INTL JOURNAL OF ELECTRONICS AND TELECOMMUNICATIONS, 2020, VOL. 66, NO. 4, PP. 769-774
Manuscript received June 28, 2020; revised October, 2020. DOI: 10.24425/ijet.2020.134039

Localization technique of IoT Nodes Using Artificial Neural Networks (ANN)

Beata Krupanek, and Ryszard Bogacz

***Abstract*—One of the ways to improve calculations related to determining the position of a node in the IoT measurement system is to use artificial neural networks (ANN) to calculate coordinates. The method described in the article is based on the measurement of the RSSI (Received Signal Strength Indicator), which value is then processed by the neural network. Hence, the proposed system works in two stages. In the first stage, RSSI coefficient samples are taken, and then the node location is determined on an ongoing basis. Coordinates anchor nodes (i.e. sensors with fixed and previously known positions) and the matrix of RSSI coefficients are used in the learning process of the neural network. Then the RSSI matrix determined for the system in which the nodes with unknown positions are located is fed into the neural network inputs. The result of the work is a system and algorithm that allows determining the location of the object without processing data separately in nodes with low computational performance.**

***Keywords* —wireless networks, node localization, location errors, WSN, IoT, neural networks**

I. Introduction

THE technology of wireless sensor networks enables the construction of systems in which data obtained from sensor nodes are used to control other objects. The combination of a large number of small devices allows you to create an extensive, fully autonomous, functional and distributed system. A well-known example of such systems are Wireless Sensor Networks (WSN) [1]. Such networks consist of hundreds or even thousands of small, relatively cheap and uncomplicated devices equipped with various types of sensors, capable of making measurements, pre-processing the results of these measurements and sending them if necessary [1].

In order to locate network devices inside buildings, technology is needed that allows multiple devices to communicate using one wireless standard. Currently, the most commonly used are WiFi, Bluetooth and ZigBee. Accurate determination of the distance between the transceivers, and hence the ability to determine their geographical coordinates, is one of the key problems in the field of construction and implementation of sensory networks, especially when the nodes are not or cannot be equipped with GPS receivers.

The location of nodes is an important element of IoT systems, where it is necessary to determine the position of objects that can change their location. One of the most frequently used methods of locating nodes in wireless systems is the RSS (Received Signal Strength) method based on the important fact about radio wave propagation - the increasing wave attenuation with distance from the transmitter [1, 2].

The RSSI (Received Signal Strength Indicator) value determined by the radio receiver can indicate the distance of the device transmitting the message and thus can be used to determine the relative or absolute coordinates of the transmitter [1]. Unfortunately, the signal from the transmitter, apart from the situation of mutual visibility of devices in the open space, is often disturbed by multiple signal reflections caused by interior fittings, as well as walls and ceilings. In addition, this type of signal is disturbed by other devices operating at the same frequency.

The purpose of this study is to present the basic issues related to the location of nodes in IoT systems and to present the developed correction method based on the RSS method that allows increasing the accuracy of determining the location of the object. Our idea is to apply an Artificial Neural Networks (ANNs) to enhance a localization accuracy.

The work was to verify the validity of the concept localization of IOT nodes using ANN. In addition, various ANN networks were examined and the quality of results obtained was assessed. The main purpose of the work was to minimize the node position error. The positioning time and energy consumption are also important.

This method has been chosen because in this case the RSSI database can be used for supervised learning of the neural network. Also ANNs localization techniques are capable of representing complicated relationship between input and output variables, and acquire knowledge about these relationships directly from the data.

Theoretical results for ANN assert that one single hidden layer network with sufficient numbers of sigmoidal nodes (in the hidden layer) are capable of approximating any continuous function arbitrarily well. This approach has been used in some of the reported works [3, 4, 5].

The rest of paper is organized as follows. Section II shows measuring the RSSI values techniques and determining of first part of coordinates. Section III introduces the artificial neural network and conjugate gradient based training algorithms. Section IV describes the experiment design. The simulation results are presented in Section V. Finally in section VI results are discussed followed by conclusion in section VI.

Authors are with Silesian University of Technology, Faculty of Electrical Engineering (e-mail: beata.krupanek@polsl.pl; ryszard.bogacz@polsl.pl).

II. RSSI MEASUREMENT

In order to assess the accuracy of position determination based on RSSI measurements, a series of measurement experiments was performed, in which the appropriate coefficients were measured at the changing distance of the transmitter from the receiver. Then, on this basis, the node coordinates were determined to estimate the coordinate uncertainty.

RSSI is an indicator of the strength of the received radio signal [2]. Due to noise, interference and unpredictable radio beam path it is impossible to determine the constant RSSI dependence on distance. RSSI is usually given in dBm. The approximate relationship between RSSI and distance is shown by the formula (1):

$$\mathrm{RSSI} = \mathrm{RSSI}_0 + 10n\left(\log_{10}\frac{d}{d_0}\right), \tag{1}$$

where:
d - distance between devices,
$d \geq d_0$,
n - coefficient describing signal power losses in a given environment,
RSSI_0 - means the RSSI reference value at a distance of one meter (d_0) from the transmitter in dBm [6].

Thus, the distance between the node of unknown location and the anchor node can be determined from the relationship:

$$d = d_0 10^{\frac{\mathrm{RSSI}_0 - \mathrm{RSSI}}{10n}}. \tag{2}$$

The RSSI coefficient for various wireless devices (ZigBee, WiFi, etc.) from different manufacturers is given to a certain range, e.g. 0-60, 0-100 etc. This value should be scaled to the power value in dBm according to the manufacturer's guidelines. If the measuring system consisted of modems with different RSSI ranges then this could cause additional errors due to different scaling factors.

Therefore, in measuring systems it is worth choosing, if possible, devices with the same parameters. In turn, the coefficient n for open space is 2, while in indoor buildings it is suggested to use an experimentally determined value in the range 2 - 6, depending on the parameters of the room and the mutual visibility of wireless devices.

To determine the coordinates of a single wireless network node, you must know the coordinates of at least three other nodes - the so-called anchors, i.e. nodes with a fixed and known position. The method of calculating coordinates using the trilateration method on a 2D plane is shown in Figure 1. Nodes marked K1, K2 and K3 act as anchor nodes, N is a node with an unknown location [7].

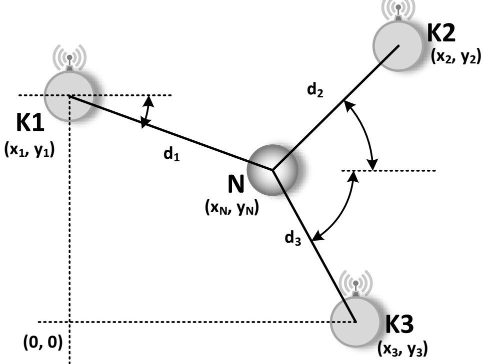

Fig. 1. Determining coordinates using the trilateration method

We can write a system of equations:

$$\begin{cases}(x_1 - x_n)^2 + (y_1 - y_n)^2 = d_1^2 \\ (x_2 - x_n)^2 + (y_2 - y_n)^2 = d_2^2 \\ (x_3 - x_n)^2 + (y_3 - y_n)^2 = d_3^2\end{cases} \tag{3}$$

After transformations, the given system of equations (3) can be written in the matrix form [7]:

$$2\begin{bmatrix}x_a - x_1 & y_a - y_1 \\ x_a - x_2 & y_a - y_2\end{bmatrix}\begin{bmatrix}x_N \\ y_N\end{bmatrix} = \\ = \begin{bmatrix}(d_1^2 - d_a^2) - (x_1^2 - x_a^2) - (y_1^2 - y_a^2) \\ (d_2^2 - d_a^2) - (x_2^2 - x_a^2) - (y_2^2 - y_a^2)\end{bmatrix} \tag{4}$$

To solve the system of equations (4) it is necessary to determine the distance from the examined node based on the RSSI measurement, which is explained in section 2.2. The distances determined on the basis of all three anchor nodes allow the location of the examined node to be determined. However, due to errors in determining RSSI values, which translate into error in determining distance, it is possible that the three determined distances do not give a convergent point of node location, as shown in Figure 2 [6, 7].

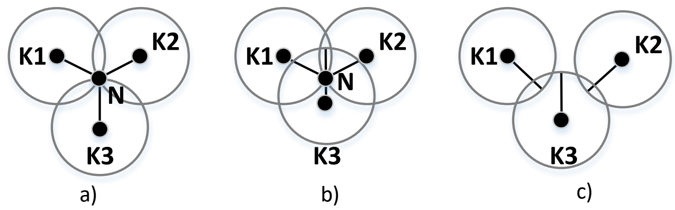

Fig. 2. Correct distance determination a), distance determination based on two nodes b), distance c) cannot be estimated.

In the case of a) shown in Figure 4, the determination of the distance is correct and the distances to three adjacent anchor nodes give a common intersection (location of the node sought). In the case of b) it is possible to determine the distance but it may be incorrect, while in the case of c) it is not possible to obtain information about the distance of the searched node.

Finally, it is possible to determine the error [4, 7] of determining the position δ of the searched node based on the obtained coordinates, which is equal to:

$$\delta = \sum_{i=1}^{k}\sqrt{(x - x_i)^2 + (y - y_i)^2}, \tag{5}$$

where:
k - is the number of nodes based on which coordinates are calculated,
x, y - are the actual coordinates of the searched node,
x_i, y_i - are the estimated coordinates of the searched node.

Experimental verification of the applicability of the method based on RSSI measurement for the purpose of determining the location of nodes in the measurement network included the construction of the measurement system in the form of a sensor network, RSSI measurements, determination of the distance of the searched node, and analysis of the usefulness of this type of methods for determining coordinates.

In order to eliminate interference from multiple reflections of the radio signal, the tests were carried out in the central part of a large seminar room, from which chairs and tables were removed for the time of the test. The location of the measuring network nodes is shown in Figure 3. A grid was laid out in the room to determine relative coordinates. The location of anchor nodes was selected and four possible locations for the examined node were indicated.

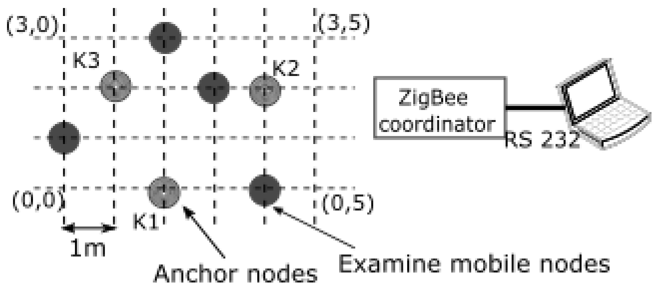

Fig. 3. Arrangement of nodes in the tested measuring network

Generally, the entire network consists of 7 ZigBee modems, of which three have fixed and known relative coordinates, which are node K1 (0, 2), node K2 (2, 4) and node K3 (2, 1) and the other four can move occupying positions on the grid, which gives 21 possible locations for each node, and this publication presents the results when the examined nodes occupy only four locations highlighted in Figure 5.

The ZigBee coordinator communicating with the other nodes is connected via RS232 directly to the computer. Each node sends in a broadcast manner node data (identifier, battery voltage, current time, hop count) along with information on the RSSI value (so-called health information) every 2 minutes. This data is recorded by a computer and saved to a file. The coordinator is equipped with a display that allows viewing selected parameters of the connection with network end nodes.

In order to determine the distance from the given RSSI value, it is necessary to determine in advance $RSSI_0$ and to determine the value of the damping factor n. This process can be called calibration. $RSSI_0$ was designated in the same room where the wireless network was later deployed; by measuring the RSSI value at a distance of 1 m from the transmitter. The measurements were carried out in five different places of the room by making a series of 50 measurements every 15 s. The values obtained were averaged to obtain $RSSI_0$ -36.7 dBm with an uncertainty of 4.3 dBm. Then the coefficient n was determined, which for a given room has a value after averaging equal to 3.27 with uncertainty of 0.31.

Then five measurement series were made, in each of them the mobile node was located in one of four marked places (Fig. 5). For each location, 100 RSSI measurements were taken through each of the nodes - anchors. The RSSI value was used to determine the distance and then the coordinates of the examined node. The measurement results are shown in Figure 4.

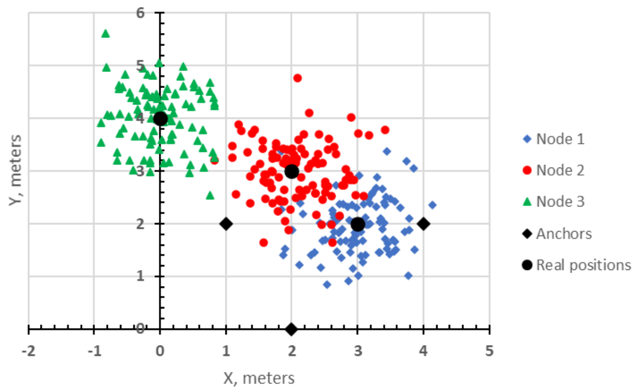

Fig. 4. Results of the node location process

Figure 4 shows that the dispersion of the determined coordinates is significant. The average error in determining the coordinates of the examined node is 1.68 m for the x coordinate and 1.73 m for the y coordinate. Based on the results obtained, it can be concluded that such a system is not suitable for precise location of devices or people, especially in hazardous conditions. It allows you to roughly determine the position and for measurements in continuous mode to estimate the direction of position changes.

III. Artificial Neural Networks

Artificial Neural Networks are interconnection structures among artificial neurons (also called nodes). The artificial neurons are modeled in order to mimic biological neurons through the use of activation functions. Each neuron consists of multiple inputs, weights and a single output. Also, its transfer function is responsible for mapping its inputs to output [8].

Neural networks are generally classified into two categories: the feed-forward neural networks and the feedback neural networks. A feed-forward neural network is one in which the outputs from one layer of neurons feeds forward into the next layer without skipping any layer and have no backward connection. It is the network of interconnected nodes, termed 'Neurons', with linear or nonlinear active functions.

Individual neuron in ANN emulate the biological neuron, it takes input data and make simple processing of it, selectively passing on the processed data to other neurons. Before sending out the processed data, each neuron use its "activation" function to format the data. The training of the network is a procedure that the network gets the knowledge from the data of experiences. The iterative repeat of training allows the network to determine its best fitted weight values.

A BPNN is one of the most frequently utilized ANN techniques for learning both linear and nonlinear functions. BPNN is a neural network that uses a supervised learning method and feed-forward structure for computer learning and modeling. BPNN consists of an input layer, an output layer, and usually one or more hidden layer(s).

In this paper, a multi-layer neural network (a feed-forward ANN) consisting of three layers has been employed using the Matlab software. The output layer can have one or more nodes, depending on the problem. The input signal propagates forward and error signal propagates backward through the network.

Weight adjustments are made to reduce error. In multi-layer feed forward back propagation network model have more than one hidden layers of sigmoid transfer function (nonlinear) neurons and output layer consists of linear activation function neurons. Multiple layers of neurons with nonlinear activation functions allow the network to learn nonlinear and linear relationships between input and output vectors.

The data's propagation through the network is presented below. Firstly, it will be normalized by the input layer neurons and multiplied by the associated weight. Then, the data are summed and reach to the output neurons where they are sent through an activation function, Finally, after the activation function, they become the output of the network:

$$y_j = f\left(\sum_i^n w_{ij} a_i\right) \qquad (6)$$

where n is the total number of input layer neurons, w_{ij} represent the weight updating from the input layer neuron i to the output layer neuron j, a_i is the activations function of the neurons in layer i, y_j is the network's output and f is the activation function of the output layer.

Typical network structure is presented in Fig.5.

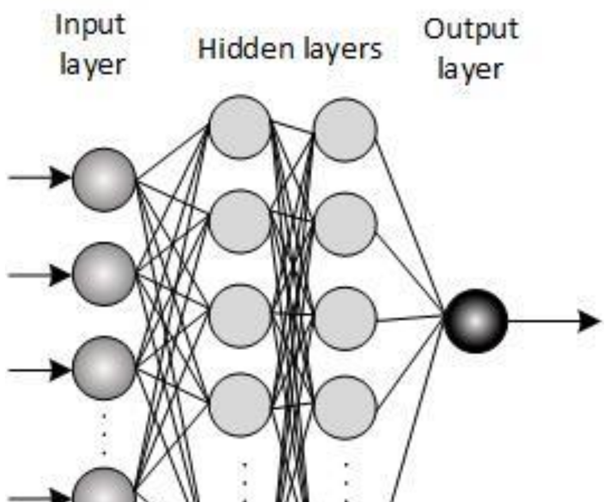

Fig. 5. Multilayer feed forward Back Propagation neural network

The training procedures of our neural network are composed of initialization, a forward pass, and a backward pass. The training process of neural network is obtained through the use of a training pattern, which consists of a set of input vectors with a corresponding output vectors. At the beginning of training, the set of training patterns is given to the input layer of the network. In the forward pass, the training pattern is applied to the input layer and its effect propagates through the network. For training we used a data from previous RSSI measurement.

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 position and for all 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.

IV. Implementation of the Proposed Method

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. The database in this particular experiment is prepared by the measuring of the RSSI values for each coordinate by the sensor nodes. The location vector denotes the position of the reference point. The experimental room where the measurements were done is a research laboratory.

The experimental room structure consists of 4 x 6 measurement points where the RSSI values for the fingerprint are measured. For the testing dataset, measurements from the known positions as well as measurements from several unknown positions that are between the training measurement points are taken. The structure of the dataset, which is a matrix containing the inputs (RSSI values) and the output coordinates are given below, where R_{ij} denotes the RSSI values of the signal perceived from the j-th anchor node, at the i-th reference point while X_i and Y_i denote the x and y coordinates of the i-th reference point:

$$data = \begin{bmatrix} R_{11} & R_{12} & R_{13} & R_{14} & X_1 & Y_1 \\ R_{21} & R_{22} & R_{23} & R_{24} & X_2 & Y_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ R_{n1} & R_{n2} & R_{n3} & R_{n4} & X_n & Y_n \end{bmatrix}. \qquad (7)$$

The ANN obtained using Matlab can be represented by equation 1. In this relation, R is a 4 dimensional input row vector consisting of the RSSI values of the signals received from the four anchor nodes, (l) k W is the weight vector of kth node of l-th layer, (l) k b is the bias vector of kth node of l-th layer. This method can also be used to implement neural network using other programming languages.

$$[x \quad y] = \tanh\left[\tanh\left(R \bullet \left(W_k^{(1)}\right)^T + b_k^{(1)}\right) \bullet \left(W_k^{(2)}\right)^T + b_k^{(2)}\right] \bullet \left(W_k^{(3)}\right)^T + b_k^3 \qquad (8)$$

The detailed steps of the training process are as follows:

(1) Utilize three feasible intersections to establish an input data set for training purposes.

(2) The training process with a training set composed of input patterns together with the required output pattern.

(3) The network has the following input-output mapping:

Input: three feasible intersections (U, V, W).

Output: desired MS location.

(4) The feasible intersections and the true MS location are used to train the network until it establishes the desired relationship.

(5) During training, neural network repeats and adjusts the weights of the connections in the network, and the objective is to minimize the difference between the actual MS location and the desired MS location.

(6) After training, the feasible intersections are input data passing through the trained neural networks to predict the MS location.

All simulations made in this work were performed using MATLAB.

V. RESULTS

Supervised learning is required by ANNs and in this research three different learning/training algorithms: Levenberg-Marquardt, Bayesian Regularization, Back-propagation have been evaluated to get the best result. First, different ANN structures were tested with different number of hidden layers and nodes.

System architecture is show in fig 6. From all the ANN structures evaluated, a 12-12-2 structure gave the most promising results considering the computational complexity and cost of the system. This ANN structure was then used to train the ANN using the above mentioned learning algorithms. The Levenberg-Marquardt training algorithm is the fastest and most efficient training method, but it requires a significant amount of working memory. 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.

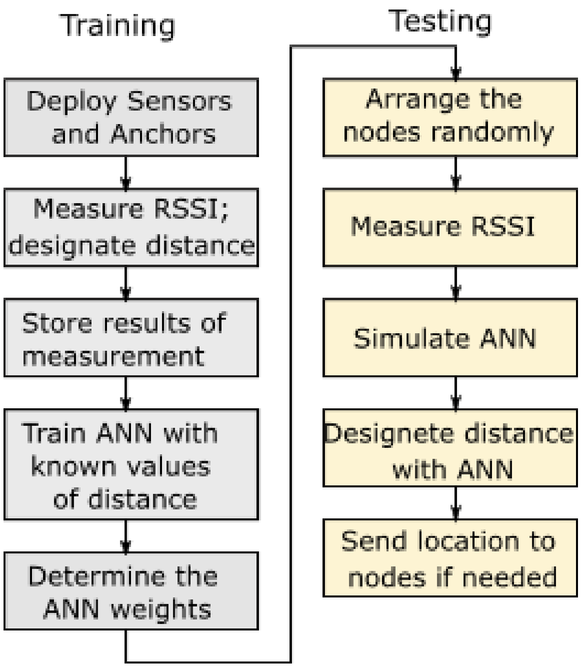

Fig. 6. System architecture for node localization

Bayesian regularized artificial neural networks (BRANNs) are more robust than standard back-propagation nets and can reduce or eliminate the need for lengthy cross-validation. Bayesian regularization is a mathematical process that converts a nonlinear regression into a "well-posed" statistical problem in the manner of a ridge regression.

Backpropagation algorithm is probably the most fundamental building block in a neural network. The algorithm is used to effectively train a neural network through a method called chain rule. In simple terms, after each forward pass through a network, backpropagation performs a backward pass while adjusting the model's parameters (weights and biases).

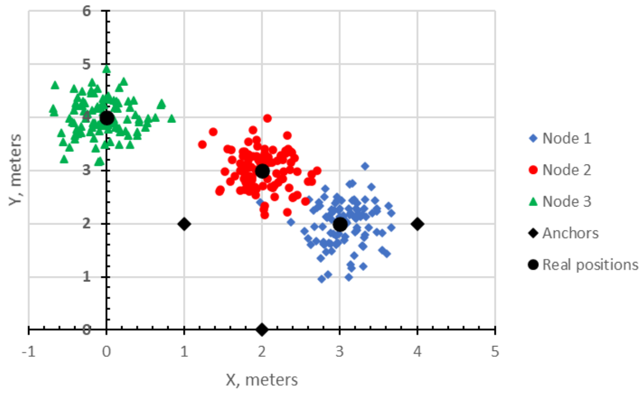

Fig. 7. Results of the node location process after simply correction method

The results of the experiments using the neural network trained by the LM method are compared to the results given by the next two methods. We performed computer simulations to examine the performance of the proposed location algorithm.

Figures 7 and 8 show the results of localization process of sensor nodes. Figure 7 shows effect of simply correction method described in [9]. This method rely on least squares method and it is very simply to use. Also does not consume large computing resources of small sensor nodes. Average error is equal 0,33 m.

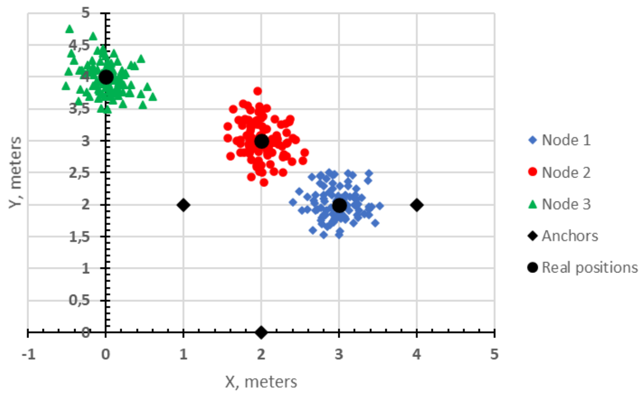

Fig. 8. Results of the node location process after ANN correction

Figure 8 shows localization results of the same three nodes but with use of neural networks computing. The localization error is much smaller then in the first method (fig. 7) and its average value is equal about 0,22 m. Method LM was used.

Learning process of the ANNs is very important part of implementation the neural network into physical measurement. We tested three different learning algorithms: LM, BRANN and BP to get the best results. Different ANN structures were tested with different numbers of hidden layers and nodes. We evaluated ANN structure 12-12-2 structure proposed in [10]. This ANN structure was used to train the ANN using the three learning algorithms. The results of training ANN is shown figure 9. The maximum and average distance error is used to compare the learning methods. The smallest average error obtain of method LM, but the smallest maximum error obtain of BRANN method. However error between methods are small.

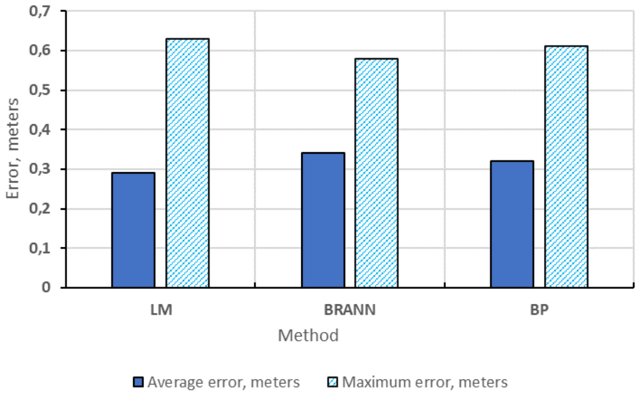

Fig. 9. The graph showing localization error for different learning algorithms

The evaluation of mentioned structures and methods was made through use of root mean square error *RMSE* (9) between the real position and estimated position obtained from measurement or simulation.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}[(x_i - \overline{x_i})^2 + (y_i - \overline{y_i})^2]} \qquad (9)$$

where:
n - the number of testing sensor nodes,
x_i - real x coordinates
y_i - real y coordinates,
$\overline{x_i}$ - estimated x coordinates,
$\overline{y_i}$ - estimated y coordinates,
i - sensor node index.

CONCLUSION

This paper presented an approach to localization possibilities for wireless sensor networks using RSSI measurement and artificial neural networks to improve the localization accuracy.

The major benefit of using ANNs is that the method doesn't require prior knowledge of the environment and noise distribution. It is very important because the RSSI measurements are highly unstable and depends on environmental noise and potential possibility moving sensor nodes.

The results of experiment using neural networks trained by the Levenberg-Marquardt method are compared to two different methods: Bayesian regularized artificial neural networks and Backpropagation algorithm. Furthermore, we compared the neural networks method of increasing accuracy to simply method based on least square method. Use of neural networks give smaller localization error but the computing complexity is much higher then the simply method.

ANN method needs an implementation of computer or high computing power unit. The important factor influencing the performance of the neural network are the number of training iterations and the number of neurons in the hidden layer. This issue will be the subject of further research.

The effect of research and simulation is to increase the accuracy of the location of measuring nodes in the wireless network, but the developed structure requires optimization, especially in terms of computational complexity and data processing time.

ACKNOWLEDGEMENTS

We are very grateful to experts for their appropriate and constructive suggestions to improve this template.

REFERENCES

[1] K. Anup, S. Takuro: Localization in Wireless Sensor Networks: A Survey on Algorithms, Measurement Techniques, Applications and Challenges, *J. Sens. Actuator Netw*., 2017, vol. 6.
[2] E. Figueiredo, P. Silva, V. Kaseva, E.S. Lohan: Wireless Positioning in IoT: A Look at Current and Future Trends, *Sensors*, 2018, 18, pp. 2470.
[3] B. Wu, C. Jen, and K. Chang, Neural fuzzy based indoor localization by Kalman filtering with propagation channel modelling in: Proceedings IEEE International Conference on Systems, Man and Cybernetics, 2007, pp. 812–817
[4] R. Battiti, T.L. Nhat, A. Villani, Location-aware computing: a neural network model for determining location in wireless LANs, Technical Report DIT-02-0083, University of Trento, Department of Information & Communication Technology, 2002
[5] J.P Tian, and H.C Shi, Study of localization scheme base on neural network for wireless sensor networks' in the Proceedings of IET Conference on Wireless, Mobile and Sensor Networks , pp.64-67
[6] W. H. Kuo, Y. S. Chen, K. T. Cheng, T. W Lu.: *Signal Strength Based Indoor and Outdoor Localization Scheme in Zigbee Sensor Networks*, IAENG International Journal of Computer Science, vol. 43, 2016.
[7] J. Luo: *Range Error Correction In RSSI-based Wireless Sensor Node Localization*, Proceedings of 2014 IEEE International Conference on Mechatronics and Automation, August 3 - 6, Tianjin, China.
[8] S. H. Chagas, L. de Oliveira, J. Baptista, S. Martins: Node localization in Wireless Sensor Networks Using Artificial Neural Networks and Optimization Based on Simulated Annealing Algorithm. XXVII SIM - South Symposium on Microelectronics.
[9] B. Krupanek, R. Bogacz: Zwiększenie dokładności lokalizacji węzłów systemów bezprzewodowych. *Pomiary w nauce i technice*. 2019
[10] S. Kumar, S. Lee: Localization with RSSI values for Wireless Sensor Networks: An Artificial Neural Network Approach. Conference Proceedings Paper – Sensors and Application. International Electronics Conference on Sensors and Applications. 2014