Chalmers Publication Library

(article starts on next page)

A Novel Multilayer Neural Network Model for TOA-Based Localization in Wireless Sensor Networks

Sayed Yousef Monir Vaghefi, Reza Monir Vaghefi

Abstract—A novel multilayer neural network model, called artificial synaptic network, was designed and implemented for single sensor localization with time-of-arrival (TOA) measurements. In the TOA localization problem, the location of a source sensor is estimated based on its distance from a number of anchor sensors. The measured distance values are noisy and the estimator should be able to handle different amounts of noise. Three neural network models: the proposed artificial synaptic network, a multi-layer perceptron network, and a generalized radial basis functions network were applied to the TOA localization problem. The performance of the models was compared with one another. The efficiency of the models was calculated based on the memory cost. The study result shows that the proposed artificial synaptic network has the lowest RMS error and highest efficiency. The robustness of the artificial synaptic network was compared with that of the least square (LS) method and the weighted least square (WLS) method. The Cramer-Rao lower bound (CRLB) of TOA localization was used as a benchmark. The model's robustness in high noise is better than the WLS method and remarkably close to the CRLB.

I. INTRODUCTION

WIRELESS sensor network (WSN) localization problem is one of the interesting subjects studied in recent years. In this problem, the locations of anchor sensors are known and the location of each source sensor is estimated based on its distance from the anchor sensors. The approximate distance of a source sensor from an anchor sensor is obtained using different measurement methods such as time of arrival (TOA) [1], time difference of arrival [2], and received signal strength (RSS) [3]. Throughout this work we assume that the distances are obtained using TOA measurements, however, the problem can easily be extended to the other methods.

The maximum likelihood (ML) estimator is the optimal estimator when the number of data records is sufficiently large [1], [4]. However, the cost function of the ML estimator is non-linear and non-convex and finding the global minimum requires convoluted computations. The solution of the ML estimator is computed using iterative optimization methods [1]. Since the cost function has many saddle points and local minima, a good initialization guess is inevitable to make sure that the algorithm converges to the global minimum. To deal with this behavior of the ML estimator, different methods such as semidefinite programming (SDP) relaxation [5], [6] and multidimensional scaling (MDS) [7] are introduced.

Sayed Yousef Monir Vaghefi is with the School of Computer Science and Information Technology at Royal Melbourne Institute of Technology, Melbourne, Australia (email: s3268713@student.rmit.edu.au), Reza Monir Vaghefi is with Department of Signals and Systems, Chalmers University of Technology, Gothenburg, Sweden (email: vaghefi@student.chalmers.se).

In SDP, the cost function is approximated and relaxed to a convex optimization problem and solved with efficient algorithms that do not require any initialization. In MDS, the locations of the sources are estimated using data analysis of the coordinates and distances. It is stated that the MDS approach is very sensitive to measurement noise and not applicable to low connectivity networks [7].

Linear estimators [8], [9], [10] are also investigated in a number of studies. The model of the TOA localization, which is basically a non-linear problem, is linearized using some approximation. The least squares (LS) method is studied in [9]. The method is simple to implement but cannot handle the measurement noise. The TOA measurement noise is usually modeled as a zero-mean Gaussian random variable with a variance depending mostly on the distances (i.e., the larger measured distances have higher noise than the shorter ones). To improve the performance of the LS method, the weighted least squares (WLS) algorithm [11] is introduced which can tolerate unequally sized noises. However, since linearization of the TOA localization model is done under the assumption that the measurement noise is sufficiently small, the performance of LS and WLS declines considerably as the measurement noise increases. Different extensions to the LS method such as the constrained weighted least squares (CWLS) method [8] and corrected least squares method [9] are also introduced. Although these algorithms have better performance in high noise, their computation cost and complexity is higher.

In this project, the TOA-based sensor localization is tackled using artificial neural network models. Shareef et al. [12] have compared the performance of three types of neural networks namely RBF, MLP, and recurrent neural networks in sensor localization. Their result shows that the RBF network performs better than the other networks but it has higher memory and computation costs. On the other hand, the MLP network has the lowest memory and computation costs. Rahman et al. [13] have implemented a MLP network for the WSN localization problem. The network has reached the RMSe of 0.54 meter for a $20m \times 20m$ problem. The localization problem in a UWB sensor network is tackled in [14]. A neural network model is introduced. The performance of the model has not been as good as the LS method. In [15], a neural network model is developed for identification of undetected direct paths (UDP) in TOA localization.

The advantage of artificial neural network models is that they are adaptable to different conditions and situations. Indeed, our proposed model is designed to tolerate a specific condition or situation. For instance, if the sensor network is

set up in an indoor environment where the signals from the sources are blocked and diminished by many objects and the measurements are subject to high noise, the algorithm can be trained to deal with the high-noise measurements. On the other hand, if the connectivity of the sensor network is high and it is required to have an accurate estimate, the algorithm can be trained to handle the low-noise measurements.

The rest of paper is organized as follows. In the networks section, the neural network models are described. In the data section, the training and test data are presented. In the experiments and results section, the models are compared based on their performance and efficiency and the robustness of the artificial synaptic network (ASN) model is compared with that of the LS, and WLS methods. The study results are summarized in the conclusion section.

II. NETWORKS

In this study, the TOA localization problem was considered as a supervised learning problem, in which a neural network is trained with a set of input-output data called the training dataset, and tested with another set of input-output data called the test dataset. Three neural network models: an artificial synaptic network (ASN), a generalized radial basis functions (GRBF) network, and a multilayer preceptron (MLP) network were applied to the problem. The models were implemented in C sharp . The performance and efficiency of the models were studied. The best model was identified and then tested for robustness. The designed ASN model is comprised of a number of artificial synaptic networks - each of them working on a data cluster. The architecture of each network is presented in Fig. 1. The output of each network is computed using:

$$f(x) = \sum_{i=1}^{n} w_i * x_i + w_0. \qquad (1)$$

where n is the number of inputs, and is constructed in a deeper layer:

$$w_i = \sum_{j=1}^{n} w_{ij} * x_j + w_{i0}. \qquad (2)$$

where n is the number of inputs. In training, a center is assigned to each ASN by the k-means clustering algorithm. Each training data goes to the closet network. The error is computed at the output layer of the network. The error is then backpropagated to the deepest layer. The weights at the deepest layer are updated. The weights at the next layers are not updated; they are instead constructed layer by layer from the weights of the deepest layer. The learning algorithm:

1. Initialize the centers, the output layer, the second layer, and the third layer weights, and the maximum acceptable error

2. For each data point

a. Compute the Euclidean distance of the point from all the centers

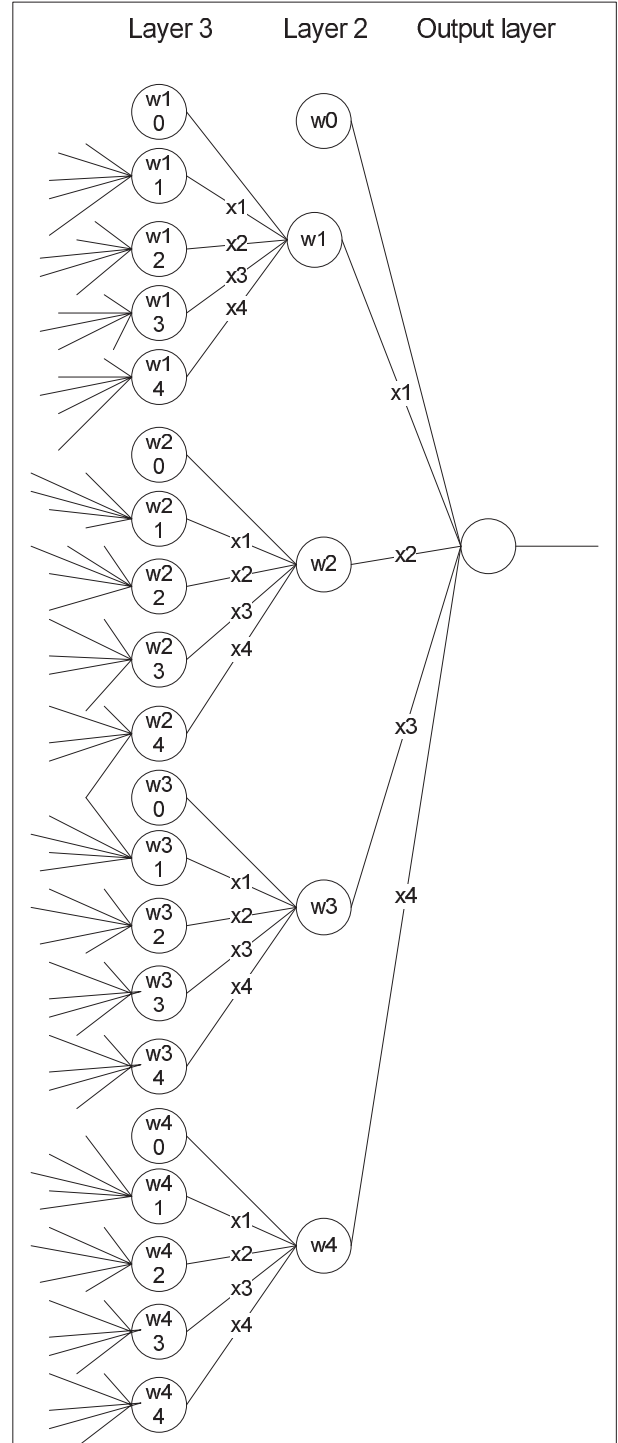b. Assign the closet center to the point

Fig. 1.   Artificial Synaptic Network (ASN) architecture

3. For each center

Change the center to the centroid of the data points assigned to that center

4. If not yet converged go to step 2

5. For each data point: x

a. $backpropagate(x)$

b. Compute the networks output:

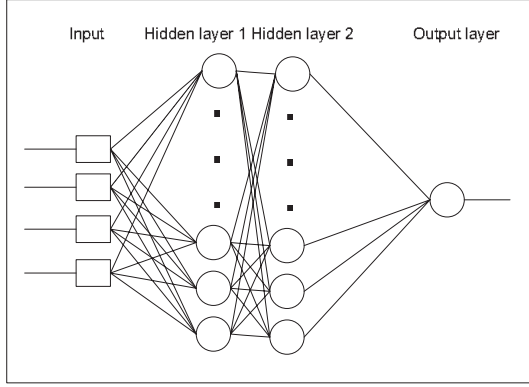$networkout(x) = computeoutput(x)$
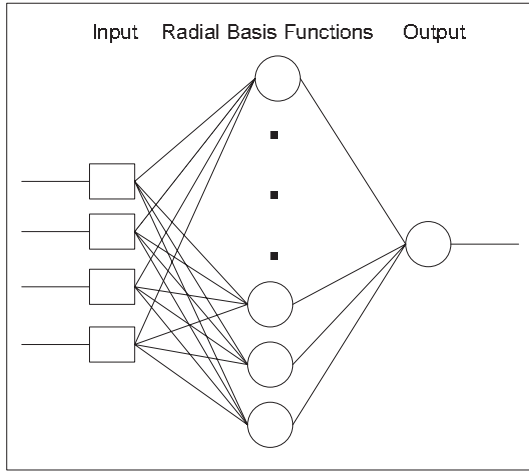

Fig. 2.   Multilayer Preceptron Network Architecture

Fig. 3.   GRBF Network Architecture

6. Compute the Root Mean Square Error:

$$rmse = \sqrt{\frac{\sum_{i=1}^{n} \left(output_i - networkout_i \right)^2}{n}} \quad (3)$$

where n is the number of training data.

7. If $(rmse > maxerror)$ go to step 5

The computeoutput(x: the data point) method:

1. Compute the Euclidean distance of x from all the centers

2. Send x to the network with the closet center

3. Compute the output of the chosen network:

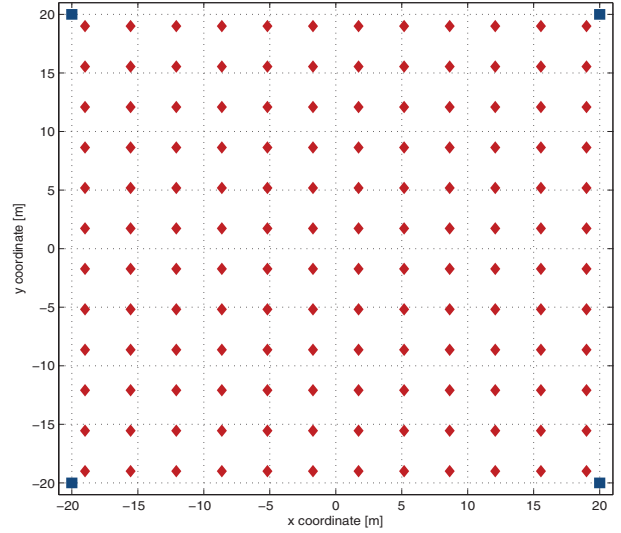$$w_i = \sum_{j=1}^{n} w_{ij} * x_j + w_{i0}. \quad (4)$$

Fig. 4.   Training dataset

Fig. 5.   Test dataset

$$networkout(x) = \sum_{i=1}^{n} w_i * x_i + w_0. \quad (5)$$

where n is the number of inputs.

The backpropagation(x:the data point) method:

1. Compute the output for the data point:

$computeoutput(x)$

2. Backpropagate the error layer by layer

a. Output layer:

$delta1 = output(x) - networkout(x)$

b. Second layer:

$delta2_i = delta1 * x_i$

c. Third layer:

$delta3_{ij} = delta2_i * x_j$

3. Update the weights of the deepest layer

$layer3weights = layer3weights + eta * delta3 * x$

The multilayer preceptron (MLP) network implemented for the problem has two hidden layers. The architecture of the network is presented in Fig. 2. The activation function of the hidden neurons is a logistic function. The network was trained using the backpropagation algorithm [16].

The GRBF network [17] consists of a number of radial basis function neurons, each of them working on a center. In the training, the fixed centers of the network are specified using the k-means clustering algorithm. The network is then trained using the gradient descent algorithm. Fig. 3 shows the architecture of the network.

III. DATA

The performance of a neural network model depends on the density of the training data and complexity of the problem. If the training data is not dense, the network does not have enough information to build the model. Considering the modeling process as a hyperplane reconstruction process, the more complex the hyperplane is the more data is required for reconstruction.

The datasets were generated in MATLAB. The training dataset was a set of 144 data points evenly distributed over the input space. Fig. 4 presents the training dataset. The red points, diamonds, are the source sensors and the blue points, squares, are the anchors. The test dataset was 300 data points randomly distributed over the input space. The distances of each data point from the anchors were calculated by MATLAB. A random amount of noise was added to the distances. The test dataset is shown in the Fig. 5.

IV. EXPERIMENTS AND RESULTS

A model with two outputs is required to estimate the location, x and y, of a sensor. However, since one-output models have less complexity and less training time, two separate models of one output were employed to estimate x and y of a sensor. In the first experiment the models were trained with the training dataset of Fig. 4. The performance of the models was compared based on the Root Mean Square error (RMSe). The efficiency of the models was compared based on the memory cost. Table 1 shows the memory cost and RMSe of the models.

The memory cost was calculated based on the number of memory blocks required to store the centers and synaptic weights. In the GRBF network, there are 15 centers that require 60 memory blocks, and there are 15 synaptic weights that require 15 blocks of memory. The MLP network has 7 neurons in each hidden layer which means 28 synaptic weights between the input layer and the first hidden layer, 49 synaptic weights between the first and the second hidden layer, and 7 synaptic weights between the second hidden layer and the output layer. The number of synaptic weights in the ASN model is equal to the number of centers multiplied by the number of neurons in the deepest layer. The model has 3 centers and there are 20 neurons in the deepest layer.

In the second experiment, more neurons were added to the MLP network. The RMSe decreased but the training was considerably slow. Fig. 6 shows the RMSes of networks

	RMSe	Memory Cost	Iterations
Generalized Radial Basis Functions (GRBF) Network	2.829	4*15+15=75	99917
Multilayer Preceptron (MLP) Network	3.204	4*7 + 7*7 +7 = 84	97186
Artificial Synaptic Network (ASN)	0.2999	3*4*5 +3*4= 72	91697

of different number of hidden neurons. The number of hidden neurons in the first and second hidden layer is equal. The networks were trained for 100 thousand iterations. By increasing the number of hidden neurons, the RMSe first decreases. But then it increases because networks with more hidden neurons require more iterations to converge. Lower RMSe can be achieved, if the learning rate goes up. After changing the number of neurons to 12, the learning rate was increased. As a result the RMSe went down. To conclude, the MLP network could not achieve better performance than the ASN model, even with more hidden neurons.
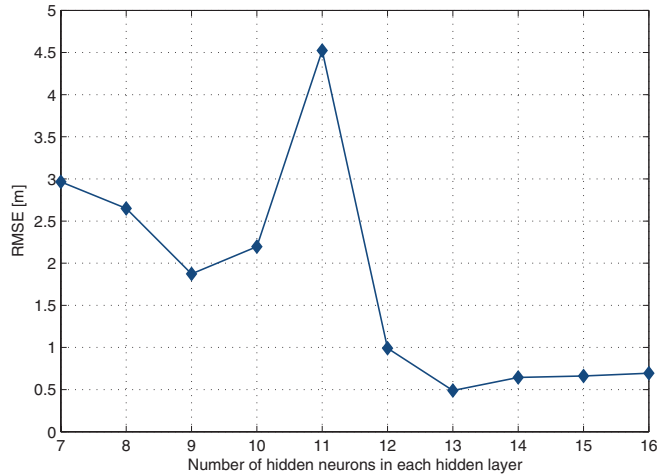

Fig. 6.   Influence of number of hidden neurons on RMSe

The ASN model had the best performance and efficiency. Compared to the other models, the ASN model converges faster and requires less memory. It has also achieved the lowest RMSe. Therefore, the ASN model was selected for the TOA localization problem.

In the third experiment, the ASN model was trained with a termination condition of $RMSe < 0.3m$. The model was then tested on the test dataset. The computed RMSe is 0.335 m. Fig. 7 shows the estimated and true locations of the source nodes. The model was again trained with a termination condition of $iterations > 400000$. The model reached the RMSe of 0.258 m. The model was then tested with the test dataset. The RMSe of the test dataset is 0.271 m. Thus, in a $40m*40m$ square area, the location of a source sensor can be estimated with an average error of 27.1 cm.

In the next experiment, the ASN model was tested for robustness. A total of 300 random locations were selected. For each location, the distance between the location and

each anchor was computed. A zero-mean Gaussian noise was added to the distances. A set of 10 different four-distance inputs were computed for each location. A total of 3000 inputs were sent to the ASN model and the RMSe was computed. The experiment was repeated for different
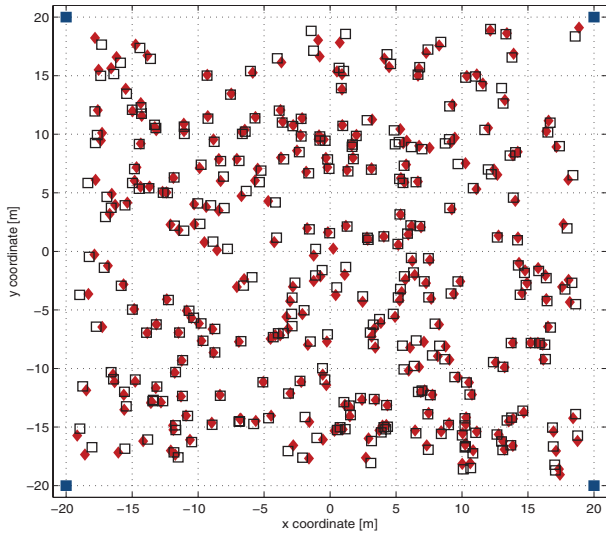

Fig. 7. Diamonds depict estimated locations and squares depict true locations.

Gaussian noises. Two other methods applied to the WSN localization problem, the LS method and the WLS method, were tested using the same procedure. Fig. 8 shows the computed RMSe in the presence of different amounts of noise. As shown in the figure, although the ASN model has achieved lower RMSes compared to the LS method, the model has not been as robust as the WLS method.

To improve the robustness of the model, two centers were added to the model. The memory cost increased to 100. The model was trained using the same training dataset. The training stopped with the RMSe of 0.3 m. The model was tested with the same set of test inputs. Fig. 9 presents the robustness of the model. As shown in the figure, the model is almost as robust as the WLS method. However, WLS outperforms the model when the standard deviation of noise is lower than 0.5.

If the model is less trained, it would be more robust. In other words, if the RMSe of the training data is higher, the model's tolerance of noise is higher. To test this theory, the model was trained with the termination condition of $RMSe < 0.5m$. The model was then tested on the test dataset. Fig. 10 compares the model's robustness with that of the other methods. As demonstrated by this figure, when the noise standard deviation is higher than or equal to 2, the model has lower RMSes compared to the WLS method. Conversely, when the noise standard deviation is lower than 2, the model has higher RMSes in compare with the WLS method.

Another way to improve the robustness is to train the model with noisy data. A zero-mean Gaussian noise with the standard deviation of 2 was added to the training data. The
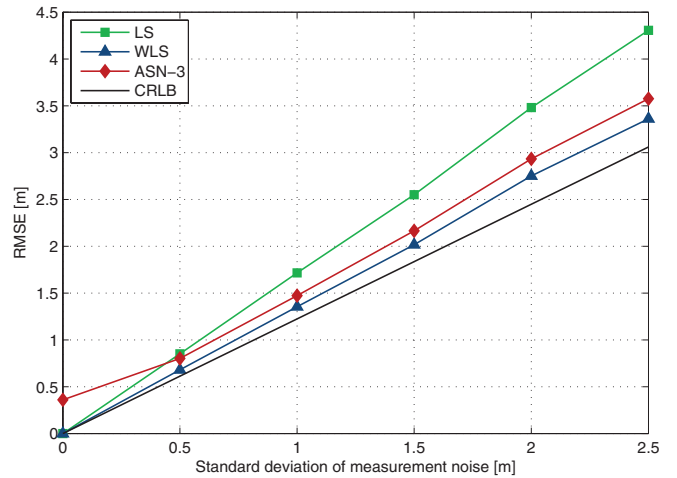

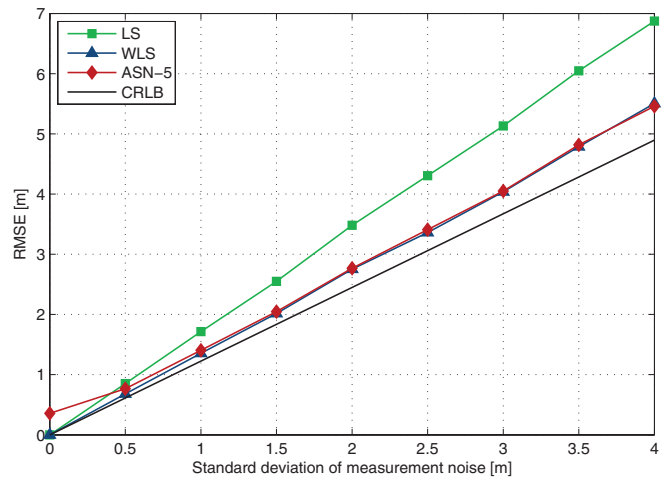Fig. 8. ASN model with 3 centers

Fig. 9. ASN model with 5 centers

model was trained with the new training dataset. The model was then tested with the same test inputs. Fig. 11 shows the results. When the noise is high, the model outperforms the WLS method. In contrast, when the noise is low, the model fails to compete with the WLS method. Overall, the model's robustness is better than the LS method and almost as good as the WLS method. If trained with noisy data, the model can outperform the WLS method.

V. CONCLUSION

In TOA localization, our artificial synaptic network (ASN) model has better performance and efficiency compared to GRBF and MLP models. The model converges faster and its memory cost is lower. Tested on the training dataset, the model reached the RMSe of 0.258 m. Tried on the test dataset the model achieved the RMSe of 0.271 m. Therefore, in a 40 m * 40 m square area, the location of a source sensor can be estimated with an average error of 27.1 cm.

The robustness of the ASN model is almost as well as the weighted least squares (WLS) method. Adding more centers to the model improves the robustness of the model. However,
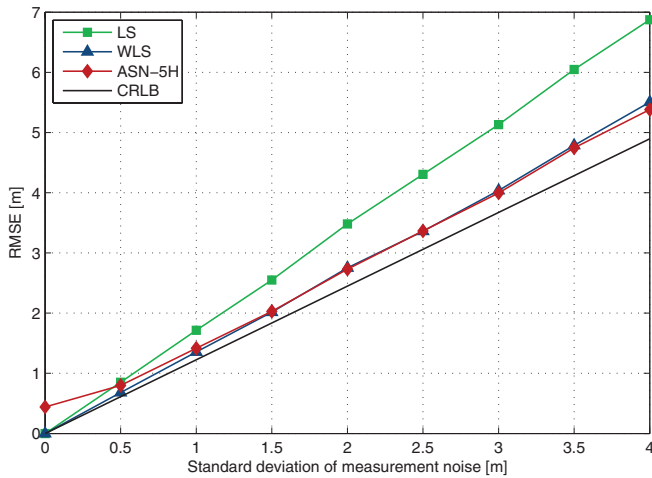
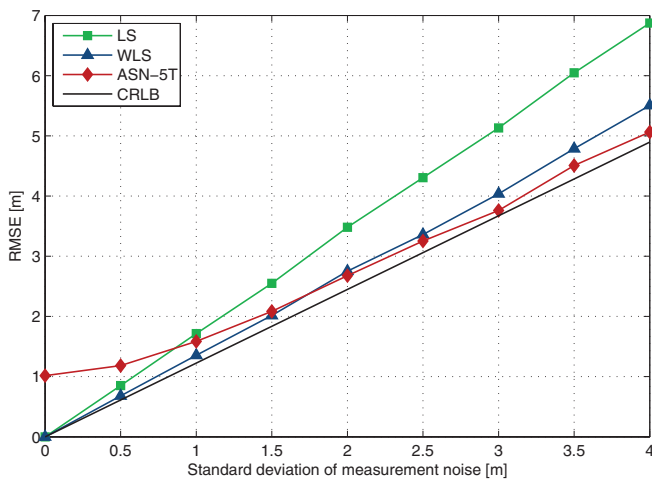Fig. 10. Influence of lowering the expected training RMSe

Fig. 11. 5-center ASN model trained with noisy data

there is a limit in this improvement. If trained with noisy data, the model can outperform the WLS method. In high noise, the model performs better than the WLS method. On the contrary, in low noise, the model fails to perform as well as the WLS method.

REFERENCES

[1] S. Kay, *Fundamentals of statistical signal processing: estimation theory*. Prentice-Hall, 1993.
[2] K. Ho and Y. Chan, "Solution and performance analysis of geolocation by tdoa," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 29, no. 4, pp. 1311 –1322, oct 1993.
[3] N. Patwari, I. Hero, A.O., M. Perkins, N. Correal, and R. O'Dea, "Relative location estimation in wireless sensor networks," *Signal Processing, IEEE Transactions on*, vol. 51, no. 8, pp. 2137 – 2148, aug. 2003.
[4] Y.-T. Chan, H. Yau Chin Hang, and P. chung Ching, "Exact and approximate maximum likelihood localization algorithms," *Vehicular Technology, IEEE Transactions on*, vol. 55, no. 1, pp. 10 – 16, jan. 2006.
[5] A. M.-C. So and Y. Ye, "Theory of semidefinite programming for sensor network localization," in *Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms*, ser. SODA '05. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 2005, pp. 405–414. [Online]. Available: http://portal.acm.org/citation.cfm?id=1070432.1070488
[6] C. Meng, Z. Ding, and S. Dasgupta, "A semidefinite programming approach to source localization in wireless sensor networks," *Signal Processing Letters, IEEE*, vol. 15, pp. 253 –256, 2008.
[7] K. Cheung and H. So, "A multidimensional scaling framework for mobile location using time-of-arrival measurements," *Signal Processing, IEEE Transactions on*, vol. 53, no. 2, pp. 460 – 470, feb 2005.
[8] K. W. Cheung, H. C. So, W.-K. Ma, and Y. T. Chan, "A constrained least squares approach to mobile positioning: algorithms and optimality," *EURASIP J. Appl. Signal Process.*, vol. 2006, pp. 150–150, January 2006. [Online]. Available: http://dx.doi.org/10.1155/ASP/2006/20858
[9] Y. Chan and K. Ho, "A simple and efficient estimator for hyperbolic location," *Signal Processing, IEEE Transactions on*, vol. 42, no. 8, pp. 1905 –1915, aug 1994.
[10] R. Vaghefi, M. Gholami, and E. Strom, "Bearing-only target localization with uncertainties in observer position," in *Personal, Indoor and Mobile Radio Communications Workshops (PIMRC Workshops), 2010 IEEE 21st International Symposium on*, sept. 2010, pp. 238 –242.
[11] M. Spirito, "On the accuracy of cellular mobile station location estimation," *Vehicular Technology, IEEE Transactions on*, vol. 50, no. 3, pp. 674 –685, may 2001.
[12] A. Shareef, Y. Zhu, and M. Musavi, "Localization using neural networks in wireless sensor networks," in *Proceedings of the 1st international conference on MOBILe Wireless MiddleWARE, Operating Systems, and Applications*, ser. MOBILWARE '08. ICST, Brussels, Belgium, Belgium: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2007, pp. 4:1–4:7. [Online]. Available: http://portal.acm.org/citation.cfm?id=1361492.1361497
[13] M. Rahman, Y. Park, and K.-D. Kim, "Localization of wireless sensor network using artificial neural network," in *Communications and Information Technology, 2009. ISCIT 2009. 9th International Symposium on*, sept. 2009, pp. 639 –642.
[14] S. Ergut, R. Rao, O. Dural, and Z. Sahinoglu, "Localization via tdoa in a uwb sensor network using neural networks," in *Communications, 2008. ICC '08. IEEE International Conference on*, may 2008, pp. 2398 –2403.
[15] M. Heidari, N. Alsindi, and K. Pahlavan, "Udp identification and error mitigation in toa-based indoor localization systems using neural network architecture," *Wireless Communications, IEEE Transactions on*, vol. 8, no. 7, pp. 3597 –3607, july 2009.
[16] S. Saarinen, R. Bramley, and G. Cybenko, "Neural networks, back-propagation, and automatic differentiation," in *Automatic Differentiation of Algorithms: Theory, Implementation, and Application*, A. Griewank and G. F. Corliss, Eds. Philadelphia, PA: SIAM, 1991, pp. 31–42.
[17] S. Haykin, *Neural Networks: A Comprehensive Foundation*, 2nd ed. Upper Saddle River, NJ, USA: Prentice Hall PTR, 1998.