Indoor Location Estimation in Sensor Networks using AI Algorithm

József Dániel Dombi
Department of Software Engineering, University of Szeged, Árpad tér 2., Szeged, Hungary

Keywords: Indoor Location and Tracking, Sensor Network, Fingerprinting, Machine Learning.

Abstract: To determine the indoor location of a person or object, we can use a suitable wireless network. There are different kinds of wireless networks available for this. Independent of the type of the network, using RSSI it is possible to find the position of the moving person close by. Here, we present Wireless Sensor Network and apply it in a real environment. We will mainly concentrate on locating a person using standard artificial intelligence methods. In our system we define nodes (the fingerprint), and supervised learning algorithms that should predict these nodes. In addition, we test whether we can get nice results if we change the granularity of the nodes. Real simulation demonstrates that this system can supply the current position of the moving person with good accuracy.

1 INTRODUCTION

Here, a locating system is used for tracking and defining the current position of a person or object. The most important distinguishing feature of such a system is the type of wireless communication used, and the application information presented to the user. The granularity of the position can vary from one application to another. For example, finding out whether a person is in a room requires less information, while locating a person who is sitting in front of a desk requires more accurate information.

Therefore, many different systems and technologies have been proposed. GPS devices are available for everyday use in modern outdoor applications (Enge and Misra, 1999). The GPS system has a limited accuracy, and can be used where satellites are ”visible”, because buildings block the GPS transmissions. The earliest investigation for indoor positioning was done by Bahl et al. who observed that an RF signal source exhibits spatial variation, but is consistent in time. They created a system called Radar (Bahl and Padmanabhan, 2000). They used four 802.11 access points to locate a laptop at its true position to an accuracy of 2-3 meters. Since then, there have been a lot of improvements in Radar's fingerprint matching algorithms (Agrawala and Shankar, 2003) (Haeberlen et al., 2004) (Ladd et al., 2005).

These studies showed that the Received Signal Strength Indicator (RSSI) has a larger variation because it is subject to the detrimental effects of fading and shadowing.

Other techniques, such as Active Badge (Hopper et al., 1993) and a commercial system like Versus (Versus, 2012), use infrared emitters and detectors to achieve an accuracy of 5-10m. Active Bet (Harter et al., 1999) (Ward and Jones, 1997) and Cricket (Priyantha et al., 2000) combine the RF and ultrasound signal to estimate the distance. These systems have accuracies ranging from a few meters to a few centimeters. In a commercial system (Ubisense, 2012), ultra-wideband emitters and receivers have been used to realize indoor locations.

In this study we use a wireless sensor network. If a large number of sensors are deployed, the network can monitor large areas. We can apply a sensor network in a variety of situations like those for monitoring the environment. Sensor nodes can measure temperature, a heartbeat, humidity and so on. However, collecting a large amount of data leads to an increase in traffic and in the energy consumption of sensors. Moreover, increasing the data collection time has a negative impact on the location data collection method. In a wireless sensor network it is vital to keep the energy consumption low. Our Sensor Network protocol is similar to the ZigBee (ZigBee, 2012) protocol, which includes IEEE 802.15.4 for MAC and PHY. Here, we implemented a positional estimation technique based on standard artificial intelligence methods using RSSI in a sensor network and evaluated its position-estimation ability. The remainder of this paper is organized as follows. Sec-

Dombi J..
Indoor Location Estimation in Sensor Networks using AI Algorithm.
DOI: 10.5220/0004098303490352
In *Proceedings of the 14th International Conference on Enterprise Information Systems* (ICEIS-2012), pages 349-352
ISBN: 978-989-8565-10-5

tion 2 outlines the standard AI methods, then Section 3 describes the experimental setup. After, Section 4 presents the result of our experiments. In the last section, we summarize our findings and draw some pertinent conclusions.

2 AI METHODS

In our system the signal strengths are got by a router. Currently, different routers send the RSSI to the PC. More than three RSSI values are used to determine the position of the node inside the building. First, we have to investigate the relationship between the distance and signal strength from a given router point. If one knows the distances from a node to at least three different routers, one can calculate the position of the node in the system.

In a real environment the power received is a very complex function of distance. Even if a good model is available to determine the position of the node, it still requires a lengthy calculation. Hence, the location of the RSSI is more complicated and it is harder to solve. In our model, we simplify the system. We do not worry about calculating the exact position of the object. For us, it is sufficient to determine the nearest node (fingerprint).

Standard artificial intelligence methods offer a good solution for estimating the location and reducing the distance error. Here, we implemented the decision tree model and neural network model. An exact knowledge of the position is not required by either method. We can train and use the methods without asking for it. Both methods have good classification capabilities and are suitable for our purpose, where we wish to determine the location that best matches the observed signal strength data.

Using a decision tree means we have to generate all possible decision trees that correctly classify the training set and then choose the simplest one. The number of such trees is finite, but very large. One of the most widely used decision tree method is ID3 (Quinlan, 1986). It constructs the simple decision tree, but this approach cannot guarantee that better trees have not been overlooked. The basic structure of ID3 is iterative. The window, which is a subset of a training set, is chosen at random and a decision tree is formed from it. ID3 examines all candidate attributes and chooses attribute A to maximize the gain. This tree correctly classifies all objects in the window. All other objects are then classified using the tree. If the tree returns the correct answer for all these objects, it is then correct for the entire training set and the process terminates. If not, a selection of the incorrectly classified objects is added to the window and the process continues. Recent articles (Yim, 2008) have examined how a decision tree works in a location system based on a fingerprint, and it is found that the accuracy of the decision tree is no worse than a Neural Network or Bayesian system.

A neural network is capable of representing the relationship between the inputs (signal strengths) and outputs (nodes). The learning strategy should calculate the free parameters of the model (also called the ”weights” of the network). Here, the standard multilayer perceptron (MLP) is implemented. The architecture of MLP is organized as follows: the signals flow sequentially through the different layers from the input to the output layer. For each neuron, it first calculates a scalar product between a vector of weights and the vector given by the output of the previous layer. A transfer function is then applied to the result to produce the input for the next layer. A commonly applied transfer function is the sigmoid function. In a single hidden layer, if the number of hidden layers is sufficiently large then any continuous function can be approximated to some desired accuracy. Roberto Battiti et al. (Battiti et al., 2002) examined how a neural network might be used to locate an object. They found that with MLP it is possible to determine the position of the person within 1.82 meters.

In our study, we compare the performance of both methods to see how well they determine the location of an object in a sensor network environment.

3 EXPERIMENTAL SETUP

Our experimental testbed is located on the first floor of a 2-storey building. We define nodes (position of the fingerprint), and the distances between the nodes are equal, namely a distance less than 2 m. Part of the layout of the floor and position of fingerprint are shown in Figure 1. In the tests, we employed a special type of sensor network called RTLS (RTLS, 2012). We placed four routers per room and two on the corridor at the locations indicated in Figure 2.

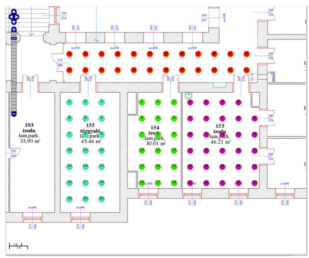

Figure 1: Map of the floor and position of each fingerprint.

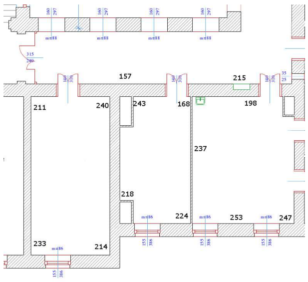

Figure 2: The position of routers inside the building.

In this figure, routers are represented by numbers (i.e.: 211, 240, 243, 168, etc.) and we see that the position of each router is usually in the corner of a room. Users wear a transmitter (also called a tag) device on their wrist as a watch (see Figure 3, which has a unique ID called the address.

Figure 3: Two different kinds of watch. Both of them function as a sensor.

The tag can measure and transmit the temperature and battery level; and, of course, the routers can measure the RSSI value. It is also possible to send audio data through this sensor network.

By default, the tag will send a broadcast message every 4 seconds. When a router gets a message it can transmit this data to the coordinator (zero in Figure 4, a special router). The packet received by the coordinator contains the address of the measured tag, the RSSI value measured by the router, and any other data measured by the tag. In this network there is a time delay in the routers. The routers wait for a while to receive RSSI values, then they aggregate them and transmit this data to the coordinator as a single packet. There is a size limit of the packet so in this way the router should be able to send a packet to coordinator every 400 milliseconds. The aggregated package contains only the latest RSSI value received from the tag. As we mentioned above, the coordinator can receive packets from different routers and it forwards them to a PC. The program running on this PC can collect the RSSI values. The primary task of the program is to determine which measurement belongs to the given tag at any one time. Our network is self-organized. This means that a tag can communicate with the routers and these routers send the received information on to the next router, which is closer to the coordinator and is connected to the PC. A tag tries to reach the nearest router, and if it cannot communicate with this router then it will search for another router. Figure 4 shows how communication is established and maintained. In this figure we can see that there is coordinator (C1), which is connected to the PC and there are six routers and a Tag (E1).

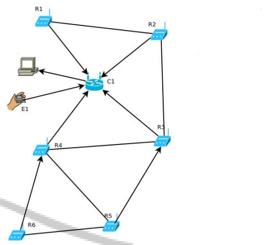

Figure 4: Communication between routers and tags.

We generated a fingerprint by performing calibration measurements. For each node, we measured over 20 values in a second and stored the RSSI values obtained by the router. These were the reference values that were used for testing the system.

4 RESULTS

As mentioned previously, we tested the AI methods on a first floor of a building. We positioned the routers and coordinator. First we had to collect samples and then we used the cross-validation method. This method partitions a given data sample into complementary subsets. Then we performed an analysis on one subset (called the train set), and validated our analysis on the other subset (called the test set). Multiple rounds of cross-validation were performed using different partitions, and the average over the rounds was the result of the validation.

It should be noted that the objective of our training algorithm was to build a model with good generalization capabilities when it was tested with values not present in the train set. The number of parameters and the length of the train phase determined the goodness of the generalization.

In a real environment it may happen that the given tag cannot reach the router (missing value). In that case, we define the worst RSSI value. In addition we define a new attribute that contains this information. When the value is one, the router receives a signal, and when the value is zero, the router doesn't receive an RSSI value of the given time. The maximum value of RSSI that we measured was -54dBm and minimum value was -90 dBm. For each measurement, out of 14 routers 6 on average send a message saying that they receive an RSSI value, and only 3 routers on average can measure valuable RSSI values - which means that they can measure values better than -85 dBm. We created different kinds of tests which varied the granularity of the nodes: single position, triple position, and the room. Single position means that we would like predict the current position of the object. Triple position means that we aggregated 3 nearest node values into one, and we tried to predict this new position. In this case, we were only interested in locating the object in a certain part of the room. Room position means that we merged all the node values in the room into a single node in order to locate the object. The results are shown in the following table.

Table 1: The results of the methods.

Granularity of nodes	Decision tree	Neural network
Single Position	38%	40%
Triple position	65%	53%
Room	91%	89%

As we see, the two methods have a similar performance in most cases. The percentage value tells us the degree of certainty of location an object. We tried different kinds of parameter input for the two learning methods and we obtained similar results. In the decision tree, we get the whole tree and examine the decisions. The decision tree has an average size of 250 and an average number of leaves around 125. The time needed for the learning method and the evaluation of the values is less for a tree than that for a neural network.

5 CONCLUSIONS

Many indoor positioning methods have been published that can be used in a variety of situations. For any kind of wireless network, the fingerprint method is the most commonly used approach. Previous studies showed that AI algorithms can perform well in locating an object. These studies used different types of networks. In this paper we compared two different kinds of AI method in a wireless sensor environment, which is similar to the ZigBee network. The bandwidth of this network is very low, but it can transmit audio and data measurements in real time with just one radio chip. We carried out different kinds of tests using this wireless sensor network, and we discovered that in most cases the decision tree and neural network approaches have a similar performance. When we increase the granualty of the nodes, we get much better results in terms of accuracy.

ACKNOWLEDGEMENTS

The study presented here was supported by the Hungarian national grant GOP-1.1.1-07. I would like to thank Ákos Kiss for his valuable advice, and also Péter Kenderesi, Péter Molnár and Balázs Szabó for providing position data.

REFERENCES

Agrawala, A. K. and Shankar, A. U. (2003). WLAN location determination via clustering and probability distributions. In *PerCom*.

Bahl, P. and Padmanabhan, V. N. (2000). RADAR: An in-building RF-based user location and tracking system. In *INFOCOM*.

Battiti, R., Villani, A., Villani, R., and Nhat, T. L. (2002). Neural network models for intelligent networks: Deriving the location from signal patterns. In *Citeseer*.

Enge, P. and Misra, P. (1999). Special issue on gps: The global positioning system. In *Proceedings of the of the IEEE*.

Haeberlen, A., Flannery, E., Ladd, A. M., Rudys, A., Wallach, D. S., and Kavraki, L. E. (2004). Practical robust localization over large-scale 802.11 wireless networks. In *Proceedings of the 10th Annual International Conference on Mobile Computing and Networking*.

Harter, A., Hopper, A., Steggles, P., Ward, A., and Webster, P. (1999). The anatomy of a context-aware application. In *Proceedings of the Fifth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom-99)*.

Hopper, A., Harter, A., and Blackie, T. (1993). The active badge system. In *INTERCHI'93 Conference on Human Factors in Computing Systems*.

Ladd, A. M., Bekris, K. E., Rudys, A., Kavraki, L. E., and Wallach, D. S. (2005). Robotics-based location sensing using wireless ethernet. *Wireless Networks*.

Priyantha, N. B., Chakraborty, A., and Balakrishnan, H. (2000). The cricket location-support system. In *MOBICOM*.

Quinlan, R. (1986). Induction of decision trees. *Machine Learning*, 1.

RTLS (2012). Rtls. http://www.rtls.eu.

Ubisense (2012). Ubisense. http://www.ubisense.net.

Versus (2012). Versus. http://www.versustech.com.

Ward, A. and Jones, A. (1997). A new location technique for the active office. In *IEEE Personal Communications*.

Yim, J. (2008). Introducing a decision tree-based indoor positioning technique. *Expert Syst. Appl*, 34(2).

ZigBee (2012). Zigbee. http://www.zigbee.org/Specifications.aspx.