Design of a Nonlinear PID Neural Trajectory Tracking Controller for Mobile Robot based on Optimization Algorithm

Khulood E. Dagher
Science College, University of Baghdad / Baghdad

Dr. Ahmed Al-Araji
Control and Systems Engineering Department, University of Technology/ Baghdad
Email:ahmedsas2040@yahoo.com

Received on: 23/6/2013 & Accepted on: 9/1/2014

ABSTRACT

This paper presents a trajectory tracking control algorithm for a non-holonomic wheeled mobile robot using optimization technique based nonlinear PID neural controller in order to follow a pre-defined a continuous path. As simple and fast tuning algorithms, particle swarm optimization algorithm is used to tune the nonlinear PID neural controller's parameters to find best velocity control actions for the mobile robot. Simulation results show the effectiveness of the proposed nonlinear PID control algorithm; this is demonstrated by the minimized tracking error and the smoothness of the velocity control signal obtained, especially with regards to the external disturbance attenuation problem.

Keywords: Nonholonomic Mobile Robots; Nonlinear PID Controller; Particle Swarm Optimization; Trajectory Tracking; Matlab package.

تصميم مسيطر تتابع مسار عصبي لأخطي تناسبي تكاملي تفاضلي لإنسان آلي متنقل مبني على أساس الخوارزمية ألأمثليه

الخلاصة

يقدم هذا البحث, خوارزمية المسيطر ألتتابعي لمسار عجلة الإنسان الآلي المتحرك باستخدام التقنية ألأمثليه أساسه المسيطر التناسبي التكاملي التفاضلي العصبي اللاخطي لكي يتبع مسار مستمر معرف مسبقا.

أن الخوارزمية المستخدمة تتميز بسرعة وببساطة تنغيم عناصر المسيطر اللاخطي التناسبي التكاملي التفاضلي وذلك باستخدام خوارزمية حشد الجسيمات ألأمثليه وإيجاد أفضل أشارة سرعة للإنسان الآلي المتحرك.

من خلال نتائج المحاكاة, أن فعالية خوارزمية المسيطر اللاخطي المقترح تقوم بتقليل الخطأ ألتتابعي لمسار الإنسان الآلي المتحرك مع توليد أشارة سرعة ناعمة, برغم من وجود التأثير الاضطرابي الخارجي.

INTRODUCTION

In recent years, there has been an increasing amount of research on the subject of wheel-based mobile robots which have attracted considerable attention in various industrial and service applications [1]. For example, room cleaning, lawn mowers,

https://doi.org/10.30684/etj.32.4A.13
2412-0758/University of Technology-Iraq, Baghdad, Iraq

factory automation, transportation, nuclear-waste cleaning, etc [2]. These applications require mobile robots to have the ability to track specified path stably. In general, nonholonomic behaviour in robotic systems is particularly interesting because the most mobile robots are nonholonomic wheeled mechanical systems and the fact of control problems of the mobile robot caused by the motion of a wheeled that has three degrees of freedom while for control of the mobile robot, only two control signals under the nonholonomic kinematics constraints.

During the past few years, several studies have been published for solving mobile robot control problems which can be classified into three categories: The first category is the position estimation control approach for navigation problems of the mobile robot on interactive motion planning in dynamics environments and obstacle motion estimation [3]. Since the working environment for mobile robots is unstructured and may change with time, the robot must use its on-board sensors to cope with dynamic environment changes while for proper motion planning such as environment configuration prediction and obstacle avoidance motion estimation it uses sensory information [4]. The second category for navigation problems of the mobile robot is path planning and execution. The path planning is generated based on a prior map of the environment while the executed path is planned using certain optimization algorithms based on a minimal time, minimal distance or minimal energy performance index. Many methods have been developed for avoiding both static and moving obstacles as presented in [5].

The third category for the navigation problems of mobile robot is designing and implementing the driving control that the mobile robot must track to follow a desired path accurately and minimize the tracking error. Tracking errors of mobile robot causes collisions with obstacles due to deviations from the planned path and also causes the robot to fail to accomplish the mission successfully. It also causes an increase of the traveling time, as well as the travel distance, due to the additional adjustments needed to satisfy the driving sates. There are three major reasons for increasing tracking error for mobile robot:

First of the major reasons for tracking error is the discontinuity of the rotation radius on the path of the differential driving mobile robot. The rotation radius changes at the connecting point of the straight line route and curved route, or at a point of inflection. At these points it can be easy for differential driving mobile robot to secede from its determined orbit due to the rapid change of direction [6].

Therefore, in order to decrease tracking error, the trajectory of the mobile robot must be planned so that the rotation radius is maintained at a constant, if possible.

Second of the major reasons for increasing tracking error is due to the small rotation radius interferes with the accurate driving of the mobile robot. The path of the mobile robot can be divided into curved and straight-line segments. While tracking error is not generated in the straight-line segment, significant error is produced in the curved segment due to centrifugal and centripetal forces, which cause the robot to slide over the surface [6].

The third of the major reasons for increasing tracking error is due to the rotation radius is not constant such as the complex curvature or randomly curvature, that is, the points of inflection exist at several locations lead to the mobile robot wheel velocities need to be changed whenever the rotation radius and traveling direction are changed [7].

The traditional control methods for trajectory tracking the mobile robot have used linear or non-linear feedback control while artificial intelligent controller were carried out using neural networks or fuzzy inference [8]. There are other techniques for trajectory tracking controllers such as predictive control technique. Predictive approaches to path tracking seem to be very promising because the reference trajectory is known beforehand. Model predictive trajectory tracking control was applied to a mobile robot where linearized tracking error dynamics was used to predict future system behaviour and a control law was derived from a quadratic cost function penalizing the system trucking error and the control effort [9]. In addition, an adaptive trajectory-tracking controller based on the robot dynamics was proposed in [10]. Intelligent control architecture for two autonomously driven wheeled robot was developed in [11] that consists of the fuzzy inference as main controller and the neural network is an auxiliary part.

The novelty of the presented approach here can be understood considering the following points.

- The analytically derived control law which has significantly high computational accuracy to obtain the best control action and lead to minimum tracking error of the mobile robot based on optimization algorithm.
- Investigation of the controller robustness performance through adding boundary unknown disturbances.
- Verification of the controller adaptation performance through change the initial pose state.
- Validation of the controller capability of tracking continuous gradient trajectories.

Simulation results show that the proposed controller is robust and effective in terms of minimum tracking error and in generating best velocity control action despite of the presence of bounded external disturbances.

The remainder of the paper is organized as follows: Section two is a description of the kinematics model of the nonholonomic wheeled mobile robot. In section three, the proposed nonlinear PID neural controller is derived. The simulation results of the proposed controller are presented in section four and the conclusions are drawn in section five.

MODEL OF A NONHOLONOMIC WHEELED MOBILE ROBOT

The schematic of the nonholonomic mobile robot, shown in Figure (1), consists of a cart with two driving wheels mounted on the same axis and an Omni-directional castor in the front of cart. The castor carries the mechanical structure and keeps the platform more stable [12 and 13]. Two independent analogous DC motors are the actuators of left and right wheels for motion and orientation. The two wheels have the same radius denoted by r, and L is the distance between the two wheels. The center of mass of the mobile robot is located at point c, centre of axis of wheels.

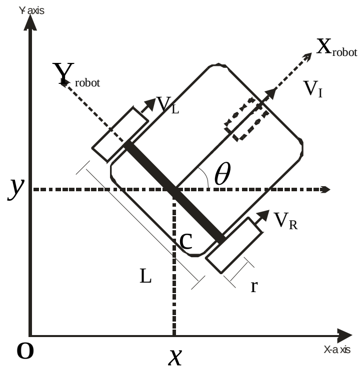


Figure (1) Nonholonomic mobile robot model.

The pose of mobile robot in the global coordinate frame $[O, X, Y]$ and the pose vector in the surface is defined as:

$$q = (x, y, \theta)^T \quad \text{... (1)}$$

Where:

$q(t) \in \Re^{3\times1}$,

x and y are coordinates of point c and θ is the robotic orientation angle measured with respect to the X-axis. These three generalized coordinates can describe the configuration of the mobile robot.

It is assumed that the mobile robot wheels are ideally installed in such a way that they have ideal rolling without skidding [14], as shown in equation (2):

$$-\dot{x}(t)\sin\theta(t) + \dot{y}(t)\cos\theta(t) = 0 \quad \text{... (2)}$$

Therefore, The kinematics equations in the world frame can be represented as follows [15]:

$$\dot{x}(t) = V_l(t)\cos\theta(t) \quad \text{... (3)}$$

$$\dot{y}(t) = V_l(t)\sin\theta(t) \quad \text{... (4)}$$

$$\dot{\theta}(t) = V_w(t) \quad \text{... (5)}$$

Where: V_L and V_w, are the linear and angular velocities.

In the computer simulation, the currently form of the pose equations, as follows [15]:

$$x(k) = 0.5[V_R(k) + V_L(k)]\cos\theta(k)\Delta t + x(k-1) \quad \text{... (6)}$$

$$y(k) = 0.5[V_R(k) + V_L(k)]\sin\theta(k)\Delta t + y(k-1) \quad \text{... (7)}$$

$$\theta(k) = \frac{1}{L}[V_L(k) - V_R(k)]\Delta t + \theta(k-1) \quad \text{... (8)}$$

Where: $x(k), y(k), \theta(k)$ are the components of the pose at the k step of the movement and Δt is the sampling period between two sampling times.

To check controllability of the nonlinear MIMO kinematic mobile robotic system in equation (3, 4, 5), the accessibility rank condition is globally satisfied and is implied controllability.

The mobile robot kinematics can be described by the left and right velocities as follows [16]:

$$\begin{bmatrix} \dot{x}(t) \\ \dot{y}(t) \\ \dot{\theta}(t) \end{bmatrix} = \begin{bmatrix} 0.5\cos\theta(t) & 0.5\cos\theta(t) \\ 0.5\sin\theta(t) & 0.5\sin\theta(t) \\ 1/L & -1/L \end{bmatrix} \begin{bmatrix} V_R(t) \\ V_L(t) \end{bmatrix} \quad \text{... (9)}$$

By using Jacobi-Lie-Bracket of f and g to find $[f, g]$ [16].

$$[\dot{q}] = [f]V_R(t) + [g]V_L(t) \qquad \text{... (10)}$$

and f and g can be defined in two vectors with components as:

$$f = \begin{bmatrix} 0.5\cos\theta(t) \\ 0.5\sin\theta(t) \\ 1/L \end{bmatrix} \text{ and } g = \begin{bmatrix} 0.5\cos\theta(t) \\ 0.5\sin\theta(t) \\ -1/L \end{bmatrix} \qquad \text{... (11)}$$

$$[f,g] = \begin{bmatrix} [f,g]^1 \\ [f,g]^2 \\ [f,g]^3 \end{bmatrix} = \begin{bmatrix} \frac{-1}{L}\sin\theta(t) \\ \frac{1}{L}\cos\theta(t) \\ 0 \end{bmatrix} \qquad \text{... (12)}$$

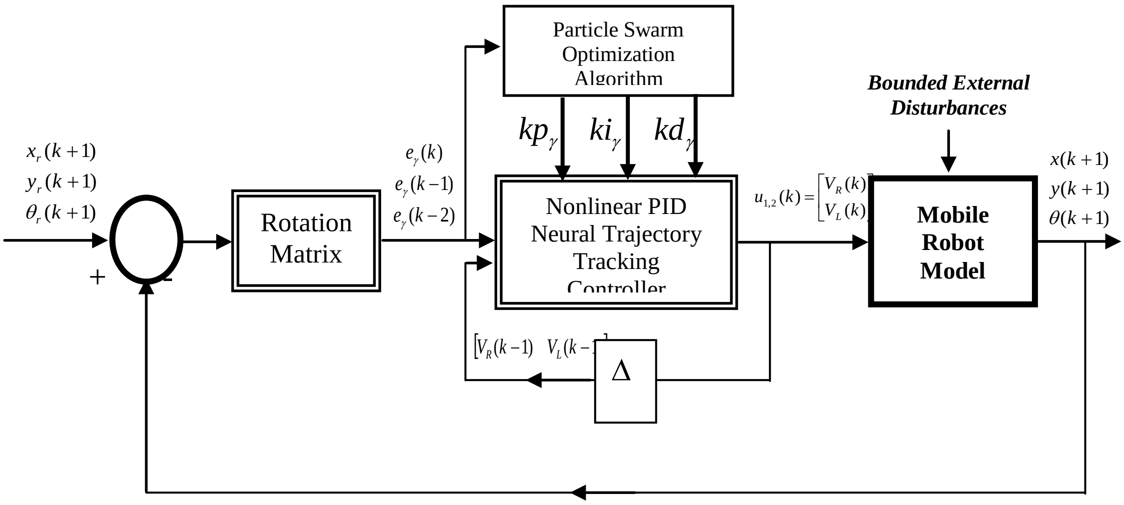
$$rank\{f, g, [f,g]\} = rank \begin{bmatrix} 0.5\cos\theta(t) & 0.5\cos\theta(t) & \frac{-1}{L}\sin\theta(t) \\ 0.5\sin\theta(t) & 0.5\sin\theta(t) & \frac{1}{L}\cos\theta(t) \\ 1/L & -1/L & 0 \end{bmatrix} \qquad \text{... (13)}$$

The determent of the matrix in equation (13) is equal to $(1/L^2) \neq 0$, then the full rank of matrix is equal to 3, therefore, the system in equation (3, 4 and 5) is controllable.

NONLINEAR PID NEURAL CONTROL METHODOLOGY

The approach to control the wheeled mobile robot depends on the available information of the unknown nonlinear system can be known by the input-output data and the control objectives. The optimization algorithm will generate the optimal parameters for the nonlinear PID neural controller in order to obtain best velocity control signal that will minimize the tracking error of the mobile robot in the presence of external disturbance. The proposed structure of the nonlinear PID neural controller can be given in the form of block diagram as shown in Figure (2).

The feedback PID neural controller is very important because it is necessary to stabilize the tracking error of the system when the output of the mobile robot is drifted from the desired point.



Δ is defined as a delay mapping.

Figure (2) The general proposed structure of nonlinear PID Neural trajectory tracking controller for mobile robot model.

The nonlinear PID neural controller for MIMO mobile robot system can be shown in Figure (3).

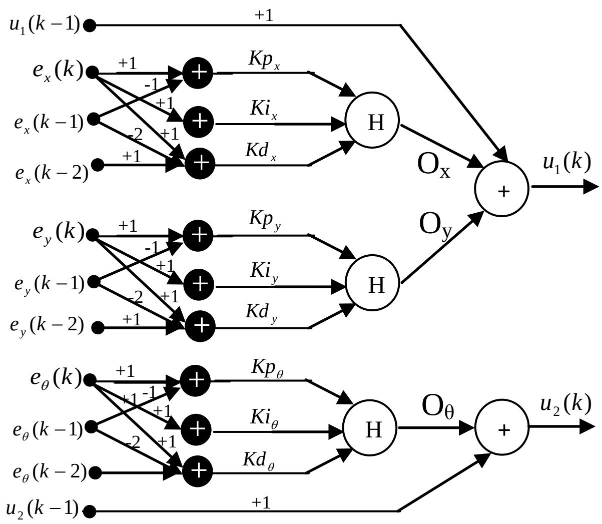


Figure (3) the nonlinear PID neural feedback controller structure.

It has the characteristics of control agility, strong adaptability, good dynamic characteristic and robustness because it is based on that of a conventional PID controller that consists of three terms: proportional, integral and derivative where the standard form of a PID controller is given in the s-domain as equation (14) [17].

$$Gc(s) = P + I + D = K_p + \frac{K_i}{s} + K_d s \qquad \dots (14)$$

Where K_p, K_i and K_d are called the proportional gain, the integral gain and the derivative gain respectively.

The proposed nonlinear PID neural controller scheme is based on the discrete-time PID as equation (15) [18].

$$u_{1,2}(k) = u_{1,2}(k-1) + Kp_\gamma[e_\gamma(k) - e_\gamma(k-1)] + Ki_\gamma e_\gamma(k) + Kd_\gamma[e_\gamma(k) - 2e_\gamma(k-1) + e_\gamma(k-2)] \quad \ldots (15)$$

Where $\gamma = x, y, \theta$.

Therefore, the tuning PID input vector consists of $e_\gamma(k)$, $e_\gamma(k-1)$, $e_\gamma(k-2)$ and $u_{1,2}(k-1)$, where $e_\gamma(k)$ and $u_{1,2}(k-1)$ denote the input error signals and the PID output signal respectively.

The proposed control law of the feedback right and left velocity (u_1 and u_2) respectively can be proposed as follows:

$$u_1(k) = u_1(k-1) + o_x + o_y \quad \ldots (16)$$

$$u_2(k) = u_2(k-1) + o_\theta \quad \ldots (17)$$

o_x, o_y and o_θ are the outputs of the neural networks that can be obtained from sigmoid function has nonlinear relationship as presented in the following function:

$$o_\gamma = \frac{2}{1 + e^{-net_\gamma}} - 1 \quad \ldots (18)$$

net_γ is calculated from this equation:

$$net_\gamma(k) = Kp_\gamma[e_\gamma(k) - e_\gamma(k-1)] + Ki_\gamma e_\gamma(k) + Kd_\gamma[e_\gamma(k) - 2e_\gamma(k-1) + e_\gamma(k-2)] \quad \ldots (19)$$

The control parameters Kp_γ, Ki_γ and Kd_γ of the nonlinear PID neural controller are adjusted using particle swarm optimization.

LEARNING PARTICLE SWARM OPTIMIZATION ALGORITHM

Particle Swarm optimization (PSO) is a kind of algorithm to search for the best solution by simulating the movement and flocking of birds. PSO algorithms use a population of individual (called particles) “flies” over the solution space in search for the optimal solution.

Each particle has its own position and velocity to move around the search space. The particles are evaluated using a fitness function to see how close they are to the optimal solution [19, 20 and 21].

The previous best value is called as *pbest*. Thus, *pbest* is related only to a particular particle. It also has another value called *gbest*, which is the best value of all the particles *pbest* in the swarm.

The nonlinear PID neural controller with nine weights parameters and the matrix is rewritten as an array to form a particle. Particles are then initialized randomly and updated afterwards according to equations (20, 21, 22, 23, 24 and 25) in order to tune the PID parameters:

$$\Delta Kp_{\gamma,m}^{k+1} = \Delta Kp_{\gamma,m}^{k} + c_1 r_1 (pbest_{\gamma,m}^{k} - Kp_{\gamma,m}^{k}) + c_2 r_2 (gbest^{k} - Kp_{\gamma,m}^{k}) \quad \ldots (20)$$

$$Kp_{\gamma,m}^{k+1} = Kp_{\gamma,m}^{k} + \Delta Kp_{\gamma,m}^{k+1} \quad \ldots (21)$$

$$\Delta Ki_{\gamma,m}^{k+1} = \Delta Ki_{\gamma,m}^{k} + c_1 r_1 (pbest_{\gamma,m}^{k} - Ki_{\gamma,m}^{k}) + c_2 r_2 (gbest^{k} - Ki_{\gamma,m}^{k}) \quad \ldots (22)$$

$$Ki_{\gamma,m}^{k+1} = Ki_{\gamma,m}^{k} + \Delta Ki_{\gamma,m}^{k+1} \quad \dots (23)$$

$$\Delta Kd_{\gamma,m}^{k+1} = \Delta Kd_{\gamma,m}^{k} + c_1 r_1 (pbest_{\gamma,m}^{k} - Kd_{\gamma,m}^{k}) + c_2 r_2 (gbest^{k} - Kd_{\gamma,m}^{k}) \quad \dots (24)$$

$$Kd_{\gamma,m}^{k+1} = Kd_{\gamma,m}^{k} + \Delta Kd_{\gamma,m}^{k+1} \quad \dots (25)$$

$m = 1,2,3,.....pop$

Where

pop is number of particles.

$K_{\gamma,m}^{k}$ is the weight of particle *m* at *k* iteration.

c_1 and c_2 are the acceleration constants with positive values equal to 2.

r_1 and r_2 are random numbers between 0 and 1.

$pbest_{\gamma,m}$ is best previous weight of m^{th} particle.

$gbest$ is best particle among all the particle in the population.

The numbers of dimensions in particle swarm optimization are equal to nine because there are three nonlinear PID and each one has three parameters.

The mean square error function is chosen as criterion for estimating the model performance as equation (26):

$$E = \frac{1}{2}\sum_{j=1}^{pop}((x_r(k+1)^j - x(k+1)^j)^2 + (y_r(k+1)^j - y(k+1)^j)^2 + (\theta_r(k+1)^j - \theta(k+1)^j)^2) \quad \dots (26)$$

The steps of PSO for nonlinear PID neural controller can be described as follows:

- **Step1** Initial searching points $Kp_{\gamma}^{0}, Ki_{\gamma}^{0}, Kd_{\gamma}^{0}, \Delta Kp_{\gamma}^{0}, \Delta Ki_{\gamma}^{0}$ and ΔKd_{γ}^{0} of each particle are usually generated randomly within the allowable range. Note that the dimension of search space consists of all the parameters used in the nonlinear PID neural controller as shown in Figure (2). The current searching point is set to *pbest* for each particle. The best-evaluated value of *pbest* is set to *gbest* and the particle number with the best value is stored.
- **Step2** The objective function value is calculated for each particle by using equation (26). If the value is better than the current *pbest* of the particle, the *pbest* value is replaced by the current value. If the best value of *pbest* is better than the current *gbest*, *gbest* is replaced by the best value and the particle number with the best value is stored.
- **Step3** The current searching point of each particle is update using equations (20, 21, 22, 23, 24 and 25).
- **Step4** If the current iteration number reaches the predetermined maximum iteration number, then exit. Otherwise, return to step 2.

SIMULATION RESULTS

The proposed controller is verified by means of computer simulation using MATLAB package. The kinematic model of the nonholonomic mobile robot described in section 2 is used. The simulation is carried out off-line by tracking a desired position (x, y) and orientation angle (θ) with a lemniscates and square trajectories in the tracking control of the robot. The parameter values of the robot model are taken from [22 and 23]: M=0.65kg, I=0.36kg.m^2, L=0.105 m, r=0.033 m and sampling time is equal to 0.5 second.

The proposed nonlinear PID neural controller scheme as in Figure (2) is applied to the mobile robot model and it is used the proposed learning algorithm steps of PSO for tuning the nonlinear PID controller's parameters. The fist stage of operation is to set the following parameters of the PSO algorithm:

Population of particle is equal to 30 and number of iteration is equal to 100. Number of weight in each particle is 9 because there are nine parameters of PID. The acceleration constants c_1 and c_2 are equal to 2.
r_1 and r_2 are random values between 0 and 1.

CASE STUDY

The desired lemniscates trajectory which has explicitly continuous gradient with rotation radius changes, this trajectory can be described by the following:

$$x_r(t) = -2.5 \times \sin(\frac{2\pi t}{30}) \quad \text{... (27)}$$

$$y_r(t) = 2 \times \sin(\frac{2\pi t}{20}) \quad \text{... (28)}$$

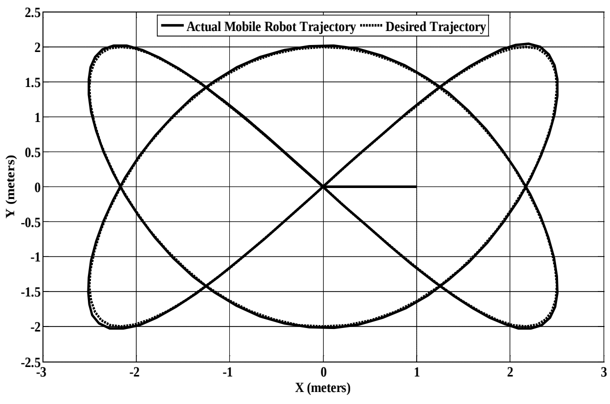
$$\theta_r(t) = 2\tan^{-1}(\frac{\Delta y_r(t)}{\sqrt{(\Delta x_r(t))^2 - (\Delta y_r(t))^2} + \Delta x_r(t)}) \quad \text{... (29)}$$

For simulation purposes, the desired trajectory is chosen as described in equations (27 and 28) while the desired orientation angle is taken as expressed in equation (29). The robot model starts from the initial posture $q(0) = [1, 0, \pi/2]$ as its initial conditions.

A disturbance term $\overline{\tau d} = [0.01\sin(2t) \quad 0.01\sin(2t)]^T$ [12 and 13] is added to the robot system as unmodelled kinematics disturbances in order to prove the adaptation and robustness ability of the proposed controller. The mobile robot trajectory tracking obtained by the proposed controller is shown in Figure (4).

These Figures demonstrate excellent position and orientation tracking performance in spite of the existence of bounded disturbances the adaptive learning and robustness of nonlinear PID neural controller show small effect of these disturbances. The simulation results demonstrated the effectiveness of the proposed controller by showing its ability to generate small smooth values of the control input velocities for right and left wheels without sharp spikes.

(a)



(b)

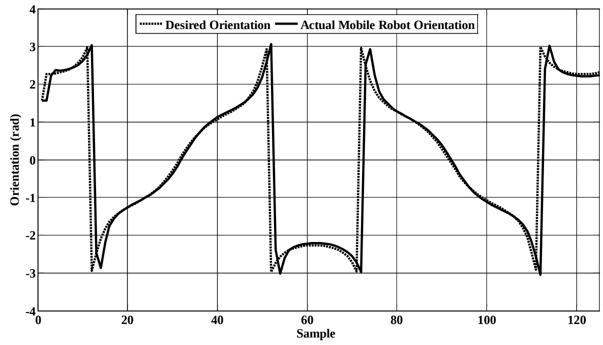


Figure (4) Simulation results (a) desired trajectory and actual mobile robot trajectory; (b) desired orientation and actual mobile robot orientation.

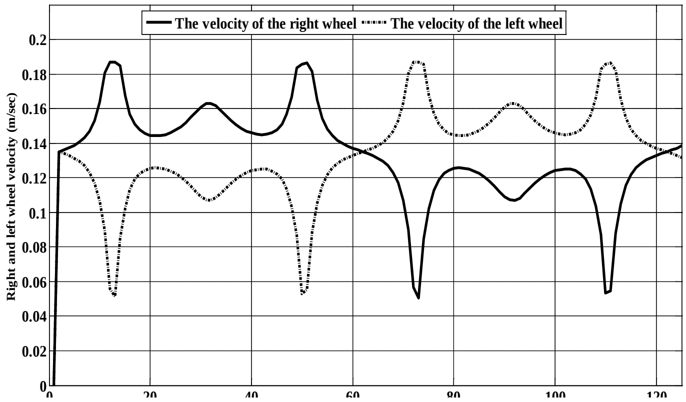


Figure (5) the right and left wheel action.

The actions described in Fig. 5 shows that smaller power is required to drive the DC motors of the mobile robot model.

The mean linear velocity of the mobile robot is equal to 0.135 m/sec, and the maximum peak of the angular velocity is equal to ± 0.45 rad/sec can be shown in Fig. 6.

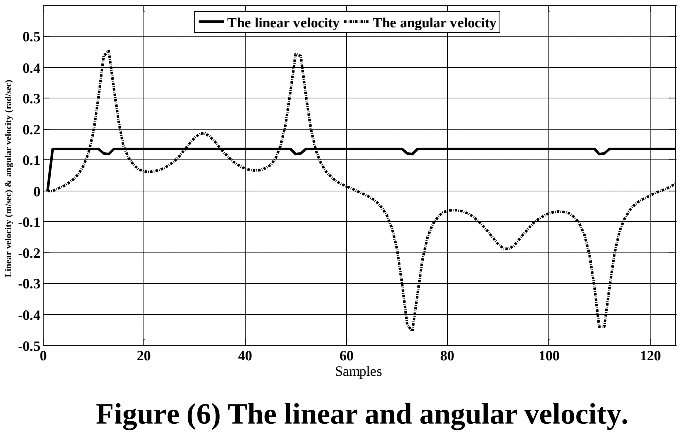


Figure (6) The linear and angular velocity.

The optimised-auto-tuning based on particle swarm optimization is used for tuning the parameters of the PID neural controller (k_x, k_y, k_θ) which has demonstrated, as shown in Table (1).

Table (1) Parameters of the PID controller.

kp_x	ki_x	kd_x	kp_y	ki_y	kd_y	kp_θ	ki_θ	kd_θ
0.47	0.09	-0.30	-0.26	0.12	0.14	-0.19	0.18	-0.12

It is used Mean Square Error (MSE) as the performance index in the control methodology that is clear by showing the convergence of the pose trajectory and orientation errors for the robot model motion at 100 iteration, as shown in Figure (7).

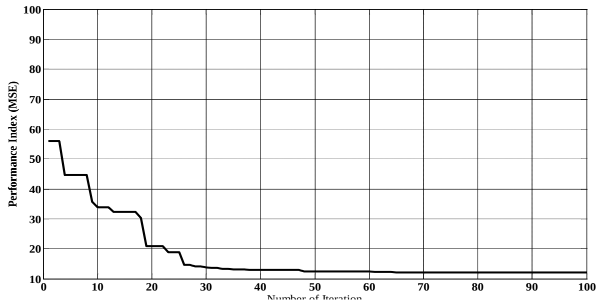


Figure (7) The performance index (MSE).

The effectiveness of the proposed nonlinear PID neural control algorithm is clear by showing the convergence of the pose trajectory and orientation errors for the robot model motion, as shown in Figure (8).

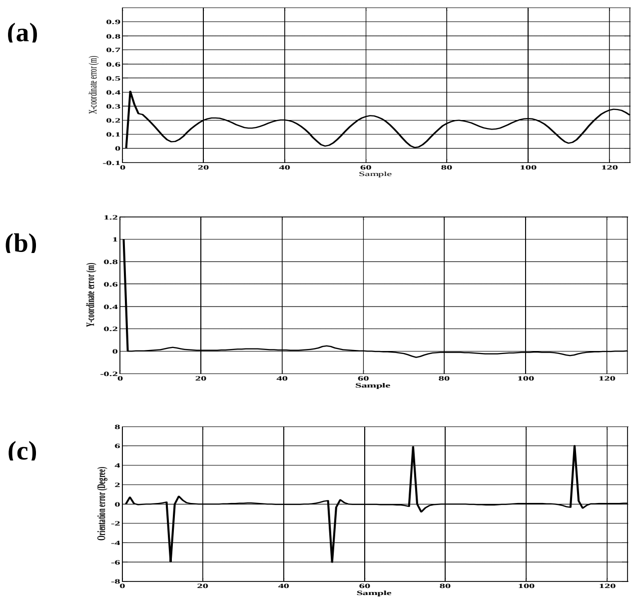


Figure (8): Position tracking error (a) in X- coordinate; (b) in Y-coordinate; (c) Orientation tracking error.

CONCLUSIONS

The nonlinear PID neural network controller with particle swarm optimization algorithm technique for MIMO nonholonomic wheeled mobile robot motion model has been presented in this paper. The state outputs of the mobile model are position and orientation and they are followed the desired inputs because there are two control actions right and left velocities that are generated from the proposed controller with PSO algorithm.

PSO was used off-line to tune the parameters of the nonlinear PID neural controller and find the best value of the control action with minimum time and more stability with no oscillation in the output response.

Simulation results show evidently that the proposed controller model has the capability of generating smooth and suitable velocity (V_R and V_L) commands and did not exceed 0.2 m/sec, without sharp spikes.

The proposed controller has demonstrated the capability of tracking continuous gradients desired trajectory and effectively minimization the tracking errors of the nonholonomic wheeled mobile robot model with $MSE(ex, ey, e\theta)$ is equal to $(0.0415, 0.0033, 1.28)$ respectively, especially with regards to the external disturbances attenuation problem.

REFERENCES

[1]. Yasuda and H. Takai, G. Sensor-based path planning and intelligent steering control of nonholonomic mobile robots, The 27th Annual Conference of the IEEE Industrial Electronics Society (IECON01), 2001, pp. 317-322.

[2].R-J Wai and C-M Liu, Design of dynamic petri recurrent fuzzy neural network and its application to path-tracking control of nonholonomic mobile robot, IEEE Transactions on Industrial Electronics. Vol. 56, No. 7, (2009), pp. 2667-2683.

[3]. Yaonan, W. Y. Yimin, Y. Xiaofang, Z. Yuanli, Y. Feng and T. Lei, Autonomous mobile robot navigation system designed in dynamic environment based on transferable belief model". Measurement. Vol. 44, (2011), pp.1389-1405.

[4]. Toibero, J. M. F. Roberti, R. Carlli and P. Fiorini, Switching control approach for stable navigation of mobile robots in unknown environment, Robotics and Computer-Integrated Manufacturing. Vol. 27, (2011), pp.558-568.

[5]. Kokosy, A. F-O. Defaux and W. Perruquetti, Autonomous navigation of a nonholonomic mobile robot in a complex environment, Proceedings of the 2008 IEEE International Workshop on Safety, Security and Rescue Robotics, Sendai, Japan, October 2008, pp. 102-108.

[6]. Hague,T. N. Tillet, Navigation and control of an autonomous horticultural robot, Mechatronics. Vol. 6, No 2, (1996), pp.165-180.

[7]. Choi and S. Ryew, H. Robotic system with active steering capability for internal inspection of urban gas pipelines, Mechatronics. Vol. 12, No. 5, (2002), pp.713-736.

[8]. Das and N. Kar, T. Design and implementation of an adaptive fuzzy logic-based controller for wheeled mobile robots, IEEE Transactions on Control System Technology. Vol.14, No.3, (2006), pp. 501-510.

[9]. Klancar and I. Skrjanc, G. Tracking error model-based predictive control for mobile robots in real time, Robotics and Autonomous Systems. Vol. 55, (2007), pp.460-469.

[10]. Park, B. S. S. J. Yoo, J. B. Park and Y. H. Choi, A simple adaptive control approach for trajectory tracking of electrically driven nonholonomic mobile robots, IEEE Transactions on Control Systems Technology. Vol.18, No. 5, (2010), pp.1199-1206.

[11].K-H. Su, Y-Y. Chen and S-F. Su, Design of neural-fuzzy-based controller for two autonomously driven wheeled robot, Neurocompting. Vol. 73 (2010), pp.2478-2488.

[12]. J. Ye, Adaptive control of nonlinear PID-based analogue neural network for a nonholonomic mobile robot. Neurocompting. Vol. 71 (2008), pp.1561-1565.

[13]. Al-Araji, A. S. M. F. Abbod and H. S. Al-Raweshidy, Design of a neural predictive controller for nonholonomic mobile robot based on posture identifier, Proceedings of the IASTED International Conference Intelligent Systems and Control (ISC 2011). Cambridge, United Kingdom, July 11 - 13, 2011, pp. 198-207.

[14]. Siegwart and I. R. Nourbakhah, R. Introduction to autonomous mobile robots, MIT Press, 2004.

[15]. Han, S. B. Choi and J. Lee, A Precise Curved Motion Planning for a Differential Driving Mobile Robot. Mechatronics, Vol. 18, (2008), pp. 486-494.

[16]. Bloch, A. M. Nonholonomic Mechanics and Control. New York: Springer-Verlag, 2003.

[17]. Zhong,Q. Robust Control of Time-delay Systems, Springer – Verlag London Limited 2006

[18]. Omatu,S. M. Khalid, and R. Yusof, Neuro-Control and its Applications. London: Springer-Velag, 1995.

[19]. Derrac, J. S. Garc, D. Molina and F. Herrera ,a Practical Tutorial on the use of Nonparametric Statistical Tests as a Methodology for Comparing Evolutionary and Swarm Intelligence Algorithms, Journal of Swarm and Evolutionary Computation, Vol.1, (2011), pp. 3–18.

[20]. Zhou, J. Z. Duan, Y. Li, J. Deng and D. Yu, PSO-based Neural Network Optimization and Its Utilization in a Boring Machine, Journal of Materials Processing Technology, Vol.178, (2006), pp.19–23.

[21]. Lee,Y. S. S.M. Shamsuddin, and H.N. Hamed, Bounded PSO Vmax Function in Neural Network Learning, Proceedings of the Eighth International Conference on Intelligent Systems Design and Applications, 2008, pp.474-479.

[22].Internet website http://www.parallax.com. Robotics with the Boe-Bot text manual v3.0. Accessed April 2013.

[23].A. Al-Araji, M. Abbod and H. Al-Raweshidy, Applying Posture Identifier in Designing an Adaptive Nonlinear Predictive Controller for Nonholonomic Mobile Robot. Neurocomputing. Vol. 99, (2013), pp. 543-554.