ISSN No. 0976-5697

Volume 2, No. 2, Mar-Apr 2011

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

Design Neural Control System for Full Vehicle Nonlinear Active Suspension with Hydraulic Actuators

Ammar A. Aldair*
School of Engineering and Design,
University of Sussex
Brighton, BN1 9QT, UK
aa386@sussex.ac.uk

Weiji J. Wang
School of Engineering and Design,
University of Sussex
Brighton, BN1 9QT, UK
w.j.wang@sussex.ac.uk

Abstract: The main objective of designed the controller for a vehicle suspension system is to reduce the discomfort sensed by passengers which arises from road roughness and to increase the ride handling associated with the pitching and rolling movements. This necessitates a very fast and accurate controller to meet as much control objectives, as possible, this paper deals with an artificial intelligence Neural Control technique to design a robust controller. The advantage of this controller is that it can handle the nonlinearities faster than other conventional controllers. The approach of the proposed controller is to minimize the vibration on each corner of vehicle by generating suitable control signals. This control signals will be used as input to the hydraulic actuators which will generate appropriate control forces to improve the vehicle performances. A full vehicle nonlinear active suspension system with hydraulic actuators is introduced and tested. The robustness of the proposed controller is being assessed by comparing with an optimal Fractional Order PI^iD^d (FOPID) controller. The results show that intelligent neural controller have improved dynamic response measured by a decreased cost function.

Keywords: Full vehicle, Nonlinear Active Suspension System with Hydraulic Actuators, Neural Controller.

I. INTRODUCTON

A number of researchers have suggested control methods for vehicle suspension systems. Some have designed a linear controller for a quarter or half vehicle [1-9]. Gaspar et al. in Reference [10] have used a robust controller for a full vehicle linear active suspension system using the mixed parameter synthesis. A sliding mode technique is designed for a linear full vehicle active suspension system [11]. In this Reference a method is developed for the purpose of sensor fault diagnosis and accommodation. In Reference [12], the authors present the development of an integrated control system of active front steering and normal force control using fuzzy reasoning to enhance the full vehicle model handling performance. A fuzzy logic based fast gain scheduling controller is proposed for control nonlinear suspension systems for quarter car system [13]. In fact, nonlinearity inherently exists in damper and spring models [14-16]. Therefore, the nonlinear effect should be inevitably taken into account to design the controller for practical active suspension system.

This paper will be developed a novel Neural Controller for full vehicle nonlinear active suspension system with hydraulic actuators. The full vehicle model will be investigated to take into account the three motions of the vehicle: vertical movement at centre of gravity, pitching movement and rolling movement.

Neural Networks (NNs) are capable of handling complex and nonlinear problems, process information rapidly and can reduce the engineering effort required in controller model development. Artificial neural networks are made up of a simplified individual models of the biological neuron that are connected together to form a network. It consists of a pool of simple processing units which communicate by sending signals to each other over a large number of weighted connections. Capability of learning information by example; ability to generalize to new input and robustness to noisy data are the important properties of neural networks. From these properties, neural networks are able to solve complex problems that are currently intractable.

The artificial neural network is an intelligent device wildly used to design robust controllers for nonlinear processes in engineering problems. In many publications, neural networks are used to design controllers, such as the model reference adaptive control, model predictive control, nonlinear internal model control, adaptive inverse control system and neural adaptive feedback linearization [17, 18]. The control architectures in these papers depend on designing a neural network identifier and then this identifier is used as a path to propagate the error between the output of the process and output of the reference model to train and select the optimal values of the neural network control. Therefore, in those methods two neural networks were trained to track several control objectives.

One of the main advantages of using a neural network as a controller is that neural networks are universal function approximations which learn on the basis of examples and may be immediately applied in an adaptive control system because of their capacity to adapt in real time. There are many learning algorithms available to obtain the optimal values of the trainable parameters of neural network. The back-propagation algorithm (BPA) has been known as an algorithm with a very poor convergence rate. The Levenberg-Marquardt Algorithm (LMA) is an iterative technique that locates the minimum of a multivariate function that is expressed as the sum of squares of nonlinear real-valued functions [19, 20].

In this paper Fractional Order PI^lD^d (FOPID) will be designed for full vehicle nonlinear active suspension using the Evolutionary Algorithm (EA). The data obtained from the FOPID controller are used as reference to design the neural controller. The Levenberg-Marquardt training algorithm has been used to obtain the optimal values of the trainable parameters. The performance of the neural controller has been improved by adding the Scaling Gains. The scaling gains of the neural controller have been adjusted using Golden Section Search (GSS) method. The effectiveness and robustness of the proposed neural network controller and FOPID controller will be compared. Four types of the disturbances will be investigated to establish the robustness of the proposed controller: change the amplitude of the sine shape of the road profile input, change the amplitude of the square shape of the road profile input, change the bending inertia torque with random road profile input and change the breaking inertia torque with random road profile input. The results will show whether the proposed controller is more robust than the optimal PID controller.

II. MATHEMATICAL MODEL OF THE HYDRAULIC ACTUATOR

Hydraulic actuators are important equipment and widely used because of their high power capability, fast and smooth response characteristics and good positioning capability [21]. The hydraulic actuators are commonly used in various industries and engineering such as materials test machines, vehicle active suspension systems, mining machinery, flight simulators, paper making machines, ships and electromagnetic marine engineering, injection melding machines, robotics, and steel and aluminum mill equipment.

For understanding the performance of the improved suspension system and for developing a robust controller for this system, to develop an accurate dynamic model for a hydraulic servo-system is the first step. Therefore, a description of the dynamics for the fluid subsystem, the servo-valve, the cylinder and the load are required. The electro-hydraulic system here is a cylinder controlled by the input voltage signal to the servo-valve. The cylinder is attached to sprung mass of the vehicle and connected in parallel with a passive system unit, i.e. nonlinear spring and nonlinear damper. The Hydraulic actuators are used to generate the forces between the vehicle's body and the axle to enhance and improve the riding and handling qualities in a modern vehicle.

Figure 1 shows the physical model of a hydraulic actuator with a nonlinear spring and nonlinear damper of a quarter vehicle model.

The hydraulic actuator consists of a hydraulic cylinder, a piston, servo-valves, an electrical pump and a reservoir. P_s and P_r are the pressures of the hydraulic fluid supplied from and returned to the reservoir, respectively. When there is a difference between fluid pressure in upper cylinder chamber and fluid pressure in lower chamber, the piston in the hydraulic cylinder extends or compresses and a suitable damping force is generated for the suspension to improve the vehicle's dynamic performance. The dynamic equation of the hydraulic actuator is given as [22]

$$\dot{P}_{Li} = -\beta P_{Li} - \sigma(A_p \dot{x}_{pi} - Q_i) \quad (1)$$

where $\sigma = \frac{4\beta_e}{V_t}$, $\beta = \sigma C_{tp}$, $x_{pi} = z_i - w_i$ is the relative displacement between the suspension point and wheel, P_{Li} is the hydraulic pressure inside actuator, β_e is the effective bulk modulus of hydraulic system, V_t is the total volume of fluid under compression, C_{tp} is the leakage coefficient of piston and A_p is the cross section area of the piston. Hydraulic flow through the piston inside the i^{th} actuator Q_i is governed by the following equation

$$Q_i = C_d \omega x_{vi} \sqrt{\frac{1}{\rho}(P_{s\,i} - sgn(x_{v\,i}) P_{L\,i})} \quad (2)$$

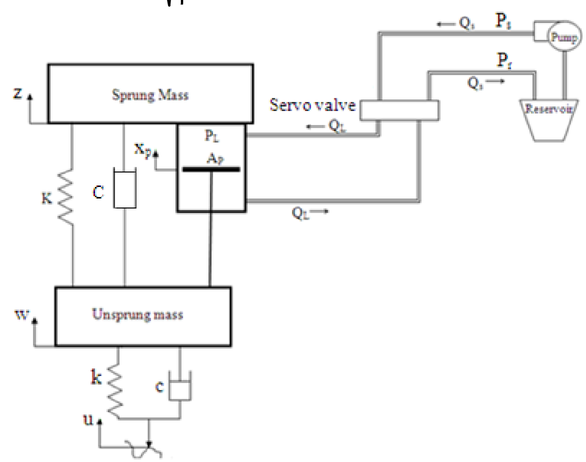

Figure 1 Active Suspension with Hydraulic Actuator

where C_d is the discharge coefficient, ω the area gradient, x_v the spool valve displacement, ρ the fluid density. The spool valve displacement is controlled by an input voltage u_m. The corresponding dynamic relationship can be simplified as a first order differential equation

$$\dot{x}_{v\,i} = \frac{1}{\tau}(u_{mi} - x_{vi}) \quad (3)$$

The nonlinear force produced by the active hydraulic actuator is applied between body and wheel axles. This force is governed by the following equation [23]

$$F_{A\,i} = A_p P_{Li} \quad (4)$$

where A_p the cross section area of the piston inside the i^{th} actuator, P_{Li} the hydraulic pressure inside the i^{th} actuator.

Due to rubbing of the piston with the inside actuator wall, heat will be generated. Therefore, the actual force generated by the i^{th} hydraulic actuator F_{Ai} is not equal to the force supply by i^{th} hydraulic actuator F_{Pi}. The difference between these two forces is named as frictional force F_{fi}. This force can not be neglected because the value of this force is big and can be greater than 200 Nm [24]. Frictional force is modelled with a smooth approximation of Signum function

$$F_{fi}=\begin{cases}\kappa\, sgn(\dot z_i-\dot w_i)\ \ if\ \ |\dot z_i-\dot w_i|>0.01\\ \kappa\, sin\left(\dfrac{\dot z_i-\dot w_i}{0.01}\dfrac{\pi}{2}\right)\ \ if\ |\dot z_i-\dot w_i|<0.01\end{cases} \tag{5}$$

III. MATHEMATICAL MODEL OF THE CONTROLLED SYSTEM

A framework is being suggested in which an active suspension controller generates suitable command signals as the inputs of the hydraulic actuators to improve the vehicle performance including riding comfort and road handling stability. The riding comfort can be measured by evaluating the displacement and acceleration of the sprung mass. The handling stability can be obtained by minimising the vertical and the rotational motions of the vehicle body including rolling and pitching motions during sharp manoeuvres of cornering and braking.

A full vehicle physical model with active suspension is proposed by the authors and shown in Figure 2. This model consists of five parts: the vehicle body mass (*M*) and four unsprung masses m_i (where $i\in[1,2,3,4]$). The vehicle body mass is assumed to be a rigid body and has degrees of freedom in vertical, pitch and roll directions. The vertical displacements at each suspension point are denoted by z_1, z_2, z_3 and z_4. The z_c, α and η denote the displacement, pitch angle and roll angle at the centre of gravity of the vehicle, respectively. J_x and J_y are the moments of inertia about x-axis and y-axis, respectively. The cornering torque and breaking torque are denoted by T_x and T_y, respectively. In the model, the disturbances u_1, u_2, u_3 and u_4 are caused by road roughness. The vertical displacements of unsprung masses are denoted by w_1, w_2, w_3 and w_4.

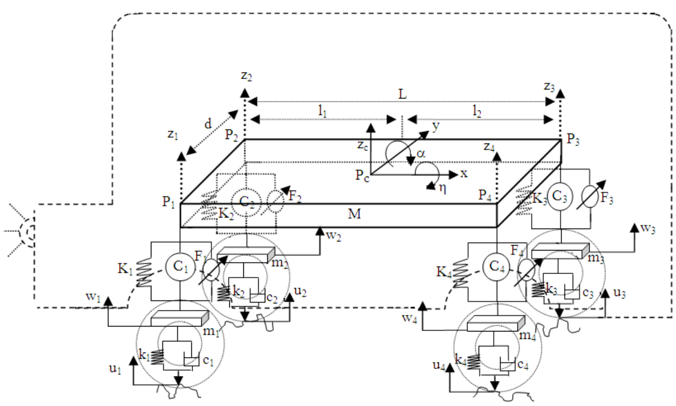

Figure 2 Full Vehicle Nonlinear Active Suspension System

The suspension components possess nonlinear property. Therefore, each suspension will be assumed as a nonlinear device with nonlinear spring and nonlinear damper placed in parallel with the hydraulic actuator. The main purpose of using the suspension control is to generate an actuating force between the vehicle body mass and unsprung masses. The i^{th} nonlinear suspension has stiffness coefficient and damping coefficient denoted by K_i and C_i, respectively. Each tyre will be simulated as linear oscillator with stiffness and damping coefficient denoted by k_i and c_i, respectively.

The motion of the vehicle body mass is governed by the following equations:

- **Vertical motion**

Using Newton's Second law of motion

$$M\ddot z_c=-\sum_{i=1}^{4}F_{Ki}-\sum_{i=1}^{4}F_{Ci}+\sum_{i=1}^{4}F_{Pi} \tag{6}$$

where F_{Ki} is the i^{th} nonlinear suspension spring force which can be written as[25]

$$F_{Ki}=K_i(z_i-w_i)+\xi K_i(z_i-w_i)^3$$

$$F_{Pi}=F_{Ai}-F_{fi}\ ,$$

where F_{Ai} is the nonlinear hydraulic force provided by the i^{th} actuator and F_{fi} the nonlinear frictional force.

- **Pitching motion**

Using the Newton's law for the pitching motion

$$\begin{aligned}J_x\ddot\alpha&=(F_{K1}-F_{K2}-F_{K3}+F_{K4})\frac{b}{2}+\\&(F_{C1}-F_{C2}-F_{C3}+F_{C4})\frac{b}{2}+\\&(F_{P4}-F_{P1}+F_{P3}-F_{P2})\frac{b}{2}+T_x\end{aligned} \tag{7}$$

where *b* is the distance between the front wheels.

- **Rolling motion**

Using the Newton's law again for the rolling motion

$$\begin{aligned}J_y\ddot\eta&=(F_{K3}+F_{K4})l_2-(F_{K1}+F_{K2})l_1+\\&(F_{C3}+F_{C4})l_2-(F_{C1}+F_{C2})l_1+\\&(F_{P1}+F_{P2})l_1+(F_{P3}+F_{P4})l_2+T_y\end{aligned} \tag{8}$$

where l_1 is the distance between the centre of front wheel axle and centre of gravity of the vehicle. l_2 is the distance between the centre of gravity of the vehicle and the centre of rare wheel axle.

The motion of the i^{th} unsprung mass is governed by the following equation

$$m_i\ddot w_i=-k_i(w_i-u_i)-c_i(\dot w_i-\dot u_i)+F_{Ki}+F_{Ci}-F_{Pi} \tag{9}$$

IV. THE STRUCTURE OF NEURAL NETWORK

Neural network architecture is quite simple to create and involves two or more neurons combined to form one or more layers. Figure 3 depicts the structure of multilayer neural network (or some time called multilayer perceptron network). In this figure, the neural network model has three layers: input layer, hidden layer and output layer. The input layer represents the input variables related to the problem. The output layer represents the desire output response of the system. The nodes in the hidden layer and output layer are the processing elements that allow the network to develop a behavioural representation of the problem space being addressed. Processing of the input information occurs at each of the hidden and output nodes within the network and is computationally relatively simple. Each node in the particular layer of the network is connected to all of the nodes in the previous layer. There is a weight value associated with each of the connection between nodes. The weighted inputs to a particular node are summed and the resultant value is passed through a nonlinear activation function to determine the output value of the node. Therefore, each node has multiple inputs and single output.

The output of the k^{th} node in the hidden layer can be given as:

$$q(1,k) = \Gamma_1(\sum_{i=1}^{n} w(k,i)x_i + b_1(k)) \quad k = 1,2,\dots,s_1 \quad (10)$$

where $q(1,k)$ is the output of the k^{th} node in the hidden layer, Γ_1 is the activation function of the hidden layer, $w(k,i)$ is the weight between i^{th} input and k^{th} node, x_i is the i^{th} input and $b_1(k)$ is the bias of the k^{th} node.

The output of the l^{th} node in the output layer can be given as

$$q(2,l) = \Gamma_2\left(\sum_{j=1}^{s_1} v(j,l)q(1,j) + b_2(l)\right) \quad l = 1,2,\dots,s_2 \quad (11)$$

where $q(1,l)$ is the output of the l^{th} node in the output layer, Γ_2 is the activation function of the output layer, $v(j,l)$ is the weight between j^{th} node and l^{th} node, $q(1,j)$ is the j^{th} output of the hidden layer and $b_2(l)$ is the bias of the l^{th} node.

There are many training algorithms which can be used to determine the optimal values for the trainable parameters (weights and biases) between each of the nodes to minimise a Mean Squared Error (MSE) function

$$MSE = \frac{1}{2P}\sum_{i=1}^{P}\sum_{j=1}^{s_2}(q_i(2,j) - T_i(j))^2 \quad (12)$$

Where P is the number of data for each epoch, $q_i(2,j)$ is the output of j^{th} node in the output layer for i^{th} epoch and $T_i(j)$ is the j^{th} desire output for i^{th} epoch.

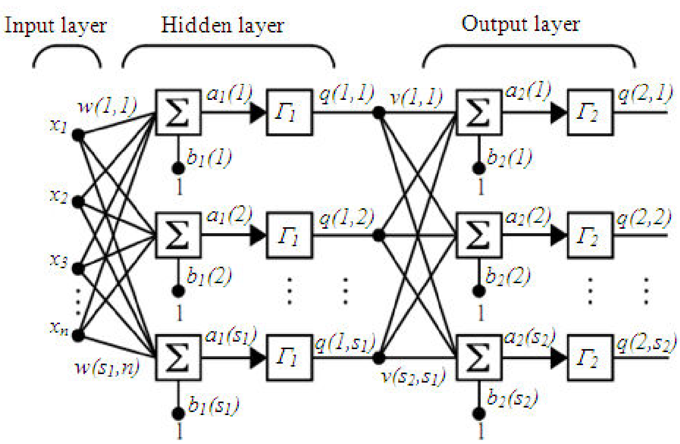

Figure 3 Multilayers Neural Network

V. LEVENBERG-MARQUARDT TRANINING ALGORITHM

The training phase of the neural networks required a set of examples of proper network behaviour i.e. network inputs and target output. The trainable parameters of neural network are adjusted during training phase to minimize the mean squared error (MSE) between the neural network outputs and the target outputs. At the first iteration, the trainable parameters of the neural network are randomly initialized. The neural network processes each input vector and the output of the neural network is compared with the desired output (target output).

The Back-Propagation Algorithm has been a significant improvement in neural network researches [26-30]. The simplest implementation of backpropagation learning updates trainable parameters in the direction in which the performance function (mean squared error) decreases most rapidly (the negative of the gradient). The Backpropagation Algorithm has been known as an algorithm with a very poor converging rate for practical problems [31]. Many researches were carried out to accelerate the convergence of the algorithm. These researches focused on two different categories. The first category uses heuristic techniques, which were developed from an analysis of the performance of the backpropagation algorithm [32-34]. There are three different heuristic techniques: Momentum Technique, Variable Learning Rate Technique and Resilient Technique. In the second category of fast algorithms uses standard numerical optimization techniques. There are three main techniques of numerical optimization: Conjugate Gradient Technique [35], Quasi-Newton Technique [36] and Levenberg Marquardt Technique [37].

The Conjugate Gradient Technique produces generally faster convergence than steepest descent directions by searching along conjugate direction. The Quasi-Newton Technique is faster than the conjugate gradient technique, but it is complex and expensive to compute the Hessian matrix for feedforward NNs. These two algorithms lead to little acceptable result when the nonlinearity is heavy.

The Levenberg-Marquardt Technique (LMT) is an iterative technique that locates the minimum of a multivariate function expressed as the sum of squares of nonlinear real-valued functions. It has become a standard technique for nonlinear least-squares problems [31], widely adopted in a broad spectrum of disciplines. The LMT can be thought of as a combination of Backpropagation Algorithm and the Quasi-Newton Technique [38]. When the current solution is far from the correct one, the algorithm behaves like a Backpropagation Algorithm: slow, but guaranteed to converge. When the current solution is close to the correct solution, it becomes a Quasi-Newton method.

With the LMA, the increment of the trainable parameters vector can be calculated as follows:

$$\boldsymbol{\Delta\psi} = [\boldsymbol{J}^T\boldsymbol{J} + \boldsymbol{\lambda I}]^{-1}\boldsymbol{J}^T\boldsymbol{e} \quad (13)$$

where $\boldsymbol{\psi}$ is the trainable parameters vector; $\boldsymbol{I}$ is identity matrix; $\boldsymbol{J}$ is the Jacobian matrix and $\boldsymbol{\lambda}$ is the learning rate which automatically adjust during learning phase;

$\boldsymbol{e}$ is the cumulative error vector, it can be written as follows:

$$\boldsymbol{e} = \left[e_{11}\ e_{21}\ \dots\ e_{s_21}\ e_{12}\ e_{22}\ \dots\ e_{s_22}\ \dots\ e_{1P}\ e_{2P}\ \dots\ e_{s_2P}\right]^T$$

where P is number of input data, s_2 number of outputs.

$$e_{ji} = q_i(2,j) - T_i(j)$$

where $i = 1,2,\dots,P$ and $j = 1,2,\dots,S_2$.

If performance measure (MSE) in epoch $p+1$ is greater than the performance measure in epoch p, λ is divided by constant number ζ $(0 < \zeta < 1)$, whenever performance measure decreased, λ is multiplied by ζ. Equation (5.25) shows that if λ is equal to zero the LMT becomes Quasi-Newton Technique (in this technique the increment of the trainable parameters vector $\boldsymbol{\Delta\psi} = [\boldsymbol{J}^T\boldsymbol{J}]^{-1}\boldsymbol{J}^T\boldsymbol{e}$) and if μ is high the LMT becomes Backpropagation Algorithm.

The Jacobian matrix (it has $\beta \times s_2P$ dimension) can be compute as follows:

$$[J] = \begin{bmatrix} \frac{\partial e_{11}}{\partial \psi_1} & \frac{\partial e_{11}}{\partial \psi_2} & \cdots & \frac{\partial e_{11}}{\partial \psi_\beta} \\ \frac{\partial e_{21}}{\partial \psi_1} & \frac{\partial e_{21}}{\partial \psi_2} & \cdots & \frac{\partial e_{21}}{\partial \psi_\beta} \\ \vdots & \vdots & \cdots & \vdots \\ \frac{\partial e_{m1}}{\partial \psi_1} & \frac{\partial e_{m1}}{\partial \psi_2} & \cdots & \frac{\partial e_{m1}}{\partial \psi_\beta} \\ \vdots & \vdots & \cdots & \vdots \\ \frac{\partial e_{1P}}{\partial \psi_1} & \frac{\partial e_{1P}}{\partial \psi_2} & \cdots & \frac{\partial e_{1P}}{\partial \psi_\beta} \\ \frac{\partial e_{2P}}{\partial \psi_1} & \frac{\partial e_{2P}}{\partial \psi_2} & \cdots & \frac{\partial e_{2P}}{\partial \psi_\beta} \\ \vdots & \vdots & \cdots & \vdots \\ \frac{\partial e_{s_2P}}{\partial \psi_1} & \frac{\partial e_{s_2P}}{\partial \psi_2} & \cdots & \frac{\partial e_{s_2P}}{\partial \psi_\beta} \end{bmatrix}_{(\beta \times s_2 P)}$$

where β is the total number of trainable parameters in neural network which must be optimized; e_{kp} is the error of k^{th} output at p^{th} epoch.

Therefore, trainable parameters vector can be updated as follows:

$$\boldsymbol{\psi}^{p+1} = \boldsymbol{\psi}^{p} + \boldsymbol{\Delta\psi} = \boldsymbol{\psi}^{p} + [\boldsymbol{J}^T\boldsymbol{J} + \boldsymbol{\lambda I}]^{-1}\boldsymbol{J}^T\boldsymbol{e} \qquad (14)$$

VI. DESIGN THE PROPOSED NEURAL CONTROL

Neural Networks (NNs) are capable of handling complex and nonlinear problems, process information rapidly and can reduce the engineering effort required in controller model development. Figure 4 depicts the controlled vehicle system with a neural controller as a key component. A neural controller has been designed to generate suitable control signals. The control signals will be applied as a control input signals to govern the hydraulic actuators to generate suitable damping forces for improving the vehicle performance. To find the optimal values of the trainable parameters of the neural controller for driving the plant to meet all control objectives, an FOPID sub-controller should be designed (the details for the full design of FOPID controller for full vehicle nonlinear suspension are described in Reference [39]). In author's work, the input and output data obtained from the FOPID controller should be used to train the parameters of neural controller using the LM Training Algorithm. Figure 5 depicts the training phase of the Neural Controller.

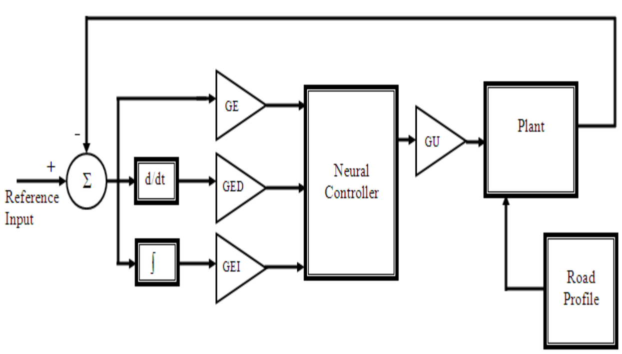

Figure 4 Neural Controller for a Full Vehicle Model

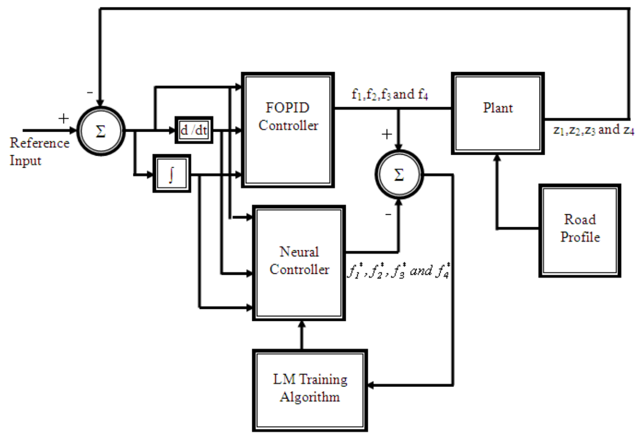

Figure 5 Training Phase of Neural Controller

The bold line means multiple input or output signals. After the optimal values of trainable parameters are obtained the neural controller design should be improved by adjusting the scaling gains, i.e. GE, GED, GEI and GU, as shown in Figure 4. To select the optimal values of the scaling gains, four dimensional Golden Section Search (4-D GSS) method are introduced to reduce the trial time (For more detail about 4-D GSS method, see Reference [40]).

VII. SIMULATION AND RESULTS

The input road profile is selected as a white noise random signal. To design the neural controller, the optimal parameters of the FOPID controller should be obtained first using the EA. In author's work, the input and output data obtained from the FOPID controller have been used to design the neural controller. The LM Training Algorithm has been used to modify the training parameters to track the output data obtained from the FOPID controller. Four neural controllers have been designed, in which one for each suspension. To improve the performance of the neural controller the scaling gains should be adjusted. The 4-D GSS method has been used to adjust the scaling gains (GE, GED, GEI and GU). As the result, the optimized values of scaling gains are 4, 3, 10 and 0.5, respectively.

All vehicle body variables, including vertical displacement at the centre of gravity: z_c, vertical displacement at P_3: z_3, pitch angle: α and roll angle: η, depend on the vertical displacements at points P_1, P_2 and P_4 (z_1, z_2 and z_4, respectively). The suspension deflection ($z_i - w_i$) and body acceleration ($\ddot{z}_c$) are used to evaluate the road handling and riding comfort of the passengers, respectively. By supplying the control signal, it is expected that just the vertical displacements of sprung mass (z_i) and body acceleration ($\ddot{z}_c$) will be targeted to reduce while the vertical displacements of unsprung masses (w_i) are not concerned. Therefore, when z_i decreases, the road handling performance will be improved while body acceleration decreases, the riding comfort is improved. In this paper just the responses of the vertical displacements at P_1, P_2 and P_4 and the body acceleration have been shown for comparison. In Figures 6-8 the responses of vertical displacements at P_1, P_2 and P_4 are compared, respectively for the full vehicle nonlinear active suspension system without controller (passive system) and with neural controller. Figure 9 shows the response of the acceleration at the vehicle's centre of gravity. From these figures it can be seen that the proposed controller is powerful and efficient.

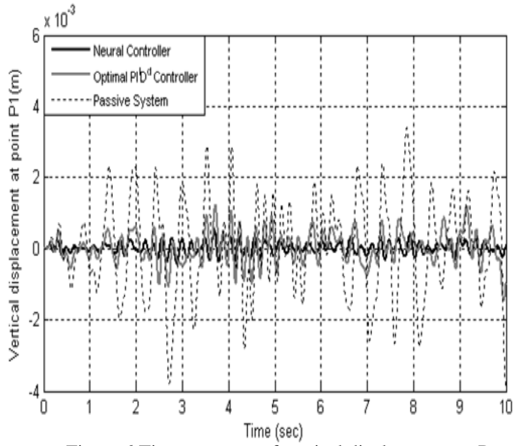

Figure 6 Time response of vertical displacement at P_1

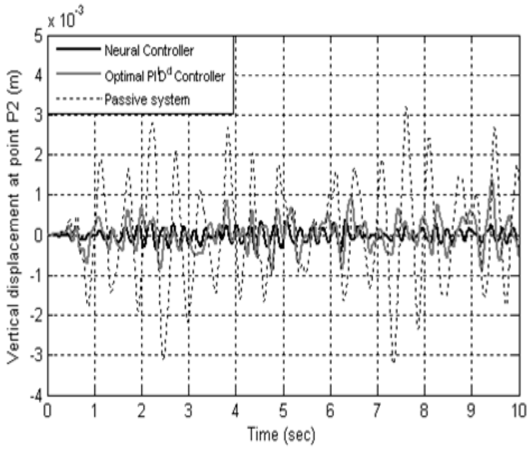

Figure 7 Time response of vertical displacement at P_2

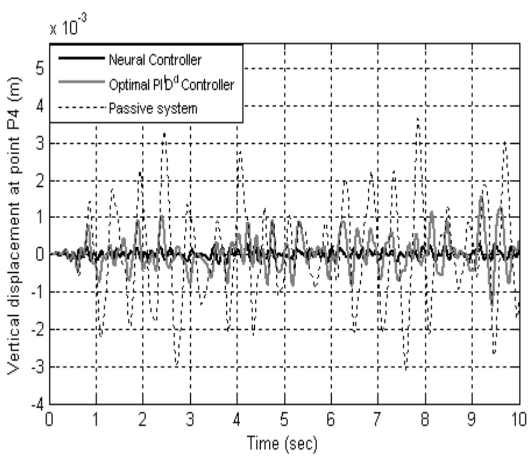

Figure 8 Time response of vertical displacement at P_4

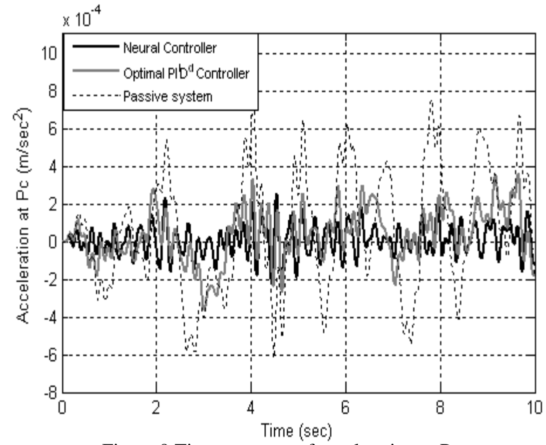

Figure 9 Time response of acceleration at P_c

The efficient controller is the controller that it is still stable even the disturbance signal is applied on the plant. Therefore, to establish the effectiveness of any controller the robustness should be examined. Four types of disturbances are applied in turn to test the robustness of the neural controller.

- ***Square input signal with varying amplitude applied as road input profile***

The square input signal has been applied as road input. The amplitude of this signal has been changed from 0.01m to 0.08m. At each value, the cost function (as described in equation 15) has been calculated:

$$\phi = 0.5 \sum_{\varepsilon=1}^{4} z_{\varepsilon}^{2} \qquad (15)$$

Figure 10 shows the time response of the cost function as function of amplitude of square signal input.

- ***Sine wave input signal with varying amplitude applied as road input profile***

The different amplitude of sine wave input from 0.01m to 0.08m has been applied as road profile input. The time response of the cost function for the full vehicle without control, the result of optimal FOPID controller and neural controller are shown together in Figure11.

- ***Bending inertia Torque (T_x) applied***

The value of bending torque (from 1000 Nm to 9000Nm) in addition to random signal as road profile has been applied. The cost function response is plotted as function of bending torque (T_x) in Figure 12.

- ***Breaking inertia Torque (T_y) applied***

The value of breaking torque (from 1000 Nm to 9000Nm) in addition to random signal as road profile has been applied. In Figure 13, the cost function response is plotted as function of braking torque (T_y).

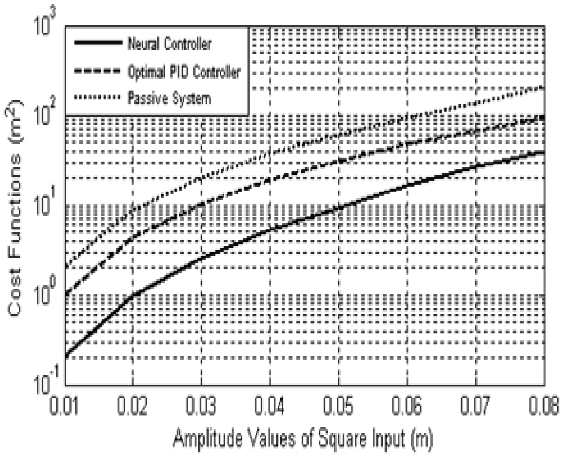

Figure 10 Time Response of the Cost Functions Against the Different Amplitude of Square Input.

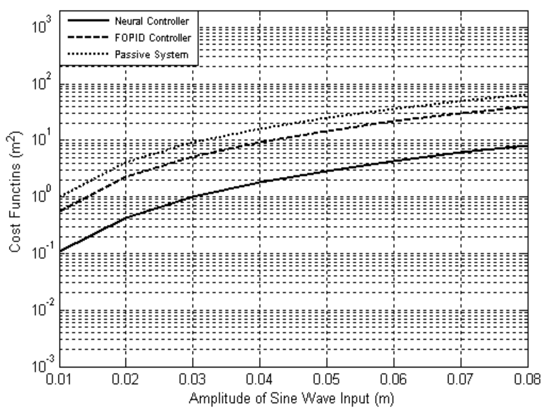

Figure 11 Time Response of the Cost Functions Against the Different Amplitude of Sine Wave Input.

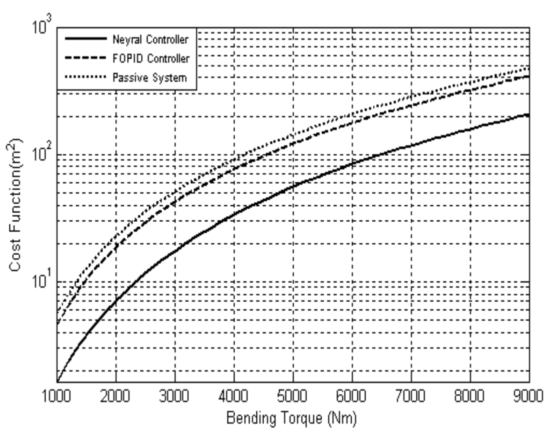

Figure 12 Time Response of the Cost Functions Against Bending Torque (T_x)

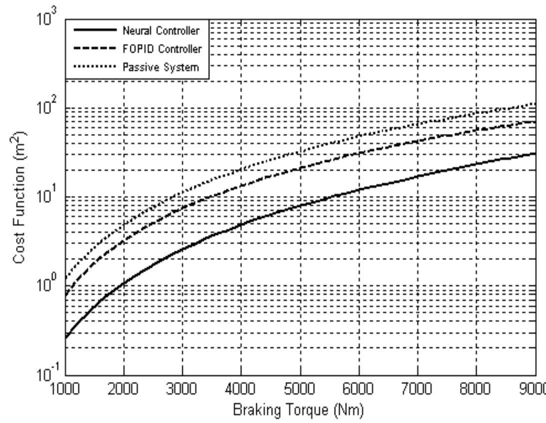

Figure 13 Time Response of the Cost Functions Against Braking Torque (T_y)

VIII. CONCLUSION

From the results, the neural controller has capability of minimizing the control objectives better than the FOPID controller. The ride comfort and the road handling will be improved by using the neural controller. It has been confirmed that the proposed controller is more robust than the FOPID controller. The test of the robustness proves that the neural controller is still stable and it forces the cost function to be minimum even significant disturbances occurred. The proposed controller just has one neural network, which means it will be more economic than the other neural network controllers, such as model reference adaptive neural control, model predictive neural control, nonlinear internal model neural control, adaptive inverse neural control system or neural adaptive feedback linearization. In all of these methods a minimum of two neural networks to design the controller are used (in the adaptive inverse control system it must be used three neural networks to design the controller) one as identifier and other one as controller.

IX. REFERENCES

[1] Sung, K., et al., Discrete-time Fuzzy Sliding Mode Controller for a Vehicle Suspension System Featuring an Electrorheological Fluid Damper. Smart Materials and Structures, 2007. **16**: p. 798-808.

[2] Kuo, Y. and T. Li, GA_Based Fuzzy PI/PD Controller for Automotive Active Suspension System. IEEE Transactions on Industral Electronics, 1999. **46**(6): p. 1051-1056.

[3] Feng, J. and F. Yu, GA-Based PID and Fuzzy Logic Controller for Active Vehicle Suspension System. International Journal of Automative Technology, 2003. **4**(4): p. 181-191.

[4] Smith, M. and G. Walker, Performance Limitations and Constraints for Active and Passive Suspensions: a Mechanical Multi-port Approach. Vehicle System Dynamics, 2000. **33**(3): p. 137-168.

[5] Biglarbegian, M., W. Melek, and F. Golnaraghi, A Novel Neuro-fuzzy Controller to Enhance the Performance of Vehicle Semi-active Suspension Systems. Vehicle System Dynamics, 2008. **46**(8): p. 691-711.

[6] Yue, L., C. Tang, and H. Li, Research on Vehicle Suspension System Based on Fuzzy Logic Control, in International Conference on Automation and Logistics. 2008: Qingdao, China.

[7] Biglarbegian, M., W. Melek, and F. Golnaraghi, Design of a Novel Fuzzy Controller to Enhance Stability of Vehicles, N.A.F.I.P. Society, Editor. 2007. p. 410-414.

[8] Kumar, M., Genetic Algorithm-Based Proportional Derivative Controller for the Development of Active Suspension System. Information Technology and Contro, 2007. **36**(1): p. 58-67.

[9] He, Y. and J. Mcphee, A design methodology for mechatronic vehicles: application of Multidisciplinary optimization, multibody dynamics and genetic algorithms. Vehicle System Dynamics, 2005. **43**(10): p. 697-733.

[10] Gaspar, P., I. Szaszi, and J. Bokor, Design of Robust controller for Active vehicle Suspension Using the Mixed μ Synthesis. Vehicle System Dynamics, 2003. **40**(4): p. 193-228.

[11] Chamseddine, A., H. Noura, and T. Raharijana, Control of linear Full Vehicle Active Suspension System Using Sliding Mode Techniques, in International Conference on Control Applications. 2006: Munich, Germany.

[12] March, C. and T. Sjim, Integrated Control of Suspension and front Steering to enhance Vehicle Handling. Processing IMechE, 2006. **221 Part D**: p. 377-391.

[13] Lee, S., G. Kim, and T. Lim, Fuzzy logic Based Fast Gain Scheduling Control for Nonlinear Suspension System. IEEE Transaction on Industrial Electronics, 1998. **45**(6): p. 953-955.

[14] Li, S., S. Yang, and W. Guo, Investigation on Chaotic Motion in Hysteretic Non-linear Suspension System with Multi-frequency Excitations. Mechanics Research Communication 2004. **31**: p. 229-236.

[15] Dixon, J., The Shock Absorber Handbook. 1999, USA: Society of Automotive Engineers, Inc.

[16] Joo, D., et al., Nonlinear Modelling of Vehicle Suspension System, in Proceeding of the American Control Conference. 2000: Chicago, Illinois.

[17] Hussain, M., Review of the applications of neural networks in chemical process control. Simulation and on-line implementations. Artificial Intelligence engineering 1999. **13**: p. 55-68.

[18] Norgaard, M., O. Ravn, and N. Poulsen, NNSYSID and NNCTRL tools for system identification and control with neural networks. Computing and Control Engineering Journal 2001. **23**: p. 29-36.

[19] Lera, G. and M. Pinzolas, Neighborhood Based Levenberg-Marquardt Algorithm for Neural Network Training. IEEE Transactions on Neural Networks 2002. **13**(5): p. 1200-1203.

[20] Martin, T. and B. Mohammed, Training Feedforward Networks with Marquardt Algorithm. IEEE Transaction on Neural Networks, 1994. **5**(6): p. 989-993.

[21] Zulfamtman and Rahmat M., Application of Self-tuning Fuzzy PID Controller on Industral Hydraulic Actoutor Using Syatem Identification approach. International journal on Smart Sensing and Intelligaent System, 2009. **2**(2): p. 246-261.

[22] H, M., Hydrulic Control Systems 1967, New york: John wiley and Sons, Inc.

[23] Merritt, H., Hydraulic Control Systems. 1969, USA: John wiley and Sons,Inc.

[24] Rajamani, R. and J.K. Hedrick, Adaptive Observers for Active Automotive Suspensions: Theory and Experiment. Ieee Transaction on Cintrol Systems Tehnology, 1995. **3**(1): p. 86-92.

[25] Ando, Y. and M. Suzuki, Control of Active Suspension System Using the Singular Perturbation Method Control. Engineering Practice, 1996. **4**(3): p. 287-293.

[26] Rumelhart, D., G. Hinton, and R. Willams, Learning represatations by back-propagation error. Nature, 1986: p. 533-536.

[27] Narendra, K. and K. Parthasarathy, Identification and Control of Dynamical Systems Using Neural Network
IEEE Transaction on Neural Network, 1990. **1**(1): p. 4-27.

[28] Sharkaway, A., Fuzzy Control for the Automobiles Active Suspension System. Vehicle System Dynamics, 2005. **43**(11): p. 795-806.

[29] Samarasinghe, S., Neural Network for Applied Sciences and Engineering. Taylar and Francis Group LLC, 2007.

[30] Yue, L., C. Tang, and H. Li, Research on Vehicle Suspension System Based on Fuzzy Logic Control, in International Conference on Automation and Logistics. 2008: Qingdao, China. p. 1817-1821.

[31] Wilamowski, B.M., et al., An Algorithm for Fast Convergence in Training Neural Networks, in International Joint Conference on Neural Networks. 2001: Washington, DC p. 1778-1782.

[32] Vogl, T.P., et al., Accelerating the convergence of the backpropagation method. Biological Cybernetics, 1988. **59**: p. 257-263.

[33] Tollenaere, T., SuperSAB: Fast adaptive back propagation with good scaling properties. Neural Network, 1990. **3**: p. 561-573.

[34] Rigler, A.K., J.M. Irvine, and T.P. Vogl, Rescaling of variables in back propagation learning. Neural Network, 1991. **4**: p. 225-229.

[35] Peng, C. and D. Magoulas, Adaptive Non monotone Conjugate Gradient Training Algorithm For Recurrent Neural Networks, in 19th IEEE International Conference on Tool with Artificial Intelligence. 2007. p. 374-381.

[36] Ishikamn, T., Y. Tsukui, and M. Matsunami, Optimization of Electromagnetic Device Using Artificial Neural Network with Quasi Newton Algorithm. IEEE Transaction on Magnetics, 1996. **32**(3): p. 1226-1229.

[37] Lera, G. and M. Pinzolas, Neighborhood Based Levenberg Marquardt Algorithm for Neural

Network Training. IEEE Transaction on neural Networks, 2002. **13**(5): p. 1200-1203.

[38] Suratgar, A.A., M.B. Tavakoli, and A. Hoseinabadi. Modified Levenberg Marquardt Method for Neural Networks Training. in Proceedings of World Academy of Science, Engineering and Technology. 2005.

[39] Aldair, A. and W. Wang, Design of Fractional order Controller Based on Evolutionary Algorithm for a Full Vehical Nonlinear Active Suspension System. International journal of Control and Automation (IJCA), 2010. **3**(4): p. 33-46.

[40] Chang, Y., N-Dimension Golden Section Search: Its Variants and Limitations. 2nd International conference on Biomedical Engineering and Informatics, BMEI'09, 2009: p. 1-6.

