Opportunistic Object Binding and Proximity Detection for Multi-modal Localization

Maarten Weyn, Isabelle De Cock, Yannick Sillis, Koen Schouwaert, Bruno Pauwels,
Willy Loockx and Charles Vercauteren
Dept. of Applied Engineering
Artesis University College of Antwerp
Antwerp, Belgium
maarten.weyn@artesis.be
{Isabelle.decock, yannick.sillis, koen.schouwaert, bruno.pauwels}@student.artesis.be
{willy.loockx, charles vercauteren}@artesis.be

***Abstract*—In this paper, opportunistic object binding is proposed to improve multi-modal localization. Object binding and proximity detection will be realized using Bluetooth and Wireless Sensor Networks. Multi-modal localization is created using an opportunistic seamless localization system, fusing Wi-Fi, Bluetooth, Wireless Sensor Networks, GSM, GPS, RFID and inertial sensors. In this paper object binding is used to locate devices which can not be located without the help of bound objects.**

***Index Terms*—object binding, localization, opportunistic localization, multi-modal localization, Bluetooth, WSN, Wi-Fi, proximity detection.**

I. Introduction

Today, location based services are widely spread and already integrated in many applications such as GPS navigation systems, Google Earth, track and trace systems, Foursquare, etc. Outdoor localization is mostly accomplished by means of GPS, but usually GPS does not work indoor because there has to be a minimum of four satellites in line of sight, which is usually not the case indoor. Indoors, we can use Bluetooth [1], [2], Wi-Fi [3] or GSM [4], or even other techniques such as Wireless Sensor Network (WSN) [5], [6] and Ultra Wide Band (UWB) [7].

One big challenge is fusing these techniques into a single system. Acquiring the sensor data of multiple sensors can be realized because most mobile devices such as Personal Digital Assistants (PDAs) and smart phones are very often equipped with GSM, GPS, WiFi or a combination of these. A system which combines this technologies is called Opportunistic Seamless Localization System (OLS) [3].

The future of localization systems most likely will evolve towards systems that can adapt and cope with any available information provided by mobile clients without the need to install any additional dedicated infrastructure. This type of localization is called opportunistic localization. It is defined as [8]: *"An opportunistic localization system is a system, which seizes the opportunity and takes advantage of any readily available location related information in an environment, network and mobile device for the estimation of the mobile device absolute or relative position without relying on the installation of any dedicated localization hardware infrastructure."*

The OSL system combines the earlier mentioned technologies together with the information of accelerometers, compass and camera.

Currently, in OSL, the clients or 'trackable objects' can be any laptop running Windows or Linux, any smartphone running Windows Mobile, Android or OpenMoko or dedicated OSL Wi-Fi or Zigbee tags. The complete system overview is shown in Figure 1.

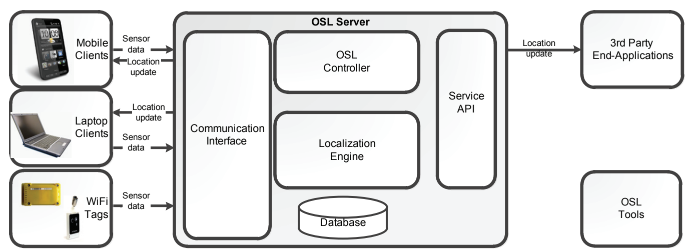


Fig. 1. The OSL system architecture

The clients send raw sensors data of the above mentioned technologies to the server, where the communication interface will parse these messages and send the appropriate data to the localization engine which will calculate a position estimation. This estimation is sent to the Service API which facilitates the communication with 3rd party application to, for example, visualize the positions on a map or trigger any events.

The localization engine seamlessly fuses the heterogeneous sensor data using an adaptive observation model for the particle filter, taking the availability of every technology and sensor data into account. A particle filter [9] is a sequential Monte Carlo based technique used for position estimation. Since we are working with a real-time system, it is even harder to estimate the correct position therefore heavy and numerous calculations are not recommended.

Limiting the number of particles is recommended in order to avoid extensive time-consuming calculations. For example, when the system is implemented in a large scale environment, such as an airport where many devices are present, the system might be delayed due to these calculations for all those devices. Obviously, some objects will travel together such as people traveling by bus. In such cases, it is not necessary to calculate all their positions with different particle clouds. Instead, we could combine all these objects and bind them in one group, in which case we only have to calculate one position for this group.

Besides from this optimization related reason for object binding, object binding also enables the system to locate objects which can not be located by its own.

Bluetooth, for example, is a useful technology to detect other adjacent Bluetooth devices. Which would enable the possibility to detect whether people are moving together. Another interesting reason to use Bluetooth may be the possibility to locate unknown people. This can, for example, be useful to estimate the amount of people in a given area.

Another technology, which can be used to detect the proximity of one device towards another, is WSN.

A third way of using object binding is to combine multiple tags or devices which are related to one object, for example, a person having a laptop and a smartphone. In this case the location data of the two devices has to be analyzed. Two possibilities can happen, first the object can be merged, for example, when the laptop and the smartphone are both in the neigbhourhood of each other and most probably also in the neighbourhood of the person. Alternatively, heuristics can determine that the two devices are not at the same place, for example, when the laptop is still in the office but the person is walking with his smartphone through the building. In this case the position of the laptop can not be connected with the position of the person anymore.

This paper is structured as follows: at first, Bluetooth object binding is discussed, where the scanning method for Bluetooth is analyzed followed by some real experiments to determine the operational range of Bluetooth devices. Thereafter, Bluetooth signal strength values are discussed. This is then followed by a short introduction about opportunistic seamless localization and the explanation of the Bluetooth measurement model. Afterwards, WSN proximity detection is discussed with some corresponding experiments. Finally, before the conclusion, multiple device binding is explained.

II. Bluetooth Object Binding

In this section, the use of Bluetooth for object binding and the localization algorithm will be explained.

A. Bluetooth

Bluetooth [10] is a technology developed by Ericsson. This universal radio interface in the 2.45 GHz band makes it possible to connect portable wireless devices with each other. Bluetooth uses frequency hopping to avoid interference with other devices, which also use the license-free 2.45 GHz band.

1) Discovering: There are two ways of discovering [11] devices when using Bluetooth. The first, and mostly used method, is inquiry-based tracking. In case of inquiry-based tracking, the base station needs to scan for devices and to page all present devices in order to find them. All devices need to be detectable but they need not to be identified in advance.

Scanning for devices absorbs a relatively large amount of time because primarily every base station sends a search-packet on all 32 radio channels. Every detectable device that receives this packet will answer. To avoid collision, every device will send his packet with a random delay. This is the reason why an inquiry has to run for at least 10.24 s to be reliable. Many devices are undiscoverable in order to increase the security and privacy of the owner. This is another technical problem that could occur and consequently it is not possible to find these devices by scanning the area.

A second method of tracking is the connection-based tracking. With connection-based tracking, devices are considered to be in a close range when one device has the possibility to connect with another device. All devices have to be paired with each other and this is a major problem when using the Radio Frequency Communications (RFCOMM) layer [12] connections with connection-based tracking. Practically, this requires human input which is time-consuming. Although, some communication services do not require this, it is still necessary that one of both devices knows the other one exists.

In practice, the creation of an Asynchronous Connectionless Link (ACL) [12] and a basic Logical Link Control and Adaptation Protocol (L2CAP) layer [12] connection is universal and authorization-free. These connections are limited but they are in compliance with the requirements for tracking usage. It is only necessary to know whether a connection is possible and if this is the case, these 2 devices are in the same range. This connection also supports some low-level tasks such as RSSI measurements and L2CAP echo requests.

Both tracking techniques have their own advantages and disadvantages and they are both not ideal. Choosing the correct technique will depend on the situation. When using inquiry-based tracking, it is possible to find every detectable device without the need of knowing the devices in advance. The major disadvantage will be the relatively long scan time. When we choose the other option, connection-based tracking, the time to find the devices will be shorter and there is also the possibility to find undiscoverable devices. The major disadvantage here is the requirement that at least one party knows about the existence of the other one.

Another option could be a combination of both techniques. Combining these two techniques will not decrease the relatively long scan time because we always need to take the longest scan time in account. The advantage of combining both techniques is the possibility to find known 'undiscoverable' devices as well as unknown discoverable devices.

In this paper, the first option is chosen because inquiry-based tracking has the possibility to track unknown devices, which will be useful for object binding.

2) Range: Bluetooth devices can be divided in three different classes. Generally, class 1 and class 2 are used instead of class 3, which is due to the very short operating range of class 3.

Class	Maximum Power	Operating Range
1	100 mW (20 dBm)	Up to 100 m
2	2.5 mW (4 dBm)	Up to 10 m
3	1 mW (0 dBm)	Up to 1 m

These operating ranges are frequently used to estimate a position since signal strength is not always a good parameter due to effects like reflection and multi-path propagation [13] .

The operating range of a Bluetooth device can be defined by the maximum allowable path loss which can be calculated with Equation 1:

$$L_{total} = 20 * \log_{10}(f) + N * \log_{10}(d) + L_f(n) - 28 \quad (1)$$

$$L_{total} = 40 + 20 * \log_{10}(d) \quad (2)$$

where N is the *Distance Power Loss Coefficient*, f is the Frequency (Mhz), d is the distance (meters) between the nodes , L_f is the *Floor Penetration Loss Factor* (dB) and n is the number of floors penetrated.

When working in an open-air environment, Equation 2 which is the simplified version of Equation 1, can be used [14].

As operating ranges will be used to estimate a position, some tests were done in order to decide which maximum range will be used. A Dell XPS M1530 laptop has been set up as a base station. The two test devices were a Samsung E250 mobile phone (test device 1) and a Samsung F450 mobile phone (test device 2). All devices, including the base station are devices of class 2. The measurements were started at a distance of one meter away from the base station and afterwards extended by steps of one meter. Every measurement was repeated five times in order to have reliable results.

Fig. 2. Experiment 1

The first experiment, as shown in Figure 2, was done in open space in which the two test devices are in line-of-sight of the base station.

Both test devices could easily bridge a distance of 9 m. Once the distance was increased, test device 1 was not longer detectable. Test device 2 was detectable until we reached a distance of 12 m.

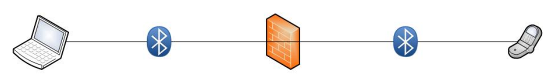


Fig. 3. Experiment 2

In the next experiment, the influence of obstacles between the base station and the test devices was tested. This experiment was firstly done with a window between the base station and the test devices. Secondly the experiment was repeated with a 14 cm thick brick wall instead of a window, see Figure 3.

Theoretically, obstacles comparable to a wall should significantly decrease the Bluetooth signal or even make it impossible to connect with devices behind such obstacles. According to [?] the attenuation of a 2.4 Ghz signal through a brick wall of 8.9 cm is 6 dBi, of a concrete wall of 45 cm is 17 dBi and the attenuation of an exterior single pane window is 7 dBi. It is very hard to predict the attenuation because the exact material of the obstacle is generally not know. Our test with a window started showing problems with detecting test device 1 at a distance of 4 m. Test device 2 remained detectable up to 7 m and at larger distances it started to show some discontinuities.

The following test with a wall instead of a windowpane showed these results: at a distance of 4 m, test device 1 started to disappear and at larger distances, test device 1 was rarely detected. Test device 2 on the other hand, was much longer visible. In a range up to 7 m, test device 2 was still detectable.

These results, as can be seen in Figure 4, show a general range of 10 m when the base station and test device reside in the same area hence we are working in an open space.

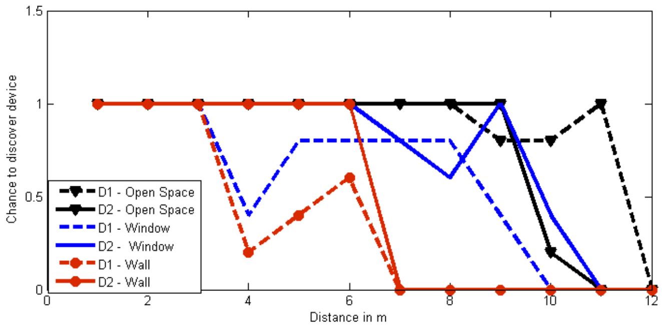


Fig. 4. Results

Obstacles like walls obviously have some influence on this range. Generally we can decrease the range down to 5 m.

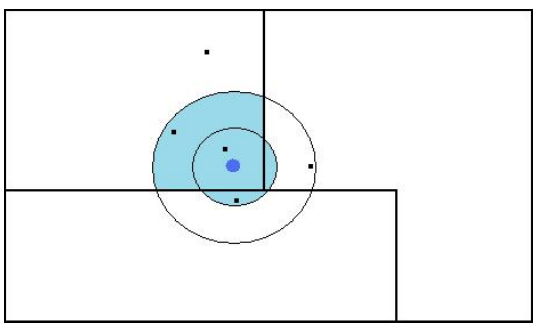


Fig. 5. Range

Consequently, when a Bluetooth device detects another Bluetooth device, this estimation will be located in a circular area with a radius up to 10 m in open space. Walls will limit the radius up to 5 m.

3) Signal Strength: RSSI values are often used in order to estimate the proper distance between 2 devices because Bluetooth does not offer an interface to extract the real received signal strength directly [15]. Theoretically, RSSI values should vary exponentially with the real distance but in practice this is not always the case [16].

Although there is no deterministic relationship between distance and RSSI, due to fading, reflection etc., there is a correlation: when the RSSI value decreases, we know the distance becomes longer and conversely; when the RSSI value increases, the distance diminishes. This information can be used to discover whether devices move away from each other, towards each other or together.

Hallberg and Nillson [17] show that using RSSI values for calculating the distance between 2 devices is not reliable. Nevertheless, RSSI values could be useful to implement object binding. Object binding should only be realized when 2 or more objects are very close. At this point, the RSSI values will be higher. Nonetheless, these values will fluctuate. In this way, it is necessary to use a range of RSSI values in order to decide whether objects should be bound or not.

In this paper, RSSI values are not used because they bring up another disadvantage: a device needs to set up a connection with the other device and this will increase the scanning time. Considering the fact that we are working with a real-time system, the scanning time should be as short as possible.

B. Opportunistic Seamless Localization and Bluetooth Object Binding

The opportunistic seamless localization system combines all location related information readily available from multiple technologies such as Wi-Fi, GSM, GPS, accelerometers [18] etc. In this paper we propose a novel method, which allows taking into account object binding via a Bluetooth link to other devices as an additional source of location related information which may be successfully used by the OSL system for further improvement on location estimation reliability and accuracy. As presented by Hallberg *et al.* [2], the Bluetooth link connectivity on its own does not provide sufficiently accurate location information for most of the mobile applications. Therefore, to successfully fuse the Bluetooth connectivity information for locating Bluetooth enabled devices, a specific method described in this paper has been developed for efficient incorporation into the OSL system fusion location data engine. The OSL fusion engine is based on the recursive Bayesian estimation implemented as a particles filter, therefore, also a likelihood observation function used for the particles weighting was developed.

1) Communication: Firstly, the client scans for all nearby devices. The MAC address of every found Bluetooth device is sent to the server. In the mean time, the client keeps scanning for devices and will regularly send an update.

At the server side, every incoming MAC address will be compared to a list of known MAC addresses. In this list all primarily known Bluetooth devices are saved. Every Bluetooth measurement has 4 arguments, at first the MAC address, secondly a boolean to indicate whether the device is fixed or mobile, thirdly the coordinates when the device has a fixed place and at last every mobile device has an ID.

When a match between incoming MAC address and a MAC address in the list is found, these MAC addresses are saved in a list.

2) Measurement Model: The Bluetooth measurement model is designed to deal with different situations. A complete overview of this measurement model can be found in Figure 6.

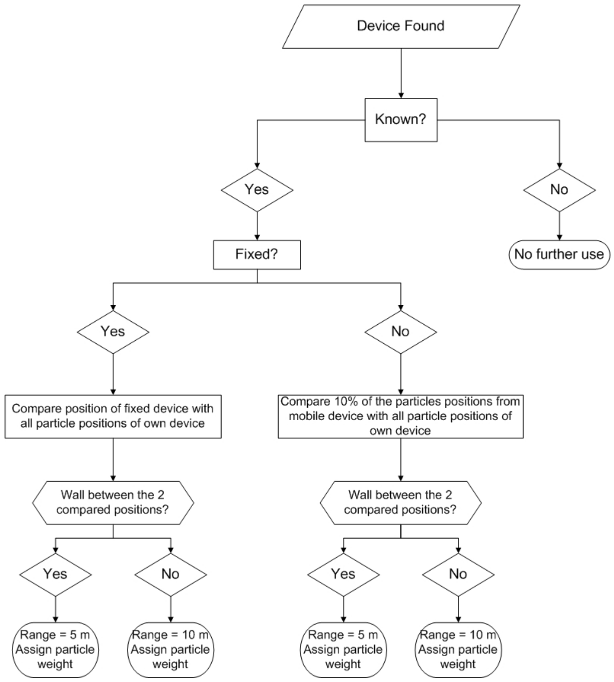


Fig. 6. Flowchart

There are 3 possible options when one or more Bluetooth devices are found. The first option happens when the found devices are unknown. These devices can not be used to localize the client device. Though, these devices can give some interesting information, such as how many devices were present at a certain time in a certain place. This is already implemented at some places such as Brussels Airport [19]. Every Bluetooth device that is discoverable will be detected by fixed antennas. In this way it is possible to measure the time necessary to move from one point to another and consequently it will be possible to calculate the waiting time to pass for example through the safety zone. When the found device is known, there are 2 options left: this device can be a fixed device, this is the second option, or a mobile device which is the third option.

Dealing with the second option, returns a fixed position with the exact coordinates of the fixed device. With the knowledge that a Bluetooth device is only visible within a certain area around that device, the weight of all particles from the client can be adapted.

Calculating the euclidean distance between every particle and the fixed device is the first step. After having calculated the distance between one particle and the fixed device, there will be a wall check. A wall has a big influence on the signal strength and for that reason it is important to know whether there is a wall between the fixed device and the particle. The choice to work with a larger or smaller range depends on the absence or presence of a wall. Based on this range, the new particle weight will be calculated.

If the third option occurs, a known mobile device is found. This device does not show exact coordinates since the location of every mobile device is predicted with a particle cloud. Depending on the situation, a particle cloud can consist out of 100 particles up to 1000 particles. Comparing every particle of the found device with every particle of the client device would be too heavy for a real-time system. For this reason, 10 percent of random particles from the found device are compared to all particles of the client device. Choosing 10 percent still gives us a reliable amount of particles. The coordinates of these particles are loaded and the distance between these particles and the client device particles is calculated. Again, we need to check if there is no wall between the particles. Based on this information, the particle weight can be calculated.

Obviously, it is possible that more than one device is found. For all those devices, previously mentioned options will be looked at and for every device, the correct option will be chosen. Working with multiple found devices, all calculated particle weights are multiplied for every client particle. In this way all found devices are brought into the calculation and the result becomes more accurate.

3) Particle Weight: According to the test results in the section 'Range', a range of 10 m will be used in open space and there will be a range of 5 m when there is an intersection of a wall. It would be inaccurate to assume that discovered devices are always in a range of 10 m with equal chances to be everywhere in that circle. For this reason, using the sigmoid function gives a more realistic image. In this case, the following functions are used:

$$y = \frac{1}{1 + e^{x-10}} \tag{3}$$

$$y = \frac{1}{1 + e^{x-5}} \tag{4}$$

Equation (3) is used for open space. This function gradually decreases and the particle weight will be based on this function, see Figure 7. Equation (4) is used when a wall between the 2 devices is detected. This function will decrease earlier because the obstacle has a big influence on the signal strength which consequently will decrease quickly.

The measurement model for using Bluetooth measurements with fixed devices is shown in Algorithm 1. An example measurement probability is shown in Figure 8. In Figure 8(a) the likelihood function when a device at position (0,0) is discovered, is shown. A Class 2 Bluetooth device can be discovered up to 10 m distance in line-of-sight. A Sigmoid function is used to create a soft threshold between the discoverable and the non-discoverable distance. In the example, there is a wall from (-20,-3) to (-20,3). Since a wall attenuates the Bluetooth signal, the maximum discoverable

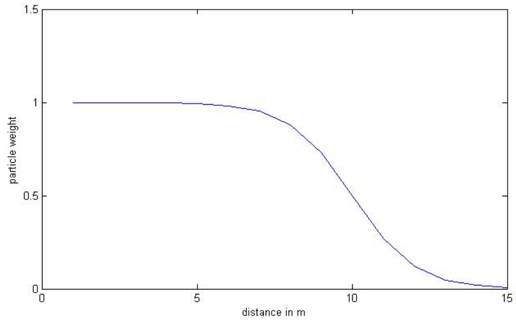
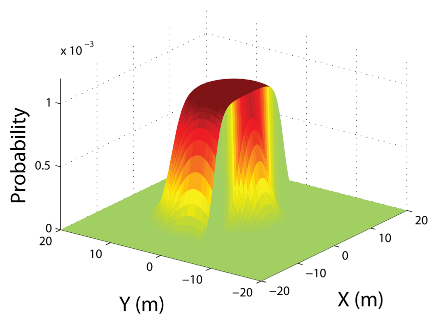
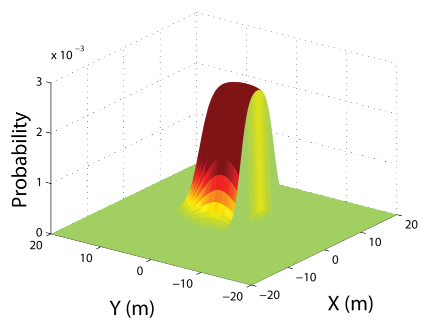


Fig. 7. Sigmoid function



(a) Bluetooth device at (0,0).



(b) Bluetooth device at (0,0) and (10,0).

Fig. 8. Example of Bluetooth measurement probability with a wall at y = -3.

distance will be lowered to 5 m if passing a wall. In Figure 8(b) two devices are discovered, one at (0,0) and one at (10,0). In the case of multiple devices, the Likelihood Observation Function (LOF) for each device is multiplied to get a LOF, which incorporates all discoverable devices.

The measurement model for using Bluetooth devices with mobile devices using object binding is shown in Algorithm 2 and an example of such a likelihood based on a bound object located with Wi-Fi is shown in Figure 9.

III. Bluetooth Based Object Binding Experiments

For these experiments, indoor localization is accomplished by using Wi-Fi and Bluetooth. In these tests, the client is only located by using Bluetooth. Multiple tests with fixed

Algorithm 1: **Bluetooth_Measurement_Model** ($\mathbf{z}_t$, $\mathbf{x}_t$)

$w = 1$
for all Bluetooth devices $b \in \mathbf{z}_t$ **do**
  if b is known and fixed position x_b **then**
    if no wall between x_t and x_b **then**
      $w = w.\dfrac{1}{1 + \mathrm{e}^{\mathrm{d}(\mathrm{x_t},\mathrm{x_b})-10}}$
    else
      $w = w.\dfrac{1}{1 + \mathrm{e}^{\mathrm{d}(\mathrm{x_t},\mathrm{x_b})-5}}$
    end if
  end if
end for
return w

Algorithm 2: **Object_Binding_Bluetooth_Measurement_Model** ($\mathbf{z}_t$, $\mathbf{x}_t$)

$w = 1$
for all Bluetooth devices $b \in \mathbf{z}_t$ **do**
  if b is known and particle distribution $\mathcal{X}_b$ known **then**
    take sample set $\bar{\mathcal{X}}_b$ from $\mathcal{X}_b$
    for all x_b^i in $\bar{\mathcal{X}}_t$ **do**
      if no wall between x_t and x_b^i **then**
        $w = w.\dfrac{1}{1 + \mathrm{e}^{\mathrm{d}(\mathrm{x_t},\mathrm{x_b^i})-10}}$
      else
        $w = w.\dfrac{1}{1 + \mathrm{e}^{\mathrm{d}(\mathrm{x_t},\mathrm{x_b^i})-5}}$
      end if
    end for
  end if
end for
return w

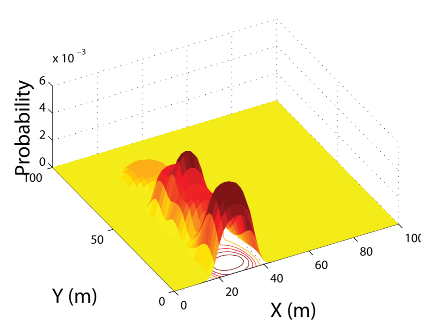
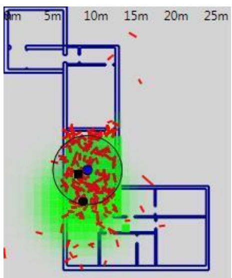


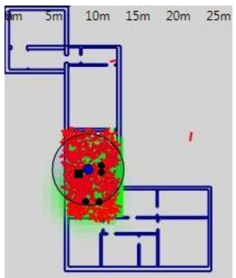
Fig. 9. Example of Bluetooth measurement probability using object binding.

and mobile Bluetooth devices were done. The first test was done with one fixed and known device, see Figure 10(a).

The estimated position is located at the center of the circle, the real position is represented by a square and the position of the found and known Bluetooth devices is represented by dots. It shows good room level accuracy, although still some particles -representing different hypothesises- are in adjacent room.



(a) Test with 1 fixed device



(b) Test with 4 fixed devices

Fig. 10. Comparison between test with 1 or 4 fixed devices

Repeating this test, but now with 4 known and fixed devices gives us a better result, see Figure 10(b). You see that all hypothesises, represented by the particles, are now inside the correct room. Using more found and known devices results logically in a more accurate estimation. This is due to trilateration. The location of every fixed device will also have an influence on the accuracy, as shown in Figure 11(c) and 11(d). 11(c) shows a good location of fixed devices, the area where the client can be located is very small and consequently more accurate. In 11(d), all fixed devices are close to each other and therefore, the area where the client can be located is still large.

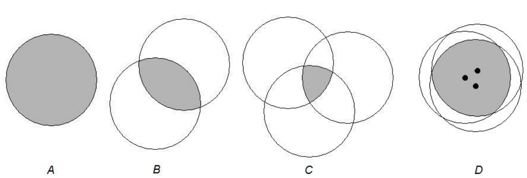


Fig. 11. Trilateration

Obviously the area where the client can be located is a lot smaller when more devices are found. This illustrates why the error rate decreases when the amount of found and known devices increases. Because we are using fixed devices only, it is possible to compare the clients particles with one exact position. Every fixed device has a known position which does normally not change. Therefore the estimated position can be easily calculated with a 100 percent certainty of the location of the fixed Bluetooth device.

Of course this is a kind of localization which is previously already developed in other research such as [2]. But Bluetooth can be used stronger as a sensor when combined with other technologies to perform object binding.

In dynamic object binding, instead of static devices, other mobile devices will be used as references. Mobile devices do not have one exact and correct position. The likelihood of their position is estimated with a particle cloud. In order to calculate the position of the client, all particles will be compared with 10 percent of the particles from a found and known Bluetooth device. It is possible to increase the threshold of 10 percent, but using more particles will result in heavy calculations, using less particles will make the final result inaccurate.

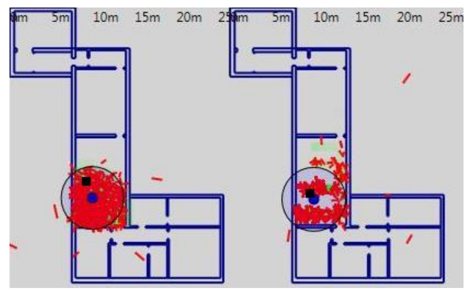


Fig. 12. Test with 1 mobile device

In this test, the client location, shown in 12(a), is calculated based on the particles of another mobile device, shown in 12(b). Due to the fact that we do not have an exact position of the mobile device, we have to estimate the client position based on another estimation. Consequently, the error rate is increased, compared to the test with fixed devices. The error depends largely on the correctness and distribution of the likelihood of the dynamic reference device.

Dynamic object binding makes it possible to locate any found Bluetooth device without the necessity to have any other technology embedded in the device itself. Localization information from all found devices will be used to correctly locate the client device. Merging different technologies improves the final result but within this structure, the position estimation of each device has always been created independent from other devices.

Of course we can combine dynamic reference devices and fixed devices when they are both discovered by the device. This increases the reliability of the estimation.

IV. Wireless Sensor Network Proximity Detection

Wireless sensor networks are characterized by low-cost wireless sensors to perform some action. The ideal wireless sensor should meet certain conditions. Properties like scalability, low power consumption, integration in a network, programmability, capability of fast data transmission and little cost to purchase and install are very important during the fabrication of the sensors. It is not possible to meet all these requirements. Therefore it is very important to know all prerequisites of the application where the sensors will be used. There are two considerations to make, namely the use of low data rate sensors or high data rate sensors. Examples of low data rate sensor include temperature and humidity. Examples of high rate sensors include strain, acceleration and vibration.

Today it is possible to assemble the sensors, radio communications and digital electronics into a single package. Therefore it is possible to make a wireless sensor network of very low cost sensors communicating with each other using smart routing protocols. Basically a WSN network consists of a base station (gateway) and some sensor nodes. These sensor nodes send information directly to the gateway or if necessary use some other wireless sensor nodes to forward the data to the gateway. Eventually the data received in the gateway is presented to the system for processing.

Minimizing power consumption of any wireless sensing node is a key feature to deal with. Mostly the radio subsystem requires the largest amount of power. To minimize power consumption it is recommended to send data over the network only when required. There is also a possibility to minimize the power consumption of the sensor itself. A lot of energy can be saved by only performing sensor measurement when needed instead of continuously. For example to locate people, it is not necessary to send data every second so energy could be saved by only send data every 5 seconds.

A. WSN Network Topology

Different topologies can be used to organize a WSN network:

In a star topology, all nodes are connected to a single hub node. This node handles the routing and must be able to perform more intensive messaging since it handles all the traffic in the network. The hub node is very essential, when it goes inactive, the network will be destroyed.

By using the ring topology, there is no coordinator. All messages travel in one direction, so when one node leaves the ring, the communication is broken.

A bus topology has the property of broadcasting messages to all the nodes connected to the bus. Each node checks the destination address of the message's header and checks if the address is equal to its address. When there is a match, the node accepts the message, otherwise the node does nothing.

More complex, fully connected networks are characterized by a connection from every node to every node. There are a lot of backups, so when one node leaves the network, messages can still be routed via the other nodes. By adding nodes to the network, the number of links increase exponentially, so the routing becomes too complex.

Finally, mesh networks are generally described as distributed networks. Such networks allow communication between a node and all other nodes including those outside its radio transmission range. A big advantage of using this topology is the use of multi-hop communication. Multi-hop communication allows transmission between 2 nodes that aren't in each other range.

By using self-healing algorithms a mesh network has the property to enable a network to operate even when one node breaks down.

B. Existing WSN Localization Systems

Localization using WSN can be applied by using different algorithms. Getting the best results for the localization process depends on two major parts: the influence of noise and the different system parameter settings. Each algorithm perform better in on other environments or with other WSN motes, so for good localization, the used motes and the environments have to be taken into account. The localization techniques for WSN can be divided into two categories: range-based and connectivity-based.

Range-based methods estimate the distance between nodes with ranging methods such as Time-of-Flight, Angle of Arrival and Received Signal Strength. These techniques typically provide better accuracy compared to connectivity-based algorithms, but are more complex. Connectivity-based algorithms do not estimate the distance between nodes but determine the position of a blind node by their proximity to anchor nodes [20].

Langendoen *et al.* [5] present in a survey 3 categories of algorithms for WSN localization: ad-hoc positioning [21], n-hop multi-lateration [22] and robust positioning [23]. The survey concludes that no single algorithm performs perfectly in every situation.

Another comparison is done by Zanca *et al.* [24]. This paper compares four algorithms: Min-Max, multi-lateration [5], Maximum Likelihood [25] and ROCRSSI [26]. The absolute ranging errors of the algorithms are presented with the number of anchor nodes as a parameter. The authors conclude that multi-lateration provides superior accuracy compared to the other algorithms when the number of anchor nodes is high enough. Interestingly, despite its simplicity, Min-Max achieves reasonable performance.

MoteTrack [27] is a decentralized location tracking system. The location of each blind node is computed using a RSS signature from the anchor nodes. This database of RSS signatures is stored at the anchor nodes themselves.

Blumenthal *et al.* [28] present weighted centroid localization, the position of a blind node is calculated as the centroid of the anchor nodes.

C. OSL and WSN Proximity Detection

The goal of this research is to use the nodes of a WSN network to perform localization. Proximity localization will be used to determine the mobile terminals position relative to the nodes with known position. Figure 13 shows the architecture of the localization system.

1) Three-tier network architecture:

a) Mobile tier: For proximity localization to work, the WSN network is divided into three tiers. First we have the mobile tier. This tier contains the mobile devices carried by the people or assets being tracked. Each mobile mote has its own unique ID, used for localization.

b) Fixed tier: The WSN devices in the fixed tier are the nodes on known locations. They can be seen as the routers or access points of a Wi-Fi network. The location of the mobile motes is equated to the location of the router with the highest measured signal strength.

c) Gateway tier: The gateway tier is a WSN node connected to a PC. The WSN node receives the packets coming from the routers. This device handles the communication between the WSN network and the OSL framework and can be seen as a client of the OSL server. It houses the algorithms that transform the data coming from the WSN network into information the OSL server can process, i.e. location updates with a fixed ID and mobile ID as arguments.

2) WSN-to-OSL gateway: Information forwarded from the WSN device is raw data represented in a serial way. In order to access the useful data, we need to parse the serial data coming from the WSN device so that we can access the ID's and RSS measurements. As Figure 13 shows data coming form the WSN network is relayed through a gateway which acts as a client of the OSL server.

3) Localization server: The localization engine for a proximity localization system is pretty straightforward. When the gateway has detected a new nearest fixed router for a given mobile device it sends a location update message to the localization server. The message contains the ID of the new nearest fixed node along with the ID of the mobile terminal itself. The server matches the ID of the nearest router to its actual coordinates and thus locates the mobile device.

V. Proximity Detection Experiments

A. RSS characteristic

$$FSPL(dB) = 10\log_{10}((\frac{4\pi}{c}df)^2) \qquad (5)$$

Since this paper discusses signal strength based localization it is important to know the RSS versus distance characteristic. Figure 14 shows the received signal strength between two identical Zigbit [29] based tags with dual chip antenna which are placed in Line-of-Sight at increasing distance.

Equation 5 gives the Free-Space-Path-Loss, which is the attenuation of a RF signal traveling through a medium, in this case air. The characteristic given by the formula along with the graph show the signal strength has a steep decrease in the first tens of meters. As the two devices move further apart the decrease becomes less steep. A first conclusion we can deduct from this characteristic is that RSS based localization will perform better in close range. The steeper the RSS curve the better the system can distinguish different distances between router and mobile terminal. Since indoor locations such as offices or classrooms are typically limited in size, the RSS based localization should perform reasonably well in indoor environments. In addition to the steep RSS curve, the presence of walls will improve the room-based localization. When each room is equipped with one fixed node, the signal coming from that node will be dominant compared to the signals of the fixed nodes in other rooms because of the RF attenuation caused by the walls between two rooms.

Because this system is intended for indoor localization, we tested the WSN system in an indoor office environment. We tested both Line-of-Sight and Non-Line-of-Sight conditions to determine what's the optimal choice when positioning the

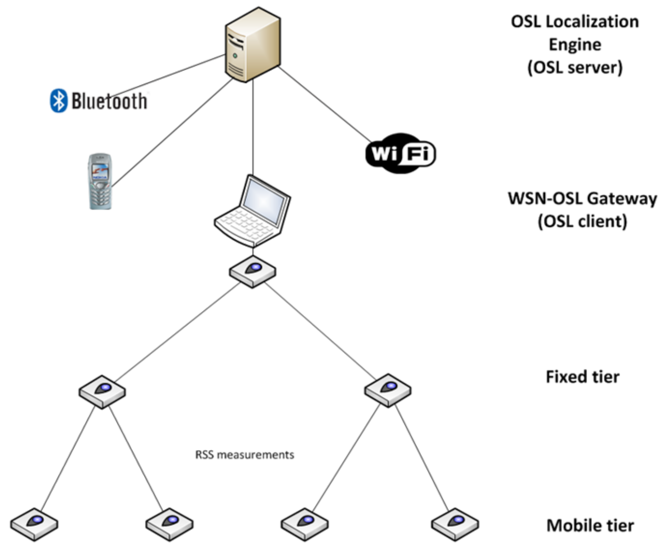


Fig. 13. Achitecture of the localization system

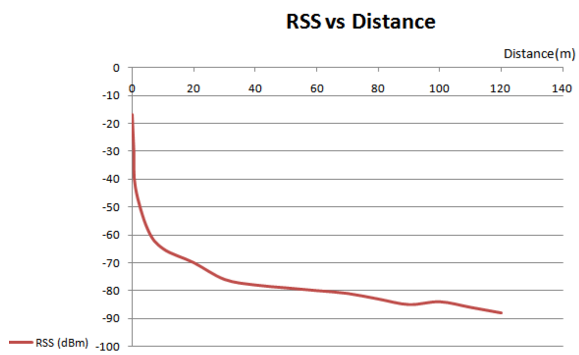
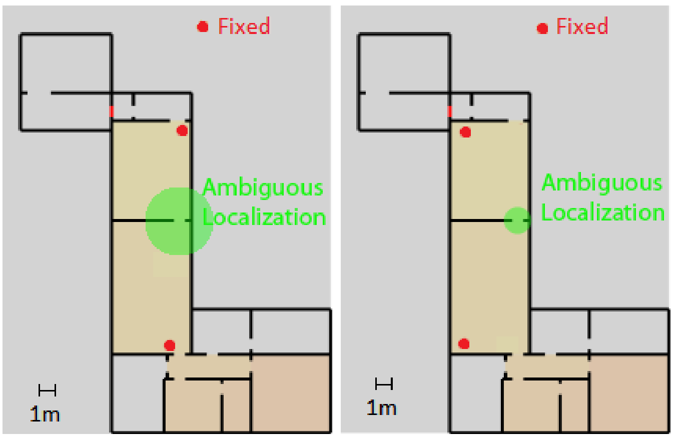


Fig. 14. Received signal strength over increasing distance

fixed nodes. The fixed nodes are placed 15m apart in each other's line of sight in the first test and out of each other's line of sight in the second test. A test person carrying the mobile devices moves from one fixed node to another in steps of one meter. After every step the person turns 360 degrees to check the localization's dependency of the orientation of the tag.



(a) Line-Of-Sight (b) Non-Line-Of-Sight

Fig. 15. Indoor Localization Results

In figure 15 the position of the fixed nodes are indicated by the dots and the area where the localization depends on the orientation of the tag by a highlighted area. As figure 15 shows there is a small area round the door where the localization performs poorly. Figure 15(b) shows this area is clearly smaller in the NLOS case than in the LOS case of figure 15(a). The reason for this is that when a fixed node in one room has a line of sight into another room, it's RF signals will propagate through that door, or other opening for that matter, without the attenuation caused by the walls. When the fixed nodes have no line of sight into the adjacent rooms, as in figure 15(b), the signals of the fixed nodes will attenuate when propagating through the walls or through the door after reflecting on the other walls.

VI. Multiple Device Binding

The concept of object binding can also be considered in two extra ways. The first one is tackling the issue of people wearing more than one tracked device. The second focusing on storage areas where dozens of tracked assets are stocked. In both cases the goal is to reduce objects visible on the client user interface [30].

The multiple device issue shouldn't so much be seen as a problem but as an improvement. If a person is carrying 3 devices, this means the server will calculate his position 3 times. This results in 3 coordinates, each with their own quality of location circle suggesting the area where the persons actual position is. Figure 16 below shows how the most likely area can be narrowed in 2 ways.

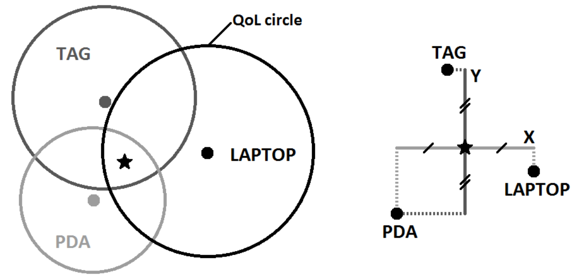


Fig. 16. Multiple device binding. On the left side by trilateration. On the right side using the average of only the outermost X and Y values.

On the left trilateration is used to merge the 3 coordinates. The quality of location (QoL) is used as the radius. QoL is a measure of how confident the OSL server is about it's calculated position. This technique should be the most accurate. On the right side another technique uses the minimum and maximum coordinate value of all points in each dimension to calculate the middle. Although this calculation needs less processing power, it is also the least accurate. Both show the concepts in only 2 dimensions, OSL calculates the position in 3. Another way is to take the average of each dimension. This technique should score between previous two techniques in terms of accuracy and uses about the same processing power as the second technique.

The second issue concerning the assets could be handled by adding a static location. When the assets are within range of the storage area, they can be snapped to the static location. This way coupling stored assets into 1 location will increase end-user data comprehensibility. The implementation for this can be done in roughly the same way as stated for the multiple device issue only there is no need to calculate an average position.

VII. Conclusion and Future work

In this paper, a method to realise dynamic object binding is presented. We choose Bluetooth to accomplish object binding because of its appearance in many mobile devices. For this project, the Bluetooth technology is fused with multiple other technologies in order to get an accurate localization system. Some real experiments were done to test the Bluetooth measurement model. These results showed room accuracy when only Bluetooth was used. Obstacles like walls have a big influence on the signal strength which will make it easier to achieve room-level accuracy. This information is incorporated in the Bluetooth measurement model.

Dynamic object biding is used to locate devices which cannot be located by any other technology but can discover other devices which are located by other means. Dynamic object binding can increase the likelihood of the position of these devices.

This paper shows that Wireless Sensor Networks can be used for localization purposes and how it is incorporated int the Opportunistic Seamless Localization framework. In indoor environments however the room-level localization is reasonably accurate depending on the location of the fixed nodes and the configured transmission power depending on the layout of the indoor environment.

Tests indicate that this Wireless Sensor Networks Proximity Localization system performs poorly in large outdoor environments, for outdoor use a GPS is recommended. For indoor environments however, exactly where this system is designed for, the localization seems to be pretty accurate. Also the position of the fixed nodes plays a role in the reliability of the locations yielded by the system. The ideal conditions are an indoor environment with rooms up to 20 m long, with fixed nodes placed in such a way so they have a limited line of sight into the adjacent rooms. The thicker the walls separating the rooms, the better the room-level localization will perform.

References

[1] I. De Cock, W. Loockx, M. Klepal, and M. Weyn, "Dynamic Object Binding for Opportunistic Localisation," in *UBICOMM 2010: The Fourth International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies*. Florence, Italy: IARIA, October 2010, pp. 405–411.

[2] J. Hallberg, M. Nilsson, and K. Synnes, "Positioning with bluetooth," in *10th International Conference on Telecommunications. ICT 2003*, 2003, pp. 954–958.

[3] M. Weyn, "Opportunistic Seamless Localization," Ph.D. dissertation, University of Antwerp, 2011.

[4] A. Varshavsky, E. de Lara, J. Hightower, A. LaMarca, and V. Otsason, "GSM indoor localization," *Pervasive and Mobile Computing*, vol. 3, no. 6, pp. 698–720, 2007.

[5] K. Langendoen and N. Reijers, "Distributed localization in wireless sensor networks: a quantitative comparison," *Computer Networks*, vol. 43, no. 4, pp. 499–518, 2003.

[6] A. Nasipuri and K. Li, "A Directionality based Location Discovery Scheme for Wireless Sensor Networks," in *Proceedings of the First ACM International Workshop on Wireless Sensor Networks and Applications*, 2002.

[7] S. Gezici, Z. Tian, G. V. Giannakis, H. Kobaysahi, A. F. Molisch, H. V. Poor, and Z. Sahinoglu, "Localization via Ultra-Wideband Radios: A Look at Positioning Aspects for Future Sensor Networks," *IEEE Signal Processing Magazine*, vol. 22, pp. 70–84, 2005.

[8] M. Weyn and M. Klepal, "OLS - Opportunistic Localization System for Smart Phones Devices," in *Mobile Phones: Technology, Networks and User Issues*. Nova, 2011.

[9] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, J. Jansson, R. Karlsson, and P. Nordlund, "Particle filters for positioning, navigation, and tracking," *IEEE Transactions on signal processing*, vol. 50, no. 2, pp. 425–437, 2002.

[10] J. Haartsen, "Bluetooth-The universal radio interface for ad hoc, wireless connectivity," *Ericsson review*, vol. 3, no. 1, pp. 110–117, 1998.

[11] S. Hay and R. Harle, "Bluetooth Tracking without Discoverability," in *Location and Context Awareness: 4th International Symposium, LoCA 2009 Tokyo, Japan, May 7-8, 2009 Proceedings*. Springer-Verlag New York Inc, 2009, pp. 120–137.

[12] J. Bray and C. Sturman, *Connect without cables*. Prentice Hall PTR Upper Saddle River, NJ, USA, 2000.

[13] A. Huang, "The use of Bluetooth in Linux and location aware computing," Ph.D. dissertation, Citeseer, 2005.

[14] Weidler, "Rugged Bluetooth Scanners," Motorola, Tech. Rep., 2010.

[15] S. Feldmann, K. Kyamakya, A. Zapater, and Z. Lue, "An indoor Bluetooth-based positioning system: concept, implementation and experimental evaluation," in *International Conference on Wireless Networks*, 2003, pp. 109–113.

[16] U. Bandara, M. Hasegawa, M. Inoue, H. Morikawa, and T. Aoyama, "Design and implementation of a bluetooth signal strength based location sensing system," in *2004 IEEE Radio and Wireless Conference*, 2004, pp. 319–322.

[17] J. Hallberg and M. Nilsson, "Positioning with bluetooth, irda and rfid," *Computer Science and Engineering, Luleå University of technology/2002*, vol. 125, 2002.

[18] I. Bylemans, M. Weyn, and M. Klepal, "Mobile Phone-Based Displacement Estimation for Opportunistic Localisation Systems," in *The Third International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM 2009)*. IEEE, 2009, pp. 113–118.

[19] B. A. Company, "Accurate real-time information on waiting times," 2010. [Online]. Available: http://www.brusselsairport.be/en/news/newsItems/361700

[20] J. Hightower and G. Borriello, "A survey and taxonomy of location systems for ubiquitous computing," *IEEE Computer*, vol. 34, no. 8, pp. 57–66, 2001.

[21] D. Niculescu and B. Nath, "DV based positioning in ad hoc networks," *Telecommunication Systems*, vol. 22, no. 1, pp. 267–280, 2003.

[22] A. Savvides, H. Park, and M. B. Strivastava, "The bits and flops of the n-hop multilateration primitive for node localization problems," in *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*. ACM, 2002, p. 121.

[23] C. Savarese, J. Rabaey, and K. Langendoen, "Robust positioning algorithms for distributed ad-hoc wireless sensor networks," in *USENIX technical annual conference*, vol. 2. Monterey, CA, 2002.

[24] G. Zanca, F. Zorzi, A. Zanella, and M. Zorzi, "Experimental comparison of RSSI-based localization algorithms for indoor wireless sensor networks," in *Proceedings of the workshop on Real-world wireless sensor networks, April*, 2008, pp. 01–01.

[25] N. Patwari, R. O'Dea, and Y. Wang, "Relative location in wireless networks," in *Vehicular Technology Conference, 2001. VTC 2001 Spring. IEEE VTS 53rd*, vol. 2. IEEE, 2002, pp. 1149–1153.

[26] C. Liu, K. Wu, and T. He, "Sensor localization with ring overlapping based on comparison of received signal strength indicator," in *Mobile Ad-hoc and Sensor Systems, 2004 IEEE International Conference on*. IEEE, 2005, pp. 516–518.

[27] K. Lorincz and M. Welsh, "MoteTrack: a robust, decentralized approach to RF-based location tracking," *Personal and Ubiquitous Computing*, vol. 11, no. 6, pp. 489–503, 2007.

[28] J. Blumenthal, R. Grossmann, F. Golatowski, and D. Timmermann, "Weighted centroid localization in Zigbee-based sensor networks," in *Intelligent Signal Processing, 2007. WISP 2007. IEEE International Symposium on*. IEEE, 2008, pp. 1–6.

[29] "ZigBit Development Kit 2.3 User Guide," ZigBit, Tech. Rep., October 2008.

[30] R. T. Ng and J. Han, "Efficient and effective clustering methods for spatial data mining," p. 12, 1994.