

Control Strategy for the DC-DC Converter in Fuel Cell Vehicles.

Xinran Liu^{1,a}, Xiaorong Huang², Yiqiang Peng³, Liting Bai⁴

¹Chengdu Jincheng college Department of Technology Manugacting, Chengdu 611731, China

²Key Laboratory of Automotive Measurement, Control and Safety, XiHua University, Chengdu 610039, China

³Key Laboratory of Automotive Measurement, Control and Safety, XiHua University, Chengdu 610039, China

⁴Chengdu Jincheng college Department of Technology Manugacting, Chengdu 611731, China

^a15351222492@163.com

Abstract: A BP neural network PID control strategy optimized based on a genetic algorithm is proposed to meet the requirements of a wide buck-boost range, low input current ripple, and a wide input voltage range for fuel cell vehicle buck-boost converters. By leveraging the global optimization capability of the genetic algorithm, the optimal initial weights of the BP neural network are calculated, avoiding the issue of local optima caused by dependence on initial weights during the BP neural network iteration process. Simulation experiments were conducted in MATLAB/Simulink, and the results show that compared with the traditional PID control strategy and the BP neural network PID control strategy, the system's response time was reduced by 62.5% and 59.46%, respectively, in the absence of load disturbances. When load disturbances were introduced, the system's average error was reduced by 81.36% and 25.57%, respectively, demonstrating the stronger robustness and the faster recovery to the desired voltage value. These results indicate that the proposed control strategy exhibits strong robustness and disturbance rejection, fast response speed, and high accuracy.

Keywords: Fuel Cell Vehicle (FCV), DC-DC, On-Board Charge, Genetic Alorithm, Neural Network PID Control.

1. Introduction

With environmental pollution and the shortage of fossil-fuel energy, the world has turned to developing the new-energy vehicle industry[1-2]. As a leader in this industry, China has developed various types of new-energy vehicles, including battery electric vehicles, hydrogen fuel-cell vehicles and hybrid-fuel vehicles. In new-energy vehicles, the output voltage of hydrogen fuel-cell stacks is highly affected by external factors like temperature 、 load. Therefore, a DC-DC converter is required between the fuel-cell stack and the motor, which can adjust the wide-range output voltage of the fuel-cell stack to the input voltage demanded by the motor[3-5].

Currently, some scholars have proposed various types of DC/DC converters. These converters can be classified into isolated and non-isolated types based on whether they contain high-frequency transformers[6]. There are three types of isolated converters: single-ended, push-pull, and bridge. The single-ended type is further divided into forward and flyback, with the open-circuit voltage of the power switch typically higher than the input voltage, and the peak current greater than the average input current, making it suitable for low-power applications. The push-pull type has a simple structure and small current ripple, but its power switch devices have higher open-circuit voltage, making it unsuitable for high-input voltage applications. The bridge type is further classified into full-bridge and half-bridge. It is suitable for high-power applications but involves more power devices, leading to higher costs. Therefore, FCEVs mostly use non-isolated DC/DC converters, which are easier to implement with soft switching, have higher power density, and are more compact. Among non-isolated DC/DC converters, traditional Buck and Boost converters can only perform simple step-down or step-up functions, and cannot match the complex

operating requirements of FCEVs. The Cuk converter, compared to the Buck-Boost converter, adds an inductor and a capacitor, which increases the size and weight of the converter. The Sepic converter, compared to the Buck-Boost converter, also has more power components, which further increases the size and weight of the converter and leads to higher power losses. On the other hand, the Buck-Boost converter can achieve both step-up and step-down functions, with a simple circuit topology that is easy to integrate. It offers high efficiency, fewer power switching devices, low losses, and a compact size. Therefore, this design selects the Buck-Boost converter.

The DC-DC converter is an important component in the power system of FCEVs. The output voltage of the fuel cell is easily influenced by factors such as load and environmental conditions, which can lead to inaccurate control of voltage and current, limiting its application. Therefore, it is necessary to choose effective modulation and control methods to achieve precise control of the fuel cell in terms of voltage, current, and power. This can enhance the stability and reliability of the entire power system and expand the application range of fuel cells.

There are three major common circuit modulation modes applied to DC-DC converters, namely Pulse-Width Modulation (PWM) [7-10], Pulse-Frequency Modulation (PFM) [11-13] and Pulse-Amplitude Modulation (PAM) [14-15].

It controls the output voltage and current by adjusting the pulse amplitude. For FCEVs, which require high precision real-time control, strong robustness, and high conversion efficiency in the Buck-Boost converter, a suitable modulation method must be selected. Although PWM modulation is stable, it causes significant losses due to the switching of the transistors, which is not conducive to achieving high-frequency conduction. PAM modulation is greatly affected by

noise and other interferences, which hinders its ability to achieve strong robustness. Therefore, this design selects the PFM modulation method, which can achieve low conduction loss, high efficiency, strong robustness, and easily realize good soft-switching characteristics.

Common control strategies mainly include traditional PID control, model predictive control, sliding mode control, fuzzy control, as well as neural network control and reinforcement learning control that emerge in recent years. Among them, PID control has the advantages of simple design, mature theoretical system and wide application scenarios, and it only requires the adjustment of three parameters to realize control. Paper [16] proposed a robust two-degree-of-freedom PID control based on quantitative feedback design to regulate the output voltage of the Buck circuit. Considering parametric uncertainties, unstructured uncertainties, set-point tracking, supply-voltage disturbances, load disturbances, and high-frequency (HF) sensor noise, a novel two-degree-of-freedom control architecture of quantitative feedback theory was adopted to implement the robust control design. Experimental results showed that it reduced the deviation of voltage peak value, accelerated the settling time, and significantly improved the attenuation of high-frequency noise. Model predictive control exhibited strong capability in dealing with multi-input multi-output systems and multiple constraints, and it was convenient for handling variable operating conditions and multiple operation modes. Paper [17] presented an operation-mode-dependent model predictive control. It utilized the predictive capability of MPC to predict system behaviors and optimize control strategies targetedly under different operation modes. Experimental results demonstrated that the strategy enhanced the stability and responsiveness of the system. Sliding mode control possessed the characteristics of insensitivity to errors and strong robustness. Paper [18] proposed a novel cascaded super-twisting sliding-mode control strategy based on PWM modulation. Different from the conventional ST-SMC control, it decoupled current regulation and voltage regulation via a terminal sliding-mode function to achieve convergence within a prescribed time, and significantly improved the dynamic response under high-current transient conditions. Experimental results showed that the strategy effectively realized voltage regulation within finite time and possessed favorable dynamic performance. Fuzzy control tolerated strong uncertainties and established fuzzy rules for control. Paper [19] presented a hybrid fractional-order PID (FOPID) controller integrated with fuzzy logic to regulate the output voltage of the DC-DC Buck converter. Extended from conventional PID control, it built the mathematical model of the Buck circuit, approximated the fractional-order operator into an integer-order equivalent form, and dynamically adjusted fuzzy rules in real time to achieve stronger robustness and more flexible control. Experimental results demonstrated that the strategy outperformed traditional tuning strategies and obtained satisfactory control performance. Neural networks imitated the neuron system in the human brain. They had powerful self-learning ability and did not require accurate mathematical modeling; modeling was completed only by obtaining the relationship between inputs and outputs. Paper [20] proposed a neural-network control scheme for DC-DC converters based on the nonlinear autoregressive exogenous (NARX) model and adaptive dynamic programming (ADP). It only captured the dynamic input and output data of the converter through the NARX

model to provide the required gradient information and equivalent model for the ADP algorithm, so that the ADP algorithm adjusted the weight parameters of the neural network controller (NNC). Experimental results showed that the anti-interference performance and robustness of the proposed strategy were superior to those of traditional control strategies. Adaptive control realized dynamic real-time self-tuning without a large amount of offline training data. Paper [21] presented a novel preset-time control scheme for DC-DC boost converters, which directly specified the upper bound of convergence time. Aiming at disturbances caused by unknown load variations, a preset-time observer integrated with adaptive parameter estimation was designed. It accurately estimated unknown loads and system states simultaneously, solved the algebraic-loop problem, and guaranteed stability without any anti-windup strategy. Experimental results demonstrated the superiority of the proposed strategy.

2. The Modeling and Analysis

Figure 1 illustrates the circuit configuration of the Buck-Boost converter. Its main topology comprises a fully controlled switch Q , an inductor L , a filter capacitor C , a diode D , and a load resistor R . Under Pulse Frequency Modulation (PFM) control, the converter operates with a switching period of t_s and a duty cycle of d . The output voltage is regulated to the desired value by toggling the conduction state of transistor Q .

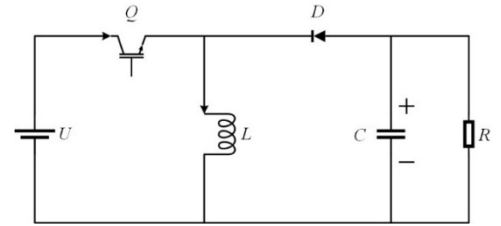


Figure 1. Buck-Boost Circuit Topology

2.1. Establish the state-space equation

Two conduction regimes are possible for the DC-DC converter: Continuous Conduction Mode (CCM) and Discontinuous Conduction Mode (DCM), of which the former is employed in the present design. Within a single switching period (T_s , with duty cycle d), the converter undergoes two distinct stages: the conduction interval ($0 \leq t \leq dT_s$) and the cut-off interval ($dT_s \leq t \leq T_s$), with their circuit structures illustrated in Fig. 2.

In the first stage (Fig. 2(a)), Q is turned on and D is off, causing V_{in} to charge the inductor while the load is sustained by the inductor L . The state-space model of the system under this condition is expressed as:

$$\frac{di_L(t)}{dt} = \frac{V_{in}}{L} \quad (1)$$

$$\frac{dV_c(t)}{dt} = \frac{-V_c}{RC} \quad (2)$$

In the equation, $i_L(t)$ is the current through the inductor, V_{in} is the input voltage, and $V_{c(t)}$ is the capacitor voltage.

During the interval (Fig2(b)) $dT_s \leq t \leq T_s$, the switch Q is off,

the diode D is also off, and the inductor L supplies energy to the filter capacitor and the load. At this point, the voltage drop across the diode is V_D . The state-space equations for the system in this condition are as follows:

$$\frac{di_L(t)}{dt} = \frac{V_C(t) - V_D}{L} \quad (3)$$

$$\frac{dV_C(t)}{dt} = -\frac{i_L(t)}{C} - \frac{V_C(t)}{RC} \quad (4)$$

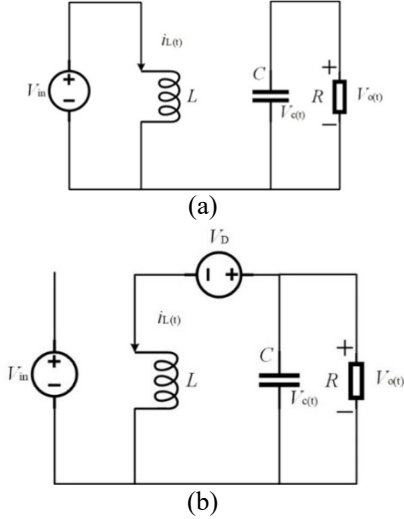


Figure 2. a).Equivalent Circuit of the Buck-Boost Converter Switch On. b).Switch Off

2.2. Establish a small-signal model

The small-signal model of the Buck-Boost circuit is developed through the state-space averaging method. By averaging the switching variables and substituting the nonlinear switching devices with equivalent controlled sources, an averaged equivalent circuit is obtained. On this basis, the transfer function of the converter is derived via small-signal analysis.

$$G(s) = \frac{u_o(s)}{d(s)} = \frac{u_o \left(\frac{1-d}{d} - \frac{sL}{R(1-d)} \right)}{LCs^2 + \frac{L}{R}s + (1-d)^2} \quad (5)$$

3. Design of Control Strategy

3.1. Design of Control Strategy

The genetic algorithm (GA) is first employed to perform offline training of the BP neural network, thereby determining the optimal initial weights. These weights are then assigned to the network, which proceeds to regulate the output voltage of the Buck-Boost circuit. As a result, the output voltage can be stabilized and rapidly converge to the desired value.

The BP neural network is trained using optimal initial weights derived from genetic algorithm optimization. After iterations, the BP neural network calculates the KP, KI, and KD parameters. Incremental PID control is applied, and the control law obtained from the PID controller is used to control the output voltage of the Buck-Boost circuit, allowing it to stabilize and rise to the desired output voltage. The control block diagram of the entire process is shown in Fig.3.

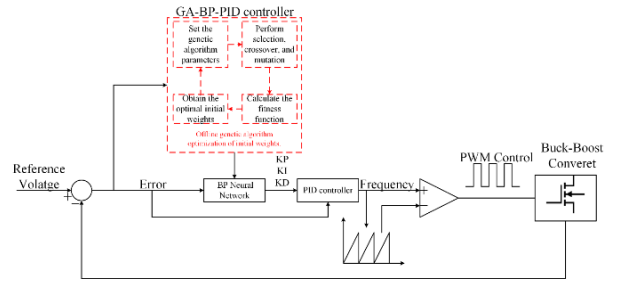


Figure 3. Control Block Diagram

BP neural networks have characteristics of complexity and non-linearity, with initial weights being randomly generated, which results in weak stability. The genetic algorithm is a search algorithm based on the biological mechanism of "survival of the fittest" and "natural selection." It is a global optimization probabilistic algorithm. By using genetic encoding, the parameters to be optimized are introduced into the population of encoded strings. Then, based on a certain fitness function and a series of genetic operations, each individual is selected. Individuals with higher fitness are retained to form a new population. This approach enables global optimization searches in complex spaces independently of the solution domain, offering strong robustness. Additionally, it does not impose high requirements on the internal structure of the optimization problem, thus reducing the mathematical complexity of the optimization problem.

In contrast to conventional search methods, the genetic algorithm initiates optimization by randomly generating a chromosome population that serves as the initial weights of the BP network. Through iterative selection, crossover, and mutation guided by the fitness function, the algorithm evolves until it converges to the optimal chromosome, thereby yielding the best initial weights. The overall optimization scheme is illustrated in Fig. 4.

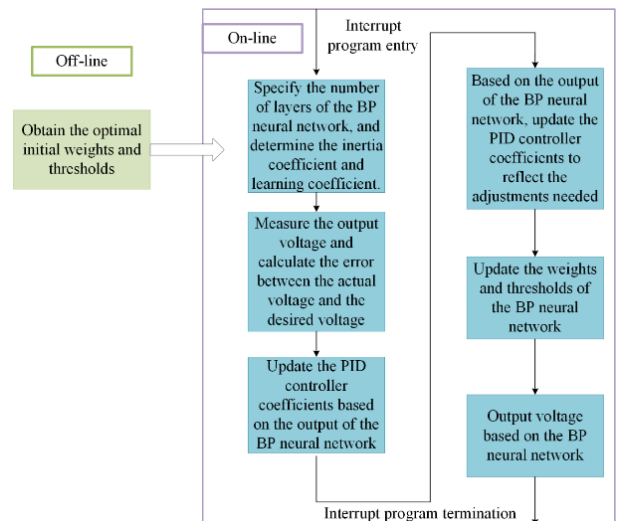


Figure 4. Genetic Algorithm Optimized BP Neural Network Process

3.2. The off-line optimization process of genetic algorithms

1.Using the genetic algorithm, the initial weights to be optimized are encoded as chromosomes. The number of individuals to be optimized is the sum of the weights from the

input layer to the hidden layer and from the hidden layer to the output layer: $4 \times 5 + 5 \times 3 = 35$. The fitness function of this genetic algorithm is defined as the difference between the reference value and the actual value of the PID control signal. The fitness of each chromosome is calculated sequentially using the fitness function, where the smaller the individual's fitness and error, the higher the probability of survival.

2. A mathematical model is established based on the Buck-Boost circuit topology. According to this model, the fitness of each chromosome is calculated using the fitness function. The roulette wheel method is then used to randomly select individuals based on their fitness, with individuals having higher fitness being more likely to be selected. After selection, the individuals form a new population.

3. Perform crossover operation on the generated new population. The crossover process uses a single-point crossover model, with a crossover probability of P_c . A random number between 0 and 1 is generated and compared with P_c to determine whether the crossover operation will be performed. Gene crossover is randomly performed between two individuals to increase the diversity of the population.

4. Perform mutation operation on the generated new population. The mutation probability is P_m . Chromosomes selected for mutation follow this formula: $X_{new} = X + b - X \cdot \left[a \cdot \left(1 - \frac{n}{m} \right) \right]^2$, where X_{new} represents the mutated individual, X represents the individual before mutation, b is the maximum boundary value of the individual, a is a random number between 0 and 1, n is the current iteration number, and m is the maximum number of iterations.

5. After m iterations of the genetic algorithm, the optimal initial weights are determined.

The above diagram (Fig.5) shows the optimal initial weights for the BP neural network obtained through offline genetic algorithm calculation. After completing the offline part, the BP neural network will perform online tuning of the PID control parameters.

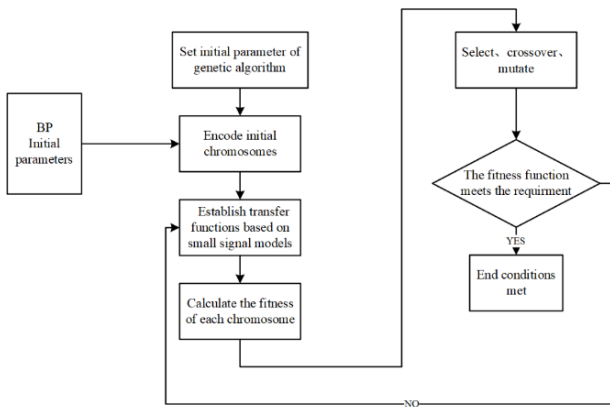


Figure 5. Offline Optimization Process of Genetic Algorithm

3.3. Online Calculation Process of BP Neural Network

The optimal weights derived from the offline genetic algorithm are loaded into the BP neural network, which subsequently performs real-time online tuning of the PID controller parameters.

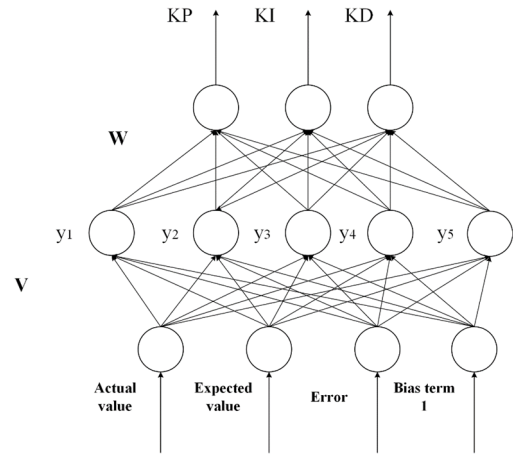


Figure 6. BP Neural Network Structure

The neural network structure of this design is shown in Fig. 6, with a 4-5-3 configuration. The input vector is:

$$\vec{X} = (x_1, x_2, x_3, x_4)^T \quad (6)$$

In the formula: x_1 is the error; x_2 is the sum of the errors; x_3 is the rate of change of the error; x_4 is the bias term 1. The number of neurons in the hidden layer is 5.

$$\vec{Y} = (y_1, y_2, y_3, y_4, y_5)^T \quad (7)$$

The output vector of the output layer is:

$$\vec{O} = (o_1, o_2, o_3)^T \quad (8)$$

In the formula: O_1 is KP; O_2 is KI; O_3 is KD.

The weight between the input layer and the hidden layer is $\vec{V}$.

$$\vec{V} = \begin{Bmatrix} v_{11} & v_{12} & v_{13} & v_{14} \\ v_{21} & v_{22} & v_{23} & v_{24} \\ v_{31} & v_{32} & v_{33} & v_{34} \\ v_{41} & v_{42} & v_{43} & v_{44} \\ v_{51} & v_{52} & v_{53} & v_{54} \end{Bmatrix} \quad (9)$$

The weight between the hidden layer and the output layer is $\vec{W}$

$$\vec{W} = \begin{Bmatrix} w_{11} & w_{12} & w_{13} & w_{14} & w_{15} \\ w_{21} & w_{22} & w_{23} & w_{24} & w_{25} \\ w_{31} & w_{32} & w_{33} & w_{34} & w_{35} \end{Bmatrix} \quad (10)$$

In the formula: v_{ij} is the weight between the input layer node i and the hidden layer node j ; w_{jk} is the weight between the hidden layer node j and the output layer node k .

It should be noted that x_0 is the threshold of the BP neural network. The activation function from the input layer to the

hidden layer is $f(x)$, which uses the Sigmoid function: $f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$. Sigmoid is an odd function, which makes it easy to differentiate, facilitating gradient descent optimization during the training process of the neural network. Typically, the output layer of the BP neural network chooses a non-negative Sigmoid function: $g(x) = \frac{1}{2}(1 + \tanh(x)) = \frac{e^x}{e^x + e^{-x}}$. However, when x approaches -1 or 1 , this function is prone to issues of vanishing or exploding gradients. Additionally, when x is near 0 , the derivative of the function can change drastically, which may lead to numerical instability during the backpropagation process. Therefore, a more robust and faster-converging activation function, the ReLU function, is used for the output layer. The traditional ReLU function is given by $g(x) = \max(0, x)$. However, since the PID control parameters output by the BP neural network can be negative, this design improves the traditional ReLU function, using $g(x) = \max(-2, x)$, which addresses the "dying neuron" problem in traditional ReLU. This modification resolves the issue where neurons with inputs less than 0 cannot update their weights, thereby preventing performance degradation in the neural network.

The mapping relationship of the output layer is:

$$net_k = \sum_{j=0}^m w_{jk} y_j \quad (11)$$

$$o_k = g(net_k) \quad (12)$$

The mapping relationship of the hidden layer is:

$$net_j = \sum_{i=0}^m v_{ij} x_i \quad (13)$$

$$y_j = f(net_j) \quad (14)$$

In the equation, net_k and net_j represent the net outputs of the hidden layer and output layer nodes, respectively; o_k and y_j represent the outputs of the hidden layer and output layer, respectively; L is the number of hidden layer nodes;

The difference between the actual output and the expected output is the error:

$$E = \frac{1}{2} (\text{yout}(k) - \text{rin}(k))^2 = \frac{1}{2} \sum_{k=1}^L (d_k - o_k)^2 \quad (15)$$

In the equation, $\text{yout}(k)$ is the output value of the neural network; $\text{rin}(k)$ is the target value; and d_k is the target value of the k -th output unit.

After the error backpropagation through the BP neural network, the error of the hidden layer is:

$$\begin{aligned} E &= \frac{1}{2} \sum_{k=1}^L (d_k - g(net_k))^2 \\ &= \frac{1}{2} \left[d_k - g \left(\sum_{j=0}^m w_{jk} y_j \right) \right]^2 \end{aligned} \quad (16)$$

The error of the output layer is:

$$\begin{aligned} E &= \frac{1}{2} \sum_{k=1}^L \left\{ d_k - g \left[\sum_{j=0}^m w_{jk} f(net_j) \right] \right\}^2 \\ &= \frac{1}{2} \sum_{k=1}^L \left\{ d_k - g \left[\sum_{j=0}^m w_{jk} f \left(\sum_{i=0}^n v_{ij} x_i \right) \right] \right\}^2 \end{aligned} \quad (17)$$

Through gradient descent, the gradient at a given point is used to compute the first-order partial derivatives to update the weights. Therefore, the errors of the weights w and v need to be calculated separately. In the equation, η represents the learning rate of the neural network. The learning rate determines whether the objective function can converge to a local minimum, making the setting of the learning rate crucial for the training process of the neural network.

$$w_{jk} = -\eta \frac{\partial E}{\partial w_{jk}} = -\eta \frac{\partial E}{\partial net_k} \frac{\partial net_k}{\partial w_{jk}} \quad (18)$$

$$v_{jk} = -\frac{\partial E}{\partial v_{ij}} = -\eta \frac{\partial E}{\partial net_j} \frac{\partial net_j}{\partial v_{ij}} \quad (19)$$

The error signal err can be defined as the error of the output layer and the error of the hidden layer, and is obtained through the chain rule of differentiation as follows:

$$\begin{aligned} err_k^o &= -\frac{\partial E}{\partial net_k} \\ &= -\frac{\partial E}{\partial o_k} \frac{\partial o_k}{\partial net_k} = -\frac{\partial E}{\partial o_k} g'(net_k) \end{aligned} \quad (20)$$

$$\begin{aligned} err_j^y &= -\frac{\partial E}{\partial net_j} \\ &= -\frac{\partial E}{\partial y_j} \frac{\partial y_j}{\partial net_j} = -\frac{\partial E}{\partial y_j} f'(net_j) \end{aligned} \quad (21)$$

The weight vectors of the output layer and hidden layer are as follows:

$$\Delta w_{jk} = \eta \cdot err_k^o \cdot y_j \quad (22)$$

$$\Delta v_{ij} = \eta \cdot err_j^y \cdot y_j \quad (23)$$

Substituting equations (22) and (23) into equations (18) and (19) yields:

$$\frac{\partial E}{\partial o_k} = -(d_k - o_k) \quad (24)$$

$$\Delta w_{jk} = \eta \cdot (d_k - o_k) g'(net_k) \cdot y_j \quad (28)$$

$$\begin{aligned} \frac{\partial E}{\partial y_j} &= \frac{\partial E}{\partial o_k} \frac{\partial o_k}{\partial y_j} \\ &= - \sum_{k=1}^L (d_k - o_k) g'(net_k) w_{jk} \end{aligned} \quad (25)$$

$$\Delta v_{ij} = \eta \cdot \left(\sum_{k=1}^L err_k^o \cdot w_{jk} \right) f'(net_j) \cdot x_i \quad (29)$$

Therefore, equations (16) and (17) can be rewritten as:

$$err_k^o = (d_k - o_k) g'(net_k) \quad (26)$$

$$\begin{aligned} err_j^y &= \left[\sum_{k=1}^L (d_k - o_k) g'(net_k) w_{jk} \right] f'(net_j) \\ &= \left(\sum_{k=1}^L err_k^o \cdot w_{jk} \right) f'(net_j) \end{aligned} \quad (27)$$

By updating the values of equations (24) and (25) in each calculation, the weights of the BP neural network can be adjusted, thereby obtaining the optimal PID control parameters. The selection of the inductor value follows the characteristics of the Buck-Boost circuit and the volt-second balance principle, after which the calculated value is multiplied by 1.2 to 1.3 times to obtain the critical inductance value. The selection of the capacitor value depends on the output voltage ripple requirement, with the output voltage ripple coefficient generally set at 0.2%. The capacitance and inductance derived from theoretical calculation are generally ideal in nature. To obtain a more stable and dependable output voltage, these values are subsequently adjusted on the basis of circuit simulation results. The resulting parameters selected for this work are presented in Tab. 1.

Table 1. Circuit Parameters

Parameters	Value
Input Voltage/V	20
Inductance/mH	2.5
Capacitance/mF	5
Resistance/ Ω	18.5
Output Voltage/V	40

4. Simulation and Analysis

To validate the effectiveness and feasibility of the control strategy proposed in this paper, simulation verification was conducted in MATLAB/SIMULINK.

4.1. Parameter Setting

To verify the effectiveness of this control strategy, a system simulation experiment is conducted using Matlab/Simulink. The performance of the BP neural network PID control optimized by the genetic algorithm is compared with that of the standard BP neural network PID control and the traditional PID control. The parameter values for the BP neural network PID control are set as follows: the learning rate is set to 0.25, and the inertia coefficient is set to 0.01.

The disturbances in the circuit system are mainly divided

into three types: input disturbances, load disturbances, and component parameter disturbances. Considering that the temperature has little effect on the capacitors and inductors in this design, and it has a minimal impact on resonance, the variation in the components themselves is not discussed in this design regarding its impact on the circuit.

Considering the issues of poor circuit connections and internal battery aging in practical applications, input interference and external load disturbances were included in the experiment. Tab. 2 lists the parameters configured for the genetic algorithm in the present design. As shown in Fig. 7, the fitness value varies with the number of generations during training, and the iterative process ceases upon the fitness value attaining 0.

Table 2. Genetic Algorithm Parameters

Parameters	Value
Maximum Population Size	120
Population Size	40
Crossover Probability	0.2
Mutation Probability	0.2

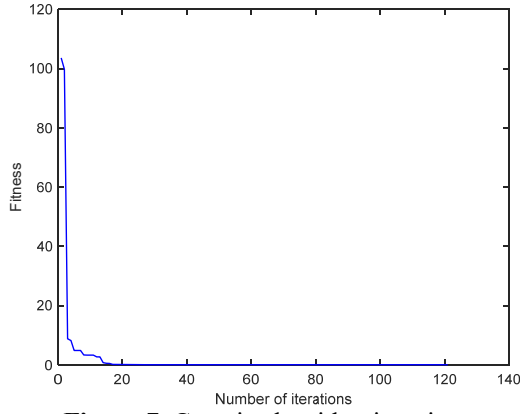


Figure 7. Genetic algorithm iteration

Fig.8 shows the variation of the fitness function after online optimization using the genetic algorithm. The x-axis represents fitness, and the y-axis represents the number of iterations. As can be seen from the figure, after approximately 20 iterations, the fitness is very small, indicating that the weights are in a better optimized state at this point.

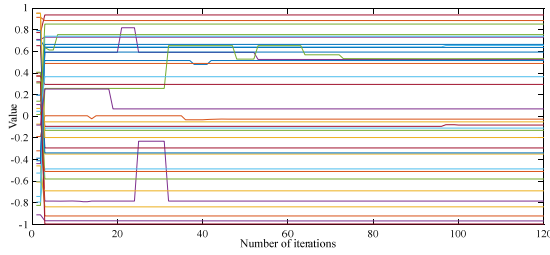


Figure 8. The Variation Process of Initial Weights Optimized by Genetic Algorithm

4.2. Buck-Boost Circuit Boost Experiment

As shown in Fig. 9, conventional PID control overshoots and settles at 0.04 s, with maximum and average errors of 56.5 V and 8.73 V. The BP-PID controller, limited by its randomly initialized weights, requires iterative tuning and thus shows a slight overshoot, reaching the target at 0.037 s with errors of 28.2 V and 2.19 V. The GA-BP-PID controller, whose initial weights are optimized by the genetic algorithm, nearly eliminates overshoot and reaches the desired output at 0.015 s, exhibiting the fastest response and strongest robustness, with errors of 5.6 V and 1.63 V.

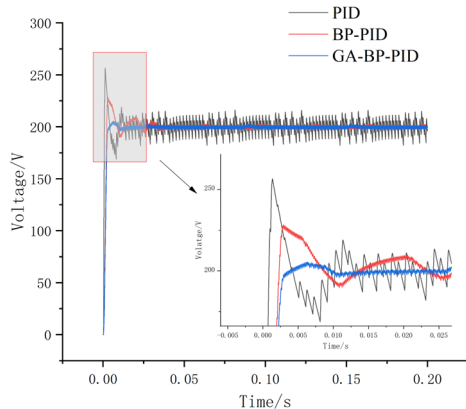


Figure 9. Normal operating condition output voltage

Table 3. Out Voltage Statistics Comparison

Control strategy	PID	BP-PID	GA-BP-PID
Response time/s	0.004	0.037	0.015
Maximum error/V	56.5	28.2	5.6
Overall average error/V	8.73	2.19	1.63

4.3. Buck-Boost Circuit Load Disturbance Experiment

In the absence of interference at the input of the subsequent Buck-Boost stage, a sudden 50% load is applied at 0.02s and removed at 0.021s. The output voltage waveform of the 50% load disturbance experiment is shown in Figure 11, with specific parameters listed in Table4. As shown in Fig.10, when the load disturbance is suddenly applied at 0.2s after the output voltages of all three control strategies have reached a steady state, the PID voltage single-loop control oscillates to 160.5V and then slowly converges toward the desired voltage, recovering to a steady state at 0.052s. The BP-PID voltage single-loop control oscillates to 147.4V before gradually converging toward the desired voltage, achieving steady-state recovery at 0.061s. The GA-BP-PID voltage single-loop control oscillates to 169.4V but quickly converges near the desired voltage, recovering to a steady state at 0.038s. Based on the above content, it can be concluded that when subjected to disturbance, the convergence time of GA-BP-PID control is reduced by approximately 43.8% and 56.1% compared to PID control and BP-PID control, respectively, demonstrating the optimal dynamic response speed.

By comparing the statistical results of the GA-BP-PID, BP-PID, and traditional PID voltage single-loop control, it is evident that when the subsequent Buck-Boost circuit experiences a load disturbance, PID control exhibits voltage overshoot and has a long convergence time to restore steady state, indicating moderate robustness. BP-PID shows a larger voltage overshoot and relatively poor robustness, although its convergence time to steady state is faster. GA-BP-PID has the smallest voltage overshoot, the strongest robustness, and the fastest convergence time to steady state, demonstrating superior control performance compared to the other two control strategies.

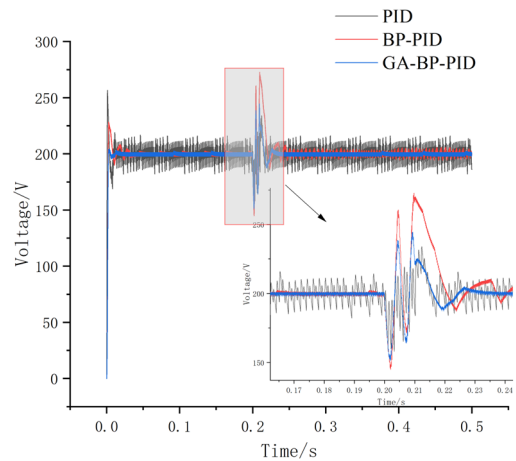


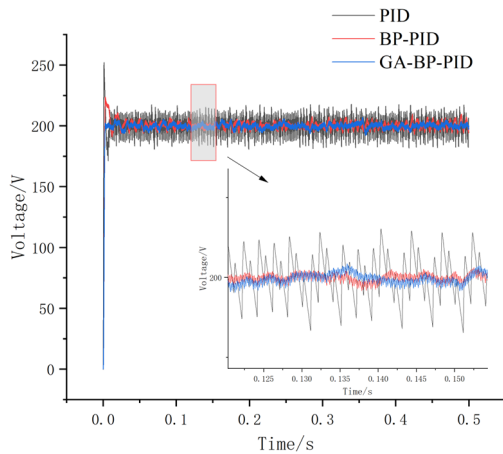
Figure 10. 50% Load Sudden Change Experimental Output Voltage

Table 4. Comparison of Response in GA-BP-PID, BP-PID, and PID Control Systems

Control strategy	PID	BP-PID	GA-BP-PID
Response time	0.040s	0.037s	0.015s
Maximum error	56.5V	28.2V	5.6V
Overall average error	8.73V	2.19V	1.63V

4.4. Buck-Boost Circuit Load Input Interference Experiment

In the case of the post-stage Buck-Boost converter without load, 20% white noise interference is added to the input voltage. The output voltage diagram of the 20% input interference experiment is shown in Figure 10, with specific parameters listed in Table 5. As shown in Fig.11, after adding 20% white noise interference to the input voltage, the maximum oscillation voltage of the PID voltage single-loop control is 52.1V, with an average error voltage of 6.74V, indicating a large oscillation range. The BP-PID voltage single-loop control shows a maximum oscillation voltage of 23.8V and an average error voltage of 3.938V, indicating a relatively large oscillation range. The GA-BP-PID voltage single-loop control achieves a maximum oscillation voltage of 3.8V and an average error voltage of 1.753V, with a smaller oscillation range. It can be concluded that the GA-BP-PID control exhibits the strongest robustness, maintaining a more stable output voltage under input voltage disturbances.

**Figure 11.** 20% Input Interference Experiment**Table 5.** Input with 20% noise statistic comparison

Control strategy	PID	BP-PID	GA-BP-PID
Response time	0.043s	0.051s	0.034s
Maximum error	52.1V	23.8V	3.8V
Overall average error	6.74V	3.938V	1.753V

5. Conclusions

This paper proposes an adaptive control strategy that optimizes BP-neural-network initial weights via a genetic algorithm and integrates it into PID control for the onboard DC-DC converter topology of FCEVs. Matlab/Simulink simulations confirm its effectiveness, showing that the strategy achieves boost control in the Buck-Boost configuration with smooth start-up, negligible oscillation, no marked voltage shocks, and fast convergence. Under load disturbances it maintains strong anti-disturbance capability, enabling real-time dynamic adjustment and high-precision,

overshoot-free output, thereby demonstrating excellent overall performance.

References

- [1] Ouyang, M., & Li, Y. (2025). Perspectives on new energy revolution in China. *CSEE Journal of Power and Energy Systems*, 11(6), 2623–2638. <https://doi.org/10.17775/CSEEJPES.2025.06840>.
- [2] Xie, Z. (2025). Optimization of new energy vehicle industry structure based on optimization link prediction similarity algorithm and CGE. *IEEE Access*, 13, 47722–47733. <https://doi.org/10.1109/ACCESS.2025.3547735>
- [3] Patel, N., Lopes, L. A. C., Li, X., & Rathore, A. K. (2026). A novel current-fed soft-switched DC–DC converter for vehicle-to-vehicle energy exchange. *IEEE Journal of Emerging and Selected Topics in Industrial Electronics*, 7(2), 743–754. <https://doi.org/10.1109/JESTIE.2026.3656509>
- [4] Reddy, R. M., Gupta, H., & Das, M. (2025). Multioutput DC–DC converter with optimal torque distribution for split drive electric vehicle applications. *IEEE Journal of Emerging and Selected Topics in Industrial Electronics*, 6(3), 1166–1171. <https://doi.org/10.1109/JESTIE.2025.3556575>
- [5] Lizana, A., Pereda, J., Rubio, F., & Rojas, F. (2025). Finite control set model predictive current control (FCS-MPCC) of three-port converter for fuel cell hybrid electric vehicles. *IEEE Transactions on Vehicular Technology*, 74(2), 2706–2714. <https://doi.org/10.1109/TVT.2024.3476924>
- [6] Filba-Martinez, A., Busquets-Monge, S., & Bordonau, J. (2020). Modulation and capacitor voltage balancing control of multilevel NPC dual active bridge DC-DC converters. *IEEE Transactions on Industrial Electronics*, 67(4), 2499–2510.
- [7] Huang, A. Q., & Chen, C. (2026). Converter PWM switch (CPS): A new PWM switch concept. *IEEE Transactions on Power Electronics*, 41(1), 185–189. <https://doi.org/10.1109/TPEL.2025.3605520>
- [8] Gu, X., Liu, Y., Zhang, G., Cao, Y., & Xia, C. (2025). Multimode carrier-based PWM strategy of open-end winding PMSM with isolated DC sources. *IEEE Transactions on Power Electronics*, 40(1), 1826–1835. <https://doi.org/10.1109/TPEL.2024.3475475>
- [9] Song, M.-S., & Kim, H. (2025). One-leg half-bridge diode-clamped four-level PWM inverter based on MNRV PWM. *IEEE Access*, 13, 155778–155796. <https://doi.org/10.1109/ACCESS.2025.3605879>
- [10] Panuganti, K., Mishra, R., Figarado, S., & Agarwal, V. (2025). Improved bus clamping PWM schemes for modular multilevel converters with reduced switching losses and capacitor voltage ripple. *IEEE Journal of Emerging and Selected Topics in Power Electronics*, 13(6), 7396–7405. <https://doi.org/10.1109/JESTPE.2025.3621704>
- [11] Jiang, H., et al. (2025). Efficient two-step PFM process employing novel crossed fields combination. *IEEE Transactions on Applied Superconductivity*, 35(7), 1–7. <https://doi.org/10.1109/TASC.2025.3596783>
- [12] Tay, S.-Y., Adrian, V., & Chang, J. S. (2026). A subminiature light-load efficient integrated DC–DC converter featuring high inductance density bond-wire inductor and low quiescent current PFM controller. *IEEE Journal of Solid-State Circuits*,

- 61(10), 5812–5824. <https://doi.org/10.1109/JSSC.2026.3703599>
- [13] Soares, J. W. M., Tonolo, É. A., & Badin, A. A. (2026). Integration of Vienna rectifier and SC-PFM for soft-switching current sensorless PFC with simplified control. *IEEE Transactions on Industrial Electronics*, 73(2), 1536–1547. <https://doi.org/10.1109/TIE.2025.3607969>
- [14] Wang, C., Gao, J., Zhang, T., Liu, J., & Chen, Z. (2026). Fast accurate measurement of time-domain metrics for baseband flexible PAM-N signals with interconnect behavioral modeling. *IEEE Transactions on Instrumentation and Measurement*, 75, 1–4. <https://doi.org/10.1109/TIM.2026.3659657>
- [15] Kang, T., Yang, J., Song, E., Son, S., Kim, H., & Han, J. (2025). A 96-Gb/s 1.6-Vppd PAM-8 transmitter with high-swing and low-loading cascaded driver in 40-nm CMOS technology. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 33(7), 2084–2088. <https://doi.org/10.1109/TVLSI.2025.3563497>
- [16] Kalvapalli, S. R., Jeyasenthil, R., Kobaku, T., & Siwakoti, Y. (2026). Quantitative feedback design based robust 2-DOF PID control of voltage mode controlled DC–DC buck converter with unstructured uncertainty and sensor noise suppression. *IEEE Journal of Emerging and Selected Topics in Industrial Electronics*, 7(2), 570–580. <https://doi.org/10.1109/JESTIE.2026.3650794>
- [17] Prakash, S., Singh, R., Dwivedi, D., Ghosh, S., & Chinmaya, K. A. (2026). Mode-dependent model predictive control for multi-source converter. *IEEE Transactions on Industry Applications*, 62(1), 1197–1209. <https://doi.org/10.1109/TIA.2025.3591581>
- [18] Bagheri, F. (2026). Cascaded terminal super twisting sliding mode control for DC–DC SEPIC converters. *IEEE Transactions on Industrial Electronics*, 73(2), 3221–3231. <https://doi.org/10.1109/TIE.2025.3608035>
- [19] Campos, M. W. d. S., et al. (2025). Development of a hybrid fractional-order fuzzy-PID controller applied to a DC-DC buck converter. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 72(9), 1218–1222. <https://doi.org/10.1109/TCSII.2025.3585464>
- [20] Yang, Z., & He, L. (2025). Neural network control based on nonlinear autoregressive exogenous model and adaptive dynamic programming algorithm for high-gain DC-DC converter. *IEEE Transactions on Industrial Electronics*, 72(12), 14128–14137. <https://doi.org/10.1109/TIE.2025.3566758>
- [21] Liu, Y., Wang, X., & Pang, N. (2025). Predefined-time nonsingular adaptive control for DC–DC boost converter with experimental verification. *IEEE Journal of Emerging and Selected Topics in Industrial Electronics*, 6(4), 1448–1458. <https://doi.org/10.1109/JESTIE.2025.3582831>