International Journal of Engineering & Technology, 7 (4) (2018) 2256-2262

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET
doi: 10.14419/ijet.v7i4.16700
Research paper

Design fuzzy neural petri net controller for trajectory tracking control of mobile robot

Ameer L. Saleh [1] *, Mohammed J. Mohammed [2], Ahmed Sabri Kadhim [3], Hana'a M. Raadthy [4], Hesham J. Mohammed [5]

[1] *Department of Electrical Engineering, College of Engineering, University of Misan, Iraq*
[2] *Department of Information and Communications, Basra University College of science and technology, Iraq*
[3] *Department of Electronic Technologies, Basra Technical Institute, Southern Technical University, Iraq*
[4] *Department of Electronic Technologies, Basra Technical Institute, Southern Technical University, Iraq*
[5]*Fustka ice cream company Basra , Iraq*
**Corresponding author E-mail: the_prince377@yahoo.com*

Abstract

In this paper, a Fuzzy Neural Petri Net (FNPN) controller has been designed established on Particle Swarm Optimization (PSO) for controlling the path tracking of Wheeled Mobile Robot (WMR). The path planning controller problem has been solved using two FNPN controllers to get the desired velocity and azimuth. The PSO method has used to detection the optimal values parameters of FNPN controllers. The overall models of wheeled mobile robot for path tracking control created on PSO algorithm are implemented in Simulink-Matlab. Simulation outcomes demonstrate the suggested FNPN controllers is more effectiveness and has good dynamic performance than the conventional methods.

Keywords: *Mobile Robot; Modeling and Simulation; Trajectory Tracking Technique; PSO Algorithm; FNPN Controller.*

1. Introduction

Nowadays, wheeled wheel robots have been generally utilized in many applications such as manufacturing automation, in hazardous areas and inaccessible areas such as space and war environments, chemical waste cleaning, individual use in different service, etc. Is a very important field in science. Humans have tried to build an independent robot several years ago [1].

The development of another advances of individuals end up genuine. Mechanical autonomy draws in like manner individuals not just those informed in this field. The wheeled versatile robots are the most prevalent outlines [2]. There are two classes for robot which are stage robots and portable robots. the stage robot is mounted on a one physical area and materials transport to the stage close to the robot. The stage robot is generally used in auto processing plants, for welding or stamping which implies that is utilized as a part of large scale manufacturing. Versatile robots are not settled at one area and they can move around in their condition; along these lines, this sort of portable robots can be depicting as a movement gadget that execute computerized assignments, by utilizing manmade brainpower (AI) methods, contingent upon the human introduction or a particular program. Versatile robots likewise separated into wheeled, followed or legged robots, and they are more helpful than stage robots [3-4].

Automated frameworks have non-holonomic conduct which makes them especially fascinating and accessible in light of the fact that the most portable robots are non-holonomic wheeled mechanical frameworks which is caused troublesome control on the development of the wheeled has three points of adaptability to control the versatile robot, two controller movements further down the non-holonomic kinematics [3].

When all is said in done, the portable robot's route control issues can be requested into three orders: The primary arrangement is the position estimation of the robot in its workplace is one of huge issues in versatile robots which can be understood by utilizing its on-board sensors with dynamic condition changes. The second order is direction arranging and age. The direction arranging is executed by using certain enhancement procedures with taking parameters other than robot's elements and kinematics into thought. The third characterization is outlining and executing the route control framework so the versatile robot must have the capacity to track it to get the coveted direction with high precision and least mistake [3-5].

The portable robots have different practices that could be displayed way arranging, way following, Trajectory(path) following, versatile objective chasing. The way following control of portable robot is a basic part for present day robots. The principle undertaking for any portable robots is to proceeding onward a particular way. Lately, numerous scientists have been managing looks into that worry with control methods to control way following. Moreover, numerous controlling systems have been used, for instance, sliding mode control, back venturing procedure, versatile control, fluffy control, and neural system control, and so on [5-6].

This paper describes first the dynamic and kinematic model which used to simulate the movement of the mobile robot in an environment with used path planning and path tracking methodology to calculate the distance and the angle that has to be covered in order to meet the destination.

This paper present an effective and more important controller based on fuzzy neural petri net (FNPN) controller are used for controlling the wheeled mobile robot. The PSO algorithm used for numerical calculation of optimal FNPN controller parameters which is used to adjust the linear velocity and the azimuth for the wheeled mobile robot and give better performance.

2. Modeling of mobile robot

Modeling and simulation for mobile robot has been designated in kinematic and dynamic model. The model deals with the geometrical relations that studies the mathematics of motion and dominate the system with taking into consideration effect the kinematics and dynamic forces.

2.1. Kinematic model

These model examinations the science of movement paying little respect to the powers that influence the movement. It is managing the geometrical relations that command the framework, and furthermore manages the connection between the movement of a framework and the control parameters [4-6].

Fig.1. demonstrate the kinematics planner of a two-wheel robot. Where the universal coordinate represented in (O,X,Y), v is a robot velocity, w is a robot angular velocity, v_l the left driving wheel velocity, v_r the right motivating wheel velocity, 2b is a space in the middle of the two motivating wheels, r is a driving wheels' radius, the position of the robot represented by x and y, and finally θ is a the orientation of the mobile robot.

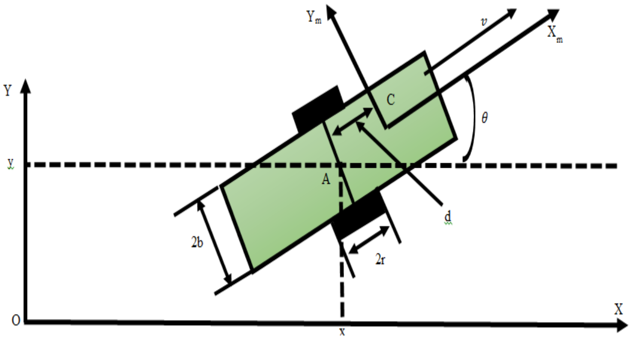

Fig. 1: Demonstrates the Kinematics Planner of A Two-Wheel Mobile Robot.

Equations (1, 2) represent the motion differential drive for a two-wheel mobile robot [1].

$$v_l = w_l r \quad (1)$$

$$v_r = w_r r \quad (2)$$

Where w_l and w_r are angular linear speeds for left and right motivating wheels in that order [4-7].

The non-holonomic limitation equation of the robot is as following:

$$\dot{x}\sin\theta - \dot{y}\cos\theta = 0 \quad (3)$$

From the directly above formula the following equations have been obtained:

$$v = \frac{w_r + w_l}{2} r, \qquad w = \frac{w_r - w_l}{2b} r \quad (4)$$

Equation (5) defined the dynamic function of the robot:

$$\dot{x} = v\cos\theta, \qquad \dot{y} = v\sin\theta, \qquad \dot{\theta} = w \quad (5)$$

The combination of the latest two equations will produced the following:

$$\begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{\theta}\end{bmatrix} = \begin{bmatrix}\frac{r}{2}\cos\theta & \frac{r}{2}\cos\theta\\ \frac{r}{2}\sin\theta & \frac{r}{2}\sin\theta\\ -\frac{r}{2b} & \frac{r}{2b}\end{bmatrix}\begin{bmatrix}w_l\\ w_r\end{bmatrix} \quad (6)$$

Accordingly, equation (6) should be separated. Where θ is just related to ω , x and y are just related only to v. So, the kinematics conventional model of the mobile robot is found as:

$$\begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{\theta}\end{bmatrix} = \begin{bmatrix}\cos\theta & 0\\ \sin\theta & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}v\\ w\end{bmatrix} \quad (7)$$

Where $\dot{x}$ and $\dot{y}$ represent the robot velocity in the direction of X-axis and Y-axis, respectively, v represent the robot linear velocity, θ' represent the robot rotational velocity. Therefore, the kinematics classical model of the robot can be characterized by the following matrix:

$$\begin{bmatrix}v\\ \dot{\theta}\end{bmatrix} = \begin{bmatrix}\frac{r}{2} & \frac{r}{2}\\ \frac{r}{2b} & -\frac{r}{2b}\end{bmatrix}\begin{bmatrix}w_l\\ w_r\end{bmatrix} \quad (8)$$

2.2. Dynamics of mobile robot

In this paper, a dynamic typical model of mobile robot will be derived according to a non-holonomic robot scheme is shown in fig.1. The location of the mass middle point of the robot can be used to defined the robot position in the complete coordination scheme {X, O, Y}. The equations of kinematic moveable mobile robot can be described from fig. (1) as follows [1], [6] [8]:

$$I_v\ddot{\varphi} = D_r I - D_l I \quad (9)$$

$$M\dot{v} = D_r + D_l \quad (10)$$

$$I_w\ddot{\varphi}_i + c\dot{\varphi}_i = k u_i - r D_i \ (i = r, I) \quad (11)$$

Where φ the Azimuth of robot, I_v Moment of inertia around the C.G. of robot, D_r and D_l are the forces for right and left driving, respectively; l is the space flanked by the right and left steering wheel; M is the mass of robot, v the linear speed of the robot, I_w the moment of inertia of wheel, c the viscous friction factor, k the driving gain factor, u_i is the driving input, r the radius of wheel. further, the geometrical relationships between the three variables φ, v, θ. Given by:

$$r\dot{\theta}_r = v - I\dot{\varphi} \quad (12)$$

$$r\dot{\theta}_l = v - I\dot{\varphi} \quad (13)$$

The mathematical model for the dynamic part of mobile robot can be described in state space. Where the state variables for the robot is defined as X= [, v, θ], the worked adaptable inputs variables such as u=[u_r, u_l] and the adaptable output variable such as y = [v, θ] . Then:

$$\dot{x} = Ax + Bu \quad (14)$$

$$y = Cx \quad (15)$$

Where:

$$A = \begin{bmatrix}a_1 & 0 & 0\\ 0 & 0 & 1\\ 0 & 0 & a_2\end{bmatrix} \qquad b = \begin{bmatrix}b_1 & b_1\\ 0 & 0\\ b_2 & -b_2\end{bmatrix}$$

$$C = \begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\end{bmatrix}$$

$$a_1 = \frac{-2c}{Mr^2 + 2I_w} \qquad a_2 = \frac{-2cI^2}{I_v r^2 + 2I_w I^2}$$

$$b_1 = \frac{k\,r}{Mr^2 + 2I_w} \qquad b_2 = \frac{-krI}{I_v r^2 + 2I_w I^2}$$

The following equations (16) and (17) represent the relation between the input torques to the robot (u_r and u_l) and the output of the controller(u_v and u_φ):

$$u_r = u_v + u_\varphi \tag{16}$$

$$u_l = u_v - u_\varphi \tag{17}$$

Where u_v, u_φ are the torque required to control the linear speed and azimuth of mobile robot

3. Navigation of a mobile robot

Route of versatile robots in a hazy and vague condition which is spoken to a stress for the portable robot because of different deterrents that versatile robots confronted it in his way which need to distinguished and dodged with no impacting. Route of the versatile robots can be created by using wise frameworks with enhancement strategies. The route issue control of portable robots is arranged to three conceivable movement undertakings as takes after [1], [4], and [8]:

4. Path tracking (trajectory tracking)

A ton of looks into are occupied with the direction following control for portable robots. It alludes to the situation where a robot needs to direction the way as indicated by a period reference [9-10]. In the trajectory tracking control system the velocity error ev and the azimuth error e_θ are represented as the inputs while the developing torque desired for driving the wheels u_r and u_l are represented as the output. The following equations represent input deviation e_v and e_φ:

$$e_v = v_d - v \tag{18}$$

$$e_\varphi = \varphi_d - \varphi \tag{19}$$

Where v_d, represents the desire linear speed, φ_d is the reference azimuth, v is the actual linear speed, and φ the azimuth of the mobile robot.

5. Path planning

The delinquent covenants with discovering the track near the certain objective. The difficult develops even supplementary multifaceted if the robot requirements to catch an optimum track known roughly restrictions comparable shortest path or smallest time or even minimization of motors power [4-6]. First step, the mobile robot can travel in the direction of the favorite power point since it distinguishes first and last point. The distance between robot and its objective is [4-5]:

$$d = \sqrt{(y_{trg} - y_{rbt})^2 + (x_{trg} - x_{rbt})^2} \tag{20}$$

$$\varphi = tan^{-1}\frac{y_{trg}-y_{rbt}}{x_{trg}-x_{rbt}} \tag{21}$$

Where

$y_{trg} = y_{target}$ $y_{rbt} = y_{robot}$, $x_{trg} = x_{target}$ and $x_{rbt} = x_{robot}$

Allowing for that φ is the favorite angle attendant to the conventional distance starting from the present location towards the object.

6. Fuzzy neural petri net (FNPN)

The configuration of the projected FNPN is displayed in Fig.2, and the system scheme has the subsequent three stratums and them can be called as layers [11-12]:

- Input layer containing of N input spaces, they can be called input layer.
- Transmission layer containing of hidden layers, they can be called transition layer.
- Output layer containing of M output spaces, they can be called output layer.

Input layers are noticeable by the rate of the quality. Transitions turn as dispensation components. The firing has been determined by the factors of transmission layer, which called as thresholds, and the factors of the relations are called weights. The shapes of the output apartment reproduce a side by side of membership of the form in the conforming class.

The system requirement is presented in Fig. 3 as shadows [11-12]: Pj is the pattern equal of j-th input place manufactured by a triangular charting function. The maximum of the triangular charting function is positioned on the average opinion of the input principles. The dimension of triangular base is designed since the variance in the middle of the smallest and determined values of the contribution input, the structure is found in Fig.2 [11].

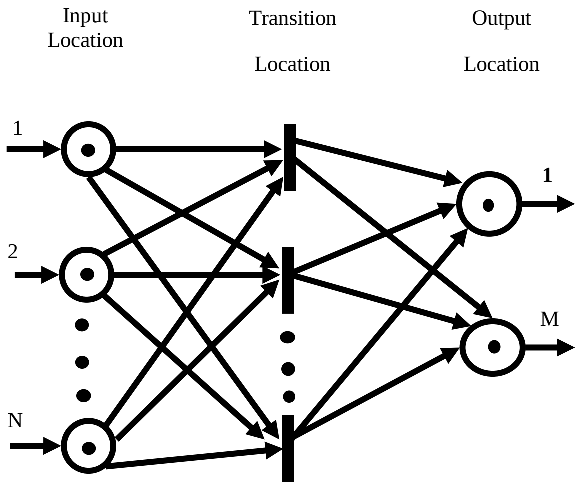

Fig. 2: The Structure of FNPN.

The altitude of the triangle is union. This development holds onto the involvement of the system inside the period [0, 1]. This oversimplification of the Petri Net will be in complete stabilization with the two-valued general description of the Petri Net [11].

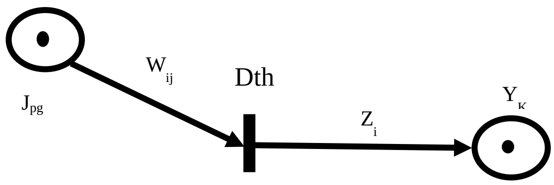

Fig. 3: Fragment Net Frameworks the Systems.

$$P_j = f(input(j)) \tag{22}$$

Where f is a trilateral planning formula that is displayed in Fig. 4.

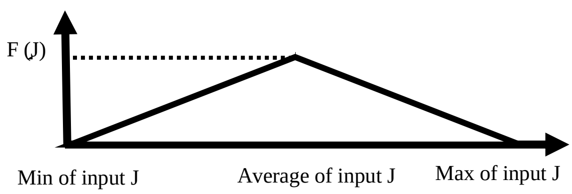

Fig. 4: The Triangular Mapping Function.

W_{ij} is the weightiness between the i-th transition and the j-th input apartment, d_{th} is a threshold level accompanying with the side by side of design of the j-th input place and the i-th transition, Z_i is the activation level of i-th transition and defined as founds in [11]:

$$f(j) = \begin{cases} \frac{j-min(j)}{avg(j)-min(j)} & if\ j < avg(j) \\ \frac{max(j)-j}{max(j)-avg(j)} & if\ j > avg(j) \\ 1 & if\ j = avg(j) \end{cases} \quad (23)$$

- Petri Net Layer

This layer is applied to yield demonstrations that create use of emulation commandments for node firing as found in [11]:

$$t_{ij} = \begin{cases} 1 & if \quad \mu_{ij} \geq d_{th} \\ 0 & if \quad \mu_{ij} < d_{th} \end{cases} \quad (24)$$

Where t_{ij} is the value of transition point and d_{th} is the efficient value of threshold point that show a discrepancy with miscalculation and it has been modified by the next formula [11]:

$$d_{th} = \frac{\alpha e^{(-\beta E)}}{1+\alpha e^{(-\beta E)}} \quad (25)$$

Where α and β are authentic factors can be selected arbitrarily. If the error develops big, the threshold magnitude will be reduced to excitement further rules for the present position. Of course unique can use a persistent magnitude for the threshold.

- Rule Layer

The produce of output at respectively output node is the result of its inputs values and hidden values are followed in [11]:

$$t_{ij} = \begin{cases} \prod_i^n \mu_{ij} & if \quad t_{ij} = 1 \\ 0 & if \quad t_{ij} = 0 \end{cases} \quad (26)$$

Where $\emptyset_j$ is the production of the j_{th} node of the rule layer; n is the measure of crisp inputs.

- Output Layer

Output node determines complete output y as a collection of inputs indications as found in [11]:

$$y = \sum_j^{ni} w_j \emptyset_j \quad (27)$$

Where the construction weights w_j is the output accomplishment strong point connected with the j_{th} rule; n_i is the quantity of rules.

7. Particle swarm optimization

PSO is a randomly determined optimization technique which resulting from the fundament of cooperative actions of natures (bird congregates, fish groups) in the course of their investigation on sustenance.

Each constituent part protected own aforementioned knowledge and involvement of bordering constituent part in its group to choice the best location for itself which known as the personal best position (known as the Pbest). When a constituent part in the entire flight consume the best position in the group which is called the global best position(gbest), the best value of FNPN in all search to give the best fitness function or saying the best values to give minimum error in trajectory. Where the best foregoing location of any constituent part is known as local best position (lbest), the best values of FNPN parameters in iteration k. The presentation of apiece constituent part to adopt whether the best clarification is consummate conferring to the dispassionate purpose [13-15]. The dispassionate purposes reconnoitered are founded on the looked-for criterion. The most corporate presentation principles are founded on the miscalculation criterion such as Integrated Absolute Error (IAE), Integrated of Time weight Square Error (ITSE) and Integrated of Error Square (ISE) [15-17]. Miscellany of these principles be determined by on both organization and controller. The speed and location of each constituent part in the group are modernized by using the next formulas:

$$S_i^{k+1} = w\, S_i^k + c_1 R_1 \left(lbest_i - Q_i^k\right) + c_2 R_1 \left(gbest_i - Q_i^k\right) \quad (28)$$

$$Q_i^{k+1} = Q_i^k + S_i^{k+1} \quad (29)$$

Where: Q_i^k is the direct location of the value of FNPN parameters in i at iteration k ; v_i^k is the jump step of the value of FNPN parameters in i at iteration k; c1 andc2 are the hastening coefficients; R1, R2 are arbitrary variables sandwiched between 0 and 1.

In this article, an Integral of Squared Error (ISE) dispassionate function is applied to treasure trove the optimal explanation with a minimum speed and azimuth miscalculation. The cost fitness is calculated as found in [14-18]:

$$fitness\ function = \ min(ISE) \quad (30)$$

Where

$$ISE \ = \int e^2\,(t)dt \quad (31)$$

$$e(t) = ev^2(t) \ + \ e\theta^2(t) \quad (32)$$

In PSO algorithm apiece constituent part considerations are started to generate a residents and then comprehensive the procedure as in flowchart in Fig.5. which consist of altering the factors of FNPN controller for, to guarantee the minimization of dispassionate function.

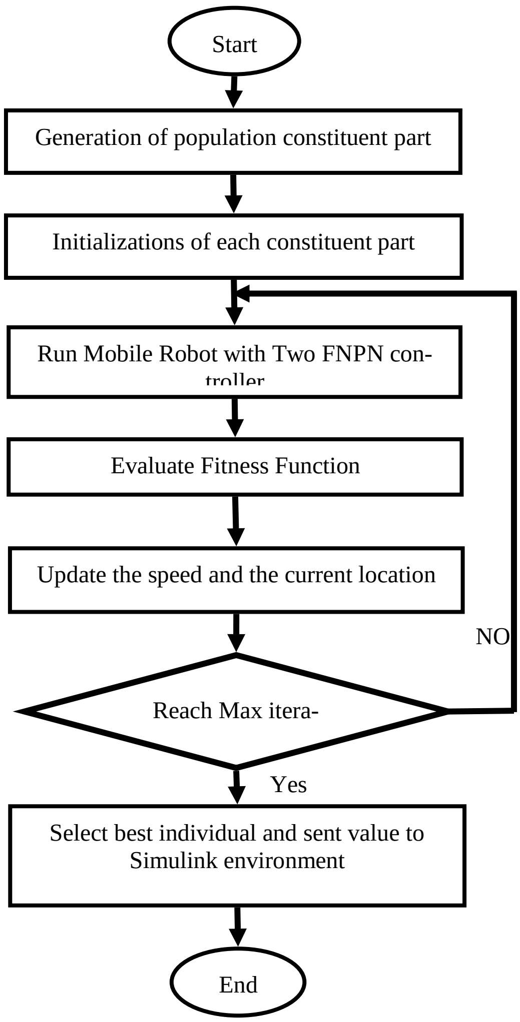

Fig. 5: Flowchart of PSO.

8. Design and Simulink implementation of trajectory tracking for mobile robot using FNPN controller

The Fuzzy Neural Petri Net (FNPN) controller based on PSO algorithm is proposed for controlling the path tracking of Mobile Robot. The dynamic and kinematic model of a mobile robot with the trajectory tracking controller are implemented according to the mathematical equations given in sections [2-3]. The overall block diagram of mobile robot with path tracking control system is shown in Fig.6.

In this paper, a FNPN controller is proposed to ensure the mobile robot can track target trajectory with high precision. Two FNPN controllers are utilized for motion control of mobile robot. The first one of FNPN controller is utilized to control the linear speed and other to control azimuth of the mobile robot. A PSO algorithm is utilized for tuning the parameters of FNPN controller to get better performance according to the fitness function. Fig.7 shows the complete Simulink model of the mobile robot control with two FNPN controllers. While the Simulink model of FNPN controller is shown in Fig. 8.

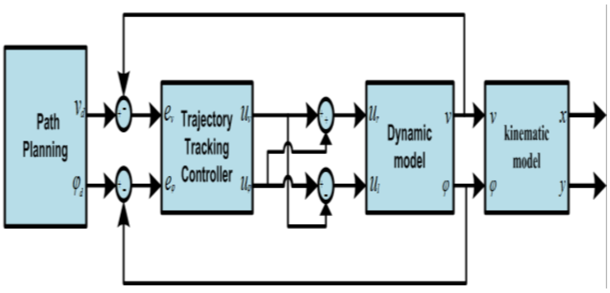

Fig. 6: Overall Block Diagram of Mobile Robot with Trajectory Tracking Control System.

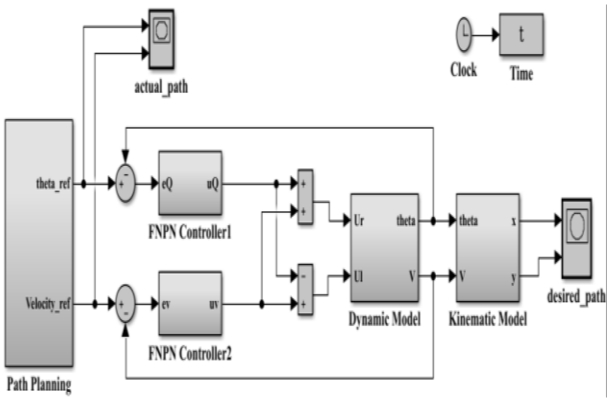

Fig. 7: The Overall Simulink Model of the Mobile Robot with Path Tracking Control Using Two FNPN Controllers.

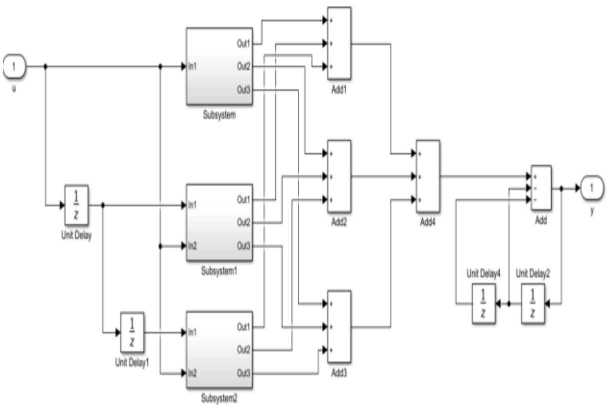

Fig. 8: The Simulink Model of FNPN Controllers.

9. Simulation results

The simulink model of mobile robot with FNPN controller based on PSO algorithm for controlling the trajectory tracking is implemented in Simulink / Matlab program. The mobile robot 's parameters values in simulation are taken from [7]: M = 24 kg, Iv=0.4732 kg.m2, Iw=0.0198 kg.m2, l=0.36m, r=0.057m, c=0.15833kg/s, k=1.7.

Fig.9 shows the optimization value fitness function (optimization function) of the two FNPN using PSO algorithm.

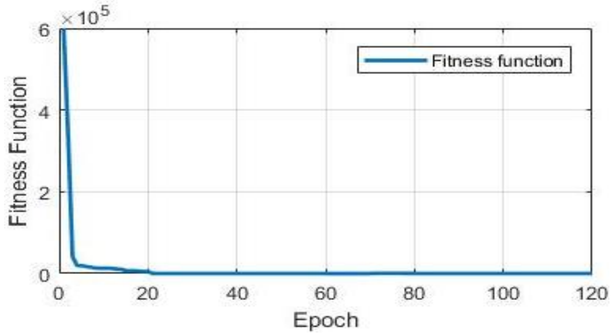

Fig. 9: Fitness Function.

Fig.10 shows the optimization value for the parameters of the two FNPN using PSO. This paper has been solved path tracking problem of navigation for wheeled mobile robot. In path tracking strategy using different shapes to investigation the goal of the FNPN controller with minimum error.

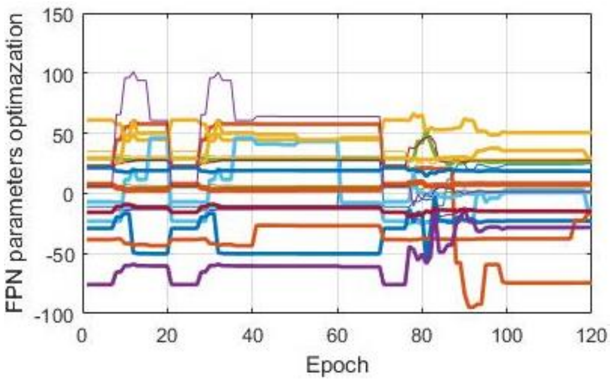

Fig. 10: FNPN Parameters under Optimization.

Fig.11 shows the optimization value for memberships for FNPN and sketch at optimization values.

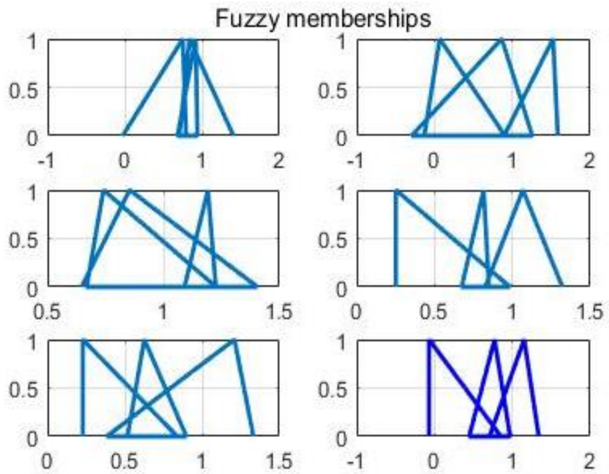

Fig. 11: FNPN Memberships.

In this paper, three desired trajectories are tracked in order to investigation the features of FNPN controller:

1) Linear Trajectory

To achieve the mobile robot tracking through a Linear path with linear desired velocity vd = 0.25 [m/sec]. While the X-Y axis for linear path tracking and Error in X-Y for linear path tracking is shown in Figs. (12-14).

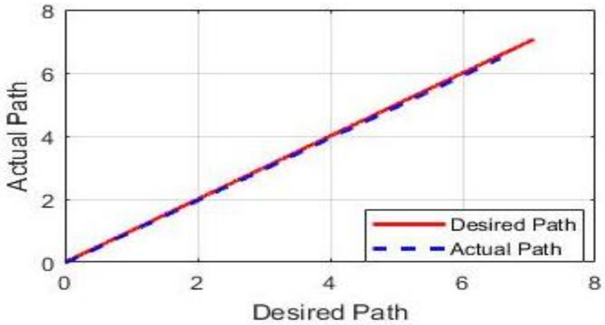

Fig. 12: Linear Path Tracking.

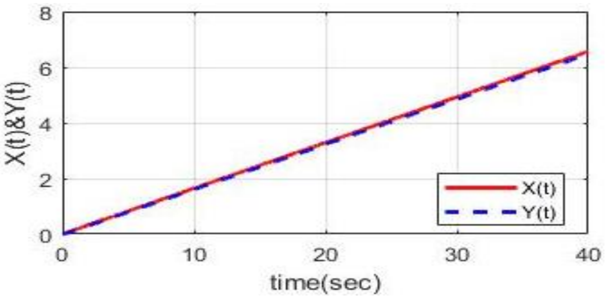

Fig. 13: X&Y for Linear Path Tracking.

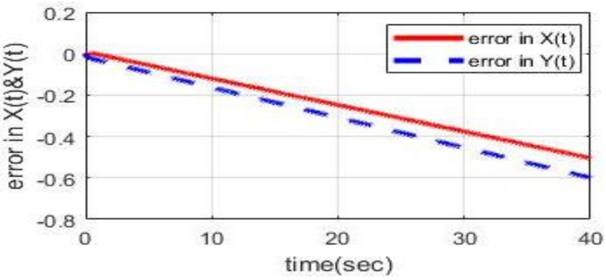

Fig. 14: Error in X&Y for Linear Path Tracking.

2) Circular Trajectory

To achieve the mobile robot tracking through a circular path with desired linear velocity v_d = 0.25 [m/sec] and desired azimuth given as:

θ_d = (2 ∗ 3.14 ∗ f (t)/m) rad, with m= -50 (33)

Where: f (t) = t , 0 ≤ t ≤ 50, m is the slop and T_s= 0.001 sec. The simulation result of the reference and actual circular path is presented in Figs. (15-17).

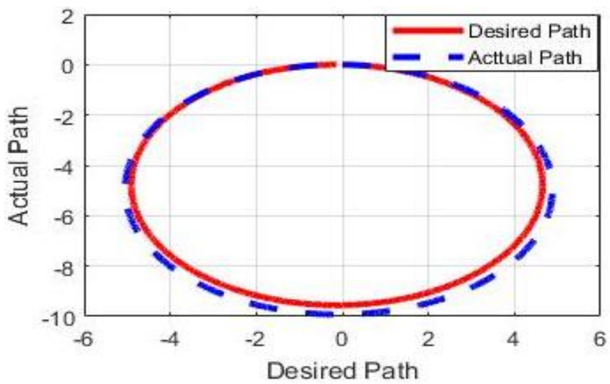

Fig. 15: Circular Path Tracking.

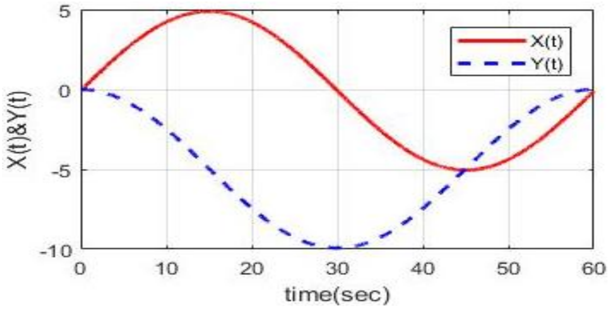

Fig. 16: X&Y for Circular Path Tracking.

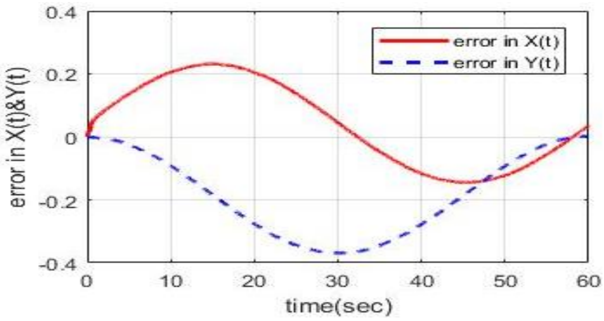

Fig. 17: Error in X&Y for Circular Path Tracking.

3) Sinusoidal Trajectory

To achieve the mobile robot for tracking square path with desired velocity v_d = 1 [m/sec] and desired azimuth given by: $\theta_d=\sin(t)$. The simulation result of the sinusoidal trajectory is shown in Figs. (18-20).

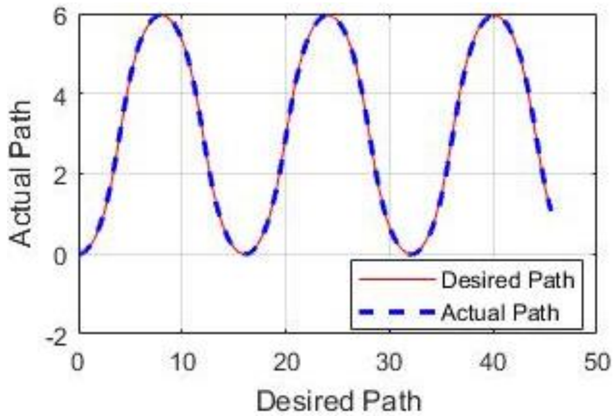

Fig. 18: Sinusoidal Path Tracking.

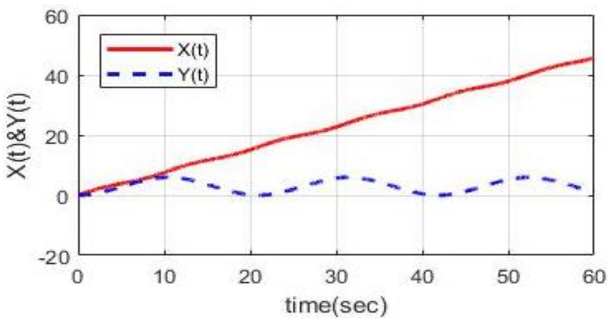

Fig. 19: X&Y for Circular Path Tracking.

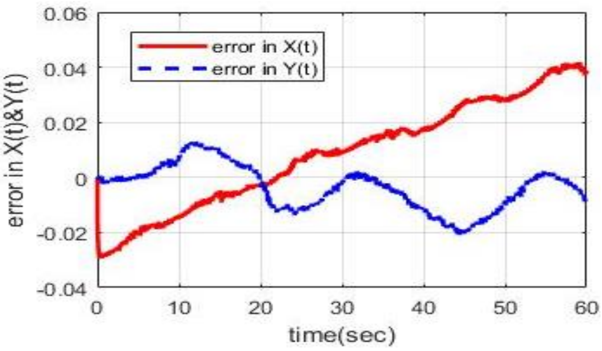

Fig. 20: Error in X&Y for Circular Path Tracking.

10. Conclusion

This paper presents FNPN controller with PSO algorithm for controlling the path tracking of mobile robot. The trajectory tracking problematic for the velocity and azimuth of a wheeled mobile robot has been solved by utilizing two FNPN controllers with PSO method. This controller offers a required performance as it makes the robot able to track the different tracks accurately so it is eligible to work in difficult and dangerous environments. The PSO method is utilized for learning the parameters of FNPN controllers. The simulated results demonstrate that the proposed controller is more efficient to solve problem of target tracking with different paths.

References

[1] Ameer L. Saleh Maab A. Hussain and Sahar M. Klim, Optimal trajectory tracking control for a wheeled mobile robot using fractional order PID controller, Journal of University of Babylon, Engineering Sciences, 26 (4), 2018 pp. 292-306. https://journalofbabylon.com/index.php/JUBES/article/view/1087

[2] Auday Al-Mayyahi1 Weiji Wang1 and Phil Birch1, Design of fractional-order controller for trajectory tracking control of a non-holonomic autonomous ground vehicle, Springer, J Control Autom Electr Syst, vol. 27, 2016, pp. 29-42. https://link.springer.com/article/10.1007/s40313-015-0214-2

[3] Khulood E. and Ahmed Al-Araj, Design of a nonlinear PID neural trajectory tracking controller for mobile robot based on optimization algorithm, Eng. & Tech Journal, Vol.32, part (A), no.4, 2014. https://www.iasj.net/iasj?func=fulltext&aId=88219

[4] Turki Y. Abdalla and Mustafa I. Hamzah, Trajectory tracking control for mobile robot using wavelet network, International Journal of Computer Applications (0975 – 8887) vol. 74, no.3, pp 32-37, July 2013. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.403.2341

[5] Emina P. Saša P. Vlastimir N. Marko M. Ivan Ć. Boban R. and Miloš S. Kinematic model and control of mobile robot for trajectory tracking, ANNALS of Faculty Engineering Hunedoara – International Journal of Engineering, May 2016. http://annals.fih.upt.ro/pdf-full/2016/ANNALS-2016-2-25.pdf

[6] Thoa T. Mac Cosmin Copot Robin De Keyser Trung D. Tran and Thich Vu, MIMO fuzzy control for autonomous mobile robot, Journal of Automation and Control Engineering Vol. 4, No. 1, pp. 65-70, February 2016. https://biblio.ugent.be/publication/7034246

[7] Altamiro V. Silveira and Elder M. Hemerly, Control of mobile robots via biased wavelet networks, Learning and Nonlinear Models – Revista da Sociedade Brasileira de Redes Neurais, Vol. 2, No. 2 pp. 84-98, 2004. http://abricom.org.br/wp-content/uploads/sites/4/2016/07/vol2-no2-art3.pdf

[8] M. J. Mohammed M. T. Rashid and A. A. Ali. Design optimal PID controller for quad rotor system, International Journal of Computer Applications 106, Volume 106 – No.3, November 2014. https://www.ijcaonline.org/archives/volume106/number3/18500-9565

[9] Jakub Osuský and Ján Cigánek, Trajectory tracking robust control for two wheels robot, IEEE, 2018 Cybernetics & Informatics (K&I), Slovakia, Feb. 2018. https://doi.org/10.1109/CYBERI.2018.8337559

[10] Ali Keymasi Khalaji and Mostafa Jalalnezhad, Modeling and backstepping control of a wheeled robot, 2017 IEEE 4th International Conference on Knowledge-Based Engineering and Innovation (KBEI), Iran, Dec. 2017. https://doi.org/10.1109/KBEI.2017.8324950

[11] Mohammed J. Mohammed Abduladhem A. Ali and Mofeed T. Rashid, Fuzzy petri net controller for quadrotor system using particle swam optimization, Iraq J. Electrical and Electronic Engineering, Vol. 11 No. 1 pp 132-144,2015. https://www.iasj.net/iasj?func=article&aId=102735

[12] Ali A. Abed Abduladhem A. Ali Ali F. Marhoon and Nauman-Aslam, A field bus network with CAN protocol and a fuzzy neural petri net controller, Basrah Journal of Science,31, no.2, pp. 86-102, 2013. https://www.iasj.net/iasj?func=fulltext&aId=73941

[13] S. Marco P. Nobileac D. Cazzanigabc R. Besozziac G. Colomboac and Mauriac G. Pasia, Fuzzy self-tuning PSO: a settings-free algorithm for global optimization, Science Direct journal, Swarm and Evolutionary Computation journal, Vol. 39, pp.70-85, April 2018. https://www.altmetric.com/details/25137846

[14] Ameer L. Saleh and Adel A. Obed, Speed control of brushless DC motor based on fractional order PID controller, International Journal of Computer Applications (0975 – 8887) Vol. 95– no.4, June 2014. https://www.researchgate.net/publication/323693009/download

[15] Niraj Kumar Goswami and Prabin Kumar Padhy, Gain tuning of Lyapunov function based controller using PSO for mobile robot control, IEEE, 11th International Conference on Industrial and Information Systems (ICIIS), vol. 4, no. 6, pp. 775-781, 18 January 2018. https://doi.org/10.1109/ICIINFS.2016.8262954

[16] Adel A. Obed and Ameer L. Saleh, Speed control of bldc motor based on recurrent wavelet neural network, Iraq J. Electrical and Electronic Engineering, Vol.10 No.2, 2014 https://www.iasj.net/iasj?func=fulltext&aId=95599

[17] M. Munlin and M. Anantathanavit, New social-based radius particle swarm optimization, 2017 12th IEEE Conference on Industrial Electronics and Applications (ICIEA), Cambodia, pp. 838 – 843, June 2017. https://doi.org/10.1109/ICIEA.2017.8282956

[18] Xingyang Lu, Xiangyin Zhang, Songmin Jia and Jichao Shan, Design of quadrotor hovering controller based on Improved particle swarm optimization, 2017 10th International Symposium on Computational Intelligence and Design (ISCID), China, pp. 414-417, Dec. 2017 https://doi.org/10.1109/ISCID.2017.196