



International Journal of Engineering Technologies and Management Research

A Knowledge Repository

DEVELOPMENT OF ARTIFICIAL NEURAL NETWORK STRUCTURES FOR PREDICTING NAVIGATION TASKS OF A MOBILE ROBOT

Irem Sahmutoglu ¹, Erhan Akdogan ^{*2}

¹ Industrial Engineering, Yildiz Technical University, Turkey

^{*2} Mechatronics Engineering, Yildiz Technical University, Turkey



Abstract:

Determining trajectories in mobile robot navigation tasks is a difficult process to apply with conventional methods. Therefore, intelligent techniques produce highly effective results in trajectory optimization and orientation prediction. In this study, two different ANN (Artificial Neural Network) structures have been developed for the navigation prediction of the SCITOS G5 mobile robot. For this aim, RBF (Radial Basis Function) and MLP (Multi-Layer Perceptron) structures were used. Information obtained from 24 sensors of the robot was used as network inputs and network output determines robot direction. Accordingly, structures that have 24 inputs and one output were created. The best performance network structures obtained were compared among them in simulation environment. Accordingly, RBF has been observed to produce more accurate results than MLP.

Keywords: Artificial Neural Network; Robot Navigation; SCITOS G5.

Cite This Article: Irem Sahmutoglu, and Erhan Akdogan. (2020). “DEVELOPMENT OF ARTIFICIAL NEURAL NETWORK STRUCTURES FOR PREDICTING NAVIGATION TASKS OF A MOBILE ROBOT.” *International Journal of Engineering Technologies and Management Research*, 7(3), 42-50. DOI: 10.29121/ijetmr.v7.i3.2020.553.

1. Introduction

The general aim of technological progress is to reduce human labor and increase productivity. For this reason, robots have become a part of this technological progress. Today, after years of research and development, robots are used in a wide variety of fields, from the service sector to the medical field [1,2].

Intelligent mobile robots are used in applications such as space, transportation, industry and defense. Therefore, navigation of mobile robots that can move freely in a static or dynamic environment is an important problem. The main purpose of mobile robot navigation is to provide a smooth and safe transportation of the mobile robot in a dispersed environment by following a safe path from its starting position to its target position and producing an optimum trajectory [3]. Robots are basically designed to perform complex and nonlinear real-time operations in their environment. Creating a successful road navigation plan is a difficult and complex problem [4].

It is difficult to address such problems with traditional techniques. Because the problem has a nonlinear and dynamic structure. ANNs are one of the effective control techniques in the solution of nonlinear problems. ANNs have been used successfully in areas such as trajectory control for underwater vehicles [5], unmanned flight [6], human-robot interaction [7].

There are various problems especially in the navigation control of mobile robots. In order to solve these problems, researchers have proposed various ANN structures for different purposes. Junior et al. used a Multilayer Perceptron (MLP) structure for trajectory detection and classification of robotic systems in trajectory programming [8]. In this study, it is analyzed whether robotic failures are classified by MLP. For this application, the number of neurons in the hidden layer best meets the criteria of Kolmogorov and Weka topologies. Compared to an algorithm that uses the same dataset, MLP has achieved the best performance in any classification topology. Kruse et al. conducted a survey of current approaches to human-aware navigation and provided a general classification scheme for the presented methods [9].

Panigrahi et al. performed motion control of an autonomous mobile robot using intelligent multi-layer sensor (MLP) and radial-based function (RBF) techniques. This study focused on avoiding obstacles and looking for targets. They trained these ANN-based controllers using different training models [10]. Shinzato et al. used a multi-layered neural network and image processing technique to define navigable terrain. To evaluate the proposed method, they evaluated with an outdoor robot using different network topologies [11, 12].

Dezfoulian et al. proposed a globalized navigation algorithm to reduce training time and sensor costs. The algorithm is trained using 2D data and interprets this data [13]. Wang et al. proposed a Spiking Neural Network (SNN) network for robot control, unlike traditional artificial neural networks. The SNN network gives good temporal and spatial results for mobile robot control [14]. Malleswaran et al. introduced a new method that integrates GPS and INS data for navigation in mobile robots. To overcome the complexity of the method, the RBF network used [15].

In some studies, different neural network structures have been carried out for the wall tracking robot. Budianto et al. presented a comparison of two artificial intelligence methods for a robot that follows the wall. For this purpose, they compared back propagation based neural network and fuzzy logic methods. The robot has three input variables and two output variables. The mobile robot is designed to prevent collisions with obstacles such as walls or other mobile robots. They found that the motion of the robot using neural network is faster than the fuzzy logic controller [16]. Dash et al. attempted to use a new neural network training algorithm based on gravity search (GS) and feed forward neural network (FFNN) for automatic robot navigation of the wall following mobile robots. The GS strategy has been used to adjust the optimum weight set of FFNN to improve neural network performance [17]. Karakus and Er used a probabilistic neural network (PNN) structure for robot navigation tasks. They compared this method with the results of the PNN, logistics perceptron, multilayer perceptron, experts and Elman neural networks. It has been observed that PNN has the best classification accuracy with 99.635% accuracy using the same data set [18]. Larasati et al. used artificial neural network for wall and object tracking behaviors of a robot with data from 4 distance sensors. The contribution of the method they developed is simplicity [19]. Dash et al. proposed a multi-layered neural network for trajectory control with the data of the wall tracking robot. It achieved successful performance outcomes based on different

learning rates for the network and the number of neurons in the hidden layer [20]. Faisal et al. used the wireless control approach to control the mobile robot swarm in the warehouse with static and dynamic obstacles. Using fuzzy logic techniques, they developed the online navigation technique for the mobile robot in an unknown dynamic environment [21]. Singh et al. used the MLP network to navigate the mobile robot in a dynamic environment. The network they developed determines the area where there will be no collision among the five segments and also enables the target to be reached by controlling the speed of the robot [22].

In addition to ANNs, some researchers used genetic and fuzzy approaches for navigation control. Mucientes et al. described the automated design of the fuzzy controller using genetic algorithms to implement wall tracking behavior in a mobile robot [23]. The algorithm is based on an iterative rule-learning approach. Desouky and Schwartz membership functions focus on setting the set of fuzzy control parameters such as scaling factors and control configuration for robot navigation problems [24].

In this study, MLP and RBF network structures have been developed for navigation of a mobile robot. These structures were tested with different parameters and an optimum controller was obtained. For this purpose, the sensor values obtained from the navigation data set of the SCITOS G5 wall tracking robot from the UCI data platform were used.

The contribution of the study to the literature is the detailed comparison of MLP and RBF network structures in robot navigation. Also, the lowest error value was tried to be obtained with different parameter combinations for both MLP and RBF networks. It was determined as a result of the test studies that the RBF network structure performance better than MLP network.

2. Materials and Methods

In this study, artificial neural network models have been developed for the navigation of a mobile robot. Artificial Neural Networks (ANN) is one of the artificial intelligence techniques that model the human brain. ANNs can produce highly effective results in model estimation processes. It can also manage linear / nonlinear processes by learning from data relationships and generalizing to unknown situations. In this study, MLP and RBF network structures which are best known ANN networks were used for navigation of the robot named SCITOS G5.

2.1. Data Information of SCITOS G5

In this study, an open-access ready dataset was used for the SCITOS G5 robot [25]. This data set contains location data containing the distance from the wall thanks to 24 sensors around a mobile robot and decision information about the direction of the robot with this data. Accordingly, there are 24 inputs and 1 output information. The data set contains 4 different directions and these directions are represented by a number. In the data set, there are 5456 data obtained from the sensors. Data set content is given in Table 1.

Table 1: Data Set Content

Output value	Direction command	Data Number
1	Move forward	2205
2	Slight right turn	826
3	Slight left turn	2097
4	Sharp right turn	238
TOTAL		5456

2.2. Data Set Normalization

The approach that makes the nonlinear feature of neural networks meaningful is the normalization process. In this study, Min-Max normalization, which is frequently used in the literature, was preferred. Min-Max normalization process is as given in Equation 1.

$$X' = \frac{X_i - X_{min}}{X_{max} - X_{min}} \quad (1)$$

Where X_{max} maximum value of the data, X_{min} minimum value of the data and X_i actual value of the data.

2.3. Multi-Layer Perceptron (MLP)

One of the most known and used artificial neural network topologies is a multi-layer perceptron. It is a powerful modeling tool that implements a supervised training procedure using data samples with known outputs [26]. The most important learning algorithm used in MLP is back-propagation method. It was developed by Rumelhart et al. [27]. The back-propagation algorithm uses a learning method called “delta learning rule”. MLP works effectively especially in classification and generalization situations. Most generally, an MLP network structure is given in Figure 1.

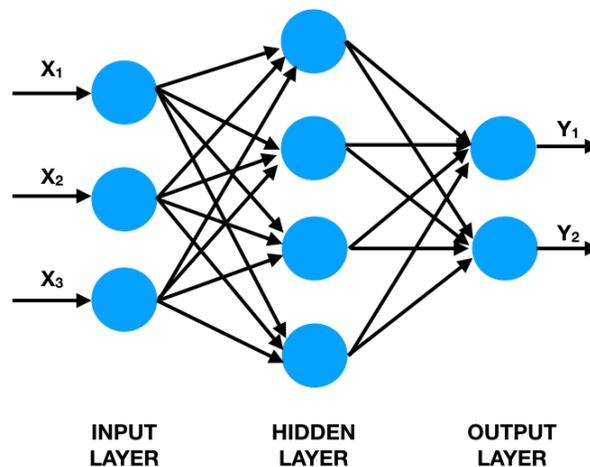


Figure 1: Feed-forward multi-layer neural network.

A MLP is a network structure that combines with additional sensors stacked in several layers to solve complex problems. For each connection, there are different weights. Each layer can have

multiple perceptron. Thus, the multi-layered perceptron can quickly become a very complex network structure.

In this study, a MLP network structure with 24 inputs and 1 output has been created. There are two hidden layers and 40 neurons in each hidden layer. As the activation function, combinations of tangent sigmoid and logarithmic sigmoid function were used. The network structure is as given in Figure 2 in MATLAB NN Toolbox.

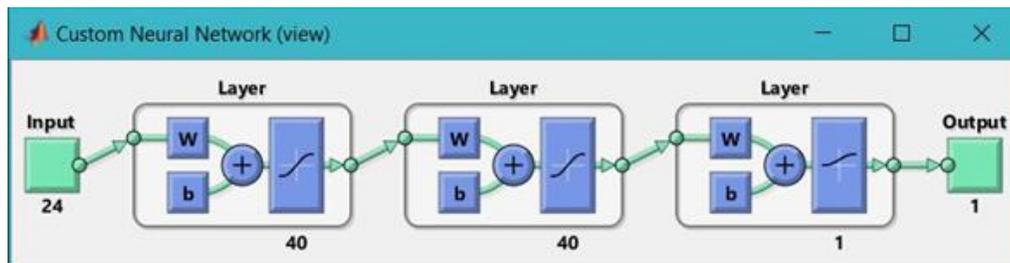


Figure 2: MLP structure in MATLAB.

2.4. Radial Basis Function (RBF)

Radial Basic Function (RBF) is another popular architecture used in ANN. RBF neural network are controlled neural networks similar to MLP [28]. RBF networks are identical in structure to the MLP structure with a single hidden layer. For this reason, the MLP structure in Figure 1 also expresses the RBF structure in one-layer form. RBF networks differ from other neural networks due to their universal approach and faster learning speeds [29]. An RBF network is a feed forward neural network type consisting of three layers, the input layer, the hidden layer, and the output layer [30]. The input layer consists of the input data. The hidden layer converts the data from the input layer into the hidden layer using the nonlinear activation function. The linear output layer produces the response of the network.

The hidden layer consists of many RBF neurons, and the hidden layer nodes are calculated from the Euclidean distance between the center and the network input vectors. Despite the large number of RBF, the Gauss function as RBF is used as RBF NN in applications. If a Gauss function is used as a hidden layer in the neuron activation function, the hidden layer value of the neuron for each input data point is calculated from the following equation:

$$\phi_j = e^{-\left[\frac{|x-m_j|}{\sigma_j}\right]^2} \quad (1)$$

where ϕ_j is the Euclidean distance between x input data and the j th pattern of the hidden layer, m_j is the j th pattern of the center of RBF, and σ_j is the width of the j pattern of the RBF. The network output parameter is calculated as below:

$$y = \sum_{j=1}^m w_j \phi_j \quad (2)$$

The RBF network structure has 24 inputs and 1 output. 500 neurons were not used in the hidden layer. RBF network structure was created in MATLAB NN Toolbox and given in Figure 3.

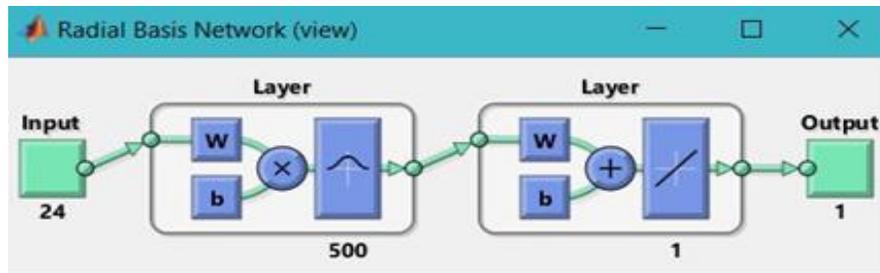


Figure 3: RBF structure.

3. Results and Discussions

Network structures have been created and tested in a system with Windows 10 operating system, 2 GB RAM and i5 processor and Matlab 2016b NN Toolbox. The initial values of the weights are randomly assigned between 0 and 1. The established MLP and RBF networks have been tested at different parameter values. The difference between the error / performance results obtained by the use of these parameters and the performance of the two network structures are compared. Comparison was made using MSE (Mean Square Error) as the performance parameter. For both networks, 75% of the data set was used for training and 25% for testing.

3.1. Training Results of the MLP Structure

Network structure is tested for different combinations of hidden layer number, number of neurons in hidden layer, activation functions used, learning rate, education algorithm parameters. In this context, 400 different combinations have been tried. 6 of these 400 combinations including the best result are given in Table 2 as an example.

Table 2: MLP structure test results.

Hidden Layer	Number of Neurons in Hidden Layer	Activation Function (hidden layer-output layer)	Learning Rate	Learning Algorithm	MSE
1	30	logsig-logsig	0.9	trainlm	0.0089
1	40	tansig-tansig	0.8	trainlm	0.0043
1	20	tansig-tansig	0.4	trainrp	0.0231
2	10 - 10	logsig-logsig	0.1	trainlm	0.0099
2	20 - 20	logsig-logsig	0.5	trainrp	0.0090
2	40 - 40	tansig-tansig	0.7	trainlm	0.00011

As shown in Table 2, the best performance was obtained for the 2 hidden layers, 40 neurons, the tansig in the hidden layer and the logsig function in the output layer, 0.7 learning rate and the trainlm learning algorithm. The MSE value for this structure was 0.00011.

3.2. Training Results of the RBF Structure

RBF network structure was tested with spread value and different neuron numbers. 400 different combinations were used. Sample values including the best result from the test results are given in

Table 3. Accordingly, the best performance was obtained with 0.5 spread value and 500 neurons. In this case, MSE value was obtained as 0.0210689.

Table 3: RBF structure training results.

Spread Value	Number of Neurons	MSE	Spread Value	Number of Neurons	MSE
0.1	100	0.0834156	5	100	0.0616567
0.1	500	0.0330311	5	500	0.0339581
0.2	100	0.0719684	10	100	0.0620606
0.2	500	0.0283962	10	500	0.0341679
0.5	100	0.0564102	50	100	0.0622398
0.5	500	0.0210689	50	500	0.0463390
1	100	0.0495453	100	100	0.0630901
1	500	0.0215438	100	100	0.0551366

3.3. MLP and RBF Comparison Results

Network structures were tested using the parameters that obtained the best MSE value during the training phase in both networks. Training and test results are given in Table 4. Accordingly, MSE value of MLP network structure was obtained as 0.0001 during the training phase. In contrast, the RBF value was obtained 0.0210689. When the network structures were tested, the MSE value of MLP and RBF was 0.114047 and 0.062893, respectively. In addition, as a result of the regression analysis for both topologies, R values were obtained for MLP and RBF, respectively 0.99958 and 0.99726. These values show that the relationship between the predictions of the network and the actual values is very close.

Table 4: Comparison results in terms of training and test phase

Topology	MSE(TRAINING)	MSE(TEST)
MLP	0.0001	0.114047
RBF	0.0210689	0.062893

According to these results, although the MSE value of MLP during the training phase is better than RBF, the RBF network has shown a higher performance in the test results. This is because the data used in this study is distance data and the RBF network is a system that operates based on the distance of the data from the centers, thus increasing the success rate in predicting real data.

According to the results obtained, robot navigation was performed using MLP and RBF network structures. The best network performance was evaluated with different parameter values. According to the results obtained, it was revealed that the RBF network structure showed a better performance during the test phase.

4. Conclusions

In this study, MLP and RBF network structures have been developed to control the navigation of a wall tracking robot (SCITOS G5). The parameters such as the number of hidden layers in the networks, the number of neurons in the hidden layer, the learning and transfer function, learning

rates, and the propagation value were tested with different combinations and the highest performance network structure was investigated. According to the simulation results, it was seen that the RBF network structure gave a better result than the MLP network structure. In the continuation of this study, it is planned to test the network structure obtained on the system and create a genetic algorithm based structure for optimization of network parameters.

References

- [1] Sigalas, M. Baltzakis, H., & Trahanias, P. Temporal gesture recognition for human-robot interaction. Month, 2010.
- [2] P. C. Giulianotti. Robotics in general surgery. *Robotics in General Surgery*, 138(July 2003), 2014, 1–511.
- [3] Gul, F., Rahiman, W., & Nazli Alhady, S. S. A comprehensive study for robot navigation techniques. *Cogent Engineering*, 6(1), 2019, 1–25.
- [4] M. Knudson and K. Tumer. Adaptive Navigation for Autonomous Robots, *Robotics and Autonomous Systems*, 59, 2011, 410–420.
- [5] Eski, I., & Yildirim, Ş. Design of neural network control system for controlling trajectory of autonomous underwater vehicles. *International Journal of Advanced Robotic Systems*, 11(1), 2014.
- [6] Fabiani, P., Fuertes, V., Piquereau, A., Mampey, R., & Teichteil-Königsbuch, F. Autonomous flight and navigation of VTOL UAVs: from autonomy demonstrations to out-of-sight flights. *Aerospace Science and Technology*, 11(2–3), 2007, 183–193.
- [7] N. Najmaei and M. R. Kermani, "Applications of Artificial Intelligence in Safe Human–Robot Interactions," in *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 41, no. 2, 2011, 448–459.
- [8] Mendes Júnior, J. J. A., Pires, M. B., Vieira, M. E. M., Okida, S., & Stevan Jr, S. L. Neural Network to Failure Classification in Robotic Systems. *Brazilian Journal of Instrumentation and Control*, 4(1), 1, 2016.
- [9] Kruse, T., Pandey, A. K., Alami, R., & Kirsch, A. Human-aware robot navigation: A survey. *Robotics and Autonomous Systems*, 61(12), 2013, 1726–1743.
- [10] Panigrahi, P. K., Ghosh, S., & Parhi, D. R. Intelligent Learning and Control of Autonomous Mobile Robot using MLP and RBF based Neural Network in Clustered Environment. 5(6), 2014, 313–316.
- [11] Shinzato, P. Y., Fernandes, L. C., Osorio, F. S., & Wolf, D. F. Path recognition for outdoor navigation using artificial neural networks: Case study. *Proceedings of the IEEE International Conference on Industrial Technology*, <https://doi.org/10.1109/ICIT.2010.5472489>, 2010, 1457–1462.
- [12] Shinzato, P. Y., & Wolf, D. F. A road following approach using artificial neural networks combinations. *Journal of Intelligent and Robotic Systems: Theory and Applications*, 62(3–4), <https://doi.org/10.1007/s10846-010-9463-2>, 2011, 527–546.
- [13] Dezfoulian, S. H., Wu, D., & Ahmad, I. S. A generalized neural network approach to mobile robot navigation and obstacle avoidance. *Advances in Intelligent Systems and Computing*, 193 AISC (VOL. 1), 2013.
- [14] Wang, X., Hou, Z. G., Lv, F., Tan, M., & Wang, Y. Mobile robots' modular navigation controller using spiking neural networks. *Neurocomputing*, 134, <https://doi.org/10.1016/j.neucom.2013.07.055>, 2014, 230–238.
- [15] Malleswaran, M., Angel Deborah, S., Manjula, S., & Vaidehi, V. Integration of INS and GPS using radial basis function neural networks for vehicular navigation. 11th International Conference on Control, Automation, Robotics and Vision, ICARCV 2010, (December), <https://doi.org/10.1109/ICARCV.2010.5707295>, 2010, 2427–2430.
- [16] Budianto, A., Pangabidin, R., Syai'In, M., Adhitya, R. Y., Subiyanto, L., Khumaidi, A., Soelistijono, R. T. Analysis of artificial intelligence application using back propagation neural

- network and fuzzy logic controller on wall-following autonomous mobile robot. 2017 International Symposium on Electronics and Smart Devices, ISESD 2017, 2018-January (1), .
<https://doi.org/10.1109/ISESD.2017.8253306>, 2017, 62–66.
- [17] Dash, T., Nayak, T., & Swain, R. R. Controlling wall following robot navigation based on gravitational search and feed forward neural network. ACM International Conference Proceeding Series, 26-27-February-2015, <https://doi.org/10.1145/2708463.2709070>, 2015, 196–200.
- [18] Ozbay Karakus, M., & Er, O. Learning of Robot Navigation Tasks by Probabilistic Neural Network, <https://doi.org/10.5121/csit.2013.3803>, 2013, 23–34.
- [19] Larasati, N., Dewi, T., & Oktarina, Y. Object Following Design for a Mobile Robot using Neural Network. Computer Engineering and Applications Journal, 6(1), <https://doi.org/10.18495/comengapp.v6i1.189>, 2017, 5–14.
- [20] Dash, T., Soumya, R. S., Nayak, T., & Mishra, G. (2015). Neural network approach to control wall-following robot navigation, ICACCCT, 2014.
- [21] Faisal, M., Hedjar, R., Al Sulaiman, M., & Al-Mutib, K. Fuzzy Logic Navigation and Obstacle Avoidance by a Mobile Robot in an Unknown Dynamic Environment. International Journal of Advanced Robotic Systems, 2013.
- [22] Singh, N. H., & Thongam, K. Mobile Robot Navigation Using MLP-BP Approaches in Dynamic Environments. Arabian Journal for Science and Engineering, 43(12), <https://doi.org/10.1007/s13369-018-3267-2>, 2018, 8013–8028.
- [23] Mucientes, M., Moreno, D. L., Bugarín, A., & Barro, S. Design of a fuzzy controller in mobile robotics using genetic algorithms. Applied Soft Computing Journal, 2007, 540–546.
- [24] S. F. Desouky and H. M. Schwartz, "Genetic based fuzzy logic controller for a wall-following mobile robot," 2009 American Control Conference, St. Louis, 2009, 3555-3560.
- [25] A. Frank, A. Asuncion, "UCI Machine Learning Repository," 2010.
- [26] Bishop, C.M., Neural Networks for Pattern Recognition. Oxford University Press Inc. New York, NY, ISBN: 0198538642, 1995.
- [27] Kashaninejad M. ve Dehghani A.A., Kashiri M., Modeling of wheat soaking using two artificial neural networks (MLP and RBF), Journal of Food Engineering, 2009, 602–607.
- [28] Celikoglu H.B. ve Cigizoglu H.K., "Modelling public transport trips by radial basis function neural networks," Mathematical and Computer Modelling, 2007, 480–489.
- [29] Yu, B., He, X., Training radial basis function networks with differential evolution. In: Proceedings of IEEE International Conference on Granular Computing, Atlanta, USA, 2006.
- [30] Hwang Y.S. ve Bang S.Y., "An Efficient Method to Construct a Radial Basis Function Neural Network Classifier," 1996.

*Corresponding author.

E-mail address: eakdogan@yildiz.edu.tr