Designing a Nonlinear PID Neural Controller of Differential Braking System for Vehicle Model Based on Particle Swarm Optimization

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

Received on: 24/3/2013 & Accepted on: 3/10/2013

ABSTRACT

This paper presents a nonlinear PID neural controller for the 2-DOF vehicle model in order to improve stability and performances of vehicle lateral dynamics by achieving required yaw rate and reducing lateral velocity in a short period of time to prevent vehicle from sliding out the curvature. The scheme of the discrete-time PID control structure is based on neural network and tuned the parameters of the nonlinear PID neural controller by using a particle swarm optimization PSO technique as a simple and fast training algorithm. The differential braking system and front wheel steering angle are the outputs of the nonlinear PID neural controller that has automatically controlled the vehicle lateral motion when the vehicle rotates the curvatures. Simulation results show the effectiveness of the proposed control algorithm in terms of the best transient state outputs of the system and minimum tracking errors as well as smoothness control signals obtained with bounded external disturbances.

Keywords: Particle Swarm Optimization, Neural Network, PID Controller, Vehicle Lateral Dynamics.

تصميم مسيطر عصبي (PID) لأخطي لنظام الكبح ألفرقي لمركبة أساسه أمثلية حشد الجسيمات

الخلاصة

أن هذا البحث يقدم مسيطر عصبي (PID) لأخطي لنموذج مركبة ثنائية الأبعاد لكي يحسن استقرارية و أداء الديناميكي للمركبة من خلال تحقيق معدل الدوران المطلوب و تقليل السرعة الجانبية في اقل وقت ممكن لمنع المركبة من الانزلاق خارج المنعطف.

أن هيكلية المسيطر العصبي (PID) مبني على أساس الشبكة العصبية و تنغيم عناصر المسيطر (PID) تتم من خلال تقنية خوارزمية أمثلية حشد الجسيمات لأنها خوارزمية سهله و سريعة التعلم.

أن نظام الكبح ألفرقي وزاوية توجيه العجلات الأمامية هي أخراج المسيطر العصبي (PID) اللأخطي الذي يسيطر بصورة تلقائية على الحركة الجانبية للمركبة عندما تدور حول المنعطف.

https://doi.org/10.30684/etj.32.1A.15
2412-0758/University of Technology-Iraq, Baghdad, Iraq

من خلال نتائج المحاكات نلاحظ فعالية خوارزمية المسيطر المقترح من حيث أفضل حالة عابرة للإخراج النظام واقل تتابع خطأ و أيضا الحصول على نعومة في أشارات السيطرة بالرغم من وجود الاضطرابات الخارجية المحددة.

INTRODUCTION

Recently, the automotive electronic technology has led to the rapid development of modem automotive system technology such as Vehicle Dynamics Control (VDC), Anti-lock Brake System (ABS), Acceleration Slip Regulation (ASR) System, Electronic Stabilization Program (ESP) and Automatic Guidance Control (AGC) [1 and 2] where they are played an important role on improving the vehicle active safety and stability in order to prevent the wheel from slipping on the braking or accelerating as well as to avoid vehicle from sliding out the curvature.

The purpose of these systems is to actively control the vehicle under emergency situations where the vehicle is at the physical limit of adhesion between the tires and the road. These emergency situations are those that the normal driver usually cannot handle, and often loses control of the vehicle [3].

A system, which automatically intervenes in such situations, allows the driver to keep control of the vehicle and enhances the chance of avoiding an accident. In fact, there are three points that an active safety system must be addressed [4].

1) A vehicle must provide good controllability by responding quickly and accurately (i.e. with the right amount of change) to the driver's operational inputs.

2) A vehicle must provide good stability, with little change in behavior in relation to changes in driving conditions.

3) There must be an effective control loop between the driver and the vehicle for conveying operational inputs and the vehicle response in order to ensure that the driver can easily recognize present operating conditions and also predict vehicle behavior.

There are various ways to address these control issues such as in [5] it is used independent front and rear wheel drive to control the vehicle to track a desired lateral velocity by using fuzzy logic controller with variable gain structures. In [6], it is an analysis of vehicle lateral dynamics and the basic look-ahead control law is suggested for controller design to investigate the characteristics of vehicle dynamics and the performance requirements of steering controllers in vehicle-following collisions.

Also a robust PI controller is presented in [7] for the tracking of predefined vehicle lateral dynamics and using a full nonlinear vehicle simulator and four wheels steering control signal with the presence of a wide variety of disturbances. The model control is optimized based on the model predictive control theory as

explained in [8] through adjusting the tire parameters on-line, the vehicle stability control performance is well, but the robustness was poor. In addition to that, the vehicle stability controller is designed based on the Linear Quadratic Regulator (LQR) and fuzzy control theory as proposed in [1] where the yaw moment is produced by differential braking and the target of control vehicle stability is achieved. And in [9] it is used neural controller based genetic algorithm to control the vehicle to track a desired lateral velocity from front and rear steering angles.

A genetic neural fuzzy antilock brake system ABS controller is applied that consists of a non-derivative neural optimizer and fuzzy-logic components (FLC) as presented by Y.Z. [10]. It is used (ABS) senses when the wheel lockup is to occur, releases the brakes momentarily, and then reapplies the brakes when the wheel spins up again.

The remainder of this paper is organized as follows: Section two is a description of the 2DOF vehicle mathematical model. In section three, the proposed of nonlinear PID neural network controller approach and PSO tuning algorithm are derived. Simulation results of the proposed robust PID neural control algorithm are presented in section four and the conclusions are drawn in section five.

Vehicle Dynamics Model 2-DOF

There are many variables that effects on the dynamics of vehicle lateral motion such as vehicle speed, vehicle mass and tires state on road. The independent control of lateral and yaw motion requires at least one additional control input, which is independent of the front steering angle. There are three possible solutions for these inputs as follows: four wheel steering system; braking forces; and torque driving wheel [5 and 9].

In this paper, the focus is on the vehicle yaw rate and lateral velocity as the desired and the differential braking and front steering angle are the control action variables.

Figure (1) shows the two DOF vehicle model and it is widely used for lateral control design and has been shown to provide accurate response characteristics compared to more complex models for conditions up to 0.3g lateral acceleration [6, 7, 11, 12 and 13].

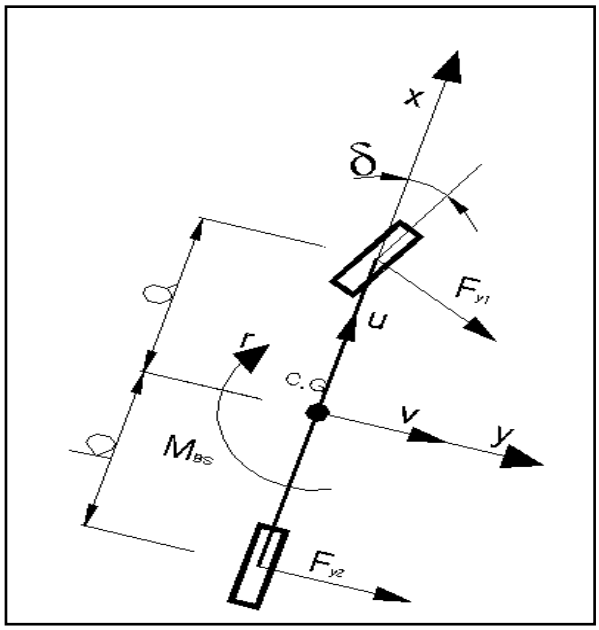


Figure (1): Two DOF Vehicle Model.

The linear dynamics model of vehicle lateral motion [13 and 14] with interaction in multi-input multi-output system are expressed as the state space equations as follows:

$$\begin{bmatrix} \dot{V} \\ r \end{bmatrix} = \begin{bmatrix} -\frac{C_f + C_r}{MU} & -\frac{C_f a - C_r b}{MU} - MU \\ -\frac{C_f a - C_r b}{IU} & -\frac{C_f a^2 + C_r b^2}{IU} \end{bmatrix} \begin{bmatrix} V \\ r \end{bmatrix} + \begin{bmatrix} \frac{C_f}{M} & 0 \\ \frac{aC_f}{I} & \frac{T}{2I} \end{bmatrix} \begin{bmatrix} \delta_f \\ F_{BS} \end{bmatrix} \quad \ldots(1)$$

$$\begin{bmatrix} y1 \\ y2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} V \\ r \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta_f \\ F_{BS} \end{bmatrix} \quad \ldots (2)$$

where y_1=V and y_2= r

Equation (1) represents linear mathematical model of vehicle lateral motion with interaction in multiple-input-multiple-output system (see appendix 1) where V is lateral velocity, r is yaw rate and both are system variable states. δ_f is the front steering angle and F_{Bs} is brake steer force and they are inputs to the system.

where the brake steering force can be described as in equation (1) from figure (1) as follows [13 and 14]:

$$M_{BS} = \frac{T}{2}(F_{XR} - F_{XL}) \quad \ldots(3)$$

$$F_{BS} = F_{XR} - F_{XL} \quad \ldots (4)$$

where:

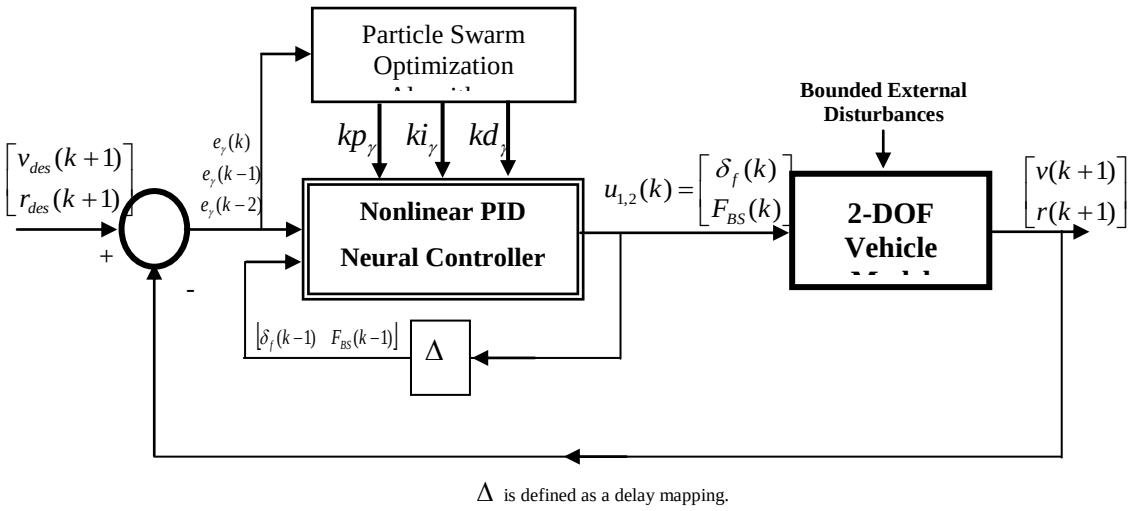
M_{BS} is brake steer moment.

F_{XR} and F_{XL} are front and rear longitudinal tire forces.

Nonlinear PID Neural Controller

The control of linear MIMO dynamics system is considered in this section. The approach used to control on the lateral velocity and yaw rate of the system that depends on the information available about the system model and the control objectives. The feedback nonlinear PID neural controller is necessary to stabilize the tracking error dynamics of the system when the output variable states of the system are drifted from the desired inputs by using two control signals front steering angle and brake steering force.

The general structure of the nonlinear PID neural controller type can be given in the form of the block diagram shown in figure (2).



Δ is defined as a delay mapping.

Figure (2): The general proposed structure of nonlinear PID Neural feedback controller for 2DOF vehicle model.

The nonlinear PID neural controller for MIMO system can be shown in figure (3). 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. The standard form of a PID controller is given in the s-domain as equation (5) [15].

$$Gc(s) = P + I + D = K_p + \frac{K_i}{s} + K_d s \quad \ldots(5)$$

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 like neural network PID controller structure as the discrete-time equation (6) [16].

$$u_\gamma(k) = u_\gamma(k-1) + Kp_\gamma[e_\gamma(k) - e_\gamma(k-1)] + Ki_\gamma e_\gamma(k) + Kd_\gamma[e_\gamma(k) - 2e_\gamma(k-1) + e_\gamma(k-2)] \quad \ldots (6)$$

where $\gamma = 1,2.$

Therefore, the tuning PID input vector consists of $e_\gamma(k)$, $e_\gamma(k-1), e_\gamma(k-2)$ and $u_\gamma(k-1)$, where $e_\gamma(k)$ and $u_\gamma(k-1)$ denote the input error signals and the PID output respectively. Particle swarm optimization algorithm technique is used to adjust the parameters of the nonlinear PID neural controller.

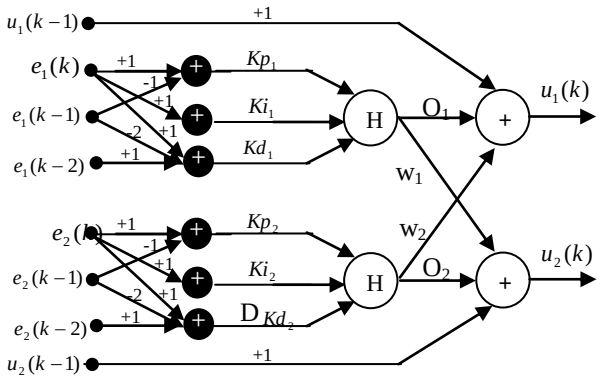


Figure (3): The nonlinear PID neural feedback controller structure.

The proposed control law of the feedback front steering angle and brake steer force (u_1 and u_2) respectively can be proposed as follows:

$$u_1(k) = u_1(k-1) + o_1 + o_2 w_2 \quad \ldots(7)$$

$$u_2(k) = u_2(k-1) + o_2 + o_1 w_1 \quad \ldots(8)$$

Assume w_1 and w_2 are constant positive weights. o_1 and o_2 are the outputs of the neural networks that can be obtained from sigmoid function has nonlinear relationship as presented in the following function:

$$o_1 = \frac{2}{1+e^{-net_1}} - 1 \quad \ldots(9)$$

$$o_2 = \frac{2}{1+e^{-net_2}} - 1 \quad \ldots(10)$$

net_1 and net_2 are calculated from these equations

$$net_1(k) = Kp_1[e_1(k) - e_1(k-1)] + Ki_1 e_1(k) + Kd_1[e_1(k) - 2e_1(k-1) + e_1(k-2)] \quad \ldots(11)$$

$$net_2(k) = Kp_2[e_2(k) - e_2(k-1)] + Ki_2 e_2(k) + Kd_2[e_2(k) - 2e_2(k-1) + e_2(k-2)] \quad \ldots(12)$$

The control parameters $Kp_{1,2}, Ki_{1,2}$ and $Kd_{1,2}$ of the nonlinear PID neural controller are adjusted using the particle swarm optimization techniques. Scaling functions have to be added at the neural network terminals to convert the scaled values to actual values and vice versa; therefore, the final feedback control actions $\delta_f(k)$ & $F_{BS}(k)$ for next sample can be calculated as the proposed law with scale factors S_δ for the front steering angle and S_F for the brake steer force.

$$\delta_f(k) = u_1(k) \times S_\delta \quad \ldots(13)$$

$$F_{BS}(k) = u_2(k) \times S_F \quad \ldots(14)$$

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 [17, 18 and 19].

$$V_{r,m}^{k+1} = V_{r,m}^{k} + c_1 r_1 (pbest_{r,m}^{k} - y_{r,m}^{k}) + c_2 r_2 (gbest_{m}^{k} - y_{r,m}^{k}) \quad \dots (15)$$

$$y_{r,m}^{k+1} = y_{r,m}^{k} + V_{r,m}^{k+1} \quad \dots (16)$$

$V_{r,m}^{k}$ is the velocity of the r^{th} particle at k iteration.

$y_{r,m}^{k}$ is the position of the r^{th} particle at k iteration.

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 six 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 (17, 18, 19, 20, 21 and 22) in order to tune the PID parameters:

$$\Delta Kp_{\gamma,m}^{k+1} = \Delta Kp_{\gamma,m}^{k} + c_1 r_1 (pbest_{\gamma,m}^{k} - Kp_{\gamma,m}^{k}) + c_2 r_2 (gbest^{k} - Kp_{\gamma,m}^{k}) \quad \dots (17)$$

$$Kp_{\gamma,m}^{k+1} = Kp_{\gamma,m}^{k} + \Delta Kp_{\gamma,m}^{k+1} \quad \dots (18)$$

$$\Delta Ki_{\gamma,m}^{k+1} = \Delta Ki_{\gamma,m}^{k} + c_1 r_1 (pbest_{\gamma,m}^{k} - Ki_{\gamma,m}^{k}) + c_2 r_2 (gbest^{k} - Ki_{\gamma,m}^{k}) \quad \dots (19)$$

$$Ki_{\gamma,m}^{k+1} = Ki_{\gamma,m}^{k} + \Delta Ki_{\gamma,m}^{k+1} \quad \dots (20)$$

$$\Delta Kd_{\gamma,m}^{k+1} = \Delta Kd_{\gamma,m}^{k} + c_1 r_1 (pbest_{\gamma,m}^{k} - Kd_{\gamma,m}^{k}) + c_2 r_2 (gbest^{k} - Kd_{\gamma,m}^{k}) \quad \dots (21)$$

$$Kd_{\gamma,m}^{k+1} = Kd_{\gamma,m}^{k} + \Delta Kd_{\gamma,m}^{k+1} \quad \dots (22)$$

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

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 number of dimension in particle swarm optimization is equal to six because there are two nonlinear PID and each one has three parameters.

The mean square error function is chosen as criterion for estimating the model performance as equation (23) [16]:

$$E = \frac{1}{2}\sum_{j=1}^{pop} (V_{ref}(k+1)^{j} - V_{out}(k+1)^{j})^2 + (r_{ref}(k+1)^{j} - r_{out}(k+1)^{j})^2 \quad \dots (23)$$

The steps of PSO for neural network like self-tuning PID 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 is consists of all the parameters used in the nonlinear PID neural controller as shown in figure (3). 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 (23). 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 by using equations (17, 18, 19, 20, 21 and 22).
- **Step4** 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/Simulink. The dynamic model of the 2-DOF vehicle described in section two is used. The vehicle parameters as given in appendix (2) is taken to clarify the features of the nonlinear PID neural controller explained in section three. Scaling function has to be added at the neural network terminals to convert the scaled values to actual values "where the differential braking range is $\pm 7000\,N$ and front steering angle is $\pm 0.1\,\text{rad}$" and vice versa in order to overcome a numerical problem that is involved within real values. Therefore the signals entering to the network have been normalized to lie within (-1 and +1).

The proposed control scheme is applied to the vehicle model and it is used the proposed learning algorithm steps of PSO for tuning nonlinear PID neural controller's parameters. The PSO algorithm is set to the following parameters:

Population of particle is equal to 30. Number of weight in each particle is 6 because there are two nonlinear PID neural controllers. Scaling functions have to be added at the neural network terminals to convert the scaled values to actual values and vice versa.

The performance of the proposed controller is evaluated using the closed-loop step lateral velocity and yaw rate responses for linear dynamic system. The desired lateral velocity must be zero to over come the vehicle may rotate around itself at high vehicle velocity. And desired yaw rate must be verified:

$$r_{ref} = \frac{U}{R} \tag{24}$$

where R is curvature radius.

After training with sampling time is equal to 0.1 sec, it can be observed that the actual yaw rate output of the system are following the desired trajectory and the yaw rate error can be shown as the figures (8-a and b) respectively. Figure (8-a) is the yaw rate response and its fast smoothness response without overshoot. Steady-state error is equal to zero while at the transient time, the error is approximately equal to 0.075rad/sec when the vehicle velocity is change as (35, 25, 15, 20, 30 and 40) m/sec with fixed curvature radius equal to 100m as shown in figure (8-b).

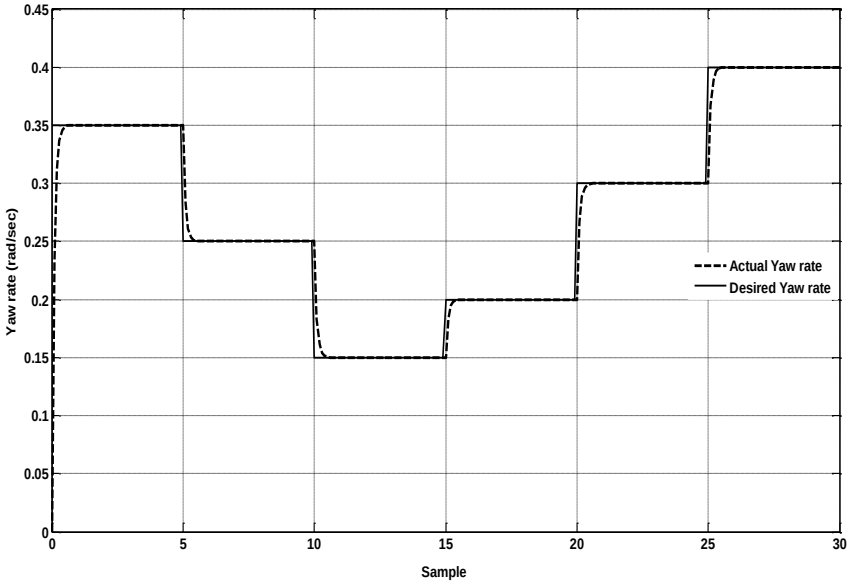


Figure (8-a): The yaw rate response

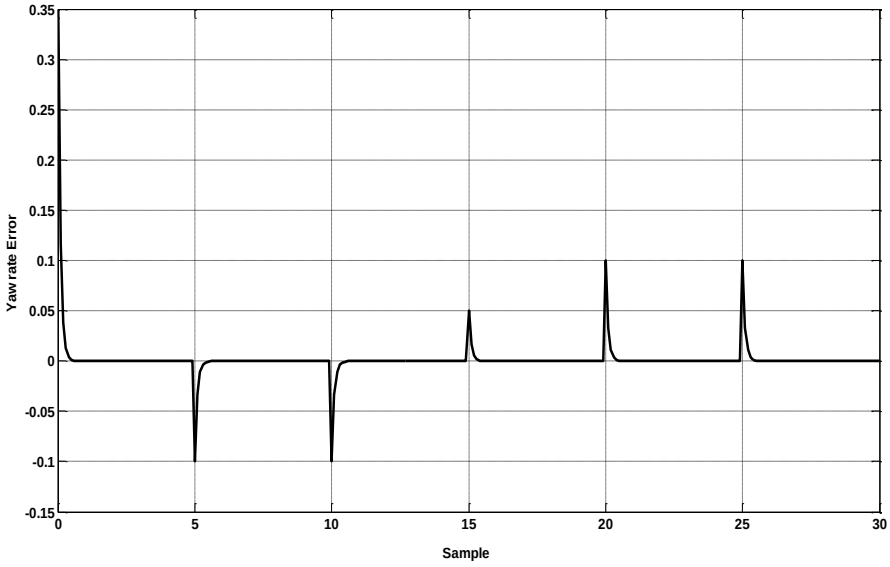


Figure (8-b): The yaw rate error response.

Figure (9-a) is the actual lateral velocity response of the system and its fast response with very small overshoot and equal to $\pm 1\times 10^{-7}$ m/sec.

The steady-state error is equal to zero while at the transient time, the error is approximately equal to $\pm 1\times 10^{-7}$ m/sec when the vehicle velocity is change as (35, 25,

15, 20, 30 and 40)m/sec with fixed curvature radius equal to 100m as shown in figure(9-b).

The robustness of nonlinear PID neural feedback control action will be kept the maximum amplitude of the lateral velocity in the transient response is equal to $\pm 1\times 10^{-7}$ m/sec when the velocity of the vehicle is changed and achievement the desired lateral velocity and yaw rate.

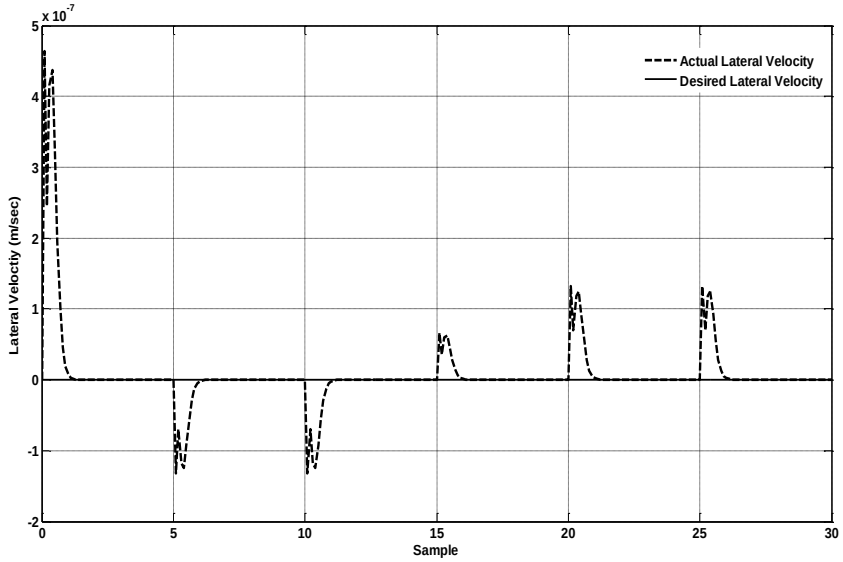


Figure (9-a): The lateral velocity response.

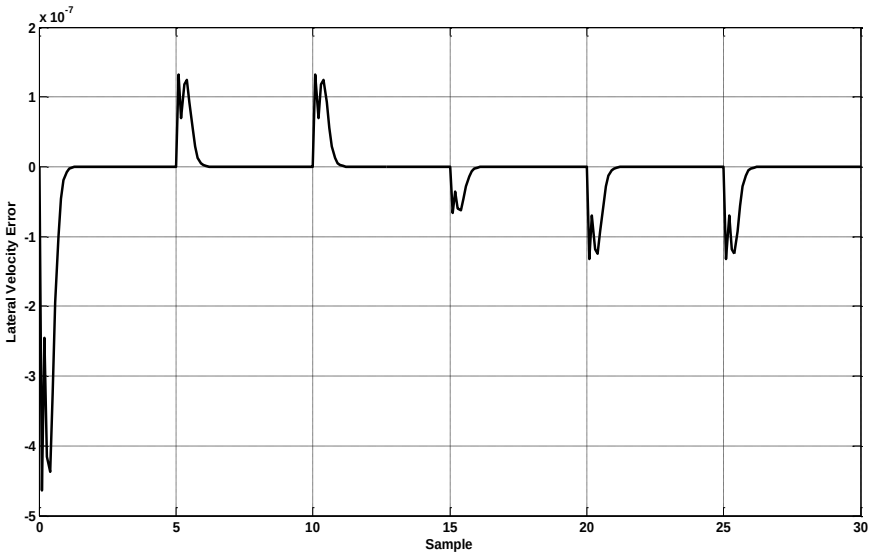


Figure (9-b): The lateral velocity error response.

The yaw rate control and the lateral velocity can be achieved by two feedback control actions brake steer force "differential braking" and front steering angle as shown in figures (10-a and b) respectively.

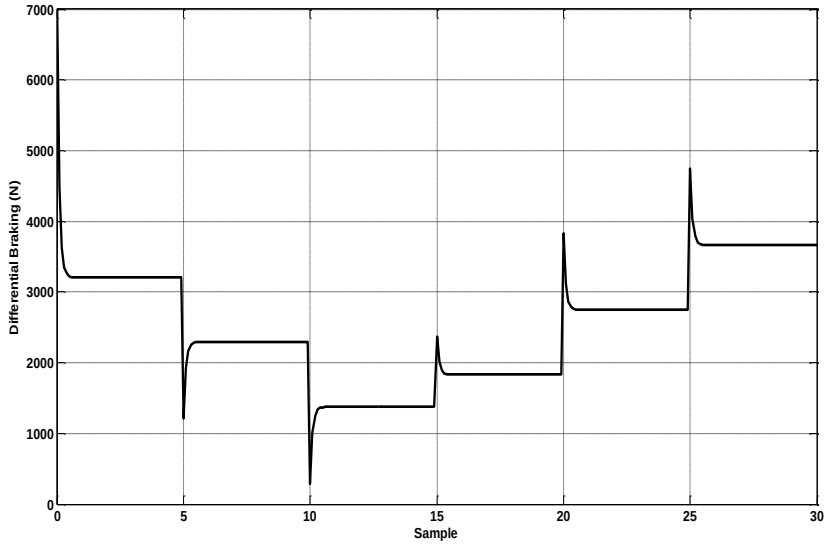


Figure (10-a): The differential brake response.

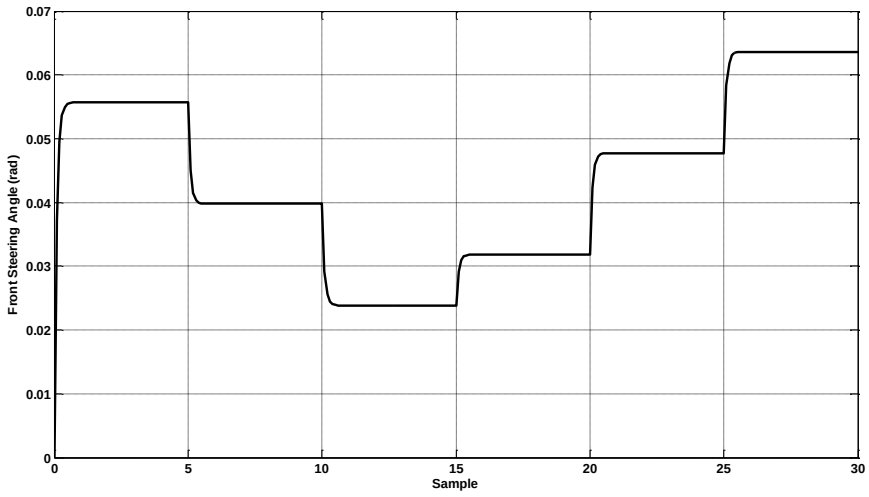


Figure (10-b): The front steer angle response.

The gains of the nonlinear PID neural controller as scale function are shown in figures (11-a and b).

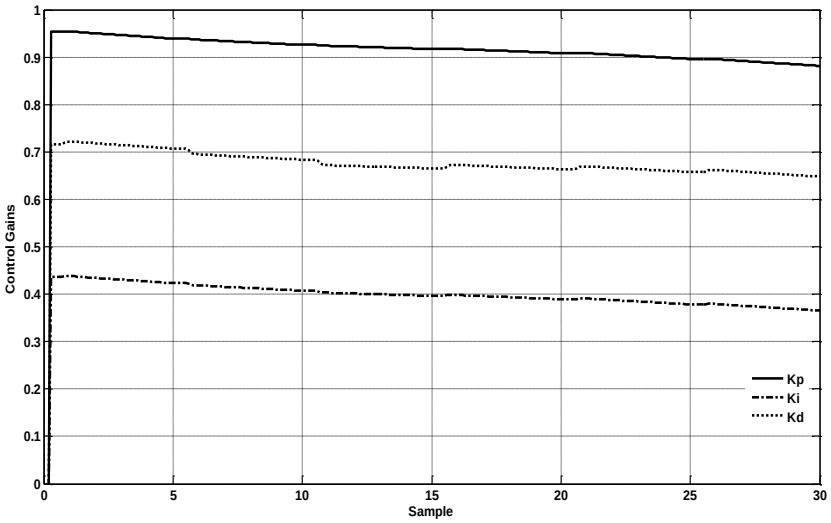


Figure (11-a): The PID control gain parameters kp_1, ki_1, kd_1

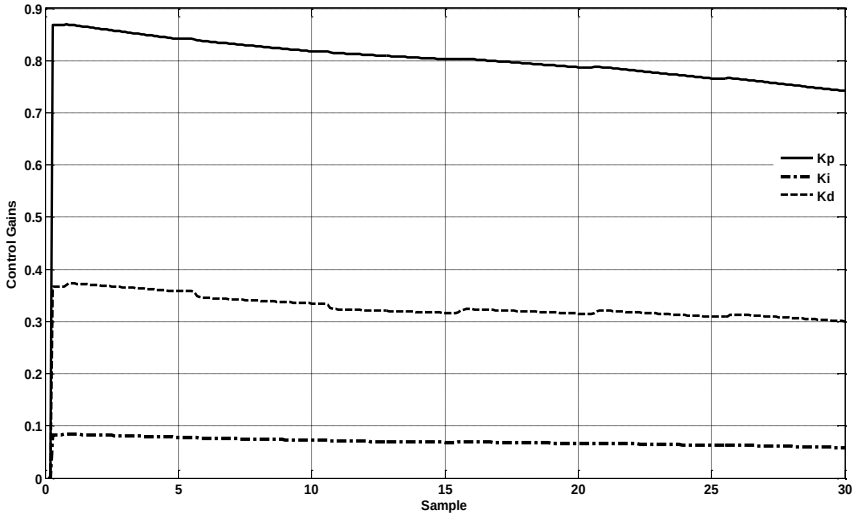


Figure (11-b): The PID control gain parameters kp_2, ki_2, kd_2

In order to prove the adaptation and robustness ability of the proposed controller, a disturbance term $\delta_{dis} = 2\times10^{-3}\sin(100t)$ is added to the vehicle model as unmodelled and dynamics disturbances then the response of the yaw rate of the vehicle is not

drifted from the desired and it has very small overshoot as shown in figure (12-a) and also the lateral velocity of the vehicle is very small oscillation magnitude $\pm 1\times 10^{-3}$ as shown in figure (12-b).

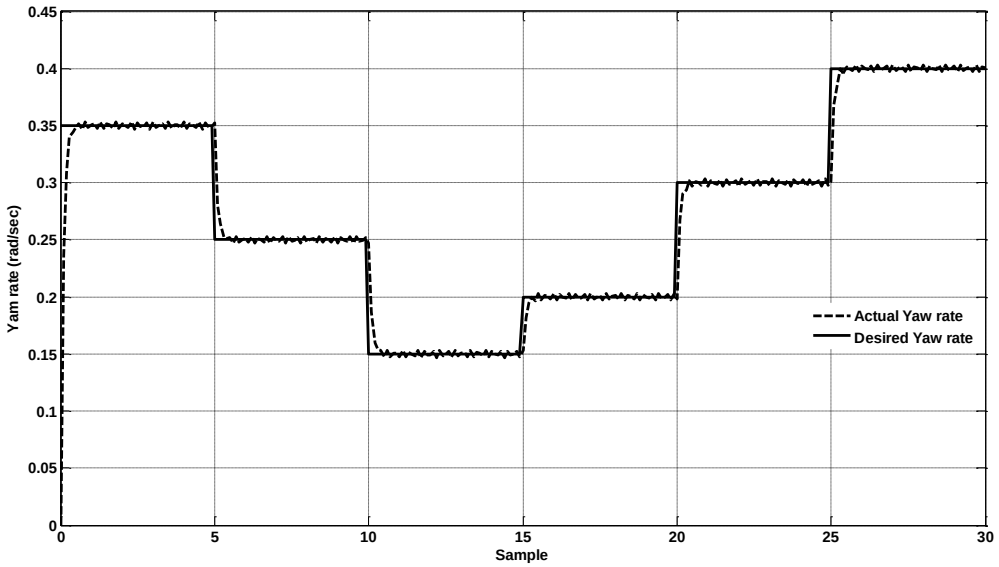


Figure (12-a): The yaw rate response.

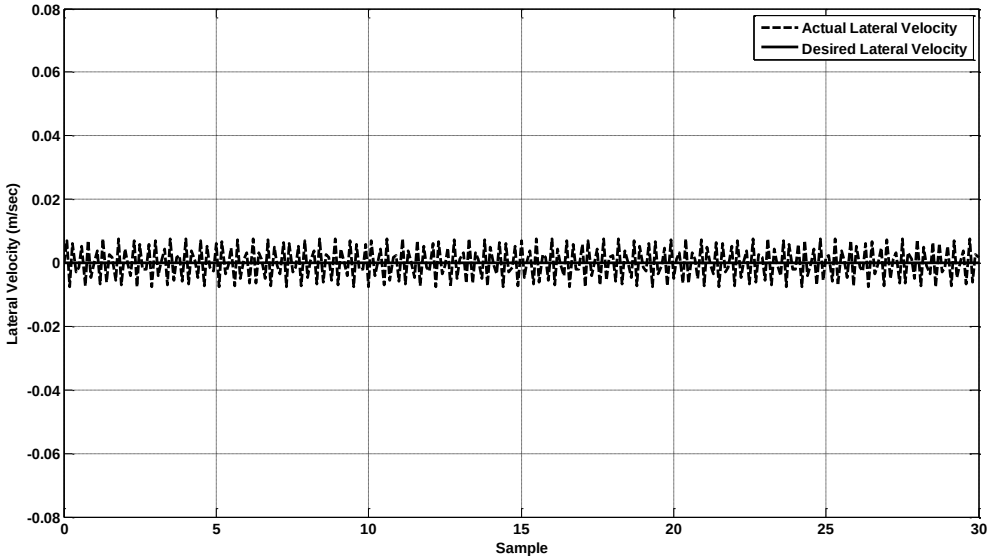


Figure (12-b): The lateral velocity response.

The differential braking and the front steering angle can be shown in figures (13-a and b) while the error between the desired yaw rate and actual output for disturbance case can be shown in figure (14-a and b).

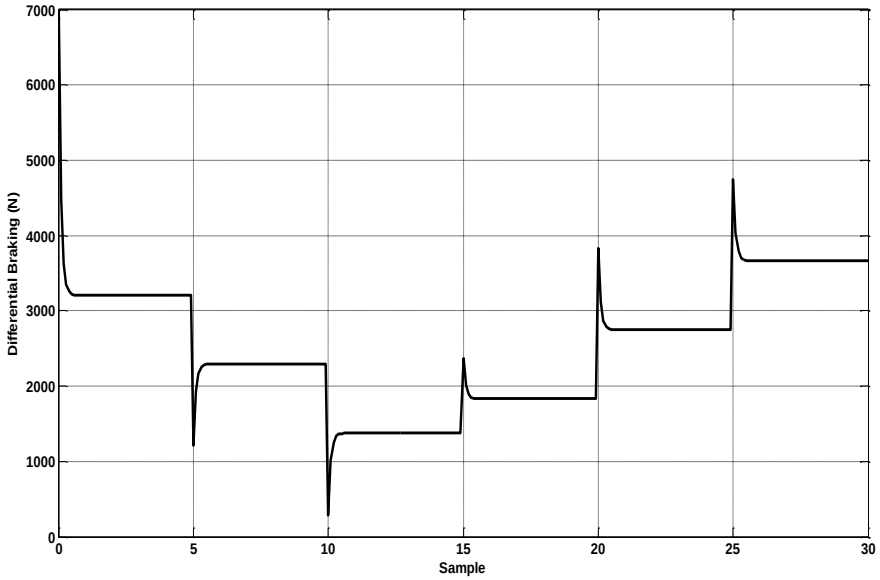


Figure (13-a): The differential brake response.

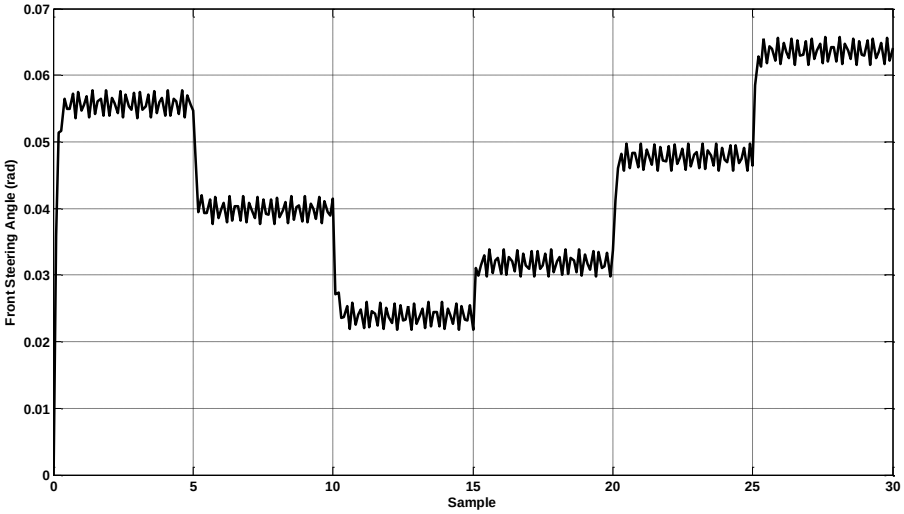


Figure (13-b): The front steer angle response.

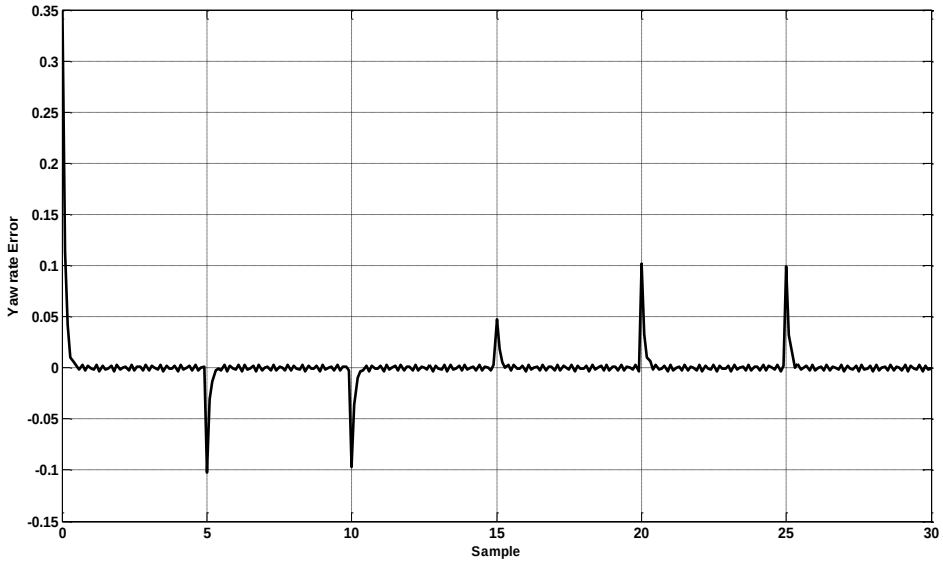


Figure (14-a): The yaw rate error response.

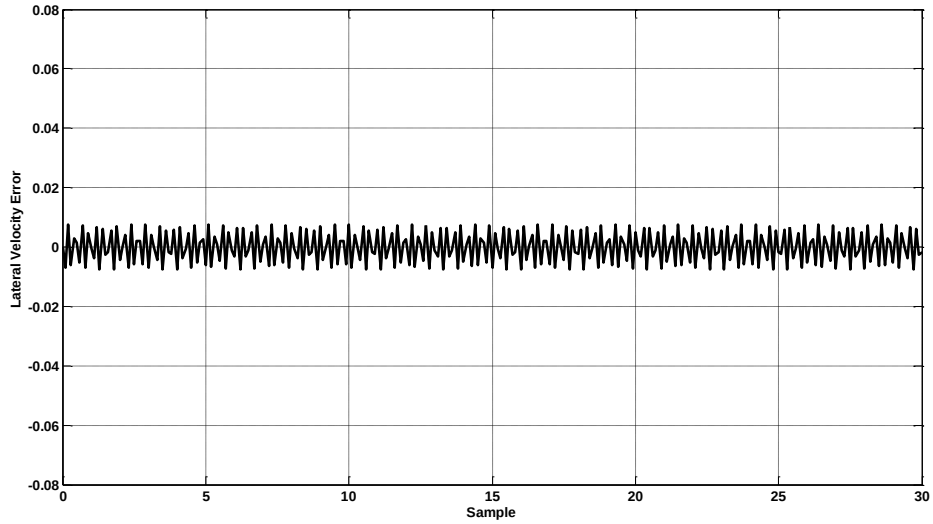


Figure (14-b): The lateral velocity error response.

The gains of the nonlinear PID neural controller as scale function are shown in figures (15-a and b).

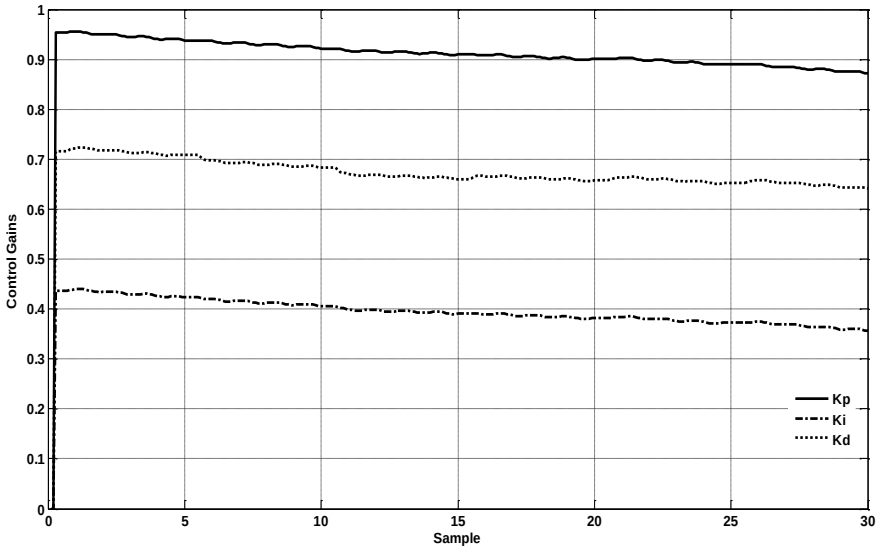


Figure (15-a): The PID control gain parameters kp_1, ki_1, kd_1.

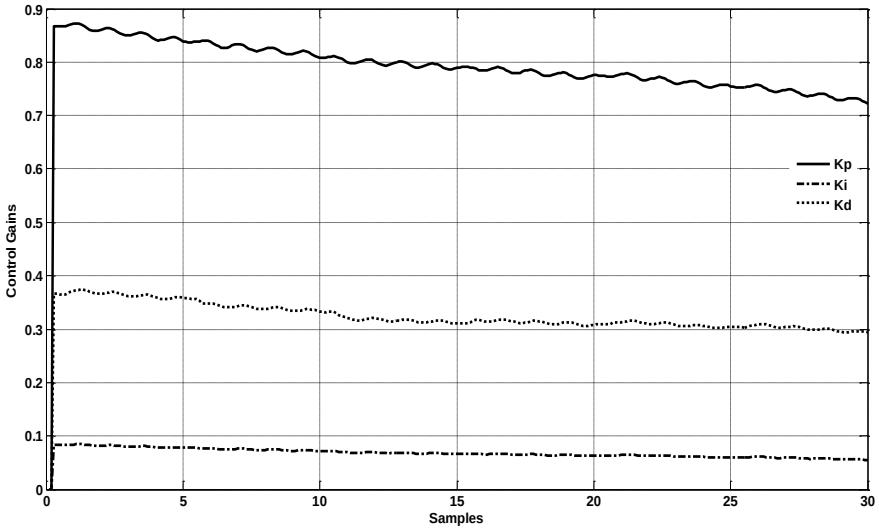


Figure (15-b): The PID control gain parameters kp_2, ki_2, kd_2.

CONCLUSIONS

The nonlinear PID neural network controller with particle swarm optimization algorithm technique for MIMO linear dynamic vehicle motion model has been presented in this paper. The state outputs of the 2DOF vehicle model are yaw rate and lateral velocity and they are followed the desired inputs because there are two control actions front steer angle and differential brake that are generated from the proposed controller with PSO algorithm that is used to tune the nonlinear PID neural controller with minimum time and more stability of the controller and no oscillation with best

parameters of the controller have been found. Simulation results of lateral motion show that the effectiveness of proposed nonlinear PID neural controller; this is demonstrated by the minimized tracking error of the lateral velocity and the yaw rate as well the smoothness control signal obtained at different velocities of the vehicle model, especially with regards to the external front steering angle disturbances attenuation problem.

REFERENCE

[1]- L. Chu, Y Zhang, M. Xu, Y. Shi and Y. Ou, Design of a Robust Vehicle Dynamics Controller Based on Optimal Fuzzy Theory for Passenger Car, Proceedings of the 2nd International Conference on Industrial Mechatronics and Automation, 2010, pp. 105-109.

[2]- C-S. Ting and Y-N. Chang, a Robust Fuzzy Neural Control Approach for Vehicle Lateral Dynamics, Procidia Engineering, Vol. 29, 2012 , pp.479-483.

[3]- A.T. Zanten, R. Erhaedt and G. Pfaff, The vehicle dynamics control system of Bosch. SAE International Congress and Exposition, 1995.

[4]- A.T Zanten, R. Erhaedt, K. Landesfeind and G. Pfaff, Systems Development and Perspective. SAE World Congress, 1998.

[5]- H. S. Al-Araji and A. S. Al-Araji, Vehicle Lateral Velocity and Yaw Rate Control Using Robust Integral Controller, Engineering and Technology Journal by University of Technology Vol. 23, No. 3, 2004, pp. 95-103.

[6]- H-S. Tan and C-Y. Chan, Design of Steering Controller and Analysis of Vehicle Lateral Dynamics under Impulsive Disturbances, Proceedings of the American Control Conference, Chicago, Illinois June 2000, pp. 2023-2027.

[7]- C.Villegas, M. Akar, R.N.Shorten, J. Kalkkuhl, a Robust PI Controller for Emulating Lateral Dynamics of Vehicles, Proceedings of the IEEE Intelligent Vehicles Symposium Istanbul, Turkey, June 13-15, 2007, pp. 828-833.

[8]- H-L. Zhou and Z-Y. Liu. Design of Vehicle Yaw Stability Controller Based on Model Predictive Control. Proceedings of the IEEE Catalog Number: 978-1-4244-3504-3/09. October 15, 2009, pp. 802-807.

[9]- A. S. Al-Araji "Design of a Neuro-Controller for Vehicle Lateral Velocity and Yaw Rate based Genetic Algorithm with Model Reference Guided, Iraqi Journal of Computers, Communication, Control and Systems Engineering by University of Technology Vol. 6, No.1, pp. 30-42, 2006.

[10]- Y. Lee and H. Zak, Designing a Genetic Neural Fuzzy Antilock-Brake-System Controller, Proceedings of the IEEE Transactions on Evolutionary Computation, Vol.6, No.2, April 2002, pp. 198-211.

[11]- M. Chadli, A. E. Hajjaji and A. Rabhi, H∞ Observer-based Robust Multiple Controller Design for Vehicle Lateral Dynamics, Proceedings of the American Control Conference Marriott Waterfront, Baltimore, MD, USA, June 30-July 02, 2010, pp. 1508-1513.

[12]- F. Cabello, A. Acuña, P. Vallejos, M. Orchard and J. Ruiz, Design and Validation of a Fuzzy Longitudinal Controller Based on a Vehicle Dynamic Simulator, Proceedings of the 9th IEEE International Conference on Control and Automation (ICCA), Santiago, Chile, December 19-21, 2011, pp. 997-1002.

[13]- B. Zheng, S. Anwar , Yaw Stability Control of a Steer-by-wire Equipped Vehicle via Active Front Wheel Steering, Mechatronics, Vol. 19, 2009, pp. 799–804.

[14]-T. Pilutti, G. Ulsoy, and D. Hrovat, Vehicle Steering Intervention through Differential Braking, Journal of Dynamic Systems, Measurement and Control. Vol. 120, September 1998, pp. 1667-1671.

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

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

[17]- J. Derrac, 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.
[18]- J. Zhou, 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.
[19]- Y.S. Lee, 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.

Appendix (1)
Nomenclature

a= distance from the center of mass to front axle
b= distance from the center of mass to rear axle
T=vehicle track
C_i = tire cornering stiffness
g= acceleration of gravity
I= vehicle moment of inertia
M= vehicle mass
r= yaw rate
r_d = desired yaw rate
R= curvature radius
U= vehicle velocity
V= lateral velocity
δ_f = front steering angle
F_{BS} =brake steer force
δ_{dis} = front steering angle disturbance
M_{BS} = brake steer moment.
F_{XR} and F_{XL} are front and rear longitudinal tire forces respectively.

Appendix (2)
Vehicle nominal parameters

M=1000Kg
a=1m
b=1.5m
T=1.5m
I=1500Kg m^2
C_f =55000 N/rad
C_r = 45000 N/rad
U= 35, 25, 15, 20, 30 and 40 m/sec
R=100m