

An Analysis of Localization Problems and Solutions in Wireless Sensor Networks

Lina M. Pestana Leão de Brito, Laura M. Rodríguez Peralta¹
lmrodrig@uma.pt

(recebido em 4 de Abril de 2008; aceite em 8 de Maio de 2008)

Resumo: Numa Rede de Sensores sem Fios, a localização consiste em determinar a posição de um nó sensor. Determinar a localização é uma tarefa de grande importância, pois é necessária para muitas outras funções da rede: comunicação, encaminhamento, criação de clusters, cobertura da rede, etc. Além disso, é essencial que este processo seja desempenhado pelos próprios nós, ou seja, sem qualquer intervenção humana (auto-localização), o que requer colaboração entre os nós. Neste artigo, começamos por abordar algumas das problemáticas que podem influenciar quer a criação quer a escolha de um algoritmo de localização. Apresentamos um resumo das várias técnicas utilizadas para localizar sensores sem fios, assim como uma comparação dos algoritmos de localização que consideramos mais relevantes. Esta comparação tem em vista a selecção de uma solução de localização que seja adequada ao caso específico da monitorização ambiental de florestas, no âmbito do projecto europeu Foresmac.

Palavras-chave: Algoritmos de localização, Auto-localização, Colaboração, Nós sensores, Redes de Sensores sem Fios.

Abstract: Determining the localization of nodes in a Wireless Sensor Network is a very important task, which involves collaboration between sensor nodes. Localization is a fundamental service since it is relevant to many applications and to the network main functions, such as: routing, communication, cluster creation, network coverage, etc. Collaboration is essential to self-localization, so

¹ Laboratory for Usage-centered Software Engineering (LabUse), Centro de Ciências Matemáticas (CCM), Mathematics and Engineering Department (DME), University of Madeira (UMa), Campus da Penteadá, 9000-390 Funchal, Madeira, Portugal

that localization can be accomplished by the nodes themselves, without any human intervention. In this paper, we first analyze the key aspects that have to be considered when designing or choosing a solution for the localization problem. Then, we present the types of current localization algorithms, making a broad comparison among the most relevant algorithms. With this comparative analysis, we aim at identifying a localization algorithm suitable to the specific problem of forests environmental monitoring, to be applied in the context of the Foresmac European project.

Keywords: Collaboration, Localization algorithms, Self-localization, Sensor nodes, Wireless sensor networks.

1. Introduction

A Wireless Sensor Network (WSN) consists of a large number of tiny wireless sensor nodes (often referred to as *sensor nodes* or, simply, *nodes*) that are, typically, densely deployed. Nodes measure the ambient conditions in the environment surrounding them. These measurements are, then, transformed into signals that can be processed to reveal some characteristics about the phenomenon.

The data collected is routed to special nodes, called sink nodes (also called Base Station, BS), in a multi-hop basis, as shown in Fig. 1. Then, typically, the sink node sends data to the user via Internet or satellite, through a gateway. Though, depending on the distance between the user and the network, a gateway might not be needed (local monitoring).

Combining the advantages of wireless communication with some computational capabilities, WSNs allow for a wider variety of applications than traditional networks: environmental monitoring, health, surveillance, catastrophe monitoring, structural monitoring, security, military, industry, agriculture, home, traffic monitoring, etc. Nevertheless, opposing to traditional networks, WSNs are useful only if sensor nodes are aware of the environment surrounding them. For instance, each sensor could only monitor its region and send the collected data to the sink node. However, the great potential of WSNs lies in its ability to correlate collected data in time and in space. This is the reason why synchronization and localization are fundamental tools to WSNs (Broxton, Lifton, and Paradiso, 2005; Hu & Servetto, 2005). In this paper, we will focus only on the localization problem.

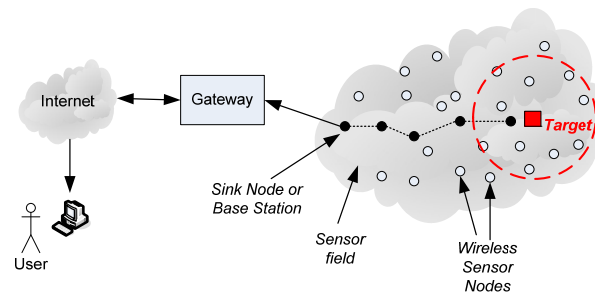


Fig. 1. Wireless Sensor Network.

Localization refers to the ability of determining the position (relative or absolute) of a sensor node, with an acceptable accuracy. In a WSN, localization is a very important task; however, localization is not the goal of the network. In fact, localization is a fundamental service since it is relevant to many applications (target tracking, intruder detection, environmental monitoring, etc.), which depend on knowing the location of nodes. Localization is also relevant to the network main functions: communication, geographical routing, cluster creation, network coverage, etc. Even collaboration typically depends on localization of nodes.

Collaboration is essential to self-localization, so that localization can be accomplished by the nodes themselves, without any human intervention (Sheng & Hu, 2003). However, due to the resource limitations of wireless sensor nodes, they have to collaborate in order to accomplish several other tasks, such as: signal processing (Ramanathan, Saluja, and Hu, 2002), communication, time synchronization (Hu & Servetto, 2005), sensors repositioning (Huang & Tseng, 2005; Wang et al., 2006), energy savings (Huang & Tseng, 2005), calibration and heuristics (Reghelin & Fröhlich, 2006), etc.

According to Ramanathan et al. (2002), WSNs originated a new collaboration concept. Traditionally, collaboration exists within the same group of nodes, even though they move (node-centric computing). In WSNs, collaboration occurs among

nodes located in a certain region, which means that the group of nodes may not be the same (location-centric computing). For instance, if a node leaves a predefined region, it stops collaborating with other nodes. However, besides localization-based collaboration, it is possible to identify other ways to collaborate, based whether in monitoring a certain phenomenon or in the hardware characteristics of the nodes themselves (Hussain, Farooq, Zia, and Akhlaq, 2004; Ranjan, Kumar, Rammurthy, and Srinivas, 2005; Medidi, Slaaen, Zhou, Mallery, and Medidi, 2006; Zhou, Cui, and Zhou, 2006).

In this paper, we analyze the key aspects that have to be considered when designing or choosing a solution for the localization problem. Then, we describe the types of current localization algorithms. We also analyse the most relevant algorithms and compare them.

In our specific case, this comparison allows us to find a solution for the localization problem to be applied in the context of the Foresmac project (Project INTERREG III B, 05/MAC/2.3/C16), an European project based on the use of WSNs for forest environmental monitoring. However, the importance of this comparative analysis relies in allowing other authors to identify the localization algorithm which best suits their specific problem. In this paper, we also illustrate the use of this comparative analysis to identify a more complete algorithm, able to adapt itself to a wide variety of possibilities (number and density of nodes, obstacles and terrain irregularities, network topology, node mobility, etc.).

This paper is organized as follows. In section 2, we present the key aspects of localization algorithms. In section 3, the types of localization algorithms are described. Section 4 presents a comparative analysis of these algorithms, according to the aspects described in sections 2 and 3. Section 5 provides some conclusions and future work perspectives.

2. Key Aspects of Localization Algorithms

In this section, we analyze the aspects that must be considered when designing or choosing a localization algorithm, such as: limited resources, number and density of nodes, network topology, existence of obstacles or terrain irregularities, types of

signals used, and node mobility (if nodes are stationary or mobile). Many of the existing algorithms can achieve good localization accuracy. But, usually, they can only adapt to some of these key aspects.

2.1 Limited resources

Typically, a sensor node consists in four basic components: sensing, processing, transceiver (transmitter and receiver) and power (usually, a battery) units, as illustrated in Fig. 2. Recent advancements allow for the current generation of sensor nodes to become even smaller and cheaper (Akyildiz, Su, Sankarasubramaniam, and Cayirci, 2002). Consequently, nodes have reduced memory and processing capacities. Battery is, typically, limited. Moreover, due to a short transmission range (caused by restrained transmission power), nodes can only communicate with its local neighbours.

A good localization solution must consider all these resource limitations (minimizing energetic, computational, transmission and hardware costs), conducting however to accurate localization results.

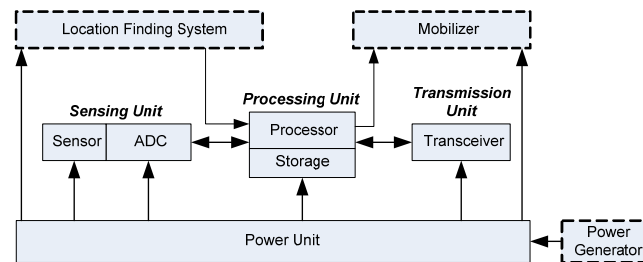


Fig. 2. Wireless Sensor Node.

2.2 Number and density of nodes

Most localization algorithms are sensitive to the number of nodes and/or to the density of nodes (amount of localizing nodes per area unit).

On one hand, some algorithms can not apply to low density WSNs, (for instance, algorithms that depend on the distance measurements to anchor nodes or hop count based algorithms) because they will cause considerable localization errors. On the

other hand, when using some algorithms in high density WSNs, localization can be an expensive process, in the sense that it can bring high cost and high delay. Furthermore, in high density networks, localization should conduct to even more accurate results (Ji & Zha, 2004). Thus, it is critical to consider the number of nodes and the node density when creating or choosing a localization algorithm.

2.3 Network topology adaptability

Some of the localization algorithms calculate the Euclidian distance between a pair of nodes, considering the shortest path between them. However, this is only valid to such cases where shortest paths are similar to a straight line. In networks where the deployed area has a concave topology (typically, S or C shape environments, such as: rivers, valleys, city streets, etc.), this is normally not valid, conducting to distorted results (Wang & Xiao, 2006). Even though a good approximation can still be achieved for some nodes (for. e.g., distance between nodes A and B, in Fig. 3), for others the Euclidean distance differs significantly from the length of the shortest path (for. e.g., distance between nodes C and D, in Fig. 3).

Some localization algorithms may also not deal well localizing nodes which are positioned in the limits of the WSN area. In this case, the distance information each node can obtain is less and with lower quality (because all the distance information comes from the same side of the network).

In either case, nodes may even not be localizable. Nevertheless, assuming nodes are localizable, localization results can be severely corrupted. So, a good localization algorithm must adapt to the network topology.

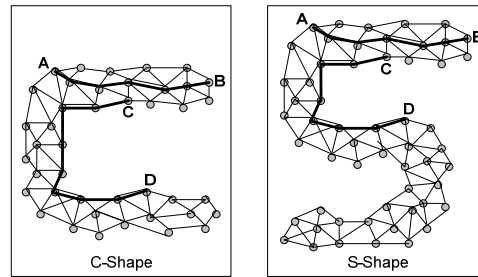


Fig. 3. Example of typical concave network topologies (C and S shape)

2.4 Obstacles and irregularities

Most of the proposed localization solutions have low accuracy in obstructed environments (Wang & Xiao, 2006). This is due to the existence of obstacles, which obstruct the line-of-sight between nodes (for instance, some nodes may not be able to see anchor nodes which make localization unfeasible). Obstacles and terrain irregularities can also cause signal reflections, what leads to wrong distance estimations. This problem exists either in indoor or outdoor environments. Hence, localization algorithms must be able to cope with it.

Most localization solutions achieve good results in uniformly distributed WSNs. Conversely, only some solutions allow a good performance in irregular networks. Therefore, this is also an important aspect.

2.5 Stationary or mobile nodes

In most WSNs, nodes are stationary. As a result, most localization algorithms are designed specifically to this kind of networks. However, due to the emergence of new applications, algorithms should adapt to the existence of mobile nodes.

Instead of considering node mobility an extra problem, some solutions take advantage of it to support the localization process. Mobility can help overcome several typical problems, such as: low node density, obstacles, concave topologies, etc.

2.6 Type of signals

It is really important to know the type of signal that is going to be used by an algorithm, due to its propagation characteristics. Usually, localization algorithms use one type of signal, among three possibilities: radio, acoustic and ultrasounds. However, there are several hybrid solutions and there is also the possibility of using light to measure the distance between two nodes (Broxton et al., 2005).

Most solutions use radio signals (radio frequency). Yet, these signals are deeply affected by the environment due, either to the presence of obstacles or to the multipath phenomenon. Also, the heterogeneity of the channel, which depends on the propagation direction, cannot be ignored. Typically, atmospheric conditions, like temperature, cause little impact on the signal. However, rain can affect the signal. Particularly, radio-based solutions that make use of the received signal strength to measure distance between nodes lead to low accurate and low reliable results.

Another way to measure distance between nodes is to use ultrasounds. Assuring that there is line-of-sight between nodes, measured distances are very close to the real ones. However, it is only possible to generalize the use of ultrasounds in WSNs overcoming two challenges (Wang, Ding, and Xiao, 2006). The first challenge concerns with the short transmission range of ultrasounds, due to energy limitations of nodes. The second challenge is associated with the amount and dimension of errors inherent to obstructed environments. Moreover, these solutions require all nodes to be equipped with an ultrasound receiver. Overcoming these challenges, ultrasound-based approaches can be more reliable and accurate than radio-based approaches (Wang et al., 2006).

The solutions based in acoustic signals (Bergamo, Asgari, Wang, Maniezzo, Yip, Hudson, Yao, and Estrin, 2004; Kwon, Mechitov, Sundresh, Kim, and Agha, 2005) have some advantages. Nodes only need to have a speaker and a microphone to measure inter-node distances. As most nodes already have both, it is a cheap solution. Furthermore, as sound propagation is simple (isotropic) and has predictable signal attenuation in an open environment, acoustic-based solutions allow for accurate results, particularly in open environments.

There is also the possibility of using light to measure the distance between two nodes (Broxton et al., 2005). Still, it is not a common solution due to light propagation characteristics.

3. Types of Localization Algorithms

The work that has been done on sensor localization algorithms can be classified in two main categories: the range-based approaches and the range-free approaches. These approaches differ from each other essentially in the way distance information is obtained (Hsieh & Wang, 2006).

Range-based approaches are based in distance or angle measurements, requiring the installation of specific and expensive hardware (for e.g., directive antennas). Range-free approaches only consider connectivity information between adjacent nodes. They use special protocols to eliminate the need for radio signal measurements. Consequently, they require no extra hardware.

However, there are also hybrid solutions, which combine the advantages of range-based approaches with the advantages of range-free approaches (Wang & Xiao, 2006). Moreover, there are either range-based or range-free approaches combined with the use of anchor nodes, based in MDS (Multidimensional Scaling), centralized or distributed, or mobile-assisted. Yet, there are other possible classifications of localization algorithms (Wang et al., 2006; Reghelin & Fröhlich, 2006; Wang & Xiao, 2006; Medidi et al., 2006).

The appropriate localization technique heavily depends on the WSN intended application.

3.1 Range-based

In range-based approaches, distance estimation is usually performed using one of the following techniques (Mao, Fidan, and Anderson, 2007): RSSI, ToA, TDoA or AoA/DoA. Then, distance estimation is used to determine nodes position. Range-based approaches are also known as one-hop approaches (Hu & Evans, 2004), because nodes have to be at one-hop distance from a minimum of anchor nodes.

The RSSI (Received Signal Strength Indicator) technique is based in the attenuation of the radio signal with the increase of distance (according to $1/r^2$, where r is the distance between the sender and the receiver). Thus, the receiver only has to measure signal attenuation to estimate the distance to the sender. This technique is suitable to outdoor environments. However, it is not simple to predict radio signal behaviour indoors.

ToA (Time of Arrival) and TDoA (Time-Difference of Arrival) techniques measure the distance between nodes using signal propagation time. Using the ToA technique, nodes transmit a signal to their neighbours at a predefined speed and wait for answers. Their neighbours, in turn, send a signal back to them. Inter-node distance is computed by measuring the difference between sending and receiving times (round trip approach) (Mao et al., 2007). There is a one-way approach to determine ToA, but it is less used because it requires an accurate synchronization between the sender and receiver clocks. Synchronization adds cost and complexity to the WSN (Hu & Evans, 2004).

In TDoA method, inter-node distance measurements require that nodes transmit two different signals, which travel at different speeds, to their neighbours. Usually, distance is calculated based on the difference in propagation times of radio and acoustic signals originated at the same point (these signals propagate at well known but distinct speeds) (Kwon et al., 2005). Sender and receiver synchronize their clocks. Then, the sender broadcasts a radio message followed by an acoustic signal (chirp). Each node that detects the chirp, computes the difference between both signals arrival times and, consequently, the distance.

AoA (Angle of Arrival) measurement techniques, also known as DoA (Direction of Arrival), can be divided into two subclasses. Those techniques making use of the receiver antenna's amplitude and those making use of the receiver antenna's phase response (Hu & Evans, 2004). The accuracy of AoA measurements is limited by the directivity of the antenna, by the shadowing effect and by multipath reflections.

Typically, algorithms based on AoA and propagation time (ToA and TDoA) measurements can achieve better accuracy than algorithms based on RSSI measurements, because the amplitude of the radio signal is affected by

environmental factors (Broxton et al., 2005). However, this accuracy is achieved at the expense of higher cost equipment.

Most range-based algorithms are not suitable to low density networks. In this case, there will be connectivity failures due to higher distance between nodes, not permitting the computation of distance measurements (Wu, Sheng, and Zhang, 2007).

3.2 Range-free

This category of algorithms is based in connectivity information, i.e., “who is within the communication range of whom”, to estimate the location of nodes (Mao et al., 2007). The connectivity can be tested by measuring the amount of received packets. The principle of these algorithms is: if two nodes can communicate, then the distance between them is, with great probability, less than their maximum transmission range.

These techniques do not require extra hardware, because they do not rely in any distance measurements. The main advantages of range-free approaches are its simplicity and low cost. They are suitable to applications where location accuracy is less critical. However, the localization error is highly dependent on the density of nodes, on the number of anchor nodes and on the network topology (Wang & Xiao, 2006; Mao et al., 2007).

Range-free approaches can be hop-based, where inter-node distance is estimated through the number of hops between the nodes, considering the shortest path (Hsieh & Wang, 2006; Vicaire & Stankovic, 2004). So, they are also known as multihop approaches (Hu & Evans, 2004). Other group of solutions is based in the Euclidean distance, which is computed also considering the shortest path between two nodes (Wu et al., 2007; Medidi et al., 2006).

3.3 Use of anchor nodes and recursive solutions

Both range-based and range-free approaches may use some nodes with known location to support localization (Mao et al., 2007). These nodes are called anchor nodes (also referred to as beacon nodes or seeds) and are, typically, less resource limited than the remaining nodes of the WSN. In general, the remaining nodes

estimate their distance to anchor nodes and, then, perform some calculation (multilateration, triangulation, etc.) to compute their own position. It is worth mentioning that the main difference between these types of algorithms is the way that distances to anchor nodes are estimated.

Anchor locations can be configured manually during deployment phase or obtained by GPS (Global Positioning System). On one hand, it is not feasible to manually configure the position of all nodes due to the WSNs characteristics (high number of nodes, nodes thrown to the deployment area by a plane, etc.). On the other hand, the installation of a GPS receiver in each node is an expensive solution and with not enough accuracy (6 to 7 m). Also, in general, they cannot be used indoors, because of insufficient GPS coverage (Kwon et al., 2005). So, some of the algorithms that use anchor nodes are not appropriated to indoor environments, leading to the need of finding more proper solutions.

For all these reasons, there are typically few anchor nodes in a WSN. Most algorithms require a minimal number of anchor nodes, in an initial phase. Still, some algorithms can achieve good results not requiring prior knowledge of any anchor node position (Broxton et al., 2005; Wu et al., 2007). This is a clear advantage for all the reasons mentioned before.

To overcome the limitation of having insufficient anchor nodes as neighbours, some solutions include a recursive approach (Wang et al., 2006; Reghelin & Fröhlich, 2006; Ji & Zha, 2004; Wang & Xiao, 2006; Vicaire & Stankovic, 2004; Zhou et al., 2006; Taylor, 2005; Hu & Evans, 2004). These solutions repeatedly apply the basic algorithm by converting ordinary nodes to anchor nodes, after their positions are determined. This makes it possible for nodes faraway from starting anchors to locate themselves with the aid of converted anchors.

One problem of recursive approaches is that the localization error may accumulate, distorting the final result. However, in most solutions, nodes only become anchors if their position is computed with a certain confidence level (with high accuracy). This way, these approaches assure an error minimization, because only certain nodes can become anchors.

3.4 MDS-based solutions

Some algorithms (Ji & Zha, 2004; Wu et al., 2007; Medidi et al., 2006; Kwon et al., 2005), either range-based or range-free, are based in one of several MDS (Multidimensional Scaling) classes. MDS is a set of data analysis techniques that display the structure of distance data as a geometrical picture (Wu et al., 2007). Comparing to other solutions, MDS can achieve very accurate localization results (except for sparse networks), but at the expense of higher computational costs.

In general, each node applies MDS to compute a local map, including only neighbours' relative positions. Local maps are then merged together to form a global map. If enough anchor nodes are available (minimum of 3 for 2-D networks; minimum of 4 for 3-D networks), absolute positions can be computed, transforming the global map in an absolute map.

3.5 Centralized vs. distributed solutions

In centralized algorithms, data collected by all nodes in the network is sent to a central processing unit. So, the central processor collects all inter-node distance measurements, being responsible to compute the localization map of the entire network.

In distributed algorithms, the localization task relies on the nodes themselves (self-localization). Localization is performed based in inter-node distance measurements and in information collected from local neighbours.

Mao et al. (2007) compare centralized and distributed solutions from the perspective of location accuracy, energy consumption, implementational and computational issues. It is possible to conclude that both approaches have pros and cons. Any decision on what approach to apply must consider the WSN application and its accuracy and delay requirements. Other aspects, such as resource consumption, complexity, node density, should also be considered. For instance, considering a small network, a centralized solution can lead to low memory, processing and transmission costs. However, in high dimension networks, distributed solutions are, usually, more appropriated (Taylor, 2005).

3.6 Mobile-assisted localization

Node mobility can bring several advantages to WSNs. Mobile nodes can dynamically reposition themselves in order to satisfy certain requirements on monitor coverage, network connectivity or fault tolerance (Wu et al., 2007).

In mobile-assisted localization, nodes move with the help of robotic platforms. It is possible to control the movement of nodes in order to achieve better localization results. This group of solutions is suitable to obstructed environments, because nodes can move around obstacles. So, there is no need of line-of-sight between nodes and anchors (Wang et al., 2006; Wu et al., 2007; Hu & Evans, 2004).

Mobility can also be used in sparse networks to get more information about the network (mobility of nodes allows to temporarily increase the density of the network through virtual nodes) and, thus improve localization accuracy (Wu et al., 2007; Zhang, Huang, and Liu, 2006).

Ramanathan et al. (2002) use mobility of anchors to improve localization coverage. In this case, anchors can communicate with neighbours to know which nodes have not been localized yet. Accordingly, the mobile anchor decides the direction of its own movement.

Some solutions (Ramanathan et al., 2002; Zhou et al., 2006; Taylor, 2005; Hu & Evans, 2004; Marinakis & Dudek, 2006) incorporate probabilistic models of node movement, which improves localization by allowing the estimation of the current position of mobile nodes.

3.7 Cluster-based solutions

Some algorithms adopt hierarchical solutions to perform localization, usually through the creation of clusters (Sheng & Hu, 2003; Ramanathan et al., 2002; Ranjan, Kumar, Rammurthy, and Srinivas, 2005; Medidi et al., 2006). In general, the network is divided into two types of nodes, as shown in Fig. 4: cluster heads, which account for a small portion of the nodes in the whole network, and the remaining nodes, the majority. The use of clustering is motivated by the need for scalability and efficiency.

For example, Medidi et al. (2006) apply complex but accurate localization algorithms (MDS) to estimate the position of cluster heads. The remaining nodes

can apply a simpler localization process that uses cluster heads as reference nodes. This approach achieved reasonable accuracy, yet with low computational and communication overheads.

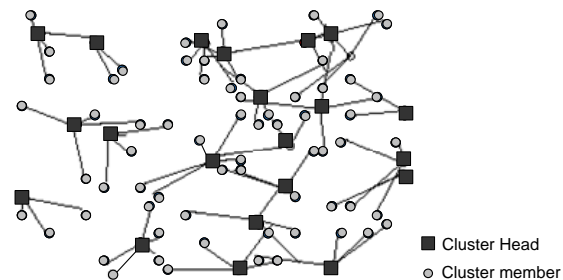


Fig. 4. Example of clustering.

4. Analysis of Localization Algorithms

Table I summarizes the main characteristics of some localization solutions, based in the aspects analyzed in sections 2 and 3. Algorithms Hu and Servetto (2005), Huang and Tseng (2005), Sheng and Hu (2003), Ramanathan et al. (2002), Hussain et al. (2004), Ranjan et al. (2005), Mao et al. (2007), Zhang et al. (2006), Marinakis and Dudek (2006) have not been considered in this study, either because they only verify a small amount of these aspects or because they refer to very specific problems (such as: target detection, routing, synchronization, coverage, etc.).

In this section, we illustrate the use of this comparative analysis to identify a more complete algorithm, able to adapt itself to a wider variety of possibilities (any network topology, any node density, any number of nodes, existence of obstacles and terrain irregularities, stationary or mobile nodes, etc.) and, yet, to achieve a good localization accuracy.

However, the importance of this comparative analysis relies in giving other authors the possibility of using this table to identify the localization algorithm which best suits their specific problem.

In our specific case, this comparison will allow us to find a solution for the localization problem to be applied in the context of the Foresmac project (Project INTERREG III B, 05/MAC/2.3/C16). This is an European project (that involves both Madeira and Canary Islands) based on the use of WSNs for forest environmental monitoring.

Regarding the selection of a more complete algorithm, we essentially compare this group of algorithms by considering the key aspects described in section 2, even though Table I also includes other characteristics.

The work related to the algorithms Broxton et al. (2005), AML (Bergamo et al., 2004) and DRL (Hsieh & Wang, 2006) does not make any assumptions of adaptation to network topology, obstacles and terrain irregularities. Thus, they are not considered a complete solution.

SMC algorithm (Hu & Evans, 2004) adapts to different network topologies. However, it is not clear how it behaves in the presence of obstacles and terrain irregularities. On the contrary, the algorithm proposed by Kwon et al. (2005) is not a complete solution for the opposite reasons. Despite being a very scalable solution (hierarchical and hybrid localization approach), the algorithm proposed by Zhou et al. (2006) has also not been chosen, since it is designed to underwater WSNs. CPL (Medidi et al., 2006) can achieve good results due to the cluster creation. However, it does not predict node mobility and, also, its behaviour in sparse networks have not been evaluated.

Ji and Zha (2004), i-Multihop (Wang & Xiao, 2006) and Virtual Ruler (Wang et al., 2006) localization algorithms are able to achieve very good results. They verify the great majority of requirements, except for node mobility. Thus, they haven't been chosen. LaSLAT (Taylor, 2005) has a very good accuracy, but only in non-concave topologies. Also, it just considers the existence of stationary nodes.

Taking into consideration all the key aspects mentioned before, we select MA-MDS-MAP(P) (Wu et al., 2007), HECOPS (Reghelin & Fröhlich, 2006) and ELA








(Vicaire & Stankovic, 2004) algorithms. These three localization algorithms have several advantages, because they adapt to a wide variety of circumstances. Furthermore, none of them require equipping sensor nodes with any extra hardware (either range-free or RSSI-based approaches). However, due to the great potentials of mobile-assisted localization and due to not requiring prior knowledge of anchor nodes position, we conclude that the MA-MDS-MAP(P) algorithm is the most complete solution.

Algorithm	Chen et al., 2005	Wang et al., 2004	(Kwon et al., 2005)	Wang, 2006	Wang & Evans, 2004	Chen et al., 2006	(Ji & Zhai, 2004)	Wu et al., 2007	Fröhlich, 2006	Wang, 2005	Wang & Xiao, 2006	(Wang et al., 2006)	Stankovic, 2004	al., 2006	
BASIC NETWORK ASSUMPTIONS	Resource consumption	-	md.	md.	low	low	md.	-	md.	md.	md.	-	-	low	md.
	Node density		-											md.	
	Number of nodes	-	-	low/md.	-	any	high	-	any	-	any	-	-	any	high
	Prior knowledge of anchor/beacon nodes position	✗	✓	✓	✓	✓	✗	✓	✗	✓	✗	✓	✓	✓	✓
	Node mobility	✗	✗	✗	✓	✓	✗	✗	✓	✓	✗	✗	✗	✓	✓
SIGNALS	Type of signals	US								US		US			

Algorithm	Characteristics	Indoor (ind.)/ Outdoor (out.)	Range-based/ Range-free	Centralized (cent.)/ Distributed (dist.)	Collaboration	Network topology adaptability	Obstacles and irregularities	Hierarchy (clusters, regions, etc.)	Mobile-assisted localization	Location coordinates
Chen et al., 2005	range-based (TDoA)	-	range-based (TDoA)	dist.	✓	-	-	×	×	□
Avila-Corajano et al., 2004	(DoA)	both	(DoA)	dist.	✓	-	-	✓	×	□
(Kwon et al., 2005)	(TDoA)	out.	(TDoA)	both	✓	-	✓	×	×	□
Wang, 2006	range-free	out.	range-free	dist.	✓	-	-	×	×	□
Stankovic & Evans, 2004	range-free	out.	range-free	dist.	✓	-	-	×	✓	□
Chen et al., 2006	range-free	out.	range-free	dist.	✓	-	✓	✓	×	□
(Ji & Zha, 2004)	range-based (any technique)	out.	range-based (any technique)	dist.	✓	-	✓	×	×	□
Wu et al., 2007	range-free	out.	range-free	both	✓	-	✓	×	✓	□
Fröhlich, 2006	range-based (RSSI)	out.	range-based (RSSI)	dist.	✓	-	✓	×	×	□
Labatut et al., 2005	range-based (TDoA)	ind.	range-based (TDoA)	both	×	-	✓	×	✓	□
Wang & Xiao, 2006	range-based (RSSI)	both	range-based (RSSI)	dist.	✓	-	✓	×	×	□
(Wang et al., 2006)	range-based (TDoA)	ind.	range-based (TDoA)	cent.	✓	-	✓	×	✓	□
Stankovic & Stankovic, 2004	range-free	out.	range-free	dist.	✓	-	✓	×	×	□
al., 2006	(any technique)	UW	(any technique)	dist.	✓	-	-	✓	×	□

Algorithm	Chen et al., 2005	Wang et al., 2004	(Kwon et al., 2005)	Wang, 2006	Wang & Evans, 2004	Wang et al., 2006	(Ji & Zha, 2004)	Wu et al., 2007	Fröhlich, 2006	Wang, 2005	Wang & Xiao, 2006	(Wang et al., 2006)	Stankovic, 2004	al., 2006
Recursive/ Iterative	x	x	x	x	✓	x	✓	x	✓	✓	✓	✓	✓	✓
Error Treatment	✓	✓	✓	x	✓	✓	✓	✓	✓	✓	✓	✓	x	x

Legend:

- : not available	md.: medium	US: Ultrasound	 : sparse network
 : Radio frequency	 : light	 : Acoustical	 : dense network
 : 2D	 : 3D	UW: Underwater	

Regarding the specific case of the Foresmac project, the localization solution must be suitable to the TMote-sky (Moteiv) motes and to forest environmental monitoring, which encompasses different requirements, more precisely:

- It should be appropriated to outdoor environments, because the project aims the forest monitoring;
- It has adapt to different network topologies;
- It has adapt to the existence of obstacles and terrain irregularities;
- It should not require any extra hardware (range-free or range-based: RSSI), for economical and practical reasons;
- Sensor nodes are stationary;

- And last but not least, it is critical to obtain 3D location coordinates, due to the terrain characteristics (rough ground) of both archipelags (Madeira and Canary islands).

By means of the same methodology used to select the most complete algorithm (analyzing Table I), we conclude that algorithms Ji and Zha (2004) and ELA (Vicaire & Stankovic, 2004) are the localization algorithms most suitable to this particular case of forest environmental monitoring. These are the only algorithms that verify all the requirements just mentioned.

5. Awareness Tool based on the WSNSCW Model

Awareness describes the knowledge acquired by a member of a collaborative session about the activities of the other members of this session. According to Gutwin and Greenberg (2002), the term "awareness" is used as a label designating various, more or less specified, practices through which cooperating actors, while engaged in their respective individual activities and dealing with their own local urgencies and troubles, manage to pick up what their colleagues are doing (or not doing) and to adjust their own activities.

As we mentioned before, opposing to traditional networks, WSNs are only useful if sensor nodes are aware of the environment surrounding them. This means that the great potential of WSNs lies in its ability to correlate collected data in time and in space (Broxton et al., 2005; Hu & Servetto, 2005). Besides, it is important that nodes are aware of the existence of other nodes and other entities (for e.g., sink nodes, anchor nodes, cluster heads, etc.), what are the relationships established between them, and what are their properties and their activities. These are some of the reasons why we are developing a web-based awareness tool for WSNs visualization.

This awareness tool will show all the entities of the WSN, their relationships and their properties. One of the most important properties of this awareness tool is the 3D representation of the network. This is very important so the user can have a more realistic view of the network, becoming more aware of the environment surrounding him (different types of terrains, different types of rooms, which obstacles might interfere with the collaboration established between nodes, etc.).

This awareness tool will allow for the visualization of different granularities: fine-grain (sensor nodes), middle-grain (clusters) and coarser (sessions) modelling levels. Also, it will allow for an interactive navigation in the map of the network.

Localization of wireless sensor nodes is obviously an essential component of this awareness tool.

We will apply this awareness tool to the specific case of a forest environmental monitoring, more precisely to the Foresmac project. In this context, one of the localization algorithms chosen in section 4 (Ji & Zha, 2004 and ELA (Vicaire & Stankovic, 2004)) is going to be implemented in our experimental WSN.



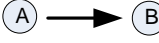
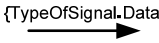





The implementation of this tool is being based in the WSN^{SCW} (Wireless Sensor Networks Supported Cooperative Work) model, published in Brito and Peralta (2008). The WSN^{SCW} model is a formal model of collaborative work created to the specific case of WSNs. It is, essentially, a graph-based model; but, it also includes other objects in order to make the modelling of all the entities of a WSN possible, which is fundamental to completely represent a WSN.

The network hierarchy (from the collected data to the user) can be represented, as well. Moreover, it is a generic model because it can be applied to heterogeneous networks (any type of nodes, any size, any hardware characteristics, any types of signals, etc.).

The entities defined in the model are all the components than might exist in a WSN. Table II shows the symbol, the concept and the description of all the entities included in the model.

TABLE II - ENTITIES OF THE WSN^{SCW} MODEL

Symbol	Concept	Description
○	Sensor node	Wireless sensor nodes, typically with limited resources. These nodes can be either stationary or mobile. Also, they can be in one of two possible states: active or inactive (sleep mode) in order to save energy.
⊙ _{S_k}	Sink node/ Base Station	Node to which data collected by ordinary nodes is sent; being responsible to send data to the gateway.
⊙ _A	Anchor node	Node with known localization.

	Cluster	Group of nodes, created according to: geographical area, type of sensor, type of phenomenon, task, etc.
	Cluster Head	Sensor node to whom all sensor nodes in the cluster send the collected data; it is responsible for sending the received data to the Sink node.
	Relationship	The arrow represents a relationship between nodes A and B. It also represents an adjacency relation between nodes A and B (see section 3.2); nodes A and B are neighbours. A relationship can be established based on: localization, phenomenon, type of sensor node, etc.
	Data flow	This label identifies both the type of signal being used (radio frequency, ultrasound, acoustical or light) and the type of data being transmitted between nodes (temperature, humidity, light, sound, video, internal voltage, etc.).
	Gateway	Device responsible to send the data to the user, through the Internet.
	Obstacle	An object (building, tree, rock, etc.) which obstructs line-of-sight between two or more nodes, not allowing for direct communication between them.
	Session	In a certain moment, there may be several collaborative sessions in a WSN. A session can be established based on the objective (type of phenomenon to monitor, geographical area to monitor, etc.) of the WSN.
	Battery	It represents the percentage of the sensor node's remaining battery.
	User	Person that interacts with the WSN, querying the network, visualizing data, etc. The user customizes the work of the sensor nodes; the data collected by sensor nodes is used by the users' application.

Next, we validate the WSNSCW model definitions, by applying it to the specific case of a forest environmental monitoring application. An example of a forest monitoring WSN can be represented as depicted in Fig. 5.

So, let's consider the example of a forest monitoring WSN. As shown in Fig. 5, in this example there are three simultaneous collaborative sessions. These sessions were initiated by the user, after he defined three different objectives: to monitor the temperature of area A1 (session CS1), to monitor the light of area A2 (session CS2), and to monitor the humidity of the same area (session CS3). So, nodes were deployed in an ad hoc manner, in two different geographical areas of a forest.

The WSN has a total of 30 nodes (CS2 and CS3 represent exactly the same area; hence, exactly the same nodes). Among these nodes, there are 2 sink nodes, 4 anchor nodes and 20 wireless sensor nodes. Within each area, clusters have been created according to the geographical localization of sensor nodes, being based in the proximity between nodes. The cluster head was chosen, among nodes in the cluster, as the node with more battery. In this case, 4 clusters have been created (2 clusters in Area A1 and 2 clusters in area A2); hence, there are 4 cluster heads. As this scenario relates to an environmental monitoring application, it is very important to be able to correlate collected data in space. So, anchor nodes had to be deployed in the WSN, because they can help the remaining nodes of the cluster in determining their own position.

The nodes in the cluster automatically start to collaborate to collect data and send it to the cluster head. Also, the cluster head starts sending data to the sink node, which, in turn, sends it to the user, through the gateway. Only the nodes of the cluster need to be in the active state, as they need to monitor the phenomenon and also need to send the data to the cluster head. The remaining nodes are in the sleep mode, in order to save energy. Nodes also become inactive if their batteries end.

As battery is the most critical resource of a sensor node, it is really important that the user knows the state of the battery of each node. In this way, the user gets to know when he has to go to the field in order to change the batteries of the sensor nodes.

Let's also consider that the user is far away from the forest being monitored. So, he monitors the WSN through the Internet (using the gateway).

Any changes that might occur on this scenario (new collaborative sessions, new clusters, nodes changing from sleep mode to the active state or vice versa, nodes moving, etc.) can be represented by a sequence of figures analogous to Fig. 5.

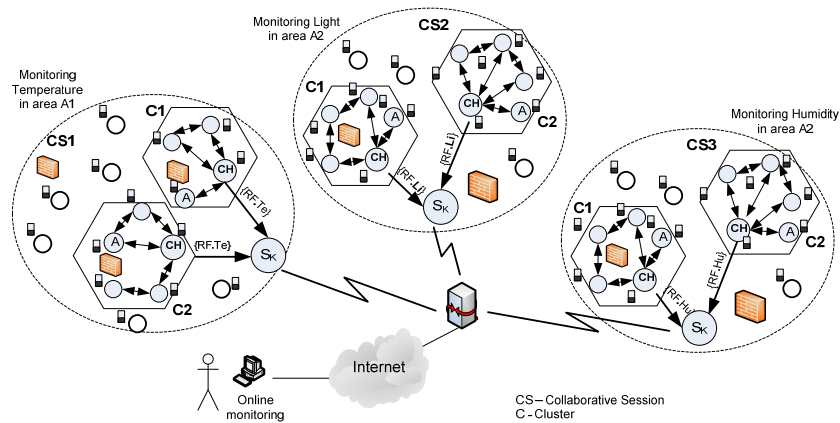


Fig. 5. . Applying the WSNSCW model to the specific case of a forest environmental monitoring.

Fig. 6 shows some pictures of the experimental WSN that was deployed in a forest in Madeira Island, Portugal, using TMote-sky nodes. Fig. 6a shows a sensor node located in a tree, whereas Fig. 6b gives an idea of the type of terrain typical in the Island (rough ground and mountains). In these particular experiments, a few nodes were manually deployed in the forest: therefore, their location is known. Consequently there is no need of implementing a localization algorithm. However, in future deployments, a larger number of sensor nodes should be deployed in an ad hoc manner (for e.g., dropped by an airplane). This is why it was so important to study localization algorithms appropriated to this application scenario.



Fig. 6. Experiments performed in a forest located in Madeira Island. In this case, nodes were manually deployed in the forest.

6. Conclusions and Future Work

Though WSNs are a recent area of research, there are already several localization solutions. In this paper, we analyzed the aspects that must be considered when designing or choosing a localization algorithm. Moreover, we presented a possible classification of localization algorithms.

The work presented in this paper allowed us to analyze and compare the more representative solutions. This comparison was based essentially in the following parameters: resource consumption; number and density of nodes; adaptability to network topology, and to the existence of obstacles and terrain irregularities; adaptability to node mobility, etc.

Among all studied solutions, this comparative analysis conducted us to the choice of one complete localization solution, in the sense that it adapts to a wide variety of conditions and, yet, allowing to achieve good localization results. It also allowed us to find a localization solution to be applied in the specific context of the Foresmac project, an European project based on the use of WSNs for forest environmental monitoring. However, the importance of this comparative analysis relies in giving other authors the possibility of using this study to identify the localization algorithm which best suits their specific problem.

Finally, we described the main characteristics of a 3D web-based awareness tool which we are developing. This tool is being based in the WSNSCW model, which main characteristics were also highlighted.

This awareness tool will be applied to the specific case of the Foresmac project. In this context, one of the localization algorithms chosen is going to be implemented in our experimental WSN.

References

- Akyildiz, I., Su, W., Sankarasubramaniam, Y., & Cayirci, E. (2002). A Survey on Sensor Networks. *IEEE Communications Magazine*, 38(4), 393-422.
- Bergamo, P., Asgari, S., Wang, H., Maniezzo, D., Yip, L., Hudson, R., Yao, K., & Estrin, D. (2004). Collaborative Sensor Networking Towards Real-Time Acoustical Beamforming in Free-Space and Limited Reverberance. *IEEE Transactions on Mobile Computing*, 3(3).
- Brito, L., & Peralta, L. (2008). A model for Wireless Sensor Networks Supported Cooperative Work. *3rd International Conference on Computer Graphics Theory and Applications (GRAPP 2008)*. Funchal, Madeira, Portugal. 505-511.
- Broxton, M., Lifton, J., & Paradiso, J. (2005). Localizing a Sensor Network via Collaborative Processing of Global Stimuli. In *Proc. of the 2nd European Workshop on Wireless Sensor Networks*, 321-332.
- Gutwin, C., & Greenberg, S. (2002). A descriptive framework of workspace awareness for real-time groupware. Computer Supported Cooperative Work. *The Journal of Collaborative Computing*, 11, 3-4.
- Hsieh, Y.-L., & Wang, K. (2006). Efficient Localization in Mobile Wireless Sensor Networks. In *Proc. of IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC'06)*, 1. 292-297.
- Hu, A., & Servetto, S. (2005). Algorithmic Aspects of the Time Synchronization Problem in Large-Scale Sensor Networks. *Mobile Networks and Applications*, 10. Springer Science + Business Media Inc. 491-503.
- Hu, L., & Evans D. (2004). Localization for Mobile Sensor Networks. In *Proc. of 10th International Conference on Mobile Computing and Networking (MobiCom 2004)*, Philadelphia. 45-47.
- Huang, C., & Tseng, Y. (2005). The Coverage Problem in Mobile Wireless Sensor Networks. *Mobile Networks and Applications*, 10. Springer Science + Business Media Inc. 519-528.
- Hussain, S., Farooq, U., Zia, K., & Akhlaq, M. (2004). An Extended Topology for Zone-Based Location Aware Dynamic Sensor Networks. In *Proc. of National Conference on Emerging Technologies (NCET 2004)*. Szabist Karachi, Pakistan.
- Ji, X., & Zha, H. (2004). Sensor Positioning in Wireless Ad-hoc Sensor Networks with Multidimensional Scaling. In *Proc. of IEEE INFOCOM*, 2652-2661.

- Kwon, Y., Mechtov, K., Sundresh, S., Kim, W., & Agha, G. (2005). Resilient Localization for Sensor Networks in Outdoor Environments. In *Proc. of 25th IEEE International Conference on Distributed Computing Systems (ICDCS)*, 643-652.
- Mao, G., Fidan, B., & Anderson, B. (2007). Wireless Sensor Networks Localization Techniques. *Computer Networks*, 51(10), 2529-2553.
- Marinakos, D., & Dudek, G. (2006). Probabilistic Self-Localization for Sensor Networks. In *Proc. of 21st National Conference on Artificial Intelligence and 8th Innovative Applications of Artificial Intelligence Conference, AAAI Press*. Boston, USA.
- Medidi, M., Slaaen, R., Zhou, Y., Mallery, C., & Medidi, S. (2006). Cluster-based Localization in Wireless Sensor Networks. In *Proc. of SPIE, Wireless Sensing and Processing, Raghuvveer M. Rao, Sohail A. Dianat, Michael D. Zoltowski, Editors, 6248* (62480J).
- Ramanathan, P., Saluja, k., & Hu, Y. (2002). Collaborative Sensor Signal Processing for Target Detection, Localization and Tracking. In *Proc. of the 23rd Army Science Conference*.
- Ranjan, G., Kumar, A., Rammurthy, G., & Srinivas, M. (2005). A Natural Disasters Management System Based on Location Aware Distributed Sensor Networks. In *2nd IEEE International Conference on Mobile Ad hoc and Sensor Systems (MASS 2005)*.
- Reghelin, R., & Fröhlich, A. (2006). A Decentralized Location System for Sensor Networks Using Cooperative Calibration and Heuristics. In *Proc. of 9th ACM international symposium on Modelling Analysis and Simulation of Wireless and Mobile Systems (MSWiM'06)*. Torremolinos, Spain. 139-146.
- Sheng, X., & Hu, Y. (2003). Collaborative Source Localization in Wireless Sensor Network System. *IEEE Globecom*.
- Taylor, C. (2005). Simultaneous Localization and Tracking in Wireless Ad-hoc Sensor Networks. Computer Science and Artificial Intelligence Laboratory Technical Report. MIT, Cambridge.
- Vicaire, P., & Stankovic, J. (2004). Elastic Localization. University of Virginia, Technical Report CS-2004-35.
- Wang, C., & Xiao, L. (2006). Locating Sensors in Concave Areas. In *25th Annual Conference of IEEE INFOCOM*. Barcelona, Spain.
- Wang, C., Ding, Y., & Xiao, L. (2006). Virtual Ruler: Mobile Beacon Based Distance Measurements for Indoor Sensor Localization. In *3rd IEEE International Conference on Mobile Ad hoc and Sensor Systems (MASS 2006)*. Vancouver, Canada.
- Wu, C.-H., Sheng, W., & Zhang, Y. (2007). Mobile Self-Localization using Multi-Dimensional Scaling in Robotic Sensor Networks. *IEEE International Conference on Robotics and Automation*.
- Zhang, Y., Huang, Q., & Liu, J. (2006). Sequential Localization Algorithm for Active Sensor Network Deployment. In *Proc. of 20th International Conference on Advanced Information Networking and Applications (AINA'06)*, 2, 171-178.
- Zhou, Z., Cui, J.-H., & Zhou S. (2006). Localization for Large-Scale Underwater Sensor Networks. UCONN CSE Technical Report: UbiNet-TR06-04