

# Neural Network-based Indoor Localization in WSN Environments

Laslo Gogolak<sup>1</sup>, Szilveszter Pletl<sup>2,3</sup>, Dragan Kukolj<sup>3</sup>

<sup>1</sup> Subotica Tech, Department of Automation, Subotica, Serbia

<sup>2</sup> University of Szeged, Department of Informatics, Szeged, Hungary

<sup>3</sup> Faculty of Technical Science, Univ. of Novi Sad, Novi Sad, Serbia  
gogolak@vts.su.ac.rs, pletl@inf.u-szeged.hu, dragan.kukolj@rt-rk.com

---

*Abstract: With the advancement of wireless technology even more wireless sensor network (WSN) applications are gaining ground. Their field of application is increasingly widening. This paper examines the WSN application which allows indoor localization based on the Fingerprint (FP) method. The communication between the modules was monitored during the experiment whereby the received radio signal strength indicator (RSSI) values from 5 modules were recorded by a mobile sensor. The received data was used for training of the feed-forward type of neural network. Through use of the trained neural network and the measured RSSI values an indoor localization was realized in a real environment. The neural network-based localization method is analyzed applying the cumulative distribution function (CDF). For the reference model the well-known weighted k-nearest neighbour (WkNN) method was used.*

*Keywords: Fingerprint localization; WSN; Received Signal Strength; Neural Network; Mobile sensor*

---

## 1 Introduction

Most technical solutions would not be available without localization processes, even transport is unimaginable nowadays without a navigation device. GPS localization devices can achieve up to 2 cm accuracy in localization for each spot in the world. For GPS localization one has to be in an open space and hold the appropriate device. There are various localization methods, but actually the GPS systems offer the simplest, cheapest and the most accurate outdoor localization technologies. In fact, indoor localization presents the bigger challenge. Although there are numerous applications and methods for indoor localization, none of those are appropriate and accurate enough. Applying different technologies, such as RF, optical, infrared, ultrasound, determination of positions is possible to certain accuracy [16, 17]. Information about an indoor position mostly can be obtained by

processing an RF signal sent from known positions. This localization methodology is called the network fingerprinting method [1, 2, 7 and 19]. In this work an indoor localization method is presented using Wireless Sensor Network – commonly known as WSN. There are many solutions for the localization algorithms. In case of the solution offered by Kannan, Guoqiang Ma and Vucetic [15], simulated annealing algorithm was used for the localization. They use the measured distance between the fixed sensor mote (anchors) and the mobile mote for the localization.

In this work the fingerprint method of indoor localization using feed-forward neural network is presented. The WSN sensor modules were placed on fixed positions (anchor motes) in an experimental room. The room is a furnished classroom with computers and glass windows, which is not an ideal environment in terms of the spread of RF signals. A mobile sensor module is part of the system, and with the help of this mobile sensor the RSSI (Received Signal Strength) values between this sensor and the anchor motes can be measured. The RSSI measurements are done in predefined positions with 0.6 m resolution. The received data are stored in a database. This set of data is used to find some useful correlation between the alteration of RSSI values and the position of the mobile mote. This work shows a solution for indoor localization based on trained neural network and created database RSSI measurements. The weighted k-nearest neighbor (WkNN) method is used for the results comparison [18]. While the recording of the RSSI values is done in real environment, the localization is done by software implemented neural network.

## **2 Wireless Sensor Network**

The current trends make the wireless techniques very popular in various fields of application. This low cost technology is available for any purpose. Thanks to the modulation techniques, the quality of the RF transmission is very high. Nowadays the wireless sensor networks are successfully used for, for instance, forest fire detection, monitoring of numerous agricultural microclimates [3, 6] and traffic systems [9].

### **2.1 The Crossbow Iris Sensor Mote**

For the experimental purposes, the wireless sensor motes with weak resources were needed, therefore the Crossbow's IRIS wireless sensor motes were chosen. These sensor modules are compliant with the IEEE 802.15.4 standard called ZigBee. The IRIS sensor modules are used for measuring the RSSI values during the experiment. The basic elements of the sensor motes are the single 8 bit low power Atmel's microcontroller and the ZigBee protocol based wireless module. These microprocessors are from the Atmel's series 1281 which are working at 8

MHz. The processor contains a 128 Kbytes program flash memory, an 8 Kbytes RAM, a 4 Kbytes configuration EEPROM, and some communication interfaces (UART, ADC, digital I/O, I2C, SPI), whose performance classifies it as a microcontroller with thin capabilities. The third part of the system is the logger flash which is a 512 Kbytes memory, which stores the measurement data. The communication between the sensor modules is done by using the above mentioned 2.4 GHz RF modules. The ZigBee standard communication range in the outdoor environment is up to 500 *m*, but in an indoor environment it is up to 50 *m*. The Iris sensor motes are supported by the open source TinyOS operating system which is programmed in the NesC programming language. The most significant characteristic of the Iris wireless sensor systems with TinyOS is that they can work in the network mode. They can communicate and they can be programmed through the self-organizing ad-hoc Xmesh network. For this work the authors used the above mentioned platform during an experimental setup.

## 2.2 Measuring the RSSI Values

The radio frequency transceiver is the most important component. It is a high performance RF-CMOS 2.4[GHz] radio transceiver targeted at IEEE 802.15.4 applications. This transceiver chip contains some special IEEE 802.15.4-2003 hardware support such as RSSI computation. The RSSI value is a 5-bit value which is updated every 2  $\mu$ s. It indicates the received signal power in the selected channel in steps of 3 dB. This value can be in the range of 0-28. The RSSI value of 0 indicates a radio frequency input power of less than -91dBm.

For an RSSI value in the range of 1 to 28, the radio frequency input power can be calculated by following equation:

$$P_{RF} = \text{RSSI\_BASE\_VAL} + 3 \cdot (\text{RSSI} - 1), \quad (1)$$

where the  $P_{RF}$  is the radio frequency input power, the  $\text{RSSI\_BASE\_VAL}$  is equal to -91 dBm.

## 2.3 The Measuring Software

### 2.3.1 The Mobile and the Anchor Node Software

The mobile node and the anchor nodes have the same software for technical reasons. In case of receiving a control message, the node starts measuring the RSSI values. This is done by the following steps: the node receives a message which contains a command to perform  $n$  measurements with a node that has an ID of  $x$ . It first sends a query message to the  $x$  node, waits for the answer, collects the results and broadcasts it, so that the mote sending the control message can receive it. The mobile node repeats it  $n$  times. Furthermore, when an anchor node receives a query message, it performs a measure on the incoming message, then sends it

back. The mobile node operates in the same way upon receiving this message. In the experiment, the value of  $n$  was set to 100. The measurement algorithm is shown in Figure 1.

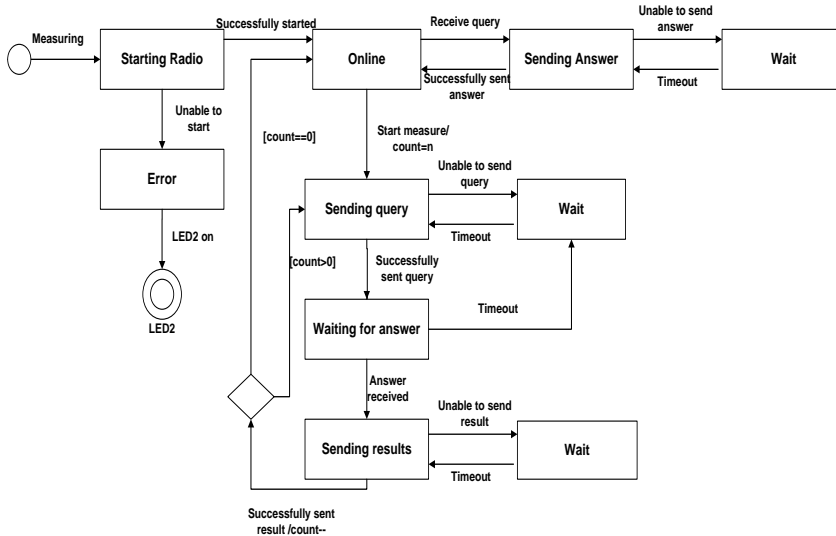


Figure 1

Block diagram of the measuring algorithm

### 2.3.2 The Base Station Node Software

The software of the base station is very simple, it works only as a gateway to the computer. The user has GUI on the control computer with two text inputs for the location and a button to start the measurement. This program can be manually configured as to which nodes to test (values of  $x$ ) and how many times (value of  $n$ ) along with the timeout. When pressing the button, the program periodically sends messages to the mobile node (through the base station) and waits for the results. After the timeout had been reached, it sends the next message. This prevents the deadlock of the system. In the experiment, the nodes IDs 1, 2, 3, 4 and 5, were tested 100 times, with a 3 seconds timeout. However, there were positions where the signal of an anchor node was too weak.

### 2.3.3 The Package Format

Four different types of messages are used by the system, all of them based on the default radio message format of the TinyOS system. This framework adds a header to the messages (containing the sender and the target), the length of the message and other metadata. It also handles the *acknowledge* messages. The four user-defined messages are the *control* message, the *query* message, the *answer* message, and the *result* message. The format of four user-defined messages is shown in Table 1.

Table 1  
The format of the message packet

Byte	1	2	3	4	5	6
<b>controlMsg</b>	Source		target		count	
<b>queryMsg</b>	Id					
<b>answerMsg</b>	Id	Sig				
<b>measureMsg</b>	Time				signal1	signal2

The *control* message is designed to be the system message. Currently, the only role it has is to start the measurement. It is sent to the mobile node by the anchor node. The *query* message is sent by the mobile node to an anchor. It only contains one unsigned byte, the ID of the message (*id*) to identify each conversation. As a reply, the anchor sends an *answer* message which contains the message ID (*id*) and another unsigned byte for the measure (*sig*). The anchor node calculates this value when it receives the *query* message using a built-in RSSI measure module which provides the RSSI value of each incoming package. The mobile node generates a result message upon receiving the *answer* message. The *answer* message contains a 32 bit timestamp (*time*, the local time of the node, only for test purposes), the signal strength received from the anchor node (*signal1*) and the signal strength of the *answer* message (*signal2*) determined in the same manner as the anchor node. The output of the measuring process is an XML file which contains the count of the measurements and values (the “count” attributes, typically 500), and every measurement has the following format:

- “time” is the time of arrival of the radio packet to the mobile node
- “mote1” is the mobile node
- “mote2” is the anchor node
- “signal1” is the RSSI measured at the mobile node
- “signal2” is the RSSI measured at the anchor node

The typical size was about 69 KB per measurement. The identifiers of the XML documents depend on the measurements due to the simplification of the measurement database.

### 3 Experimental Measuring and Database Preparation

In the present work the authors use neural network for localization purposes. As in all neural network training algorithms, this application also requires a database for training and testing of the network [4, 5]. The database in this particular experiment is prepared by the measuring of the RSSI values for each  $r_j = [x_j \ y_j]$  coordinate by the sensor motes. The location vector  $[x_j \ y_j]$  denotes the position of

the  $j^{\text{th}}$  reference point. The experimental room where the measurements were done is a research laboratory. The room contains standard laboratory furniture and equipment such as tables, chairs and computers. The structure of the room is a very important factor in case of indoor localization [8]. The radio signals are very sensitive to reflections. The number of windows in the classroom affects the accuracy of the localization. In this experiment the aim was the realization of the fingerprint localization in a non-structured real environment with iron grid-protected windows and with all the interfering factors of the room.

### 3.1 The Process of the Experiment

The experimental room structure consists of  $24 \times 10$  measurement points (Figure 2 - black points) where the RSSI values for the fingerprint are measured. In Figure 2 all types of points are presented. The distance between the points, the grid resolution is 0.6 m. In the grid system there are 5 sensor motes, and anchors (see the red points in Figure 2), which are in fixed positions. The third type of positions (see the green points in Figure 2) are the unknown positions which are between the measurements points. They will be used for testing the accuracy of the positioning algorithm.

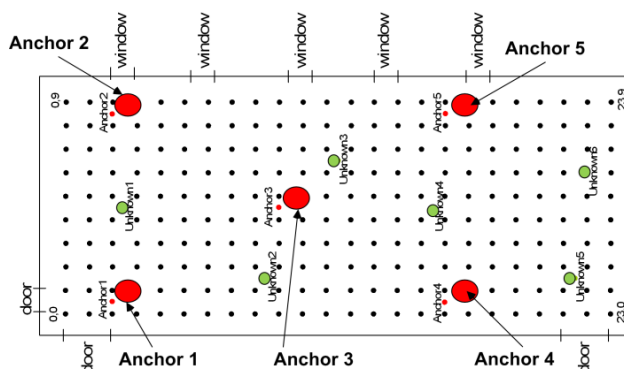


Figure 2

Layout of the experimental setup

During the experiment one measurement is done at every  $24 \times 10$  (240) position using a mobile sensor mote. The RSSI values are measured between the mobile mote and between the anchor motes. The measurements are done in two directions. The mobile mote measures the RSSI value of the anchor sensor mote and vice versa. In order to obtain satisfactory accuracy, measurement with one anchor mote is repeated 100 times. In the presented experiment there are 5 anchors. For all positions we have  $5 \times 100$  measurements in both directions. Therefore, at the end of the recording, the authors had 12000 measured values for the mobile-anchor links and 12000 measured values for the anchor – mobile links. All of the data are stored in the centralized database for further analysis.

### 3.2 Data Preparation

The calculation of RSSI values was described in Section 2.2. Due to performance of the hardware platform the highest measured value was in interval from 4 to 16 units. This means the resolution of the RSSI values was only 12 units. This resolution defines the accuracy of the localization results. Unfortunately, the RSSI values can only be measured at resolution dictated by the Iris sensor module. The arrangement of the experimental room is not ideal for the wireless signal distribution. The typical distribution of the RSSI values is shown in Figure 3 for the first 4 anchor nodes. From the sample of 100 measurements there were cases when few were invalid. This error occurs because the RSSI values cannot be measured properly. The invalid RSSI values are replaced by the mean values of the neighbouring grid point values. For all positions from the 100 measurements an additional database is created. This database consists of the mean value, median value and standard deviation for every 24 x 10 position and for all the 5 anchors. In fact, this database is used for creating the neural network, whereby 50% of the values are used for training the neural network, 25% of the values are used for the validation and 25% for testing the accuracy of the neural network.

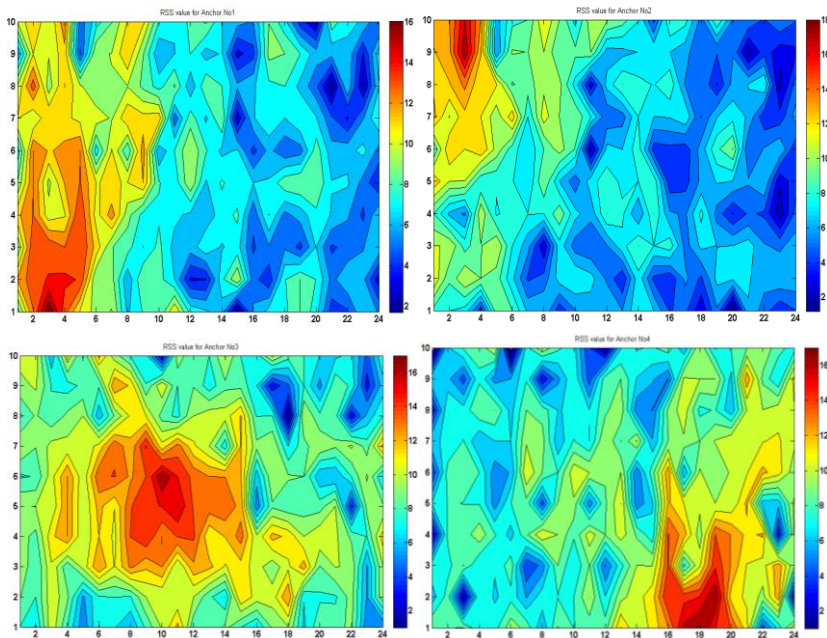


Figure 3

The map of RSSI values for the four anchor nodes

## 4 Weighted $k$ – Nearest Neighbor Algorithm

The weighted  $k$ -nearest neighbour algorithm is widely known clustering method [10] and can be efficiently applied for fingerprint localization. Assuming that the radio map database of the reference points of RSSI vectors  $S \in R^m$  exist, and  $S_t \in R^n$  measurements linked to the tracked node are performed, one may indicate the similarity in signal strength vectors of the  $j^{\text{th}}$  reference point  $r_j = [x_j, y_j]$  and the wireless sensor with unknown location. The Euclidian distance  $p_j$  is defined as

$$p_j = \sqrt{\sum_{i=1}^n (S_{ti} - S_{ji})^2} \quad (2)$$

determines the similarity between  $i^{\text{th}}$  current and  $j^{\text{th}}$  reference signal strength vectors, and If the  $j^{\text{th}}$  reference point is closer to the node, the similarity value  $p_j$  is smaller. That is to say, it may be assumed that the closest reference point would have the most similar signal strength vector to the signal strength vector of the tracked node. The algorithm has a parameter,  $k$  which affects the accuracy of the method. This parameter also needs to be determined. In this case  $k$  refers to the number of reference points. The following step of the algorithm is finding the  $k$  nearest reference points  $v_j$  with the smallest Euclidean distances  $p_j$ ,  $j=1, k$ . A weighting factor  $w_j$ ,  $j=1, k$  is associated with the  $k$  reference points with the most similar RSSI vectors to the RSSI  $S_t$  measured at the tracked wireless sensor, based on their  $k$  smallest calculated Euclidean distances from that node, and calculated according to Equation (2). The weighting factors are inversely proportional to the square of the Euclidean distance, as shown by the Equation (3).

$$w_j = \frac{1/p_j^2}{\sum_{i=1}^k 1/p_i^2} \quad (3)$$

The unknown coordinates of the tracked wireless sensor are estimated by the equation  $r_x = \sum_{i=1}^k w_i r_i$ . Finally the estimation of the coordinates is completed. Consequently, this method is called the weighted  $k$ -nearest neighbor algorithm (WkNN).

## 5 Neural Network for Position Determination

This section presents a neural network clusterization method for determination of the position of mobile sensor node. This method has been chosen because in this case the recorded RSSI database can be used for supervised learning of the neural network [11, 12]. The type of the neural network is feed-forward. The implementation of the Levenberg-Marquardt (LM) training algorithm, which is chosen for the training of the neural network, was written in Matlab software. The



Levenberg-Marquardt training algorithm is the fastest and most efficient training method, but it requires a significant amount of working memory [13, 14]. In case of applying feed-forward three-layered neural networks, the number of neurons used in the hidden layer and also the type of the activation functions used in the neurons are both significant free parameters. During the simulations, the best results were achieved by using the “tansig” type of activation functions in the hidden layer, and “purelin” type in the input and output layers. The chosen structure of the used neural network is presented in Figure. 4.

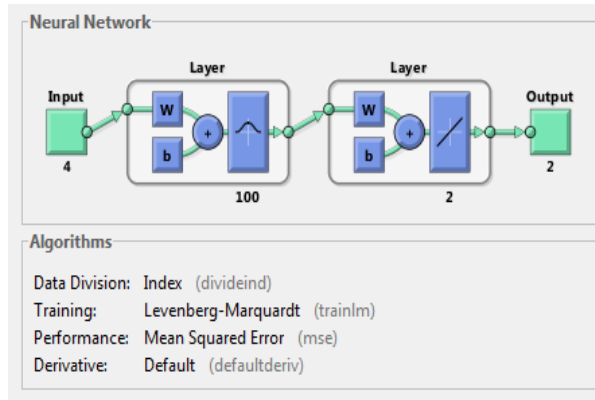


Figure 4

The Structure of the neural network in case for the four inputs (RSSI values from four anchor)

The neural network has 5 inputs and 2 outputs. The measured RSSI values from the 5 anchors are the input, while the outputs, the  $x$  and  $y$  respectively denote the coordinates. The structure of the matrix of the RSSI values is the following:

$$S = \begin{bmatrix} S_{11} & S_{1i} & \cdots & S_{15} \\ S_{j1} & S_{ji} & \cdots & S_{j5} \\ \vdots & \vdots & \ddots & \vdots \\ S_{N1} & S_{Ni} & \cdots & S_{N5} \end{bmatrix} \quad (4)$$

where  $S_{ji}$  denotes the RSSI perceived  $i^{\text{th}}$  anchor,  $i \in (1,5)$  at the  $j^{\text{th}}$  reference point. The output coordinates are used in the matrix form shown below:

$$r = \begin{bmatrix} x_1 & y_1 \\ x_j & y_j \\ \vdots & \vdots \\ x_N & y_N \end{bmatrix} \quad (5)$$

As it has been mentioned previously, the second database is used for the neural network training, where the mean, median and standard deviation values are listed. There are some other measurements apart from this database, whose coordinates are beyond the grid resolution. These points are the real test samples, for these

coordinates the neural network provides interpolated values as there are no RSSI values for the training. In the process of training and testing the neural network it can be seen that the accuracy and the speed of the training process depends on the numbers of hidden neurons. The following section demonstrates dependence of the accuracy of the suggested neural network based on the number of the hidden neurons.

## 6 Test Results

The results of the experiments using the neural network trained by the LM method are compared to the results given by the weighted  $k$  nearest neighbor (WkNN) algorithm. To represent the similarity between the estimated position of the tracked wireless sensor and its true coordinates, the Euclidean distance error was used as the basic performance metric, which represents the distance between the estimated position of the tracked wireless sensor and its true coordinate. Furthermore, the cumulative distribution function (CDF) of the distance error of all location estimates for all measurements fully describes the estimation characteristics.

During the process of testing, the “unknown points” mentioned in Section 3.1 were used for the accuracy of localization evaluation (see Figure 2 – green points). Since these points are positioned between the grid, their RSSI values are not found in the localization database.

Firstly, the localization accuracy of the neural network and WkNN method is compared. The different figures show the accuracy of the localization depending on how many anchor motes and their RSSI values were used during the localization process. Figure 5 shows the localization accuracy of the designed neural network and the WkNN methods where the RSSI values of the all five anchor motes have been used.

Figure 6 shows the 3 different structures of both methods. In the case of the neural network, HN100, HN200, HN300 refer to the number of 100, 200 and 300 hidden neurons, respectively. In the case of the WkNN method the CDF function is shown depending on the number of reference points ( $k=1,3,5$ ).

As it can be seen, when all the 5 anchor motes were applied, all the versions of the WkNN provided a more accurate localization. The WkNN achieved the highest accuracy when 4 anchors were applied in the localization process. These anchor motes were placed in the four corners of the experimental room. These were the anchors 1, 2, 4, and 5. This case is shown in Figure 6.

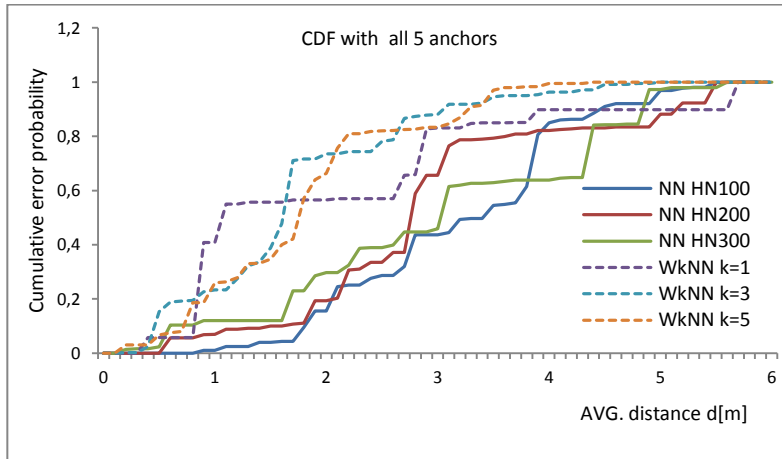


Figure 5

Cumulative distribution functions for distances in case of 5 anchors

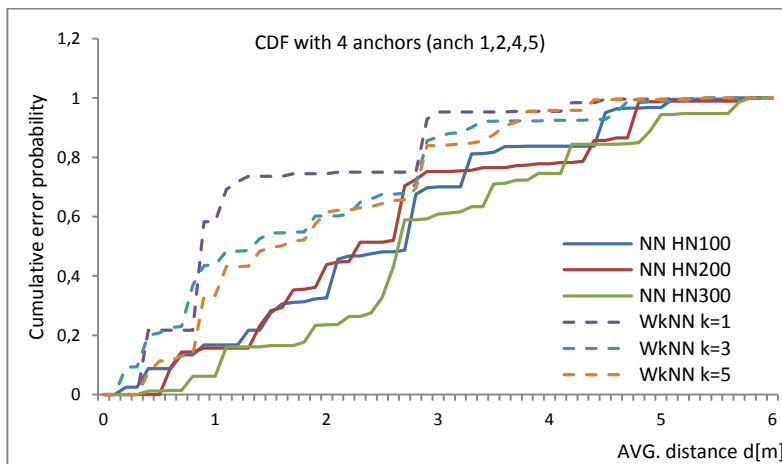


Figure 6

Cumulative distribution functions for distances in case of 4 anchors

Figure 6 shows the increase of the accuracy of the neural network too, whereby the number of neurons has not much of an effect on the accuracy. However, the highest accuracy in localization was achieved by the WkNN method with  $k=1$ .

Figure 7 shows the CDF function when only 3 anchor nodes were used for localization. These sensor nodes are anchors 1, 2, and 5. These are the nodes which divide the experimental room diagonally. In this case the accuracy of the localization with the WkNN method significantly decreases. Consequently the neural network has an advantage in this case, because it can provide a more accurate localization than the WkNN method.

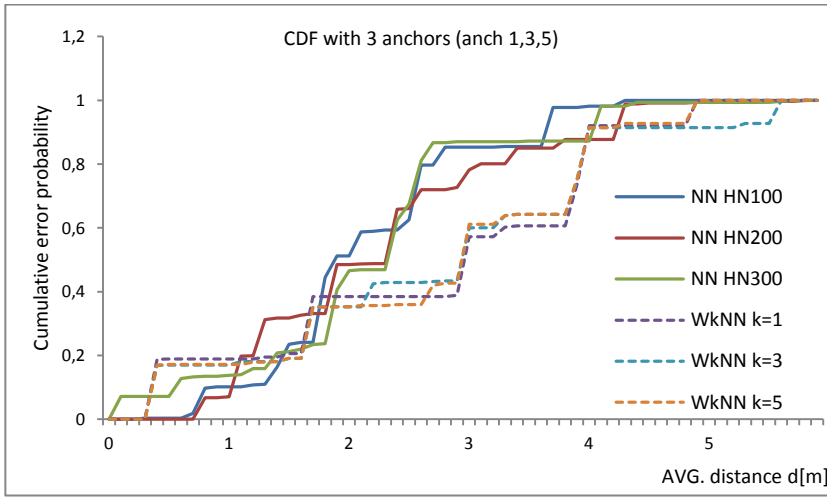


Figure 7

Cumulative distribution functions for distances in case of 3 anchors

The next two figures show the accuracy of the localization depending on the number of anchor motes in the case of the WkNN with  $k=3$  (Figure 8) as well as in the case of the neural network with the structure of 100 HN (Figure 9).

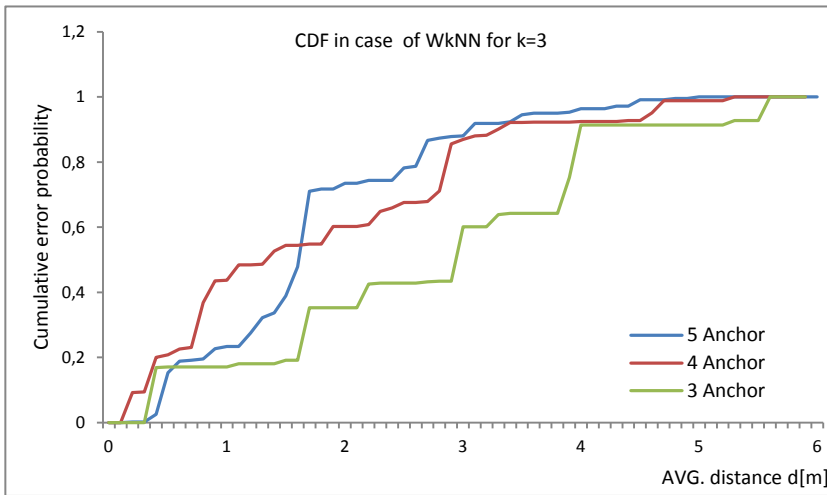


Figure 8

Cumulative distribution functions for distances in case of WkNN for  $k=3$

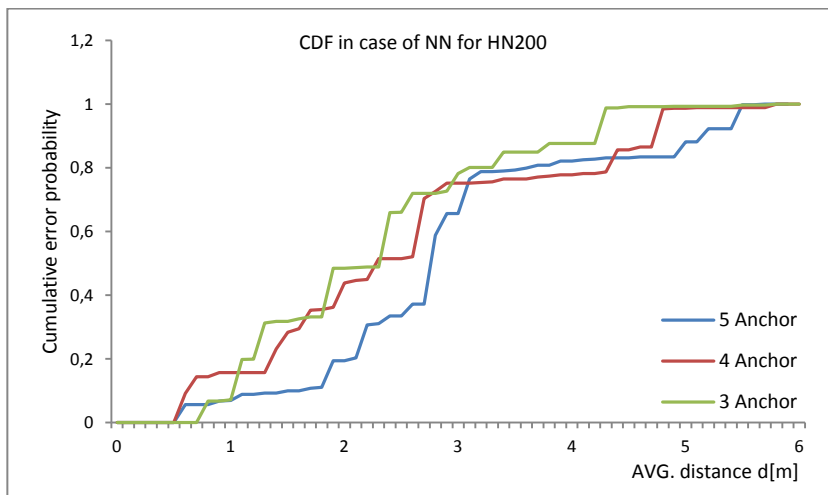


Figure 9

Cumulative distribution functions for distances in case of WkNN for  $k=3$

The localization process with the neural network has shown higher accuracy when less anchors were used. This is due to the greater robustness of the NN on increased uncertainty in conditions with less information. Additionally, the position structure of the anchors can also affect the accuracy of localization.

## Conclusions

In this paper a WSN based fingerprinting localization method was presented. The RSSI values of the communication links between the previously situated sensors and the mobile sensor were recorded in an indoor environment through the experiment. Using the recorded RSSI values a feed-forward type of neural network was trained. The result of the training is a neural network capable of performing indoor localization. The WkNN algorithm was used as a reference model for the performance verification of the created neural network. The accuracy of the localization between the real and the calculated values was measured with Euclidean distance and demonstrated with the cumulative distribution function. The results have shown that the accuracy of derived neural network also depends on the target position to be determined.

During the testing phase, positions were localized whose RSSI values were not present in the previously recorded database. It has been proven that the accuracy of the applied methods greatly depends on the parameters of the method and the number of the anchors used in the process of localization. In the case of the default structure localization where the RSSI values of all the five anchor nodes were processed, the WkNN method proved to be more accurate than the neural network. Since the WkNN method reacted more sensitively to the change of the number of the anchors, when only three anchors were used, the accuracy of the WkNN

method was worse than the accuracy of the neural network. This also proves that the position structure of the anchors has a significant effect on the accuracy of localization.

The test results have shown that the number of the anchors and their spatial position have a significant effect on the accuracy of the fingerprinting localization methods discussed. The authors plan on performing more experiments using different anchor structures in their future research.

### **Acknowledgement**

This work was partially supported by the Ministry of Education and Science of the Republic Serbia under the Grant TR-32034, by the Secretary of Science and Technology Development of Vojvodina Province under the Grant 114-451-2434/2011-01 and by the Hungarian Development Agency under the TÁMOP-4.2.2.A-11/1/KONV-2012-0073 program.

### **References**

- [1] R. Stoleru, T. He, John A. Stankovic, D.Luebke, "A High-Accuracy, Low-Cost Localization System for Wireless Sensor Networks", *Proceedings of the 3<sup>rd</sup> international conference on Embedded networked sensor systems*, pp. 13-26, San Diego, California, 2005
- [2] M. Boushaba, A. Hafid, A. Benslimane, "High Accuracy Localization Method Using AoA in Sensor Networks", *Computer Networks*, Volume 53, Issue 18, 24, pp. 3076-3088, December 2009
- [3] M. Yi-Jen, L. Chih-Min, I. J. Rudas, "Wireless Sensor Network (WSN) Control for Indoor Temperature Monitoring ", *Acta Polytechnica Hungarica*, Vol. 9, No. 6, 2012, pp. 17-28
- [4] S. H. Fang and T. N. Lin, "Indoor Location System Based on Discriminant-Adaptive Neural Network in IEEE 802.11 Environments", *IEEE Trans. Neural Networks*, Vol. 19, no. 11, pp. 1973-1978, 2008
- [5] L. Gogolak, Sz. Pletl, D. Kukolj, "Indoor Fingerprint Localization in WSN Environment Based on Neural Network", *9<sup>th</sup> International Symposium on Intelligent Systems and Informatics- SISY2011, IEEE*, pp. 293-296, 8-10 Sept. 2011
- [6] J. Simon, G. Martinović, I. Matijevis, "WSN Implementation in the Greenhouse Environment Using Mobile Measuring Station" *International Journal of Electrical and Computer Engineering Systems*, pp. 1-10, Osijek, Croatia, 2010
- [7] R. Belbachir, Z. M. Mekkakia and A. Kies, "Towards a New Approach in Available Bandwidth Measures on Mobile Ad Hoc Networks", *Acta Polytechnica Hungarica*, Vol. 8, No. 4, 2011, pp. 133-148

- [8] Sz. Pletl, P. Gál, D. Kukulj, L. Gogolák, "An Optimizing Coverage in Mobile Wireless Sensor Networks" *8<sup>th</sup> International Symposium on Intelligent Systems and Informatics -SISY 2010*, Subotica, September 2010
- [9] L. K. Qabajeh, M. L. M. Kiah and M. M. Qabajeh, "Secure Unicast Position-based Routing Protocols for Ad-Hoc Networks", *Acta Polytechnica Hungarica*, Vol. 8, No. 6, 2011, pp. 191-214
- [10] D. Kukulj, M. Vuckovic, S Pletl , "Indoor Location Fingerprinting Based on Data Reduction," *Broadband and Wireless Computing, Communication and Applications (BWCCA), 2011 International Conference on* , vol., no., pp. 327-332, 26-28 Oct. 2011
- [11] T. Martinetz and K. Schulten. "Topology Representing Networks" *Neural Networks*, Vol. 7, No. 3, pp. 507-522, 1994
- [12] D. Kukulj, B. Atlagić, M. Petrov, "Unlabeled Data Clustering Using a Re-organizing Neural Network," *Cybernetics and Systems, An Int. Journal*, Vol. 37, No. 7, pp. 779-790, 2006
- [13] G. H Golub and C. F. Van Loan, "Matrix Computations", *Johns Hopkins University Press*, Baltimore, 1989
- [14] D. Kukulj, E. Levi, "Identification of Complex Systems Based on Neural and Takagi-Sugeno Fuzzy Model," *IEEE SMC-part B*, Vol. 34, No. 1, pp. 272-282, February 2004
- [15] A. A. Kannan, M. Guoqiang, B. Vucetic, "Simulated Annealing-based Wireless Sensor Network Localization" *Journal of Computers*, Vol. 1, No. 2, pp. 15-22, May 2006
- [16] G. Antal, K. Lamár, "Modern Solutions to Integrated Building Automation Systems" *Proceedings of the International Conference "Kandó 2002"*, Budapest, Hungary p. 5, 2002
- [17] T. Boros, K. Lamár, "Six-Axis Educational Robot Workcell with Integrated Vision System" *Proceedings of 4<sup>th</sup> IEEE International Symposium on Logistics and Industrial Informatics "LINDI 2012"*, Smolenice, Slovakia pp. 239-244, 2012
- [18] K. Hechenbichler, K. P. Schliep, "Weighted k-Nearest-Neighbor Techniques and Ordinal Classification" *Discussion Paper 399*, SFB 386, Ludwig-Maximilians University Munich, October 2004
- [19] S. H. Fang, C. H. Wang, T. Y. Huang, C. H. Yang, Y. S. Chen, "An Enhanced ZigBee Indoor Positioning System With an Ensemble Approach", *IEEE Communications Letters*, Vol. 16, No. 4, pp. 564-567, April 2012