# Fuzzy-Neuro based Navigational Strategy for Mobile Robot

Shubhasri Kundu, Dayal R. Parhi, B.B.V.L Deepak

**Abstract**— A new paradigm of intelligent navigation system for mobile robot has been enriched with some common features like: criteria for optimal performance and ways to optimize design, structure and control of robot. With the growing need for the deployment of intelligent and highly autonomous systems, it would be beneficial to flawlessly combine robust learning capabilities of artificial neural networks with a high level of knowledge interpretability provided by fuzzy-logic. Fuzzy-neural network is able to build comprehensive knowledge bases considering sensor-rich system with real time constraints by adaptive learning, rule extraction and insertion, and neural/fuzzy reasoning. This technique is simulated and also compared with other simulation studies by previous researcher. The training for back propagation algorithm and its navigational performances analysis has been done in real experimental setup. As experimental result matches well with the simulation result, the realism of method is verified.

**Index Terms**— Fuzzy, Mobile Robot, Navigation, Neural, Optimized path, Robotic behavior, Training pattern.

## 1 Introduction

NAVIGATION of mobile robot, which can be defined as the strings of schedules required during goal achieving without any collision with static as well as dynamic obstacle, necessitates the abilities of a mobile robot to plan and execute optimized paths within its environment; it may be vague, huge and either partially or absolutely dynamic. Development of new concepts and strategies to tolerate a wide range of uncertainty [5] in the area of mobile robot navigation has attracted attentions of many researchers.

Fuzzy Logic and Neural networks both have properties for controlling inherent uncertainties and inaccuracies in the sensor data and planning of a strategic action selection mechanism.

- Fuzzy systems are able to treat uncertain and imprecise information, they make, use of knowledge in the form of easily understandable linguistic rules. In a fuzzy inference system, fuzzy logical rules can model the qualitative aspects of human knowledge and reasoning processes without employing precise quantitative analysis [6]. Their drawbacks are caused mainly by the difficulty of designing accurate membership functions and lack of a systematic procedure for the transformation of expert knowledge into the rule base.
- On the other hand, neural networks have strong learning abilities though they are weak for expressing rule-based knowledge [10]. Artificial neural network (ANN) systems offer advantages of acquiring knowledge through learning, [6] adaptation, fault-tolerance, parallelism, and generalization. Neural networks have the ability to learn but with some neural networks, knowledge representation and extraction are difficult [3].

So the idea is to combine neural networks and fuzzy systems to overcome their disadvantages, but to retain their advantages combining the versatility of neural networks and fuzzy logic replicating aspects of human thought [6]. As the former property reduces the time required to create the model, the latter increases the usefulness of the model [1], Fuzzy neural network (FNN) provides computational intelligence that come with significant learning abilities and inherent transparency (interpretability) to provide strong mechanisms for building intelligent systems that must operate in rapidly changing environments. So, it is able to learn and approximate real-world concepts, building a knowledge base that may be interpreted and modified by the user [2].

This article delivers a narrative attitude for design of a perceptive controller for autonomous mobile robot using multilayer feed forward neural network next to FLC, as FLC is not completely perfect to deal with the increment of variables in robotic environment. Fuzzy logic has already been used for behavior design such as obstacle avoidance, wall following and target seeking. To solve the problem of insufficient knowledge, rule-based controller is trained by a back-propagation learning algorithm that allows autonomous robot to gain more accurate steering angle than sensory information in a motive to minimize the error and to maintain a time-optimal and collision-free path in unknown and cluttered environments. Simulation and experimental results are presented to demonstrate the validity of the approach.

The framework of this paper is as follows, succeeding the introduction, the fuzzy-neural architecture for navigation of mobile robot is depicted in section 2. The simulation results are discussed and to ascertain viability of the developed technique comparisons have been made with other methods in section 3. In section 4, experimental results are certified with simulation to reveal the supremacy of the recommended methodology. Finally, summary has been briefed in the last section 5.

- *Shubhasri Kundu is currently pursuing Ph.D program in robotics in NIT Rourkela,Odisha, India, PH-9800257472. E-mail: shubhasri.ee2006@gmail.com*
- *Dayal R.Parhi is a Professor in mechanical engineering in NIT Rourkela, Odisha, India, PH-06612462514. E-mail:dayalparhi@yahoo.com*
- *B.B.V.L Deepak is an Asst. Professor in industrial design in NIT Rourkela, Odisha, India, PH-06612462581. E-mail:deepak.bbvl@gmail.com*

## 2 Analysis of Fuzzy-Neuro Architecture for Navigation

To reduce travel time as well as the distance travelled, four layer perceptron neural network has been designed by using outcomes from the FLC as well as environmental information to make navigational decisions. The first layer is used as input layer which has six neurons; four for receiving the values of the distances from obstacles in front, left and right of the robot and also for the target bearing angle and other Twos are for Left and Right wheel Velocities from FLC. The inputs to the proposed fuzzy control scheme consist of the distances between a robot and the obstacles to the left, front and right locations and heading angle of robot to the target, acquired by sensors. In this research three types of membership functions (Trapezoidal, Triangular and Gaussian) are hybridized in a single controller for each input and output variables.

### 2.1 *Fuzzy Rule Base Mechanism*

Fuzzy rules are formulated based on human perception. The fuzzy rule base is a set of linguistic rules in the form of 'if a set of conditions are satisfied, then a set of consequences are inferred'. Based on the above fuzzy subsets, the fuzzy control rules are defined in a general form for four inputs and two outputs fuzzy system as follows [8]:

If (matching degree of LD is $\mu(LD_i)$ and matching degree of FD is $\mu(FD_j)$ and matching degree of RD is $\mu(RD_k)$ and matching degree of HA is $\mu(HA_m)$, Then (matching degree of LV is $\mu(LV_{ijkm})$ and matching degree of RV is $\mu(RV_{ijkm})$. (1)

where i = 1 to 5, j = 1 to 5, k = 1 to 5 and m =1 to 5 because LD, FD, RD and HA have five membership functions each.

The matching degree of final output is computed by the following formula:

Matching degree $\mu_{LV,\,RV}(vel_{ijkm}) = \min\{\mu(LD_i), \mu(FD_j), \mu(RD_k)$ and $\mu(HA_m)\}$ (2)

When the matching degree=1 the inferred conclusion is identical to the rule's consequent, and if it is zero no conclusion can be inferred from the rule.

Finally, the output firing area of the left and right wheel velocities can be computed by the following formula,

$\mu_{LV}(vel) = \max\{\mu_{LV}(vel_{1111}), \ldots\mu(vel_{ijkm}), \ldots\mu(vel_{5555})\}$

$\mu_{RV}(vel) = \max\{\mu_{RV}(vel_{1111}), \ldots\mu(vel_{ijkm}), \ldots\mu(vel_{5555})\}$ (3)

The final output (crisp value) of the fuzzy logic controller of left and right wheel velocities can be calculated by "Centre of Gravity" method [9],

$$\left.\begin{aligned} LeftVelocity = LV &= \frac{\int vel \cdot \mu_{LV}(vel) \cdot d(vel)}{\int \mu_{LV}(vel) \cdot d(vel)} \\ RightVelocity = RV &= \frac{\int vel \cdot \mu_{RV}(vel) \cdot d(vel)}{\int \mu_{RV}(vel) \cdot d(vel)} \end{aligned}\right\} \quad (4)$$

### 2.2 Neural Architecture for Navigation

The network consists of two hidden layers (shown in Fig. 1) which adjusted the weight of neuron; as with one hidden layer it is difficult to train the network within a specified error limit. The training error is the difference between desired output and actual output. The first hidden layer has eighteen neurons and the second hidden layer has five neurons. These numbers of hidden layers were also found empirically. Then an output layer with a single neuron which provide steering angle to control the direction of movement of the robot. Back propagation method is used to minimize the error and optimize the path and time of mobile robot to reach the target [4].

During training and during normal operation, the input patterns fed to the neural network comprise the following components:

$Y_1^{\{1\}}$ = Left obstacle distance from the robot

$Y_2^{\{1\}}$ = Right obstacle distance from the robot

$Y_3^{\{1\}}$ = Front obstacle distance from the robot

$Y_4^{\{1\}}$ = Target bearing angle

$Y_5^{\{1\}}$ = Left Wheel Velocity

$Y_6^{\{1\}}$ = Right Wheel Velocity

These input values are distributed to the hidden neurons which generate outputs given by

$$\mathbf{Y}_j^{\{lay\}} = \mathbf{f}(\mathbf{V}_j^{\{lay\}}) \quad (5)$$

$$\mathbf{V}_j^{\{lay\}} = \sum_i W_{ji}^{\{lay\}} \mathbf{Y}_i^{\{lay-1\}} \quad (6)$$

Where,

lay: layer number (2 or 3)

*j*: label for jth neuron in hidden layer 'lay',

*i*: label for ith neuron in hidden layer 'lay-1'

$W_{ji}^{\{lay\}}$: Weight of the connection from neuron i in layer 'lay-1' to neuron j in layer 'lay'.

f (.) : Activation function, chosen in this work as the continuous log-sigmoid function:

$$f(x) = \frac{1}{1+e^{-\beta x}} \quad (7)$$

Where, $\beta$ is a slope parameter.

The sigmoid has the property of being similar to the step function, but with the addition of a region of uncertainty. Sigmoid functions in this respect are very similar to input-output relationships of biological neurons, although not exactly the same.

During training, the network output $\theta_{actual}$ may differ the desired output $\theta_{desired}$ as specified in the training pattern presented to the network. A measure of the performance of the network is the instantaneous sum-squared difference between $\theta_{desired}$ and $\theta_{actual}$ for the set of presented training patterns:

$$\mathbf{Err} = \sum_{all\ traning\ pattrns} (\theta_{desired} - \theta_{actual})^2 \quad (8)$$

The error back propagation method is employed to train the network. This method requires the computation of local error gradients in order to determine appropriate weight corrections to reduce Err. For the output layer, the error gradient is

$$\delta^{\{4\}} = \mathbf{f}'(\mathbf{V}_1^{\{4\}})(\theta_{desired} - \theta_{actual})^2 \quad (9)$$

The local gradient for neurons in hidden layer {lay} is given by:

$$\delta_j^{\{lay\}} = \mathrm{f}'(\mathrm{V}_1^{\{lay\}})(\sum_k \delta_k^{\{lay+1\}} W_{kj}^{\{lay+1\}}) \quad (10)$$

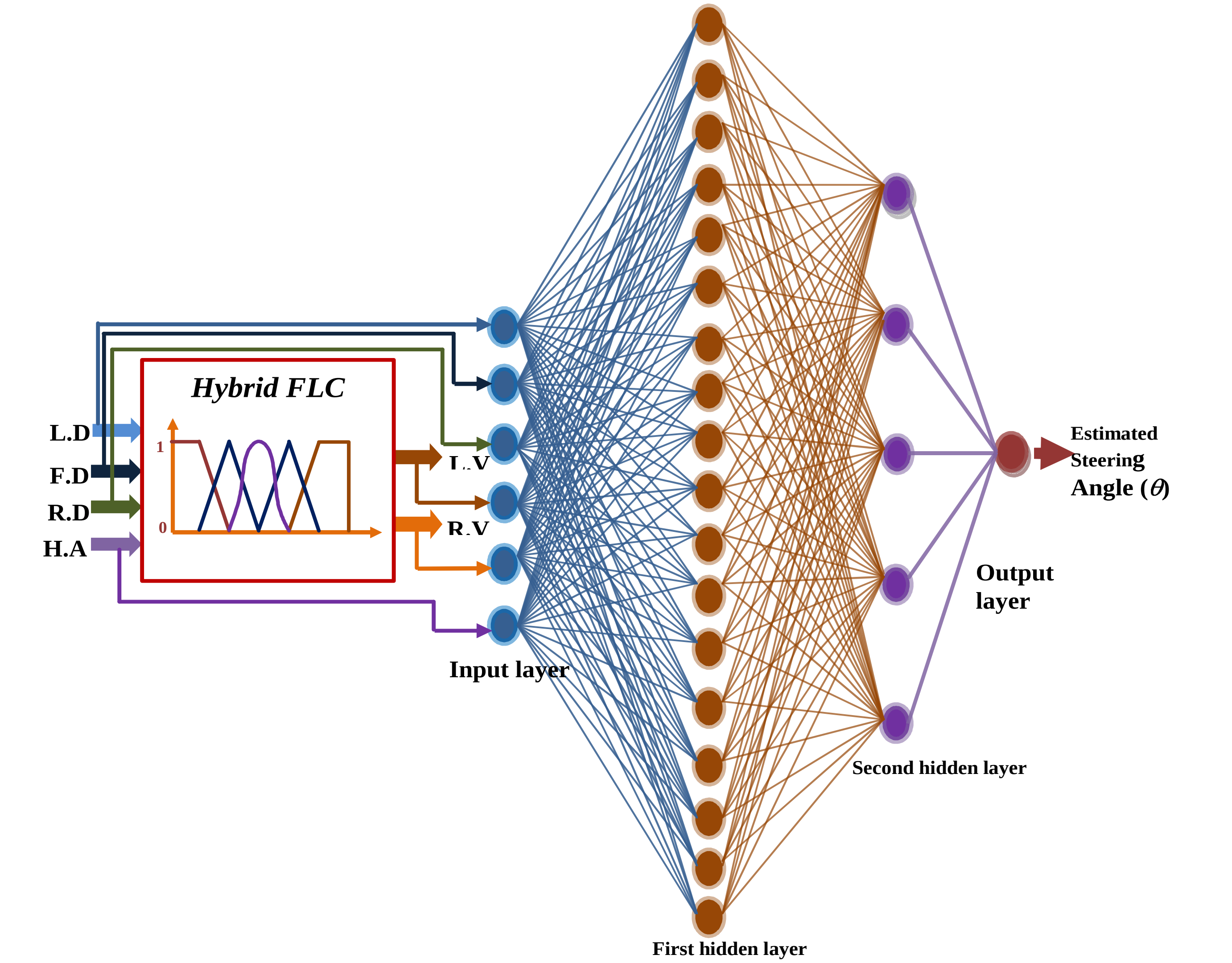


Fig. 1. Hybrid Fuzzy & Multilayer Neural Controller for Coordination of Robotic Behaviors.

The synaptic weights are updated according to the following expressions:

$$W_{ji}(t+1) = W_{ji}(t) + \Delta W_{ji}(t+1) \quad (11)$$

And $W_{ji}(t+1) = \alpha\Delta W_{ji}(t) + \eta\delta_j^{\{lay\}} y_i^{\{lay-1\}}$ (12)

Where, $a$ = momentum coefficient (chosen empirically as 0.2 in this work)

$\eta$ = learning rate (chosen empirically as 0.35 in this work)

$t$ = iteration number, each iteration consisting of the presentation of a training pattern and correction of the weights.

The final output from the neural network is:

$$\theta_{actual} = f(V_1^{\{4\}}) \quad (14)$$

Where, $$V_1^{\{4\}} = \sum_i W_{1i}^{\{4\}} y_i^{\{3\}} \quad (15)$$

It should be noted learning can take place continuously even during normal target seeking behavior. This enables the controller to adopt the changes in the robot's path while moving towards target. Mainly three behaviors (obstacle avoidance, wall following and target seeking) are required to train and to design an intelligent controller for mobile robot being used to navigate in a cluttered environment.

The numbers of neurons are found based on the number of training patterns and the convergence of error during training to a minimum threshold error to control the direction of movement of the robot. The neural network is empirically trained to navigate by 200 training patterns representing typical scenarios, some of which are depicted in Table 1. For example, in training pattern no. (vii) a robot is surrounded by left, front and right obstacles at distances of 21c.m, 15c.m and 19c.m respectively. Ultrasonic sensor is giving the reading of $43^0$ for the current position of robot; i.e. target is located at an angle of $43^0$ at the right side of robot. Output of FLC for this situation as left and right wheel velocities are 5.8c.m/s and 5.1c.m/s respectively based on sensor data. In this scenario, neural network is trained to steer robot towards its right with

TABLE 1
SOME OF THE TRAINING PATTERN OF FUZZY-NEURAL CONTROLLER

| Sl. No. of Training Patterns | Inputs to the neural network | | | | | | Output |
|---|---|---|---|---|---|---|---|
| | Left obstacle distance(cm) | Front obstacle distance (cm) | Right obstacle distance(cm) | Target angle (degree) | Left Wheel Velocity from FLC (cm/s) (in approximate) | Right Wheel Velocity from FLC (cm/s) (in approximate) | Steering angle (degree) |
| (i) | 22 | 20 | 15 | 0 | 4.5 | 6.8 | -23 |
| (ii) | 11 | 17 | 15 | 0 | 5.2 | 4.3 | 19 |
| (iii) | 15 | 21 | 9 | 0 | 3.8 | 4.9 | -14 |
| (iv) | 8 | 13 | 15 | 27 | 6.7 | 3.9 | 23 |
| (v) | 20 | 10 | 12 | -39 | 3.2 | 7.1 | -27 |
| (vi) | 13 | 20 | 10 | 0 | 4.3 | 4.7 | -9 |
| (vii) | 21 | 15 | 19 | 43 | 5.8 | 5.1 | 26 |
| (viii) | 16 | 30 | 25 | 36 | 6.5 | 4.2 | 29 |
| (ix) | 14 | 33 | 19 | 24 | 5.23 | 4.65 | 17 |
| (x) | 27 | 35 | 16 | -54 | 4.29 | 6.7 | -41 |
| (xi) | 17 | 29 | 14 | 0 | 4.7 | 4.9 | -5 |
| (xii) | 23 | 14 | 27 | 0 | 5.9 | 4.1 | 13 |
| (xiii) | 30 | 10 | 15 | -45 | 3.4 | 6.9 | -37 |
| (xiv) | 12 | 17 | 14 | 8 | 5.1 | 3.78 | 6 |
| (xv) | 10 | 10 | 15 | 17 | 5.74 | 3.69 | 11 |
| (xvi) | 19 | 23 | 15 | 0 | 4.67 | 4.93 | -13 |
| (xvii) | 38 | 10 | 33 | -63 | 4.15 | 7.1 | -51 |
| (xviii) | 21 | 7 | 16 | -49 | 3.97 | 6.75 | -37 |
| (xix) | 35 | 18 | 20 | -37 | 4.61 | 6.98 | -29 |
| (xx) | 31 | 17 | 15 | -47 | 3.54 | 6.87 | -41 |
| (xxi) | 24 | 32 | 13 | -56 | 4.35 | 6.79 | -43 |
| (xxii) | 18 | 25 | 9 | 0 | 4.25 | 4.67 | -13 |

an angle of $26^0$ with respect to goal position to maintain the shorter trajectory.

## 3 SIMULATION RESULTS

The series of simulations test have been conducted to exhibit that the anticipated method can partially fulfill the most of the fundamental as well as critical robotic behaviors during navigation in complex and uncertain environments. All simulation environments are generated artificially containing one movable robot and static obstacles as well as only one static target.

To justify the worth and robustness of Fuzzy-Neuro control algorithm, simulation result (Fig. 2(b))on mobile robot navigation has been compared with previous research work (Fig. 2 (a)) where a new FNN is applied to a simulation mobile robot of three DOF by Ma et al. [7]. Comparison is illustrating degree of restricted optimization ability of the controller which resolves that fuzzy-neuro controller can find the definite goal using minimum path length than path traced by Ma et al. [7] in Fig. 2. So the comparison of performances has shown a good agreement and also verifies the ability and flexibility of the present approach.

It has been perceived that the robot controlled through fuzzy-neural control has improved performance than the fuzzy controller in terms of positioning accuracy and collision avoidance and it provides optimize path to reach the goal.

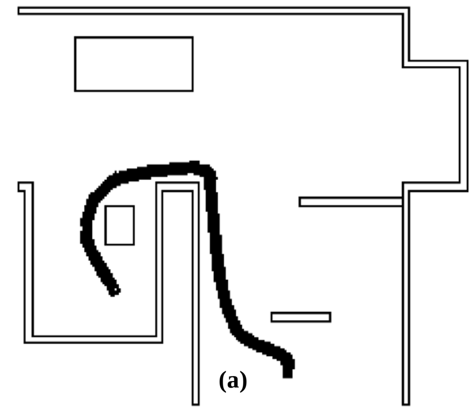


Fig. 2(a). Mobile robot reference trajectories by Ma et al. [7]

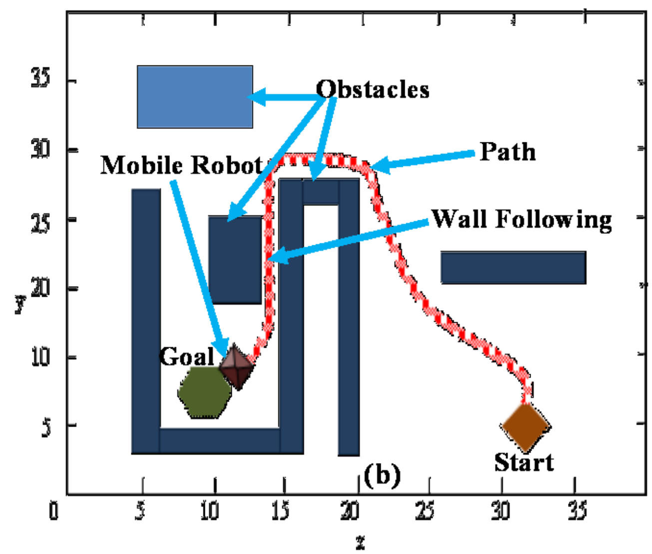


Fig. 2(b). Simulation result by developed fuzzy neural controller

## 4 EXPERIMENTAL RESULT

The experimental work has been made by loading the Fuzzy-Neuro algorithm into the developed mobile robot through software and hardware interface. After learning and training, Fuzzy-Neuro algorithm is able to provide Left wheel and Right wheel velocities and optimized steering angle which are sufficient to avoid obstacles and achieve target in real world environment. The assumptions about the mechanical structure and motion of a mobile robot, to which the proposed method is applied, are as follows:

(1) Mobile robot moves on a plane surface.

(2) The wheel of a mobile robot rolls on the floor without translational slip.

(3) The wheel of a mobile robot makes rotational slip at the contact point between each wheel and the floor.

During experiment, paths traced by the robot are marked on the floor by a pen (fixed to the front of the robots) as they move in Fig. 3. The experimentally acquired paths closely follow the paths sketched by the robots during simulation for analogous arrangement of obstacles, start and goal point. It has been acquired that the experimental path lengths and time taken are more than the simulation path lengths and time taken. This is due to presence of various errors (e.g. signal transmission error in data-cable, obstacle or target tracking error, presence of friction in rotating elements, slippage between floor and wheels, friction between supported point and floor etc.).

The path lengths are taken in average from 9 different experiments which have been performed in environmental scenario as shown in Fig. 3. Elementary as well as significant robotic behaviors have been addressed in both simulation and experimental modes employing fuzzy-neural approach.

It has been found that the results obtained from experimental setup are more close to results obtained from simulation mode (shown in Fig. 2(b)) which validate the proposed method. FLC along with Neural network affords much more rapid response in an unidentified environment and has less computational effort than other conventional approaches.

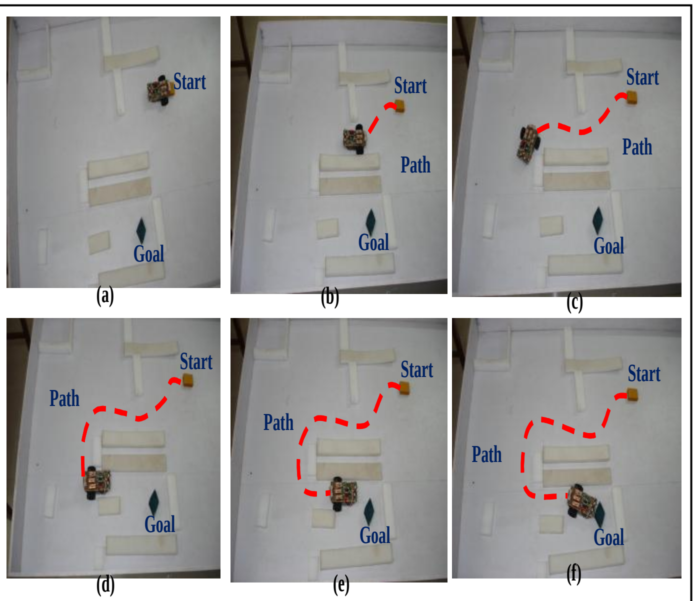


Fig. 3. Experimental Results Using Fuzzy-Neuro Controller in comparison with simulation result (Fig. 2(a)) by Ma et al. [7]

## 6 CONCLUSIONS

On the basis of hypothetical, simulation and experimental investigations, some precious features of Fuzzy-Neuro algorithm in mobile robot navigation can be briefed here:

TABLE 2
UNITS FOR MAGENTIC PROPERTIES

| Path Length in simulation by proposed algorithm (in pixel) | Path Length in Experimental mode by proposed algorithm (in pixel) | Path Length in simulation of Previous Research work by Ma et al.[7] (in pixel) | % of error between simulation and experimental results by proposed technique |
|---|---|---|---|
| 193 | 221 | 197 | 12.669% |

1. It has been seen that, by using the Hybrid Fuzzy-Neuro controller the robots are able to avoid any obstacles, escape from dead ends, and find target in complex hazardous environments.
2. Various navigational control strategies (e.g. obstacle avoidance, wall and edge following and target seeking) have been addressed in simulation and experimental environments using developed controller.
3. Training patterns of Back-Propagation Algorithm based neural network can be generated by simulation rather than by experiment, saving considerable time and effort.
4. Comparison of developed algorithm (Table 2) with simulation result by Ma et al. [7] employing FNN technique (Fig. 2(a)) both in simulation (Fig. 2(b)) and experimental (Fig. 3) environment delivers a good performance measure concerning veracity of the method.

This hybrid approach has been trialed for achieving a resolution of reducing inaccuracy in steering angle and optimization with respect to path length and time in both simulation and experimental mode. This issue has partially been solved here.

## REFERENCES

[1] Amidis I.S. and Roberts G. N., "Fuzzy Modelling & Fuzzy-Neuro Motion Control of an Autonomous Underwater Robot", In Proc. of IEEE International Conference on AMC '98 – COIMBRA, pp. 641-646.

[2] Gobi Adam F. and Pedrycz Witold, "The potential of fuzzy neural networks in the realization of approximate reasoning engines", Fuzzy Sets and Systems 157 (2006) 2954 – 2973.

[3] Godjevac J. and Steele N., "Neuro-fuzzy control of a mobile robot", Neurocomputing, 28, 1999, 127-143.

[4] Haykin S., "Neural Networks a Comprehensive Foundation", Second Ed., (India: Pearson prentice hall), 2006.

[5] Jin Yaochu and Jiang Jinping, "Techniques in Neural-network-based Fuzzy System Identification and Their Application to control of complex systems", Fuzzy Theory Systems Techniques and Applications, 1 (1999) 112-128.

[6] Jolly K.G., Kumar R. Sreerama and Vijayakumar R., "Intelligent task

planning and action selection of a mobile robot in a multi-agent system through a fuzzy neural network approach", Engineering Applications of Artificial Intelligence 23 (2010) 923–933.

[7] Ma Xiaowei, Li Xiaoli and Qiao Hong, "Fuzzy neural network –based real-time self-reaction of mobile robot in unknown environments", Mechatronics, 2001, Vol.11, pp.1039-1052.

[8] Parhi Dayal R., "Navigation of Mobile Robots Using a Fuzzy Logic Controller", Journal of Intelligent and Robotic Systems, Volume 42, Number 3 / March, 2005.

[9] Parhi D R and Singh M K, "Intelligent fuzzy interface technique for the control of an autonomous mobile robot", Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science, Volume 222, Number 11 / 2008.

[10] Matijevics I., "Infrared Sensors Microcontroller Interface System for Mobile Robots", 5th International Symposium on Intelligent Systems and Informatics, Subotica, Serbia, 24-25 August, 2007.