


A new control method based on fuzzy controller, time delay estimation, deep learning, and non-dominated sorting genetic algorithm-III for powertrain mount system

Li-Xin Guo¹ and Dinh-Nam Dao² 

Journal of Vibration and Control
2019, Vol. 0(0) 1–12
© The Author(s) 2019
Article reuse guidelines:
sagepub.com/journals-permissions
DOI: 10.1177/1077546319890188
journals.sagepub.com/home/jvc


Abstract

This article presents a new control method based on fuzzy controller, time delay estimation, deep learning, and non-dominated sorting genetic algorithm-III for the nonlinear active mount systems. The proposed method, intelligent adapter fractions proportional–integral–derivative controller, is a smart combination of the time delay estimation control and intelligent fractions proportional–integral–derivative with adaptive control parameters following the speed range of engine rotation via the deep neural network with the optimal non-dominated sorting genetic algorithm-III deep learning algorithm. Besides, we proposed optimal fuzzy logic controller with optimal parameters via particle swarm optimization algorithm to control reciprocal compensation to eliminate errors for intelligent adapter fractions proportional–integral–derivative controller. The control objective is to deal with the classical conflict between minimizing engine vibration impacts on the chassis to increase the ride comfort and keeping the dynamic wheel load small to ensure the ride safety. The results of this control method are compared with that of traditional proportional–integral–derivative controller systems, optimal proportional–integral–derivative controller parameter adjustment using genetic algorithms, linear–quadratic regulator control algorithms, and passive drive system mounts. The results are tested in both time and frequency domains to verify the success of the proposed optimal fuzzy logic controller–intelligent adapter fractions proportional–integral–derivative control system. The results show that the proposed optimal fuzzy logic controller–intelligent adapter fractions proportional–integral–derivative control system of the active engine mount system gives very good results in comfort and softness when riding compared with other controllers.

Keywords

Active mount system, fuzzy control, deep learning, non-dominated sorting genetic algorithm-III, time delay estimation, fractions proportional–integral–derivative controller

1. Introduction

In the automotive industry, researchers are always concerned about passenger comfort, volume, and performance. So, to reduce vehicle vibration or motor while maintaining efficient engine power to ensure the passengers are most comfortable and ensuring a powerful, lightweight, and fuel-efficient engine is the topic that is widely studied today. Some researchers have shown that vibrations in vehicles are caused by disturbances resulting from the unevenness of the road surface with a frequency range below 20 Hz and due to the disturbances arising from the engine during its operation. It is usually in the high range of 20–50 Hz. The engine vibration itself causes vibrations and noise in the passenger cabin (Darsivan et al., 2008; Yang et al., 2001). Thus, the

costs and processing efficiency of the process are also greatly influenced by these vibrations (Nandi et al., 2005). Seidel (1993) has shown that when the human body is subjected to long-term vibrations, it will significantly affect

¹School of Mechanical Engineering and Automation, Northeastern University, China

²Control Technology College, Le Quy Don Technical University, Vietnam

Received: 8 May 2019; accepted: 4 October 2019

Corresponding author:

Dinh-Nam Dao, School of Mechanical Engineering and Automation, Northeastern University, No. 3-11, Wenhua Road, Heping District, Shenyang 110819, China.

Email: daodinhnam@gmail.com

the health, cause a number of diseases such as insomnia, back pain, and muscle/nerve and spinal disease as shown in Figure 1. So, to reduce the vibration of the engine on the chassis, we need to use active engine mount.

The proportional–integral–derivative (PID) controller with simple structure and easy-to-implement control algorithms has been used in various technical and industrial sectors worldwide (Barbosa et al., 2010; Biswas et al., 2009; Jiang et al., 2006). Currently, there are many researchers offering solutions to improve the PID controller to improve control performance. Podlubny (1999) published the fractions proportional–integral–derivative (FOPID) or PI^mD^n controller, where they added two control parameters n and m (integral order and derivative) as fractions. With five control parameters, proportional, integral, derivative gains, the integral, and derivative orders have increased the efficiency and control power compared with the classical PID controller. The biggest advantage of this controller is that it has the ability to eliminate steady-state errors, robustness toward plant uncertainties, and also eliminates good noise. Subsequently, there were a number of researchers offering methods to improve the FOPID controller to improve quality and control accuracy (Copot et al., 2017; De Keyser et al., 2015, 2018; Muresan et al., 2015). Similarly, Aldair and Wang (2010) have published how to apply the optimal algorithm to correct control parameters. This result was verified in the full nonlinear active suspension of the car system. Zhihuan et al. (2014) have also used genetic algorithms (GAs) that are not dominated by II to optimize the

FOPID controller. However, the problem of synchronous control of controllers in the control system is still a problem to be studied.

Recently with the strong development of deep learning techniques, artificial neural networks (ANNs) are used in various fields such as economy, engineering, society, foreign exchange, stock market, etc. (Chan et al., 2013; Hansen et al., 2015; Huang et al., 2014; Leelavathi and Sahana Devi, 2016; Lesinski and Corns, 2018; Montavon, 2012; Shakouri and Banihashemi, 2012; Sreekanth and Datta, 2010; Wang et al., 2016). ANNs with many hidden layers and trained with deep learning algorithms have been used for accurate predictive models. It is very accurate to predict the fixed behavior on the basis of using neural networks in the optimal predictive model of many objectives. Meanwhile, with other methods such as mathematical modeling methods, methods that use statistical model are completely inconsistent with abnormal data patterns that cannot be written clearly in the form of functions or deduced from formulas. But the algorithm using ANNs can be highly effective with chaotic components. Recently, some researchers like Shen et al. (2018), Smith et al. (2014), Kakaee et al. (2015), and Vieira and Tome (2005) used ANNs to predict optimal targets in different areas.

In this article, this is a combination of GA, deep neural network (DNN), and non-dominated sorting genetic algorithm (NSGA)-III to find the best of the pareto-optimal front of the parameter of intelligent adapter fractions proportional–integral–derivative (IAFOPID) controllers according to the engine's rotation speed. This combination gave a faster computing time than when using NSGA-III. On the other hand, this combination also significantly reduced the number of samples needed for the training of DNN. Therefore, it has facilitated the quick application to find the optimal set of parameters of the IAFOPID controller. Besides, we have proposed optimal fuzzy logic controller (OptFLC). Here, control parameters are optimized via particle swarm optimization (PSO) algorithm. These two controllers operate complementary controls. When the IAFOPID controller is operating in error, OptFLC will operate to eliminate this error. OptFLC has the function of tuning to improve the accuracy of the entire system.

2. Structure

2.1. Nonlinear half-car active mount system

Using Newton's law, the mathematical model of Figure 2 can be written as below

$$M\ddot{\mathbf{x}}_i + K\dot{\mathbf{x}}_i + C\mathbf{x}_i = \mathbf{Q}(t) \quad (1)$$

Here, \mathbf{x}_i : vector-column of displacements and angular oscillations of masses; \mathbf{M} : matrix of inertial coefficients of car parts \mathbf{C} : matrix of coefficients of stiffnesses and torsional

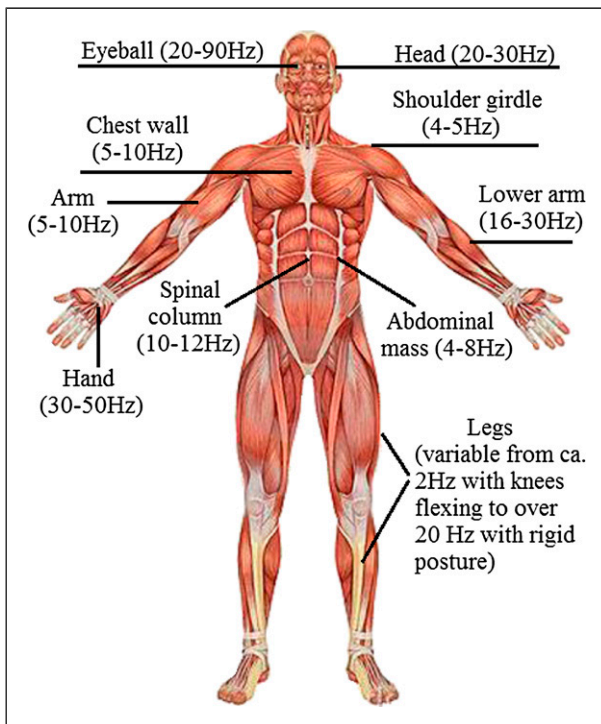


Figure 1. Limits of the body to vibration.

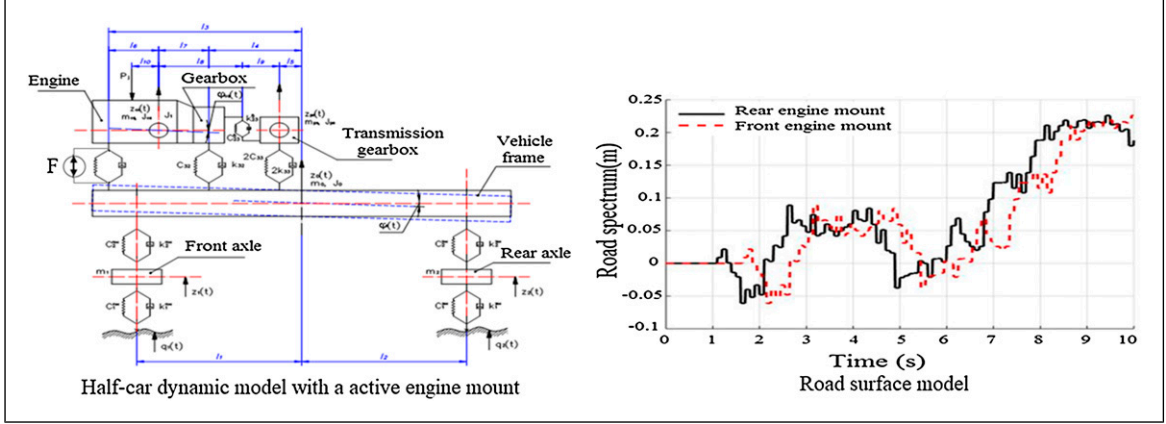


Figure 2. Half-car dynamic and road surface model.

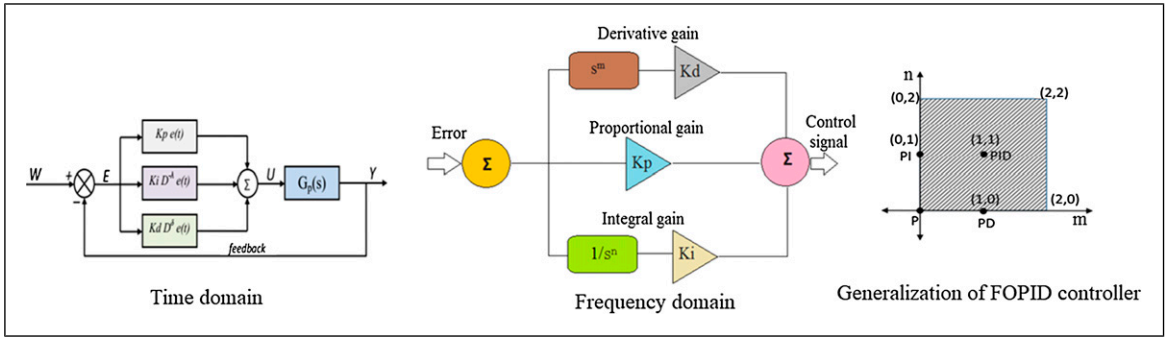


Figure 3. Fractions proportional–integral–derivative controller.

rigidity; \mathbf{K} : matrix of damping coefficients; $\mathbf{Q}(t)$: column vector of the perturbing forces and moments. $q_2(t) = q_1(t + \tau)$ with τ : time interval, via vehicle speed

$$\mathbf{Q}(t) = \begin{bmatrix} 2k_1^{\text{IM}} \cdot \dot{q}_1(t) + 2C_1^{\text{IM}} \cdot q_1(t), 2k_2^{\text{IM}} \cdot \dot{q}_2(t) \\ + 2C_2^{\text{IM}} \cdot q_2(t), 0, 0, P_j(t), P_j(t) \cdot l_{10}, 0, 0, 0, \\ M_{\text{fl}}(t), 0, 0, 0, 0, \gamma \cdot [2 \cdot k_1^{\text{IM}} \cdot \dot{q}_1(t) + 2 \cdot C_1^{\text{IM}} \cdot q_1(t)], \\ \gamma \cdot [2 \cdot k_2^{\text{IM}} \cdot \dot{q}_2(t) + 2 \cdot C_2^{\text{IM}} \cdot q_2(t)], 0, 0, 0 \end{bmatrix}^T \quad (2)$$

$$\mathbf{X} = \begin{bmatrix} z_1, z_2, z_0, \varphi_0^y, z_{ca}, \varphi_{ca}^y, z_{pk}, \varphi_{ca}^x, \varphi_{pk}^x \\ \varphi_1, \varphi_2, \varphi_3, \varphi_4, \varphi_5, \varphi_6, \varphi_7, \varphi_8, \varphi_{\text{IM}}, \varphi_{3M} \end{bmatrix}^T \quad (3)$$

2.2. Road features

To be general, we use random functions as a function of the pavement as shown in Figure 3 and simulation parameters as shown in Tables 1–3.

2.3. Control system design

In this section, we have described the fundamental principles of OptFLC–IAFOPID control system. It is the smooth coordination between OptFLC and IAFOPID

Table 1. Geometric parameters of engine (m).

l_1	l_2	l_3	l_4	l_5
1.225	1.175	1.330	0.520	0.190
l_6	l_7	l_8	l_9	l_{10}
0.187	0.623	0.760	0.210	0.030

Table 2. General settings information.

Mass of the equipped automobile, m_0 (kg)	1210
Payload (kg)	400
The weight of the front wheels, m_1 (kg)	37
The weight of the rear wheels, m_2 (kg)	37
The weight of the power unit, m_{ca} (kg)	152.2
Transfer case weight, m_{pk} (kg)	27.8
Radius crank, r (m)	0.04
The ratio of the crank radius to the length of the connecting rod, λ	0.308
Wheel radius in slave mode, r_k (m)	0.325

controller. Inside, IAFOPID controller is the smart combination of the time delay estimation control (TDEC) and IAFOPID with adaptive control parameters following the speed range of engine rotation via the DNN with the optimal

Table 3. Model parameters of non-dominated sorting genetic algorithm and deep neural network.

Maximum number of iterations	Iter_max = 1000	Mutation rate	0.02
Population size	100	Number of parents (offsprings)	$2 \times \text{round}(p\text{Crossover} \times n\text{Pop}/2)$
Crossover percentage	0.5	Number of mutants	$\text{round}(p\text{Mutation} \times n\text{Pop})$
Mutation percentage	0.5	Mutation step size	$0.1(\text{VarMax} - \text{VarMin})$
Deep neural network parameters	nn = [100 200 200 200 200 100]	Generating reference points	nDivision = 10; Zr = GenerateReferencePoints (nObj, nDivision)

NSGA-III deep learning algorithm. The OptFLC is the optimal fuzzy logic control in which the parameters of OptFLC are optimized via PSO algorithm. This controller does a lot of fine tuning, eliminating errors in the control process of IAFOPID controller. Therefore, the accuracy of the control system is enhanced, with better performance and higher reliability.

2.3.1. TDE controller. Dong-Ji et al. (2007) show the non-linear dynamics change over time

$$\dot{x}(t) = f(x(t)) + h(x(t), t, w(t)) + u(t) \quad (4)$$

Here, $x(t)$ the state variables; $u(t)$ control vector; $f(x(t))$ known dynamics; $w(t)$ disturbance input; $h(x(t), t, w(t))$ unknown dynamics.

TDE controller provides a simple and effective solution (Hsia et al., 1991).

In equation (4), equation (5) can be substituted as

$$\dot{x}(t) + \Psi(x, t) = u(t) \quad (5)$$

Here

$$\Psi(x, t) = (-f(x(t)) - h(x(t), t, w(t))) \quad (6)$$

Control request is to achieve the output according to the reference input orbit x_{ref}^* .

Here, we define $e = x_{\text{ref}}^*(t) + y(t)$, $\dot{e} = \dot{x}_{\text{ref}}^*(t) + \dot{y}(t)$, and $\ddot{e} = \ddot{x}_{\text{ref}}^*(t) + \ddot{y}(t)$.

Here, dynamic error is defined as

$$\ddot{e} + \Lambda_d \dot{e} + \Lambda_p e = 0 \quad (7)$$

with Λ_d and Λ_p as constant gains, the control input can be selected as

$$u(t) = u_0 + \hat{\Psi}(x, t) \quad (8)$$

$$u_0 = \ddot{x}_{\text{ref}}^* + \Lambda_d \dot{e} + \Lambda_p e \quad (9)$$

Here, $\hat{\Psi}(x, t)$ is an estimated value by using TDE (Fliess and Join, 2009) as

$$\hat{\Psi}(x, t) = \Psi(x, t - L) \quad (10)$$

where L is the estimated time delay. From equation (8), we can obtain

$$\hat{\Psi}(x, t) = u(t - L) - \dot{x}(t - L) \quad (11)$$

Therefore, from equations (8)–(11), the TDEC law is expressed as

$$u(t) = u(t - L) - \dot{x}(t - L) + \left(\dot{x}_{\text{ref}}^* + \Lambda_d \dot{e} + \Lambda_p e \right) \quad (12)$$

2.3.2. IAFOPID controller. FOPID controller consists of five parameters (n , m , K_p , K_i , and K_d) that need to be optimized according to the control requirements as in Figure 3 so that more details can be referenced (Das et al., 2011)

$$U(t) = K_p e(t) + K_i D^{-n} e(t) + K_d D^m e(t) \quad (13)$$

Equation (13) is rewritten as follows

$$U(s) = K_p e(s) + K_i D^{-n} e(s) + K_d D^m e(s) \quad (14)$$

2.3.2.1. Combination of two FOPID controllers with TDE controller. It has been shown as an unknown non-linear model (Fliess and Join, 2009; Gédouin et al., 2011). It is defined as follows

$$y^{(v)}(t) = \Psi(t) + \Gamma u(t) \quad (15)$$

where $y(t)$ is the powertrain acceleration as output signal, v is the order derivative, which is usually selected as 1 or 2, and $\Psi(t)$ is an unknown term which estimated via the control input $u(t)$ and output $y(t)$. In addition, $\Psi(t)$ not only denotes the structure of the unknown system but also the noise, when the Γ is the constant and $u(t)$ are input signals. Figure 4 illustrates the model of IAFOPID controller. It consists of three controllers that are parallel to each other. They are acceleration FOPID, displacement FOPID, and TDE controllers.

The control input of IAFOPID is defined as

$$G_{AD} = \left(K_{pA} e(t) + K_{dA} D^{-n_A} \dot{e}(t) + K_{iA} D^{m_A} \int e(t) \right) + \left(K_{pD} e(t) + K_{dD} D^{-n_D} \dot{e}(t) + K_{iD} D^{m_D} \int e(t) \right) \quad (16)$$

where K_{pA} , K_{iA} , K_{dA} , n_A , and m_A are the equivalent classical proportional integral derivative (acceleration FOPID) coefficients, and K_{pD} , K_{iD} , K_{dD} , n_D , and m_D are the equivalent classical PID (displacement FOPID) coefficients.

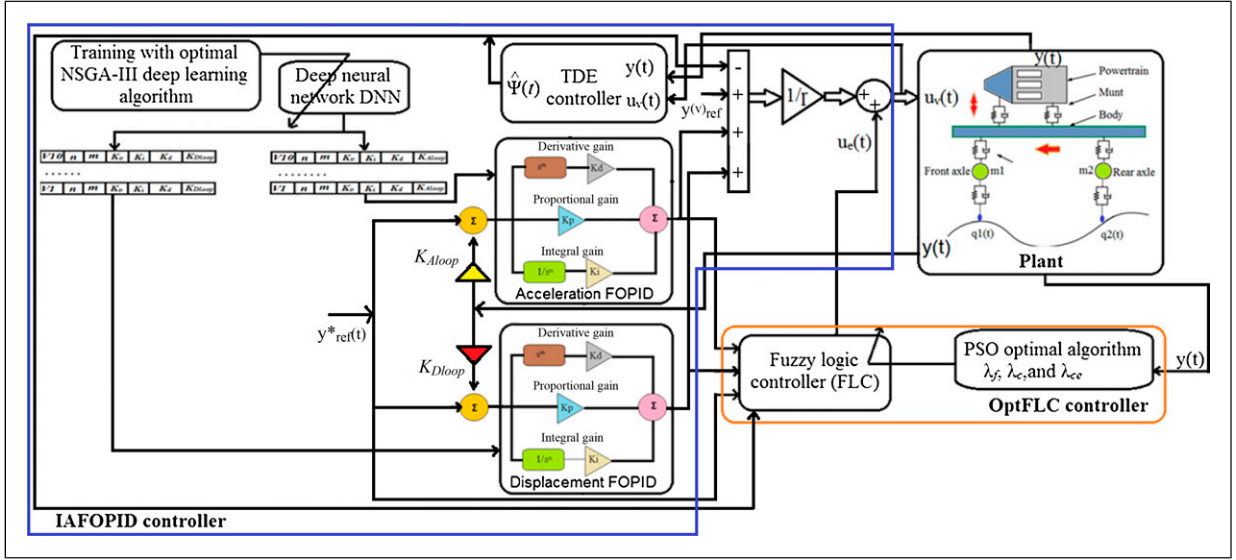


Figure 4. Diagram of the control system.

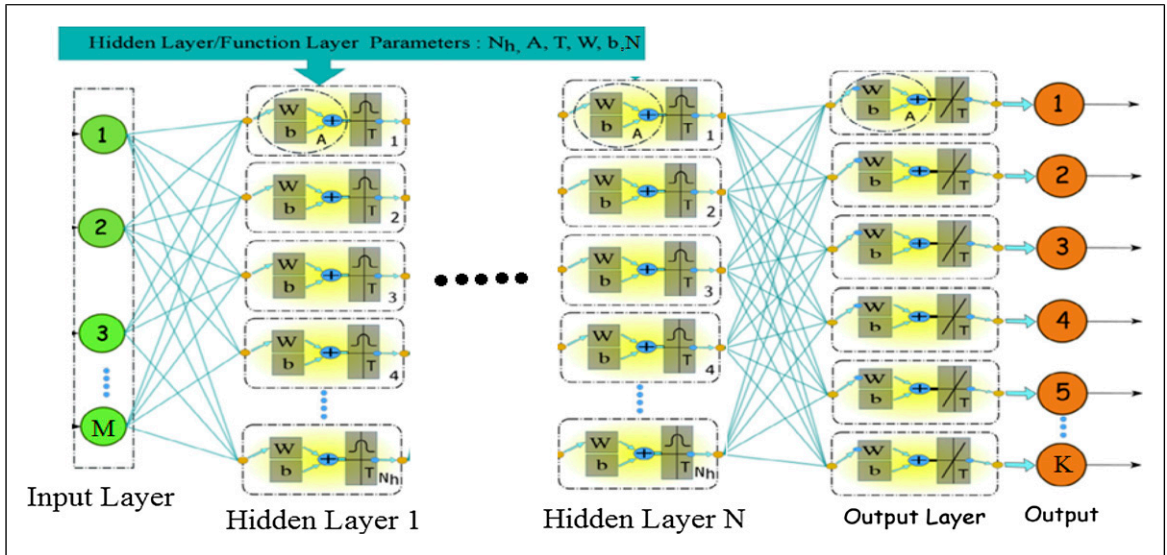


Figure 5. Deep neural network.

Equation (16) can be written in the following equivalent form

$$G_{AD} = K_p e(t) + K_d D^{-n} \dot{e}(t) + K_i D^m \int e(t) \quad (17)$$

$$u_v(t) = \frac{1}{\Gamma} \left[\left(K_p e(t) + K_d D^{-n} \dot{e}(t) + K_i D^m \int e(t) \right) + \dot{y}_{ref}^*(t) - \hat{\Psi}(t) \right] \quad (18)$$

\dot{y}_{ref}^* described the output reference trajectory, $\hat{\Psi}(t)$ described estimated value of $\Psi(t)$, and $e(t) = \dot{y}_{ref}^*(t) - y(t)$ described

the output error. Substituting equation (9) into equation (8), we get the following result

$$\dot{e}(t) + K_p e(t) + K_d D^{-n} \dot{e}(t) + K_i D^m \int e(t) + \Psi(t) - \hat{\Psi}(t) = 0 \quad (19)$$

2.3.2.2. DNN with the NSGA-III optimal deep-learning algorithm. We have selected the optimal parameters for IAFOPID controller so that the error in the nonlinear control is minimal. We have researched, analyzed, and decided to create the optimal parameter set according to the speed range of the engine. We use neural networks with linear transmission structure (multilayer perceptron neural network (MLP)). This is the deep neuron network with

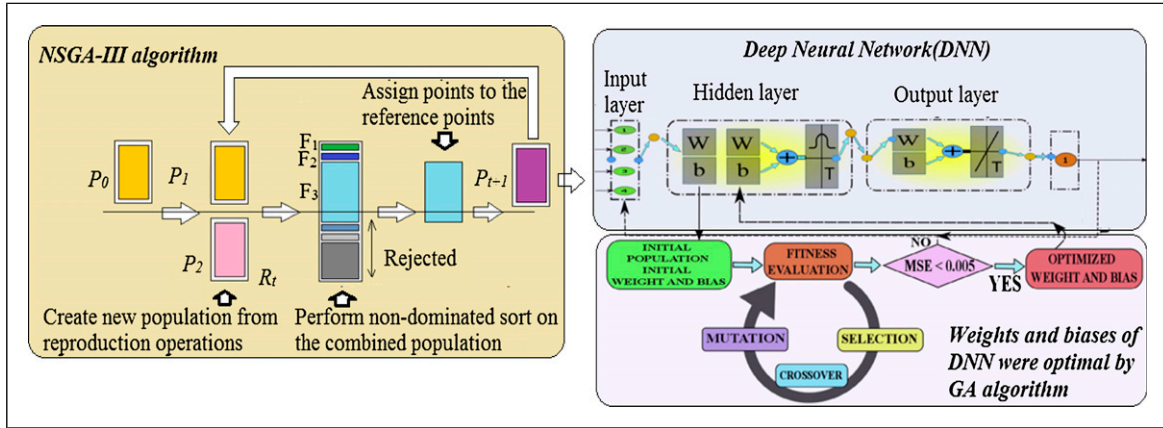


Figure 6. Training structure for hybrid method.

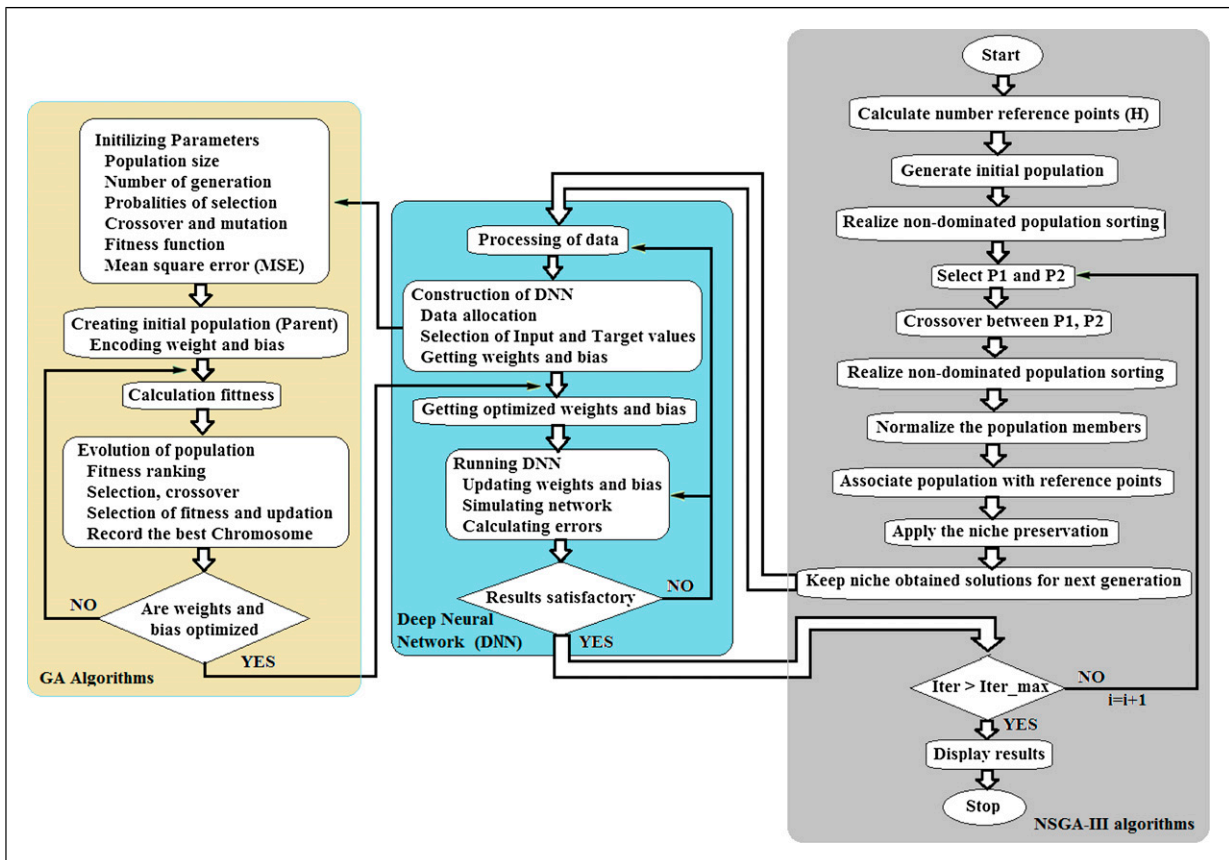


Figure 7. Create hierarchical deep neural network and non-dominated sorting genetic algorithm.

four hidden layers to learn the optimal algorithm NSGA-III. A specific model of MLP network is set up with: $nn = [M = 100; Nh1 = 200; Nh2 = 200; Nh3 = 200; Nh4 = 200; K = 100]$. Here, M is the input neuron and $i = 4$ is the number of hidden classes with each hidden layer having neuron Nhi and output layer having K the neuron is described in Figure 5.

The problem of training ANNs is an important issue. To optimize the weights and deviations of neural networks, it is often used as a GA (Filho and Filho, 2010). In the process of training the network, the weights and bias of the DNN adjust through GA. Thus, the exchange interaction between the GA and DNN is the process of exchanging weights and bias. The weight and biases $[W, b]$ of the DNN

are started with a random value as shown in Figure 6. Similarly, in the GA, the initial population is also started randomly. Next, the population of the next generation is algorithm GA creating melon on the current population. An exercise function is used to evaluate the difference between the predicted output values and the actual output value. The result is obtained when the average square of GA is less than 0.005. Weight and biases are expressed through

$$N_w = (I_n + 1)N_h + (N_h + 1)O_p \quad (20)$$

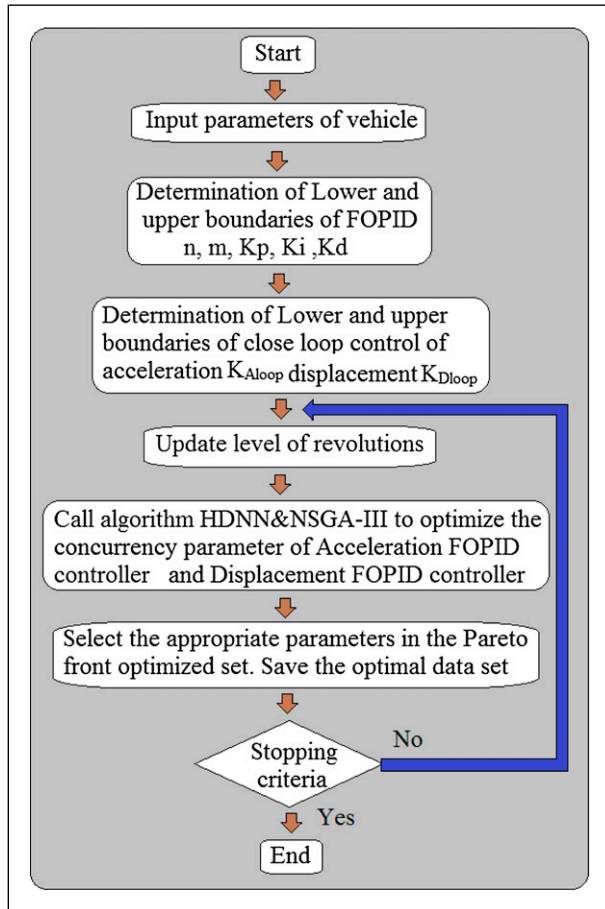


Figure 8. Flow chart of optimal parameter of cascade fractions proportional-integral-derivative controller.

Here, number of weights and biases is N_w , number of neurons in the input layer is I_n , number hidden layer is N_h , and number of neurons in the output layer is O_p . Optimal GA value has been achieved through the training process. Simulation parameters of the GA are as follows:

Population size: 20	Mutation rate: 0.15
Crossover rate: 0.65	Number of generations: 250

Deep learning method to create optimal hierarchical deep neural network (HDNN) and NSGA-III. Because the NSGA-III is a sequential computation process and is repeated many times through loops, the calculation time will be very slow. Meanwhile with ANN, the calculation process is conducted in parallel and dispersed over many nerve cells almost simultaneously. This calculation is a learning process, not according to the previous plan. So the calculation speed is very fast and the DNN can learn the characteristics of the NSGA-III. We have proposed HDNN algorithm and NSGA-III. This is the DNN that has received optimal deep learning training according to the NSGA-III, as shown in Figures 6 and 7. Thus, this neural network has the same function as the NSGA-III gene algorithm, but with many times higher accuracy and faster calculation time.

Method of turning the IAFOPID controller. To increase the accuracy and fast response time, we need to find the data set $(n, m, K_p, K_i, K_d, \text{ and } K_{Aloop})$ of acceleration controllers and data set $(n, m, K_p, K_i, K_d, \text{ and } K_{Dloop})$ of displacements controllers, which are optimal with engine rotation speed range. To match the vehicle's performance characteristics such as sudden acceleration and deceleration and the controller's processing time, we have divided the engine rotation speed range (700 r/min–6000 r/min) into 500 revolutions in each level. Each level will be used to find the optimal set of parameters for the control system. Algorithm diagram calculates the optimal coefficients varying according to the speed range to be included in the IAFOPID controller as shown in Figures 8 and 9.

2.3.3. OptFLC. Because the IAFOPID controller simultaneously controls both acceleration and displacement, it is

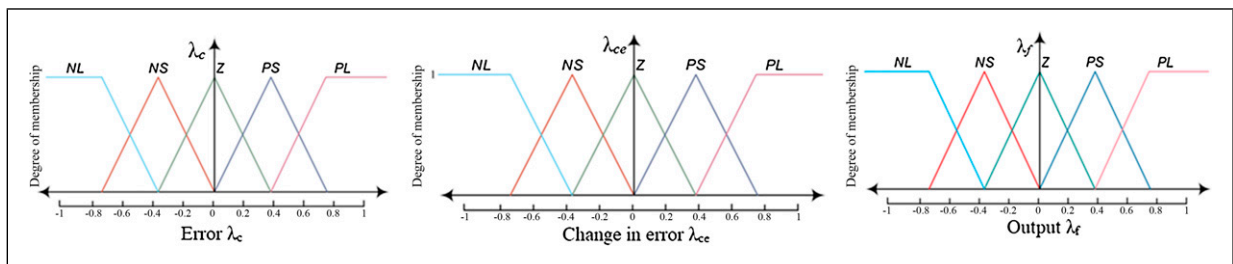


Figure 9. Membership functions.

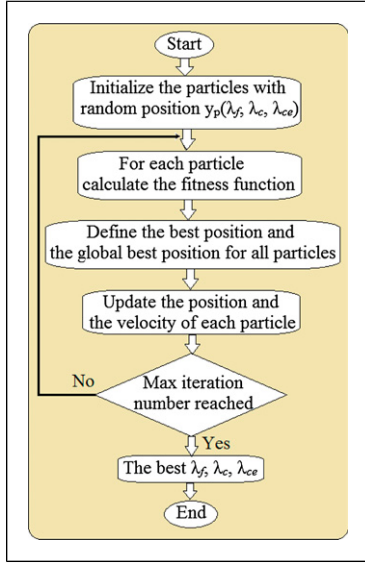


Figure 10. Particle swarm optimization algorithm.

Table 4. The rule of fuzzy control.

e/c_e	NS	NS	Z	PL	PS
NS	NS	NL	NL	PS	PL
NL	Z	NS	Z	PL	Z
PL	PS	PL	PS	PS	NS
Z	Z	PS	Z	NS	NL
PS	PL	PS	NL	NS	NL

easy to generate errors. Although this controller has a TDE controller, it only reduces the error. Therefore, it is necessary to add OptFLC to eliminate the remaining errors in the system. The OptFLC is proposed as shown in Figure 4 and can be defined as follows

$$u_v(t) = \frac{1}{\Gamma} \left[\left(K_p e(t) + K_d D^{-n} \dot{e}(t) + K_i D^m \int e(t) \right) + y_{\text{ref}}^*(t) - \hat{\Psi}(t) \right] + u_e(t) \quad (21)$$

Here, the extra input of the OptFLC is $u_e(t)$. After substituting equation (21) into equation (14), the new closed-loop error can be obtained as below

$$\dot{e}(t) + K_p e(t) + K_d D^{-n} \dot{e}(t) + K_i D^m \int e(t) + \Gamma u_e + \Psi_m = 0 \quad (22)$$

Here, the estimation error is $\Psi_m = \Psi(t) - \hat{\Psi}(t)$. OptFLC has an easy to implement, highly generalized advantage. This controller has the role of overcoming errors that can be generated by the controller IAFOPID. Controller output OptFLC can be described as follows

$$u_e(t) = \lambda_f \phi_f(\lambda_c e, \lambda_{ce} c e) \quad (23)$$

where λ_f is the output, λ_c is the error, and λ_{ce} is the change of error gains, and the optimal of the OptFLC is dependent on these gains (Ha, 1998), where a fuzzy nonlinear function is ϕ_f . To generalize the control law, we use five language variables NB, NS, Z, PS, and PB (i.e., NB, NS, PB, PS, and Z labels denote “big vibration,” “small vibration,” “extreme vibration,” “extremely small vibration,” and “no,” respectively); corresponding triangular member functions selected to represent linguistic variables are shown in Figure 10.

Based on the literature (Hasanien and Matar, 2015; Mustafa et al., 2013), we have shown the method of deduction and opacity. Equation (23) can be substituted for equation (21) as follows

$$u_v(t) = \frac{1}{\Gamma} \left[\left(K_p e(t) + K_d D^{-n} \dot{e}(t) + K_i D^m \int e(t) \right) + y_{\text{ref}}^*(t) - \hat{\Psi}(t) \right] + \lambda_f \phi_f(\lambda_c e, \lambda_{ce} c e) \quad (24)$$

This algorithm is controlled according to control rules as in Table 4.

2.3.3.1. OptFLC optimal parameter with PSO algorithm. PSO algorithm (Babu et al., 2018) is to determine the optimal parameters of the logic fuzzy controller. This is to determine how to select the optimal values λ_f , λ_c , and λ_{ce} . The PSO stages are summarized in Figure 10.

3. Simulation results and discussion

3.1. Results

Simulation results from MATLAB software and optimal data set of IAFOPID controller are shown in Table 5. Simulation analysis is performed in this section to evaluate the ride comfort.

Figure 11 shows the results in the form of four-dimensional space via the isosurfaces function in MATLAB. This is the result of a case of a rotation speed range at 1700–2200 r/min as shown in Table 5. In the figure, pareto-optimization front is set according to the parameters of two accelerator and displacement controllers. Based on the need to rejuvenate comfort and softness, we have selected the values that correspond to the mean square acceleration, and mean square displacement of the powertrain mount system is the smallest in this pareto front set (these are the gray circles with the bold net).

Figure 12 shows the result of a case of a rotation speed range at 1700–2200 r/min. The black points in the figure show the results of calculating the acceleration and displacement values of magnetorheological (MR) engine mount corresponding to the stiffness values of the mount to be controlled. This value is given by the OptFLC–IAFOPID control system to control actual MR damper force.

Table 5. Intelligent adapter fractions proportional–integral–derivative controller optimal parameter set.

Revolutions level (r/min)	Acceleration FOPID controller						Displacement FOPID controller					
	n_A	m_A	K_{pA}	K_{iA}	K_{dA}	K_{Dloop}	n_D	m_D	K_{pD}	K_{iD}	K_{dD}	K_{Aloop}
700–1200	0.53	0.24	56,422.8	25,655.2	7522.1	2.60	0.38	0.36	67,553.7	12,450.6	4526.0	3.78
1200–1700	0.59	0.65	61,332.3	30,765.8	3349.5	1.85	1.74	0.53	62,424.1	67,358.1	5472.8	2.03
1700–2200	0.55	0.29	51,664.6	41,536.4	4155.7	1.39	2.36	0.68	31,352.3	35,063.8	4459.5	3.67
2200–2700	0.74	0.46	45,462.3	54,350.3	4362.2	6.32	2.45	0.35	57,562.6	44,385.4	4355.7	3.33
2700–3200	0.76	0.56	56,522.5	32,405.3	4503.4	5.68	5.76	0.65	79,562.8	24,695.2	4352.1	5.68
3200–3700	0.45	0.43	77,642.9	45,710.7	4432.7	2.50	4.65	0.42	61,644.8	45,150.8	5239.3	5.80
3700–4200	0.57	0.24	62,424.1	56,928.1	4562.8	5.78	0.45	0.43	53,570.2	56,582.9	5342.3	3.64
4200–4700	0.83	0.65	58,433.2	43,732.9	4262.3	6.46	0.65	2.35	45,672.5	35,475.3	6473.4	5.44
4700–5200	0.76	0.255	76,655.7	23,450.6	4520.0	5.67	0.38	4.36	73,512.9	57,699.7	5532.7	2.64
5700–6000	0.55	0.24	81,454.8	45,153.8	7539.3	3.85	2.54	2.54	52,362.3	31,380.3	3862.2	4.84

FOPID: fractions proportional–integral–derivative.

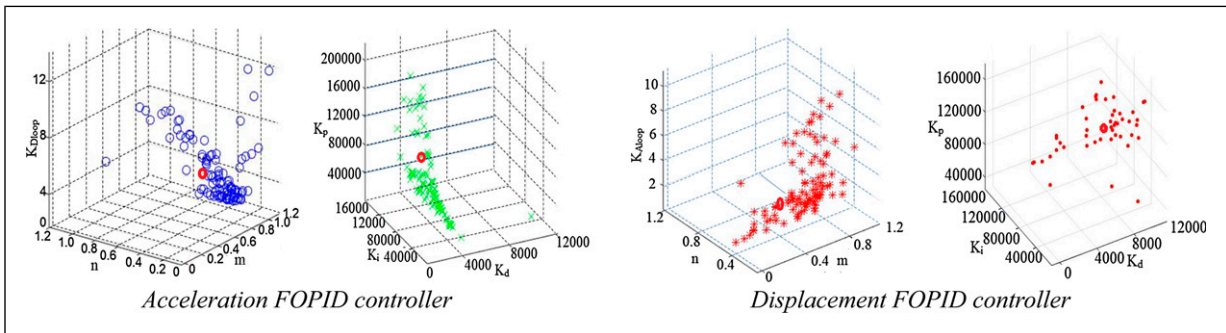


Figure 11. Optimal parameter intelligent adapter fractions proportional–integral–derivative controller at a speed range of 1700–2200 r/min.

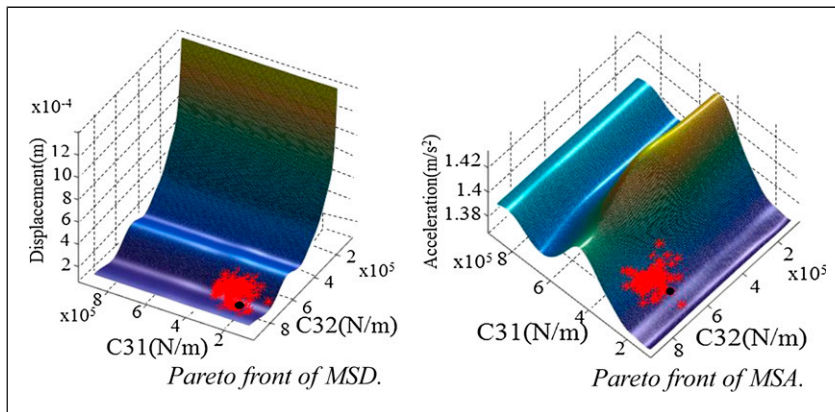


Figure 12. Global pareto front of mean square displacement and mean square acceleration front engine mount.

The result of method OptFLC–IAFOPID is compared with the results using the LQR algorithm in Shen et al. (2018). In Figure 13, we see that the effect of the direction is very good, and the displacement and rotation angle according to the engine speed range are lower than those of the LQR method.

In Figure 14, with the OptFLC–IAFOPID control system, we see that the transmitted force active mount and acceleration response of the chassis are the smallest compared with other controllers. Thus, the OptFLC–IAFOPID control system has superior advantages over other controllers. It has eliminated high-frequency noise,

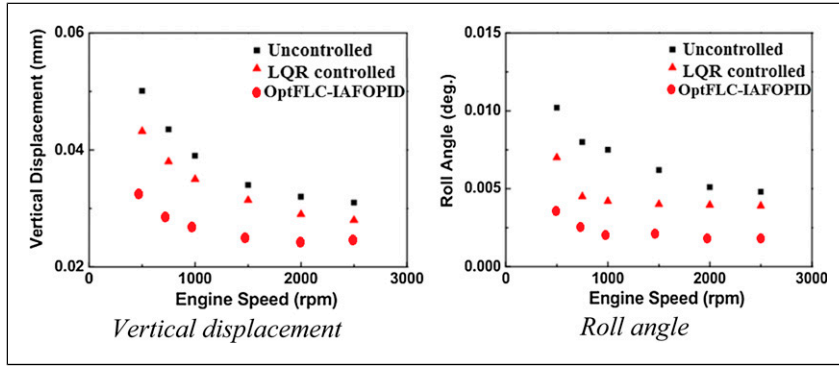


Figure 13. Frequency response at CG.

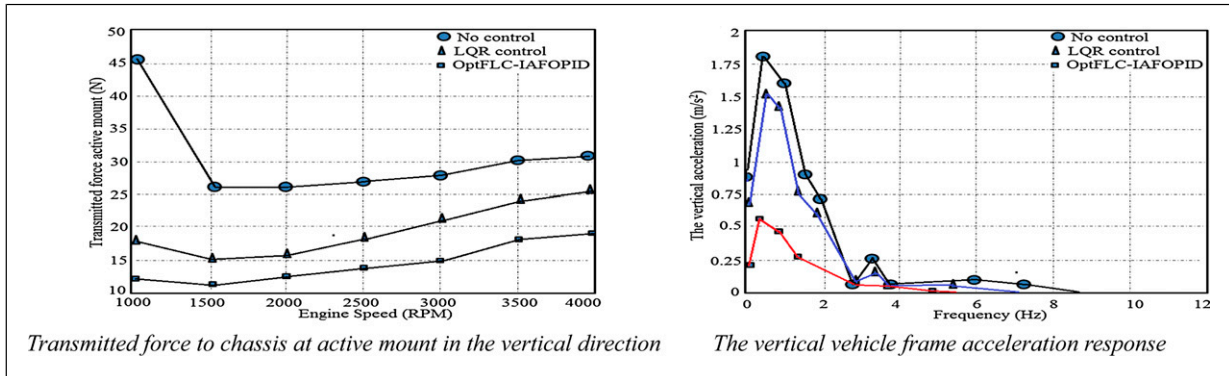


Figure 14. Transmitted force to chassis and vehicle frame acceleration response.

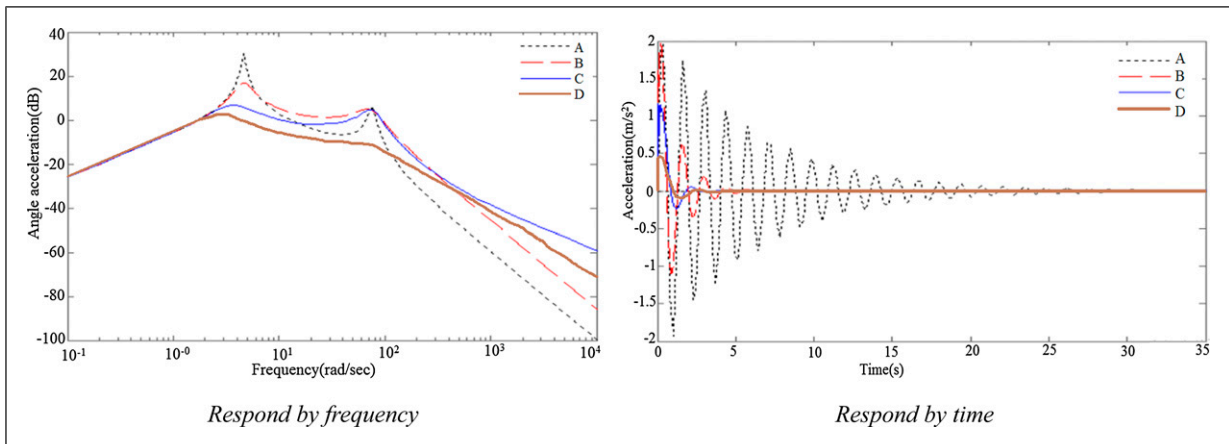


Figure 15. Response by frequency and time.

preventing the sudden acceleration and deceleration of the vehicle.

In Figure 15, the passive drive system mounts is curve A, traditional PID systems is B, optimal PID parameter adjustment using GAs is C, and OptFLC-IAFOPID control system is D.

In the frequency domain, there are two resonant points, and the OptFLC-IAFOPID control system has a lower oscillation transmission than the other three systems. This

proves that the OptFLC-IAFOPID controller is the best. In the time domain itself, the minimum oscillation amplitude is achieved at the OptFLC-IAFOPID controller, and the fastest time to turn off oscillation is also achieved due to OptFLC-IAFOPID controller. Through two frequency and time zones, we can conclude that the OptFLC-IAFOPID controller has the best control, and it controls the sudden nonlinear vibrations.

4. Conclusion

This article has published OptFLC–IAFOPID control system and control algorithms. This control system has an optimal set of control parameters that adapt to the changing speed range of the engine rotation. OptFLC–IAFOPID control system has been designed successfully. It has demonstrated superior advantages over previous controllers. On the other hand, it has met the requirements set out to prevent vibrations during sudden increase or decrease when the engine changes different rotation speeds, thus creating a comfortable and soft feeling and roof for people when riding. This is completely proven in the simulation results.

Besides, this article has published a new combination method between NSGA-III, DNN, and GA. So DNNs with many hidden layers are trained with intelligent deep learning algorithms, and it has created a deep learning network to realize the optimal problem simultaneously of many objects in the technology. This method is extremely effective and highly practical because first, it offers a way of deep learning of the network with a much smaller number of standard samples than previous deep-learning networking methods announced. For test cases, optimal parameter IAFOPID controller, we only need a standard set of 10 samples to train DNNs. It is this combination of training that the actual number of samples generated for the training process is $10 \times (\text{Iter_max}) = 10,000$ samples. However, training with the old method is extremely difficult to collect a large number of samples. Second, for test cases, optimal parameter IAFOPID controller, the time for multi-objective optimal analysis of this hybrid method at one revolutions level (r/min) is only 1.2 h, whereas using NSGA-III takes 32 h. So, the total time for calculating the entire process for the optimal controller will be $(32 - 1.2) \times 10 = 308$ h.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported by the National Natural Science Foundation of China (51875096, 51275082).

ORCID iD

Dinh-Nam Dao  <https://orcid.org/0000-0001-9810-2096>

References

- Aldair AA and Wang JW (2010) Design of fractional order controller based on evolutionary algorithm for a full vehicle

- nonlinear active suspension systems. *International Journal of Control and Automation* 3: 33–46.
- Babu TS, Ram JP, Dragicevic T, et al. (2018) Particle swarm optimization based solar PV array reconfiguration of the maximum power extraction under partial shading conditions. *IEEE Transactions on Sustainable Energy* 9: 74–85.
- Barbosa RS, Tenreiro Machado JA and Jesus IS (2010) Effect of fractional orders in the velocity control of a servo system. *Computers & Mathematics with Applications* 59: 1679–1686.
- Biswas A, Das S, Abraham A, et al. (2009) Design of fractional-order PI^2D^{μ} controllers with an improved differential evolution. *Engineering Applications of Artificial Intelligence* 22: 343–350.
- Chan KY, Dillon T, Chang E, et al. (2013) Prediction of short-term traffic variables using intelligent swarm-based neural networks. *IEEE Transactions on Control Systems Technology* 21(1): 263–274.
- Copot C, Ionescu C, Vanlanduit S, et al. (2017) Vibration suppression in multi-body systems by means of disturbance filter design methods. *Journal of Vibration and Control* 24(14): 2957–2969.
- Darsivan FJ, Faris WF and Martono W (2008) Active engine mounting controller using extended minimal resource allocating networks. *International Journal of Vehicle Noise and Vibration* 4(2): 150–168.
- Das S, Saha S, Das S, et al. (2011) On the selection of tuning methodology of FOPID controllers for the control of higher order processes. *ISA Transactions* 50: 376–388.
- De Keyser R, Copot C, Hernandez A, et al. (2015) Discrete-time internal model control with disturbance and vibration rejection. *Journal of Vibration and Control* 23(1): 3–15.
- De Keyser R, Muresan CI, Ionescu CM, et al. (2018) An efficient algorithm for low-order direct discrete-time implementation of fractional order transfer functions. *ISA Transactions* 74: 229–238.
- Dong-Ji X, Jin-Wan K, Joung-Il Z, et al. (2007) A study on active suspension system using time delay control. In: International conference on control, automation and systems, Seoul, South Korea, 17–20 October 2007. ■■■: IEEE.
- Filho ACP and Filho RM (2010) Hybrid training approach for artificial neural networks using genetic algorithms for rate of reaction estimation: application to industrial methanol oxidation to formaldehyde on silver catalyst. *Chemical Engineering Science* 157: 501–508.
- Fliess M and Join C (2009) Model-free control and intelligent PID controllers: towards a possible trivialization of nonlinear control? *IFAC Proceedings Volumes* 42: 1531–1550.
- GédouinJoin P-A, Delaleau E, Bourgeot J-M, et al. (2011) Experimental comparison of classical PID and model-free control: position control of a shape memory alloy active spring. *Control Engineering Practice* 19: 433–441.
- Ha QP (1998) A fuzzy sliding mode controller for power system load-frequency control. In: Second international conference. Knowledge-based intelligent electronic systems. Proceedings KES'98, Adelaide, Australia, 21–23 April 1998. ■■■: IEEE.
- Hansen K, Biegler F, Ramakrishnan R, et al. (2015) Machine learning predictions of molecular properties: accurate many-body potentials and nonlocality in chemical space. *The Journal of Physical Chemistry Letters* 6: 2326–2331.
- Hasanien HM and Matar M (2015) A fuzzy logic controller for autonomous operation of a voltage source converter-based

- distributed generation system. *IEEE Transactions on Smart Grid* 6: 158–165.
- Hsia TCS, Lasky TA and Guo Z (1991) Robust independent joint controller design for industrial robot manipulators. *IEEE Transactions on Industrial Electronics* 38: 21–25.
- Huang W, Song G, Hong H, et al. (2014) Deep architecture for traffic flow prediction: deep belief networks with multitask learning. *IEEE Transactions on Intelligent Transportation Systems* 15(5): 2191–2201.
- Jiang C, Ma Y and Wang C (2006) PID controller parameters optimization of hydro-turbine governing systems using deterministic-chaotic-mutation evolutionary programming (DCMEP). *Energy Conversion and Management* 47: 1222–1230.
- Kakaee A, Rahnama P, Paykani A, et al. (2015) Combining artificial neural network and multi-objective optimization to reduce a heavy-duty diesel engine emissions and fuel consumption. *Journal of Central South University* 22(11): 4235.
- Leelavathi MV and Sahana Devi KJ (2016) An architecture of deep learning method to predict traffic flow in big data. *International Journal of Research in Engineering and Technology* 5(16): 461–468.
- Lesinski G and Corns S (2018) Multi-objective evolutionary neural network to predict graduation success at the United States military academy. *Procedia Computer Science* 140: 196–205.
- Montavon G (2012) Learning invariant representations of molecules for atomization energy prediction. In: Pereira F, Burges CJC, Bottou L and Weinberger KQ (eds) *Advances in Neural Information Processing Systems*. New York: Curran Associates, Inc., 440–448.
- Muresan CI, Dulf EH, Copot C, et al. (2015) Design and analysis of a multivariable fractional order controller for a non-minimum phase system. *Journal of Vibration and Control* 22(9): 2187–2195.
- Mustafa GIY, Ali AT, Bashier E, et al. (2013) Neuro-fuzzy controller design for a DC motor drive. *University of Khartoum Engineering Journal* 3: 7–11.
- Nandi S, Toliyat HA and Li X (2005) Condition monitoring and fault diagnosis of electrical motors—a review. *IEEE Transactions on Energy Conversion* 20(4): 719–729.
- Podlubny I (1999) Fractional-order systems and PI^2D^α -controllers. *IEEE Transactions on Automatic Control* 44: 208–214.
- Seidel H (1993) Selected health risks caused by long-term, whole-body vibration. *American Journal of Industrial Medicine* 23(4): 589–604.
- Shakouri HM and Banihashemi S (2012) Developing an empirical predictive energy-rating model for windows by using artificial neural network. *International Journal of Green Energy* 16: 961–970.
- Shen Y, Wang X and Chen J (2018) Wind power forecasting using multi-objective evolutionary algorithms for wavelet neural network-optimized prediction intervals. *Applied Science* 8: 185.
- Smith C, Doherty J and Jin Y (2014) Multi-objective evolutionary recurrent neural network ensemble for prediction of computational fluid dynamic simulations. In: IEEE congress on evolutionary computation (CEC), Beijing, China, 6–11 July 2014. ■■■: IEEE.
- Sreekanth J and Datta B (2010) Multi-objective management of saltwater intrusion in coastal aquifers using genetic programming and modular neural network based surrogate models. *Journal of Hydrology* 393(3–4): 245–256.
- Vieira A and Tome RS (2005) A multi-objective evolutionary algorithm using neural networks to approximate fitness. *International Journal of Computers, Systems and Signals* 6(1): 18–36.
- Wang J, Fang W and Niu H (2016) Financial time series prediction using Elman recurrent random neural networks. *Computational Intelligence and Neuroscience* 2016: 4742515.
- Yang J, Suematsu Y and Kang Z (2001) Two-degree-of-freedom controller to reduce the vibration of vehicle engine-body system. *IEEE Transactions on Control Systems Technology* 9(2): 295–304.
- Zhihuan C, Xiaohui Y, Bin J, et al. (2014) Design of a fractional order PID controller for hydraulic turbine regulating system using chaotic non-dominated sorting genetic algorithm II. *Energy Conversion Management* 84: 390–404.