Università degli Studi di Firenze

International Doctoral Program in
"RF, Microwaves and Electromagnetism"
Course XIX
Department of Electronics and Telecommunications
University of Florence, Italy

RESOURCE-CONSTRAINED LOCALIZATION IN SENSOR NETWORKS

by

Gianni Giorgetti

ING-INF/01

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
in Electrical Engineering

2007

Advisors:

Prof. Gianfranco Manes
(Program Coordinator)
Università degli Studi di Firenze

Dr. Alessandro Cidronali
Università degli Studi di Firenze

Prof. Stephen M. Goodnick
Arizona State University

Prof. Sandeep K.S. Gupta
Arizona State University

Ai miei genitori

Abstract

In sensor networks applications, localization is an essential service that computes the node positions on the basis of a limited amount of initial information. The task is particularly challenging in resource-constrained deployments typical of many real-world applications, where nodes have reduced computational capabilities, do not have hardware for range measurements and operates in sparse topologies. In this thesis we propose a *range-free*, *anchor-free* solution that works using connectivity information only. The approach, suitable for deployments with strict cost constraints, is based on the neural network paradigm of Self-Organizing Maps (SOM). We present a lightweight SOM-based algorithm to compute *virtual* coordinates that are effective for location-aided routing. If absolute coordinates are required, this algorithm can efficiently exploit information of few anchor nodes to compute absolute maps. Results of extensive simulations show improvements over the popular Multi-Dimensional Scaling (MDS) scheme, especially for networks with low connectivity, which are intrinsically harder to localize, and in presence of irregular radio pattern or anisotropic deployment. We analytically demonstrate that the proposed scheme has low computation and communication overheads; hence, making it suitable for resource-constrained networks.

In the second part of this work, we introduce a directional antenna designed to operates with COTS sensor nodes. After using experimental tests and theoretical models to characterize the communication improvements, we implements a simple algorithm that exploits the directivity of the antenna to estimate the angular position of nearby nodes. Experimental results demonstrate that an inexpensive and compact antenna can be used to derive angle information useful in solving the localization problem.

Acknowledgments

This thesis is the result of work done during a joint doctoral program between the Università degli Studi di Firenze and Arizona State University, a course of studies that brought me to spend several months in US. I owe sincere gratitude to my advisors Prof. Gianfranco Manes and Prof. Stephen Goodnick for making this program possible. I also want to thank Dr. Alessandro Cidronali who enthusiastically supervised my research activity and offered invaluable help in solving all the administrative issues that arose in organizing the joint program. Together with other colleagues and friends at the Microelectronic Lab, he also designed the directional antenna that I used in my experiments. Thanks to Stefano Maddio, Dr. Iacopo Magrini and Stefano Maurri who worked on the antenna project. I also owe a big thank-you to Stefano Maurri for many stimulating discussions and for his help in realizing various hardware prototypes. I warmly thank my other office mates: Carmelo Accillaro, Matteo Camprini, Giovanni Loglio and Rossano Fagotti for all the good time at the Microelectronics lab.

I am grateful to my advisor at ASU, Prof. Sandeep Gupta, for his support and guidance during my studies in Arizona and for continuing to follow my activity while in Italy. Thanks to my student colleagues at IMPACT lab for making my stay in AZ particularly enjoyable: Guofend Deng, Su-Jin Kim, Tridib Mukherjee, Qinghui Tang, Krishna Venkatasubramanian and Edward Raleigh for his help during the antenna experiments. I also want to thank Dr. Kari Torkkola for the interesting course on neural networks where I found the idea for the SOM based localization solution discussed in this thesis. A special thank you goes to Lisa Sparbel for proofreading some chapters of this thesis (or nerd stuff, as she calls it) and to Sal Mastroianni and Jonathan Lewis for their helpful advice and many interesting discussions during my internship at Motorola Labs.

Finally, my deepest gratitude goes to my family: my father (he also helped me during the antenna experiments), my mother and my two sisters Laura and Monica who all encouraged and supported me during these years of studies.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In the era of pervasive computing, *position-awareness* is rapidly becoming a key feature in many applications [1]. This trend is confirmed by the fast growth of the Global Positioning System (GPS) that has recently invaded the consumer market. Once restricted to military applications, GPS receivers are now common on cars, trucks, PDAs and cell phones. With an estimated 14 million units sold in 2006 [11], the success of this technology underlines the importance of locating people and things in a world where computation and communication are becoming ubiquitous.

Position-awareness is also of primary importance in Wireless Sensor Networks (WSN) [23], which is an enabling technology for pervasive computing. Consisting of small sensors with wireless capabilities, these networks are easy to deploy and represent a cost effective alternative to traditional wired systems. Typical applications include environmental monitoring, asset tracking, surveillance and disaster relief [2]. In each of these cases, and in almost any other WSN application of practical interest, knowledge of the node positions is required to correctly evaluate the network results. For example, in a disaster relief scenario, a sensor network deployed "on the fly" needs to provide search-and-rescue teams with location information necessary to quickly arrive on the emergency scene.

Given the importance of this information, several research efforts have focused on incorporating location awareness in those applications where the use of GPS is not a viable solution [31, 50]. In Chapter 2 we give a formal definition of the problem and we introduce some background information that characterizes the computational complexity of the solution. Since the localization problem is intrinsically hard to solve and no efficient solutions exist for the general case, many heuristic-based algorithms have been proposed over the past few year in the attempt to compute approximate, but still useful, node coordinates. We review some of the most relevant localization schemes in Chapter 3, giving a particular attention to "range-free" schemes (i.e. schemes that do not rely on range measurements to recover the node positions). Although these schemes are appealing because they can be implemented on simple sensor nodes without dedicated hardware for range estimation, many of the proposed approaches rely on a large number of anchor nodes (i.e. nodes at known positions) which limit the applicability of the solution to real-case applications.

We also note that many of these localization schemes target unrealistically large WSNs with high connectivity. While a few examples of such large, dense deployments are being experimentally evaluated within the research community [5], a recent survey [10] reveals that most of the future WSN applications will exploit small to medium networks with less than 100 nodes. Contrary to other network services, a small number of nodes and low connectivity are problematic for the existing localization schemes since determining the sensor positions becomes intrinsically harder as the number of constraints (e.g. range measurements or radio links) diminishes [28].

Motivated by these considerations, in Chapter 4 we define the characteristic of a novel localization scheme that works without anchor nodes, does not rely on range measurements and is designed to work on networks with low connectivity or irregular topologies. This localization method, which is based on a neural network formalism known as Self-Organizing Map (SOM), is presented in Chapter 5. The SOM approach leads to a light-weight implementation of a localization scheme capable of generating virtual coordinates that describe the relative positions of nodes. In Chapter 6 we

demonstrate through extensive simulations that these virtual maps can be used for efficient geographic routing, with results that are very close to the case where the real coordinates are available.

We also evaluate the localization error when absolute positioning is required: using only three or four anchor nodes, the virtual coordinates can be converted into absolute positions by means of a linear transformation. The results are compared to those of the popular Multi-Dimensional Scaling (MDS) technique [82], showing substantial improvement especially for networks with low connectivity or anisotropic layout. In these cases, which are intrinsically harder to localize, the proposed approach reduces the localization error of factor comprised between 43% and 75% with respect to the MDS technique. At the end of Chapter 6 we evaluate the computational and communication complexity of the solution. This analysis, in addition to benchmark tests on real hardware, shows that our lightweight implementation is suitable to solve the localization problem in resource-constrained networks.

After having presented the SOM based algorithm, in Chapter 7 we introduce a directional antenna which is suitable for WSN applications and can be used to derive *Angle of Arrival* (AoA) information useful to solve the localization problem. The antenna was designed by the Microelectronics Lab at the University of Florence to meet the size, cost and complexity constraints of sensor nodes. In-field experiments with COTS motes are used to demonstrate substantial benefits to WSN applications: used outdoors, the antenna extends the communication range from 140m to more than 350m, while indoors it suppresses the interference due to multipath fading by reducing the signal variability of more than 70%. We also show interference suppression from IEEE 802.11g systems and discuss the use of the antenna as a form of angular diversity useful to cope with the variability of the radio signal. Experimental data are analyzed to derive model parameters intended for use in future network simulations.

In Chapter 8, we use part of the results introduced above to define an algorithm that analyzes the signal power on the four faces to derive the angular position of a node that communicates with the directional antenna. We use experimental data

to compare the estimation accuracy of different approaches, showing that an average error of about 15° is possible using an algorithm based on an error function designed to take into account missing readings on some of the antenna patches. This preliminary study is intended to serve future extensions to the localization algorithm, where angle information will be used to improve the localization accuracy of the SOM based solution.

Chapter 2

Background

In many applications of practical interest, the information gathered by a WSN is of scarce utility unless stamped with the location of the sensors that collected it. For example, in precision agriculture, temperature and moisture values are correlated with position to identify micro-climate zones [85]. Knowing the sensors' position is also critical for locating an intruder vehicle in a military application [4] as well as to guide a team of firefighters to the location of an emergency [18]. Finally, locations are used to support network services like geographical routing [41], location-based queries [35] and resource directories [55]. Unfortunately, this information is generally not available since WSNs are deployed without control and sensor nodes do not posses enough hardware resources to determine their location. Ideally, we would like to equip each sensor node with a GPS receiver, but the cost, size and power consumption of these devices are prohibitive for sensor nodes that need to be small and inexpensive. In addition, sensor nodes are often deployed indoors or in other environments (e.g. woods with dense foliage) where the absence of line of sight with the GPS satellites prevents the use of this technology. Another solution would be to manually configure the position of each node, but this is infeasible when a WSN comprises a large number of nodes or is deployed in a hostile/inaccessible environment.

Given the importance of location information, the research community has produced a great effort to implement localization schemes suitable for sensor networks. Before reviewing some relevant algorithms in Chapter 3, here we introduce the theoretical foundations of the localization problem, proposing a summary of relevant literature that discusses the computational complexity of finding exact or approximate node coordinates.

2.1 Problem Definition

To introduce the problem of locating a set of sensors, we consider a Wireless Sensor Network with N nodes randomly deployed. Each node i, $i \in \{1, \ldots, N\}$, is characterized by its (unknown) position $p_i \in R^d$ ($d = 2$ or 3), by a unique identifier and by a set of *neighboring* nodes that we indicate with the notation NB(i). In general, given any two nodes i and j, we define them as *neighbors* if they can successfully exchange radio messages over a *symmetric, bidirectional radio link.*[1]. Given the above conditions, the localization task consists in determining the unknown positions p_i using information about the network structure, a problem that is formally analogue to the one of *embedding a graph* in an Euclidean space. This subject has been extensively studied in the area of computational geometry and graph rigidity. Sample applications include satellite ranging, the study of rigid structures and the analysis of molecular conformation using inter-atomic distances obtained through magnetic resonance. In the next section, we will introduce results from graph theory to characterize the problem's complexity and identify the conditions under which it admits a unique solution. After first considering the case where edge lengths are known, we will analyze literature results that apply to locating the nodes using connectivity information only.

[1] In real applications, radio links are subjected to interference and sporadic failures. The presence of a radio link needs to be defined on the basis of some statistic on the percentage of messages correctly exchanged between the two units.

2.2 Embedding With Known Edge Lengths

A wireless sensor network can be represented as a undirected graph $G = (V, E)$, where each vertex is associated to a node, $V = \{1, \ldots, N\}$ and the set E contains an edge $\{i, j\}$ if nodes i and j are neighbors. Finding a *graph embedding* consist in determining a mapping function $\rho : V \rightarrow R^d$ that assigns a position to each vertex using constraints derived from the edges. A coordinate assignment produced by the mapping function is called *realization*. A common instance of the problem arises when the edge lengths are known and the vertexes must be assigned to positions such that the inter-node distances are preserved, $d(p_i, p_j) = d(\rho(i), \rho(j)),\ \forall \{i, j\} \in E$, where $d(\cdot)$ indicate the Euclidean distance. This case corresponds to the situation where node posses dedicated hardware (e.g. ultrasound transceiver) and are able to obtain distance information from their neighbors.

A fundamental question related to this problem is to determine the conditions such that the graph G admits a unique realization. We point out that identifying the conditions for unique graph realization allow us to define the characteristic of a WSN (mainly in term of connectivity) such that the nodes position can be computed without ambiguities. We also note that since only relative measurements are used, every solution will be correct up to global translations, rotations or reflections (see Figure 2.1). The graph can be properly oriented by fixing the relative positions of three non-collinear nodes in the 2D space or four such nodes in the 3D space. Fixing these points is not sufficient to produce an unique solution since the graph can generate multiple realizations when its structure is not rigid (see Figure 2.2).

In the 2D space, a graph with N nodes is characterized by $2N - 3$ degree of freedom (two for each node minus a global rotation or translations): since each edge introduces a constraint, a rigid graph needs at least $2N - 3$ *well-distributed* edges [49]. The condition is not sufficient since rigid graphs are still susceptible of discontinuous motions. In Figure 2.3 we report an example of a rigid graph that can generate multiple realization without violating the constraints on the edge lengths. If we

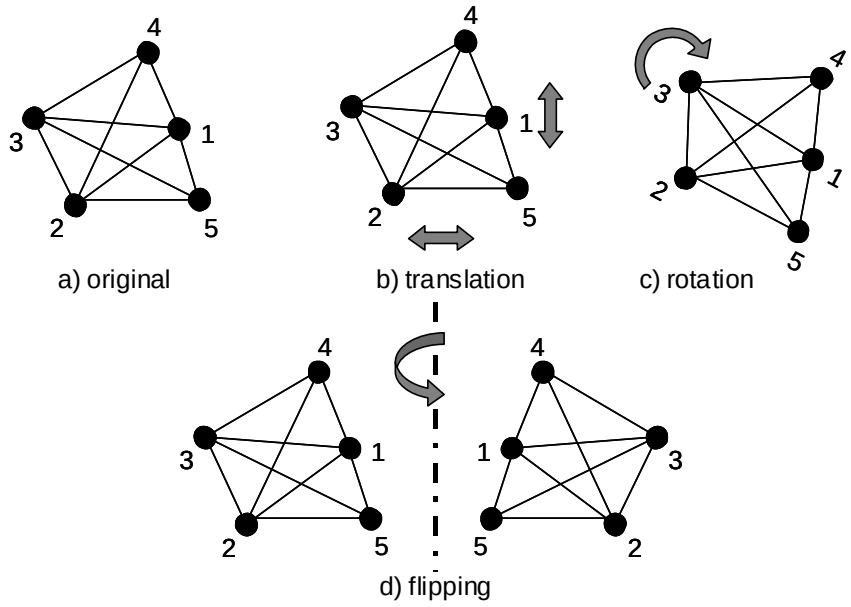


Figure 2.1: Localization ambiguities in absence of reference points.

temporarily remove the edge between nodes 2 and 6, the quadrilateral defined by nodes 1-3-4-6 can be deformed to generate a new configuration where the edge 2-6 can be reinserted without changing its length (discontinuous flex ambiguity). For a complete characterization of unique realization the graph needs to be $(d+1)$ connected and *redundantly rigid*, meaning that the graph is still rigid upon removal of an edge [30]. While this condition has been proved to be necessary and sufficient [36] for unique graph realization in 2D and can be tested in polynomial time [37], no such

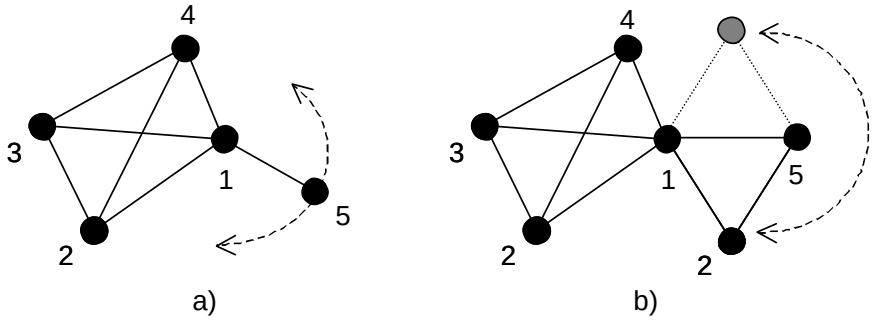


Figure 2.2: Flex Ambiguities.

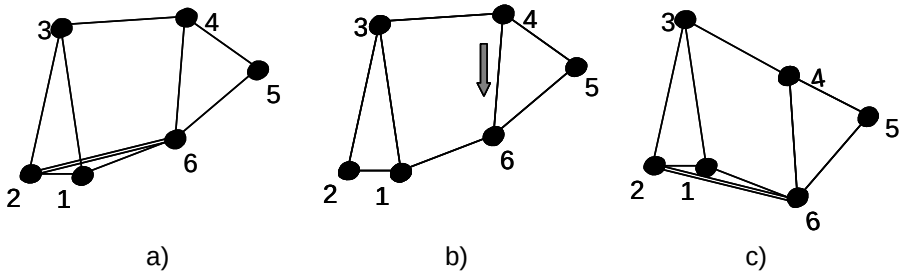


Figure 2.3: Discontinuous flex ambiguities.

characterization exists for graphs in higher dimensions. The result mentioned above have been used to determine the conditions under which the problem of localization with known inter-node distances can be solved [25],[28] or to improve the performance of basic trilateration algorithm under noisy measurements [62]

2.3 Embedding Using Connectivity Information

The results mentioned in the previous section apply to problem of embedding a graph with known edge lengths. From a WSN perspective, this implies that nodes have to perform range measurements, a capability that requires extra hardware (e.g. ultrasound transceiver) and thus increases the complexity of the network. A different approach consists in refraining from range measurements and use *connectivity* information to recover the node positions. Theoretical work under this assumption uses the Unit Disk Graph (UDG) [48] model to represent the network, where two nodes are neighbors *iff* their euclidean distance is less than one. By a proper coordinates scaling, this model can be used to represent an idealized wireless network, where two nodes are neighbors *iff* their distance is less than the communication range R (see Figure 2.4).

Intuitively, a localization scheme should produce a coordinate assignment where neighboring nodes are within the maximum radio range and non-neighbors are further apart. Although the problem can be stated in simple terms, the problem of embedding

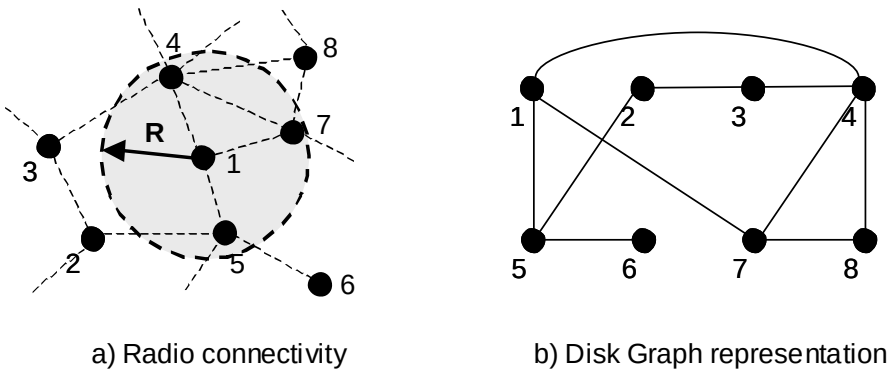


Figure 2.4: Unit Disk Graph.

an UDG in an Euclidean space is NP-Complete in one dimension and NP-hard in two dimensions [14]. More recently, the problem has been proved to be APX-hard [59], meaning that the solution cannot even be efficiently approximated. In fact, there are node configurations for which even an optimal algorithm cannot produce an embedding with quality better than $\sqrt{3/2}$ [48]. While this value limits the worst-case error for an optimal algorithm, localization schemes with bounded errors are very few. A recent work [63] proposes a scheme based on spreading constants and random projection with a bound error of $O(\log^{2.5} n\sqrt{\log\log n})$, where n is the number on nodes. Although this work has an appreciable theoretical value, from a practical point of view we are still far from approaching the theoretical bound $\sqrt{3/2}$ [72].

Chapter 3

Localization Schemes in Wireless Sensor Networks

The goal of a localization scheme is to assign a position (i.e. a pair of coordinates) to each node in a wireless sensor network. The solutions proposed over past few years achieve this goal using schemes that work under different assumptions and therefore are suitable for applications with different requirements. In particular, the trade-off between system complexity and accuracy plays a central role in WSN applications due to the limited resources (memory, computation, communication and above all, energy) available at each sensor node. A fundamental distinction in localization systems is whether they assume the possibility to measure the distance from some reference point (range-based) or not (range-free). The most popular example of range-based system is given by the widespread Global Positioning System [33], which allows a mobile node to accurately compute its position using the distances from three or more satellites. These distances, obtained by measuring the signal TOF (Time Of Flight), allow the receiver to compute its position using trilateration. The second type of systems, the range-free localization schemes, can usually provide a coarser resolution, but are less expensive to implement since they do not rely on sophisticate hardware. An example

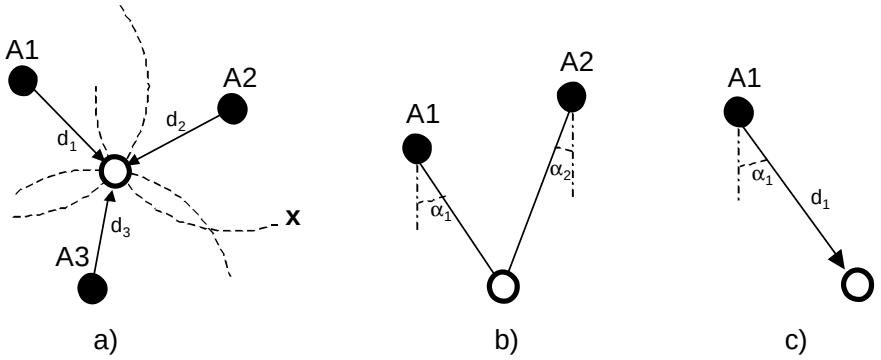


Figure 3.1: Localization in 2D: a) Trilateration: distance measurements from 3 non-collinear anchor points, b) Triangulation: angle measurements from 2 anchor point, c) angle and range from a single anchor point

of range-free scheme is a RFID system [33] that detects the proximity of an object close to some reference point. In the next sections we review some of the solutions proposed to solve the problem in a range-base or range free context. Surveys and comparisons among different localization techniques can be found in [31, 50, 65].

3.1 Range-Based Algorithms

An object in the 2D space can be localized when one of the following information is known: **i)** the distance from three non-collinear anchor nodes (lateration), **ii)** the angle from two anchor points (triangulation) or **iii)** the distance and the range from a reference point (see Figure 3.1).

Obtaining range and angle information is not a trivial task, in fact, the need to keep sensor nodes simple and inexpensive rules out the possibility to use sophisticate hardware for range or angle measurements. Some of the techniques used to obtain ranging information are outlined in the following sections.

3.1.1 Ranging Using Received Signal Strength Index (RSSI)

The RSSI is a measure of the signal power received by a wireless device. RF propagation in the free space follows the Friis equation and the path loss is proportional to $1/d^2$, where d is the distance between the source and the receiver. In real case applications, the path loss is harder to predict since it depends on the characteristic of the environment where the communication takes place. For example, the attenuation can be as low as $1/d^{1.5}$ along straight corridors that act as wave-guide, proportional to $1/d^4$ for near the ground transmission, where the component reflected by the ground destructively interferes with the LOS (line of sight) component, or even higher in complex indoor settings. Despite the unpredictability of the radio signal propagation, several research works assume the possibility to obtain range information from RSSI measurements (eg. [32]) and propose localization results based on simulations where the range measurements are affected by Gaussian noise [71]. The effectiveness of such solutions in real-world applications remains difficult to evaluate due to the differences in RF propagation when moving from an environment to another.

3.1.2 Ranging Using RF Time of Flight (ToF)

RF Time of Flight ranging techniques are conceptually more accurate than RSSI solutions because the measurements are based on signals that propagate at the speed of light (constant) and thus are less dependent on the environment. In practice, this technique can be used only if nodes are equipped with fast clocks capable of nanosecond accuracy (RF signals travel at 30cm per nanosecond); in addition the sender and the receiver nodes must be accurately synchronized. If synchronization is not feasible with high accuracy, an estimate of the range can be obtained by measuring the *round-trip* time of flight, but this also requires a precise evaluation of the time used by the target node to process the message and send a reply back to the source. Given these requirements, ToF ranging techniques, although promising (e.g. UWB location systems), are not suitable for WSN applications where nodes are clocked at

only few MHz. The use of faster clocks would increase both the cost of the hardware and the power consumption.

3.1.3 Ranging Using Time Difference of Arrival (TDoA)

Ranging using acoustic ultrasound is more attractive to WSN applications mainly because of two reasons: **i)** ultrasound transceivers are available as COTS components easy to interface with sensor nodes, **ii)** accurate localization can be achieved using low-rate clocks (given the speed of sound, a 32KHz clock is sufficient for 1cm localization accuracy). Because ultrasound ranging is relatively easy to implement on sensor nodes, several solutions [91, 75, 93, 80, 62] have been proposed where the source node transmits an ultrasound pulse and an RF packet at the same time. The radio message, which travels at much higher speed than the acoustic pulse, is used to trigger the receiver node which in turn measures the *Time Difference of Arrival* (TDoA) between the two signals. The distance between the two nodes is computed by taking into account the TDoA and the difference of speed between sound and RF signals. The main disadvantages of ultrasound ranging techniques is that sound propagation is affected by weather condition and the effective range is reduced to only a few meters when the transmitter and the receiver are not aligned (in facts, many transmitters emit a conical directional beam). This, together with the costs of the additional hardware, limits the applicability of this technology to WSN deployed in restricted area which need high accuracy.

3.1.4 Angle of Arrival (AoA)

Angle of Arrival (AoA) estimation using beamforming or phased antenna arrays is not appealing to sensor network applications due to the cost and complexity of these technologies. However, recent work on directional antennas demonstrated that simple switched patch units can meet the size and cost constraints of sensor nodes and several localization algorithm uses AoA to localize sensor nodes [69, 67, 94, 60]. In

Chapter 7, we introduce the four beam directional antenna developed at the University of Florence and we propose a technique for angle estimation that can be useful in localization algorithm (see Chapter 8).

3.2 Range-Free Algorithms

Range-free algorithms [87] overcome the high cost and system complexity of range-based schemes by using solutions that do not rely on dedicated hardware for distance or angle measurements. The location of each node is estimated by exploiting proximity information that is inferred using the radio or the sensors available at each node. In the first case, nodes that can successfully exchange radio messages must be not farther than R, where R is the maximum communication range. In the second case, sensors can be used to sense natural (or artificially generated) phenomenon that are used as basis for the localization process. The schemes are further classified on whether they rely on the presence of anchor nodes placed at known position or not (anchor-based vs anchor-free).

3.2.1 Anchor-Based Solutions

Centroid

The Centroid [15] scheme, which is one of the simplest solutions, works by assuming that the network contains a set of anchor nodes $A = \{a_1, \ldots, a_n\}$ placed at known locations $(x_i, x_j)_{i,j=\{1,\ldots,n\}}$. The anchors periodically broadcast their coordinates to the other nodes, which keep track of each message received. The nodes at unknown position determine their location by computing the average value of the anchor coordinates heard, i.e. the center of gravity, COG, of a system of masses placed in correspondence of the anchor nodes heard (see Figure 3.2). The robustness of the scheme is improved by maintaining statistics on the number of message received from each anchor. Only anchor nodes with a communication ratio greater than 90% are

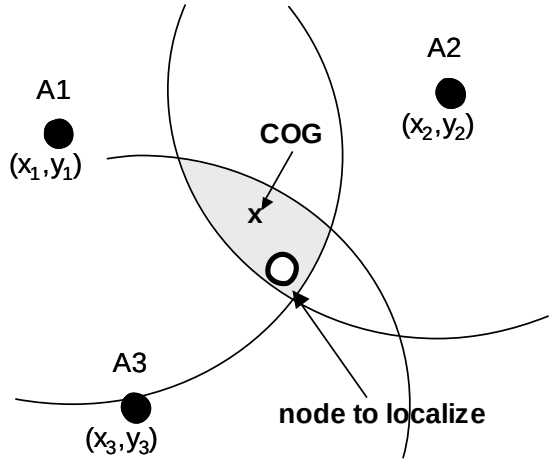


Figure 3.2: Centroid scheme: the position of the node to localize is given by the center of gravity (COG) of anchors heard

used in the computation. The localization accuracy of the centroid method is heavily affected by the number of anchor nodes used. In a subsequent work the authors propose a solution to adaptively place additional anchor nodes to decrease the localization error [16].

DV-Hop Scheme

In the "DV-Hop" scheme [70], anchors flood the network with message beacons that are re-transmitted by each node with the hop-count value increased by one unit. Using this approach, each node in the network will eventually be able to compute the shortest path distance (in terms of hop count) from any anchor in the network. To convert the path length into an absolute distance, the average hop count length is first computed using the following expression:

$$d_{hop} = \frac{\sum_i \sum_j \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}}{\sum_i \sum_j hi_j}.$$

In the above expression, the hopcount distance between any two anchor nodes is used to divide the actual Euclidean distance separating them (we recall that the

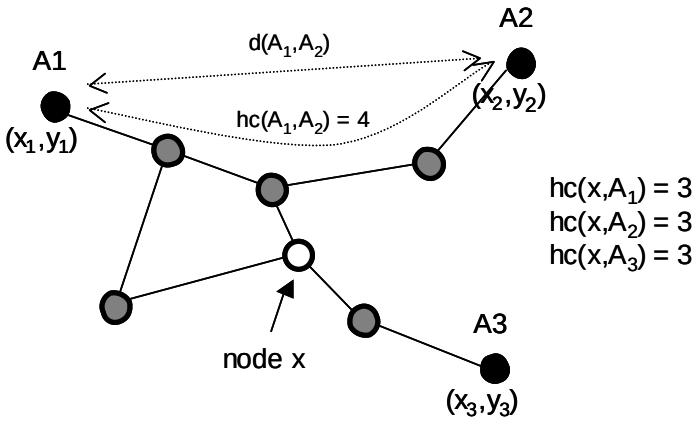


Figure 3.3: DV-Hop scheme: the shortest path (hop-count, hc) is used to estimate the distance from a node x to the anchor nodes in the network. The actual position is computed using multilateration.

location of anchor nodes is known). The result of the computation is an average hop count length that can be used to convert an hop-count value into a distance value (see Figure 3.3). Having determined the distance between three or more anchors, each node computes its location using multilateration. The authors use a least square method (the Householder method) to compute the actual position. The scheme works well when the path connecting nodes and anchor nodes lie approximately on a straight line: in this case the hop-count distance is a good approximation of the actual inter-node distance. When the network connectivity is low, the performance degrades since the hop count distance is not a good approximation of the actual distance (see Section 6.6 for more details).

A similar approach is proposed in [66], but in this case the estimation of the average hop-count length benefits from a priori knowledge of the nodes density through the use of the well known Kleinrock and Slivester formula [43] do determine the hop size:

$$d_{hop} = r\left(1 + \exp(-n_{local}) - \int_{-1}^{+1} \exp(\frac{-n_{local}\arccos(t - t\sqrt{1-t^2})}{\pi})dt\right)$$

where r is the average communication range and n_{local} is the local node density.

A.P.I.T.

The APIT scheme proposed in [29] is based on an approximate test to determine if a node is within the triangular area defined by three anchor nodes. The idea is that if the node to localize could move, it would detect increasing (decreasing) RSSI levels as it get closer (farther) to an anchor node. The PIT (point in triangle test) determines if a point is inside a triangle by checking for the existence of a direction such that moving on that path the node would get closer to all of the three anchor nodes (see Figure 3.4a). If such direction does not exist, the node is considered to be inside the triangle. Obviously the node cannot move, therefore an approximate version of the test (Approximated PIT) is performed by simulating virtual movements in the direction of the neighboring nodes (e.g. by comparing the RSSI values seen by adjacent nodes). The final node position is computed by intersecting the area of all the triangles a node belong to and then computing the COG (center of gravity) of such area (see Figure 3.4b).

We note that the scheme proposed is not a truly range-free solution since is based on RSSI comparisons, however the work is interesting because the authors use extensive simulation to compare their solution to the Centroid and DV-Hop schemes presented before. Results show that all the schemes previously mentioned perform well only when a high number of anchor nodes is present and the network density is high. For uniform topologies with connectivity equal to 8, each node needs to be able to receive the beacon messages from more than twelve anchor nodes to reduce the localization error under 1.0R (see Figure 3.5).

SeRLoc

SeRLoc [51] also implements an area-based, range-free approach similar to Centroids and APIT to determine the location of a node. The anchor nodes are equipped with switched directional antennas capable of covering the 360 horizon with non-overlapping sectors. The anchor nodes periodically switch beam and transmit *sec-*

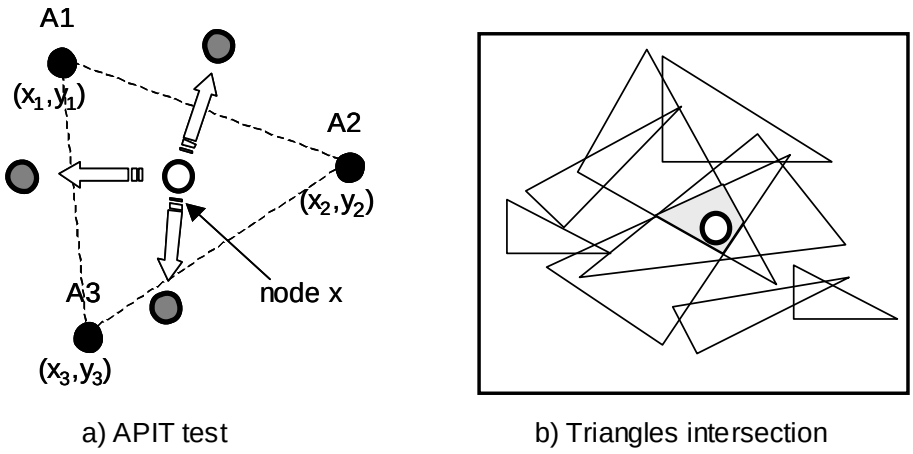


Figure 3.4: APIT scheme: a) Each node uses the APIT test to determine if it is inside the triangle area defined by three anchors. b) The final position is the COG of the intersection of all triangles a node belong to.

torized beacons to the rest of the nodes along with angular information about the beam used. Similarly to the other approaches considered so far, nodes compute their position by determining the intersection of the beams seen from each anchor node (see Figure 3.6). SeRLoc also addresses the problem of security in sensor network localization. Other algorithms designed to implement secure localization services are presented in [68, 56, 57, 52, 17].

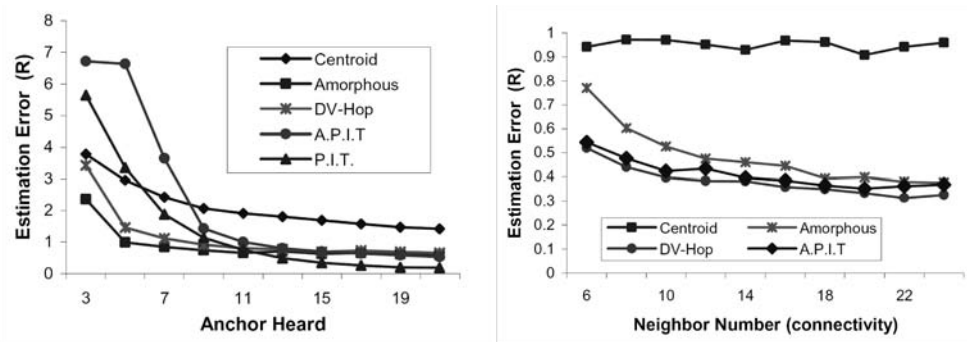


Figure 3.5: Comparison of the localization accuracy achieved by different range-free schemes. He et al.[29]

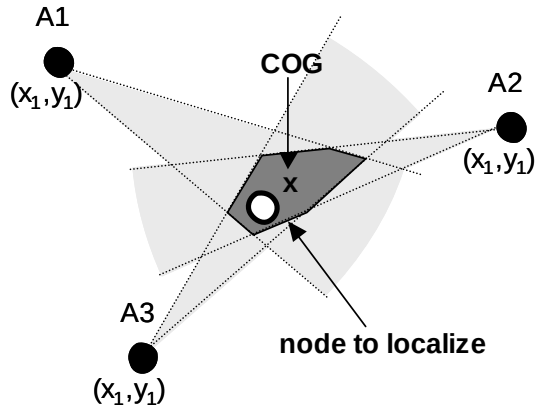


Figure 3.6: SeRLoc scheme: the anchor nodes generate *sectorized* beacons using directional antennas. The final node position is the COG of intersection area.

Probability Grid

Proablity grid [88] is a localization scheme based on the assumption that nodes are placed on a regular grid. It uses a very similar idea to the DV Hop positioning algorithm since anchor nodes flood the network with messages containing their position. Each node estimates the shortest path from each anchor node and then computes the probability of being on each intersection point of the grid. The location with maximum probability is chosen as an estimate for the node's position.

3.2.2 Anchor-Free Solutions

MDS

Multi-Dimensional Scaling (MDS) [13] is a technique that has been extensively used in psychometrics and many other applications to visualize multidimensional data sets. MDS implements a projection technique (to a 2D or 3D space) capable of preserving the similarities present in the original data set. The use of MDS to solve the localization problem in WSN was originally proposed in [82]. The node positions in the 2D space are computed by first creating a N by N matrix containing the squared

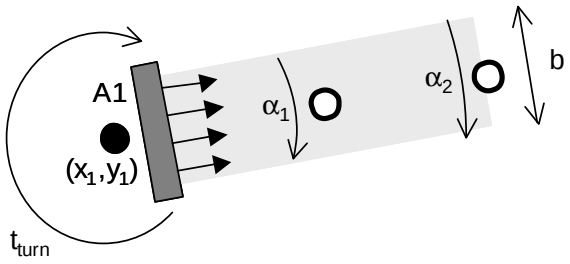


Figure 3.7: Lighthouse: a parallel beam is generated by a rotating base station. Nodes determine the distance from the base station by timing the light beam.

distance between each node in the network (where N is the number of nodes). If node distances are not available, the matrix is generated using the hop-count value between each node. The final coordinates are obtained by first double-centering the distance matrix and then using singular value decomposition and retaining the largest two eigenvectors (three for 3D localization). The method was successively extended to work in a distributed fashion [81, 39], motivated in part by the scarce performance with anisotropic layout like the ones described in section 6.6.

LightHouse

LightHouse [78] implements a different approach since node localization is achieved by exploiting the sensing capabilities of nodes. In this solution a base station mounted on a rotating support propagates a beam of light having width b (the beam is generated using an array of laser diodes) that is detected by the light sensors mounted on each of the nodes to localize (see Figure 3.7).

The main idea behind the algorithm is that it is possible to compute the distance from the base station by simply having each node measuring the time (t_{beam}) during which it sees the light from the base station. The distance is obtained through the equation:

$$d = \frac{b}{2\sin(\alpha_1/2)} = \frac{b}{2\sin(\pi t_{beam}/t_{turn})}$$

where b is the width of the beam, t_{beam} is time measured by a node and t_{turn} is the

rotation period. Localization in the 2D space is achieved by using three base stations mounted on orthogonal directions.

SpotLight

A similar approach to Lighthouse is used by the SpotLight system [86], which also relies on synchronized light events to localize a set of nodes. Three different scenarios are analyzed (see Figure 3.8): **i)** point scan: if the nodes lies on a straight line (e.g. nodes deployed along a road), they can be localized by a base station that emits a beam of light that is moved at constant speed along the line where the nodes lie. Since each sensor will detect the light at a different time, the node distance from a reference point can be computed by measuring the detection time and dividing it by the beam speed. **ii)** line scan: some devices (e.g. lasers) can generate lines of light that can be used to localize nodes on a 2D plane. A first line scanning in one direction (e.g. vertical) allows the nodes to measure their distance from the vertical origin of the deployment area (again, the distance is inferred by the time a node detects the light beam). 2D localization is achieved by a second beam that scans the network in direction perpendicular to the first one (e.g. horizontal). Finally, **iii)** a third method uses a video projector to illuminate the whole deployment area, which is partitioned in non-overlapping zones. Each zone is illuminated with an unique pattern light, where the presence of light denote a "1" bit and dark is "0". The temporal sequence of light/dark event is used to transmit a code representative of each area in the network.

3.3 Scene Analysis Algorithms

The major problem in using the RSSI signal to estimate a distance is that the signal propagation is affected by *hard-to-predict* phenomena like multipath fading, shadowing, scattering and interferences from other source operating in the same band. The problem is especially severe in indoor environments, where the presence of obstacles

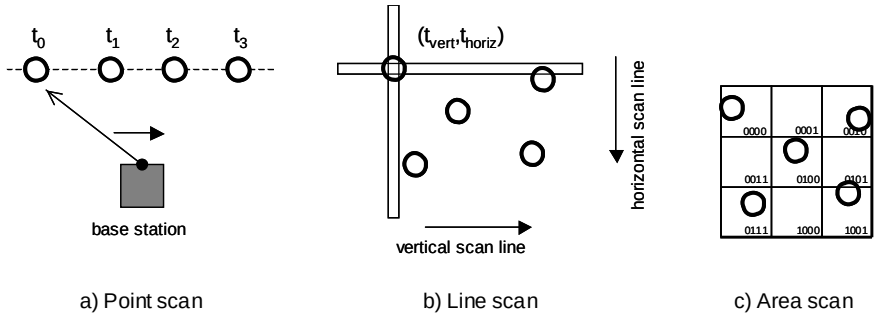


Figure 3.8: SpotLight: localization is implemented by timing the arrive of a light beam. Three options are available.

(e.g. large metal cabinets, doors, windows, ceiling fans) increases the variability of the signal strength. A localization approach that tries to overcome these difficulties is based on generating RF maps of an environment and then using these maps to locate moving people or objects [6, 58]. The method requires a setup phase during which a mobile device is used to record the signal strength from several base stations present in the network. The result of this phase is the creation of a *RF fingerprints* database that is later used to locate people or objects within the mapped environment. Scene analysis methods have the advantage that can adapt to complex environments and provide acceptable performances when a sufficient number of base stations is available. These methods are also computationally inexpensive, but on the other hand they need a time consuming setup phase that needs to be repeated every time there are substantial changes in the environment (e.g. new base stations are added, large piece of furniture are moved, etc.).

Chapter 4

Problem Statement

As seen in previous sections, in most WSN applications a localization service is required to provide each node with knowledge of its position. Determining such information is a challenging task since the problem is computationally hard (Chapter 2) and an approximate solution must be found using only a limited amount of initial data (e.g. the known positions of a small set of anchor nodes or constraints derived from the radio connectivity among nodes). **This thesis addresses the localization problem in conditions where the amount of initial information is reduced to the minimum**, seeking to find a solution under realistic network configurations. For realistic configurations, we mean sensor networks suitable for real-world applications, where cost and implementation constraints often impose solutions that differ sensibly from those considered in many WSN simulation studies.

In this thesis, we address the problem of localization for networks with the following characteristics:

- **Small to medium number of nodes**: networks with ten to one hundred of nodes.
- **Sparse Deployments**: networks that have low density and where each node

has only a small number of radio neighbors.

- **Absence *a-priori* knowledge**: we assume the possibility that sensor positions have to be computed in the absence of initial information about anchor nodes (i.e. none of the sensor locations is initially known).

- **Resource constrains**: sensor nodes are equipped only with the hardware strictly necessary to execute the sensing application: a set of transducers, a small microcontroller for data processing and a radio transceiver for wireless communication. None of the nodes has a GPS devices or other hardware for accurate range measurements.

- **Anisotropic layouts**: we assume that networks can be deployed in environments where the presence of large obstacles (e.g. buildings, lakes or mountains) results in a layout with holes or other irregularities.

Finding an algorithm capable of working under these assumptions represents an interesting research challenge because it would enable localization in low-cost networks suitable for many application scenarios. Many of the schemes presented in Chapter 3 have been validated using simulations that consider large networks with high connectivity and a large number of anchor nodes. Although in the early stages of research, the "Smart-Dust" concept envisioned networks with tens of thousands of tiny sensor nodes, recent experiences with a few large-scale deployments [5] (about one thousand nodes) have demonstrated that current technology still has to mature before that vision will become a reality. Beyond technological issues, the size of actual WSNs is limited by economic factors and the high costs related to deploying and maintaining large-scale sensor networks are affordable only in specific application fields (e.g. military applications). According to a recent survey [10], many of the WSN applications deployed in the next few years will exploit small sensor networks with less than one hundred sensor nodes; therefore, implementing a localization scheme capable of working with these kinds of networks provides a solution to a problem of practical interest.

Similar considerations apply to the network connectivity parameter, i.e. the density of nodes in a WSN. Placing a high number of nodes in a small area serves to achieve great spatial-resolution, but again, cost considerations often limit the number of units available and may force sparse deployments where each sensor is in radio range with only a small number of neighbors. Furthermore, high spatial (or temporal) resolution might not be required in applications where the dynamic of the phenomena to monitor varies slowly with time and space (e.g. in environmental monitoring applications or precision agriculture). As seen in Chapter 2, networks with small numbers of nodes and low connectivity are intrinsically hard to localize because the number of constrains that can be inferred are lower. While for many communication protocols and other network services (e.g. service directories, distributed databases) simulating large, high-density networks serves to validate the scalability of the proposed approach, we note that the true challenge for a localization scheme is to maintain acceptable performance when the number of nodes and network density decreases.

In addition to considering networks with low connectivity, this thesis addresses the problem of localization in networks where none of the nodes possess ranging capabilities. The range-free approach is appealing because it does not require nodes with special hardware for range measurements (e.g. ultrasound) and therefore can be implemented in networks consisting of inexpensive sensor nodes. Although several range-free schemes have been proposed, many of them (Chapter 3) rely on the presence of a large number of nodes at known positions (anchor nodes). The purpose of this thesis is to further reduce the assumptions on the system requirements and investigate an anchor-free scheme capable of computing the sensor positions in the absence of anchor nodes. Enabling such a scheme would not only decrease the system complexity (no GPS devices are needed on anchor nodes), but would also improve the reliability and robustness of the scheme since it would eliminate the dependence on a few special nodes. In many anchor-based approaches, the number of anchor nodes in the network heavily affects the localization performance, therefore even the failure of a small subset of them can sensibly decrease the accuracy.

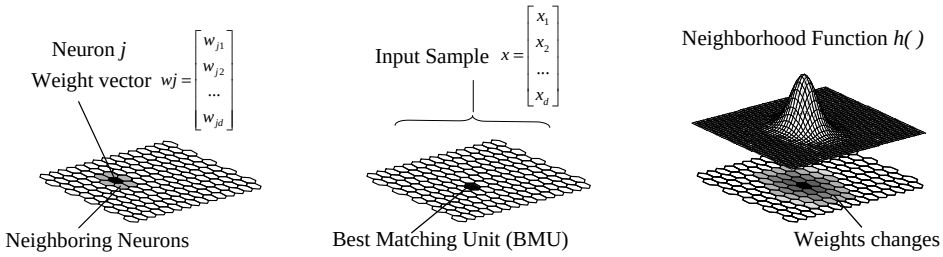
We finally complete the list of realistic system assumptions, by taking into consideration the problem of localization in networks affected by an anisotropic layout or where the connectivity among nodes (i.e. the maximum communication range) varies within the networks. Under these conditions, some schemes fail to produce acceptable results and the problem needs to be solved by computing small local maps that are later stitched together into a global map. The "map-stitching" technique greatly increases the solution complexity and is susceptible to large errors when the network connectivity is low, thus a solution robust to layouts irregularities would greatly simplify the implementation of a localization service in real deployments.

While the main focus of this thesis centers on the study of a novel range-free, anchor-free solution capable of localizing networks with low connectivity or anisotropic layout, we also want to evaluate the use of inexpensive switched antennas as a way to infer angle position in sensor network applications. Part of this work addresses the characterization of one such antenna with the purpose of implementing a simple solution to derive angle information in WSNs. This information will serve in future extensions of the localization work to improve the overall accuracy of the algorithm.

Chapter 5

Range-Free, Anchor-Free Localization

In this chapter we propose an range-free, anchor-free localization method capable of computing the node coordinates without relying on range measurement or anchor nodes. The solution is based on neural network paradigm known as *Self-Organizing Maps* (SOMs) [44, 45]. Introduced in the early 80's, these maps have found numerous applications in many areas such as speech recognition, data mining and bioinformatics ([42, 73] contain an extensive bibliography of SOM papers). In the next sections, after introducing the map structure and the learning algorithm, we show how the SOM formalism leads to an intuitive solution of the localization problem. Unfortunately, despite the attention received, SOMs have proved to be surprisingly resistant to mathematical characterization and convergence results are only available for the case of one-dimensional configuration of neurons [22], therefore we use extensive simulations (Chapter 6) to characterize the localization results and to compare our solution to the MDS technique.



(a) SOM (b) BMU Election (c) Adaptation

Figure 5.1: a) A two-dimensional map with the unit forming an hexagonal pattern, b) the first step of the learning algorithm: the unit whose weight vector is more similar to the input weight is elected as best matching unit (BMU). c) the adaptation phase: the weights of neurons around the BMU are adapted toward the input sample. A Gaussian-shaped neighborhood function controls the degree of adaptation.

5.1 Self-Organizing Maps

A SOM [44] is a neural network that learns application information as a set of weights associated with the neurons (nodes). In comparison with other techniques (e.g. Multi-Layer Perceptron), SOMs are unique because the neurons are arranged in regular geometric structures, typically two-dimensional lattices with rectangular or hexagonal patterns like the one in Figure 5.1a. As we will soon see, this spatial arrangement plays a central role in the training process of the maps and results in a topological organization of the information learned[1].

The training of a SOM is performed in an *unsupervised* fashion: the map is able to learn the underlying properties of the training set without the aid of labeled samples or reward functions (hence, they are characterized as "self-organizing"). Assuming that the input samples and the map weights w_j's are d-dimensional real valued vectors, the three phases of the training algorithm are as follows:

1. **Sampling**: A sample is extracted from the training set and presented to the network. We use the notation $x(n)$ to denote the sample at current iteration.

[1]This model vaguely resembles the structure of the cerebral cortex, where neurons are placed on a 2D surface and interact preferentially over lateral synaptic connections.

2. **Competition**: The sample $x(n)$ is compared with the map weights (there is one weight per neuron) through the use of a discriminating function $f = f(x, w)$. The neuron that scores the maximum value wins the competition and become the *Best Matching Unit* (BMU). If the discriminating function is implemented using the Euclidean distance, the election rule is given by:

$$BMU(n) = \arg\min_j \|x(n) - w_j(n)\| . \tag{5.1}$$

3. **Adaptation**: Finally, the weight vectors of the BMU and its neighbors are adapted according to the following rule:

$$w_j(n+1) = w_j(n) + \eta(n)\, h(j, BMU(n))[x(n) - w_j(n)]. \tag{5.2}$$

The update formula in 5.2 is controlled by the global learning rate parameter η and by a neighborhood function $h = h(i, j)$ (see Figure 5.1c). For ensuring convergence, the learning rate η must decrease monotonically with the number of iterations. A common choice is to implement the learning rate as an exponential function that decays from η_{max} to η_{min} over a given number of iterations. Typically, η decreases within the range $[\eta_{max}, \eta_{min}] = [0.1, 0.01]$, while the number of iterations goes from few hundreds to several thousands depending on the size of the training set.

The update rule is also controlled by the neighborhood function $h = h(\cdot, \cdot)$. This function regulates the weight changes on the basis of the map distance between BMU and the neuron being adapted. In the case of a Gaussian shaped neighborhood function, the expression of h is given by:

$$h(i, j) = \exp\left(-\frac{dist_{map}(i, j)^2}{2r(n)}\right), \tag{5.3}$$

where $dist_{map}(i, j)$ measures the distance on the map between two neurons. According to this expression, the magnitude of the changes is maximum for the BMU and decreases for units that are far from it. The extent of the area affected by the changes

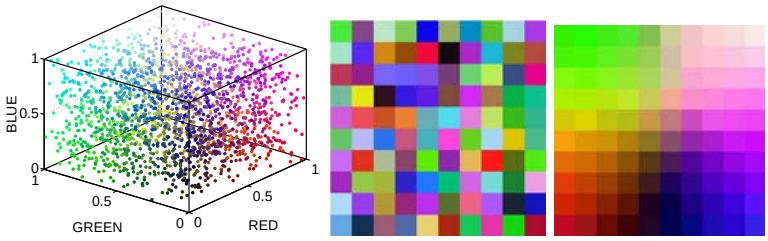


Figure 5.2: 10×10 **SOM trained with samples from the RGB color space: a) input space, b) initial weights, c) final weights.**

depends on the radius $r(n)$, a global parameter that controls the "width" of the neighborhood function. As in the case of the learning rate, the value of $r(n)$ decreases with the number of iteration: a relatively large radius during the initial iterations allows the map to quickly organize the neurons, while a smaller value toward the end determines localized changes, such that different parts of the map become sensitive to different input features.

The SOM technique is simple yet effective in capturing the properties of the input space and organizing them in an ordered fashion. An example of the SOM method in action is reported in Figure 5.2, where a 10×10 rectangular map is trained with random samples $x(n) = [r_n, g_n, b_n]$ from the RGB color space (Figure 5.2a). In this case, the weight vectors have the form $w_j = [r_j, g_j, b_j]$ and can be displayed using the corresponding color. Figure 5.2b shows the initial configuration of randomly assigned weights. After training the map with a few thousand random samples, the SOM assumes the configuration shown in Figure 5.2c. The result shows that among the 2^{24} colors of the input space, not only the map was able to select 100 representative samples (SOMs are vector coding techniques), but it also generated a topologically ordered representation of the color space, in the sense that similar colors were mapped to nearby locations. This property emerges as a consequence of the update rule: since adjacent neurons are subjected to similar weight changes, they eventually converge to similar values.

5.1.1 Topological Properties

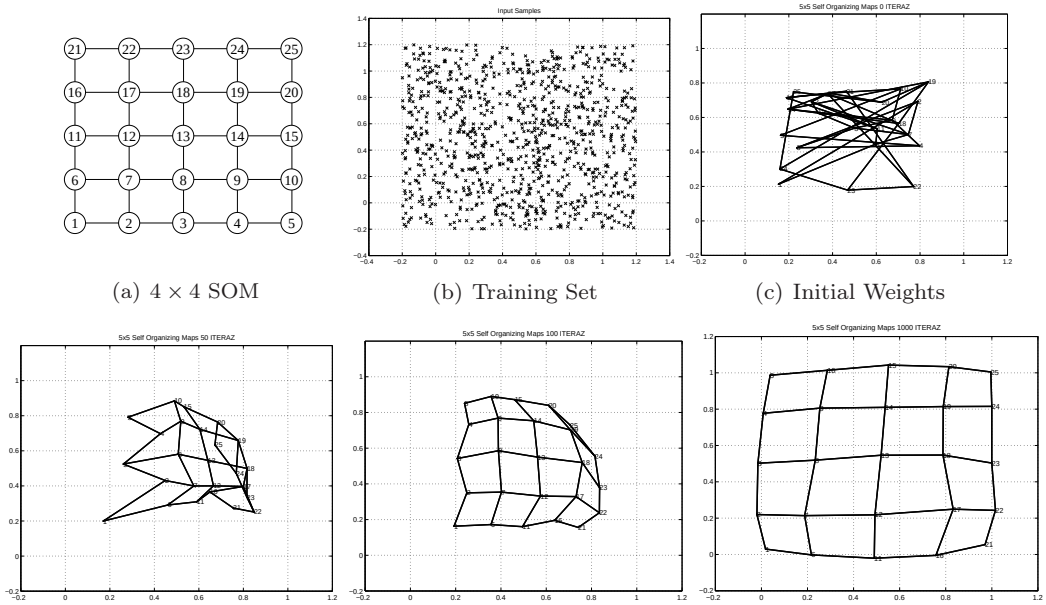
In the previous algorithm, if both the learning rate and the "width" of the neighborhood function are decreased monotonically with the number of iterations, the weights of the SOM converge to a stable configuration.

As noted in the example of Figure 5.2, the properties of the learning algorithm are such that the final weights ultimately produce a *topologically ordered* description of the input space. In order to provide and a further exemplification of this property, we will consider the case of a bidimensional 4×4 SOM with the neurons arranged in a rectangular pattern (Figure5.3.a). The weights w_j associated to each neuron j are two-dimensional real valued vectors, $w_j = [x\ y]'$, and are trained used samples from an input set $A \subset R^2$ (Figure 5.3b). In this case, since the dimensionality is two, we can plot the value of the weights in the plane: Figure 5.3c shows the initial configuration (weight randomly assigned). Segments are used to connect the weights that belong to adjacent neurons on the map. Figures 5.3c to 5.3d show the weights configuration after presenting 50, 100 and 1000 training samples. We note that as the training algorithm executes, the weights spread to adapt to the values from the input space, with the final position that are ordered with respect to each others.

5.2 Localization using SOMs

At the end of the training phase, the neurons contain model vectors that are representative of the input space, therefore the map can be used as a *codebook* for arbitrary samples. The code is given by the weight vector that best matches (BMU) the given sample. In addition, since each BMU defines a position on the two-dimensional grid, SOM implements a projection technique[2] from the input space to the plane defined by the lattice of neurons (see Figures 5.4).

[2]In this sense SOM can be seen as non-linear version of the Principal Component Analysis (PCA) technique.



(a) 4×4 SOM (b) Training Set (c) Initial Weights

(d) 50 iterations (e) 100 iterations (f) 1000 iterations

Figure 5.3: 4x4 SOM trained with random inputs from a two-dimensional training set.

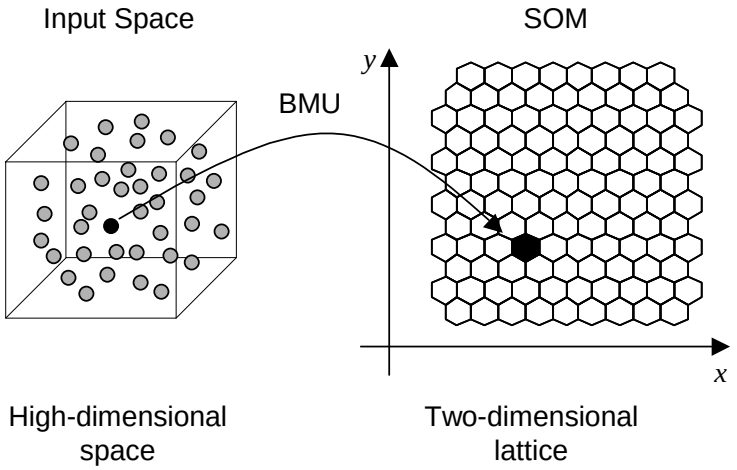


Figure 5.4: Mapping from a high dimensional input space to a two-dimensional lattice of neurons.

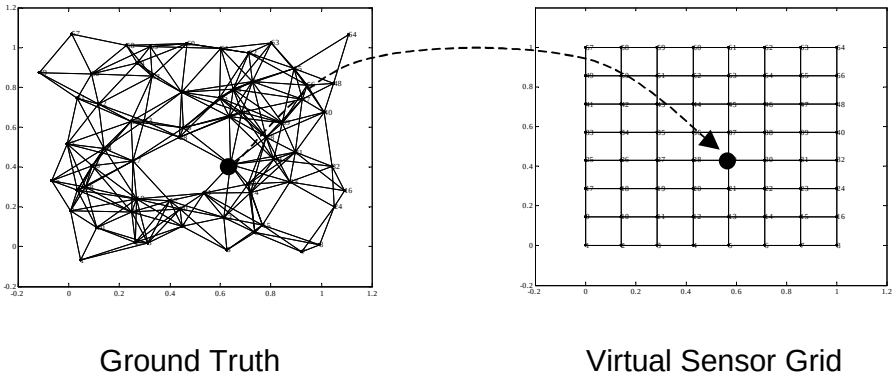


Figure 5.5: Mapping of sensor positions to a virtual sensor grid as defined in [26].

The dimensionality reduction property has been widely exploited in many applications for data analysis and visualization of large data sets [42, 73]. More recently, SOMs have been used to implement localization schemes for mobile robots in unknown environments [38, 27]. The SOM, initially trained with information collected by onboard sensors during the exploration phase, is then used as a virtual map to translate new sensor readings into grid positions or to recognize different environments (e.g. different rooms).

Ertin and Priddy [26] have used a similar approach to solve the localization problem in WSNs. In their work, synchronous readings collected by all the sensor nodes are used to build the training set for the SOM. After training the model, the localization task is performed using new sensor readings to sort nodes on the basis of their proximity to a virtual grid of nodes (see Figure 5.5). Although no attempt is made to compute individual node positions, the authors suggest possible applications to the target tracking problem.

Our solution is similar to [26] in the sense that it is also based on the SOM formalism, but the approach taken is rather different since it does not rely on sensor readings or time synchronization services. In addition, our scheme explicitly computes individual node positions as a result of the training phase of the map.

The intuition behind the proposed solution is that, with no prior information on sensor locations, the best assumptions we can make are:

1. **Uniform coverage.** Sensor nodes provide an (approximately) uniform coverage of the deployment area.
2. **Neighbors Proximity.** Nodes that are within their radio range are relatively close to each other

In Section 6.6, we consider non-uniform deployments and the effect of irregular radio patterns, nevertheless the two assumptions (uniform coverage, radio neighbors close to each other) are realistic for many WSNs and are useful to give an intuitive illustration of our approach. To solve the problem we must therefore generate a location assignment that is approximately uniform, taking care of placing neighbor nodes close to each other. This is accomplished by associating the unknown node positions (x_i, y_i) to the weights of a SOM and then training the model with random samples from an uniform distribution. As a result of the training phase, the weights (i.e. nodes position) will eventually spread to cover the sampling area and, if associated to adjacent neurons on the map, neighboring nodes will be kept close to each other.

Using an approach substantially analogous to the one exposed here, SOMs has been previously applied to graph drawing [61, 12], a branch of graph theory that deals with the visualization of complex graphs. The graph layout problem is similar to the localization problem in the sense that it also seeks to find a coordinate assignment such that vertices connected by edges are positioned close to each other. However, while the evaluation of a graph layout is mostly based on aesthetic factors (e.g. uniform distribution of nodes and edge lengths, separation between graph elements, number of edge crossing, etc.), the results of the localization assignment are directly comparable with the true sensor locations. In this work we explicitly evaluate the effectiveness of SOM in producing maps similar to the ground truth and we focus on reducing the localization error.

5.3 System Model

We consider a connected network with N nodes placed at unknown locations in the 2D space: $(x_i, y_i)_{i=1,\ldots,N}$. None of the nodes is equipped with hardware for position, range or angle estimation (e.g. GPS, ultrasound receivers or smart antennas) and no assumption is made regarding availability of sensors at each location. We only assume that every node can determine the set of its radio neighbors[3] and can transmit this information to a central point of computation. Also, during the *neighbor discovery* phase, nodes use the same transmission power in the effort to ensure an approximately uniform communication range. Once the connectivity information is known, the network can be represented as an undirected graph $G_{net} = (V, E)$, where two vertices are connected if the corresponding nodes are radio neighbors. The graph also serves to introduce the hop distance metric $d = dist_{hop}(\cdot, \cdot)$ defined as the length of the shortest path connecting two nodes.

5.4 Modified SOM Model

The core of the SOM technique is the update rule defined in (5.2). In that expression, the neighborhood function $h(\cdot, \cdot)$ takes into account the spatial arrangement of the neurons through the map distance $dist_{map}(\cdot, \cdot)$. Now we note that, as long as a distance function between two elements on the map is provided, the regular lattice can be replaced by an arbitrary structure of interconnected neurons (see Figure 5.6). Consequently, we modify the original SOM architecture by using the network graph in place of the lattice of neurons and exchanging the map distance with the hop distance $dist_{hop}(\cdot, \cdot)$. The new neighborhood function is given by:

$$h(i, j) = \exp\left(-\frac{dist_{hop}(i, j)^2}{2r(n)}\right). \tag{5.4}$$

[3] By neighbors, we mean *symmetrical* radio neighbors: messages from node j are received by i and vice versa.

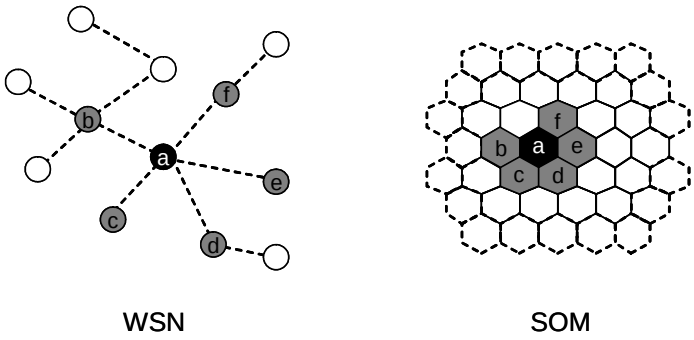


Figure 5.6: Correspondence between a WSN and the SOM structure. Intuitively, the localization algorithm maps the neighboring nodes of a WSN to the neighboring neurons of a SOM.

Having defined the new neighborhood function, the training algorithm illustrated in Section 5.1 can be applied to the localization problem. In this modified SOM model, neurons are located on the vertices of G_{net}, hence we have a direct correspondence between the neurons and the network nodes. The weight vector associated with each neuron/node j has the form $w_j = (x_j, y_j)$. *This vector, initially picked at random, will eventually contain the estimated location for the corresponding node.*

5.5 Localization Algorithm

Since the proposed algorithm is centralized, each node needs to communicate the list of its radio neighbors to the unit in charge of the computation. This information is necessary first to build the adjacency matrix G_{net}, and then to compute the hop-count distances between each pair of network nodes, which are stored in a matrix H_C with elements given by $\{hc\}_{i,j} = dist_{map}(i, j)$. The matrix H_C is the only input parameter required by the localization algorithm (Figure 5.7).

According to the scheme of Section 5.1, the weight vectors (x_j, y_j) are initialized with random numbers and then trained with a set of input samples. Since we are using only connectivity information, we are free to work in a relative reference system where absolute coordinates are not important. In light of this model, we can easily generate

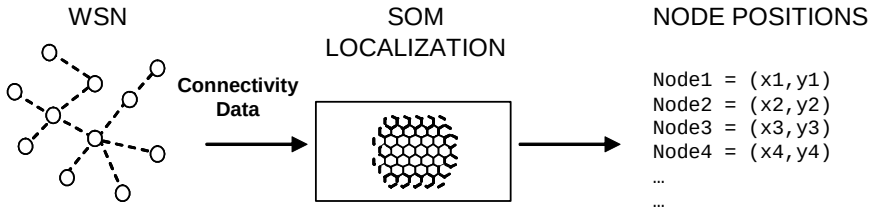


Figure 5.7: Input and output parameters of the SOM localization scheme.

the training set by sampling random points from an arbitrary uniform distribution (e.g. $0 \leq x, y \leq 1$). This fact greatly simplifies the implementation of the algorithm since the localization task can be performed without having to rely on any other external information (e.g. network's physical dimensions or sensor readings like in [26]).

Algorithm 1 contains the pseudo-code of the localization scheme. In the proposed scheme, the learning parameter $\eta(n)$ and the radius $r(n)$ are decreased linearly with the number of iterations (see lines 7 and 8). As a side note, we mention that, as the radius $r(n)$ shrinks, the level of adaptation for neurons far from the BMU becomes negligible, so the update rule (line 13) can be more efficiently restricted to neurons within a short hop distance from the BMU. In Section 6.4, after defining the simulation setup and evaluation metric for the algorithm, we provide additional considerations on the weight initialization and the number of iterations required by the algorithm.

Algorithm 1: SOM Based Localization

Input: matrix H_c: hop count distances among nodes
Output: $(x_j, y_j)_{j=1,\ldots,N}$: node positions

% Initialization

$[\eta_{max}; \eta_{min}] = [0.1; 0.01]$
$[r_{max}; r_{min}] = [\frac{\max H_c}{2}; 0.001]$

for all nodes n **do**
  $(x_n, y_n) =$ **random**()
end for

% Main Loop

for $n = 1$: to N_ITER-1 **do**
  $\eta(n) = \eta_{max} - n[\eta_{max} - \eta_{min}]/(\text{N\_ITER} - 1)$
  $r(n) = r_{max} - n[r_{max} - r_{min}]/(\text{N\_ITER} - 1)$

  $(x, y) =$ **random**()
  $BMU = \arg\min_j \|(x, y) - (x_j, y_j)\|$
  for all network nodes j **do**
    $h = \exp\left(-\frac{H_c(BMU,j)^2}{2r(n)}\right)$
    $(x_j, y_j) \mathrel{+}= \eta(n)h[(x, y) - (x_j, y_j)]$
  end for
end for

Chapter 6

Implementation Details and Simulation Results

In this chapter we report the implementation details of the algorithm proposed in the previous chapter and the results of the simulations used to validate it. Since we are interested in evaluating our scheme's performance in localizing small to medium size networks (10 - 100 nodes) with low connectivity, we need to impose some constraints on how the random topologies are generated. We refrain from purely random deployments (coordinates selected as *i.i.d.* random numbers) for two reasons:

1. It is unrealistic to assume that nodes will be positioned independently from each other. Even when nodes are scattered from an aircraft, the probability of two nodes ending in the same location is very low. In this case, the probability of a node to be located in a specific location depends, at least, on the time at which the node is thrown from the airplane and therefore varies from node to node. If nodes are manually deployed, even if accurate control over the final locations is not possible, it is reasonable to assume that nodes will be placed to provide a roughly uniform coverage of the sensing area.

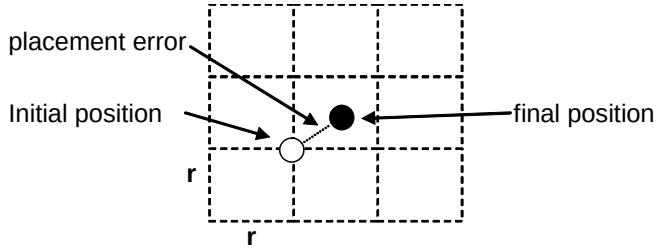


Figure 6.1: The regular grid used to generate the initial node positions.

2. In purely random deployments, the probability to obtain connected networks rapidly decreases to zero as we reduce the communication range [47].

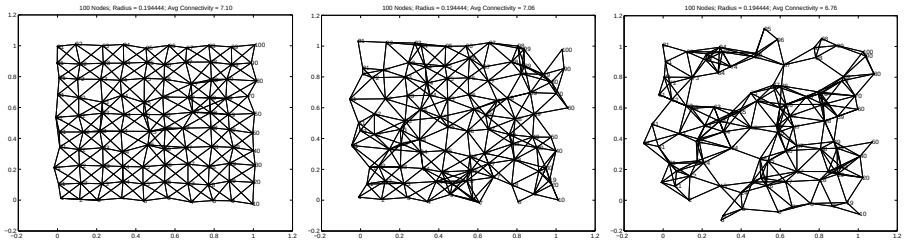
Since it is difficult to generate meaningful low-connectivity topologies, we consider a model in which the node density is kept roughly uniform by having the nodes positioned on the intersection points of a grid with rows and columns spaced by a factor **r**. We capture the nature of an ad hoc deployment by perturbing the positions with random noise and allowing for large placement errors.

6.1 Simulation Parameters

The parameters used to generate our simulation are the following:

- **Number of nodes, N**: We simulated networks with 16, 25, 36, 64, 81 and 100 nodes. Although the simulations were executed using a number of nodes equal to powers of 2, the algorithm can work with arbitrary network sizes (e.g. see Section 6.6).

- **Placement Error**, $\sigma_{\mathbf{PE}}$: The initial positions are given by a regular grid of $\sqrt{N} \times \sqrt{N}$ elements spaced by a factor r (see Figure 6.1). Node positions are obtained by perturbing the grid positions with Gaussian noise having zero mean and standard deviation $\sigma_{\mathbf{PE}} = \{0.1, 0.2, 0.3, 0.4, 0.5\}\, r$. Figure 6.2 reports three topologies for different values of the placement error.

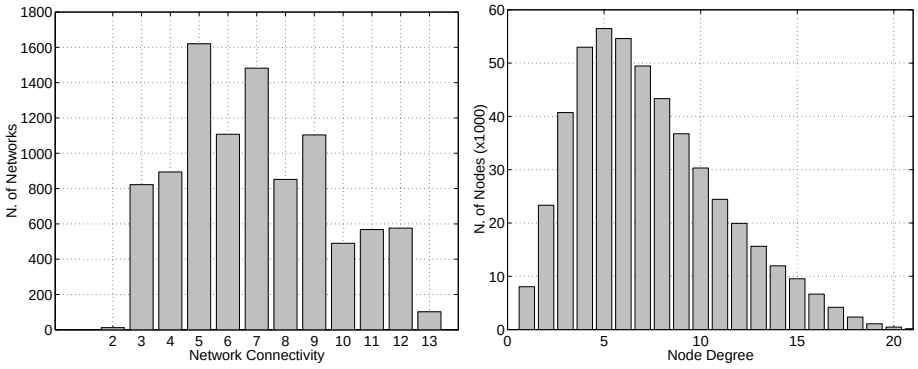


(a) $\sigma_{PE} = 0.1r$ (b) $\sigma_{PE} = 0.2r$ (c) $\sigma_{PE} = 0.5r$

Figure 6.2: Three 100-node networks with different placement errors.

- **Communication Radius, R**: The maximum communication radius is chosen as a function of the spacing factor r: $R = \{1.25,\ 1.5,\ 1.75,\ 2.0,\ 2.25\}\, r$. In this first set of simulations, we consider two nodes as neighbors if their distance is less than R. We analyze the effect of irregular radio pattern in Section 6.6.1.

The algorithm was evaluated by generating 50 networks for each combination of the above parameters. After discarding disconnected networks, the number of simulated to-pologies is 9630, with an average connectivity of 6.98. The results presented in the following sections were obtained by executing 2000 iterations of the algorithm presented in Section 5.5.



(a) (b)

Figure 6.3: Distribution of the average network connectivity (a) and node degree (b) in the simulation experiments.

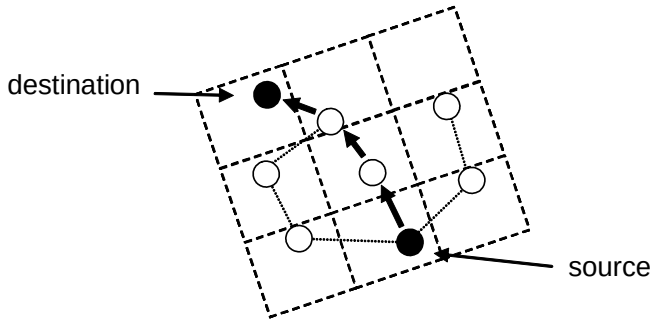


Figure 6.4: Geo-routing using virtual coordinates. At each step, the scheme forwards the packet to the node whose virtual coordinates are closer to the destination.

6.2 Virtual Coordinates

The SOM based scheme is truly an anchor-free, range-free algorithm in the sense that it can generate virtual coordinates without relying on anchor nodes or distance measurements. Since the virtual coordinates cannot be compared to the true network coordinates, we use the delivery ratio of a *greedy* routing algorithm as evaluation metric. At each hop, the routing scheme forwards the packet to the neighbor node that is closer to the recipient of the message (see Figure 6.4), according to the rule:

$$next\_hop = \arg\min_{n} \|(x_n, y_n) - (x_{dest}, y_{dest})\|\,,$$

where (x_n, y_n) are the virtual coordinates of the neighboring nodes and (x_{dest}, y_{dest}) are those of the destination. Although the scheme is extremely unrefined (it simply gives up if it is unable to get closer to the destination), it is still useful to define a baseline for the performance achievable using more advanced schemes (e.g. GPRS [41]).

Using the simulation setup previously introduced, we have compared the performance of our approach with the results of MDS, a popular projection technique that has been successfully applied to the localization problem in WSNs [82, 81, 39]. Figure 6.5 reports the percentage of packets successfully delivered using the greedy algorithm that operates on the basis of the relative maps generated by SOM and

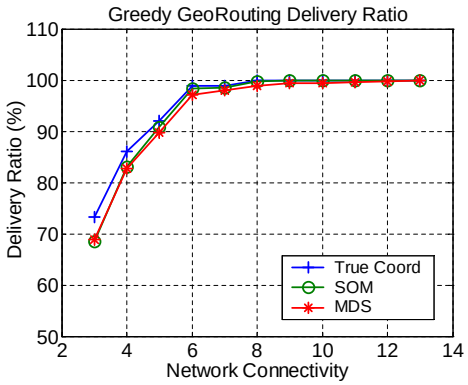


Figure 6.5: Delivery Ratio of a greedy geo-routing scheme using virtual coordinates.

MDS. The results show that the virtual coordinates produced by both methods are effective when used for geographical routing, with a delivery ratio that is very close to that obtained using the true network coordinates. Similarly, the length of the routing path does not differ substantially from the case where the true node positions are known to the routing scheme (graph is not shown).

6.3 Absolute Coordinates

As seen in the previous section, virtual coordinates can be computed solely on the basis of connectivity information and are useful for important network tasks such as packet routing. Nevertheless, there are applications where absolute positions are required (e.g. a WSN to support a first responder team that needs to quickly locate the emergency scene). In order to convert relative node positions into absolute coordinates, at least three anchor points are needed for the bidimensional case. In this section, we have used four anchor nodes on the perimeter of the map to resolve rotational, translational and flipping ambiguities and align the map to an absolute coordinate system. The linear transformation to align the virtual coordinates to an absolute reference system is found, in general, by considering the position of a set of m anchor nodes:

$$A = \{a_1, a_2, ..., a_m\}.$$

For each anchor node, let $(x_{a_i}, y_{a_i})_{a_i \in A}$ be the true coordinates and $(p_{a_i}, q_{a_i})_{a_i \in A}$ be the virtual coordinates computed by the localization algorithm. To compute the 2×2 matrix L that defines the linear transformation, we need to remove the mean value from the two set of coordinates[1] (the true coordinates (x_{a_i}, y_{a_i}) and the virtual coordinates (p_{a_i}, q_{a_i})) and then solve the overdetermined linear system of equation given by:

$$\begin{bmatrix} (p_{a_1} - \bar{p}_a) & (q_{a_1} - \bar{q}_a) \\ (p_{a_2} - \bar{p}_a) & (q_{a_2} - \bar{q}_a) \\ \dots & \dots \\ (p_{a_m} - \bar{p}_a) & (q_{a_m} - \bar{q}_a) \end{bmatrix} = \begin{bmatrix} l_{11} & l_{12} \\ l_{21} & l_{22} \end{bmatrix} \begin{bmatrix} (x_{a_1} - \bar{x}_a) & (y_{a_1} - \bar{y}_a) \\ (x_{a_2} - \bar{x}_a) & (y_{a_2} - \bar{y}_a) \\ \dots & \dots \\ (x_{a_m} - \bar{x}_a) & (y_{a_m} - \bar{y}_a) \end{bmatrix},$$

where the average values are the following:

$$\begin{aligned} (\bar{x}_a, \bar{y}_a) &= \frac{1}{m} \sum_{i=1}^{m} (x_{a_i}, y_{a_i}), \\ (\bar{p}_a, \bar{q}_a) &= \frac{1}{m} \sum_{i=1}^{m} (p_{a_i}, q_{a_i}). \end{aligned}$$

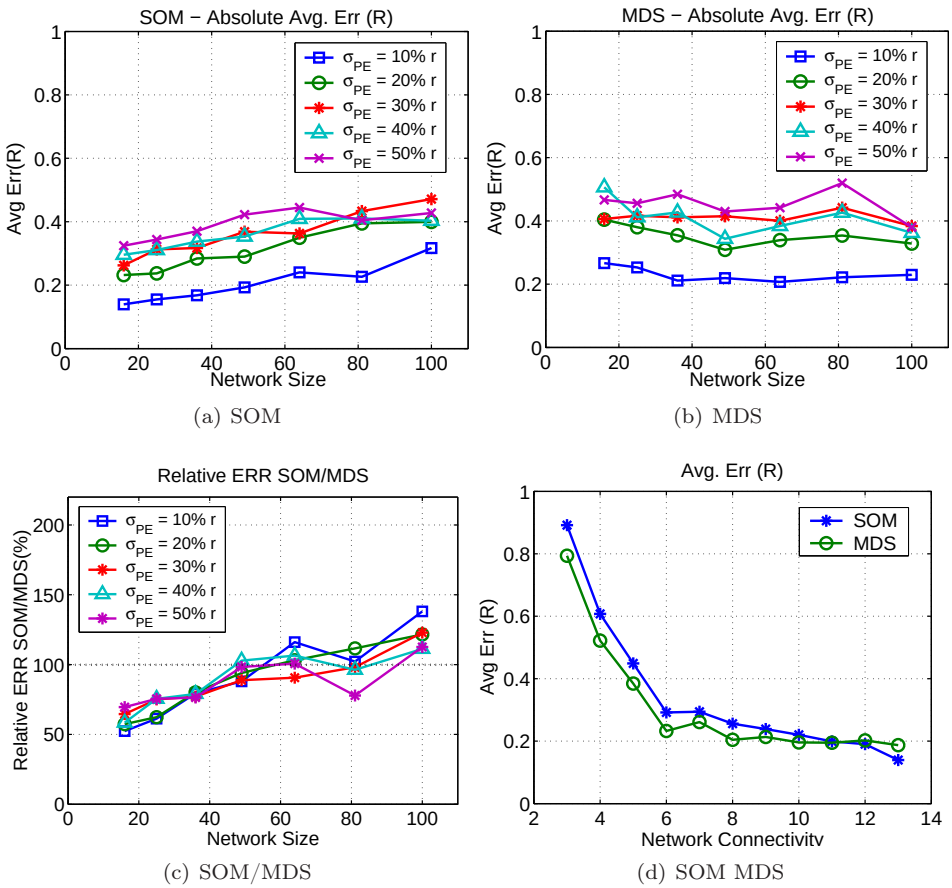
The matrix L, which is obtained from the *least-squares* solution of the system above, is then used to convert the virtual coordinates (p, q) in estimates of the true coordinates as follows:

$$\begin{aligned} \hat{x}_i &= \bar{x}_a + l_{11} (p_{a_i} - \bar{p}_a) + l_{21} (q_{a_i} - \bar{q}_a), \\ \hat{y}_i &= \bar{y}_a + l_{12} (p_{a_i} - \bar{p}_a) + l_{22} (q_{a_i} - \bar{q}_a). \end{aligned}$$

As a result of this transformation, the estimated positions $(\hat{x}_i, \hat{y}_i)$ can be compared with the true positions (x_i, y_i) and the localization error can be expressed quantitatively:

$$\text{Avg. Err(R)} = \frac{1}{\text{R}} \frac{\sum_{i=1}^{N} \sqrt{(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2}}{N},$$

[1] By doing this we first resolve the affine transformation given by an eventual translation of nodes. After this operation we can determine the linar part of the transformation.



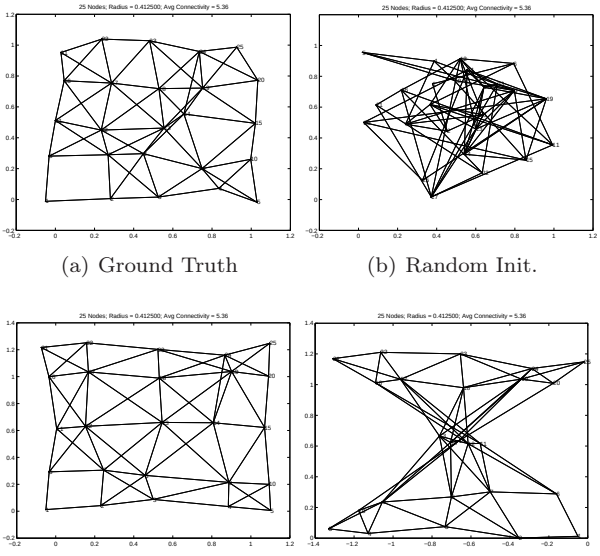
(a) SOM (b) MDS

(c) SOM/MDS (d) SOM MDS

Figure 6.6: a,b) Average Error (R) as function of network size for SOM and MDS. c) Ratio between the error achieved by the two schemes. d) Average Error (R) as function of the network connectivity.

where N is the number of nodes in the network and R is the communication range.

Figures 6.6b and 6.6c show the error of SOM and MDS for different network sizes and placement errors. The error is expressed as a value relative to the communication range R. As expected, the accuracy of the localization schemes decreases as the placement error on the map increases. We note that while MDS works by actively using the hop-count distances between each pair of nodes, and thus it works better for larger and denser networks (where the number of constraints is higher), SOM is based on localized constrains and works better for smaller networks.



(a) Ground Truth (b) Random Init.

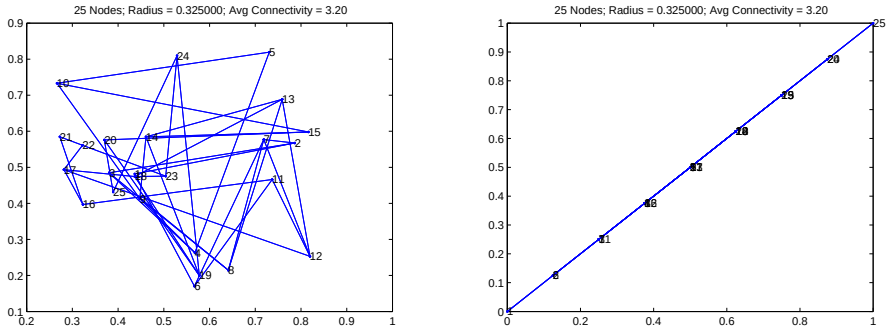
(c) Conv. OK (d) Conv. ERR

Figure 6.7: Localization convergence.

6.4 Weight Initialization and Convergence

Having defined the simulation parameters and the localization error, we analyze the effect of weights initialization and number of iterations on the algorithm's performance. Weight initialization influences both the convergence speed and the localization accuracy. In Sections 5.1 and 5.2 we stated that weights are initialized at random (usually with samples from the input set or other small values). In our simulations we have verified that this approach works well on average, but there are few occurrences where the final error is large ($> 1.0R$).

Figure 6.7 shows: a) a random topology b) the initial weight configuration and c) a case where the localization algorithm produced a substantially acceptable result. On the other hand, Figure 6.7d shows an occurrence where the algorithm failed to converge to an acceptable solution for the same network topology. Although the relative positions of the majority of nodes are correct with respect to each other, the network is "twisted", with the nodes of the upper half in inverse order respect to the lower half. Such problem is caused by unfortunate initial weight configurations that



(a) Random Initialization

(b) Modified Initialization

Figure 6.8: Weights Initialization.

determine a topological flipping of large blocks of nodes. In our experiments we found that the occurrence of such cases can be greatly reduced by initializing the weights with values lying on a straight line. The initialization rule is given by:

$$(x_n, y_n) = \frac{H_c(o, n)}{\max\limits_j H_c(o, j)}, \tag{6.1}$$

where o identifies a node placed on the perimeter of the map. According to the equation, weights are initially aligned along a line starting from $(0, 0)$, the position of node o, and ending at $(1, 1)$, the position of the node with maximum hop count distance from o (see Figure 6.8).

This scheme, which partially sorts the initial node positions, helps in reducing the final localization error as well as the occurrence of "twisted" networks. In our simulations we found that the proposed solution reduces the average localization error by about 43% with respect to random initializations, while the pergentage of networks with final error $> 0.5R$ is only 10.6%, against 31.5% for random initialization and 20.12% for MDS.

The localization accuracy also depends on the number of iterations used in the algorithm. Figure 6.9 reports the average localization error for a test set containing 100 topologies generated using the simulation parameters defined in Section 6.1. We

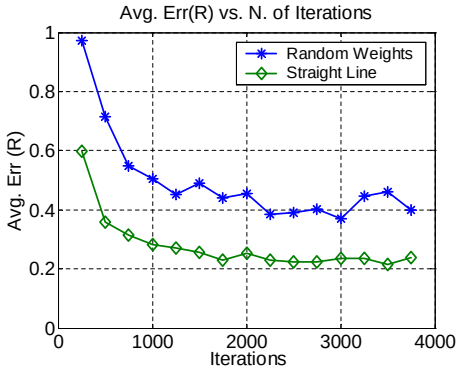


Figure 6.9: Average Error (R) as function of the number of iterations (results averaged over 100 random topologies).

note that error rapidly decreases during the first 500 to 1000 iterations and then only reduces marginally.

6.5 Exploiting Anchor Information

In previous sections, the maps generated by SOM and MDS have been scaled and oriented by using the position of four anchor nodes. However, the structure of our approach is such that anchors' information (if available) can be exploited *during* the training phase of the map, with valuable effects on the final results. The modification to the algorithm involve two points: **i)** coordinates of anchor nodes are never updated (since they are already correct) and **ii)** whenever an anchor node is elected as BMU, the sample at current iteration is replaced with the anchor's position. The two modifications have the effect that anchor nodes not only remain in their position, but they also facilitate the map organization during the initial iterations. In addition, if the number of anchors is equal or greater than three, the method generates absolute coordinates without needing any further transformations. We have evaluated the performance of the algorithm using *a priori* knowledge of three and four anchor nodes (SOM_3A and SOM_4A respectively). The results show a substantial improvement in the localization accuracy: using four anchor nodes during the computation (SOM_4A) reduces the average localization error by about 30% with

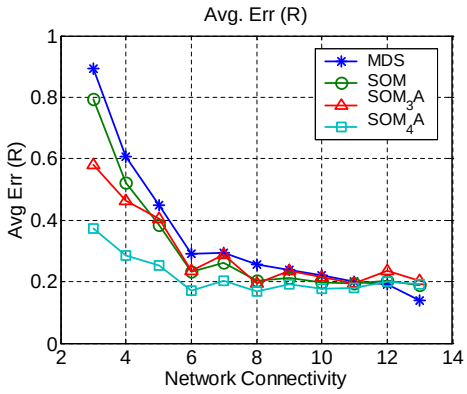


Figure 6.10: Average Error (R) as function of network connectivity using anchor information.

respect to the basic SOM algorithm, while the percentage of networks with localization error >0.5% drops to only 3.15% of the total cases. Figure 6.12a reports the results for various values of network connectivity. The plot shows that the SOM algorithm significantly outperforms MDS when the network connectivity is low, with an average error reduction of 43% for networks with connectivity between 3 and 10 (using SOM_4A). The explanation is that SOM is "less aggressive" in the use of node distances. The effect of the neighborhood function $h(\cdot,\cdot)$ is such that the distance constraints among nearby sensors are weighted more than those of nodes several hops away. Consequently, the SOM scheme is less sensitive to condition of low connectivity, where high values of the hop count distance between two nodes do not necessarily imply that nodes are far from each other.

6.6 Anisotropic Deployments

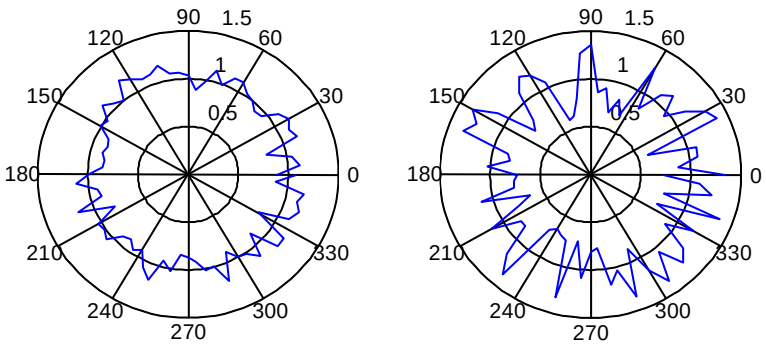
Our localization scheme has been derived under the assumptions of approximately uniform deployment and communication range. In this section we use simulations to evaluate the effect of irregular radio patterns and anisotropic deployment on the algorithm's performance.

6.6.1 Irregular Radio Pattern

The results reported in the previous section were obtained assuming an idealized radio model, where two nodes are neighbors *iff* their distance is equal to or less than the communication range R. This assumption is very strong and does not take into account the nature of radio propagation in the space. To get an insight on the effect of multi-path, scattering and shadowing on the transmission range, we repeated experiments using a less ideal radio model. In particular we took into account the influence of an additional parameter, the *Degree of Irregularity* (DOI) with values 0.2 and 0.4 (see Figure 6.11). A DOI equal to 0.4 means that the effective transmission range for each sensor is uniformly drawn from the interval [0.6R - 1.4R], where R is the average radio range. In Figures 6.12b and 6.12c we report the experimental results for DOI = 0.2 and DOI = 0.4, showing that the localization error does not significantly increase in conditions of irregular radio pattern (especially in the SOM_4A modification).

6.6.2 Anisotropic Networks

In addition to considering irregular radio patterns, we have simulated networks with anisotropic layouts resulting from the presence of large obstacles (e.g. buildings) in the region of the deployment. It is known that under such scenarios MDS, similar to the case of low connectivity, does not perform well. The reason is that MDS uses the hop count as a distance measure between each pair of nodes. While this approach works well when the path connecting two nodes lies approximately on a straight line,

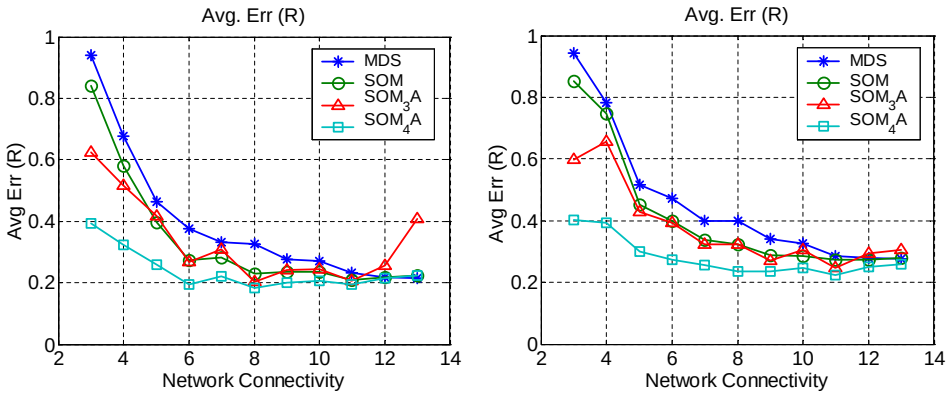


(a) DOI=0.2 (b) DOI=0.4

Figure 6.11: Effect of Degree of Irregularity (DOI) on the transmission range.

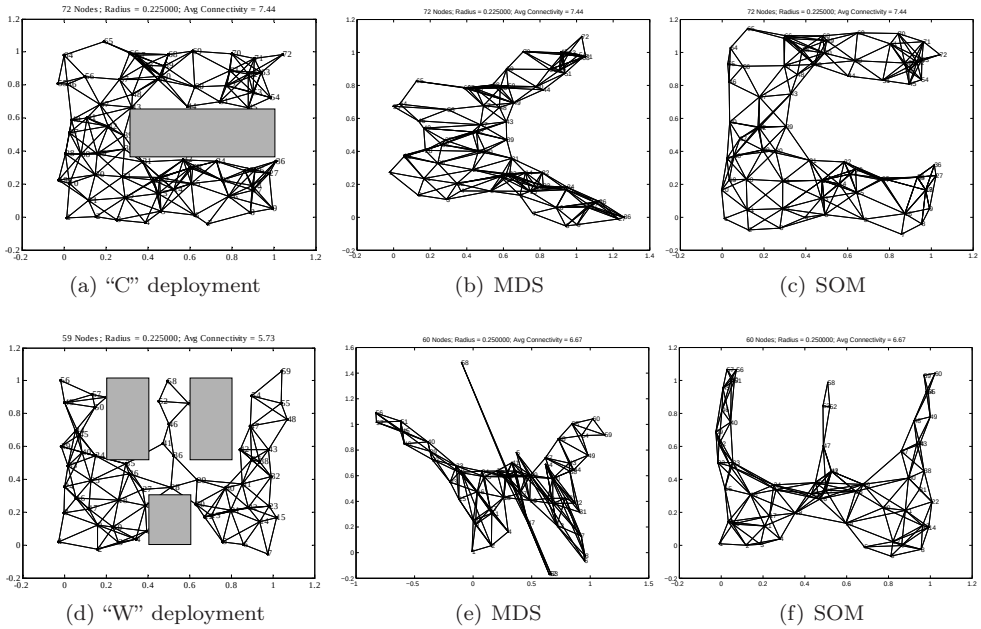
it generates large errors in the presence of obstacles. In this case two nodes can be physically close even if their hop distance is large.

The large error in the case of anisotropic networks has motivated alternative approaches where MDS is used to compute small local maps that are then stitched together into a global map [82, 39]. Although this approach can be useful to solve the problem in a distributed manner, the process of map stitching greatly increases the complexity of the solution and is susceptible to large errors when the network



(a) DOI=0.2 (b) DOI=0.4

Figure 6.12: Average Error (R) as function of network connectivity using anchor information.



(a) "C" deployment (b) MDS (c) SOM

(d) "W" deployment (e) MDS (f) SOM

Figure 6.13: Sample results for anisotropic layouts: in this case, the SOM algorithm reduces the average localization error of 75% with respect to MDS.

connectivity is low. It would be useful to have a scheme capable of localizing irregular networks without having to partition the map and encumber the complexity of map stitching. To validate the performance of the SOM algorithm we have evaluated anisotropic deployments obtained by randomly placing the nodes around few obstacles. Two sample topologies are represented in Figures 6.13a and 6.13d.

SOM and MDS were tested by simulating 200 random networks for both the "C" and "W" shaped maps with connectivity between 4.6 and 6.8. As in the previous simulation, we evaluated the average error after orienting the map using four anchor points on the perimeter. For the MDS localization scheme, the simulation results confirmed our expectation: the average final error was large in both cases, 1.56R for the C-shaped and 1.32R for the W-shaped network. The SOM algorithm did not suffer the same problem and produced results with the accuracy comparable to the case of uniform networks: 0.33R for the C-shaped network and 0.38R for the W-shaped one, with an average error reduction of 75% with respect to MDS. As explained previously,

the better results are due to the fact that SOM mainly exploits the constraints derived by neighbors nodes that are placed few hops away from each other; consequently, it does not incur in large errors trying to relate the position of nodes that are several hops away. Figures 6.13b, 6.13d, 6.13e and 6.13f present four sample maps generated by the MDS and SOM algorithm for the "C" and "W" topologies to give a qualitative illustration of the results.

6.7 Computational Complexity

Recently, several research efforts have been directed toward the study of *distributed* localization algorithms. This interest is motivated by the fact that centralized computation is not viable in the following circumstances: 1) the communication overhead to transfer the input data to a central unit is too high, 2) none of the devices in the system possess the computational resources to compute the whole solution, 3) the result is critical and introducing a single point of failure puts the reliability or security of the system in jeopardy. In this section we analyze the overhead of our scheme, showing that the SOM approach, although centralized, does not suffer from the above mentioned drawbacks and is suitable for highly constrained deployments consisting of nodes with limited hardware resources.

The algorithm operates on the basis of connectivity information, therefore each sensor needs to communicate the set of its radio neighbors to the unit in charge of the computation. Assuming that node IDs are coded using two bytes (up to 65536 nodes), the information can be transmitted using a fairly small size radio messages. For example, the average connectivity of the networks in our simulations was less than 7, thus, on average, only 14 bytes need to be transmitted by each node. Since the amount of data can be further reduced by means of data aggregation techniques, the overhead to transfer the initial information to the central node does not pose a problem for many cases of practical interest.

Having received the neighbor sets, the data is used to generate the adjacency matrix of the network graph requiring $[N(N-1)/2]/8$ bytes[2] and then to compute the table H_c with the hop count distances between nodes. The solution can be obtained by repeating N executions of the popular Dijkstra's algorithm or using the Floyd's scheme. The complexity is $O(N^3)$ in both cases, while the table needs enough storage space for $N(N-1)/2$ elements. The memory requirements for this table can be reduced by taking into account the maximum hop count distance between any two nodes (i.e. the network diameter). In our simulations, the average diameter was equal to 6.19 with a maximum value of 16. Using 4 bits to code the hop-count distances[3], the size of the table is reduced to $N(N-1)/4$ bytes of memory. Finally, we need to reserve the memory space to hold the coordinates of the sensor nodes (i.e. the SOM weights). Assuming, that each coordinate is represented with 2 bytes, the total occupation is $4N$ bytes. As for the computational complexity of our approach, the iterative solution allows a trade-off between accuracy and execution time (cf. Section 6.4). Each iteration determines the BMU (requiring N comparisons), and then applies the update rule 5.2 to the map weights. Considering that the radius of the neighborhood function shrinks from a value initially equal to the network radius and then goes to zero, the average number of weight updates is $N/2$.

While the algorithm executes in a few second on a PC, we have implemented a TinyOS [54] version to test the scheme on WSN nodes. The code was executed using TelosB [74], a low cost, commercially available sensor node. The board is equipped with Texas Instrument MSP430 F1611, a low power 16-bit RISC microcontroller featuring 10KB of RAM, 48KB of code memory and an internal oscillator working at the frequency of 8MHz. The algorithm was implemented as reported in Section 5.5, with the only exception that the Gaussian neighborhood function was replaced with a triangular function, which produces similar results using much less computation. Table 6.1 reports the memory occupation of the data structures described above and

[2] We recall that the graph is undirected, so both the adjacency matrix and the hop count table are symmetric.

[3] We note that even if some hop count distance exceeds the upper limit allowed, replacing this value with the upper limit does not have a noticeable impact on the algorithm because the interactions between units far from each other are very weak.

N. Nodes	Memory	Exec. H_c	Exec. 1000 iter.
36	0.42 KB	1 sec	62 sec
64	1.48 KB	6 sec	102 sec
100	3.42 KB	22 sec	156 sec

Table 6.1: Memory requirements and execution time of the SOM algorithm on a TelosB node.

the execution time to compute the table H_c and then to perform 1000 iterations of the localization algorithm.

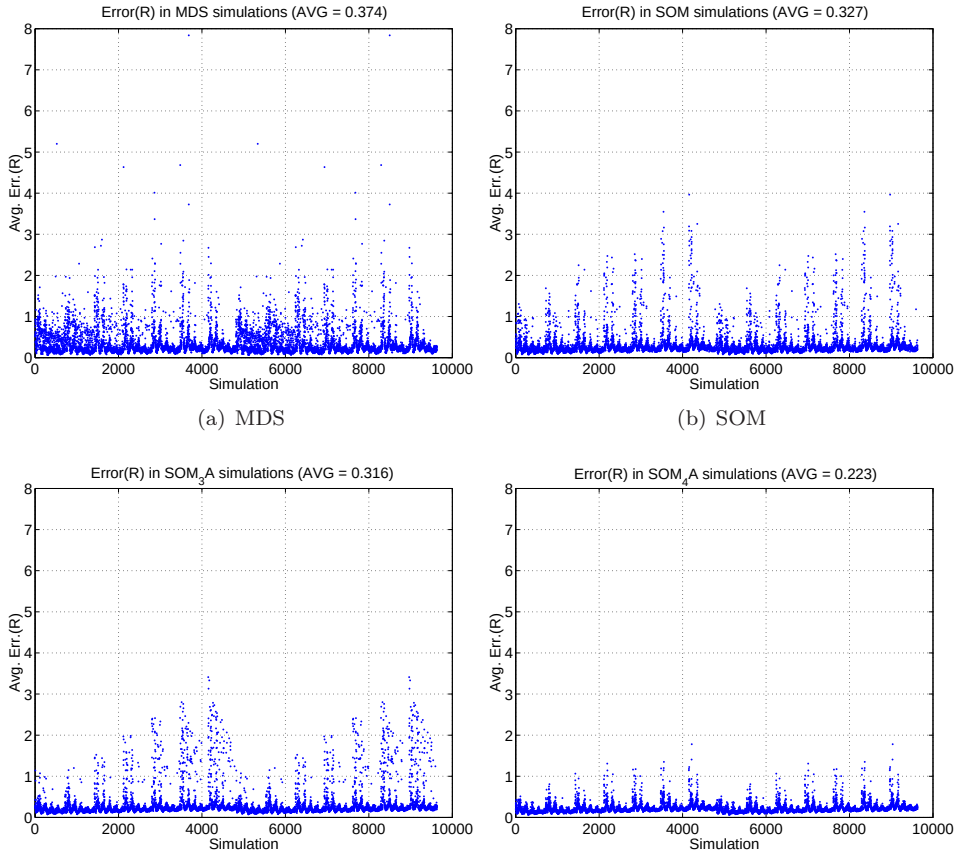
As can be seen from the table, the limited hardware resources of an inexpensive sensor node are sufficient to generate a solution within a limited amount of time even for networks of 100 nodes. During the computation, the radio can be turned off and the microcontroller draws only few milliamp of current, with negligible impact on the energy budget of the sensor node. Since the algorithm runs with limited overhead on the same hardware used to implement the sensing task, the system reliability can be improved by simply running the computation on a few back-up units.

6.8 Summary of Simulation Results

In Table 6.2 we report the 95% confidence interval for the simulation results of MDS, SOM, SOM3_A and SOM4_A. Figure 6.14 reports a plot wiht the average error obtained in each of the 9360 topologies (DOI=0) simulated.

Algorithm	Network Connectivity				
	4	**6**	**8**	**10**	**12**
			DOI=0.0		
MDS	0.608 ± 0.032	0.292 ± 0.017	0.256 ± 0.013	0.220 ± 0.012	0.190 ± 0.009
SOM	0.522 ± 0.034	0.233 ± 0.011	0.204 ± 0.005	0.196 ± 0.005	0.202 ± 0.006
SOM_3A	0.465 ± 0.034	0.234 ± 0.017	0.195 ± 0.010	0.214 ± 0.020	0.234 ± 0.016
SOM_4A	0.286 ± 0.012	0.171 ± 0.003	0.167 ± 0.002	0.177 ± 0.004	0.199 ± 0.002
			DOI=0.2		
MDS	0.678 ± 0.049	0.376 ± 0.025	0.376 ± 0.025	0.326 ± 0.022	0.217 ± 0.010
SOM	0.582 ± 0.044	0.274 ± 0.017	0.274 ± 0.017	0.231 ± 0.008	0.218 ± 0.004
SOM_3A	0.516 ± 0.048	0.268 ± 0.029	0.268 ± 0.029	0.204 ± 0.012	0.256 ± 0.023
SOM_4A	0.323 ± 0.015	0.195 ± 0.005	0.195 ± 0.005	0.181 ± 0.005	0.214 ± 0.003
			DOI=0.4		
MDS	0.783 ± 0.047	0.473 ± 0.022	0.400 ± 0.021	0.325 ± 0.016	0.279 ± 0.010
SOM	0.748 ± 0.063	0.398 ± 0.023	0.322 ± 0.015	0.286 ± 0.007	0.273 ± 0.008
SOM_3A	0.656 ± 0.055	0.393 ± 0.040	0.322 ± 0.033	0.306 ± 0.027	0.294 ± 0.024
SOM_3A	0.393 ± 0.017	0.272 ± 0.010	0.236 ± 0.006	0.247 ± 0.005	0.250 ± 0.004

Table 6.2: 95% confidence intervals of the simulation results.



(a) MDS (b) SOM

(c) SOM_3A (d) SOM_4A

Figure 6.14: Error summary for all the simulated topologies.

Chapter 7

Directional Antennas

The range-free, anchor-free localization algorithm presented in the last two chapters computes the node positions on the basis of radio connectivity, a piece of information that is readily available because obtained from the radio transceiver. The use of additional information, such as range or angle estimates, would allow for better accuracy, but the hardware necessary to obtain this data needs to be simple and compatible with the limited resources available at each sensor node. In this chapter, we introduce an inexpensive and simple directional antenna[1] that can be used to improve communication among nodes and also to obtain angle information useful in solving the localization problem. The following sections presents experimental results that characterize communication improvements in different settings, while Chapter 8 will address the problem of using the antenna to derive angle information.

7.1 Introduction

The directional antenna (DA) is an established technology that is effective in improving the performance of wireless networks. The ability to radiate the RF signal

[1] The antenna was developed at the Microelectronics Lab, University of Florence.

toward the receiver results in a more efficient utilization of the power, in a better link quality and in an increased transmission range. In addition, since communication is restricted in space, interferences between devices transmitting at the same time are reduced and spatial reusability can be exploited to increase network capacity and throughput [34, 96, 9]. While these characteristics have made DAs suitable for cellular towers and base stations, the use of directional antennas in ad-hoc wireless networks and WSNs is not equally widespread, mainly because of the need to design specific directional protocols. Many extensions have been proposed to the popular IEEE 802.11 MAC layer [9, 46, 21, 89] to support DAs and several other works have addressed the problem of directional routing [84, 90, 20, 40], but, in general, the lack of central coordination typical of ad-hoc networks makes difficult to fully take advantage of the directive technology, especially when nodes are mobile [8].

In sensor networks applications, although simplified communication protocols[2] can be adopted [79, 24], the main limitation is given by the complexity (cost and size) of this technology, which is not suitable for nodes that strive to be simple, small and inexpensive. Nevertheless, as radio communication moves to higher frequencies and antenna dimensions shrink, the use of DAs on sensor nodes appears not only feasible [53], but also desirable to compensate for the higher path loss intrinsic of shorter wavelengths, to ensure higher link quality and to implement a form of antenna diversity [95].

In light of the above considerations, we analyze a switched-beam, directional antenna that satisfies the size, cost and complexity constraints typical of a sensor node. The antenna, designed for operations in the 2.4GHz ISM band (e.g. using the IEEE 802.15.4 standard), has dimensions comparable to those of commercially available sensor nodes (it can actually be used to shelter the node itself), is built using inexpensive materials and implements a simple selection logic (the beam selection is implemented using two digital lines). After a brief description of the design principles

[2]The design of MAC protocols using directional antennas is in part simplified by the fact that sensor nodes are static and transmissions are sporadic. In addition, directive routing protocols can exploit the fact that sensor nodes typically transmit all their data to a single aggregation point (a cluster head or a base station).

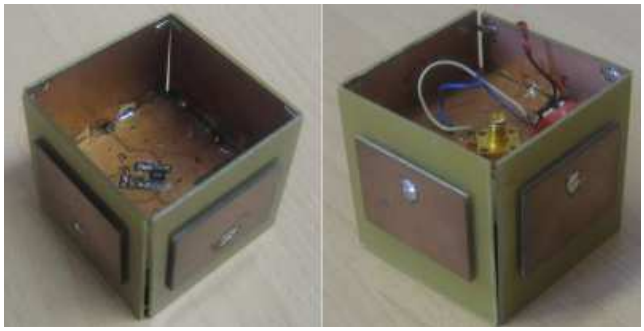


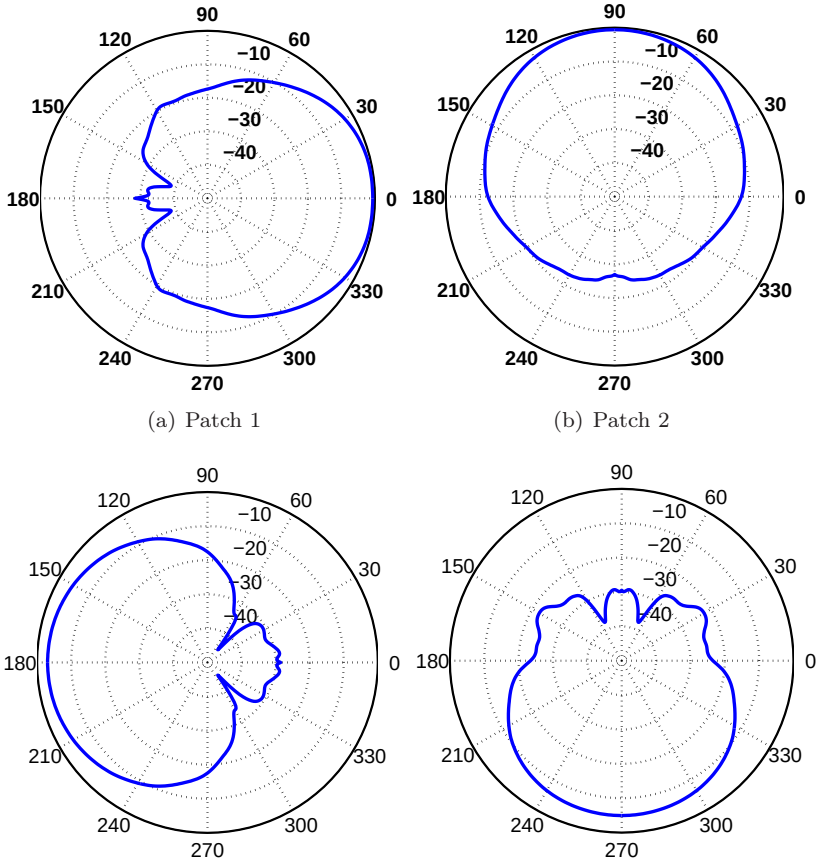
Figure 7.1: Two views of the *Four-Beam Directional Antenna* (FBDA).

provided in section 7.2, we use in-field experiments and theoretical fading models to characterize the mote-to-mote link in different environments.

7.2 Four-Beam Directional Antenna

The antenna used in our test, the *Four-Beam Directional Antenna* (FBDA), is composed of four coaxially fed planar patch antennas arranged in a "box like" structure as shown in fig. 7.1. Each face is realized on a two-layer RF4 substrate [7] having planar dimension of 56mm × 56mm and thickness of 2.4mm. The four patches, which operate in linear polarization, share a common design that has been optimized by using the Ansoft-HFSS CAD [3] to work in the 2.4 GHz ISM band. The mechanical arrangement of the four patches and their coaxial feeding is such that the vertical axis of the box coincides with the intersection of the E-planes of the single patches (i.e. the E-field is perpendicular to the ground).

The RF signal is distributed to the four faces by a single-pole four-trough switch, which is controlled by two digital lines and allows the wireless node to dynamically select the face to use (see Section 7.8). The loss due to the switch, the distribution network and the mismatches are compressively evaluated in about 1.5 dB within the selected ISM band. The characterization in the anechoic chamber has given the patterns reported in Figure 7.2. In spite of the low-cost substrate and reduced thickness, the patches gain measured to the external SMA connector, hence including



(a) Patch 1 (b) Patch 2

(c) Patch 3 (d) Patch 4

Figure 7.2: Radiation patterns of the four antenna faces. The patch used for transmission/recepetion is selected using two digital lines.

the losses listed before, are comprised between 8.3dBi and 7.5dBi. In Figure 7.2 we observe that the combined patterns ensure a uniform coverage of the 360 degree horizon.

7.3 Link Quality Experiments

In this section we begin the characterization of the FBDA with a series of in-field experiments based on commercially available WSN nodes. Although these experiments are not designed to be an exhaustive test for all possible operative conditions, the

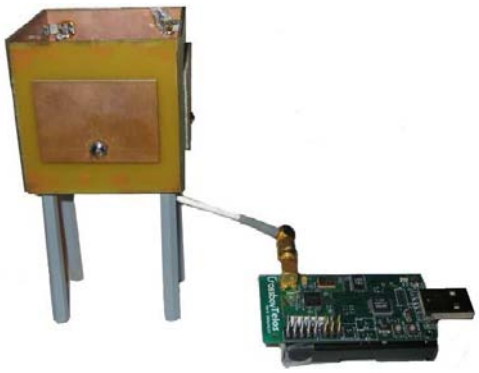


Figure 7.3: The directional antenna connected to the TelosB board.

measures attempt a characterization of communication in two different environments: **i)** a outdoor field where multi-path interferences are absent (except for the reflection from the ground) and **ii)** an indoor setting where the presence of reflecting surfaces and moving people is cause of severe multi-path effects. The results found in these two opposite scenarios can be used to derive more general design guidelines to implement WSNs using FBDAs.

The hardware chosen for our tests is the TelosB platform [64], a low-power sensor board equipped with the Chipcon CC2420 radio module [19]. The transceiver operates in the 2.4 GHz band using O-QPSK modulation with *Direct Sequence Spread Spectrum* (DSSS) coding, achieves a maximum data rate of 250Kbps and is compatible with the PHY and MAC layers defined by the new IEEE 802.15.4 standard. The omnidirectional (OD) antenna integrated on the TelosB board, an Inverted-F microstrip antenna (PIFA), has been used as baseline for our tests

7.3.1 Control Software

The tests were performed using a custom software designed to measure relevant radio link parameters and to evaluate the Packet Error Rate (PER) during communication between two sensor nodes. The application software consists of three software modules:

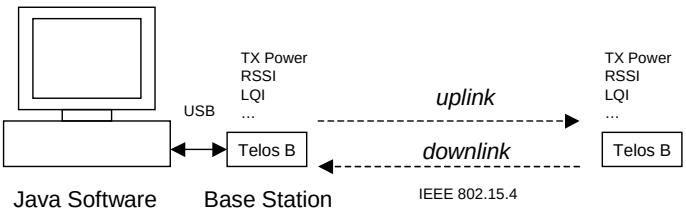


Figure 7.4: A block diagram of the software used to execute the measures.

1. A Java application running on a PC.
2. A TinyOS module executed by a node connected to the PC (the base station).
3. A TinyOS module that controls each of the nodes (target nodes) used for the experiment.

In the default operative mode, the Java software uses the base station (BS) to transmit beacon messages addressed to one or more target nodes in radio range (see Figure 7.4). Upon reception of a beacon, the target node records the *Radio Signal Strength Index* (RSSI) and the *Link Quality Indicator* (LQI) of the incoming packet[3]. The first value is expressed in dBm and defines the average power of the received radio signal, while the LQI is a new metric introduced by the IEEE 802.15.4 that measures the error in the incoming modulation and provides an estimation of the expected delivery ratio on a given radio link. The RSSI and LQI values recorded by the target node, together with the transmission power used by the BS, characterize the *uplink channel*, i.e. the link connecting the base station to the node. This information is transmitted back to the base station, which in turn records the parameters (transmission power, RSSI and LQI) that define the *downlink channel* (the link connecting the mobile node to the base station). The PC software collects the uplink and downlink information, displays them in a graphical form and provides an option to log the data into a file.

The software allows the user to control the transmission power for each node used in the test and to activate the four digital I/O lines exposed by the TelosB connector,

[3]These values, which are both coded using 8 bits, are read from the CC2420's internal registers upon reception of a successfully decoded packet

which are used to select the active FBDA patch. In addition, the software supports a feature for the evaluation of the packet error using three different modes:

- **Upload P.E.R.**: the test is executed by transmitting a train of radio messages from the base station to a specific target node. The transmission power, the number of messages and the delay interval are user selectable parameters. The target node records the number of messages received, the number of messages lost[4], mean and standard deviation of the uplink values (RSSI and LQI). Results are sent back to the base station at the end of test for data logging and processing.

- **Download P.E.R.**: the test is analogous to the upload test, but the packets are sent from the mobile node to the base station. In this case, in addition to the aggregate values (mean and variance), the RSSI and LQI values of each received packet are logged to enable more accurate data analysis (cf. Section 7.6).

- **Roundtrip P.E.R**.: the test is executed sending beacon messages from the base station to the target node, which in turn replies with an ACK message for each received packet (the ACK message is sent right after the reception of a message from the base station, before the arrival of the next one). The RSSI and LQI values (uplink and downlink) for each packet are logged for further analysis.

7.4 Outdoor Experiments

In the first set of experiments we measured the RSSI, LQI and the PER using the *roundtrip* mode described in the previous section. The experiments were performed in an open, grassy field with the BS and the target node placed at 1.3 m above the ground (see Figure 7.6). The values were measured at step of two meters using message sequences of 200 radio packets (100 messages in each direction) with a delay

[4]Each message include a sequence number that is used in determine the number of messages lost.

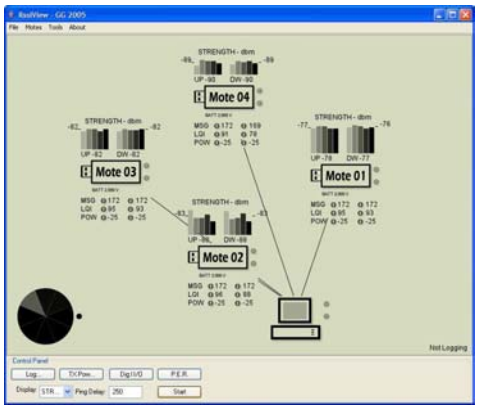


Figure 7.5: A screenshot of the control software used to execute the experiments.

interval of 100 ms between each message sent by the BS. Both nodes were set to use a transmission power of 0 dBm (1mW), which is the maximum power allowed by the transceiver.

Two different antenna configuration were tested:

1. **Directional antenna to directional antenna (DD)**: the two FBDA units were oriented with the active patches pointing toward each other (i.e. the direction of maximum angular gain).

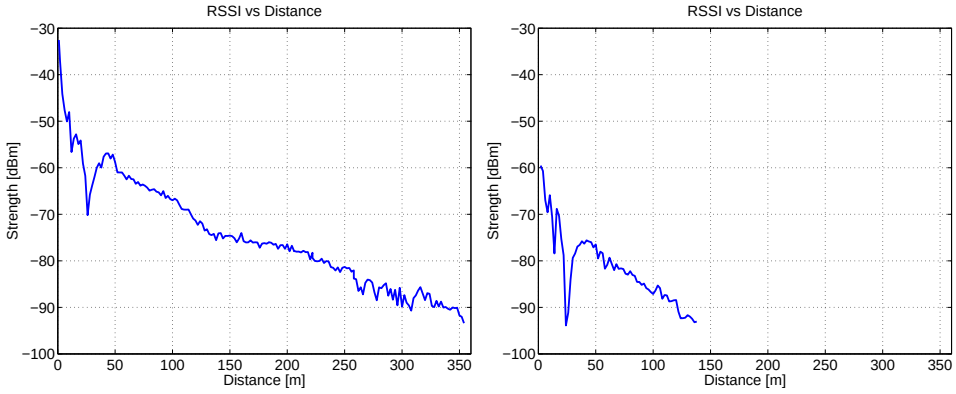
2. **Omnidirectional antenna to omnidirectional antenna (OO)**: the two TelosB board were oriented with the tail of the BS facing the head (the part

Figure 7.6: Setup used for outdoor experiments.

with the USB connector) of the target node. Since the radiation pattern for the PIFA integrated on the TelosB is not perfectly omnidirectional ([64]) and in a real deployment is hard (or impossible) to place the nodes to achieve the maximum gain, we chose this configuration as a representative of the average performance achievable with this antenna configuration.

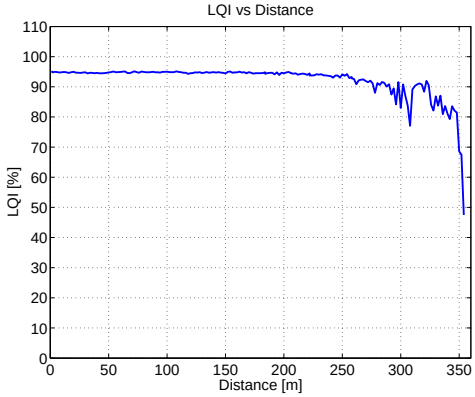
Figure 7.7 reports the average values for the RSSI, LQI and PER measures. The plots show that the higher gain of the directional antennas results in an increased link budget of +20dB respect to the OO configuration, with an extension of the communication range from 140 m to over 350 m. The graphs also shows that the P.E.R. is practically constant and equal to zero until the end of the communication range for the DD configuration. In the OO case, the PER follows a similar behavior, except for the deep fade null around 25m which causes almost 90% of the packet to be lost.

The values measured in our experiment are in line with those of other researchers that used high gain antennas to extend the communication range of sensor nodes. In [92], 8dBi omnidirectional antennas were used in an active volcano deployment to implement a WSN with radio hops of 200-400m, while in a recent work [76], TelosB boards were tested with other high-gain antenna configurations: a 17dBi 90° sector antenna and a 24dBi parabolic grid with a beam width of 8°. The maximum reported ranges are comprised between 500m and 800m. Although these values are higher than those measured in our experiments, we note that the antennas used in [76] are much bulkier and expensive than the FBDA. In any case, the conclusion that can be drawn from all these experiments are the same and suggest that the use of DAs is promising in WSN deployed over large areas where, without increasing the transmission range, multiple OD radio hops can be replaced by few directional radio links. We observe that reducing the total number of hops not only decreases the transmission delay and the probability of error, but also has a considerable impact on the energy conservation within the network since it decreases the total number of transmissions and receptions. In addition, as pointed out in [76], the use of DA can greatly simplify the network

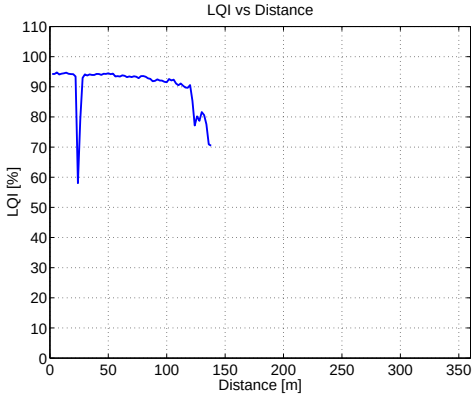


(a) RSSI - DD

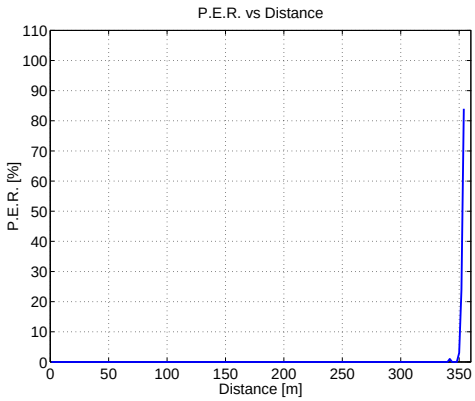
(b) RSSI - OO



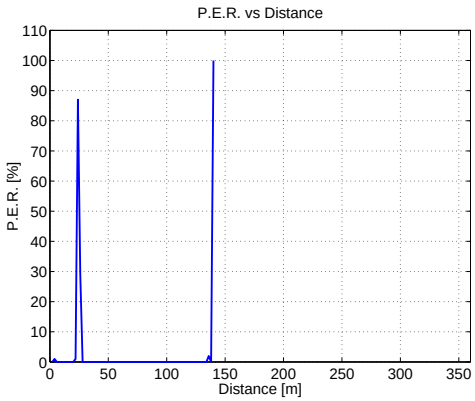
(c) LQI - DD



(d) LQI - OO



(e) PER - DD



(f) PER - OO

Figure 7.7: RSSI, LQI and PER measured using directional antennas (DD, plots on the left) and the internal TelosB antenna (OO, plots on the right).

architecture since many nodes might be able to communicate with the base station using a single hop, hence eliminating the need of multi-hop protocols. We note that managing multi-hop protocols can be problematic, especially when nodes operate in low duty-cycle mode[5] and synchronized *wake up's* need to be scheduled to allow neighboring nodes to exchange data packets.

7.5 Large-Scale Fading Models

In this section we analyze the data collected during the previous in-field experiments using the Two-Ray propagation model [77]. This model, which takes into account the contribute of the wave reflected from the ground, is often used to predict the average signal power at large distances from the transmitter. Before comparing the experimental data with the model, we review the basic laws that regulate RF propagation in free space and in presence of reflections from the ground.

7.5.1 Review of RF Propagation

Free Space Propagation

The Friis' equation predicts the power received as a function of the distance between the source and the destination according to the following formula:

$$\mathrm{P}_r(d) = \frac{P_t G_t G_r}{L} \left(\frac{\lambda}{4\pi}\right)^2 \left(\frac{1}{d}\right)^2, \tag{7.1}$$

where P_t is the transmission power, G_t and G_r are the transmitter and receiver antenna gains, L, with $(L \geq 1)$, is a factor that accounts for the losses in the system and d is the distance between the two units. The term λ is the wavelength of the radio signal; in our case, working at a frequency of about $f = 2.45$GHz, the wavelength is

[5] When a node operates in low duty-cycle mode, its radio is turned off most of the time to preserve energy.

equal to:

$$\lambda = \frac{c}{f} = \frac{3 \cdot 10^8}{2.45 \cdot 10^9} = 0.1224\, m. \tag{7.2}$$

The Friis' equation is useful to compute the *path-loss*, which defines the difference between the transmitted power and the power available at the receiver (as a function of distance). During a WSN deployment, being able to estimate the path-loss in a specific environment is important to obtain an estimate of the maximum communication range among nodes. The path-loss expressed in dB is defined as:

$$PL(d)_{dB} = 10 \log_{10} \left(\frac{\mathrm{P}_t}{\mathrm{P}_r} \right) = -10 \log_{10} \left(\frac{G_t G_r \lambda^2}{(4\pi)^2 d^2 L} \right). \tag{7.3}$$

Since it is often difficult to obtain correct estimates for the antenna gains G_t, G_r and the system losses L, while it is relatively easy to measure the received power at a given distance, it is easier to express the received power by taking into account the path-loss relative to a reference distance d_0, which allow us to write the received power as:

$$\mathrm{P}_r(d)_{dB} = \mathrm{P}_r(d_0) + 20 \log \left(\frac{d_0}{d} \right). \tag{7.4}$$

According to the expression above, the power available at the receiver decreases of 6 dB every time the distance between the transmitter and the receiver is doubled. Taking into account the receiver sensitivity (about -94 dBm for the CC2420) is possible to estimate the maximum communication range by using a single power measurement taken at a reference distance $d0$.

As a final note, we remind that the free space model is valid for communication that occur in the *far-field* of the antenna, at a distance d_f that verifies the following conditions:

$$d_f \quad \geq \quad \frac{2D^2}{\lambda}, \tag{7.5}$$

$$d_f \quad \gg \quad D, \tag{7.6}$$

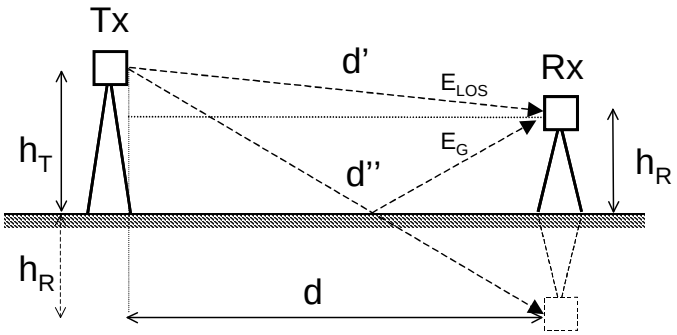


Figure 7.8: The contribute of the signal propagated by *line of sight* and the wave reflected from the ground.

$$d_f \gg \lambda, \tag{7.7}$$

where D is the largest physical dimension of the antenna ($D \cong 5cm$ for the FBDA).

Two-Ray Propagation Model

The Two-Ray propagation model takes into account the contribution of the RF signal reflected from the ground (see Figure 7.8). The total E-field at the receiver is given by the sum of E_{LOS} (the *line of sight* component) and E_g (the component reflected from the ground):

$$E_{TOT}(d,t) = E_{LOS}(d',t) + E_g(d'',t), \tag{7.8}$$

$$E_{LOS}(d',t) = \frac{E_0 d_0}{d'} \cos\left(\omega_c \left(t - \frac{d'}{c}\right)\right), \tag{7.9}$$

$$E_g(d'',t) = \frac{E_0 d_0}{d''} \cos\left(\omega_c \left(t - \frac{d''}{c}\right)\right). \tag{7.10}$$

The components E_{LOS} and E_g reach the destination at distance d after covering two different distances d' and d''. The relationship between d, d', d'' and the heights of the transmitter and receiver (h_t and h_r) can be computed using the *method of images* (see Figure 7.8):

$$d' = \sqrt{(h_T - h_R)^2 + d^2}, \tag{7.11}$$

$$d'' = \sqrt{(h_T + h_R)^2 + d^2}. \tag{7.12}$$

The length difference between the paths traveled by the two rays causes a *phase difference* θ_Δ between the waves impinging on the receiver. The phase difference is given by:

$$\theta_\Delta = \frac{2\pi\Delta}{\lambda} = \frac{\Delta\omega_c}{c}, \tag{7.13}$$

where $\Delta = d'' - d'$ is the length difference between the two paths.

To determine the resulting signal, in addition to considering the phase difference θ_Δ, we also need to consider the polarization of the antenna and the effect of the reflection. The FBDA antenna has a vertical polarization with the E-plane perpendicular to the ground (see Section 7.2), hence when the signal bounces off the ground, the reflected wave has a 180° phase delay. We also note that, as the distance between the transmitter and the receiver increases, the angle of incidence with the ground becomes small and almost all the energy is reflected. Since for high distances the path difference Δ becomes negligible, the magnitude of the two E-fields is approximatively the same:

$$\left|\frac{E_0 d_0}{d'}\right| \approx \left|\frac{E_0 d_0}{d''}\right|. \tag{7.14}$$

According to these considerations, the resulting E-field can be expressed as:

$$\begin{aligned} E_{TOT}(d,t)|_{t=\frac{d''}{c}} &= \frac{E_0 d_0}{d'} \cos\left(\omega_c \left(\frac{d'' - d'}{c}\right)\right) - \frac{E_0 d_0}{d''} \cos 0^\circ \\ &= \frac{E_0 d_0}{d'} \cos(\theta_\Delta) - \frac{E_0 d_0}{d''}, \end{aligned} \tag{7.15}$$

where the value has been evaluated at a time instant $t = d''/c$. The above expression indicates that at any given time instant, the phase differences between the two

components is θ_Δ:

$$E_{TOT}(d) = \frac{E_0 d_0}{d}(\cos\theta_\Delta + j\sin\theta_\Delta) - \frac{E_0 d_0}{d}, \tag{7.16}$$

therefore the module of E_{TOT} is given by:

$$\begin{aligned} |E_{TOT}(d)|^2 &= \frac{E_0 d_0}{d}\sqrt{(\cos\theta_\Delta - 1)^2 + \sin^2\theta_\Delta} \\ &= \frac{E_0 d_0}{d}\sqrt{2 - 2\cos\theta_\Delta} \\ &= 2\frac{E_0 d_0}{d}\sin\left(\frac{\theta_\Delta}{2}\right). \end{aligned} \tag{7.17}$$

The expression above fully characterizes the signal propagation in presence of a reflected component from the ground.

Relating the Electrical Field to the Received Power

Once the E-field is known, it can be related to the received power $P_r(d)$ by writing the Friis' free space equation in a different form:

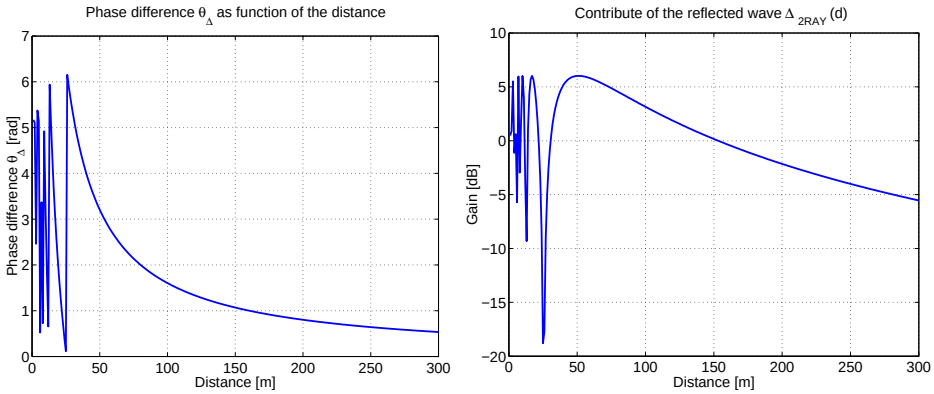
$$\mathrm{P}_r(d) = P_d A_e = \left(\frac{P_t G_t}{4\pi d^2}\right)\left(\frac{G_r \lambda^2}{4\pi}\right). \tag{7.18}$$

In the expression above, P_d is the *power flux density* $[W/m^2]$, and A_e is the *effective antenna aperture* $[m^2]$. The power flux density is related to the E-field as follows:

$$P_d = \left(\frac{P_t G_t}{4\pi d^2}\right) = \frac{E^2}{R_{fs}} = \frac{|E|^2}{377\Omega}, \tag{7.19}$$

where $R_{fs} = 377\Omega$ is the intrinsic impedance of the free space. Using the previous expression, we can relate the E-field resulting from the two components (E_{TOT} and E_g) to the power at the receiver:

$$P_{2RAY}(d)_{dB} = \frac{\left|2\frac{E_0 d_0}{d}\sin\left(\frac{\theta_\Delta}{2}\right)\right|^2}{377\Omega} \tag{7.20}$$



(a) Phase difference: θ_Δ (b) Contrib. from the ground: $\Delta_{2RAY}(d)$

Figure 7.9: Phase difference and gain in the Two-Ray Model

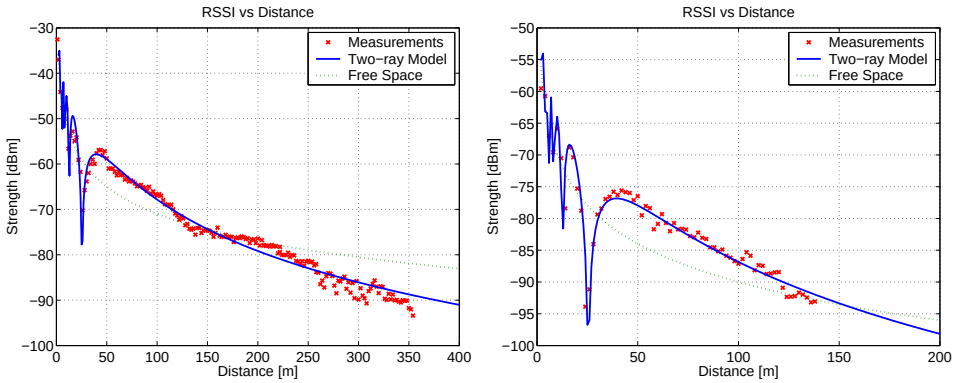
$$
\begin{aligned}
&= \frac{1}{377\Omega}\left(\frac{E_0 d_0}{d}\right)^2 + 20\log\left[2\sin\left(\frac{\theta_\Delta}{2}\right)\right] \\
&= P_{fs}(d) + \Delta_{2RAY}(d),
\end{aligned}
$$

where $P_fs(d)$ is the power received due to free space transmission and $\Delta_{2RAY}(d)$ is an additional term that accounts for the contribute of the reflection from the ground. $\Delta_{2RAY}(d)$ varies in function of the distance through the difference of phase as described in (7.13).

The plots in Figure 7.9 show that the contribute of the reflected signal rapidly fluctuates in the first 50 meters with oscillations comprised between +6dB and -18dB (theoretically it goes to $-\infty$dB). After 150 m the contribute becomes negative.

7.5.2 Comparison With The Experimental Data

In Figure 7.10 we report a comparison between the value measured in the outdoor experiments and the received power predicted by the free-space equation and the Two-Ray model. The models were compared to the measured data by adding a gain value determined experimentally, in fact, although the transmission power (+0dBm) is known, it is difficult to accurately estimate the amount of the system losses (7.1).



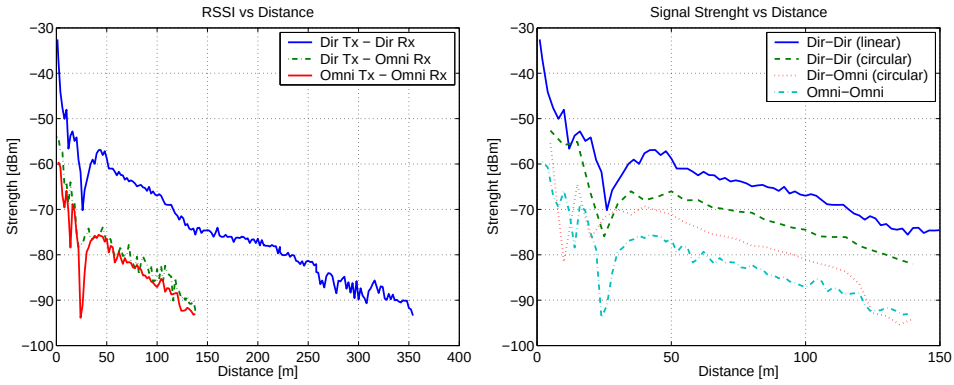
(a) Directional Antennas (FBDA) (b) Omnidirectional Antennas (TelosB)

Figure 7.10: Comparison between measured data and the Two-Ray model.

The results show a very good correspondence between the values computed using the Two-Ray model and the measured values. The accuracy of the Two-Ray model in predicting the received power suggests the possibility to use it in future simulations to evaluate the path-loss in WSN applications using both omnidirectional and FBDA antennas.

7.5.3 Effect of Antenna Polarization

As mentioned earlier, the FBDA antenna operates in vertical polarization (with the E-field perpendicular to the ground), while the PIFA antenna integrated on the TelosB board has a polarization that, given the characteristic of this kind of antenna, results difficult to predict. In order to evaluate the system performance when different antennas are used within the same WSN, we repeated the experiments described in Section 7.3. In Figure 7.11a we report the RSSI values measured in the previous experiment together with the values measured when a FBDA is used to exchange messages with an OD node. The plot shows that when the communication involves nodes with different antennas (FBDA and PIFA), the received power and the transmission range are about the same than the values measured when using two OD antennas (i.e. any increment in the link budget due to the use of a directive antenna



(a) Linear polarization.

(b) Linear and circular polarization.

Figure 7.11: Signal Strength as function of distance for different antenna configurations

is lost). The explanation for the poor performance of this configuration is given by the difference of polarization between the two antennas, which cause considerable losses at the receiver. To get a further insight on the effect of antenna polarization, we have repeated the range measurements using a DA with circular polarization[6]. In this case, the link budget when using an hybrid antenna configuration improves as shown in Figure 7.11b.

7.6 Indoor Experiments

The signal strength measurements were repeated indoors to characterize the effect of multipath interferences due to the presence of reflecting surfaces and moving objects in proximity of sensor nodes. In this case, the signal received is subjected to rapid fluctuations due to random components that sum at the receiver. In the test performed indoors, we placed the nodes at approximately 5 meters of distance in a lab environment with several pieces of equipment near the experiment area. The line of sight between the nodes was unobstructed, but the signal was perturbed by the presence of people moving in proximity of the nodes. The RSSI data was collected

[6]The circular polarization can be seen as a combination of two orthogonal antennas with linear polarizations.

Antenna Configuration	Mean	Variance
Omni-Omni	-86.58 dBm	6.33
Omni-FBDA	-78.75 dBm	2.24
FBDA-FBDA	-63.93 dBm	1.91

Table 7.1: Mean and Variance of RSSI measured indoors with different antenna configurations.

using the *download* PER mode described in Section 7.3.1. The target node was set to transmit a message every 50 ms with a power of -25dBm (the minimum transmission power programmable on the CC2420 transceiver). The number of RSSI measures collected was about 1200 for each antenna configuration. Figure 7.12 reports the RSSI distribution for three different cases:

1. **Omni-Omni**: two TelosB nodes were used to exchange the radio packets.
2. **FBDA-Omni**: the messages were transmitted by a node with FBDA antenna and received by a TelosB node with OD antenna.
3. **FBDA-FBDA**: both nodes were equipped with a FBDA antenna.

The results indicate that when communication is affected by multipath fading, the use of directive antennas greatly increases the signal strength and reduces the variability of the received power. Table 7.1 reports the mean and variance of the RSSI data, showing that the link budget increases of +7.8dBm when using just one directional antenna and of +22dBm when using FBDAs on both sides of the radio link. Also, the variance is reduced by 65% when using one FBDA and by 70% when using two FBDAs.

7.6.1 Small-scale Fading Model

Similarly to what done in Section 7.5, we analyze the measured data using a theoretical fading model. The model used is the Ricean distribution [77], a statistical model commonly used to describe the power distribution at the receiver in presence of multipath fading. The Ricean *probability density function* (pdf) defines the probability

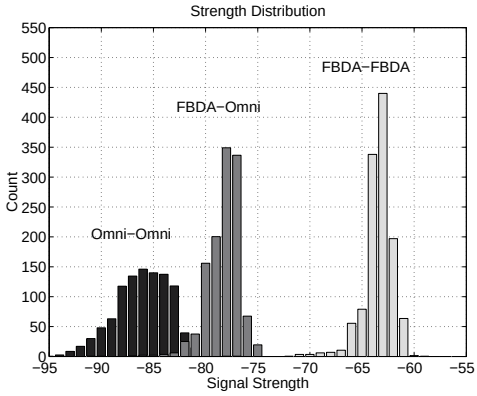


Figure 7.12: Distribution of the received signal strength in presence of multi-path fading for three configurations: Omni-Omni, Omni-FBDA, FBDA-FBDA.

of receiving a signal with amplitude of v volts, $(v \geq 0)$ as:

$$p(v) = \frac{v}{\sigma^2} e^{-\frac{(v^2+A^2)}{2\sigma^2}} I_0\left(\frac{Av}{\sigma^2}\right), \tag{7.21}$$

where I_0 is the modified Bessel function of the first kind and zero order, A is the peak amplitude of the LOS component and σ^2 is the time-average power of the received radio signal. The distribution is described by the parameter $K = A^2/2\sigma^2$, which defines the ratio between the LOS component and the contribution of secondary rays due to multi-path propagation.

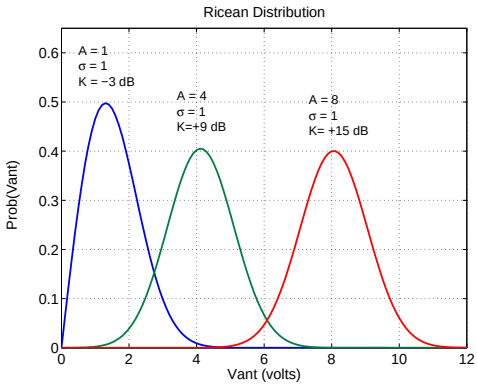


Figure 7.13: Samples of Ricean distribution for various parameters K.

Before being able to analyze the measured data using the Ricean distribution, we

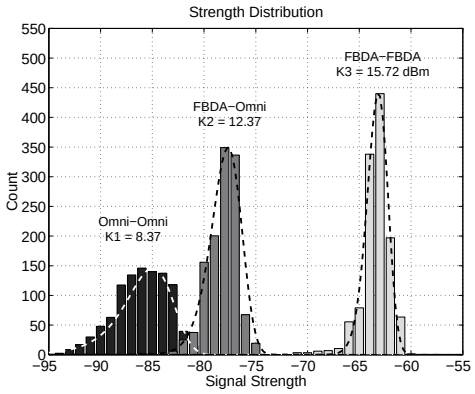


Figure 7.14: Distribution of the received signal strength with Ricean pdf's.

need to relate the amplitude of the signal (in volts) to the received power (in dBm). The power at the receiver is given by:

$$\mathrm{P}_r = \frac{V_{ant}^2}{R_{ant}}, \tag{7.22}$$

where V_{ant} is the voltage at the terminals of the antenna and R_{ant} is its resistance (50Ω). After converting the RSSI values into volts, we used the MATLAB `fminsearch` function to find the parameters A, σ that minimize the difference between the measured data and those computed using the Ricean distribution. As initial estimates for A and σ, we used the mean and the variance of the data: $A_0 = \mathrm{mean}(\mathrm{RSSI}_{\mathrm{volt}})$, $\sigma_0 = \mathrm{std}(\mathrm{RSSI}_{\mathrm{volt}})$. This choice was motivated by the fact that, for large values of the parameter K ($K \gg 1$), the Ricean distribution is approximatively Gaussian about the mean (cf. Figure 7.13). Figure 7.14 shows the distributions of the measured RSSI data together with the Ricean distribution: the use of FBDA antennas increases the K factor from 8.3 dB to 15.7 dB, demonstrating a substantial increment of the LOS component and reduction of the multi-path interferences. Again we note a good correspondence between the measured data and those calculated using the fading model, therefore the parameters found will be useful in future system simulations.

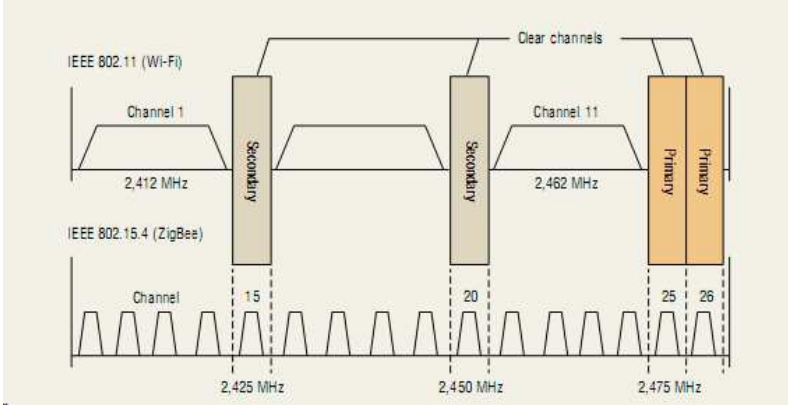


Figure 7.15: IEEE 802.11 and IEEE 802.15.4 channels (Copyright 2005 by Penton Media Inc).

7.7 Rejection to 802.11 Interferences

In a typical WSN application, the information gathered by the sensors is collected by a sink node and then forwarded to a gateway (e.g. a PC with an Internet connection) that dispatches the data to the remote users. In this section we consider the case where the gateway uses an IEEE 802.11g link that might interfere with the packets received by the sink node. Although four of the sixteen channels defined by the IEEE 802.15.4 standard fall in the guard band of 802.11 channels and thus are immune from interferences (see Figure 7.15), as the number of 2.4 GHz devices increases, it might be necessary to use one of the "non-safe" 802.15.4 channels.

To evaluate the effects of possible interferences between IEEE 802.11 devices and IEEE 802.15.4 nodes, we propose an experiment where the Chipcon CC2420 is set to operate on Ch.17 and the sink node is positioned close to an 802.11g device working on an overlapping band (Ch. 6). In our setup, the 802.11g source was located at 1.5 meters from the sink node, forming a 90 degree angle with the axis of the face used for reception (see Figure 7.16). The packets were transmitted by a FBDA node located at 5m from the sync node and set to -25dBm of power.

Figure 7.17 reports the effect of the interference on the delivery ratio and the Link Quality Index (LQI). When the sync node uses an OD antenna, the delivery ratio and

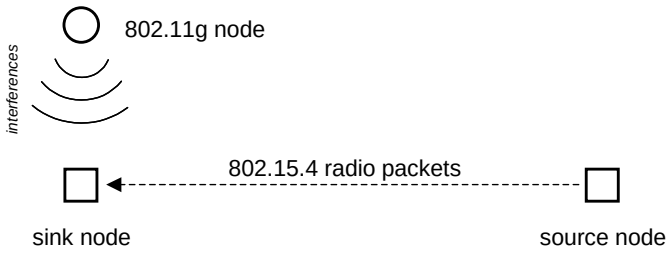
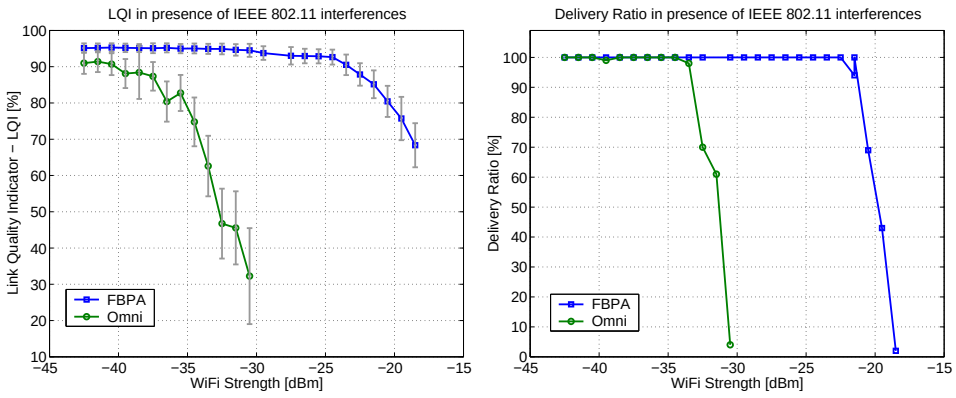


Figure 7.16: Interferences from an 802.11 device.

the LQI drops rapidly as the 802.11g interference source transmits above a value of -33 dBm. In the case of a sink node with directional antenna, the FBDA attenuates the interferences of about 14 dB and the quality of the link is not substantially affected until the 802.11g interference reaches the -19 dBm.

The results can be explained by the fact that the error probability P_E is proportional to the interference strength (P_I) at the receiver input. In the OQPSK modulation, the error probability is a function of the *Signal to Noise and Interference Ratio* (SINR) and the process gain of the coding (9dB in the case of DSSS coding):

$$P_E = erfc(SINR \cdot PG), \tag{7.23}$$



(a) LQI (b) Delivery Ratio

Figure 7.17: LQI and Delivery Ratio as function of the 802.11 interference power.

where the SINR is given by:

$$SINR = \frac{P_{RX}}{P_{N0} + P_I}. \tag{7.24}$$

In the above expression, P_{RX} is the power of the 802.15.4 signal, P_{N0} is the noise power and P_I is the power of the 802.11g interference. The interference power P_I can in turn can be written as:

$$P_I = K \cdot P_{TX}^i \cdot L_p(d) \cdot G(\theta), \tag{7.25}$$

where P_{TX}^i is the power of the 802.11g source, $L_p(d)$ is the path loss as function of the distance d and $G(\theta)$ is the gain of the antenna in the direction of the source. Finally, K is a factor that accounts for the portion of IEEE 802.11g power that falls in the 802.15.4 channel (approximately 19%) [83]. Having fixed the distance d and the angle θ, the link quality is solely affected by the interference power P_{TX}^i. In our experiments, the angle θ was equals to 90 degrees with respect to the active face (patch 3) and the increased immunity to the interference can be justified by the reduced gain of the antenna in that direction. Although the experiment was conducted in the lab using small power levels and short distances, the values can be easily scaled to determine how the sink node should be positioned with respect to the 802.11g source to avoid interferences in a real-case scenario.

7.8 Angular Diversity

In a WSN application each sensor exchanges information with other nodes located at different positions, therefore a switching policy is needed to determine the patch to be used for each communication. If the network is static and node positions are known, these data can be preloaded into each unit and the selection can be implemented by choosing the face pointing in direction of the other node. Anyway, positions are not always known in advance (ad hoc deployment) or the face pointing toward the other

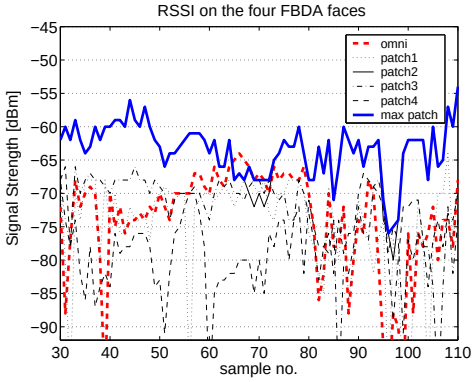


Figure 7.18: Samples of signal strength measured on the four antenna patches.

node might not be the best selection due to the presence of multipath, reflections and scattering of the signal. In these cases the best patch can be dynamically determined during the setup phase of the network by exchanging a few radio beacons with the surrounding nodes and selecting the face that ensure better signal strength. While this problem has been addressed in relation to MAC and routing protocols for ad-hoc WSNs [79, 24], we also note that by allowing a node to dynamically switch among different beams, the directive antenna implements a form of angular diversity that increases the reliability of communications [95]. This is especially important when nodes are static and the presence of a deep fade null can affect the communication for an indefinite period of time. Using the four faces of the FBDA, the node is given the opportunity to choose among four low correlated radio channels, thus increasing the probability of successful communication. In the last experiments a node with the directive antenna is used to communicate with an OD node that is moved through different positions. Figure 7.18 compares the signal strength received at the four patches with the signal received by an OD antenna placed at the same position. While all of the lines fluctuate and have deep fades (see Figure 7.18), the ability to select the face with maximum RSSI ensure stable receptions. In our experiment, the correlation coefficient for the strength values received by different faces was comprised from a max of 0.43 for face 1 and 2, to a min of 0.15 for face 1 and 3.

Chapter 8

Deriving Angle Information From Directional Antennas

As seen in previous sections, directional antennas are effective in improving the link budget and in reducing the signal variability due to multi-path fading. Another advantage of DAs is that they can be used to estimate Angle of Arrival (AoA) information useful in solving the localization problem [67, 94, 60]. In the following sections we use the antenna presented in Chapter 7 to implement an algorithm that estimates the angular position of a node in proximity of a base station with FBDA.

8.1 Angle and Range Information in Free Space

We consider a target node equipped with an omni-directional antenna that is transmitting a radio packet to a base station equipped with a FBDA (see Figure 8.1). By measuring the power received on two faces of the antenna, following a scheme similar to [94, 60], we are able to derive the equations that define the position of the target node in the 2D space. According to the Friis' free space equation:

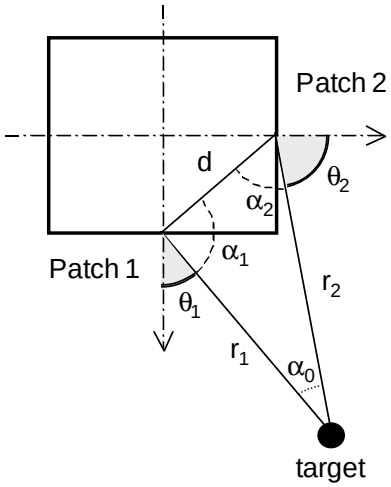


Figure 8.1: Distances and angles of a target node relative to patch 1 and 2.

$$\mathrm{P}_{r1} = \frac{P_t G_t G_{r1}(\theta_1)}{r_1} \left(\frac{\lambda}{4\pi}\right)^2, \tag{8.1}$$

$$\mathrm{P}_{r2} = \frac{P_t G_t G_{r2}(\theta_2)}{r_2} \left(\frac{\lambda}{4\pi}\right)^2, \tag{8.2}$$

where P_{r1} and P_{r2} is the power of the signal received on patch 1 and 2 when the target node transmits a signal with power P_t using an OD antenna with gain G_t. The values $G_{r1}(\cdot)$ and $G_{r2}(\cdot)$ are the angular gains of the patches 1 and 2 and depend on the angles of arrival θ_1, θ_2. The position of the target node relative to the base station is fully described by the unknown values r_1 and θ_1. The distance and the angle relative to the second patch (r_2 and θ_2) are derived using the cosine law:

$$r_2 = \sqrt{r_1^2 + d^2 - 2 r_1 d \cos(\alpha_1)}, \tag{8.3}$$

$$\alpha 2 = \arcsin\left(\frac{r_1}{r_2} \sin \alpha_1, \right) \tag{8.4}$$

where d is the distance between the center of the two patch antennas and α_1 and α_2 are the angles as defined in Figure 8.1. The relationship between the angles α_1, α_2

are derived by considering the following equations:

$$\pi = \alpha_1 + \alpha_2 + \alpha_3, \tag{8.5}$$

$$\alpha_1 = \frac{3}{4}\pi - \theta_1, \tag{8.6}$$

$$\alpha_2 = \frac{3}{4}\pi + \theta_2. \tag{8.7}$$

Solving the equations above we find that:

$$\theta_2 = \theta_1 - \frac{\pi}{2} - \alpha_0. \tag{8.8}$$

The notation can be simplified by taking into account the fact that in many situations the target's distance from the FPDA will be much larger than the value d ($r1, r2 >> d$) and therefore $r1 \cong r2$ and $\alpha_0 \cong 0$. Under these conditions, the equations that relates the angles and distances reduce to:

$$r_2 = r_1, \tag{8.9}$$

$$\theta_2 = \theta_1 - \frac{\pi}{2}. \tag{8.10}$$

Using these equations, the power received on patches 1 and 2 can be rewritten as function of r_1 and θ_1 only (we have omitted the constant term $\left(\frac{\lambda}{4\pi}\right)^2$):

$$\mathrm{P}_{r1} = \frac{P_t G_t G_{r1}(\theta_1)}{r_1}, \tag{8.11}$$

$$\mathrm{P}_{r2} = \frac{P_t G_t G_{r2}(\theta_1 - \pi/2)}{r_1}. \tag{8.12}$$

It follows that the ratio between the power received on the two patches is equal to the ratio between the antenna gains for angles θ_1 and θ_2:

$$\frac{P_{r1}}{P_{r2}} = \frac{G_{r1}(\theta_1)}{G_{r2}(\theta_1 - \pi/2)}. \tag{8.13}$$

Since the angular gains $G_{r1}(\cdot)$ and $G_{r2}(\cdot)$ are not given in analytical form[1], a solution to equation (8.13) can only be obtained using numerical methods to find the value of θ_1 that satisfies the equality, or at least minimize the difference between the ratios P_{r1}/P_{r2} and G_{r1}/G_{r2}:

$$\theta_1 = \arg\min_{\theta} \left| \frac{P_{r1}}{P_{r2}} - \frac{G_{r1}(\theta_1)}{G_{r2}(\theta_1 - \pi/2)} \right|. \tag{8.14}$$

Once the angle θ_1 has been determined, the target's distance r_1 is found by inverting (8.2):

$$r_1 = \frac{P_t G_t G_{r1}(\theta_1)}{P_{r1}} \left(\frac{\lambda}{4\pi} \right)^2. \tag{8.15}$$

The relationships between the power received on the first patch and patch n.3 and n.4 are derived using a similar approach (see Figure 8.2).

In conclusion, assuming that the target node is at a distance r such that $r \gg d$, the power levels received on the different antenna patches are related by the following system of equations:

$$\frac{P_{r1}}{P_{r2}} = \frac{G_{r1}(\theta_1)}{G_{r2}(\theta_1 - \pi/2)}, \tag{8.16}$$

$$\frac{P_{r1}}{P_{r3}} = \frac{G_{r1}(\theta_1)}{G_{r3}(\theta_1 + \pi)}, \tag{8.17}$$

$$\frac{P_{r1}}{P_{r4}} = \frac{G_{r1}(\theta_1)}{G_{r4}(\theta_1 + \pi/2)}. \tag{8.18}$$

In the next sections, we will use the relationships derived here to estimate the target's angular position θ_1. We note that although, in principle, we could also estimate the distance r_1 using (8.15), we will not to attempt to compute this value because it can be determined only if the exact values of P_t and G_t are known. Also, since equation 8.15 assumes a free space propagation model, in real applications we would need to estimate the path-loss exponent for the environment where the FBDA operates. The angular position θ_1, instead, can be recovered by only taking into account the power

[1] We use the radiation patterns described in Section 7.2, which are obtained through measurements in an anechoic chamber.

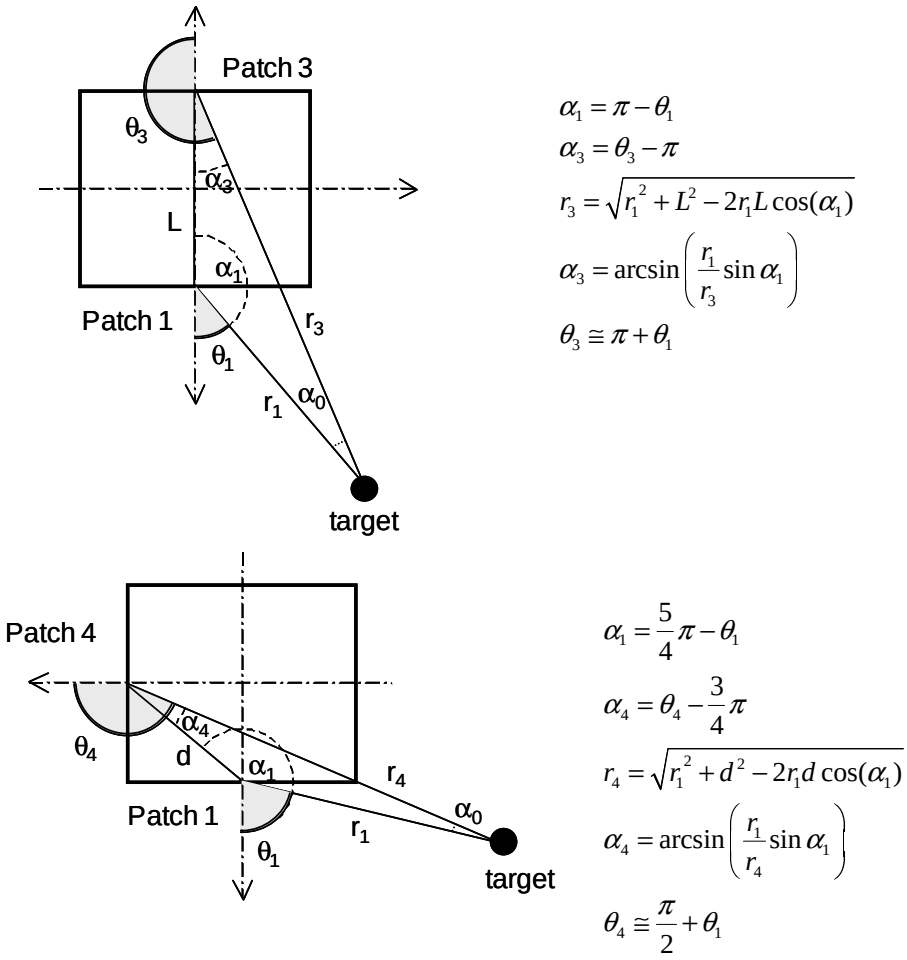


Figure 8.2: Distances and angles of a target node relative to patches 1-3 and patches 1-4.

received on the four faces and the angular gains (i.e. the radiation patterns of the four patches).

8.2 Implementation Issues

In the previous section we have derived the equations that describe how the power received on the four faces varies as a function of the target position (r_1, θ_1). The underlying assumption is that a packet transmitted by the target device can be received simultaneously by all the antenna elements and that the four power levels $P_{r1}, \ldots, P_{r4}$

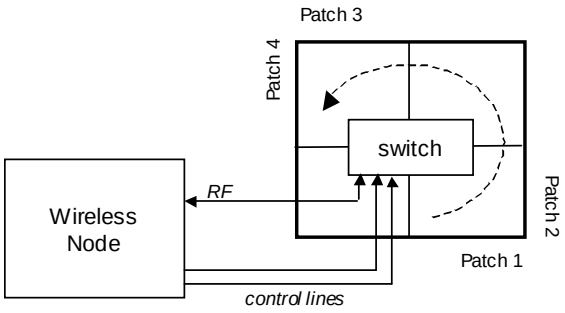


Figure 8.3: The FBDA is controlled by switching circuit that distributes the RF signal among the four faces. Only one of the four patches is active at a time.

can be measured at the same time. This is not possible with the FBDA because the patches are controlled by a switch and only one antenna can be active at any time instant (see Figure 8.3). Supporting simultaneous readings would require four RF front-ends on the wireless node or, at least, the presence of additional circuitry to measure the RF power on each patch. These solutions are not suitable for WSN applications due to the cost limitations imposed on sensor nodes and, therefore, they were not considered during the design phase of the antenna. Despite the limitations of the hardware, the problem can be overcome by modifying the communication protocol between the base station and the target device. The modification consists in activating the four faces in a rapid sequence and in transmitting a beacon message from each of them. We suppose that the target node is able to acknowledge the messages received in a time shorter than the switching time; if no reply is received, we assume that communication on that face is not possible due to insufficient radio power at that given angle. By adopting this scheme, if the switching time is short (in the orders of tens of milliseconds) and the target position does not substantially change during the four rounds of packet exchanges, the base station using the FBDA can determine the power values necessary for position estimation. Figure 8.4 shows two of the four message rounds required to measure the values $P_{r1}, \ldots, P_{r4}$.

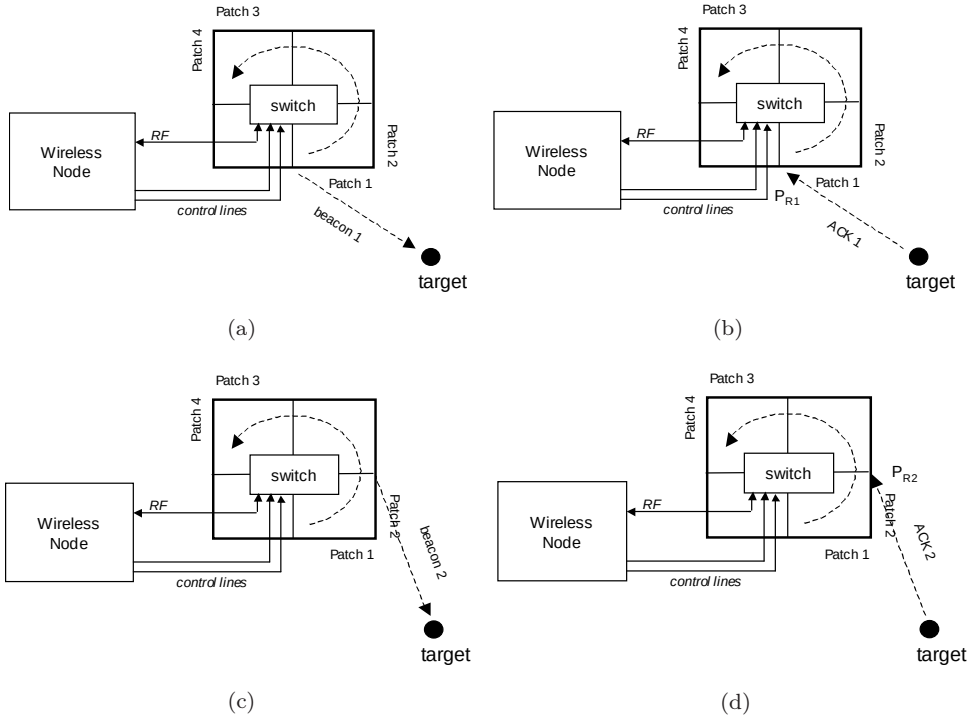


Figure 8.4: Two of the four phase sequence used to collect RSSI measurements on the faces of the FBDA.

8.3 Deriving Angle Information

Equations 8.16, 8.17, 8.18 define the ratio between P_{r1} and P_{r2}, P_{r3}, P_{r4} for any value of the angle θ_1. Since the power at the transceiver input is measured in dBm and the antenna gains are expressed in dB, it is easier to rewrite the equations as:

$$[\mathrm{P}_{r1} - \mathrm{P}_{r2}]_{dB} = [G_{r1}(\theta_1) - G_{r2}(\theta_1 - \pi/2)]_{dB}\,, \tag{8.19}$$

$$[\mathrm{P}_{r1} - \mathrm{P}_{r3}]_{dB} = [G_{r1}(\theta_1) - G_{r3}(\theta_1 + \pi)]_{dB}\,, \tag{8.20}$$

$$[\mathrm{P}_{r1} - \mathrm{P}_{r4}]_{dB} = [G_{r1}(\theta_1) - G_{r4}(\theta_1 + \pi/2)]_{dB}\,. \tag{8.21}$$

Figure 8.5 is used to provide a visual interpretation of the equations introduced above. Fixed the angle θ_1, the RF signal from the target node is received with different power levels because the signal impinges on the four patches with different relative angles and, thus, is subjected to different antenna gains. Figure 8.6a reports the patch gains

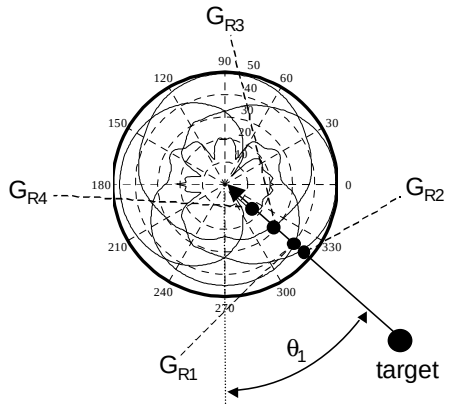


Figure 8.5: Antenna gains seen from the target node.

as a function of different angle position θ_1 (from now on, we will drop the subscript 1 and we will just use the notation θ to denote angles relative to patch 1). Fixed any other condition (e.g. P_t and G_t), the differences between P_{r1} and P_{r2}, P_{r3}, P_{r4} only depend on the difference between the gains $G_{r1}(\theta)$ and $G_{r2}(\theta), G_{r3}(\theta), G_{r4}(\theta)$. In condition of ideal RF propagation the following condition holds:

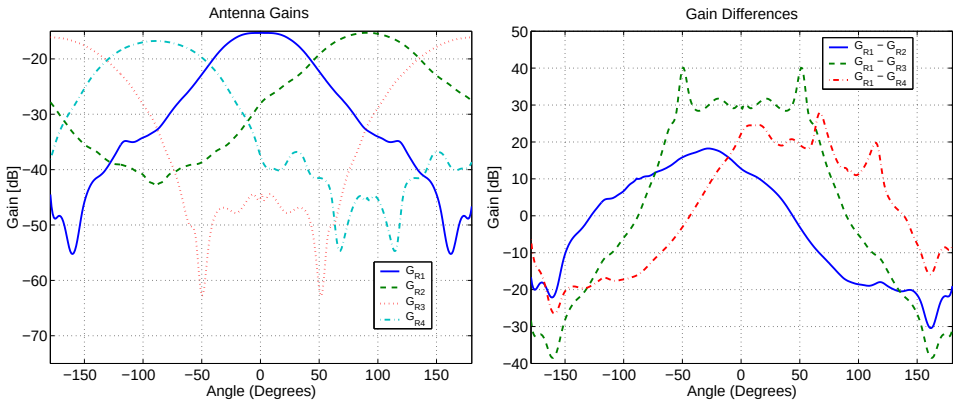
$$\|DP - DG(\theta)\| = 0, \tag{8.22}$$

where $\|\cdot\|$ denotes the Euclidean norm, DP is a column vector containing the power differences between the patch n.1 and the other faces:

$$DP = \begin{bmatrix} \mathrm{P}_{r1} - \mathrm{P}_{r2} \\ \mathrm{P}_{r1} - \mathrm{P}_{r3} \\ \mathrm{P}_{r1} - \mathrm{P}_{r4} \end{bmatrix}, \tag{8.23}$$

and DG contains the gain differences for any angle θ:

$$DG(\theta) = \begin{bmatrix} G_{r1}(\theta) - G_{r2}(\theta - \pi/2) \\ G_{r1}(\theta) - G_{r3}(\theta + \pi) \\ G_{r1}(\theta) - G_{r4}(\theta + \pi/2) \end{bmatrix}. \tag{8.24}$$



(a) Antenna gains (b) $DG(\theta)$

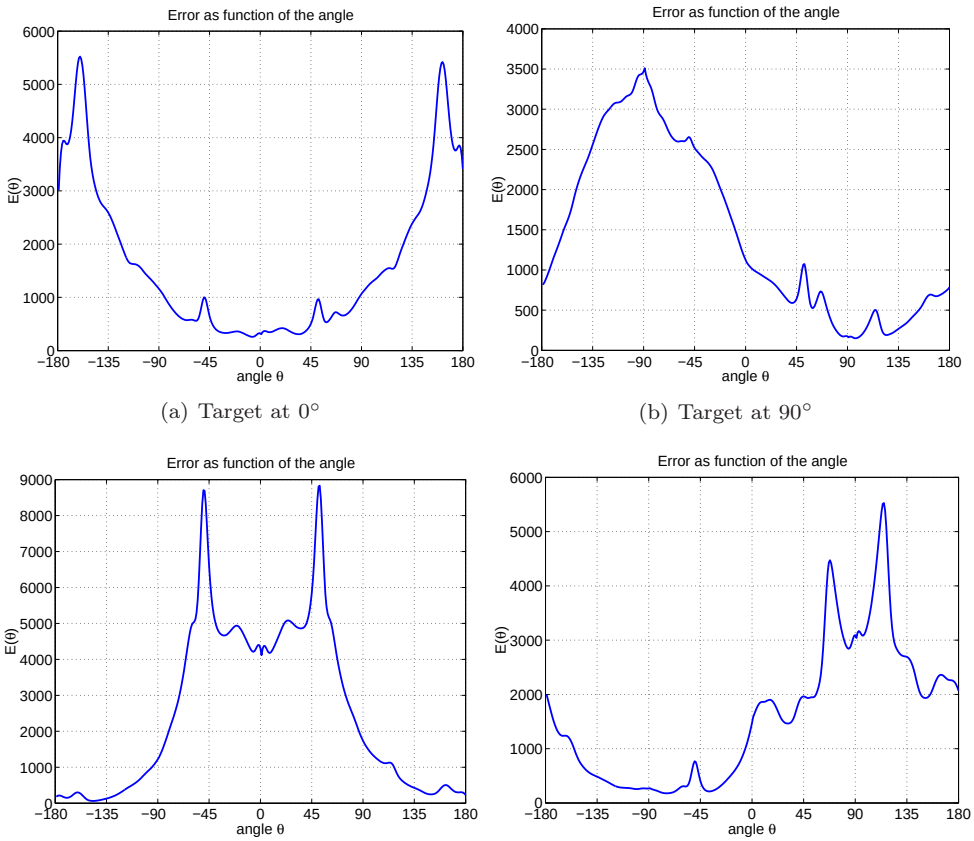
Figure 8.6: a) Antenna gains for the four faces of the FBDA. b) Difference between antenna gains at different values of angle θ.

A plot with the values of the three components of DG is reported in Figure 8.6b. $DG(\theta)$ is a vector that only depends on the radiation patterns and the antenna geometry, while DP is the obtained by measuring the powers with the scheme illustrated in Section 8.2. Once the value DP has been determined, the angle θ can be found by minimizing the difference between the two vectors:

$$\hat{\theta} = \arg\min_{\theta} \|DP - DG(\theta)\| . \tag{8.25}$$

In absence of measurement errors or other interferences, we expect that the difference between the two vector will be zero when $\hat{\theta}$ is equal to the true target angle θ, even if we cannot guarantee the uniqueness of the solution because there could be angles $\theta_1 \neq \theta_2$ such that $DG(\theta_1) = DG(\theta_2)$. In addition, the presence of interferences and measurement errors could be such that the equality $\|DP - DG(\theta)\| = 0$ is not verified by any angle. To address these uncertainties, we introduce a squared error measure defined as follow:

$$\begin{aligned} E(\theta) &= (DP - DG(\theta))'(DP - DG(\theta)) \\ &= \sum_{1}^{3} [(\mathrm{P}_{r1} - \mathrm{P}_{ri}) - (G_{r1}(\theta) - G_{ri}(\theta))]^2, \end{aligned} \tag{8.26}$$



(a) Target at 0°

(b) Target at 90°

(c) Target at 180°

(d) Target at 270°

Figure 8.7: Error $E(\theta)$ evaluated for samples of DP measured at different angles.

and we use the angle that minimize this error as an estimate of the target's angular position θ:

$$\hat{\theta} = \arg\min_{\theta} E(\theta). \tag{8.27}$$

Figure 8.7 reports the error evaluated in correspondence of different angles using power values $P_{r1}, \ldots, P_{r4}$ measured in a real setting.

8.4 Dealing With Missing Power Readings

Equation 8.27 ca be used if the four power values $P_{r1}, \ldots, P_{r4}$ are available, but in many practical situations, the communication between the target node and the base station is likely to fail when using patches pointing in directions opposite to the target node. Since one or more power levels might not be available, we redefine the error function $E(\theta)$ to take into account those missing values. We note that P_{r1}, which has been used as reference value, might be missing as well, therefore we use the patch where the maximum power is received as reference value. The new error function is defined as follows:

$$E(\theta) = E_1(\theta) + E_2(\theta) + E_3(\theta) + E_4(\theta), \tag{8.28}$$

where each of the terms $E_i(\theta)$ is given by:

$$E_i(\theta) = \begin{cases} [(\mathrm{P}_{r\,\max} - \mathrm{P}_{ri}) - (G_{r\,\max}(\theta) - G_{ri}(\theta))]^2 & \text{if } \mathrm{P}_{ri} \text{ is available} \\ 0 & \text{if } \mathrm{P}_{ri} \text{ not available } \textit{or } i = \max \end{cases} \tag{8.29}$$

8.4.1 Modified Error Function

A problem with the error function as defined in (8.28) and (8.29) is that, as the number of power readings available decreases, the error $E(\theta)$ becomes less expressive in defining the target position. In Figure 8.8a,c,e we report the error $E(\theta)$ for the same vector DP used in Figure 8.7a (target at 0°) when only three, two and one power readings are available. In this last case, no angle information can be derived (see Figure 8.8e). To deal with the problem of incomplete power readings, we modify the error function $E(\theta)$ to take into account the transceiver sensitivity P_{th} (e.g. the CC2420 has a $P_{th} = -94$dBm). The idea is that when a power value is missing, the target should be in a position such that the angular gain of the antenna attenuates the

signal below the transceiver's sensitivity P_{th}. For example, if patch 1 is the antenna that receives the maximum power (e.g. $P_{r1} = -84$ dBm), and patch 4 is unable to receive a message, we expect that the difference between G_{r1} and G_{r4} is at least greater than 10dBm (i.e. the difference between $P_{r\,\max}$ and P_{TH}). Therefore any angle θ such that $G_{r1}(\theta) - G_{r4}(\theta) \geq 10$dBm "explains" the missing value on patch 4, while angles θ's such that $G_{r1}(\theta) - G_{r4}(\theta) < 10$dBm need to be penalized in term of error function $E(\theta)$ (e.g. if $G_{r1}(\theta) - G_{r4}(\theta)$ was equal to 6dBm, on patch 4 we would expect to receive a message with power of $-84 - 6 = -90$dBm, which is 4dBm above the radio sensitivity). To modify the error function to take into account the transceiver's sensitivity, we introduce the terms:

$$\Delta_{Pi} = [(\mathrm{P}_{r\,\max} - \mathrm{P}_{ri}) - (G_{r\,\max}(\theta) - G_{ri}(\theta))], \tag{8.30}$$

$$\Delta_{Ti} = [(\mathrm{P}_{r\,\max} - \mathrm{P}_{TH}) - (G_{r\,\max}(\theta) - G_{ri}(\theta))], \tag{8.31}$$

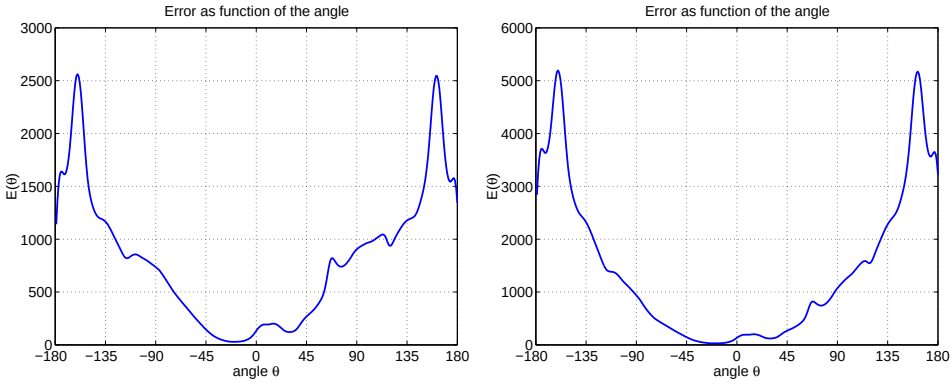
and we define the new error function as:

$$E_M(\theta) = E_{M1}(\theta) + E_{M2}(\theta) + E_{M3}(\theta) + E_{M4}(\theta), \tag{8.32}$$

where:

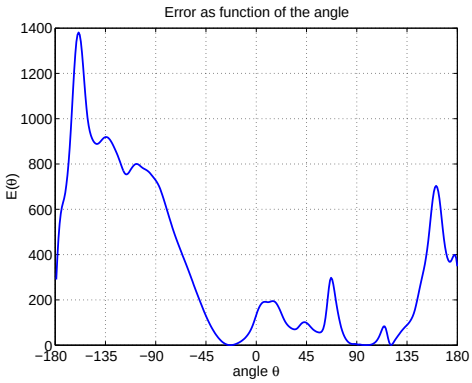
$$E_{Mi}(\theta) = \begin{cases} (\Delta_{Pi})^2 & \text{if } \mathrm{P_{ri}} \text{ available} \\ (\Delta_{Ti})^2 & \text{if } \mathrm{P_{ri}} \text{ not avail., } \Delta_{Ti} > 0, \mathrm{i} \neq \max \\ 0 & \text{else} \end{cases} \tag{8.33}$$

In Figure 8.7b,d,f we report the value of the modified error function, showing that the error profile maintain its shape when some of the power readings are missing.

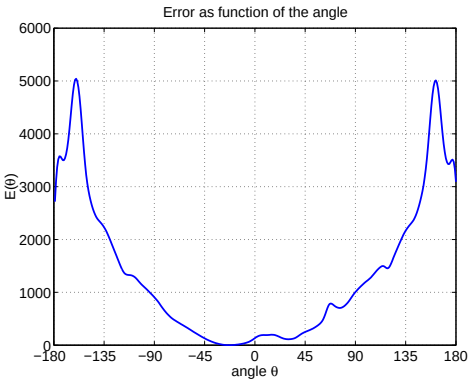


(a) $E(\theta)$ - P_{r1}, P_{r2}, P_{r4} available

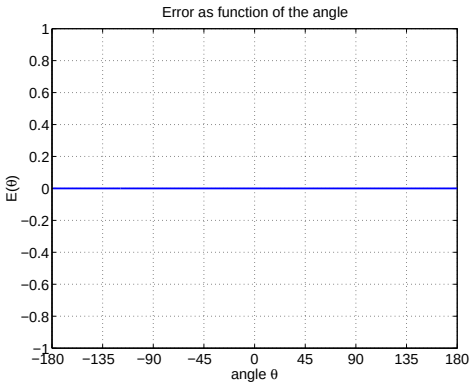
(b) $E_M(\theta)$ - P_{r1}, P_{r2}, P_{r4} available



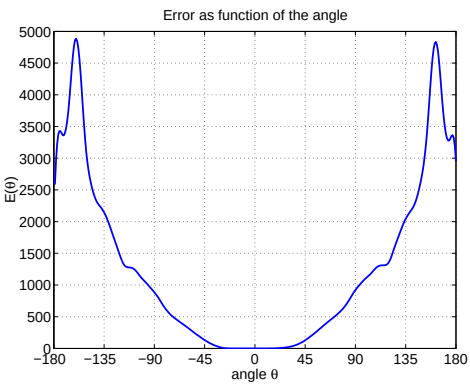
(c) $E(\theta)$ - P_{r1}, P_{r2} available



(d) $E_M(\theta)$ - P_{r1}, P_{r2} available



(e) $E(\theta)$ - P_{r1} available



(f) $E_M(\theta)$ - P_{r1} available

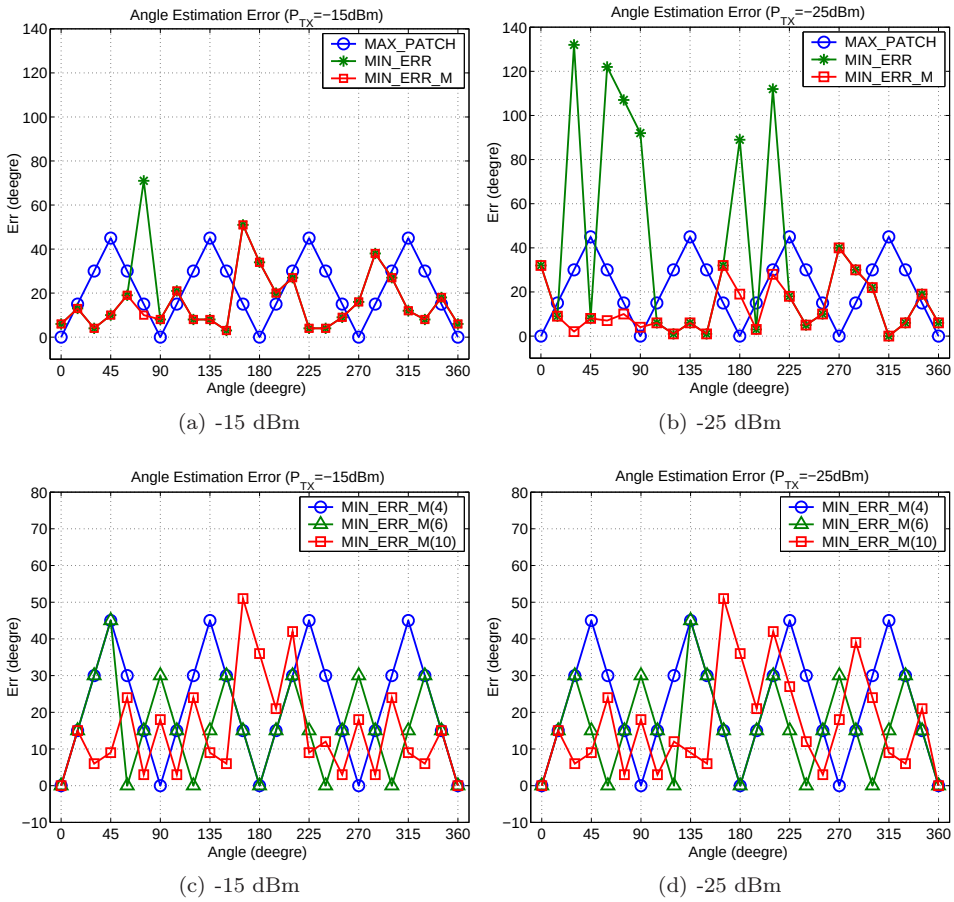
Figure 8.8: Comparison between $E(\theta)$ and $E_M(\theta)$ in case of incomplete readings.

8.5 Angle Estimation

We used data collected during in-field experiments to evaluate an angle estimation algorithm derived using the equations introduced in previous sections. Two set of measurements were made by placing the target node at about 3 meters from the base station, with both nodes elevated of 1.2 meter above the ground. The RSSI values on each patch were computed by averaging the values of 50 data packets collected using the scheme described in Section 8.2. A burst of four beacon messages was sent every 250ms with a delay time of 50ms between the activation of each patch. Two set of values were measured using a transmission power of -25dBm and -15dBm, with the FBDA antenna placed at about 3 meters from a wall (we assume that some multi-path effects was present due to the reflections from the wall). The experiments were repeated by rotating the FBDA around its vertical axis on 24 different angular position spaced by 15° each. In evaluating the angle estimation performance, we initially compared three schemes:

1. **MAX_PATCH**: the estimate is obtained using the direction of the patch that receives the maximum power. In the case of the FBDA, such estimate can only assume the values: $0°, 90°, 180°, 270°$.
2. **MIN_ERR**: the estimate is obtained by the angle that minimize the error function $E(\theta)$ defined in (8.28) .
3. **MIN_ERR_M**: the estimate is obtained by the angle that minimize the modified error function $E_M(\theta)$ defined in (8.31).

Figure 8.8a,b reports the error of the three estimation schemes for the test done using a transmission power of -15dBm and -25dBm. We note that the error of the algorithm that use the direction on the patch with the stronger signal (MAX_PATCH) oscillates between 0° and 45° and appears to be not affected by the transmission power. The error of scheme based on MIN_ERR instead, increases noticeably when the transmission power is reduced because of the errors due to missing power levels



(a) -15 dBm (b) -25 dBm

(c) -15 dBm (d) -25 dBm

Figure 8.9: Angle estimation error using different schemes.

(cf. Section 8.4). This problem instead does not affect the algorithm based on the modified error function $E_M(\theta)$ which achieve comparable results in both cases.

Using the modified error function $E_M(\theta)$ we have also implemented a different version of the algorithm (MIN_ERR_M(n)) that operates by dividing the 360° horizon in n sectors and averaging the error on each of them. The median angle of the sector with minimum error is used as estimate for the angle. Figure 8.8c,d reports the error when using 4, 6 and 10 sectors. Table 8.1 reports the mean and standard deviation of the error using the different algorithms.

	-15 dBm		-25 dBm	
	mean	**std**	**mean**	**std**
MAX_PATCH	21.60	15.05	21.60	15.49
MIN_ERR	17.80	16.38	36.32	43.70
MIN_ERR_M	15.36	12.11	12.96	11.65
MIN_ERR_M(4)	21.60	15.05	21.60	15.05
MIN_ERR_M(6)	15.60	12.61	15.60	12.60
MIN_ERR_M(8)	18.00	16.20	15.00	13.00
MIN_ERR_M(10)	14.64	13.20	16.56	13.20
MIN_ERR_M(12)	14.40	14.02	27.60	14.01
MIN_ERR_M(16)	17.40	14.64	26.70	14.63

Table 8.1: Average Error (degrees) and Standard Deviation (degrees) of different estimation algorithms using two sets of experimental data (-15dBm and -25dBm).

8.6 Conclusion

The angle estimation results using experimental data show that MIN_ERR_M, the scheme that operates by selecting the minimum of the modified error function, $\hat{\theta} = \arg\min_{\theta} E_M(\theta)$, achieves the best performances in terms of average error and standard deviation. In the experiment using a TX power of -15dBm, the average error was equal to 15.36° with a standard deviation of 12.11°. In a similar experiment using a TX power of -25dBm, the average error was equal to 12.96° with a standard deviation of 11.65°. Compared to the MAX_PATCH estimation scheme, the MIN_ERR_M reduces the estimation error by a value comprised between the 28% and 40%. Although more experimental tests would be needed to better characterize the proposed scheme, these first preliminary results show that angle estimation used the FBDA is possible with a satisfactory accuracy.

Chapter 9

Conclusions

Localization in sensor networks is a task that consists in computing the node positions on the basis of a limited amount of initial information. In this thesis a novel method based on the Self-Organizing Map formalism has been proposed to solve the problem using radio connectivity data and no a-priori information on anchor nodes. The algorithm, although centralized, is characterized by a lightweight implementation that makes it suitable for execution on inexpensive sensor nodes with limited computational resources.

The directive antenna that we evaluate in the last chapters of this thesis is also designed to work with resource-constrained nodes. Using experimental measures and theoretical models, we have demonstrated that the antenna is effective in improving communication among sensor nodes. In addition, the directivity of this inexpensive antenna has been used to implement an algorithm capable of estimating the angular position of a nearby node, deriving information useful in solving the localization problem.

9.1 Summary of Contributions

- A novel, range-free, anchor-free algorithm has been proposed to solve the localization problem using only connectivity information, and without relying on the presence of anchor nodes.

- Using extensive simulations we showed that the solution produces virtual maps that are useful when used for tasks such as geographical routing, with results that are very close to those found when the true network coordinates are known.

- A modified version of the basic algorithm was proposed to efficiently use anchor information (if available). The solution generates accurate absolute maps: using only four anchor nodes, the localization error drops below 0.3 R (i.e. 30% of the communication range) for networks with average connectivity of only four nodes. Simulation results show that the proposed scheme outperforms the popular MDS technique, with an error reduction up to 43% in networks with low connectivity and up to 75% in networks with anisotropic layout.

- We analytically demonstrate that the proposed scheme has low computation and communication overheads; hence, making it suitable for resource-constrained networks. Benchmark tests on COTS sensor nodes demonstrated that a network with 100 nodes could be localized in less than 3 minutes.

- We experimentally evaluated the use of a directive antenna designed to meet the cost and size constraints of sensor nodes. The experiments were performed to characterize the antenna in conditions of large-scale and small-scale fading. Results were explained using theoretical fading models showing good agreement between measured and calculated data. The model parameters derived in this study will be useful in future system simulations.

- An algorithm to estimate the angular position of a target node has been proposed. The algorithm evaluates the signal power received on the four faces to estimate the angular position of a nearby node. Results on a first set of ex-

perimental tests show that angle estimation is possible with an average error of about 15°.

9.2 Future Work

Future extensions to this work will cover the following points:

- To collect additional experimental data in order to provide a more accurate evaluation of the angle estimation algorithm proposed in Section 8.5.
- To extend the SOM algorithm to include angle information derived using the directional antenna characterized in the last two chapters of this thesis. The trade-off between the number of directional antennas and localization accuracy should be evaluated to provide guidelines for future in-field applications.
- To propose a distributed version of the same algorithm, where each node participate in the computation solving part of the problem. The number of radio messages exchanged among nodes should be minimized to preserve the energy budget within the network.

Bibliography

[1] F. Adelstein, S. Gupta, G. Richard, and L. Schwiebert. *Fundamentals of mobile and pervasive computing*. McGraw-Hill New York, 2005.

[2] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: a survey. *Computer Networks*, 38(4):393–422, 2002.

[3] Ansoft. HFFS. http://www.ansoft.com/products/hf/hfss/.

[4] A. Arora, P. Dutta, S. Bapat, V. Kulathumani, H. Zhang, V. Naik, V. Mittal, H. Cao, M. Demirbas, M. Gouda, et al. A line in the sand: a wireless sensor network for target detection, classification, and tracking. *Computer Networks*, 46(5):605–634, 2004.

[5] A. Arora, R. Ramnath, E. Ertin, P. Sinha, S. Bapat, V. Naik, V. Kulathumani, H. Zhang, H. Cao, M. Sridharan, et al. Exscal: Elements of an extreme scale wireless sensor network. *IEEE RTCSA*, 2005.

[6] P. Bahl and V. Padmanabhan. RADAR: an in-building RF-based user location and tracking system. *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, 2, 2000.

[7] C. A. Balanis. *Antenna theory analysis and design*. J. Wiley and Sons, New York, 1997.

[8] S. Bandyopadhyay, S. Roy, and T. Ueda. *Enhancing the Performance of Ad Hoc Wireless Networks with Smart Antennas*. CRC Press, 2006.

[9] L. Bao and J. Garcia-Luna-Aceves. Transmission scheduling in ad hoc networks with directional antennas. *Proceedings of the 8th annual international conference on Mobile computing and networking*, pages 48–58, 2002.

[10] BB Electronics and Sensicast. August 2005 wireless survey results. http://www.bb-elec.com/wirelesssurvey/.

[11] Berg Insight AB. Mobile personal navigation services (technical report). http://www.gii.co.jp/english/ber39982-personal-navi.html, 2006.

[12] E. Bonabeau and F. Henaux. Self-Organizing Maps for Drawing Large Graphs. *Information Processing Letters*, 67(4):177–184, 1998.

[13] I. Borg and P. Groenen. *Modern multidimensional scaling*. Springer New York, 1997.

[14] H. Breu and D. Kirkpatrick. Unit disk graph recognition is NP-hard. *Computational Geometry: Theory and Applications*, 9(1-2):3–24, 1998.

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

[16] N. Bulusu, J. Heidemann, and D. Estrin. Adaptive beacon placement. *Proceedings of the 21st International Conference on Distributed Computing Systems (ICDCS-21)*, pages 489–498, 2001.

[17] S. Capkun, M. Cagalj, and M. Srivastava. Securing localization with hidden and mobile base stations. *Proceedings of IEEE INFOCOM*, 2006.

[18] E. Cayirci and T. Cuplu. SENDROM: Sensor Networks for Disaster Relief Operations Management. *Proc of Med-Hoc Net 2004, the Third Annual Mediterranean Ad Hoc Networking Workshop.*, 2004.

[19] Chipcon. Chipcon cc2420 datasheet. http://www.chipcon.com/files/CC2420_Data_Sheet_1_4.pdf.

[20] R. Choudhury and N. Vaidya. Ad Hoc Routing Using Directional Antennas. *Illinois Computer Systems Symposium (ICSS) May 2002, UIUC*, 2002.

[21] R. Choudhury, X. Yang, R. Ramanathan, and N. Vaidya. Using directional antennas for medium access control in ad hoc networks. *Proceedings of ACM MOBICOM*, 2, 2002.

[22] M. Cottrell, J. Fort, and G. Pages. Two or three things that we know about the Kohonen algorithm. *Proc. ESANN, European Symp. on Artificial Neural Networks*, pages 235–244, 1994.

[23] K. Delin. The Sensor Web: A Macro-Instrument for Coordinated Sensing. *Sensors*, 2(1):270–285, 2002.

[24] T. Dimitriou and A. Kalis. Efficient delivery of information in sensor networks using smart antennas. *1st International Workshop on Algorithmic Aspects of Wireless Sensor Networks (ALGOSENSORS 2004)*, pages 109–22, 2004.

[25] T. Eren, D. Goldenberg, W. Whiteley, Y. Yang, A. Morse, B. Anderson, and P. Belhumeur. Rigidity, computation, and randomization in network localization. *Proc. IEEE INFOCOM (March 2004)*, 2004.

[26] E. Ertin and K. Priddy. Self-localization of wireless sensor networks using self-organizing maps. *Proceedings of SPIE*, 5803:138, 2005.

[27] U. Gerecke and N. Sharkey. Quick and dirty localization for a lost robot. *Computational Intelligence in Robotics and Automation, 1999. CIRA'99. Proceedings. 1999 IEEE International Symposium on*, pages 262–267, 1999.

[28] D. Goldenberg, A. Krishnamurthy, W. Maness, Y. Yang, A. Young, A. Morse, A. Savvides, and B. Anderson. Network localization in partially localizable networks. *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, 1, 2005.

[29] T. He, C. Huang, B. Blum, J. Stankovic, and T. Abdelzaher. Range-free localization schemes for large scale sensor networks. *Proceedings of the 9th annual international conference on Mobile computing and networking*, pages 81–95, 2003.

[30] B. Hendrickson. Conditions for Unique Graph Realizations. *SIAM J. Comput.*, 21(1):65–84, 1992.

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

[32] J. Hightower, R. Want, and G. Borriello. SpotON: An indoor 3D location sensing technology based on RF signal strength. *UW CSE 00-02-02, University of Washington, Department of Computer Science and Engineering, Seattle, WA, February*, 2000.

[33] B. Hofmann-Wellenhof, H. Lichtenegger, and J. Collins. *Global Positioning System: Theory and Practice*. Springer, 1993.

[34] Z. Huang, C. Shen, C. Srisathapornphat, and C. Jaikaeo. Topology control for ad hoc networks with directional antennas. *Computer Communications and Networks, 2002. Proceedings. Eleventh International Conference on*, pages 16–21, 2002.

[35] C. Intanagonwiwat, R. Govindan, and D. Estrin. Directed diffusion: A scalable and robust communication paradigm for sensor networks. *Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking*, pages 56–67, 2000.

[36] B. Jackson and T. Jordan. Connected rigidity matroids and unique realizations of graphs. *Manuscript, March*, 2003.

[37] D. Jacobs and B. Hendrickson. An algorithm for two-dimensional rigidity percolation: the pebble game. *Journal of Computational Physics*, 137(2):346–365, 1997.

[38] J. Janet, R. Gutierrez, T. Chase, M. White, and J. Sutton. Autonomous mobile robot global self-localization using Kohonen and region-feature neural networks. *Journal of Robotic Systems*, 14(4):263–282, 1997.

[39] X. Ji and H. Zha. Sensor positioning in wireless ad-hoc sensor networks using multidimensional scaling. *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, 4, 2004.

[40] I. Kang and R. Poovendran. Design issues on broadcast routing algorithms using realistic cost-effective smart antenna models. *Vehicular Technology Conference, 2004. VTC 2004-Spring. 2004 IEEE 59th*, 4, 2004.

[41] B. Karp and H. Kung. GPSR: greedy perimeter stateless routing for wireless networks. *Proceedings of the 6th annual international conference on Mobile computing and networking*, pages 243–254, 2000.

[42] S. Kaski, J. Kangas, and T. Kohonen. Bibliography of self-organizing map (SOM) papers: 1981–1997. *Neural Computing Surveys*, 1(3&4):1–176, 1998.

[43] L. Kleinrock and J. Silvester. Optimum transmission radii for packet radio networks or why six is a magic number. *NTC'78; National Telecommunications Conference, Birmingham, Ala., December 3-6, 1978, Conference Record.*, 1:A79–40501, 1978.

[44] T. Kohonen. Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, 43(1):59–69, 1982.

[45] T. Kohonen. Things you haven't heard about the self-organizing map. *Neural Networks, 1993., IEEE International Conference on*, pages 1147–1156, 1993.

[46] T. Korakis, G. Jakllari, and L. Tassiulas. A MAC protocol for full exploitation of directional antennas in ad-hoc wireless networks. *Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing*, pages 98–107, 2003.

[47] B. Krishnamachari, S. Wicker, R. Bejar, and M. Pearlman. Communications, information and network security, ch. Critical Density Thresholds in Distributed Wireless Networks, 2002.

[48] F. Kuhn, T. Moscibroda, and R. Wattenhofer. Unit disk graph approximation. *Proceedings of the 2004 joint workshop on Foundations of mobile computing*, pages 17–23, 2004.

[49] G. Laman. On graphs and rigidity of plane skeletal structures. *Journal of Engineering Mathematics*, 4(4):331–340, 1970.

[50] K. Langendoen and N. Reijers. Distributed localization in wireless sensor networks: a quantitative comparison. *Computer Networks*, 43(4):499–518, 2003.

[51] L. Lazos and R. Poovendran. SeRLoc: Robust localization for wireless sensor networks. *ACM Transactions on Sensor Networks (TOSN)*, 1(1):73–100, 2005.

[52] L. Lazos, R. Poovendran, and S. Čapkun. ROPE: robust position estimation in wireless sensor networks. *Proceedings of the 4th international symposium on Information processing in sensor networks*, 2005.

[53] D. Leang and A. Kalis. Smart SensorDVBs: Sensor Network Development Boards with Smart Antennas. *International Conference on Communications, Circuits and Systems*, 2:1476–1480, 2004.

[54] P. Levis. TinyOS: An Open Operating System for Wireless Sensor Networks (Invited Seminar). *Proceedings of the 7th International Conference on Mobile Data Management (MDM'06)-Volume 00*, 2006.

[55] J. Li, J. Jannotti, D. De Couto, D. Karger, and R. Morris. *A scalable location service for geographic ad hoc routing.* ACM Press New York, NY, USA, 2000.

[56] Z. Li, W. Trappe, Y. Zhang, and B. Nath. Robust statistical methods for securing wireless localization in sensor networks. *Proceedings of the 4th international symposium on Information processing in sensor networks*, 2005.

[57] D. Liu, P. Ning, and W. Du. Attack-resistant location estimation in sensor networks. *Information Processing in Sensor Networks, 2005. IPSN 2005. Fourth International Symposium on*, pages 99–106, 2005.

[58] K. Lorincz and M. Welsh. MoteTrack: A Robust, Decentralized Approach to RF-Based Location Tracking. *Proceedings of Intl. Workshop on Location and Context-Awareness*, 2005.

[59] Z. Lotker, M. de Albeniz, and S. Perennes. Range-Free Ranking in Sensors Networks and Its Applications to Localization. *Proceedings of 3 rdAd-Hoc, Mobile, and Wireless Networks: Third International Conference (ADHOC-NOW)*, pages 158–171, 2004.

[60] N. Malhotra, M. Krasniewski, C. Yang, S. Bagchi, and W. Chappell. Location Estimation in Ad Hoc Networks with Directional Antennas. *Proceedings. 25th IEEE International Conference on Distributed Computing Systems (ICDCS 2005)*, pages 633–642, 2005.

[61] B. Meyer. Self-organizing graphs-a neural network perspective of graph layout. *Proceedings of the 6th International Symposium on Graph Drawing*, pages 246–262, 1998.

[62] D. Moore, J. Leonard, D. Rus, and S. Teller. Robust distributed network localization with noisy range measurements. *Proceedings of the 2nd international conference on Embedded networked sensor systems*, pages 50–61, 2004.

[63] T. Moscibroda, R. O'Dell, M. Wattenhofer, and R. Wattenhofer. Virtual coordinates for ad hoc and sensor networks. *Proceedings of the 2004 joint workshop on Foundations of mobile computing*, pages 8–16, 2004.

[64] Moteiv Corporation. Telos (rev b) datasheet. http://www.moteiv.com, 2004.

[65] K. Muthukrishnan, M. Lijding, and P. Havinga. Towards Smart Surroundings: Enabling Techniques and Technologies for Localization. 2005.

[66] R. Nagpal, H. Shrobe, and J. Bachrach. Organizing a global coordinate system from local information on an ad hoc sensor network. *Proc. of Information Processing in Sensor Networks (IPSN)*, 2003.

[67] A. Nasipuri and K. Li. A directionality based location discovery scheme for wireless sensor networks. *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, pages 105–111, 2002.

[68] J. Newsome, E. Shi, D. Song, and A. Perrig. The Sybil attack in sensor networks: analysis & defenses. *Information Processing in Sensor Networks, 2004. IPSN 2004. Third International Symposium on*, pages 259–268, 2004.

[69] D. Niculescu and B. Nath. Ad hoc positioning system (APS) using AOA. *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies. IEEE*, 3, 2003.

[70] D. Niculescu and B. Nath. DV Based Positioning in Ad Hoc Networks. *Telecommunication Systems*, 22(1):267–280, 2003.

[71] D. Niculescu and B. Nath. Error characteristics of ad hoc positioning systems (aps). *Proceedings of the 5th ACM international symposium on Mobile ad hoc networking and computing*, pages 20–30, 2004.

[72] R. O'Dell and R. Wattenhofer. Theoretical aspects of connectivity-based multi-hop positioning. *Theoretical Computer Science*, 344(1):47–68, 2005.

[73] M. Oja, S. Kaski, and T. Kohonen. Bibliography of self-organizing map (SOM) papers: 1998-2001 addendum. *Neural Computing Surveys*, 3(1):1–156, 2003.

[74] J. Polastre, R. Szewczyk, and D. Culler. Telos: enabling ultra-low power wireless research. *Proceedings of the 4th international symposium on Information processing in sensor networks*, 2005.

[75] N. Priyantha, A. Chakraborty, and H. Balakrishnan. The Cricket location-support system. *Proceedings of the 6th annual international conference on Mobile computing and networking*, pages 32–43, 2000.

[76] B. Raman, K. Chebrolu, N. Madabhushi, D. Gokhale, P. Valiveti, and D. Jain. Implications of link range and (In) stability on sensor network architecture. *Proceedings of the 1st international workshop on Wireless network testbeds, experimental evaluation & characterization*, pages 65–72, 2006.

[77] T. S. Rappaport. *Wireless Communications: Principles and Practice*. P T R Prentice-Hall, pub-PHPTR:adr, second edition, 2002.

[78] K. Romer. The Lighthouse Location System for Smart Dust. *MobiSys 2003*, 2003.

[79] C. Santivanez and J. Redi. On the use of directional antennas for sensor networks. *Military Communications Conference, 2003. MILCOM 2003. IEEE*, 1, 2003.

[80] A. Savvides, C. Han, and M. Strivastava. Dynamic fine-grained localization in Ad-Hoc networks of sensors. *Proceedings of the 7th annual international conference on Mobile computing and networking*, pages 166–179, 2001.

[81] Y. Shang and W. Ruml. Improved MDS-based localization. *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, 4, 2004.

[82] Y. Shang, W. Ruml, Y. Zhang, and M. Fromherz. Localization from mere connectivity. *Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing*, pages 201–212, 2003.

[83] S. Y. Shin, S. Choi, H. S. Park, and W. H. Kwon. Lpacket error rate analysis of IEEE 802.15.4 under IEEE 802.11b interference. In *Wired/Wireless Internet Communications, Third International Conference, WWIC 2005, Xanthi, Greece, May 11-13, 2005, Proceedings*, pages 279–288, 2005.

[84] A. Spyropoulos and C. Raghavendra. Energy efficient communications in ad hoc networks using directional antennas. *INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, 1, 2002.

[85] J. Stafford, editor. *Precision Agriculture - an International Journal on Advances in the Science of Precision Agriculture*. Springer, 1999-2006.

[86] R. Stoleru, T. He, J. Stankovic, and D. Luebke. A high-accuracy, low-cost localization system for wireless sensor networks. *Proceedings of the 3rd international conference on Embedded networked sensor systems*, pages 13–26, 2005.

[87] R. Stoleru, T. He, and J. A. Stankovic. *"Secure Localization and Time Synchronization for Wireless Sensor and Ad Hoc Networks"*, volume 30, chapter Range-free Localization. Springer, 2006.

[88] R. Stoleru and J. Stankovic. Probability grid: a location estimation scheme for wireless sensor networks. *Sensor and Ad Hoc Communications and Networks, 2004. IEEE SECON 2004. 2004 First Annual IEEE Communications Society Conference on*, pages 430–438, 2004.

[89] M. Takai, R. Bagrodia, and A. Ren. Directional virtual carrier sensing for directional antennas in mobile ad hoc networks. *Proceedings of the 3rd ACM international symposium on Mobile ad hoc networking & computing*, pages 183–193, 2002.

[90] K. Wang, J. Proakis, and R. Rao. Energy-Efficient Routing Algorithms Using Directional Antennas for Mobile Ad Hoc Networks. *International Journal of Wireless Information Networks*, 9(2):105–118, 2002.

[91] A. Ward, A. Jones, and A. Hopper. A new location technique for the active office. *Personal Communications, IEEE [see also IEEE Wireless Communications]*, 4(5):42–47, 1997.

[92] G. Werner-Allen, K. Lorincz, M. Ruiz, O. Marcillo, J. Johnson, J. Lees, and M. Welsh. Deploying a wireless sensor network on an active volcano. *IEEE Internet Computing*, 10(2):18–25, 2006.

[93] K. Whitehouse. The Design of Calamari: an Ad-hoc Localization System for Sensor Networks. *University of California at Berkeley*, 2002.

[94] C. Yang, S. Bagchi, and W. Chappell. Location Tracking with Directional Antennas in Wireless Sensor Networks. *Microwave Symposium Digest, 2005 IEEE MTT-S International*, pages 131–134, 2005.

[95] C. Yang, J. Mastarone, and W. Chappell. Directional antennas for angular diversity in wireless sensor networks. *Antennas and Propagation Society International Symposium, 2005 IEEE*, 4, 2005.

[96] S. Yi, Y. Pei, and S. Kalyanaraman. On the capacity improvement of ad hoc wireless networks using directional antennas. *Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing*, pages 108–116, 2003.