10.1515/acsc-2015-0002

*Archives of Control Sciences*
*Volume* 25(LXI), 2015
*No.* 1*, pages 21–34*

# Navigation of autonomous mobile robot using different activation functions of wavelet neural network

PRATAP KUMAR PANIGRAHI, SARADINDU GHOSH and DAYAL R. PARHI

An autonomous mobile robot is a robot which can move and act autonomously without the help of human assistance. Navigation problem of mobile robot in unknown environment is an interesting research area. This is a problem of deducing a path for the robot from its initial position to a given goal position without collision with the obstacles. Different methods such as fuzzy logic, neural networks etc. are used to find collision free path for mobile robot. This paper examines behavior of path planning of mobile robot using three activation functions of wavelet neural network i.e. Mexican Hat, Gaussian and Morlet wavelet functions by MATLAB. The simulation result shows that WNN has faster learning speed with respect to traditional artificial neural network.

**Key words:** autonomous mobile robot, activation functions, obstacle avoidance, path planning, wavelet neural network.

## 1. Introduction

Path planning in an unknown environment without collision with obstacles is the key issue of any autonomous mobile robot. The autonomous mobile robot has many applications in the real world where human being cannot work safely. Some of the important applications of autonomous mobile robot are

- Automatic driving of vehicles.
- Assistance to the disabled persons.
- Cleaning railway platform and other hazardous site intelligently.
- Performing complex tasks in remote planet.
- In chemical and nuclear plants etc.

P.K. Panigrahi, the corresponding author, is with Department of Electrical Engineering, Padmanava College of Engineering, Rourkela, India, e-mail: pratapkumarpanigrahi@gmail.com. S. Ghosh is with Department of Electrical Engineering, National Institute of Technology, Dargapur, India, e-mail: sghosh.ee@gmail.com. D.R. Parhi is with Department of Mechanical Engineering, National Institute of Technology, Rourkela, India, e-mail: dayalparhi@yahoo.com.

Received 07.07.2014. Revised 24.11.2014.

Path planning may be divided into two categories; one is in the presence of a known environment of the workspace and the other is in an uncertain environment where position and the shape of the obstacles are not known a priori. The proposed path planning method is based on the latter category. The main feature of autonomous mobile robot is to generate collision free path without assisting human intervention. To achieve the above characteristics, the mobile robot should extract complete information about the working environment using ultrasonic sensors or infrared sensors and on-board vision systems. In this work, it is assumed that the robot is ideal, i.e. there is no slip which means that the robot is a rigid body and the robot acceleration is such that the forces acting in the longitudinal and lateral direction do not exceed the static friction between the surface of the wheels and the ground. The approach proposed in this paper is suitable for controlling mobile robot in unstructured static obstacle environments.

## 2. Previous work

In the recent years, many researchers have investigated different soft computing techniques in the area of navigation of mobile robot in unknown environment. Abhiyev et al. [1] have proposed a classical and fuzzy logic based algorithm for navigation mobile robot to avoid static obstacles in the environment. Raghuraman et al. [2] have proposed a sensor based mobile robot navigation technique using fuzzy logic technique in indoor environment. The approach is focused on two behaviors such as obstacle avoidance and target steering. Harisha et al. [3] have implemented a microcontroller based four-wheeled mobile robot system consisting of various I.R sensors for path planning in different environment conditions. The fuzzy logic technique was used in MATLAB environment to verify the result with experimental setup. Joshi et al. [4] have developed a neural network based controller for navigation of autonomous mobile robot. Back propagation algorithm was utilized to train a multilayered perceptron neural network to obtain the left and right wheel velocity of mobile robot. Janglova [5] has proposed a four layered feed-forward neural network for path planning and intelligent control of mobile robot in a partially unstructured environment. The back propagation algorithm was used to train the neural network. Jiang et al. [6] have proposed a fuzzy neural network and fuzzy logic control for path tracking of mobile robot in unknown dynamic environment. An improved back-propagation algorithm was used for training the network. However, the approach needs many attempts to tune the parameters of the fuzzy controller. The fuzzy logic controller is basically depends on human experience to extract heuristic rule-base without knowing the analytical model of the system.

Therefore, it is very difficult to control the mobile robot motion in a complex environment. In the literature, various neural network based controller are proposed in path planning problem. For autonomous mobile robot, there are two key issues in selecting training algorithm of neural network. One is faster in convergence while the other is function approximation accuracy. The main drawback of back propagation algorithm is slow rate of convergence and uncertain to ensure global convergence. However, there are very few literatures which address the convergence of neural network algorithm. In addition to the above there are some other techniques like artificial potential field, vector field histogram etc. have been used to solve this problem. In artificial potential field method [7-9] the robot is attracted to the target and is pushed away from the obstacles; the robot motion is determined by the resultant virtual force. Baker [10] has proposed a neuro-fuzzy rule-base technique for navigation of mobile robot in unknown environment. In his research work neural network was used to optimize the activation rules. A fuzzy logic controller has been implemented for path planning of robot. The performance is verified with fuzzy logic technique algorithm. Gigras et al. [11] have proposed an Ant colony optimization algorithm to make collision free motion with shortest navigational path and time. The strength of the algorithm was verified using simulation. Castro et al. [12] have proposed an algorithm for navigation of mobile robot in poorly structured environment using Probabilistic neural network environment to control the motion of robot during navigation. The effectiveness was verified using experimental set-up and MATLAB simulation mode. Buniyamin et al. [13] have proposed an algorithm for path planning of mobile robot based on the measurement of nearest obstacles using sensors. The algorithm was developed to minimize the use of outer perimeter of obstacle by considering a few important points on the obstacle's perimeter to generate a complete path from starting point to a target. Panigrahi et al. [14] have implemented a Radial Basis Function (RBF) neural network based path planning controller for navigation of mobile robot in unknown environment. The aim of the algorithm is to make collision free path when a mobile robot is allowed to move from starting point to a target point. The efficiency, accuracy and speed of convergence is verified using MATLAB simulation with different types of static obstacle environments. The main aim of this proposed work is to compare the performance of WNN with the existing RBFN controller for navigation of mobile robot using different activation functions of wavelet neural network in terms of training architecture, learning speed and guarantee of convergence.

## 3. Configuration of mobile robot

Let us consider a two dimensional workspace for motion of mobile robot in Fig. 1 (a). The aim of the robot is to move from an initial position to a target position with target angle (θ) by avoiding a set of obstacles without collision. Fig. 1 (b) represents Khepera II mobile robot in which there are 8 ultrasonic sensors in three different directions for detecting obstacles. Since there may be more than one obstacle in the environment, therefore three sensors each having $45^0$ range of angle of detection are considered in three different directions to know the obstacles distance in the left (LOD) $x_1 = \min\{d_1, d_2, \ldots, d_i\}$.

Obstacle distance in the front (FOD) $x_2 = \min\{d_1, d_2, \ldots, d_j\}$ and obstacle distance in the right (ROD) $x_2 = \min\{d_1, d_2, \ldots, d_k\}$ where $d_i$, $d_j$ and $d_k$ are distances of obstacles, $i$, $j$ and $k$ are positive integers.

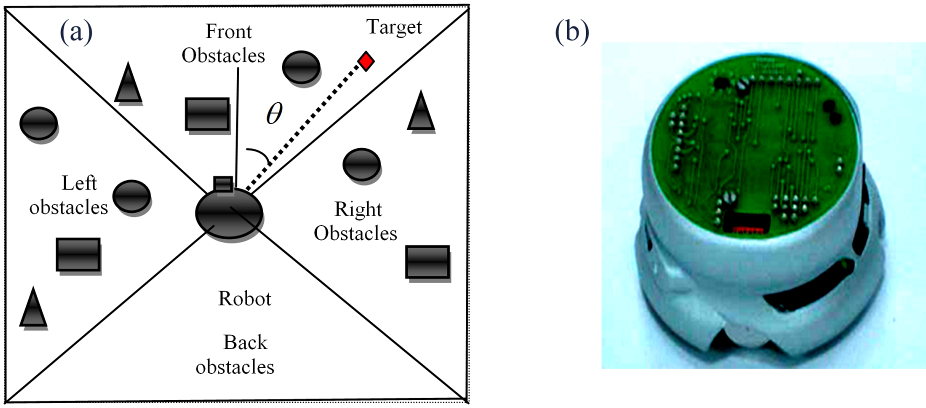



Figure 1: (a) Configuration of mobile robot. (b) Khepera II Robot.

To generate a collision free path, the mobile robot may have to move along a straight line path or to make a turn depending on the situations. In the proposed work four inputs are considered for mobile robot such as LOD, FOD, ROD and Target Angle (TA) and one output called steering angle (SA). During the motion of mobile robot the target angle is given at all time to the robot. The distance from a different set of obstacles is sensed using ultrasonic sensors which are being mounted in the front of the robot. The environment contain static obstacles Kinematic model of mobile robot.

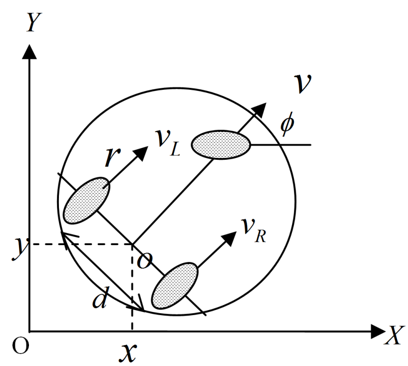



Figure 2: Kinematic model of mobile robot.

Consider a mobile robot kinematic model platform shown in the Fig. 2 which consist of two driving wheels mounted on a shaft with two independent actuators and one free wheel in the front for support. The navigation of mobile robot is controlled by changing the relative velocities of two rear driving wheels i.e. left wheel and right wheel. During kinematic analysis of mobile robot it is assumed that the robot is non-holonomic in nature i.e. the whole body of robot is rigid and motion occurs without sliding.

Let $v$ be the velocity of robot at centroid position; $v_L$ is the left wheel velocity; $v_R$ is the right wheel velocity; $r$ is the diameter of both the wheels; $d$ is the separation between two wheels; $x$ and $y$ are positions of mobile robot and $\phi$ is the steering angle/orientation of mobile robot. According to the kinematics of a rigid body, the motion of mobile robot can be expressed using equation (1) and (2)

$$v_R = w_R r \tag{1}$$

$$v_L = w_L r \tag{2}$$

where $w_L$ and $w_R$ are angular velocities of left and right wheel respectively. Let be the angular velocity of robot at the centroid. Then we can obtain

$$w = \frac{(v_R - v_L)}{d},$$

$$v = \frac{(v_R + v_L)}{2}.$$

Combining equation (1) and (2) we can write

$$w = \frac{(w_R - w_L)r}{d},$$

$$v = \frac{(w_R + w_L)r}{2}.$$

Now the dynamic function can be written as

$$\dot{x} = v\cos\theta, \quad \dot{y} = v\sin\theta, \quad \dot{\theta} = w.$$

Hence, we can derive

$$w_R = \frac{2v}{r} + \frac{w}{r},$$

$$w_L = \frac{2v}{r} - \frac{w}{r}.$$

Now the kinematic model of mobile robot in state space model can be written as

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ w \end{bmatrix} \tag{3}$$

The controlled variable of the model are position and orientation of mobile robot which is obtained by changing angular velocity of left and right wheels.

## 4. Wavelet neural network (WNN)

Wavelet neural network shown in Fig. 3 is a class of network whose architecture is similar to radial basis function (RBF) neural network but instead of radial basis function it uses different wavelet functions i.e. Gaussian wavelet, Morlet and Mexican Hat etc. as the activation function in the hidden nodes. The parameters such as center vector and width in RBFN are replaced by translation vector ($a$) and dialation vector ($b$) respectively. It can be used as a good function approximation. Like RBFN, WNN also calculate the Euclidean distance between input ($x_j$) and translation vector ($a_j$) with dialation vector ($b_j$) as the weighted scaling factor of each distance vector.

For input vector $x_i = (x_1, x_2, \ldots, x_n)$ the output of the hidden layer is calculated as

$$\theta(j) = \psi \sum_{i=1}^{n} \frac{(w_{ij}x_i - a_j)}{b_j} \tag{4}$$

where $j = 1, 2, \ldots, m$, $m$ is number of hidden layers, $\theta(j)$ is value of the output in the node of the hidden layer, $\psi$ is wavelet function, $w_{ij}$ is weight of the connection between input and hidden layer, $a_j$ is shift factor/translation vector, $b_j$ is stretch factor/dialation vector for $\phi_j$. The output layer output can be calculated as

$$\phi(k) = \sum_{j=1}^{m} w_{jk}\theta(j) \tag{5}$$

where $w_{jk}$ is weight of the connection between the hidden layer and the output layer.

In the proposed WNN model four sensor inputs are applied considering three wavelet functions as activation function separately. The learning rate $\eta$ is fixed as 0.01 with 10 hidden nodes.

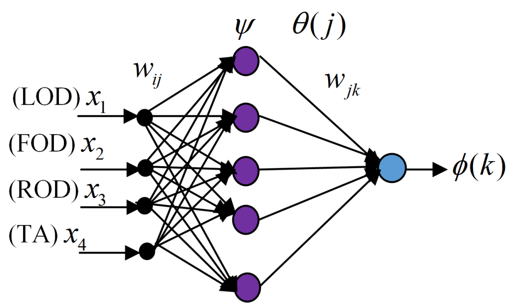



Figure 3: Wavelet neural network.

The WNN weights and parameters are updated according to the error $e = \sum_{i=1}^{m} (\phi_n(k) - \phi(k))$ where $\phi(k)$ is the predicted steering angle, $\phi_n(k)$ is the expected output steering angle of robot and

$$w_{ij}^{(i+1)} = w_{ij}^{(i)} + \Delta w_{ij}^{(i+1)}$$
$$w_{jk}{}^{(i+1)} = w_{jk}{}^{(i)} + \Delta w_{jk}{}^{(i+1)}$$
$$a_j{}^{(i+1)} = a_j{}^{(i)} + \Delta a_j{}^{(i+1)}$$
$$b_j{}^{(i+1)} = b_j{}^{(i)} + \Delta b_j{}^{(i+1)}$$

The values $\Delta w_{ij}^{(i+1)}$, $\Delta w_{jk}^{(i+1)}$, $\Delta a_j^{(i+1)}$, $\Delta b_j^{(i+1)}$ are calculated by the predictor error of the network where

$$\Delta w_{ij}^{(i+1)} = -\eta \frac{\partial e}{\partial w_{ij}^{(i)}}$$
$$\Delta w_{jk}^{(i+1)} = -\eta \frac{\partial e}{\partial w_{jk}^{(i)}}$$
$$\Delta a_j^{(i+1)} = -\eta \frac{\partial e}{\partial a_j^{(i)}}$$
$$\Delta b_j^{(i+1)} = -\eta \frac{\partial e}{\partial b_j^{(i)}}.$$

The following steps are involved in training of WNN network:

Step I: Initialization of WNN Network:
Initialize the connecting weights $w_{ij}$, $w_{jk}$ translation factor $B_j$ and scale factor $a_j$ randomly and also the learning rate $\eta$.

Step II: The original data is quantified and normalized and then separated into two groups. i.e. training set and testing set.

Step III: Compute the output steering angle using WNN i.e. $\phi_n$.

Step IV: Compare the output with the target and calculate the error $e$.

Step V: Update the parameters of mother wavelet function and weights of the network according to the predicted value of the network as close to the actual values.

Step VI: If the target value of the error is not reached at the $n$th epoch, find the difference between the target and the output obtained and then compute $\Delta w_{ij}^{(i+1)}$, $\Delta w_{jk}^{(i+1)}$, $\Delta a_j^{(i+1)}$ and $\Delta b_j^{(i+1)}$ using the learning rate ($\eta$) and the momentum factor ($\alpha$).

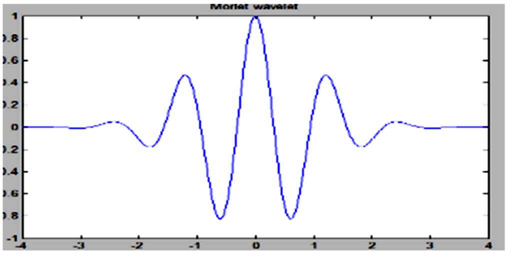
Step VII: Repeat steps I-VI till convergence is achieved.

## 5. Different types of activation functions in WNN

There are three types of activation functions which are used in hidden layer of wavelet Neural Network (WNN) namely

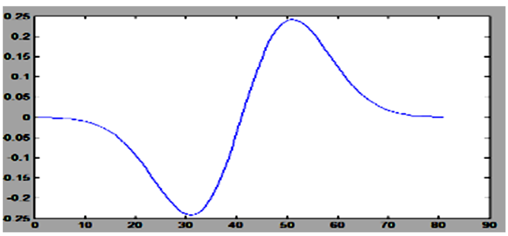
(i) **Morlet wavelet function**

$$\psi(t) = \frac{2}{\sqrt{3\sigma\pi^{\frac{1}{4}}}}(1 - \frac{t^2}{\sigma^2})e^{-\frac{t^2}{\sigma^2}} \tag{6}$$



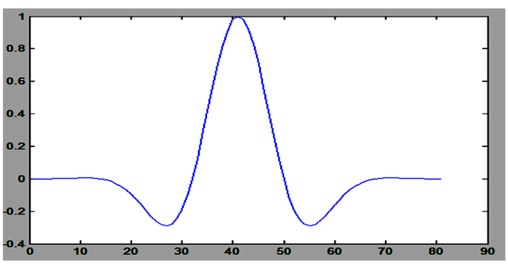

(ii) **Gaussian wavelet function**

$$\psi(t) = \frac{t}{\sqrt{2\pi}}e^{-\frac{t^2}{2}} \tag{7}$$



(iii) **Mexican hat wavelet function**

$$\psi(t) = \cos(1.75t)e^{-\frac{t^2}{2}} \tag{8}$$



## 6. Analysis of training architecture

In this proposed work about 100 different training patterns are provided to train the neural network for planning of mobile robot in different cluster environment.

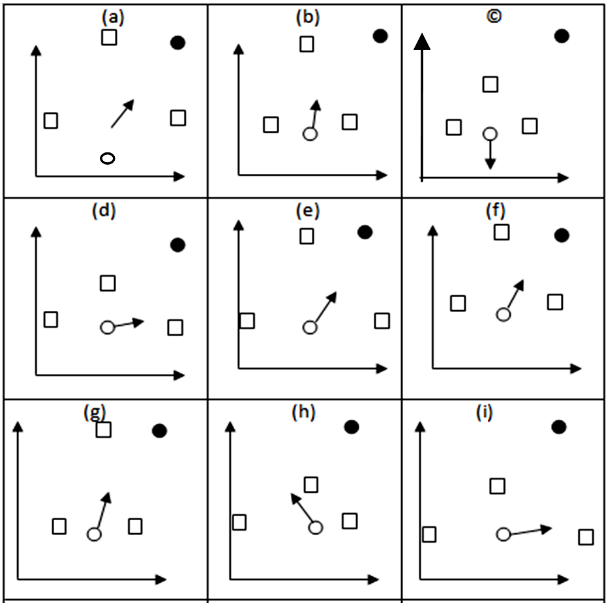



Figure 4: Training architecture.

For each set of inputs there is a corresponding steering angle of the mobile robot. Some of the training scenarios are presented in the Fig. 4. For example in Fig. 4(a) if the robot is facing a set of obstacles at a distance of 10 cm to the left, 10 cm to the front and 10 cm to the right than a change in steering angle required to avoid the obstacle is $45^0$ with respect to x-axis. During training four input patterns are fed to the controller which comprises of Left Obstacle Distance (LOD), Front Obstacle Distance (FOD), Right Obstacle Distance (ROD) and Target Angle (TA).The navigation strategy used in this work uses the sensorial information as input variables to all the three algorithm. Using target position and current position of mobile robot, the neural network algorithm generates the control action, i.e. forward right turn or left turn to make a collision free path to reach the desired target. The target plays an important role in deciding the rotating SA at different positions.

## 7. Simulation results and discussion

The proposed path planning of mobile robot is implemented based on two behaviors, such as obstacle avoidance and target seeking in MATLAB environment with Intel

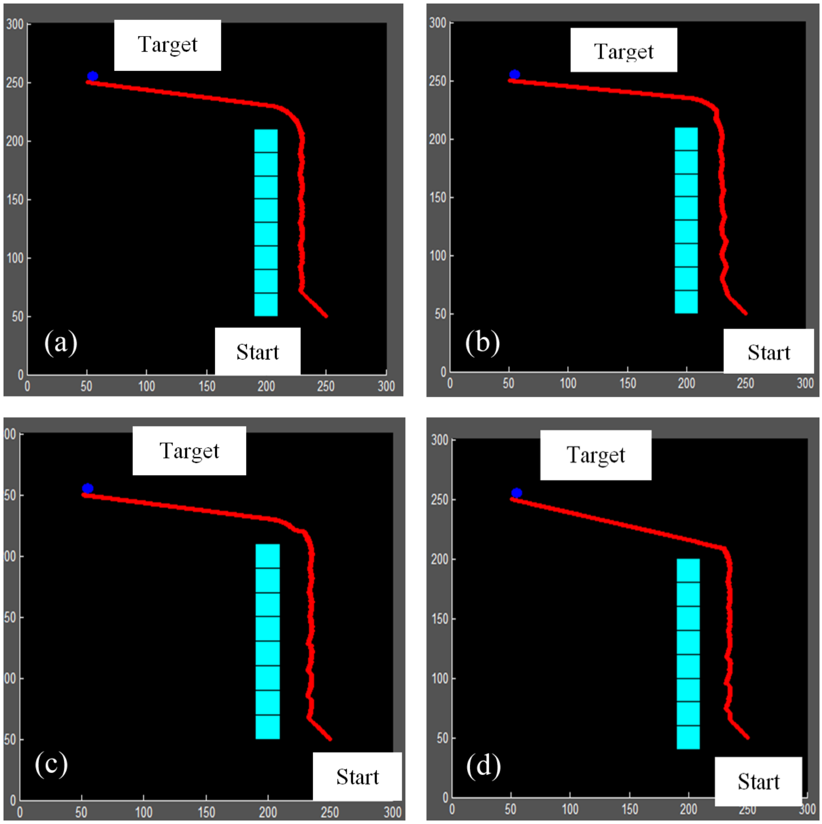



Figure 5: Robot start position (250, 50), target (50,250) unit.

Table 1: Comparative results of Fig. 5(a), 5(b), 5(c) and 5(d)

| Name of the algorithm | Number of hidden nodes | Simulation time [min] |
|---|---|---|
| Morlet WNN Fig. 5(a) | 10 | 2.01 |
| Mexican hat WNN Fig. 5(b) | 10 | 2.03 |
| Gaussian WNN Fig. 5(c) | 10 | 2.08 |
| RBFN Fig. 5(d) | 10 | 6.50 |

Pentium IV, Dual core, 1.8 GHz processor. Three simulation scenarios with multiple obstacles at different positions are presented in six different working environments using different activation functions of WNN in Fig. 5(a)-5(c) and 6(a)-6(c) with a work area of size 300×300 square unit. Each scenario the three proposed models are applied, the performance of the models is considered on the basis of smoothness of trajectory,

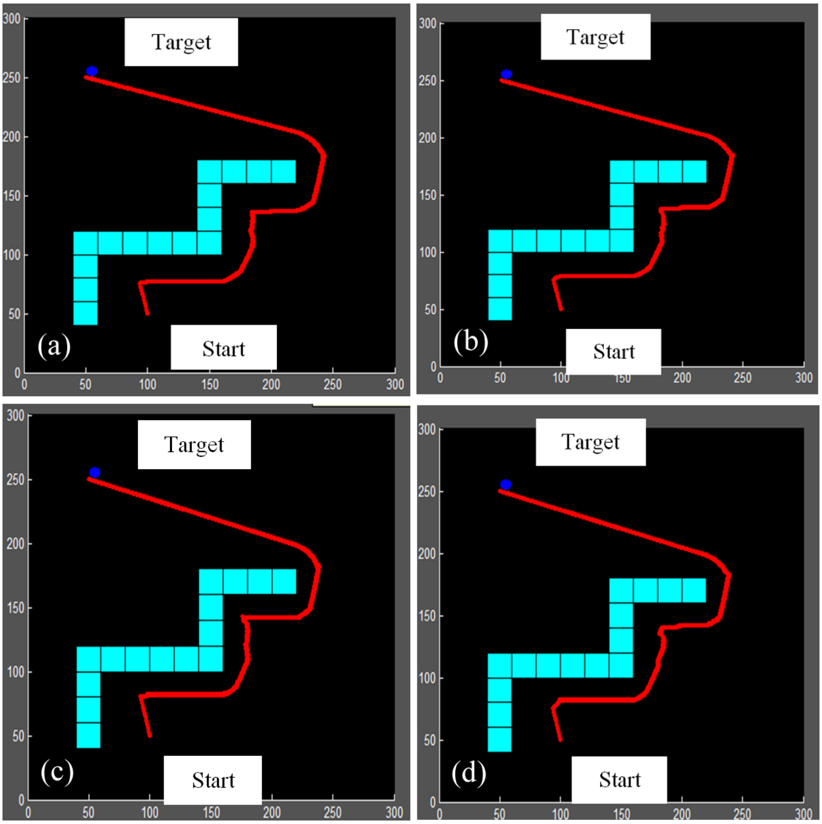



Figure 6: Robot start position (100, 50), target (50,250) unit.

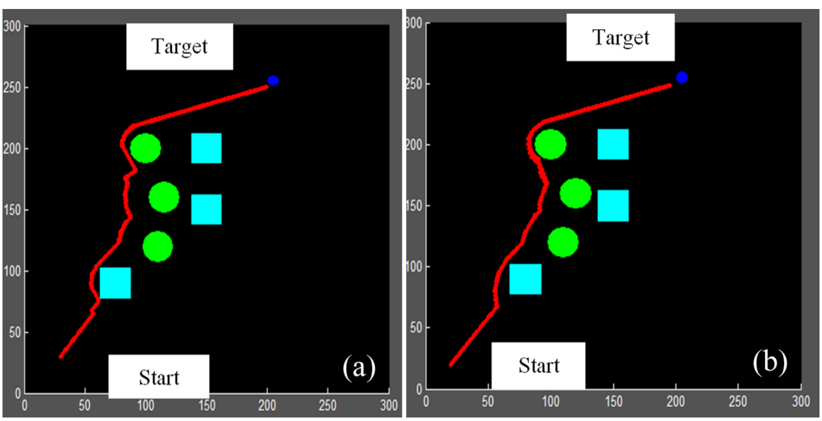



Figure 7: (a) Path obtained by Panigrahi et al. [14] with RBFN. (b) Path obtained by proposed WNN algorithm.

Table 2: Comparative results of Fig. 6(a), 6(b), 6(c) and 6(d)

| Name of the algorithm | Number of hidden nodes | Simulation time [min] |
|---|---|---|
| Morlet WNN Fig. 6(a) | 10 | 2.06 |
| Mexican hat WNN Fig. 6(b) | 10 | 2.07 |
| Gaussian WNN Fig. 6(c) | 10 | 2.09 |
| RBFN Fig. 6(d) | 10 | 6.20 |

Table 3: Comparative result between RBFN [14] and WNN

| Scenario | Number of hidden nodes | Simulation time [min] |
|---|---|---|
| RBFN Fig. 7(a) | 10 | 1.8 |
| WNN Fig. 7(b) | 10 | 1.3 |

path length and time taken to reach the target. Fig. 5(a) -5(c) and Fig. 6(a)-6(c) show the simulation results of mobile robot using WNN with Mexican hat, Morlet and Gaussian wavelet as activation function respectively. When obstacles are nearer to the robot the training of the neural network is activated which adjusts robots motion direction in terms of steering angle towards the target. The obstacle avoidance behavior is activated when obstacle distance from the robot is less than the minimum threshold value which is pre-defined in the algorithm. The working environment of Fig. 5(a)-5(c) contains vertical shape of obstacle and Fig. 6(a)-6(c) is based on stair case type obstacle, the robot is allowed to move from one start position to a target. In Fig. 5 the robot start position is (250, 50) and the target is set at (50, 250). The simulation results shows that the trajectory obtained in Fig. 5(b) is more smother than Fig. 5(a) and Fig. 5(c). Similarly Fig. 6(a)-6(c) correspond to staircase type obstacles where robot is allowed to move from start position (100, 50) to target (50, 250). Fig. 5 and Fig. 6 show the simulation results of navigation of mobile robot containing different types of obstacle environment. Aim of the all simulations allows the robot to reach the target by avoiding obstacles. It can be observed from different simulation results that if there is no obstacle or obstacles are away from the robot, then it navigates in a straight path otherwise makes curved path, i.e. either a right or a left turn. In some scenarios like Fig. 5(a), 5(c) and Fig. 6(a), 6(c) the robot has made greater steering angle as a result the length of the path becomes more than the in Fig. 5(b) and Fig. 6(b). Further, the Fig. 5(d) and Fig. 6(d) represent simulation results of mobile robot with RBFN algorithm. The Fig. 7(a) and Fig. 7(b) show the comparison of simulation result of proposed WNN algorithm with RBFN [14]. It is obvious that the speed of WNN algorithm is faster than RBFN with same environment. The results of WNN algorithm demonstrate good performance in different clustered environments as compared to RBFN algorithm but there are some differences in smoothness of trajectories.

## 8. Conclusion

The solution of motion planning in unknown static environment is presented in this paper using simulation mode. The problem may be divided into two categories, i.e. generation of collision free optimum path and obstacle avoidance in different clustered environments. A set of heuristic training patterns was considered for mobile robots to avoid collision with obstacles by changing steering angles of the robot. The accuracy of obstacle avoidance depends on training of the network for different activation functions of WNN in clustered environments. WNN has strong learning ability and adaptability. It requires smaller training patterns and fewer nodes than MLP and RBFN for same performance. It has faster training speed, convergence and good robustness characteristics. The approach can be applied even in multi-robot environments with some modification of control law.

## References

[1] R. ABHIYEV, D. IBRAHIM and B. ERIN: Navigation of mobile robots in the presence of obstacles. *Elsevier Advances in Engineering Software*, **41**(10-11), (2010), 1179-1186.

[2] S.M. RAGHURAMAN, D. TAMILSELVIAND and N. SHIVAKUMAR: Mobile robot navigation using fuzzy logic controller. *IEEE Conf. on Control Automation Communication and Energy Conversion*, Madurai, India, (2009), 1-5.

[3] S.K. HARISHA, K.P. RAMAKANTHA, M. KRISHNA and S.C. SHARMA: Fuzzy logic reasoning to control mobile robot on a pre-defined strip-path. *World Academy of Science, Engineering and Technology*, **42** (2008), 642-646.

[4] M.M. JOSHI and M.K. ZAVERI: Optimally learnt neural network based autonomous mobile robot navigation system. *Int. J. on Control System and Instrumentation*, **2** (2011), 28-32.

[5] D. JANGLOVA: Neural networks in mobile robot motion. *Int. J. of Advanced Robotics Systems*, **1** (2004), 15-22.

[6] D.Z. JIANGAND and W.Z MIN: Mobile robot path tracking in unknown environment. *IEEE Conf. in Robotics, Automation and Mechatronics*, Chengdu, China, (2008), 1201-1205.

[7] A. HADDAD and M. KHATIB: Reactive navigation in outdoor environment using potential fields. *Proc. of IEEE Int. Conf. on Robotics & Automation*, Leuven, Belgium, **2** (1998), 1232-1237.

[8] L. HUANG: A potential field approach for controlling a mobile robot to track a moving target. *IEEE Int. Symp. on Intelligent Control*, Singapore, (2007), 65-79.

[9] H. NOBORIO, S. WAZUMI, S. FUKUDA and S. ARIMOTO: A potential approach for a mobile robot on an implicit potential field without generation of local minima. *IEEE Int. Conf. on Intelligent Robots and Systems*, Taukube, Japan, (1989), 70-77.

[10] A. BAKER: A novel mobile robot navigation system using neuro-fuzzy rule-based optimization technique. *J. of Applied Sciences and Technology*, **4** (2012), 2577-2583.

[11] Y. GIGRAS and K. GUPTA: Artificial intelligence in robot path planning. *Int. J. of Soft Computing and Engineering*, **2**(2), (2012), 1203-1209.

[12] V. CASTRO, J.P. NEIRA, C.L. RUEDA and J.C. VILLAMIZAR: Autonomous navigation strategies for mobile robots using a probabilistic neural network. *IEEE Conf. of Electronics Society*, Taipei, Taiwan, (2007), 2795-2800.

[13] N. BUNIYAMIN, W.A.J. WANNGAH, N. SARIFF and Z. MOHAMAD: A simple local path planning algorithm for autonomous mobile robots. *Int. J. of Systems Applications, Engineering and Development*, **5**(2), (2011), 107-120.

[14] P.K. PANIGRAHI, S. GHOSH and D.R. PARHI: A novel intelligent mobile robot navigation technique for avoiding obstacles using RBF neural Nnetwork. *IEEE Int. Conf. on Control, Instrumentation, Energy and Communication*, Calcutta, India, (2014), 51-56.