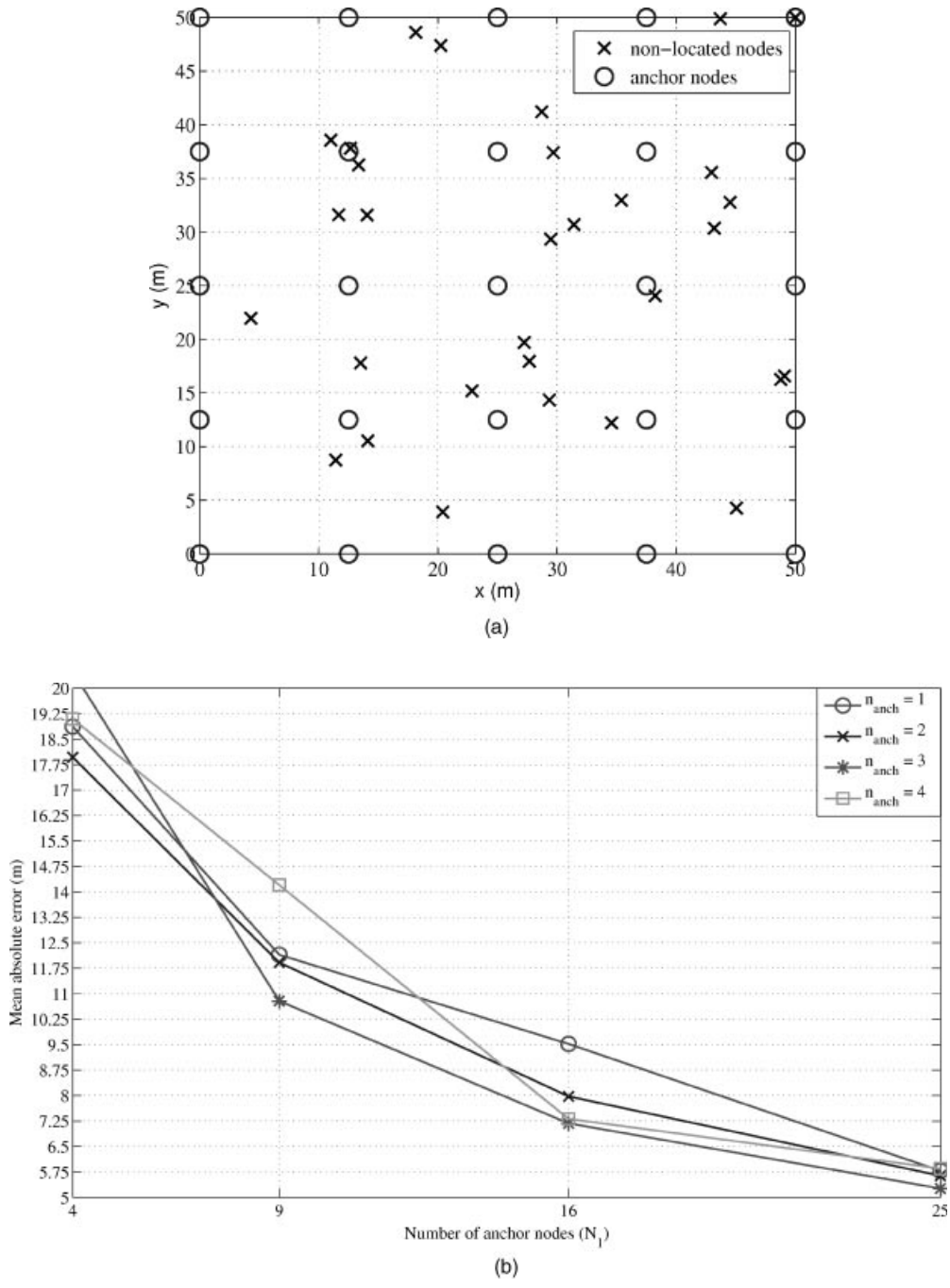

Figure 6.12 Grid-based anchor distribution: (a) simulation scenario, (b) mean absolute error versus N_1.

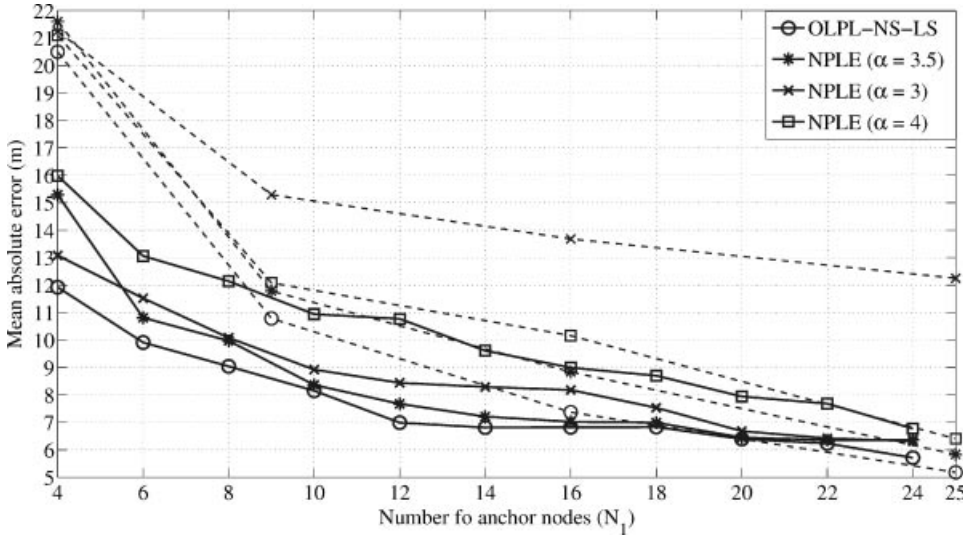

Figure 6.13 Mean absolute error versus N_1 (solid line: circular-based anchor distribution, dashed line: grid-based anchor distribution).

6.6.1 OLPL-NS-LS Performance

In order to simulate a more realistic scenario different path loss exponents are assumed. Each link has different values for the path loss exponent. With this assumption the importance of doing an on-line estimation of the path loss exponent is reflected in Figure 6.13.

With the proposed solution (OLPL-NS-LS) a gain in terms of position accuracy is achieved. With a centred anchor distribution, the gain oscillates between 2 and 0.5 m, compared to that achieved with the non path loss estimation (NPLE) algorithm. On the other hand, with results achieved using a grid-based anchor distribution the error gain oscillates between 7 and 0.7 m (not taking into account the results achieved with a fixed path loss equal to 3).

With a grid-based distribution, the centroid algorithm uses three anchors, but these nodes, when the density of of anchors is low, are, in mean, at a further distance from the non-located nodes. Hence, the starting point has low accuracy and this fact affects the final accuracy of the localization algorithm. When N_1 is increased the grid-based anchor distribution achieves better results in terms of position accuracy.

On the other hand, the best result is always achieved with the on-line path loss estimation and node selection least squares algorithm(OLPL-NS-LS). Different values of path loss exponents have been simulated. Hence having an on-line estimation of the path loss has a good influence on the localization performance accuracy. This estimation also makes possible the adaptation of the algorithm to possible changes in the scenario.

6.6.2 Comparison with Existing Methods

In this subsection, the OLPL-NS-LS algorithm is compared with two different existing solutions: a distributed method based on a maximum likelihood estimation (MLE) algorithm and

a centralized algorithm based on multidimensional scaling (MDS). The on-line path loss estimation is applied to all the methods in order to achieve a fair comparison between them. Only the OLPL-NS-LS method presents the node selection method. The comparison between the three methods is carried out in terms of both energy consumption and positioning accuracy.

The MLE localization algorithm [30] used is based on the minimization of the following cost function carried out with a distributed iterative method:

$$C_{ML}(x) = \sum_{i=1}^{N_2} \sum_{j \in S_i} \left(\log_{10}\left(\delta_{ij}(\alpha_{ij})\right) - \log_{10}\left(d_{ij}(\boldsymbol{x}_i - \boldsymbol{x}_j)\right) \right)^2$$

As presented before, the MDS algorithm is a simple centralized approach that builds a global map using classical MDS [26]. MDS works well on networks with relatively uniform node density, but less well on more irregular networks.

The simulations are carried out with both anchor distributions. The results achieved are shown in Figure 6.14. The OLPL-NS-LS algorithm outperforms the other methods in both cases. When the number of anchor nodes increases, the grid-based anchor distribution achieves a high accuracy compared to that achieved by the circular distribution.

Given a circular anchor distribution, the MLE localization algorithm achieves similar values to those achieved by the OLPL-NS-LS. It is important to remark that a gain between 0.5 and 1.5 m is obtained with the OLPL-NS-LS. This gain is achieved thanks to the selection algorithm. It is more remarkable than the gain achieved with respect to the MDS localization algorithm. On the one hand, the MDS method includes more distant nodes. Then nodes with a high error on their distance estimates are used. On the other hand, all possible nodes inside

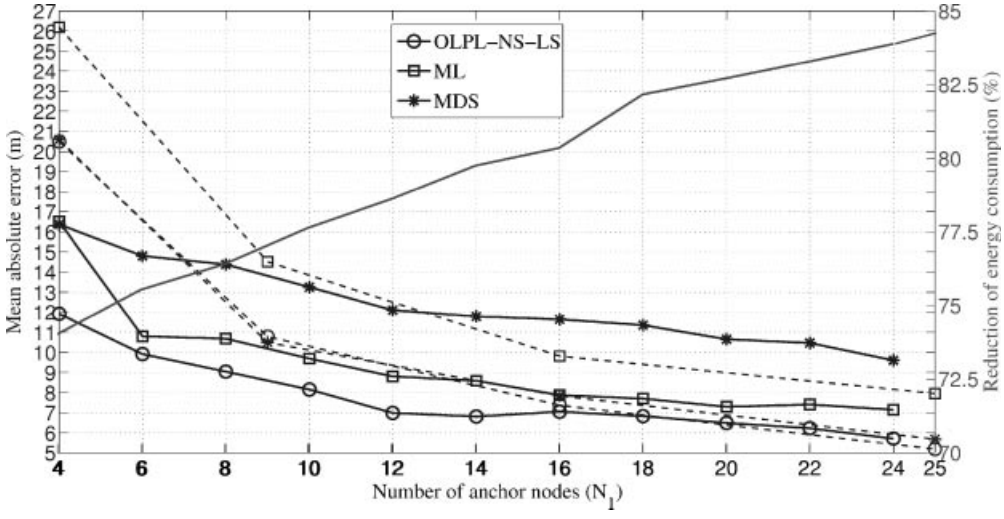

Figure 6.14 Mean absolute error versus N_1 (solid line: circular-based anchor distribution, dashed line: grid-based anchor distribution).

each group S_i are also used in the path loss estimation process. Probably nodes that are not near to node i would not have similar propagation conditions compared to those nodes that are closer.

The results achieved by the grid-based anchor distribution reflect the same tendency. The results of the OLPL-NS-LS present the best accuracy. However, in this case, the centralized MDS achieves better results than the MLE algorithm. The distribution of the anchor nodes in a grid along the scenario increases the probability of having closer anchor nodes in each S_i and hence better results are achieved when the number of anchor nodes inside S_i increases. The MDS achieves a better performance compared to that achieved with the centred distribution of the anchors. This reflects the importance of having more anchor nodes inside the cooperating group and also having nodes as close as possible. With a grid-based distribution the probability of having closer anchor nodes is higher. This higher probability is reflected in the better results obtained. The selection of the anchor node positions will depend on the possibility of having a higher density of nodes inside the network.

A node selection scheme allows the mean absolute error results to be reduced in an RSS-based localization algorithm. The reduction of the energy consumption is also important. According to the previous energy consumption model presented, the use of a reduced cooperating group S_i results in a lower consumption of energy. With the use of a number of cooperating nodes $n_\delta = 6$, the OLPL-NS-LS algorithm achieves a percentage of reduction between 74 and 83% compared to the energy consumed by a method without the node selection mechanism (see Figure 6.14).

6.7 Experimental Results

In this section experimental results are presented. The measurements taken in different indoor scenarios have been carried out with the Mica2 motes at 915 MHz of Crossbow [42]. Two different indoor scenarios are presented in Figure 6.16. In the first scenario the total number of nodes with an unknown position, N_2, is nine (see Figure 6.15a) and the number of anchor nodes, N_1, is four. These nodes are located in a $4.8\,\text{m} \times 4.8\,\text{m}$ scenario. The second scenario (see Figure 6.15b) is composed of $N_2 = 20$, $N_1 = 6$ and $N_1 = 4$ in a scenario of $8\,\text{m} \times 12\,\text{m}$. Experimental results are only carried out with the centred-based anchor distribution. As the anchor nodes used in the network are of a low value, simulation results reflect that it is better to select the circular-based distribution instead of the grid-based one.

6.7.1 Scenario 1

The localization error achieved in the first scenario is shown in Figure 6.16. The outage probability presented in Figure 6.16(a) validates the result shown in Figure 6.8. The outage probability of having an error below an error threshold shows that the best result is achieved with six cooperating nodes. Figure 6.16(b) shows the results of including the path loss estimation inside the algorithm. Considering a fixed path loss exponent in order to model the propagation channel produces worse results in terms of accuracy. The experimental values achieved are also compared with simulation results that have the same conditions. On the one hand, the lowest localization error achieved has a value of n_δ equal to 6, the same as in

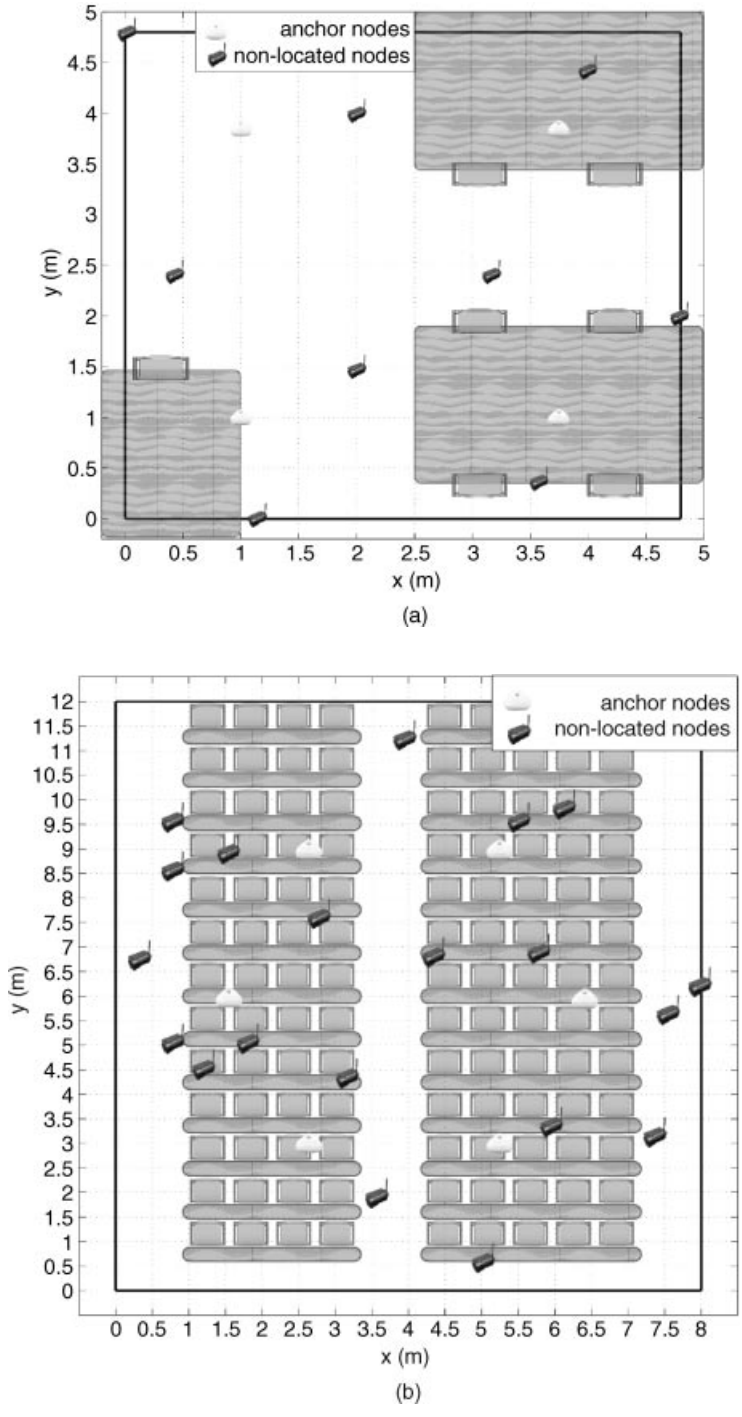

Figure 6.15 Experimental scenarios: (a) scenario 1, (b) scenario 2.

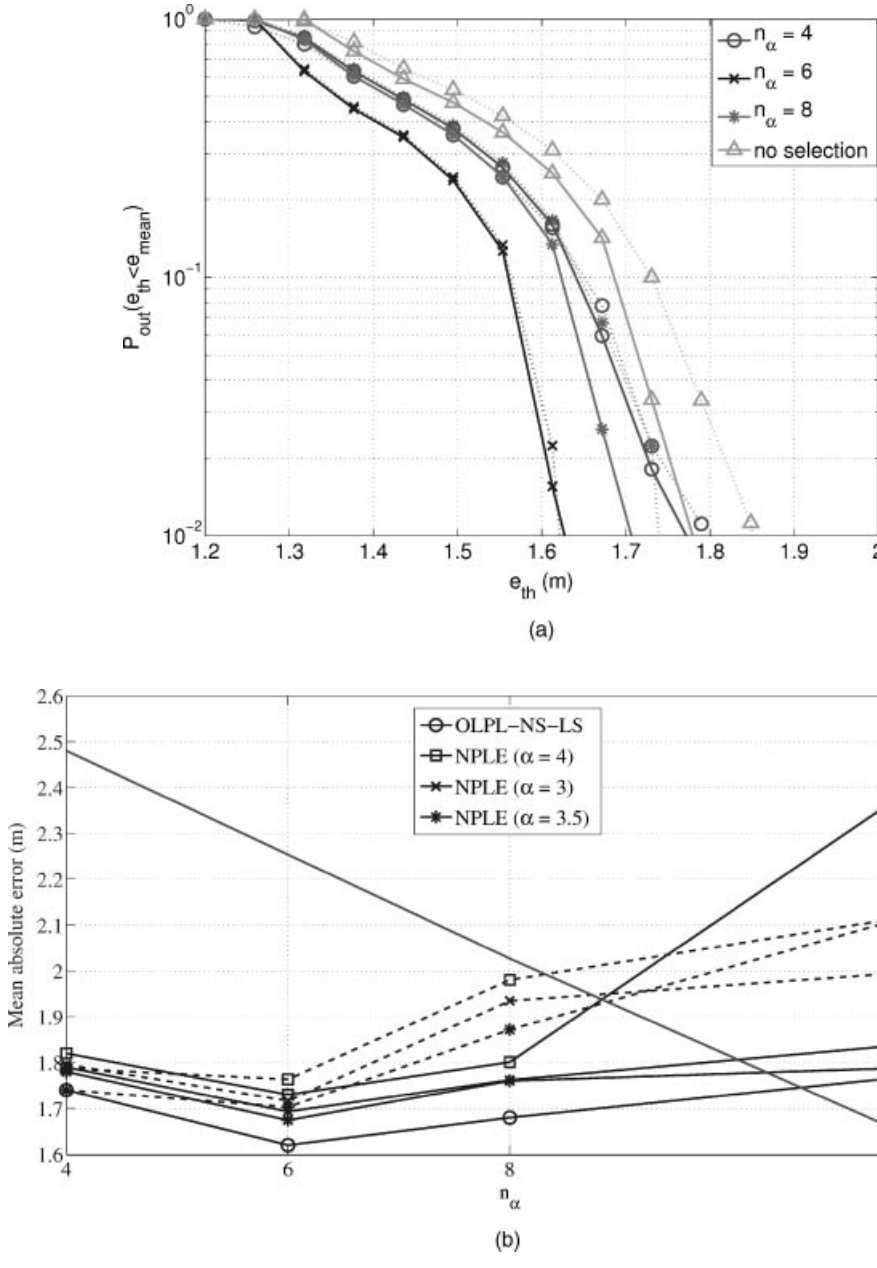

Figure 6.16 Results at scenario 1: (a) outage probability (solid line: experimental results, dashed line: simulation results), (b) mean absolute error versus n_δ (solid line: experimental results, dashed line: simulation results).

the large-scale case. On the other hand, simulation results are very similar to those achieved with the experimental scenario.

The best accuracy achieved is approximately 1.6 m. Furthermore, Figure 6.16(b) shows the percentage of reduction of the energy consumption compared to that consumed by an algorithm without node selection. With a value of n_δ equal to 6, the percentage of reduction in terms of energy consumption is about 50%. The comparison of behaviour between the OLPL-NS-LS and the NPLE with different values of the path loss exponent (α) gives the following results: the OLPL-NS-LS always achieves the best error values compared to those achieved with all the NPLE presented (see Figure 6.16b). The best behaviour is achieved with a fixed value of $\alpha = 3.5$. The differences in terms of mean absolute error oscillates between 0.2 cm when $\alpha = 3.5$ and 10 cm when $\alpha = 4$. The algorithm achieves a minimum gain of 5% and a maximum of 15%. Results of this experimental scenario show the gains obtained by considering the proposed on-line path loss estimation with respect to the case adopting an equal path loss exponent for all the links.

6.7.2 Scenario 2

As commented before, two different results are presented: one for a number of four anchor nodes and the other for six anchor nodes. Figure 6.17 shows both results. A similar behaviour is achieved when the simulation results are compared with the experimental results. As in the previous results, the best performance is achieved when the algorithm selects only six nodes in order to cooperate, validating the results achieved in Figure 6.9.

Although the number of nodes and the scenario dimensions have been increased, the performance of the algorithm (see Figure 6.17a) is similar to that achieved in the scenario shown in Figure 6.15(a). A higher scenario is considered but the probability of having closer nodes is increased. The accuracy achieved is equal to 1.7 m, on average.

Observing the results in Figure 6.17(b), one can see that the accuracy obtained is now 1.2 m. The benefit obtained with the increase in the number of anchor nodes is 0.5 m in the mean absolute error. Another important point is the difference of 0.5 m between the results achieved with the OLPL-NS-LS compared to an algorithm without path loss estimation results. Finally, the reduction in terms of energy consumption is, approximately, 75% in both scenarios.

In both figures, one can observe the benefits of the proposed OLPL-NL-LS approach when compared with the case of assuming a constant path loss exponent. A minimum gain of 18% (in the six anchor scenario) and a gain of 5% (in the four anchor scenario) is achieved. A better result in terms of position accuracy is always achieved with the OLPL-NS-LS.

6.8 Conclusions

A review of cooperative schemes for wireless sensor networks (WSNs) has been provided. Among all of them, a special emphasis has been given to the RSS-based measurement techniques, because they provide suitable solutions for practical implementations. Although RSS-based measurements are the simplest method, their accuracy depends on the suitability of the propagation model. Hence, a cooperative localization algorithm that dynamically estimates the path loss exponent has also been described. Furthermore, the reduction of the complexity and the message exchange through the node selection mechanism has been

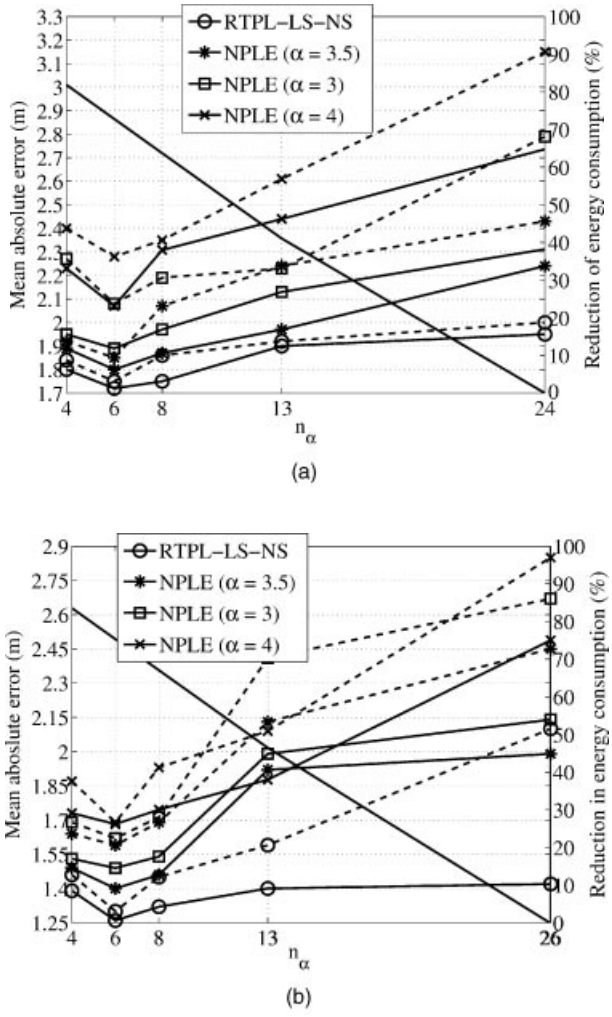

Figure 6.17 Mean absolute error versus n_δ (solid line: experimental results, dashed line: simulation results): (a) four anchor nodes (b) six anchor nodes.

presented. A good trade-off between localization accuracy versus energy consumption based on a node selection mechanism has been presented. In addition, practical examples based on a real WSN have been presented.

References

1. Bachrach, J. and Taylor, C. (2005) Localization in sensor networks, in *Handbook of Sensors Networks* (ed. J. Stojmenovic), John Wiley & Sons.
2. Boukerche F.A., Oliveira, H., Nakamura, E. and Loureiro, A. (2007) Localization systems for wireless sensor networks. *Wireless Communications*, **14**, 6–12.

3. Mao, G. and Fidan, B. (2009) *Localization Algorithm Strategies for Wireless Sensor Networks*, Information Science Reference.
4. Bulusu, N., Heidemann, J. and Estrin, D. (2000) GPS-less low-cost outdoor localization for very small devices. *IEEE Personal Communications*, **7** (5), 28–34.
5. He, T., Huang, C., Blum, B.M. *et al.* (2003) Range-free localization schemes for large scale sensor networks. MobiCom.
6. Niculescu, D. and Nath, B. (2003) DV based positioning in ad hoc networks. *Telecommunication Systems*, **22** (1), 267–280.
7. Li, R. and Fang, Z. (2010) LLA: a new high precision mobile node localization algorithm based on TOA. *Journal of Communications*, **5** (8), 604–611.
8. Yu, K., Guo, Y.J. and Hedley, M. (2009) TOA-based distributed localisation with unknown internal delays and clock frequency offsets in wireless sensor networks. *IET Signal Processing*, **3** (2), 106–118.
9. Patwari, N., Ash, J., Kyperountas, S. *et al.* (2005) Locating the nodes: Cooperative localization in wireless sensor networks. *IEEE Signal Processing*, **22**, 54–69.
10. Guvenc, I. and Chong, C.C. (2009) A survey on TOA based wireless localization and NLOS mitigation techniques. *IEEE Communications Surveys and Tutorials*, **11** (3), 107–124.
11. Mao, G., Fidan, B. and Anderson, B.D.O. (2007) Wireless sensor networks localization techniques. *Computer Networks*, **51** (10), 2529–2553.
12. Shimizu, Y. and Sanada, Y. (2003) Accuracy of relative distance measurement with ultra wideband system. IEEE Conference on Ultra Wideband Systems and Technologies, pp. 374–378.
13. Gezici, S., Tian, Z., Giannakis, G.B. *et al.* (2005) Localization via ultra-wideband radios: a look at positioning aspects for future sensor networks. *IEEE Signal Processing Magazine*, **22** (4), 70–84.
14. Boukerche, A., Oliveira, H.A.B., Nakamura, E.F. and Loureiro, A.A.F. (2007) Localization systems for wireless sensor networks. *IEEE Wireless Communications*, **14** (6), 6–12.
15. Priyantha, N.B., Chakraboty, A. and Balakrishnan, H. (2000) The cricket location-support system. 6th ACM International Conference on Mobile Computing and Networking, pp. 32–43.
16. Sallai, J., Balogh, G., Maróti, M. and Lédezci, Á. (2004) Acoustic ranging in resource-constrained sensor networks. ICWN, Las Vegas, NV.
17. Mazomenos, E., De Jager, D., Reeve, J. and White, N. (2011) A two-way time of flight ranging scheme for wireless sensor networks. 8th European Conference on Wireless Sensor Networks, Bonn, pp. 163–178.
18. Yang, J. and Chen, Y. (2009) Indoor localization using improved RSS-based lateration methods. IEEE Global Telecommunications Conference, pp. 1–6.
19. Patwari, N., O'Dea, R.J. and Wang, Y. (2001) Relative location in wireless networks. IEEE Vehicular Technology Conference Spring, pp. 1149–1153.
20. Hashemi, H. (1993) The indoor radio propagation channel. *Proceedings of the IEEE*, **81** (7), 943–968.
21. Kumar, P., Reddy, L. and Varma, S. (2009) Distance measurement and error estimation scheme for RSSI based localization in wireless sensor networks. 5th Conference on Wireless Communication and Sensor Networks, pp. 1–4.
22. Kulakowski, P., Vales-Alonso, J., Egea-López, E. *et al.* (2010) Angle-of-arrival localization based on antenna arrays for wireless networks. *Computer and Information Science*, **36** (6), 1181–1186.
23. Ash, J.N. and Potter, L.C. (2004) Sensor network localization via received signal strength measurements antennas. Allerton Conference on Communication, Control and Computing, pp. 1861–1870.
24. Maróti, M., Völgyesi, P., Dóra, S. *et al.* (2005) Radio interferometric geolocation. Proceedings of the 3rd International Conference on Embedded Networked Sensor Systems.
25. Fox, V., Hightower, J., Liao, L. *et al.* (2003) Bayesian filtering for location estimation. *IEEE Pervasive Computing*, **2** (3), 24–33.
26. Shang, Y., Ruml, W., Zhang, Y. and Fromherz, M.P.J. (2003) Localization from mere connectivity. Proceedings of the 4th ACM International Symposium on Mobile Ad Hoc Networking and Computing.
27. Kown, O.H. and Song, H.J. (2008) Localization through map stitching in wireless sensor networks. *IEEE Transactions on Parallel and Distributed Systems*, **19** (1), 93–105.
28. Biswas, P., Liang, T.C., Toh, K.C. *et al.* (2006) Semidefinite programming approaches for sensor network localization with noisy distance measurements. *IEEE Transactions on Automation Science and Engineering*, **3** (4), 360–371.
29. Wymeersch, H., Lien, J. and Win, M.Z. (2009) Cooperative localization in wireless networks. *Proceedings of the IEEE*, **97** (2), 427–450.

30. Denis, B., Pierrot, J.B. and Abou-Rjeily, C. (2006) Joint distributed synchronization and positioning in UWB ad hoc networks using TOA. *IEEE Transactions in Microwave Theory and Technology*, **54**, 1896–1911.
31. Thrun, S., Fox, D., Burgard, W. and Dellaert, F. (2001) Robust Monte Carlo localization for mobile robots. *Artificial Intelligence*, **128** (1), 99–141.
32. Patwari, N., Hero, A.O.I., Perkins, M. *et al.* (2003) Relative location estimation in wireless sensor networks. *IEEE Transactions on Signal Processing*, **51** (8), 2137–2148.
33. Tam, V., Cheng, K. and Lui, K. (2006) Using micro-genetic algorithms to improve localization in wireless sensor networks. *Journal of Communications*, **1** (4), 1–10.
34. Lieckfeldt, D., You, J. and Timmermann, D. (2008) Distributed selection of references for localization in wireless sensor networks. Proceedings of the 5th Workshop on Positioning, Navigation and Communication, pp. 31–36.
35. Das, K. and Wymeersch, H. (2012) Censoring for Bayesian cooperative positioning in dense wireless networks. *IEEE Journal on Selected Areas in Communications* (in press).
36. Bel, A., Vicario, J.L. and Seco-Granados, G. (2010) Node selection for cooperative localization: efficient energy vs. accuracy trade-off. Proceedings of the 5th IEEE International Symposium on Wireless Pervasive Computing, pp. 307–312.
37. Zou, Y. and Chakrabarty, K. (2003) Energy-aware target localization in wireless sensor networks. Proceedings of the 1st IEEE International Conference on Pervasive Computing and Communications, pp. 60–67.
38. Li, X. (2007) Collaborative localization with received-signal strength in wireless sensor networks. *IEEE Transactions on Vehicular Technology*, **56** (6), 3807–3817.
39. Bertsekas, D.P. and Tsitsiklis, J.N. (1997) *Parallel Distributed Computation: Numerical Methods (Optimization and Neural Computation*), Athena Scientific.
40. Mazuelas, S., Bahillo, A., Lorenzo, R. *et al.* (2009) Robust indoor positioning provided by real-time RSSI values in unmodified WLAN networks. *IEEE Journal of Selected Topics in Signal Processing*, **3**, 821–831.
41. Ash, J. and Moses, R. (2008) On optimal anchor node placement in sensor localization by optimization of subspace principal angles. IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 2289–2292.
42. Crossbow Technology [online], available from: HYPERLINK “http://www.xbow.com”, http://www.xbow.com.