British Journal of Mathematics & Computer Science
3(4): ……., 2013

SCIENCEDOMAIN *international*
www.sciencedomain.org

A Hybrid System Geno-Fuzzified Neural Network for Mobile Robot Control

Hussein Attya Lafta[1*] and Zainab Falah Hassan[2*]

[1]*Department of Computer Science, College Science for Women, Babylon University, Iraq.*

Research Article

Received: 23 April 2013
Accepted: 03 August 2013
Published: 15 August 2013

Abstract

Aims: The goal of mobile robot is build system able to achieve tasks without human intervention in cluttered unknown environments. A main issue of an autonomous mobile robot is the design of an intelligent controller which enables the robot to navigate in a real world environment and avoiding obstacles especially in crowded and changing environment.
Study Design: The controller uses genetic, fuzzy and neural to control of mobile robot.
Place and Duration of Study: College Science, computer department, between September 2011 and December 2012.
Methodology: In this search, fuzzy logic, genetic algorithm, and neural network (soft computing) are used to design an intelligent controller. This is due to the fact that fuzzy if-then rules are well suited for capturing the imprecise nature of human knowledge and reasoning processes. On the other hand, the neural networks are equipped for learning. Genetic algorithm has active role in the generating of fuzzy rules, it is designed to minimize the number of rules to minimum number. It is also helped to improve membership functions. Neural network is trained by using back propagation to increase efficiency of the work in time of arrive and get the shortest path to goal, it is obtained the steer angle of robot to the appropriate direction (avoid obstacles or get target).
Results: The efficiency and robust of this work is appeared by using many different unknown environments that have different numbers, sizes and shapes of obstacles. The controller enables robot to avoid obstacles and reach goal with shortest distance (757 pixels) compared with other techniques(fuzzy controller and neuro-fuzzy controller),which owns the largest distance from same start position to the same end position and also less time(14 seconds).
Conclusion: Geno – fuzzified – neural controller is efficient with different numbers, shapes, sizes of obstacles in unknown environments.

Keywords: Mobile robot, fuzzy logic, obstacle avoidance, genetic algorithm, neural net.

**Corresponding author: hzazmk@yahoo.com;*

1 Introduction

Target tracking is a crucial function for an autonomous mobile robot navigating in unknown environments. The controller calculates both the mobile robot linear and angular velocities from the distance and angle that separate it to the moving target. The controller was designed using fuzzy logics theory and then, a genetic algorithm was applied to optimize the scaling factors of the fuzzy logic controller for better accuracy and smoothness of the robot trajectory. Autonomous mobile robots are intelligent agents which can perform desired tasks in various (known and unknown) environments without continuous human guidance [1].

The path planning and obstacle avoidance are the most important tasks for an autonomous mobile robot moving in an unknown environment. Fuzzy logic controllers are constructed for target searching behavior and obstacle avoidance behavior based on the distance and angle between the robot and the target as inputs for the first behavior and the distance between the robot and the nearest obstacle for the second behavior; then a third fusion behavior is developed to combine the outputs of the two behaviors to compute the speed of the mobile robot in order to fulfill its task properly [2].

A twolink planar robot manipulator using a two dimensional conguration Space representation Parallelism within the method is identified and reductions in execution time are achieved. The basic concept of the Tangent graph is extended to develop a new path planner. The second new planner is developed based around a Genetic Algorithm. Thisprovides a probabilistic approach to the search for a path [3].

An approach based on the Artificial Potential Field (APF) method in which the target creates a virtual force that attracts the robot while obstacles create a virtual force that repels the robot. APF-based methods are very interesting because they are simple to implement; however, they have some inherent limitations, that the trajectory can have undesirable oscillations [4].

The adaptation of the system involves the tuning of the control rules thereby trimming the control actions, and adjusting the fuzzy controller output gain [5].

A fuzzy logic system and propose an obstacle avoidance algorithm for a path planning in unknown environment for a mobile robot. The ultrasonic sensors are employed for detecting the distance to obstacles and their positions. An angular velocity control for left and right wheels is implemented by a fuzzy logic system [6].

Path planning is a term used in robotics for the process of detailing a task into discrete motions. It is aimed at enabling robots with capabilities of automatically deciding and executing a sequence motion in order to achieve a task without collision with other objects in a given environment. Genetic algorithms are considered as a search process used in computing to find exact or an approximate solution for optimization and search problems. There are also termed as global search heuristics. These techniques are inspired by evolutionary biology such as inheritance mutation, selection and cross over [7].

An Artificial Neural Network (ANN) which is used to a path by its nonlinear functional approximation. The training samples of artificial neural network have been collected by a reinforcement learning method known as Q learning. This path planning algorithm has been devised using five state action mapping relationship [8].

Fuzzy logic has been introduced to deal with uncertain problems. A fuzzy logic controller can be regarded as an expert system that is able to process quality variable and to infer crisp values out of uncertainty. A mobile robot, having to navigate purposefully from a start location to a target location, needs two basic requirements: sensing and reasoning. However, the pervasive presences of uncertainty in sensing makes the choice of a suitable tool of reasoning and decision making, that can be deal with incomplete information, vital to ensure a robust control system [9].

Path planning for a mobile robot in dynamic environments proposed by [10]. They introduce a method of global path planning for a robot moving in an environment cluttered with obstacles which have arbitrary shape, size and location.

A Modeling Method for Intelligent Service Mobile Robot was presented by [11].In order to solve the current difficulties of modeling for designing Intelligent Service Mobile Robot (ISMR), a modeling method based on met synthesis was proposed from the macro and micro levels.

Type-2 Fuzzy Logic System (T2FLS) was developed to produce control strategy from physical mobile robot that proposed by Siti and Anggina [12].

2 Materials and Methods

2.1 Kinematic Model

The navigation for a two-wheeled mobile robot is controlled by the speed change of the robot. The kinematical scheme of a mobile robot can be depicted as in Fig. 1, where V is the centre velocity of the robot, V_L is the velocity of the left wheel, V_R is the velocity of the right wheel, r is the radius of each wheel, L is the distance between two wheels, x and y are the position of the mobile robot, and θ is the orientation of the robot.

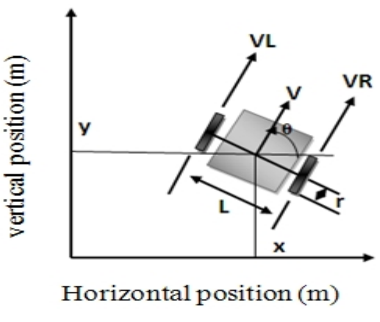

Fig. 1. Kinematical scheme of the mobile robot

According to the motion principle of rigid body kinematics, the motion of a mobile robot can be described using equations (1) and (2), where ω is the centre angular velocity Xianhua J. [13].

$$W = (V_R - V_L)/L \qquad (1)$$

$$V = (V_R + V_L)/2. \qquad (2)$$

The position of the robot in the unknown environment in any moment, is defined in the reference coordinates; [*X*, *Y*]. Considering the equations (1) and (2), the kinematic model of such a robot, is defined with the below equations:

$$X = V \cos \theta \qquad (3)$$

$$Y = V \sin \theta \qquad (4)$$

$$\theta = W \qquad (5)$$

2.2 Geno- Fuzzified -Neural Structure for Mobile Robot Navigation

Intelligent controller for mobile robot enables the robot to avoid the obstacle and improve target seeking ability. In the absence of the obstacle, robot moves towards the endpoint (goal), otherwise the robot will react to the obstacles and its relative position from the goal, based on the information gained from the sensor located in the center robot. While moving towards the goal (with the obstacle avoidance strategy), the robot changes it's angular velocity and linear velocity.

In this work, neural network is used with fuzzy logic and genetic algorithm for mobile robot navigation. The inputs first feed to the neural network, which are

1. Left obstacle distance,
2. Right obstacle distance,
3. Front obstacle distance and
4. Goal angle relative to robot position.

The output of neural network is steering angle1 which represents input to FLC with the same three inputs to the neural network which are left, right and front obstacles distances. A set of fuzzy rules are executed that are resulted from training phase of genetic algorithm.

The outputs from the FLC are left and right wheels velocities. The fuzzy rules advise robot to avoid obstacle and find target. Mobile robot moves online without any pre-defined information about environment. This technique uses back propagation neural network and a fuzzy logic system with genetic algorithm. Geno - Fuzzified - Neural controller with their inputs and outputs is shown in the Fig. 2.

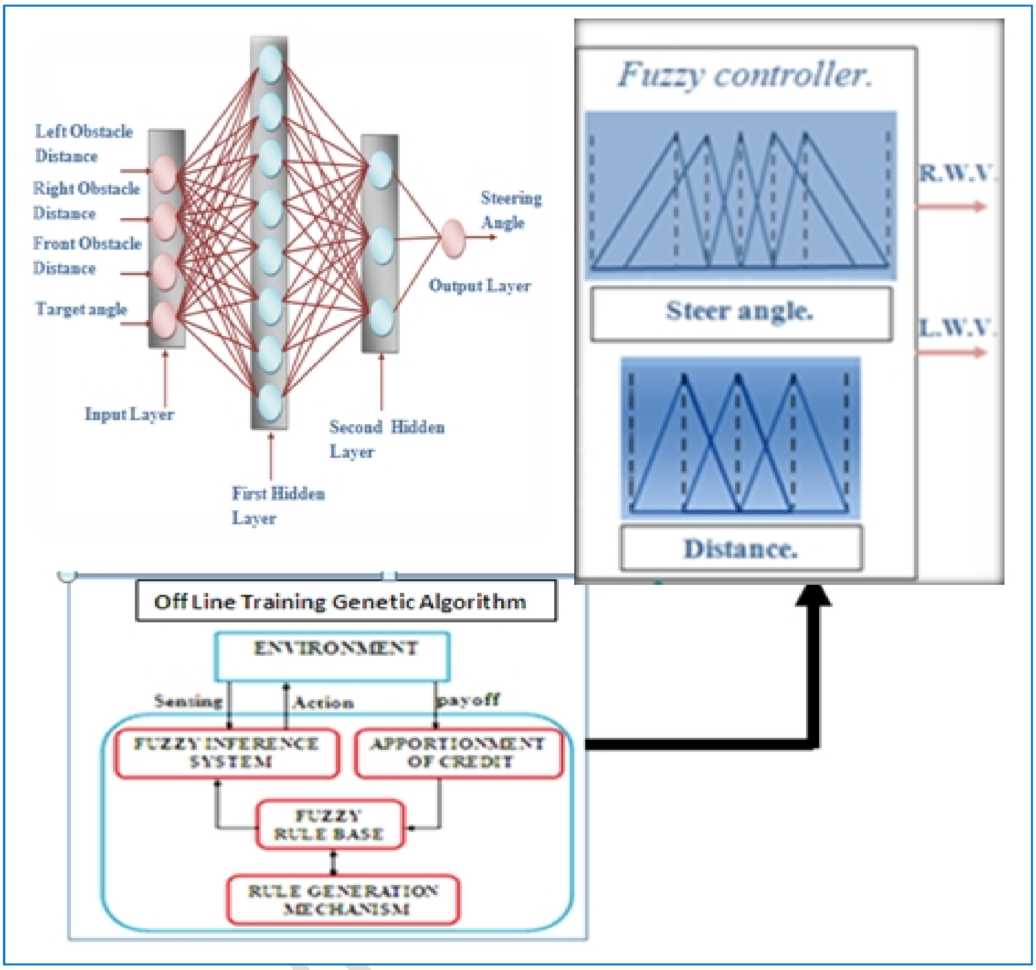

Fig. 2. Block diagram of geno – Fuzzified - Neural controller

Multilayer back propagation neural network is used here with four layers and off line trained. Four neurons for the first layer (input layer) represent obstacles distances on the left, right and front of the robot position which is also inputs for fuzzy controller and the fourth neuron is target angle. Two hidden layers have eight and three neurons consequently. The output layer has a single neuron to get the steering angle1 of robot which represent fourth input to fuzzy controller. The output of the fuzzy controller are left and right wheels velocity of robot.

Steering angle2 results from the velocities of the driving wheels to avoid obstacles and get goal. The start position of robot and the goal position are selected in the environment, then the robot moving from start to goal by controller in series of segments.

2.2.1 Neural network to get steer angle1

Neural network is trained by using back propagation Youssef [14] to get steer angle1 toward the suitable direction. This output is used with the distance of left, right and front obstacles to make decision for robot to control of its movement in the environment.

2.2.2 Input fuzzification

The information about environment including *distances* of left, front and right obstacles and *steer angle* 1 are fuzzified using triangle membership functions. The linguistic variables of distance are classified into three fuzzy sets:

- Near fuzzy set: represents distance of obstacle that is close to robot.
- Medium fuzzy set: represents distance of obstacle that is medium distance to robot.
- Far fuzzy set: represents distance of obstacle that is far to robot.

The linguistic variables of steer angle 1 of robot are classified into five fuzzy sets:

- Front fuzzy set: represents the front side of robot.
- Left1 fuzzy set: represents the left side of robot.
- Left2 fuzzy set: represents the far left side of robot.
- Right1 fuzzy set: represents the right side of robot.
- Right2 fuzzy set: represents the far right of robot.

Outputs:

Five triangle membership functions for velocity are:

- Slow.
- Very slow.
- Medium.
- Fast.
- Very fast.

The distance D can be calculated by applying the distance formula using the two value of dx and dy.

Where dx and dy are the difference in distance between the robot and the destination point in x and y coordinate .the following equations in calculating the input variables are:

Distance $=\sqrt{(dx)^2 + (dy)^2}$.

$\Theta = \tan^{-1}(dy/dx)$.

Here, Θ is the desired angle that the robot should face every instance as it goes to the destination point.

Decision making:

The antecedents of fuzzy rules consist of:

1) Front obstacle distance for robot.
2) Left obstacle distance for robot.
3) Right obstacle distance for robot.
4) Steer angle1.

The consequents of fuzzy rules are the output of fuzzy system includes:

1) Left wheel velocity of robot.
2) Right wheel velocity of robot.

Robot behavior:

- ***Obstacles Avoidance:***

The robot moves toward the goal unless there are obstacles around it. The robot tends with the nearest obstacles because they are dangerous to robot, so the robot moves toward the most safety path. The robot checks its front, right and left direction.

When the robot is closed to an obstacle, the robot changes its speed and heading angle to avoid the obstacle. The rules are employed to avoid all obstacles found in the path of robot and get target, they are represented as aproximated rules. They are resulted from off line training of genetic algorithm.

- ***Steering Action Control (Get Target):***

The task of the robot is get the target efficiently. If the robot senses a target, it will decide whether it can get the target or there is any obstacle in its path. If there is no obstacle on the path to the target, the robot will find its desired path and moving towards it.

2.2.2.1 Fuzzy inference mechanism

Fuzzy inference is the process of formulating the mapping from a given input to an output using fuzzy logic. The mapping then provides a basis from which decisions can be made, or patterns discerned. The process of fuzzy inference involves Membership Functions and If-Then Rules. The used inference fuzzy mechanism is the MAMDANI (Min-Max inferences).

2.2.3 Build rules for decision making

Generation of rules is done by using genetic algorithm. Genetic algorithm generate the rules by using one of the Michigan Approaches Oscar Cordon [15] in offline and the resulted rules are used online in the decision making. So, genetic algorithm has two processes: Train and test.

2.2.3.1 Fuzzy system for on-line learning of approximate fuzzy rules

New rules generated by the GA are placed in a so-called *Limbo,* an intermediary repository where rules are evaluated off-line before being applied to the real process. Only when a rule demonstrated its usefulness in the off-line evaluation, it is accepted and inserted into the on-line FRB. The evaluation algorithm assigns a strength to every classifier with a non-zero degree of activation, Steady state genetic algorithm is used in this work.

Training phase of genetic algorithm consists of the following steps:

1- Generate initial random population of rules with initial strength 0.5.
2- Evaluate each individual of population by get membership degree of every gene in the chromosome and then has strength of rule(individual) to represents fitness of rule.
3- Reproduction of rules.
4- Steady state replacement.
5- Evaluate new individuals.
6- Update of strength of every rule.
7- Repeat steps from 3 to 6 until the accepted rules according to the specific measures have got.

The rules in limbo which are active by applying from the environment to get strength of rule which represents fitness of the rule and update its strength and other inactive rules in limbo has zero fitness.

Reproduction stage:

Binary tournament selection is applied for the proposed population.
Uniform crossover is applied to the parent active rules.

Soft mutation adjusts the fuzzy membership function. Soft mutation modifies one of the characteristic points *(a, b, c)* of the triangle fuzzy set.

Rule parameters:

1. Rule Age *(R.A.):* Overall number of control steps since the rule was placed into the Limbo.
2. Rule Activations *(R.Ac.):* Number of control steps in which the rule would have fired in the on-line controller, i.e. the rule antecedent matches the current input state to a non-zero degree.
3. Equivalent Rule Evaluation *(E.R.E.):* Constant evaluation that the rule should have obtained to reach its present strength value after *R.Ac,* activations. It is computed as the value *Er*, starting from an initial strength *SRA* = 0.5, evolves to the current strength SR_J within *R.Ac,* time steps for T_R = **1.**

Limbo parameters:

1. Limit Age *(L.A.):* Defines the maximum evaluation period. After *L.A.* control steps, the rule is removed from the Limbo, being either promoted to the FRB or discarded.
2. Minimum Rule Activations *(M.R.Ac.):* Minimum number of activations that are required for a rule to be considered for promotion from the Limbo to the FRB.
3. Minimum Equivalent Evaluation *(M.E.E.):* Minimum value of the *E.R.E.* that is required for a rule to be considered for promotion from the Limbo to the FRB.

Entering the FRB:

Considering these parameters, whenever the GA generates new rules that are added to the Limbo, the quality of the current rules in the Limbo is estimated. The worst performing rules are discarded, whereas the best performing rules are promoted to the on-line FLC according to the following performance criteria:

1. Rules whose evaluation period expired *(R.A. > L.A.),* and that were either mostly inactive *(R.Ac. < M.R.Ac.)* or obtained a poor evaluation *(E.R.E. < M.E.E.)* are removed from the Limbo and discarded.
2. Sufficiently tested rules that underwent a minimum number of activations *(R.Ac. > M.R.Ac.)* and obtained a good evaluation *(E.R.E. > M.E.E.)* are promoted to the on-line controller.

2.2.3.2 Defuzzification

There are several ways for performing defuzzification. The strategy selected here is the center of maxima method for its simplicity. The crisp values of Left Wheel Velocity and Right Wheel Velocity are computed using center of maxima method. The left and right wheels velocities are resulting the following:-

1. *Steer angle2* of robot. It is change of head angle of robot to the suitable direction to avoid obstacle or to get goal.

-Angular velocity= (Left wheel velocity-Right wheel velocity).
-Steer angle2=Angular velocity.

2. *Velocity* that the robot moves by in its path, should be in the range of sensing distance selected by the robot.

Velocity= (Left wheel velocity+ Right wheel velocity)/2.

2.2.3.3 Set head angle of robot

In this stage, the robot's head angle is set to the desired direction for moving. Head angle is the angle that robot moves by it in the environment. So, to get the head angle, steer angle should be calculated.

From left and right wheels velocities, steer angle is computed. Then it is added to the angle of robot. Head angle is showed in the Fig. 3.

Head angle= Front angle of robot- Steer Angle2.

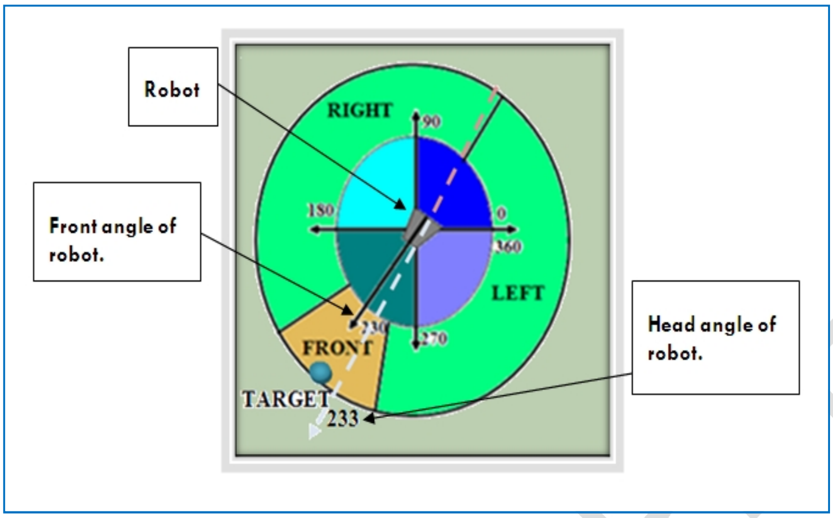

Fig. 3. Head angle of robot

3 Simulation Results and Discussions

The simulations were conducted with the visual basic language. To show the effectiveness and the robustness of the proposed method, simulation results on mobile robot navigation in various environments are explained.

The obstacle avoidance behavior is activated when the reading from any sensors are less than the minimum threshold values. This is how the robot determines if an object is close enough for a collision. When an object is detected too close to the robot, it avoids a collision by moving away from it in the opposite direction.

When the acquired information from the sensors shows that there are no obstacles around robot, its main reactive behaviour is target steer. Intelligent controller mainly adjusts robots motion direction and enable its quickly moves towards the target if there are no obstacles around the robot.

Geno – Fuzzified – Neural controller executes in different unknown environments with different numbers, sizes and shapes of obstacles to explains efficiency of system. In this execution, environment 2 is selected to explains efficiency of system.

Execution screen (visual basic environment) contains the following flexibilities:

1. Choose anyone of the available environments. Add obstacles to the environment.
2. Choose the input's way of the source point and goal point of robot manually or by click mouse on the screen.
3. New execution to restart.
4. Exit of program.
5. Show the length of path (in pixel) of the selected controller from source point to

destination point of robot.

6. Show the time (in second) spent by robot from source point to destination point of the selected controller.
7. Show the source point and destination point of robot.

The results of this controller are explained in Fig. 4 and Fig. 5.

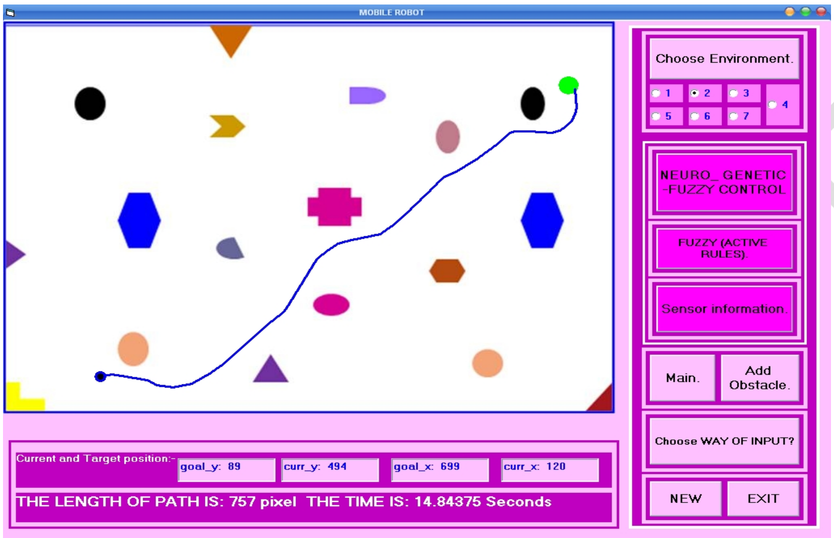

Fig. 4. Path of robot in unknown environment from source to destination point using Neuro - genetic - Fuzzy controller

Note: the green circle acts as goal and blue circle acts as start point of robot.

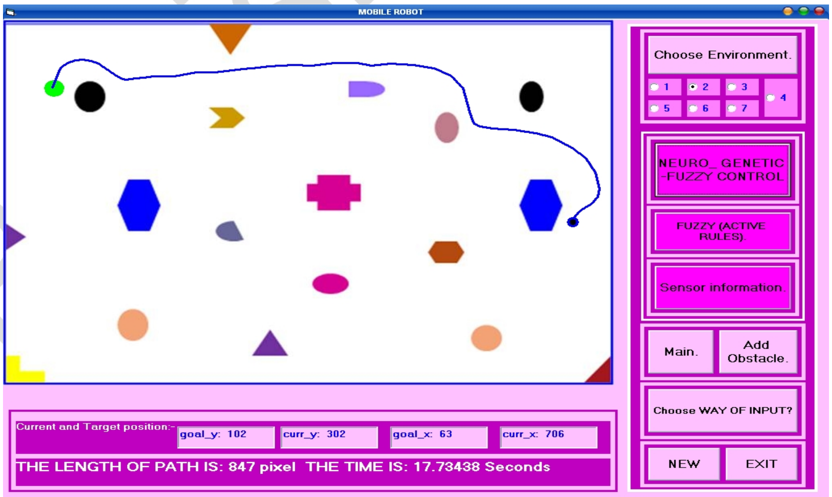

Fig. 5. Path of robot in unknown environment from source to destination point using Neuro - genetic - Fuzzy controller

4 Performance Evaluation of the Proposed System

The performance of the proposed system are compared with different controllers and at different unknown environments. There are two controllers used to prove the efficiency of the proposed system as follows:

1. Fuzzy logic controller used to control of mobile robot.
2. Neuro – Fuzzy controller used to control of mobile robot.

4.1 Fuzzy Logic Mechanism to Control of Mobile Robot

In this controller, the fuzzy logic system is used by mobile robot to evaluates proposed system's performance. This controller is the same used in the proposed system.

The information about environment, which are left obstacle distance, right obstacle distance, front obstacle distance and goal angle represent inputs to FLC.

A set of fuzzy rules are executed. The outputs of the FLC are left and right wheels velocities. The fuzzy rules advise robot to avoid obstacle and find target.

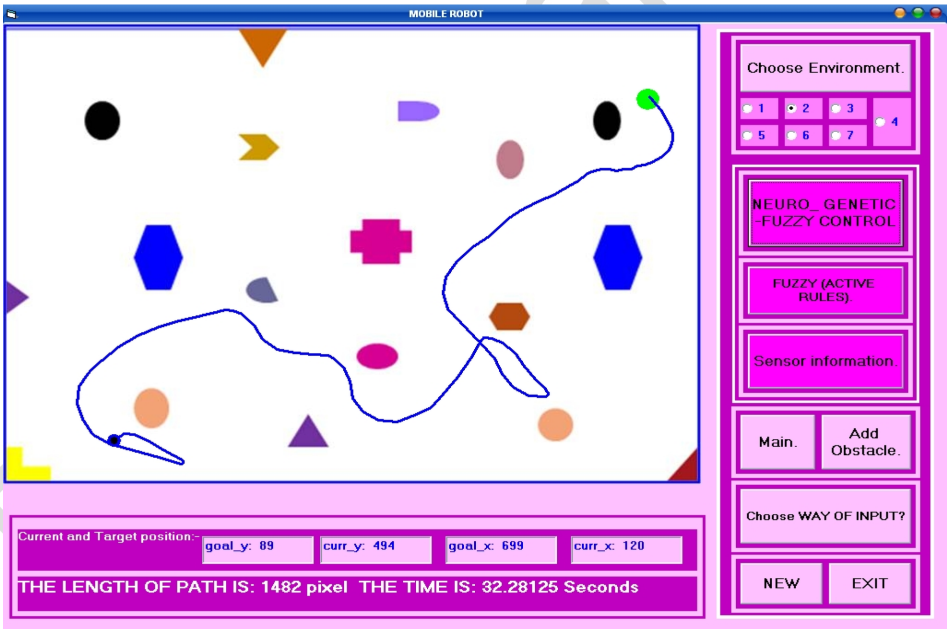

Fig. 6. Path of robot in unknown environment from source to destination point using Fuzzy logic controller

In the Fig. 6, mobile robot success to get target from same source point to destination point showed by proposed system and can successfully avoid obstacles in time and length of path explained in the Fig. 6. In this controller, the robot spent 32 seconds to get target and distance=1482. It requires distance greater than proposed system with longer time comparing with Fig. 4.

4.2 Neuro - Fuzzy Mechanism to Control of Mobile Robot

In this controller, the neural network with fuzzy logic is used by mobile robot to evaluate proposed system's performance. This neural is the same used in the proposed system. Fuzzy is showed in the previous section.

The information about environment, which are left obstacle distance, right obstacle distance, front obstacle distance and goal angle represent inputs to neural network. The output of neural network is steering angle. These inputs with steer angle are inputs to fuzzy logic, then outputs of neuro-fuzzy controller are left wheel velocity and right wheel velocity.

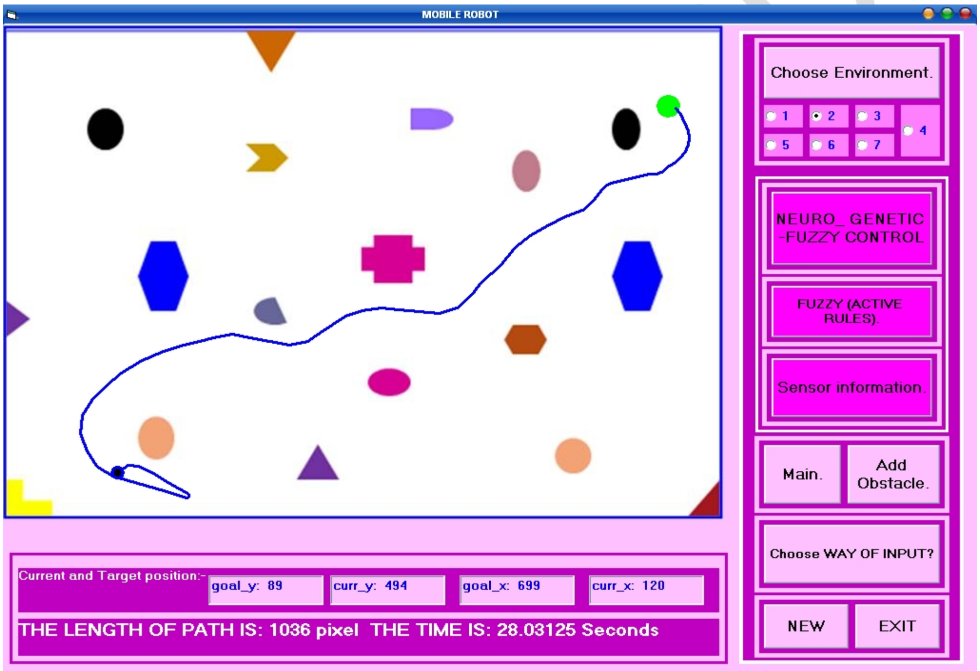

Fig. 7. Path of robot in unknown environment using Neuro - Fuzzy controller

In the Fig. 7, mobile robot success to get target from same source point to destination point showed by proposed system and can successfully avoid obstacles in time and length of path explained in the Fig. 7.

In this controller, the robot spent 28 seconds to get target and distance=1036. It requires distance greater than proposed system with longer time comparing with Fig. 4.

4.3 Discussion

In this section, a comparison of the performance of the proposed controller (neuro – genetic - fuzzy) with that of the neuro - fuzzy controller and the fuzzy controller that are executed is shown in Table 1.

Table 1. Performance comparison

Methods	Length of path.(in pixels)	Time of run. (in seconds)
Fuzzy logic Controller.	1482	32
Neuro – Fuzzy Controller.	1036	28
Geno –Fuzzified - Neural Controller.	757	14

In this table, the results of controllers that get it previously are shown to compare performance of the proposed system with other controller. This table shows that proposed system proves its efficiency over other controllers in time and length of path.

5 Conclusions

The best performing techniques are based on geno – fuzzified –neural technique, which gives robust navigation results in an unknown environment. This technique provides better results than other techniques.

Geno – fuzzified – neural controller is efficient with different numbers, shapes, sizes of obstacles in unknown environments. Although neural network resulted steer angle of robot, fuzzy logic has active role with neural for improving performance of system by time and length of path. Genetic algorithm has effective role for generating the rules and minimize it. It is also improve the membership function of variables of rules.

Collision avoidance has the highest priority and therefore, in this case, robot main reactive behaviour is steer far to avoid obstacle. When the acquired information from the sensors shows that there are no obstacles around robot, its main reactive behaviour is target steer. Intelligent controller mainly adjusts robot motion direction and enables its quickly moves towards the target if there are no obstacles around the robot.

This controller is used by one mobile robot, so petri net can be used as technique to coordinate the movement of many robots. In the current research work, the techniques developed for mobile robot navigation enables the robot to avoid collision with static obstacles. However, further development of the technique may be required for the avoidance of moving obstacles.

The navigational techniques developed in this research work are capable of detecting and reaching the static target. Further modifications in this navigational technique may be executed so that the robot can not only detect dynamic target but also reach them using an optimum path.

Acknowledgements

I would like to thank so much my supervisor Assist. Prof. Dr. Hussein Attya Lafta for his valuable guidance and encouragement during my work.

Competing Interests

Authors have declared that no competing interests exist.

References

[1] Karim Benbouabdallah, Zhu Qi-dan. Genetic Fuzzy Logic Control Technique for a Mobile Robot Tracking a Moving Target, IJCSI International Journal of Computer Science Issues. 2013.

[2] KarimBenbouabdallah, Zhu Qi-dan. A Fuzzy Logic Behavior Architecture Controller for a Mobile Robot Path Planning in Multi-obstacles Environment, Research Journal of Applied Sciences, 2013;5(14):3835-3842.

[3] Hamid Yazdani, Ali Fallah, Seyed Mahmod Hoseini. A new approach for robot Path Planning with Genetic Algorithms, Journal of Basic and Applied Scientific Research. 2012;(4):4122-4129.

[4] Hamani M, Hassam A. Mobile Robot Navigation in Unknown Environment Using Improved APF Method, the 13th international conference in unknown environment using improved APF method. 2012;0875-1812.

[5] Olumide O, Ioan D. Adaptive Neuro-Fuzzy Controler with Genetic Training for Mobile Robot Control, Int. J. of Computers, Communications & Control. March 2012.

[6] Xi Li, Byung-Jae Choi. Design of Obstacle Avoidance System for Mobile Robot using Fuzzy Logic Systems, International Journal of Smart Home. May 2013;7(3).

[7] Er. Waghoo Parvez, Er. Sonal Dhar. Path Planning Optimization Using Genetic Algorithm – A Literature Review, 10th International Journal of Computational Engineering Research. 2013;03(4).

[8] Dubey AD, Mishra RB, Jha AK. Path planning of mobile robot using reinforcement based artificial neural network, International Journal of Advances in Engineering & Technology. 2013;6(2):780-788.

[9] Najimul Hassan S, Deepika S. Fuzzy based mobile robot navigation in an unknown environment, International Conference on Computing and Control Engineering (ICCCE 2012); April 2012.

[10] Raja P, Pugazhenth S. Path planning for a mobile robot in dynamic environments, International Journal of the Physical Sciences. 23 September 2011;6(20):4721-4731.

[11] Wei X, Jiachen M, Mingli Y, Qi Z, Huiying G. A New Modeling Method for Intelligent Service Mobile Robot, Research Journal of Applied Sciences, Engineering and Technology. 2011;3(10):1164-1170.

[12] Siti N, Anggina P. Modeling of Mobile Robot System with Control Strategy Based on Type-2 Fuzzy Logic, International Journal of Information and Communication Technology Research. 2012;2(3).

[13] Xianhua J, Yuichi M, Xingquan Z. Predictive Fuzzy Logic Controller for Trajectory Tracking of A Mobile Robot, IEEE Mid-Summer Workshop on Soft Computing in Industrial Applications; 2005.

[14] Youssef B. Neural Network model for path-planning of robotic rover systems, International Journal of Science and Technology (IJST), E-ISSN. February 2012;2(2):2224-3577.

[15] Oscar Cordon, Francisco Herrera, Frank Hoffmann, Luis Magdalena. Hand book of genetic Fuzzy systems, World Scientific; 2001.

Peer-review history:
The peer review history for this paper can be accessed here (Please copy paste the total link in your browser address bar)
www.sciencedomain.org/review-history.php?iid=240&id=6&aid=1874