GUIDE: Guiding Users in Distributed Environments for WLAN and Ad Hoc Networks

Marco Gonzalez[1], Javier Gomez[2], Miguel Lopez-Guerrero[3], Jose E. Torres-Fernandez[4] and Victor Rangel[5]

[1, 2, 5] Department of Telecommunications Engineering
National Autonomous University of Mexico, UNAM
Edificio Valdez Vallejo, Tercer piso,
Circuito Exterior, Ciudad Universitaria, CP 04510
México D.F.

[3,4] Department of Electrical Engineering
Metropolitan Autonomous University, UAM - I
Av. San Rafael Atlixco 186
Col. Vicentina, CP 09340
México D.F.

[2, 5] {javierg,victor}@fi-b.unam.mx, [1]magonzalezsilva@yahoo.com.mx, [3]milo@xanum.uam.mx, [4]joseduardotorres@yahoo.com

Abstract

Whereas there is a lot of work related to finding the location of users in WLAN and ad hoc networks, guiding users in these networks remains an unexplored area of research. In this paper we introduce the concept of node-to-node guiding and present two methods that can be used to implement it. One method is based on distance estimation whereas the other relies on the computation of a gradient in the neighborhood of the moving node. We implemented all proposed guiding techniques thus creating GUIDE, which is a GPS-free and infrastructure-free node-to-node guiding system. From our results we observed that distance-based guidance is highly sensitive to distance estimation errors. On the other hand, the method based on a gradient computation proved to be more robust. In this paper we also discuss how GUIDE can be generalized to node-to-node guidance in multihop ad-hoc networks and directions for future research.

1. Introduction

The WLAN IEEE 802.11 (WiFi) standard is one of the great technology-success stories of the past decade. Used in research labs at the beginning, WLAN technology is now in a position where it has become as popular as cellular telephony. Thus, WLAN has become the preferred last-mile technology of the Internet. Off-the-shelf laptops, PDAs and even smart phones are now usually equipped with WiFi radios. At the same time, wireless metropolitan networks are beginning to be deployed, letting WLAN users to roam freely across large areas and remain connected to the Internet.

In this paper we consider that the wide availability of WiFi devices enables other possible applications. We introduce the concept of node-to-node guidance for WLAN equipped devices. Node-to-node guidance will allow Wi-Fi equipped nodes or users to obtain guiding means to get closer to other WiFi equipped devices. As it will be described below, this concept can be applied to both scenarios: WLAN and ad-hoc networks.

Guidance in general, and node-to-node guidance in particular, remain unexplored areas of research in WLAN-based networks. Nevertheless, we believe that there are several situations of potentially useful node-to-node guiding applications that are not available to WLAN users today. For instance, we can consider a user that is currently connected to an access point (AP) at 2 Mbps. Such a user can use a guiding system in order to move closer to the AP and achieve a connection at higher speed. In another example, a user printing a document over the air to a public WLAN-equipped printer can use a guiding system to get closer to the printer in order to pick up the print out. Rescue personnel can use a guiding system in order to get closer to a person, carrying a WiFi device, who has suffered an accident and needs assistance. Applications may be as simple as a situation where a WiFi user may just need to find another user. Furthermore, in case one of the nodes is not a user (e.g., robot or printer) the use of an automatic guiding system is mandatory.

A guiding scenario that requires special attention is the one corresponding to the situation when machines or users are located beyond the transmission range of each other. In this case a multihop routing protocol (i.e., a MANET routing protocol) can discover a route and provide the identities of intermediate nodes in the route. In such scenario node *A* may want to get closer to node *B*; but there will be one or more intermediate nodes along the route. In this case, node *A* can use a guiding system to get closer to the first intermediate user in a route towards the target node *B*, then get closer to the second intermediate node and so on until the target node is finally reached.

We believe that one reason why there has not been much interest of the research community in developing guiding systems for WLAN and ad-hoc networks is because developing such systems, from a pure research perspective, appears trivial once a good localization method is available. However, most localization systems need specialized infrastructure (e.g., GPS). This situation, in fact, significantly reduces their availability to the larger public since not all WiFi devices are currently equipped with positioning systems. Furthermore, in this paper we show that not all localization systems are suitable for use as the core of a guiding system.

We consider that a key characteristic of a guiding system to be widely accepted and used by wireless users is that it needs to be built on top of standard hardware. It should not require added hardware or infrastructure to work properly. At the same time, it should not require significant intervention of the target node and its implementation should not originate extra costs. In this paper we introduce GUIDE, a guiding system that is built around all these premises.

GUIDE is a GPS-free and infrastructure-free node-to-node guiding system for WLAN and ad-hoc networks. Opposite to other potential guiding systems that could be built on top of a positioning system, in GUIDE wireless nodes never know or need to know their location. It is based on real-time measurements of parameters related to the state of the wireless channel. Such readings come from a standard wireless card and are used in order to provide users with real-time instructions about the required changes in direction that need to be applied in order to get closer to the target node.

The rest of the paper is organized as follows. Section 2 presents an overview of different localization systems that could be used to develop node-to-node guiding systems. In Section 3 we describe different parameters that can be obtained from a standard wireless card in order to monitor the wireless channel state. In Section 4 we describe different proposals to design a user-to-user guiding system. In Section 5 we describe the implementation of the guiding system in a Linux box. In Section 6 we describe the experiments that were conducted and the obtained results. In Section 7 we describe how this work can be generalized to the multihop case. Finally, in Section 8, we present our conclusions and ideas for future research.

2. Related Work

Beyond GPS-based guidance, which is discussed later in this section, there are not too many examples of guiding systems for WLAN and ad-hoc networks in the literature. However, there are various localization systems that could be used as the core of a guiding system. We now review some of these systems.

Angle of Arrival (AoA) is a technique in which special receivers can measure the angle on which the signal is picked up from a specific transmitter. These measurements typically take place at the base station where arrays of directional antennas can determine the angle of arrival. Use of AoA techniques in WLAN was first introduced in [8]. They also showed that AoA has better localization precision than systems based on distance estimation. The proposal in [8] requires a modified access point (AP) in order to perform localization using triangulation techniques. As we mentioned before, we are considering a user-to-user guiding system that relies on off-the-shelf WLAN hardware only, so we discard the AoA technique.

Time of arrival (ToA) [3] and Time Difference of Arrival (TDoA) [14] estimate distances by measuring the propagation time of a radio wave travelling from the transmitter to the receiver. In these techniques the transmitter takes into account the transmission time in the packet header so the receiver can compute the propagation time and estimate the distance to the transmitter. This technique can be extended to more than one receiver so that a triangulation algorithm can be used to estimate the approximate location of the receiver. The techniques ToA and TDoA need tight clock synchronization for accurate distance estimation. The fact that propagation times in WLAN are in the order of microseconds makes it impossible to accurately estimate distances using off-the-shelf WiFi hardware (e.g., a one-microsecond discrepancy represents an error of about 300 meters).

Localization systems based on signal attenuation are based on measurements of received signal strength. These data are used in combination with the known transmitted power and a propagation model to estimate the distance between the transmitter and the receiver. Similar to ToA and TDoA techniques, measurements of signal strength from various receivers and triangulation techniques can be used to reduce the area where the transmitter could be located. The accuracy of these techniques depends on the propagation model being used. A severe drawback of those localization systems is that the signal strength at the receiver is affected by several factors difficult to incorporate in a propagation model. Such factors can be, for instance, diverse obstructions in the line of sight between transmitter and receiver. In Section 4 we describe a user-to-user guiding system based on signal strength measurements.

WiFi mapping is a practical technique to reduce the error of localization systems based on signal strength. This approach takes into account the particular propagation characteristics of each location in the network. In this category we find the RADAR system proposed in [1], which is a localization system for infrastructure-based WiFi networks. In RADAR, there are several access points (APs) in the network and there is an empirical propagation map created beforehand for each AP in the network. Every time a node wants to estimate its location, it measures the values of received signal strength with respect to each AP in range. Then it compares the collected values with the WiFi maps. Finally, it uses a searching algorithm to find the location in the network that minimizes the error with respect to all APs simultaneously. Node-to-node guidance in RADAR-like scenarios has not been explored by researchers so far; however it is not as easy as it appears. Beyond getting the location of the two nodes in the network (i.e., the moving node and the target node), the guiding system should command the moving node to move in the direction that minimizes the distance to the target node and then monitor and adjust the direction in real time. RADAR needs infrastructure to work properly (i.e., various APs), and we already mentioned that for our purposes a successful guidance should work even if only two users are involved in the network (i.e., the moving node and the target node). Similarly, WiFi maps are precise as long as the node they refer to are static (i.e., APs in RADAR). In a node-to-node guiding system the target node may be moving also (e.g., a user or a robot). In a later section we show that an RSSI map drastically changes if a node moves even a few meters.

The global positioning system (GPS) [9] may appear as the most appropriate candidate to implement localization and guiding systems. The GPS system is based on TDoA techniques using various satellites in order to estimate the location of a GPS ground device. Although GPS may be used to implement a highly accurate localization system, it has several drawbacks. First, it needs specialized infrastructure. This fact alone disqualifies GPS for our purposes. Second, knowledge about the current position is not enough to determine which way to move. For a GPS node to figure out which way to go, it needs a few movements before it can establish the direction of its trajectory. In addition, there are other reasons why GPS may not be a good candidate for node-to-node guidance. First, GPS needs to have clear line of sight with three or more GPS satellites. Therefore, it is mainly applicable to open areas. Second, power consumption in GPS devices is another concern. Although it varies according to the manufacturer, a GPS device may continuously drain 50-200 mW from the battery. Finally, GPS remains an add-on device that needs to be bought separately, thus reducing deployment possibilities and increasing costs.

We observe that guiding solutions based on GPS are not a practical alternative in the context of the node-to-node guiding system that we envision because of various factors including cost and line-of-sight requirements. Other techniques like AoA and WiFi mappings could provide good localization, but they require either special hardware or special infrastructure to work properly. Similarly, ToA and TDoA techniques cannot be considered for guiding purposes given the time scale of propagation times in WLAN networks. We conclude that in order to make a localization-and-guiding system available to the larger public, it is necessary to use WiFi radios which are already available in many commonly used devices.

3. Standard 802.11 PHY layer information

In this section we describe different parameters related to the status of the wireless channel that can be taken into account in order to implement a node-to-node guiding system. Since we are only interested on using standard WiFi hardware, we review the parameters whose measurements can be obtained from a standard 802.11 wireless card.

Signal strength

The energy level observed during the last protocol data unit (PDU) reception is reported by means of a parameter known as received signal strength indicator (RSSI). In the 802.11 standard the only restriction on the RSSI values is that there must be a minimum number of levels ranging from 0 to RSSI_Max. This laxity has a number of implications. First, although RSSI is usually a one-byte long parameter (i.e., its value could be somewhere between 0 and 255), chip vendors can choose a convenient value for RSSI_Max and in practice the full range is not used. Second, chip manufacturers can also choose an appropriate range of signal strength that will be mapped to the set of RSSI values. Third, the quantization step can also be conveniently chosen and it does not need to be constant along the whole range of RSSI values.

As a consequence of the freedom provided by the standard, RSSI readings coming from different chipsets cannot be compared. However, the intended use of this parameter does not need a specific correspondence between its values and the levels of signal strength. RSSI values do not need to be of fine granularity or high precision either. This is due to the fact that the parameter is used in a relative manner, which is enough to carry out the intended tasks. For instance, one vendor may choose an appropriate RSSI value as a threshold in order to determine whether the channel is clear or not.

One more issue worth mentioning is that wireless cards are not usually able to measure a signal strength above 1mW [2]. The rationale behind this design is that although transmit power can be dozens of times higher, this level of signal strength is good enough to provide connectivity at the highest possible data rate. This means that once the card detects 1 [mW] or higher values, it will show 100% signal strength regardless of how stronger the actual signal is. This situation happens within a couple of meters away from the RF source.

Signal quality

Signal quality (SQ) is another metric mentioned in the 802.11 (1999) standard. A precise definition of the term is not provided; but it is specified that SQ is related to the DSSS PN code correlation and its value is updated each time a code lock is achieved. Lack of further specifications means that specific implementation details are likely to differ among different chip vendors. Although the specification given by the standard only applies to the DSSS modulation scheme, wireless cards also report readings of signal quality when they transmit using a different modulation technique. Since implementation details are proprietary information, we can only speculate that those values are related to the average correlation between the transmitted and received symbols. In any case, the lack of precise definitions is not a serious issue since the SQ readings are also used in a relative manner.

Data rate

The 802.11 standard makes use of adaptive modulation in order to take into account current channel conditions in the transmission process. In this way, high data rates can be achieved at short distances where the signal is strong enough, whereas more robust but-lower-rate transmission schemes are used for long distances. This relation between data rate and distance can help to estimate the relative position of a wireless station. However, each vendor is likely to use different algorithms for rate control so that the association between data rate and distance cannot be generalized for all chipsets.

4. Guiding algorithms for WLAN users

Before getting into the details of the proposed guiding algorithm, we will list the desired characteristics of the user-to-user guiding system that we envision for WLAN and ad hoc networks.

- It should operate with off-the-shelf WiFi hardware. As mentioned before, in order to make user-to-user guiding functions available to the larger public we need a system that does not need extra hardware to avoid extra costs. WiFi is already incorporated in all laptops, most of PDAs and many cellular phones.
- It should minimize the effort needed to reach the target node. By this we mean that the time and effort spent in closing the distance with the target node should be minimized. Ideally we would like the system to guide users on a rectilinear trajectory pointing directly to the target node.
- It should work everywhere. We would like the system to operate everywhere in a distributed fashion. The use of a centralized system cannot be considered because this would limit its fault tolerance and scalability.
- It should work with as few as two nodes. We want the guiding system to work even if the moving and target nodes are the only nodes in the network. With this constraint we disqualify any system requiring three or more nodes.
- It should require minimum intervention of the target node. It is desirable that the target user remains in a passive state during the guiding process. Also, there should not be a need to run any extra piece of software at the target node.

4.1 A distance-based guiding algorithm

Assuming that nodes can reliably estimate distances to other nodes within their transmission range, some geometric arguments can be used in order to determine the direction of the movement that would close the distance to the target node. Regarding Fig. 1, a simple algorithm to achieve this purpose is described below (all angles are measured in degrees).

Step 1. *Estimate the distance to the target node at the starting point (distance a in Fig. 1)*
Step 2. *Select a direction of movement at random, then walk c meters in that direction and stop. Estimate again the distance to the target node (distance b in Fig. 1).*
Step 3. *Compute the direction of the next movement. This direction is given by angle* β_{S3} *(see Fig. 1) which can be computed at this point from the well-known law of cosines. The resulting formula turns out to be*

$$\beta_{S3} = 180^\circ - \cos^{-1}\left(\frac{c^2 + b^2 - a^2}{2cb}\right) \qquad (1)$$

As it can be seen in Fig. 1, angle β_{S3} is measured with respect to the trajectory selected in step 2. However, there are two scenarios that would satisfy the spatial relations so far deduced. The direction to the target node can be counterclockwise or clockwise as illustrated in Fig. 2. We call this situation the mirror point problem. At this point in the algorithm the moving node has no way of knowing which one of the two points corresponds to the real location of the target node, and which one is the mirror point. Thus, the node will have to randomly choose whether to turn β_{S3} counterclockwise or clockwise.

Step 4. *Walk c' meters along the selected trajectory in step 3 and stop. Estimate again the distance d to the target node. If the distance to the target was shortened by c' meters, the direction of movement was correctly chosen in the previous step. In this case the movement can continue until the target is reached. Otherwise, if the direction chosen in step 3 was incorrect, the trajectory must be adjusted. In this case the triangle formed (b-c'-d) can be used to compute the new direction of movement to the target. This is given by angle β_{S4} which is computed in a similar way as indicated by (1). This direction of movement has to take into consideration the following two criteria.*

> ***Case a.*** *If $\beta_{S3} > 90^o$, then β_{S4} must be chosen to turn the same way (i.e., clockwise or counterclockwise) as previously done in step 3 (see Figure 3a).*
>
> ***Case b.*** *If $\beta_{S3} < 90^o$, then β_{S4} must be chosen to turn in the opposite way with respect to the one taken in step 3 (see Figure 3b).*

Ideally, after turning β_{S4} degrees, the node should now be moving on a trajectory pointing directly to the target node. In practice, and due to distance estimation errors, the node will continue adjusting its trajectory as it gets closer to the target node. Note that in describing this algorithm we omitted the case when the first movement in step 2 goes beyond the transmission range of the target node.

Implementation with distance estimation based on signal strength

There might be many ways to estimate the distance between two wireless nodes. We explored one technique that combines measurements of signal strength with a propagation model, which is also the most common distance estimation technique used by researchers [5, 7, 12, 15]. In this context, we considered a

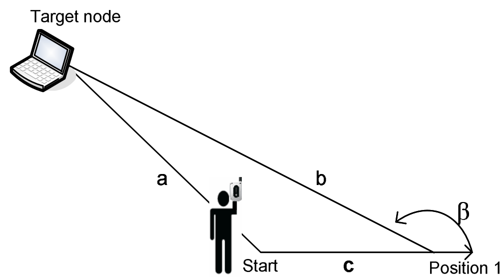


Fig.1. Distance based guiding

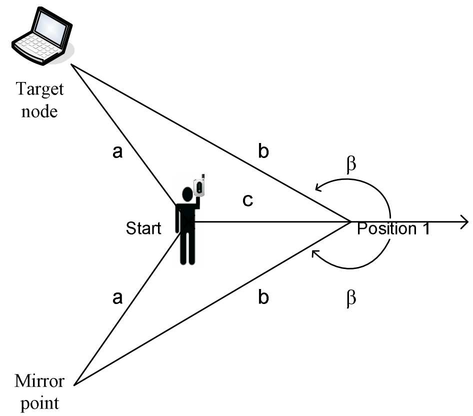


Fig. 2. Mirror point problem

commonly used propagation model for WLAN technology, the two-ray ground reflection model [11] given by

$$P_r(d) = \frac{P_t G_t G_r h_t^2 h_r^2}{d^4 L} \quad (2)$$

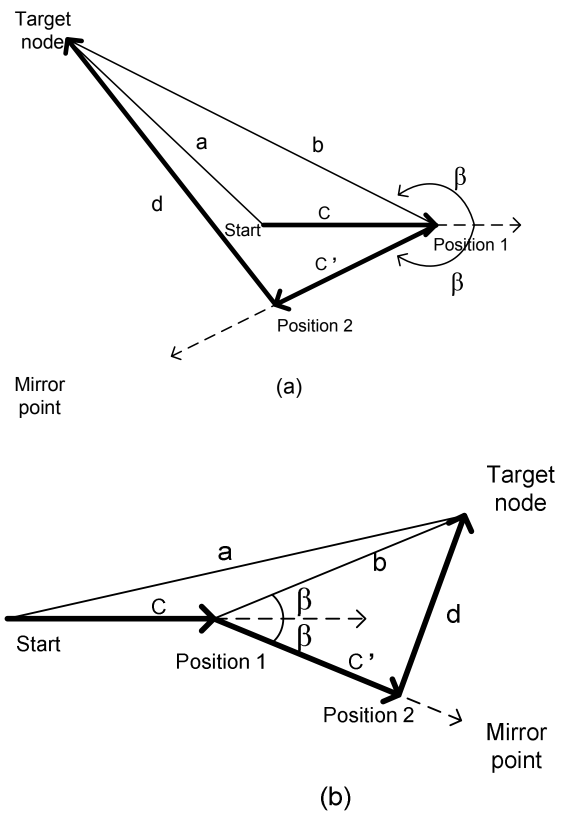


Fig.3. Mirror point cases (a) $\beta_{S3} > 90º$ y (b) $\beta_{S3} < 90º$.

where G_t, G_r, h_t and h_r are the antenna gains and antenna heights of the transmitter and the receiver, respectively. We assume omnidirectional antennas so that G_t and G_r can be considered equal to 1. Parameter L is a proportionality constant and P_t is the transmission power. With these elements, Eq. (2) can be solved for d so that if measurements of received power are available, the propagation model can be used to estimate the distance between the two antennas. For our purposes, the measurements of received power can be obtained from a standard 802.11 wireless NIC as described in Section 3. One point worth mentioning is that because of the fading nature of the signal in the wireless transmission medium, several measurements have to be collected and processed in the same location in order to obtain a reliable value of P_r to be used in (2).

We implemented a guiding system based on the described algorithm using a Linux box equipped with IEEE 802.11g. The corresponding implementation details and experimental results are reported in Section 5.

4.2 Gradient-based guidance

From Eq. (1) it is tempting to assume that signal strength can be modeled using a smooth-and-monotonically-decreasing surface with only one maximum located at the position of the target node. At least this model should reasonably hold in open spaces where there are no obstructions that could produce significant signal reflections. In such an ideal scenario, a local gradient computed from the collected samples should guide us in a rectilinear trajectory towards the target node. This would save us from going around in circles wasting time and taking redundant measurements before figuring out which way to go.

There might be many practical ways to compute a local gradient. We used a gradient-like computation that takes only into consideration the last three noncollinear measurements. Such trajectory creates a virtual triangle where the differences in the measurements across its three vertexes allow us to compute an approximation of the local gradient. In an ideal measurement case, the resulting gradient should point directly to the target node. This method is illustrated in Fig. 4 where the shown surface serves for illustration purposes only.

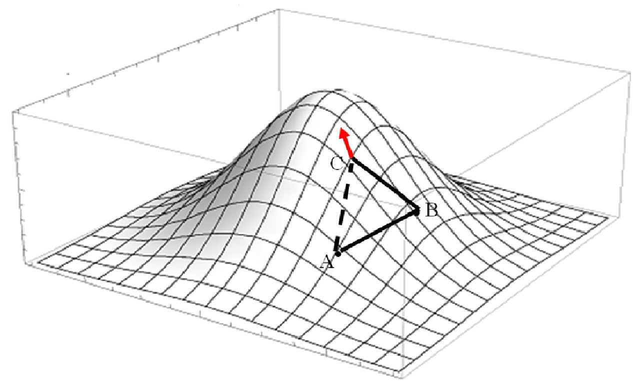


Fig. 4. Gradient approximation using the last three measurements

A gradient-based guiding algorithm

In this section we describe a simple guiding method that does not require any localization means to operate. It can employ either link quality or data rate measurements in order to do the guidance. In our description we use the term *metric* to represent a proximity criterion (i.e., link quality or data rate) or and all angles are measured with respect to the direction of the last movement. The proposed algorithm is described below and we illustrate the guiding instructions in Fig. 5.

Inicialization.

The algorithm starts by creating a virtual triangle as follows. Measure the metric (M_A) at the start location. Select a direction at random, move c meters in that direction and measure the metric again (M_B). At this point select a default turning direction to be used in the rest of the algorithm. It can be clockwise (CW) or counterclockwise (CCW). Turn 120º in the default direction. Move c meters along the selected direction and measure the metric again (M_C). At this point the triangle has been created and the measurements can be associated to its vertexes (in fact, as shown in Fig. 6 two different triangles can be created). With these data, the direction for the next movement can be determined from the rules given in Tables 1(a) and 1(b). For instance, if the differences (M_B-M_A), (M_C-M_B) and (M_C-M_A) are all positive (+), the user will be commanded to turn 60º clockwise (CW). The rules shown in Table 1 try to approximate a local gradient as previously explained. Turn in the direction indicated by the table.

Loop A.

Move c meters
$M_C \leftarrow M_B$
$M_B \leftarrow M_A$
Measure M_A
If ($M_A > M_B$) *then {*
No trajectory change is needed
Flag= False
}
If ($M_A == M_B$) *then {*
Randomly turn 45º clockwise or counterclockwise.
Flag=False.
}
If ($M_A < M_B$) *then {*
Turn 120º in the default direction.
Flag=True
}
If(Flag==True) {
Move c meters
$M_C \leftarrow M_B$
$M_B \leftarrow M_A$
Measure M_A
Compute the differences (M_B-M_A), (M_C-M_B) and (M_C-M_A) and use these results to determine the turning direction from the rules given in Table 1. Turn in the direction indicated by the table.
Flag=False
}

***Go back** to Loop A*

The operation of this algorithm is based on the gradient solution introduced before. If after moving *c* meters the metric gets better, the node continues moving on the same trajectory. When the metric remains the same, we ask the node to continue moving forward with a random change of trajectory

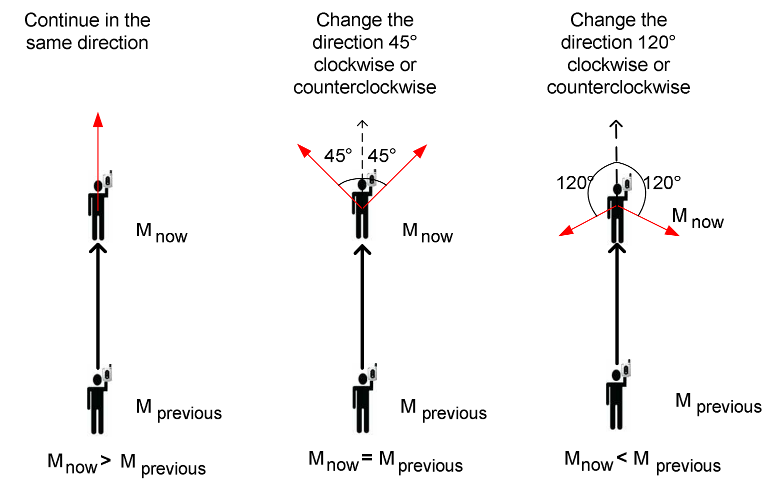


Fig. 5. Trajectory decisions the GUIDE system. The variables M_{now} and $M_{previous}$ stand for the current and the previous measurements respectively.

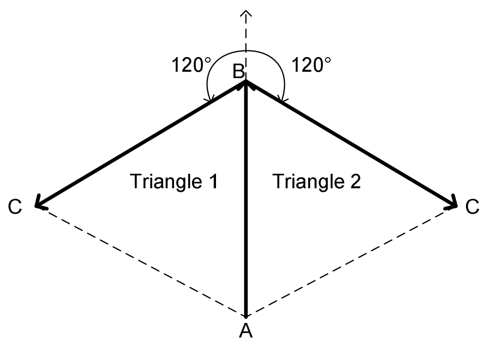


Fig. 6. Virtual triangles created with three noncollinear measurements

of 45º clockwise or counterclockwise. This is an attempt to collect new information in the neighborhood of the current location. When the metric gets worse, the node is ordered to change its trajectory 120º clockwise or counterclockwise. We decided not to order the node to move on an opposite trajectory because that will bring the node back to its previous position (and therefore no new information would be obtained). We preferred to move the node to a point located at 120º with respect to the direction of last movement so that after walking *c* meters, an equilateral triangle could be created with the last three positions.

In early experiments we realized that it did not make sense to determine the gradient with a high degree of accuracy since a human would not be able to follow indications given at a fine resolution. Therefore, we explored alternative ways to approximate this computation. In this context the method described in Table 1 yielded good results. It is worth mentioning that there are many different algorithms that we could have used to formulate the guiding procedure.

5. Implementation of the GUIDE system

For the implementation and testing of the GUIDE system (Guiding Users in Distributed Environments) we used a Toshiba Tecra A5 laptop running Ubuntu (kernel 2.6.15-23-386) with an Intel PRO/Wireless 2200BG 802.11g wireless card. We used Java to implement the GUIDE application.

When the program starts, the user has to select the algorithm to be used (based on either distance estimation or the gradient-like computation). The user must also indicate his or her approximate walking speed so that the program can approximately determine the time it takes to reach an intended position.

GUIDE uses the Linux Wireless Extensions and Wireless Tools [13] to interact with the device driver and retrieve low level information about the connection (i.e., data rate, signal strength (RSSI) and signal quality). The laptop takes several measurements of the same metric in order to reduce the variance by averaging several measurements at the same point.

Guiding indications are provided to the user by means of graphical indications and prerecorded voice commands. We observed that, for the average

Table 1(a). Action at point C (triangle 1 of Fig. 6)*

M_B - M_A	M_C – M_B	M_C - M_A	Action
+	+	+	60° CW
+	+	-	120° CCW
+	-	+	150° CW
+	-	-	150° CCW
-	+	+	30° CCW
-	+	-	80° CCW
-	-	+	120° CCW
-	-	-	120° CCW

Table 1(b). Action at point C (triangle 2 of Fig. 6)*

M_B - M_A	M_C – M_B	M_C - M_A	Action*
+	+	+	60° CCW
+	+	-	120° CW
+	-	+	150° CCW
+	-	-	150° CW
-	+	+	30° CW
-	+	-	80° CW
-	-	+	120° CW
-	-	-	120° CW

*Notation: M_A, M_B and M_C denote the value of the metric measured at points A, B and C respectively; CW and CCW stand for clockwise and counterclockwise respectively.**

user, the indications provided by the system were easier to follow if we used the familiar image of a clock instead of a scale in degrees. In Fig. 7 we provide a snapshot of the graphical user interface. The interface shows the path followed by the user in real time over an area representing slightly more than 4,000 m^2.

6. Experiments and results

In this section we present some guiding experiments with different variants of GUIDE. The experiments that follow took place in the two locations described below:

a) ***Obstacle-free location.*** This location corresponds to a place where there is a large open area with minor terrain elevations and sporadic vegetation.
b) ***Location with obstacles:*** This location corresponds to an area located inside the National Autonomous University of Mexico main campus. This is an example of typical school premises for outdoor study and leisure activities. There are tennis courts, *frontons* (with thick 18-meter high walls) and large and dense trees.

In all guiding experiments the moving node was initially located at the center of the figure at a position labeled as "Start" and the target node was placed a hundred meters away in an arbitrary direction (labeled as "TargetNode" in the figures). The first movement is supposed to take a random trajectory, but for comparison purposes we considered that it was better if it was always in the same direction. All movements were 20-meter long (value of *c* in Fig. 1).

In the following sections we provide some representative results. In each figure we show the full path followed by the moving node along with the value of the metric measured at each point. In these experiments the user stopped moving when he or she reached the target. It is worth mentioning that the criterion to determine that the target node has been reached depends on the particular application. The algorithm was also stopped if the target node was not reached after an excessive number of movements.

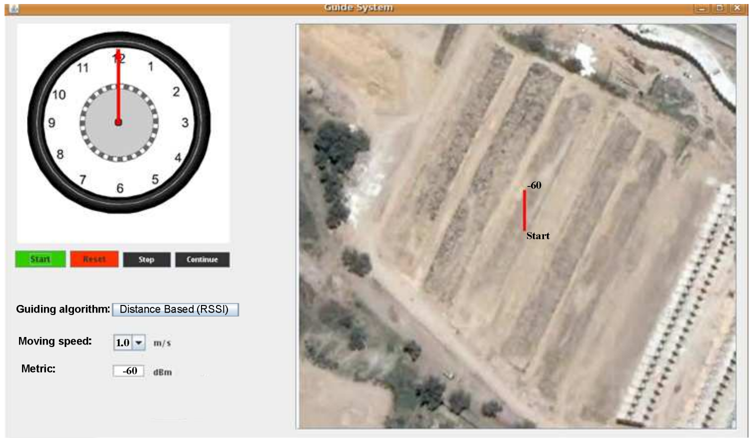
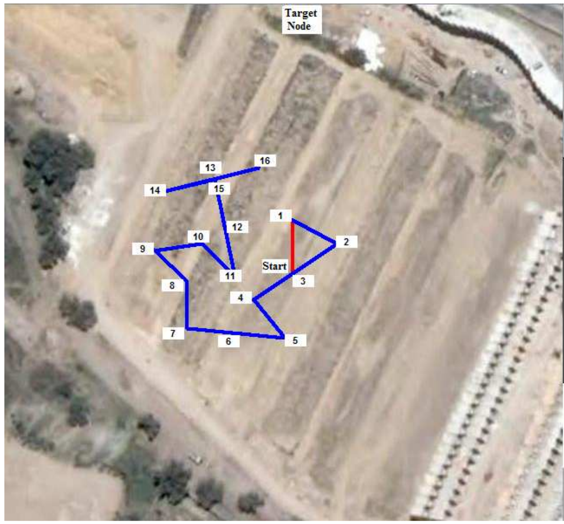


Fig. 7. GUIDE´s Graphical User Interface

6.1 Experiments with distance-based guidance

Figure 8 shows the trajectory followed by the moving node using the previously described guiding system based on distance estimation. As we can see from the figure the moving node gets contrasting and confusing signal strength (RSSI) values. Sometimes it gets closer to the target node in spite of getting lower RSSI



Points	Mesured RSSI (dBm)
Start	-67
1	-68
2	-70
3	-65
4	-64
5	-67
6	-63
7	-62
8	-59
9	-61
10	-58
11	-61
12	-57
13	-56
14	-63
15	-59

Fig. 8. Example of distance-based guiding

values, some other times it moves away from the target node in spite of getting higher RSSI values. In both cases the guiding algorithm proposed in Section 4.1 causes the node to estimate wrong distances and compute movements leading the moving node nowhere near the location of the target node. We found a similar trend in all experiments performed using RSSI values. We show one representative example only.

An important factor that creates variations in the received RSSI values is related to propagation effects in wireless channels. It is well known that wireless transmissions are affected by various phenomena including doppler shift, slow and multi-path fading effects, among others. These effects result in stochastic variations of the received signal power which are quite difficult to predict. Figure 9 shows an example of how the RSSI values change as a moving node moves behind trees (a slow fading example). Clearly unless the propagation model is aware of such obstructions, the distance estimation would be quite wrong, leading to poor guiding performance. It remains as an open area of research area to find out better methods to estimate distances that are immune to propagation effects.

6.2. Experiments with gradient-based guiding

Guidance based on differences of data rate

Figure 10 shows a representative example of the guiding system performance using data rate as the metric. As we can see in this figure, the guidance is quite disappointing bringing the moving node nowhere close to the target node. We observed a similar trend in all guiding experiments we performed using data rate.

In order to explain these results we collected data regarding the data rate achieved in the coverage area of an IEEE 802.11g network in an outdoor setup. These measurements are shown in Figure 11. From this figure we can observe some of the reasons of the poor guiding accuracy observed in the previous figure. For instance, some regions of the same data rate are too wide, leading to poor guiding performance. As a consequence of this, a node roaming in the 24 Mbps region can potentially walk a very long distance before it gets a different data date (i.e., either 18 or 36 Mbps). In case the node moves to a point within the 18 Mbps region, the guiding algorithm should bring the node quickly back to the 22 Mbps region, but again because the 22 Mbps is so large, the node will spend a long time in that region before a change in data rate is observed again. Similarly, once the node reaches the 54 Mbps region, the node is basically blind and has no other guiding means to get closer to the target node. This is particularly disappointing if we consider that the 54 Mbps region could have a large radius in outdoor scenarios.

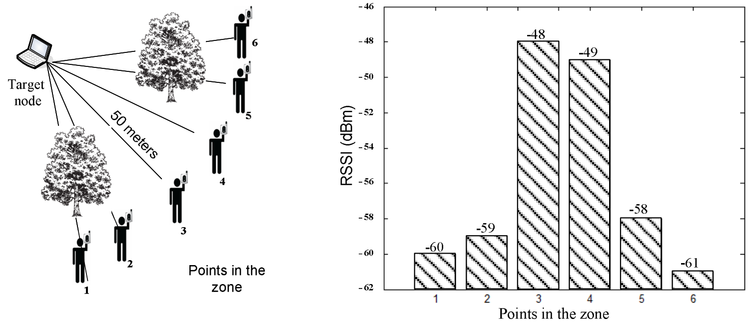
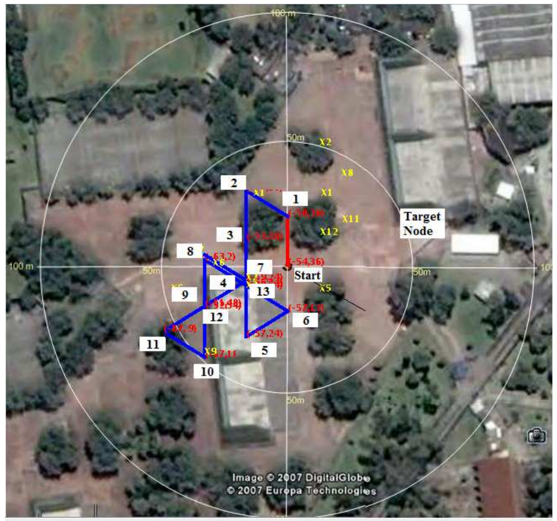


Fig. 9. Effect of obstructions on RSSI measurements



Points	Measured data rate (Mbps)
Start	36
1	36
2	18
3	36
4	54
5	24
6	12
7	24
8	2
9	48
10	18
11	9
12	54
13	36

Fig. 10. Gradient-based guidance with data rate

Guidance based on differences of link quality

As we will see now, the algorithm described in Section 4.2 with guidance based on link quality measurements is the only guiding method that fulfills the goal of guiding the moving node towards the target node. In order to illustrate its performance we show seven representative results. Three of them took place at the location with obstacles and four of them correspond to the obstacle-free scenario.

Figures 12, 13 and 14 show the performance of the algorithm in the location with obstacles. It can be observed that even though there are several guiding impairments, after a few movements the guiding system was able to find the target node. For illustrative purposes we comment on trajectory decisions taken by GUIDE in the arbitrary case shown in Fig. 13.

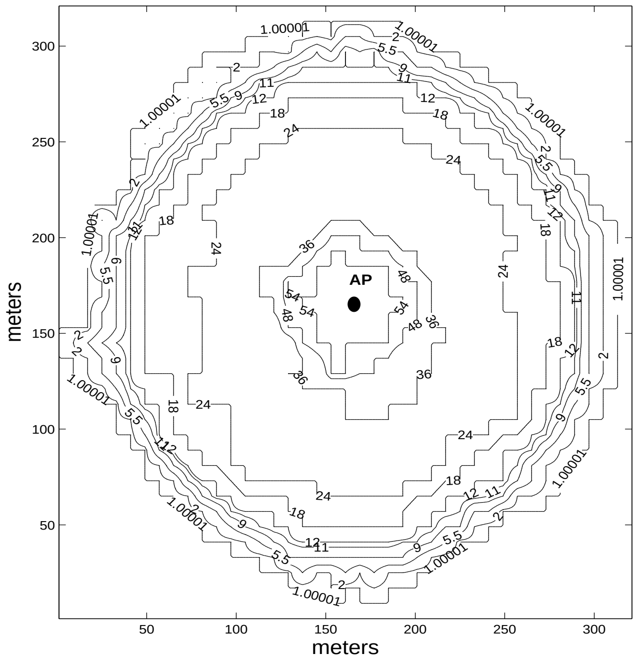
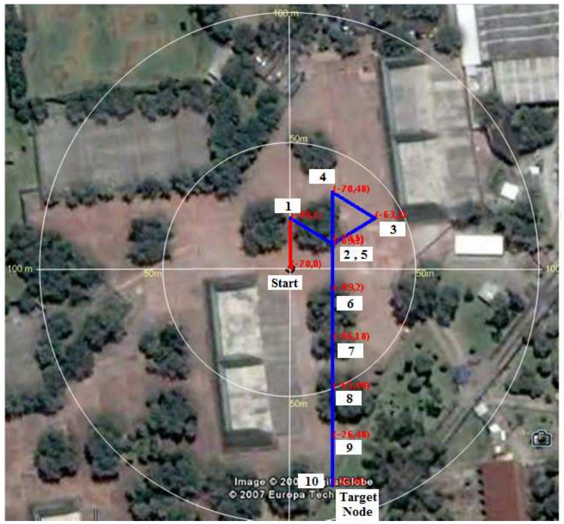


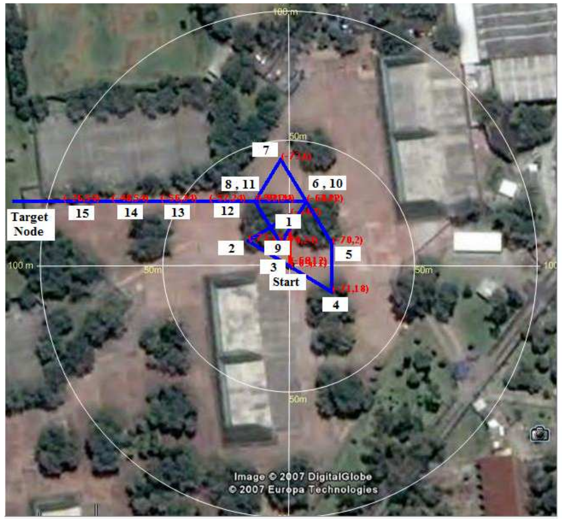
Fig. 11 Experimental propagation map

- Start. The node measures a link quality value of 61/100 and (randomly) moves to a direction that corresponds to the top of the figure.
- Point 1. The node measures a link quality value of 60/100 getting worse than the one observed at point 0. Therefore, the node turns 120° counterclockwise (CCW).
- Point 2. The node measures a link quality value of 45/100. Points 0, 1 and 2 form an equilateral triangle and the node computes the differences in the metric measured at the three vertexes. According to the rules shown in Table 1, the node turns 120° CCW.
- Point 3. The node measures a link quality value of 52/100 getting better than the one observed at point 2. Therefore, the node continues moving on the same trajectory.
- Point 4. The node measures a link quality value of 41/100 getting worse than the one observed at point 3. Therefore, the node turns 120° CCW.



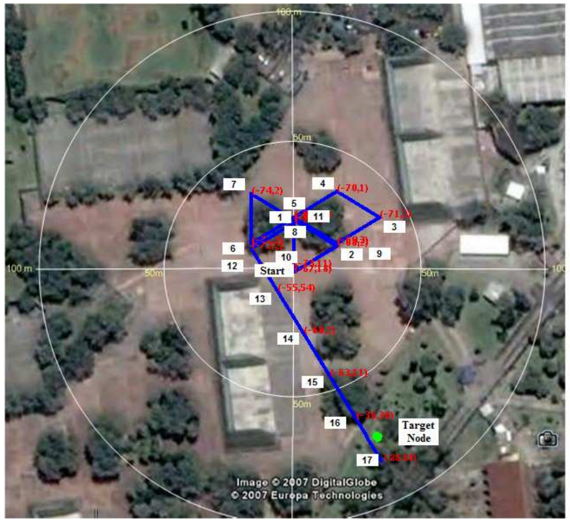
Points	Measured quality
Start	48/100
1	52/100
2	53/100
3	49/100
4	44/100
5	50/100
6	56/100
7	61/100
8	64/100
9	95/100
10	98/100

Fig. 12. Experiment 1 with obstacles



Points	Measured quality
Start	61/100
1	60/100
2	45/100
3	52/100
4	41/100
5	65/100
6	66/100
7	51/100
8	56/100
9	55/100
10	50/100
11	67/100
12	69/100
13	71/100
14	78/100
15	93/100

Fig. 13. Experiment 2 with obstacles



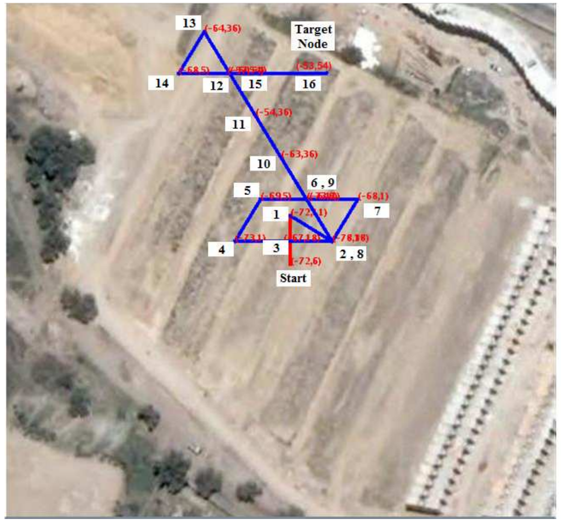
Points	Measured quality
Start	47/100
1	52/100
2	56/100
3	44/100
4	48/100
5	54/100
6	50/100
7	45/100
8	57/100
9	52/100
10	57/100
11	47/100
12	47/100
13	51/100
14	54/100
15	60/100
16	89/100
17	95/100

Fig. 14. Experiment 3 with obstacles

- Point 5. The node measures a link quality value of 65/100. Points 3, 4 and 5 again form an equilateral triangle. As explained before, the corresponding entry in Table 1 makes the node turn 30° CCW.
- Point 6: The node measures a link quality value of 66/100 getting better than the one observed at point 5. Therefore, the node continues moving on the same trajectory.
- Point 7: The node measures a link quality value of 51/100 getting worse than the one observed at point 6. Therefore, the node turns 120° CCW.
- Point 8: The node measures a link quality value of 56/100. Points 6, 7 and 8 form an equilateral triangle. According to Table 1 the node turns 80° CCW.
- Point 9: The node measures a link quality value of 55/100 getting worse than the one observed at point 8. Therefore, the node turns 120° CCW.
- Point 10: The node measures a link quality value of 50/100. Points 8, 9 and 10 form an equilateral triangle and according to Table 1, the node turns 120° CCW.
- Point 11: The node measures a link quality value of 67/100 getting better than the one observed at point 10. Therefore, the node continues moving on the same trajectory.
- Point 12-15: link quality in these points continues to get better and better, the node continues moving along the same trajectory eventually reaching the target node.

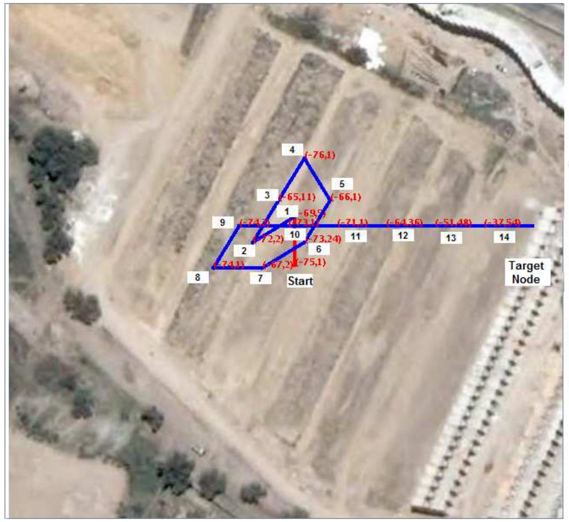
In Figures 12, 13 and 14 we observe that the moving node manages to get closer to the target node. It is also observed that it moves in circles a few times before heading directly to the target node, this situation could be due to the trees located at the starting position. Link quality measurements are not immune to propagation effects, however, GUIDE compensates for some of the variations observed in link quality measurements leading to satisfactory performance.

The following four experiments show the performance of the algorithm in an obstacle-free scenario. These experiments are depicted in Figures 15-18. The most important observation from obstacle-free experiments is that even in open areas the method presents some loops before heading to the target node. In fact, some of these loops are even larger than the ones observed in the experiments with obstacles. We speculate that this is due to the fact that in the obstacle-free scenarios a 20-meter movement does not significantly change the measured value of the metric, whereas the same movement in the other scenario creates more significant changes thus facilitating the guidance. Nevertheless, the algorithm manages to find the target node.



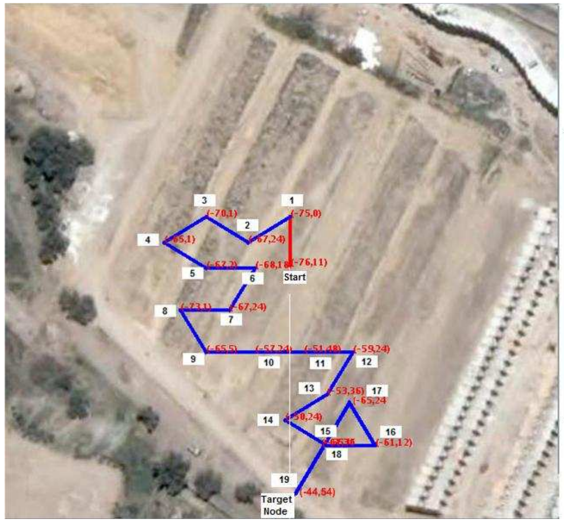
Points	Measured quality
Start	55/100
1	53/100
2	52/100
3	58/100
4	56/100
5	57/100
6	59/100
7	56/100
8	55/100
9	59/100
10	60/100
11	65/100
12	71/100
13	64/100
14	62/100
15	70/100
16	76/100

Fig. 15. Experiment 1 in an obstacle free space



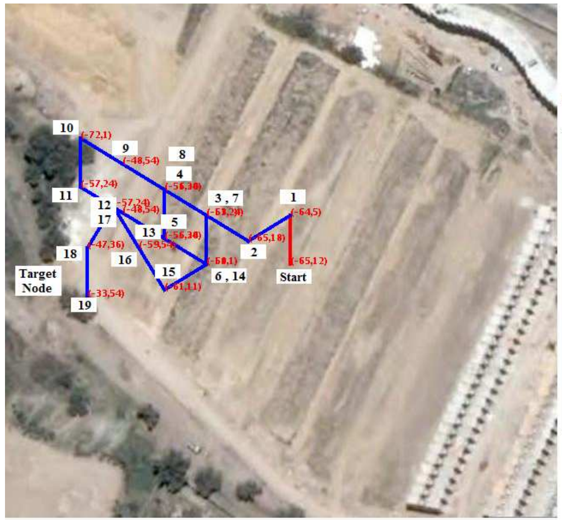
Points	Measured quality
Start	44/100
1	50/100
2	49/100
3	60/100
4	45/100
5	52/100
6	52/100
7	52/100
8	43/100
9	44/100
10	51/100
11	53/100
12	59/100
13	73/100
14	87/100

Fig 16. Experiment 2 in an obstacle free space



Points	Measured quality
Start	42/100
1	43/100
2	52/100
3	52/100
4	46/100
5	61/100
6	55/100
7	59/100
8	53/100
9	56/100
10	70/100
11	77/100
12	61/100
13	80/100
14	78/100
15	81/100
16	76/100
17	74/100
18	82/100
19	89/100

Fig. 17. Experiment 3 in an obstacle free space



Points	Measured quality
Start	52/100
1	60/100
2	63/100
3	78/100
4	74/100
5	76/100
6	58/100
7	59/100
8	75/100
9	77/100
10	52/100
11	62/100
12	74/100
13	75/100
14	58/100
15	58/100
16	73/100
17	69/100
18	76/100
19	93/100

Fig. 18. Experiment 4 in an obstacle free space

7. On node-to-node guidance in ad hoc networks

In this section we comment on how the guiding system can be generalized to multi-hop ad hoc systems. Similar to WLAN technology, so far there has not been significant research related to guiding users in ad hoc networks. In ad hoc networks there may be several intermediate nodes, working as relays in the communication path between the moving and the target node. Figure 19 below illustrates an example of a multi-hop route involving two intermediate nodes. While the guiding problem in multi-hop ad hoc networks may appear far more complex than the simpler WLAN (single-hop) problem that we have been addressing, it can be easily solved if we break the multi-hop problem in various single-hop pieces. Assuming that a routing protocol can find a route between the moving and the target nodes, the moving node only needs to get closer to the first node in route (node *A* in Figure 19) using GUIDE. Once the moving node gets closer to the second node in route (node *B*) and so on until the target node is finally reached.

There are various routing systems for ad hoc networks that could be used as the core of a multi-hop guiding system. LAR [6] is an example of a GPS based routing protocol for ad hoc networks that could be adapted for node-to-node guidance. However, we already mentioned the drawbacks of using GPS technology in a guiding context.

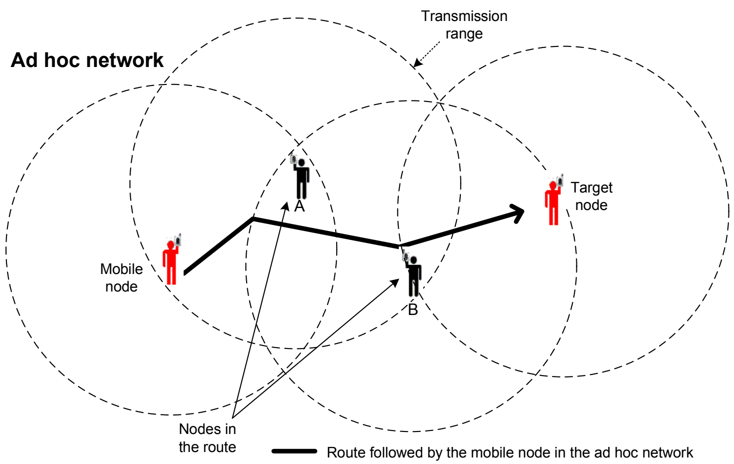


Fig. 19. Guiding users in a multi-hop environment.

In a multi-hop context, the moving node needs only to be aware of the identities (i.e., IP or MAC addresses) of the intermediate nodes in the route as it gets closer to the target node. Most routing protocols for ad hoc networks provide such information in different ways. For instance the DSR routing protocol [4] includes the full list of nodes each packet should visit as it travels from the source to the destination. Distance vector based protocols (e.g., AODV [10]) do not provide the full list of intermediate nodes in routes, each node is aware of the identity of the next one in route only. However, this problem can be solved in various ways including the addition of extra signaling or by means of overhearing traffic and figuring out which node is relaying packets to which other node.

8. Conclusions

Node- to-node guidance in WLAN and ad-hoc environments is a research topic that has just been barely addressed by researchers. However, we identified a number of useful applications that would be possible if common wireless devices had such guiding capabilities. We envision many possible applications that range from everyday duties to critical tasks.

At first glance it may seem that guiding functions can be incorporated in a wireless device by means of specialized hardware only. However, in this work we have shown that it is feasible to use standard WiFi devices in order to produce a reasonable guiding experience. The solution explored in this paper is based on monitoring of real time information regarding the status of the wireless link. Such information can be easily retrieved from any 802.11 wireless interface so that the described system can be easily deployed.

We identified at least two different approaches for implementing guiding functions in a wireless device. One of them uses estimates of the distance to the target node, which are collected at several places so that the correct direction to can be found by triangulation. In our implementation we used a propagation model to relate the received signal power and the distance. This method turns out to be highly sensitive to distance estimation errors. Although its performance can be improved with more accurate estimation techniques, we also explored a different approach that turned out to be robust enough for our purposes. The second approach is based on the computation of a gradient in the neighborhood of the moving node. In fact, in our implementation we used a simplified computation with good results.

It is worth mentioning that use of standard WiFi communication equipment for measurement purposes is not a straightforward task and it has several limitations. In addition, signal measurements are affected by noise, multipath interference and signal attenuation due to obstructions. All these factors make it necessary to take several measurements and process them to obtain a reliable estimate of the metric being measured. Additionally, a number of estimates have to be collected at different locations so that outliers can be pin pointed and discarded. Ideally, we would like that the number of measurements was very low and the temporal misleading be as short as possible. This is part of our future work in this topic. In future research we also plan to study the particularities of node-to-node guidance in mobile ad hoc networks.

References

[1] Bahl, P. and V. N. Padmanabhan, "Radar: an in-Building RF-Based User Location and Tracking System", in *Proceedings of IEEE INFOCOM 2002*, Tel Aviv, Israel, 7-9 (2002).

[2] Bardwell, J., "You Believe You Understand what You Think I Said ... -- The truth about 802.11 signal and noise metrics", Connect 802 Corporation, white paper (2004).

[3] Golden, S. A. and S. S. Bateman, "Sensor Measurements for WiFi Location with Emphasis on Time-of-Arrival Ranging", *IEEE Transactions on Mobile Computing*, vol. 6, no. 10, 1185-1198 (2007).

[4] Johnson, D. B., D. A. Maltz. And J. Broch, "DSR: The Dynamic Source Routing Protocol for Multi-Hop Wireless Ad Hoc Networks", in **Ad Hoc Networking**, edited by Charles E. Perkins, Chapter 5, Addison-Wesley, 139-172 (2001).

[5] Kitasuka, T., K. Hisazumi, T. Nakanishi and A. Fukuda, "Positioning Techniques of Wireless LAN Terminals Using RSSI between Terminals", in *Proceedings of the 2005 International Conference on Pervasive Systems and Computing (PSC-05)*, Las Vegas, Nevada, USA, 47-53 (2005).

[6] Ko, Y. B. and N. H. Vaidya, "Location-Aided Routing (LAR) in Mobile Ad Hoc Networks", in *Proceedings of the 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking* (*MOBICOM*), Dallas, U.S.A, 67-70 (1998).

[7] Niculescu, D. and B. Nath. "Ad Hoc Positioning System (APS)", in *Proceedings of IEEE GLOBECOM 2001*, San Antonio, vol. 5, 2926-2931 (2001).

[8] Niculescu, D. and B. Nath. "Ad Hoc Positioning System (APS) Using AoA", in *Proceedings of the IEEE INFOCOM 2003*, San Francisco, vol.3, 1734-1743 (2003).

[9] Parkinson, B. W. and JJ. Jr. Spilker, **Global Positioning System: Theory and Application**, American Institute of Astronautics and Aeronautics (1996).

[10] Perkins, C. E. and E. M. Royer, "Ad hoc On-Demand Distance Vector Routing." *Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications*, New Orleans, LA, 90-100 (1999).

[11] Rappaport, T. S. **Wireless Communications, Principles and Practice**. Prentice Hall (1996).

[12] Shoeb, M., F. Ahmad, and M. Amin, "Narrowband Source Localization for Indoor Wireless Environments", in *Proceedings of the Fifth IEEE International Symposium on Signal Processing and Information Technology*, Athens, Greece, 411- 416 (2005).

[13] Tourrilhes J., "Wireless Tools for Linux", Hewlett Packard, date of last update: January 16, 2008, date of consultation: July 14, 2008, Available: http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/Tools.html

[14] Yamasaki, R., A. Ogino, T. Tamaki, T. Uta, N. Matsuzawa and T. Kato, "TDOA location system for IEEE 802.11b WLAN", in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC'05)* Vol. 4, 2338-2343 (2005).

[15] Yuan, H., Yao W. and Song Y., "Location Sensing in Enhanced IEEE802.11e WLANs", *20th International Conference on Advanced Information Networking and Applications (AINA'06)*, Austria, Vienna, 423 – 428 (2006)

Acknowledgments

This work was supported in part by research funds from CONACyT grant 47197-Y and PAPIIT grant IN105307.

Biographies

Marco Antonio González Silva received the B.Sc. in Computer Engineering and the M.Sc. in Computer Engineering from the National Autonomous University of Mexico (UNAM). He has worked as project coordinator in the design and implementation of cross connection solutions for LAN, ISDN, Frame Relay, ATM and xDSL technologies. He is currently a Ph.D. student at UNAM.

Javier Gomez received the BS degree with honors in Electrical Engineering in 1993 from the National Autonomous University of Mexico (UNAM) and the MS and PhD degrees in Electrical Engineering in 1996 and 2002, respectively, from Columbia University and its COMET Group. During his PhD studies at Columbia University, he collaborated and worked on several occasions at the IBM T.J. Watson Research Center, Hawthorne, New York. His research interests cover routing, QoS, and MAC design for wireless ad hoc, sensor, and mesh networks. Since 2002, he has been an Assistant Professor with the National Autonomous University of Mexico. Javier Gomez is member of the SNI (level I) since 2004.

Miguel Lopez-Guerrero received his B.Sc. with Honors in Mechanical-Electrical Engineering in 1995 and the M.Sc. with Honors in Electrical Engineering in 1998, both from the National Autonomous University of Mexico. He received his Ph.D. in Electrical Engineering from the University of Ottawa in 2004. Currently, he is an Associate Professor with the Metropolitan Autonomous University (Mexico City). His areas of interest are in medium access control, traffic control, and traffic modeling.

Jose E. Torres-Fernandez got his Bachelor Degree in Electrical Engineering at the National Autonomous University of Mexico (UNAM) in 1995. He got his Master Degree in Electrical Engineering at UNAM (1997). He got his PhD in Electrical Engineering: Systems at University of Michigan (Ann Arbor, 2003). He worked at Delphi Research Labs from 2004 to 2006 (Shelby Township, Michigan). At the present time he is a professor at the Department of Electrical Engineering of the Autonomous Metropolitan University (UAM) in Mexico City.

Víctor Rangel Licea obtained his Bachelor Degree in Computer Engineering from the National Autonomous University of Mexico (UNAM). He obtained his Master Degree in Data Communication Systems and his doctoral degree in Telecommunications Engineering from the Centre for Mobile Communications Research, The University of Sheffield (England). His PhD thesis focused on the modeling and analysis of Cable TV networks supporting broadband Internet traffic. Dr. Rangel is currently a professor at the Department of Telecommunications Engineering, School of Engineering (UNAM).