

Article

A PSO-BPNN-Based PI Parameter Tuning Strategy for the Inner Current Loop of a Vienna Rectifier with FPGA-Based Hardware-in-the-Loop Verification

Jun Chen ^{1,2,*} , Siqi Cheng ^{1,*} and Ziyi Bai ¹ ¹ School of Electrical & Electronic Engineering, Hubei University of Technology, Wuhan 430068, China² Hubei Key Laboratory for High-Efficiency Utilization of Solar Energy and Operation Control of Energy Storage System, Hubei University of Technology, Wuhan 430068, China

* Correspondence: chenjun@hbut.edu.cn (J.C.); 102410302@hbut.edu.cn (S.C.)

Abstract

To address the difficulty of conventional fixed-parameter PI controllers in the inner current loop of Vienna rectifiers in simultaneously achieving a fast dynamic response and overshoot suppression, this paper proposes a PSO-enhanced BPNN-based PI parameter-tuning strategy (PSO-BPNN-PI). The global search capability of particle swarm optimization (PSO) is exploited to optimize the initial weights of the back-propagation neural network (BPNN), mitigating the local-minima problem of random initialization and yielding a better-performing set of inner-loop PI parameters. On the equivalent current-loop model, the proposed strategy reduces the step-response overshoot to only 3.35%. On a full switching model of the Vienna rectifier, the proposed strategy is evaluated under steady-state operation, load disturbance, and filter-inductance variation and is further benchmarked against conventional PI, randomly initialized BPNN-PI, and LADRC controllers in dynamic disturbance tests. The proposed method achieves the lowest input-current THD and a balanced overall performance, while remaining competitive in tracking accuracy and maintaining its advantage over a wide inductance-mismatch range. Finally, a Hardware-in-the-Loop (HIL) platform with a heterogeneous CPU/FPGA architecture is established, on which the offline-optimized PI parameters are deployed; the results show stable tracking of the 750 V DC-bus voltage, rapid recovery from sudden load steps, and a well-balanced neutral-point voltage, confirming the engineering feasibility of the offline-optimized parameters.

Keywords: Vienna rectifier; particle swarm optimization (PSO); back-propagation neural network (BPNN); PI parameter optimization; Hardware-in-the-Loop (HIL)



Received: 8 June 2026

Revised: 3 July 2026

Accepted: 21 July 2026

Published: 23 July 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

In recent years, the rapid advancement of renewable energy generation, electric vehicle (EV) fast-charging infrastructure, and high-performance power supply systems has imposed increasingly stringent performance requirements on power conversion devices, particularly regarding the efficiency, power factor, and total harmonic distortion (THD) [1,2]. The Vienna rectifier, a representative three-phase three-level unidirectional pulse-width modulation (PWM) topology, was initially introduced by Kolar et al. at the Vienna University of Technology in 1994 [3]. Compared to conventional six-switch two-level PWM rectifiers, this topology requires only three active switches to achieve a three-level output. Consequently, it offers salient advantages, including reduced device voltage stress, diminished switching losses, low total harmonic distortion in the input current, and a high power

factor [4,5]. By virtue of these distinctive merits, the Vienna rectifier has been extensively deployed in medium-to-high power applications, such as DC fast-charging stations for electric vehicles [6], grid-tied wind power conversion systems [7], front-end rectifiers for uninterruptible power supplies (UPS) [8], and telecommunication power modules. These applications collectively demand high-quality AC–DC power conversion from the rectification system, which necessitates maintaining a stable DC-bus output voltage while ensuring a sinusoidal grid-side input current and a unity power factor operation.

The Vienna rectifier is inherently a highly coupled, nonlinear, and time-varying system. Once the gains of a conventional fixed-parameter PI controller are tuned, they remain static; consequently, when the system encounters transient conditions such as abrupt load disturbances or reference step changes, these fixed gains fail to reconcile the inherent trade-off between a rapid dynamic response and overshoot suppression [9]. Although classical remedies such as bandwidth-based PI design, gain scheduling, and relay-feedback auto-tuning [10] can partially improve the dynamic behavior of PI controllers, they generally require accurate plant information, repeated empirical adjustment, or predefined operating points, which limits their adaptability under the strong nonlinearity and wide operating range of the Vienna rectifier. To address this limitation, extensive research has been conducted globally on advanced control strategies. For instance, Sun [11] proposed a low-complexity model predictive control (LC-MPC) algorithm that improves the dynamic response and reduces the total harmonic distortion (THD) by searching for the optimal switching vector sequence. Nevertheless, like other MPC-based schemes, its predictive performance still relies on an accurate system model and precise circuit parameters, leaving it sensitive to parameter mismatch. Feng et al. [12] combined sliding-mode control with predictive current control, improving its disturbance rejection; even with an improved reaching law, however, the chattering inherent to sliding-mode control can be mitigated but not fully eliminated, and the combined structure adds to the control complexity. In [13], a scheme based on an improved sliding mode linear active disturbance rejection control (LADRC) was presented for the Vienna rectifier. Experimental results demonstrated superior dynamic response speed and anti-disturbance performance under load variations, yet the parameter-tuning process of the system remains highly complex. For unbalanced-grid operation, Li et al. [14] combined passivity-based control with ADRC for the Vienna rectifier, using line-voltage-based PBC in the inner current loop to handle voltage unbalances without negative-sequence transformation and ADRC in the outer voltage loop to improve robustness over conventional PI control. Tian et al. [15] proposed an improved sliding-mode and multi-proportional-resonant control strategy in the $\alpha\beta$ frame for Vienna rectifiers under unbalanced weak grids, reducing the complexity of conventional dq -frame decoupling while improving the dynamic response. Furthermore, an intelligent controller incorporating fuzzy logic was designed in [16] and validated in real time on a dSPACE platform, resulting in reduced input current harmonics and an improved power factor. However, the formulation of fuzzy rules heavily relies on expert experience, thereby limiting its generalizability. Collectively, while these advanced control methods enhance the performance to some extent, they generally suffer from drawbacks such as high computational complexity, tedious parameter tuning, or strong reliance on precise mathematical models, all of which restrict their widespread deployment in practical engineering applications.

Unlike the model-based methods above, combining intelligent optimization algorithms with the classical PI controller—exploiting their global search capability to automatically tune the PI parameters—has become an important route for improving the control performance of power electronic converters. Particle swarm optimization (PSO) [17] has been widely used to optimize the PI parameters of various converters. For example, Yadav et al. [18] applied PSO to the real-time tuning of the PI parameters of a unified power

quality conditioner (UPQC), demonstrating its clear superiority over static methods such as the Ziegler–Nichols approach; Roslan [19] applied a PSO-optimized PI controller to a three-phase grid-connected photovoltaic inverter, reducing the DC-bus voltage fluctuation and improving the output current quality. However, a standalone PSO yields only a single fixed set of parameters. By virtue of its strong nonlinear mapping capability, the BP neural network can map the error feedback to a corresponding combination of PI parameters, providing a flexible, learned parameterization rather than a single fixed set [20]; its conventional form, however, is hindered by the random initialization of weights. Combining PSO with the BP neural network—using PSO to globally optimize the initial weights—forms a hybrid PSO-BPNN strategy that has proven effective in many fields. For instance, PSO-BPNN-PID schemes have been successfully applied in diverse fields, including nutrient-solution conductivity control [21], hydraulic-manipulator control for explosive-ordnance-disposal robots [22], and steering-stability control of electric vehicles [23], showing improved response speed and control accuracy compared with conventional PID, BP-PID, and other optimization-based baselines. In recent years, researchers have also explored introducing neural networks into the control of multilevel/Vienna rectifiers: in Ref. [24], an artificial neural network (ANN) was combined with model predictive control (MPC) for an MMC-based rectifier; benefiting from the data-driven nature of the ANN, the controller no longer relies on an accurate system model and maintains good performance under grid- and converter-parameter variations, as well as nonideal grid conditions, thereby improving the grid adaptability of the rectifier. However, the switching frequency of such MPC-based methods is usually not fixed, which increases the difficulty of EMI filter design. Adaptive and optimization-based tuning methods have also been reported for Vienna rectifiers and related three-level converter topologies. Ahmed and Çelik [25] used a sliding-mode-based ADALINE proportional-resonant controller to adapt the source-current controller gains of a Vienna rectifier under unbalanced and distorted grid conditions. Okilly et al. [26] applied particle swarm optimization to tune MPC weighting factors for a three-level T-type NPC inverter, balancing current tracking, neutral-point-voltage stabilization, and thermal-stress distribution. To date, studies applying the hybrid PSO-BPNN optimization strategy specifically to the tuning of the inner-current-loop PI parameters of the Vienna rectifier remain relatively scarce.

Building on the above analysis, this paper proposes a hybrid PI parameter tuning strategy based on a particle swarm optimization (PSO)-enhanced back-propagation neural network (BPNN), termed PSO-BPNN-PI, and applies it to the dynamic control of the inner current loop of the Vienna rectifier. The overall system adopts a voltage-oriented control (VOC) architecture, in which Clarke and Park transformations map the AC quantities in the three-phase stationary frame into DC quantities in the synchronous rotating dq frame. The outer voltage loop stabilizes the DC-bus voltage, while the proposed PSO-BPNN-PI strategy is deployed in the inner current loop to achieve high-performance current tracking and harmonic suppression. The main contributions of this work are summarized as follows:

- (1) This work applies the hybrid PSO-BPNN strategy, in which PSO globally optimizes the initial weights of the BPNN, to the inner-current-loop PI tuning of the Vienna rectifier, a setting for which relatively few studies are available. Rather than proposing a new optimization algorithm, the focus is on adapting this known framework to the specific requirements of the rectifier's current loop. Specifically, PSO initializes the BPNN weights, which map the current error to the PI gains; unlike direct PSO tuning, which returns a single fixed set of gains, the controller retains the standard PI structure while improving tuning reliability. The resulting controller is validated on the equivalent model, on the full switching model, and on an FPGA-based HIL platform using the offline-optimized parameters.

- (2) To make the offline optimization tractable, the digital sampling delay, the PWM modulation delay, and the inductor branch are organized into a single low-order equivalent plant model, with each coefficient related to the corresponding physical hardware parameters. This provides a low-cost and reusable optimization target for the PSO-BPNN procedure, while the resulting controller is still validated against the full switching model rather than relying on the simplified model alone.
- (3) To move beyond pure software simulation, a Hardware-in-the-Loop (HIL) validation platform based on a heterogeneous CPU/FPGA architecture is built, following real-time HIL approaches that have proven to be effective for Vienna rectifiers [27]. On this platform, only the offline-optimized fixed PI parameters are deployed, and the tuning strategy is thereby taken from simulation through to hardware-level validation. Within a physical environment that reproduces realistic hardware delays and electrical characteristics, the engineering feasibility of the obtained PI parameters and the system robustness under load disturbances are verified.

2. Modeling of the Vienna Rectifier

The main circuit topology of the three-phase Vienna rectifier is shown in Figure 1, where E_x ($x = a, b, c$) denotes the grid voltage, i_x the input current, and L_f ($f = a, b, c$) and R_x ($x = a, b, c$) the input filter inductors and their equivalent parasitic resistances, respectively. $D_1 - D_6$ are six fast-recovery diodes; every two of the six switching devices are connected in anti-series to form three bidirectional switches, S_a , S_b , and S_c . C_1 and C_2 are the DC-bus capacitors, U_{dc} is the output voltage, and R_L is the output load. The control objectives of this rectifier mainly comprise the output voltage, the input current, and the balance of the neutral-point voltage [28]. The advantage of the dq -frame-based dual-closed-loop control strategy for the Vienna rectifier is that it transforms the AC quantities in the abc frame into DC quantities in the dq frame, so that zero steady-state error of the input current can be achieved using proportional–integral (PI) regulators [29]. From an engineering-control perspective, the outer voltage loop has a relatively long dynamic-response period, and conventional PI control is sufficient to meet its steady-state power-balance requirements; however, the inner current loop, which handles high-frequency decoupling and rapid command tracking, is more sensitive to its control parameters [30]. Under transient conditions such as a sudden load increase or a command step change, a fixed-gain inner-loop PI regulator often struggles to balance overshoot against the settling time, which in turn tends to induce DC-bus voltage fluctuations.

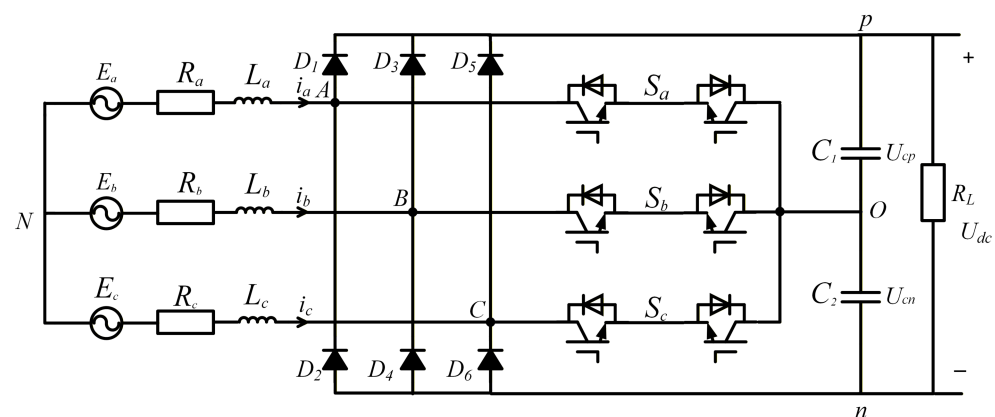


Figure 1. Main circuit topology of the Vienna rectifier.

To simplify the nonlinear model of the Vienna rectifier and achieve the independent regulation of active and reactive power, the system adopts a voltage-oriented control (VOC) vector control strategy. First, by introducing the Clarke and Park transformations [31],

the control system equivalently maps the AC voltages and currents in the three-phase stationary abc frame into DC quantities in the synchronous rotating dq frame. Under the assumption of a symmetrical grid voltage, the d -axis is aligned with the grid-voltage vector; combining Kirchhoff's laws with the state-space averaging method, the current dynamic equations of the Vienna rectifier in the dq frame are derived as follows [32]:

$$\begin{cases} L \frac{di_d}{dt} = e_d - Ri_d + \omega Li_q - u_d \\ L \frac{di_q}{dt} = e_q - Ri_q - \omega Li_d - u_q \end{cases} \quad (1)$$

In the above equations, e_d and e_q are the grid-voltage components based on the d - and q -axis components of the grid voltage, respectively; i_d and i_q are the AC-side inductor-current components on the d - and q -axes; u_d and u_q are the AC-side terminal-voltage components of the Vienna rectifier on the d - and q -axes; and ω is the grid synchronous angular frequency. From Equation (1), it can be seen that the current components i_d and i_q in the dq frame contain cross-coupling terms, which degrade the control performance. To eliminate this coupling effect, a state feed-forward compensation mechanism is introduced; the corresponding decoupling control law is designed as follows:

$$\begin{cases} u_d^* = e_d - (k_p + \frac{k_i}{s})(i_d^* - i_d) + \omega Li_q \\ u_q^* = e_q - (k_p + \frac{k_i}{s})(i_q^* - i_q) - \omega Li_d \end{cases} \quad (2)$$

where k_p and k_i denote the proportional and integral gain coefficients of the PI controller, respectively; i_d^* and i_q^* are the d - and q -axis components of the reference current vector; and u_d^* and u_q^* are the reference voltage commands finally outputted by the control system. Combining the mathematical modeling and the derivation of the feed-forward control law presented above, the complete coordinate-transformation flow of the Vienna rectifier constructed in this paper is shown in Figure 2.

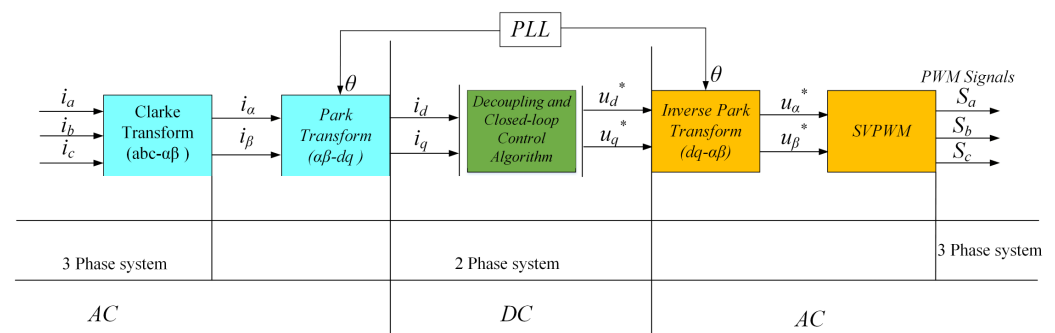


Figure 2. Coordinate-transformation control flow of the Vienna rectifier. Different colors are used only to visually distinguish the functional blocks for clarity and do not convey any additional physical meaning.

To verify the effectiveness of the feed-forward decoupling control law described above, the Laplace transform is first applied to Equation (1) to convert it into the complex-frequency domain. Assuming that the SVPWM modulation process behaves as an ideal linear gain [33] (i.e., the actual output voltage equals the commanded voltage), Equation (2) is substituted into the physical equations in the complex-frequency domain. After rearrangement, the decoupled closed-loop current equations of the system are obtained as follows:

$$\begin{cases} (Ls + R)i_d = (k_p + \frac{k_i}{s})(i_d^* - i_d) \\ (Ls + R)i_q = (k_p + \frac{k_i}{s})(i_q^* - i_q) \end{cases} \quad (3)$$

From Equation (3), the cross-coupling terms no longer appear, and the d -axis and q -axis are reduced to two completely independent single-variable control systems. Based on the above decoupled mathematical model, the detailed decoupling control logic of the inner current loop is shown in Figure 3.

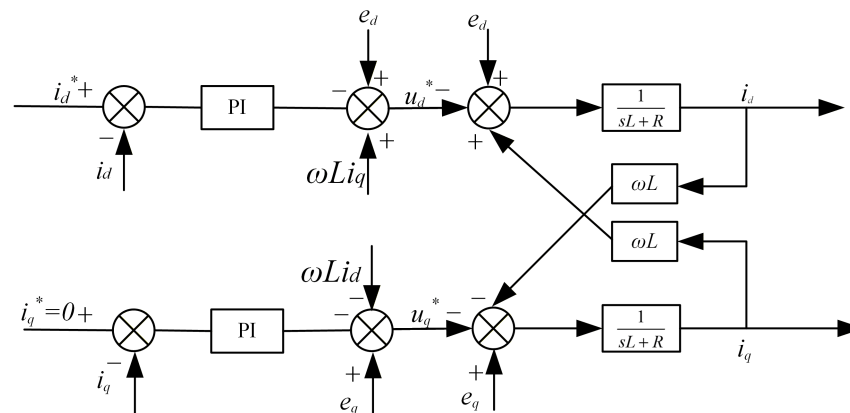


Figure 3. Decoupling control diagram.

Since the decoupled d -axis and q -axis inner current loops are fully symmetric and the q -axis current command is always zero under unity-power-factor operation [34], this paper takes the d -axis as an example for dynamic analysis.

In Figure 4, $\frac{1}{T_s s + 1}$ is the current-sampling delay element and $\frac{1}{0.5T_s s + 1}$ is the PWM waveform-output element, where T_s is the PWM switching period, which is also the current sampling period in the experiment [35]. By merging the small-time-constant elements and neglecting the higher-order infinitesimal terms when expanding the denominator [36], the open-loop transfer function of the current loop is obtained as follows:

$$G_{oi}(s) = \left(k_p + \frac{k_i}{s}\right) \left(\frac{1}{1.5T_s s + 1}\right) \left(\frac{1}{Ls + R}\right), \quad (4)$$

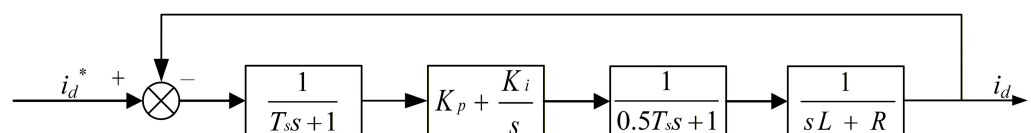


Figure 4. Control block diagram of the d -axis.

After the conventional PI controller is removed, the controlled plant consists of the digital-delay element in Equation (4) cascaded with the inductor branch. Since $(L \gg 1.5T_s R)$, this term can be neglected, and the transfer function of the actual inner-current-loop plant is obtained as follows:

$$G_p(s) = \left(\frac{1}{1.5T_s s + 1}\right) \left(\frac{1}{Ls + R}\right) \approx \frac{1}{1.5T_s L \cdot s^2 + L \cdot s + R}. \quad (5)$$

It should be noted that the equivalent inner-current-loop model in Equation (5) is established under the assumptions of a symmetrical three-phase grid voltage, dq -frame decoupling, and unity-power-factor operation. Under these conditions, the q -axis current reference is set to zero and the d -axis current loop primarily determines the active-power transfer and the DC-bus voltage-regulation dynamics. To obtain a low-order analytical model suitable for offline optimization, the SVPWM stage is approximated as an ideal linear gain, and the switching ripple, dead-time effects, modulation saturation, negative-sequence coupling, and higher-order small-signal terms are neglected. In Equation (5), R denotes the

equivalent series parasitic resistance of the input-side filter-inductor branch. Consequently, the equivalent model adequately captures the dominant low-frequency dynamics of the inner current loop and is suitable as the training target for PSO-BPNN offline optimization.

Nevertheless, the above assumptions also bound the applicability range of the equivalent model. Under conditions of pronounced grid-voltage unbalance, severe harmonic distortion, strong transients, large parameter mismatch, or operation of the modulator near its saturation region, the deviation between the simplified model and the full switching model may increase, thereby affecting the direct applicability of the control parameters or the gain-mapping relationship obtained from the equivalent model. In addition, when the input-side filter inductance is markedly reduced, the sampling period is markedly increased, or the input-side equivalent series impedance is markedly increased, the approximation condition $L \gg 1.5T_s R$ should be re-examined, and the complete term should be retained in the modeling if necessary. Therefore, the equivalent model in Equation (5) is not used as a substitute for the full switching model of the Vienna rectifier, but only as a low-computational-cost target for offline optimization; the resulting PSO-BPNN-PI controller is still further validated for its effectiveness and engineering applicability through the full switching model and the HIL platform.

3. Design of the PSO-BPNN-Optimized PI Controller

3.1. Particle Swarm Optimization

Particle swarm optimization (PSO) originates from the simulation of the cooperative foraging behavior of bird flocks. The algorithm abstracts each candidate solution as a “particle” with neither mass nor volume in the solution space [37]. During the iterative search, the system evaluates the quality of each particle’s position by computing its fitness value through the objective function. By memorizing and learning from its own historical best position (the personal extremum) and the best position shared by the swarm (the global extremum), each particle continuously and dynamically adjusts its flight velocity and direction; through cooperative information exchange within the swarm, all particles are gradually guided toward the global optimum. Mathematically, the PSO search process is described as follows. In a D-dimensional space containing N particles, the position of particle i is expressed as $x_i = (x_{i1}, x_{i2}, \dots, x_{iD})$, $i = 1, 2, 3 \dots N$, and its velocity is expressed as $v_i = (v_{i1}, v_{i2}, \dots, v_{iD})$. The best position reached by particle i during the search, i.e., the personal best ($pbest$), is given by $p_i = (p_{i1}, p_{i2}, \dots, p_{iD})$, and based on the personal best of all particles, the swarm obtains the global best ($gbest$), given by $p_g = (p_{g1}, p_{g2}, \dots, p_{gD})$. In the d -th dimension, particle i updates its velocity and position as follows:

$$v_{id}(t+1) = w(t)v_{id}(t) + c_1r_1(t)(p_{id}(t) - x_{id}(t)) + c_2r_2(t)(p_{gd}(t) - x_{id}(t)), \quad (6)$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1), \quad (7)$$

The velocity update in the d -th dimension consists of three parts. The first is the inertia term $w(t) \cdot v_{id}(t)$, which preserves the particle’s previous motion trend, where the inertia weight decreases linearly with the iteration count as $w(t) = w_{max} - \frac{(w_{max} - w_{min})t}{T_{max}}$, with w_{max} and w_{min} being its upper and lower bounds, and T_{max} is the maximum number of iterations; a larger w in the early stage favors global exploration, while a smaller w later strengthens local exploitation. The second is the individual-cognition term, by which the particle corrects its current direction according to its own historical best position, gradually approaching the personal optimum. The third is the social term, by which the particle incorporates the swarm’s global best position during the search, thereby guiding the entire swarm toward the global optimum while preserving individual search ability [38]. The position-update process of the PSO algorithm is illustrated in Figure 5.

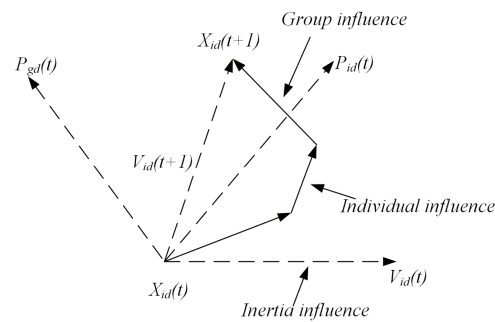


Figure 5. Position-update diagram of the PSO algorithm. Dashed arrows denote the velocity vectors and directions toward the personal and global best positions, while solid arrows show the vector composition of inertia, individual, and group influences.

3.2. BP Neural Network

The BP neural network is a multilayer feedforward network trained by the back-propagation algorithm, consisting of an input layer, a hidden layer, and an output layer. It is well suited to systems that are difficult to describe by explicit mathematical models or rules; being intrinsically nonlinear, it can realize arbitrary nonlinear mappings and is currently the most widely used type of artificial neural network [39].

After receiving the reference current signal $r(k)$, the control system compares it with the actual inductor-current signal $y(k)$ measured by the current sensor and computes the error $e(k)$. Based on this error, the system then outputs a control signal through the PI regulation mechanism, forming a closed loop. The PI controller parameters k_p and k_i are tuned based on the hybrid PSO-BPNN strategy. The BP neural network designed in this paper has a 3-5-2 topology. The input layer contains three neurons, corresponding to the current error $e(k)$, the error change rate $\Delta e(k) = e(k) - e(k-1)$, and a bias term of 1, respectively. The input signals are normalized before being fed into the hidden layer. The hidden layer comprises five neurons with the tanh activation function; the output layer contains two neurons, which apply the sigmoid function to produce the normalized values of k_p and k_i ; these are then scaled to their admissible ranges to obtain the PI parameters for the current control cycle.

The inputs to the input layer of the BPNN are expressed as follows:

$$O_i^{(1)} = x(i), i = 1, 2, 3, \quad (8)$$

The inputs and outputs of the hidden layer are calculated as follows:

$$net_j^{(2)}(k) = \sum_{i=1}^3 v_{ij}^{(2)} O_i^{(1)}, j = 1, 2, 3, 4, 5, \quad (9)$$

$$O_j^{(2)}(k) = f[net_j^{(2)}(k)], j = 1, 2, 3, 4, 5 \quad (10)$$

where the superscripts (1), (2), and (3) denote the input, hidden, and output layers, respectively; $v_{ij}^{(2)}$ represents the connection weight from the i -th node in the input layer to the j -th node in the hidden layer. Additionally, k denotes the discrete sampling step, while i and j represent the neuron indices for the input and hidden layers, respectively. The structure of the BP neural network is shown in Figure 6.

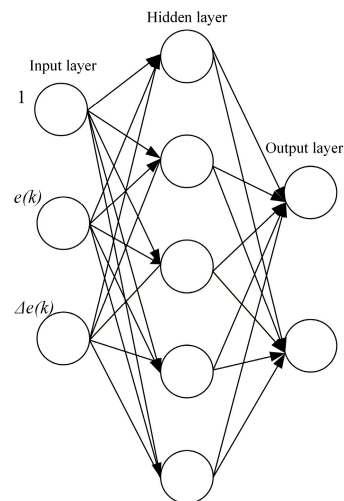


Figure 6. Neural network structure diagram.

The activation function for the neurons in the hidden layer is the hyperbolic tangent function, defined as follows:

$$f(x) = \tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}, \quad (11)$$

The inputs and outputs of the output layer are respectively given by the following:

$$net_l^{(3)}(k) = \sum_{j=1}^5 v_{jl}^{(3)} O_j^{(2)}(k), l = 1, 2, \quad (12)$$

$$O_l^{(3)}(k) = g[net_l^{(3)}(k)], l = 1, 2, \quad (13)$$

where $v_{jl}^{(3)}$ represents the connection weight from the j -th node in the hidden layer to the l -th node in the output layer, and l represents the neuron index for the output layer. Furthermore, $O_1^{(3)}, O_2^{(3)}$ denote the normalized outputs associated with kp and ki , respectively. The actual PI gains are obtained by scaling these outputs to their admissible ranges. The activation function applied to these two output parameters is the sigmoid function, given by the following:

$$g(x) = \frac{1}{1 + e^{-x}}, \quad (14)$$

The performance index function of the control system is defined as follows:

$$E(k) = \frac{1}{2} [r(k) - y(k)]^2 = \frac{1}{2} e^2(k), \quad (15)$$

The weight update formula for the hidden layer is as follows:

$$\Delta v_{ij}^{(2)}(k) = \alpha \Delta v_{ij}^{(2)}(k-1) + \eta \delta_j^{(2)} O_i^{(1)}(k), \quad (16)$$

The output layer weights are updated using a gradient descent method with a momentum term. According to the chain rule, this weight adjustment relies on the Jacobian information of the controlled plant, denoted as $\frac{\partial y(k)}{\partial u(k)}$ [40]. In engineering practice, its sign function, $\text{sgn}\left[\frac{\partial y(k)}{\partial u(k)}\right]$, is typically employed as an approximation, while the amplitude variation is absorbed into the global learning rate η of the network.

By incorporating the partial derivatives of the positional PI control law with respect to the output node k_p, k_i . The local error gradients of the output layer can be directly derived as follows:

$$\delta_1^{(3)}(k) = e^2(k) \cdot \text{sgn} \left[\frac{\partial y(k)}{\partial u(k)} \right] \cdot g'[\text{net}_1^{(3)}(k)], \quad (17)$$

$$\delta_2^{(3)}(k) = e(k) \cdot \left(\sum_{i=0}^k e(i) \right) \cdot \text{sgn} \left[\frac{\partial y(k)}{\partial u(k)} \right] \cdot g'[\text{net}_2^{(3)}(k)], \quad (18)$$

By introducing an additional momentum factor α , the weight-update rule used in the offline optimization stage for the output layer can be uniformly expressed as follows:

$$\Delta v_{jl}^{(3)}(k) = \alpha \Delta v_{jl}^{(3)}(k-1) + \eta \delta_l^{(3)}(k) O_j^{(2)}(k). \quad (19)$$

$$\delta_j^{(2)}(k) = f'[\text{net}_j^{(2)}(k)] \sum_{l=1}^2 \delta_l^{(3)}(k) v_{jl}^{(3)}(k), \quad (20)$$

$$f'(x) = 1 - f^2(x), \quad (21)$$

$$g'(x) = g(x)[1 - g(x)]. \quad (22)$$

3.3. Hybrid PSO-BPNN Algorithm

To address the parameter-optimization problem of the inner-loop PI controller, this paper proposes a composite control strategy that combines particle swarm optimization (PSO) with a back-propagation neural network (BPNN). Existing studies have shown that using a swarm-intelligence algorithm to optimize the initial weights of a neural-network controller can markedly improve convergence performance and control accuracy [41]. In this method, the BPNN acts as a nonlinear mapper for the PI parameters: taking the error and its rate of change as network inputs, it outputs k_p and k_i through the nonlinear transformation of the hidden layer. In the offline optimization stage, each PSO particle carries a complete set of network initial weights, drives the PI controller through forward propagation to perform a closed-loop simulation of the plant in Equation (5) and evaluates the system response according to the ITAE criterion [42]. PSO balances global exploration and local exploitation through a linearly decreasing inertia-weight strategy, and after several generations of iteration, it selects the globally optimal weights w_i and w_o , so that the BPNN maps the error to the corresponding PI parameter.

The procedure for applying the hybrid PSO-BPNN optimization algorithm to PI-parameter tuning is as follows:

- (1) Algorithm initialization. Set an appropriate swarm size, the search-space dimension D , the maximum number of iterations, and the decreasing range of the inertia weight.
- (2) Weight decoding and forward inference. Decode the position vector of the current particle back into the initial weight matrix of the BP neural network; taking the system error $e(k)$ and the error change rate $\Delta e(k)$ as the BPNN inputs, output the PI parameters k_p and k_i through forward propagation.
- (3) Construction of the fitness function. The ITAE (Integral of Time-weighted Absolute Error) criterion is adopted, which strengthens the suppression of residual error in the steady-state stage through a time-weighting mechanism:

$$J_{ITAE} = \int_0^\infty t|e(t)|dt, \quad (23)$$

where t is time and $e(t)$ is the control error at time t , i.e., the difference between the reference current $r(t)$ and the actual feedback $y(t)$.

- (4) Update of personal and global best. Compare the current fitness with the historical best, and update the personal best position $pbest$ and the global best position $gbest$.

- (5) Update of velocity and position. Update the search direction and step size of each particle according to the PSO velocity-update formula, combined with the linearly decreasing inertia weight and the learning factors.
- (6) Termination check. If the number of iterations reaches the maximum, output the globally optimal weights as the optimal weights of the BPNN; otherwise, return to step (2) to re-decode the updated particle positions and rerun the closed-loop simulation, entering the next generation of optimization. The overall optimization procedure is illustrated in Figure 7.

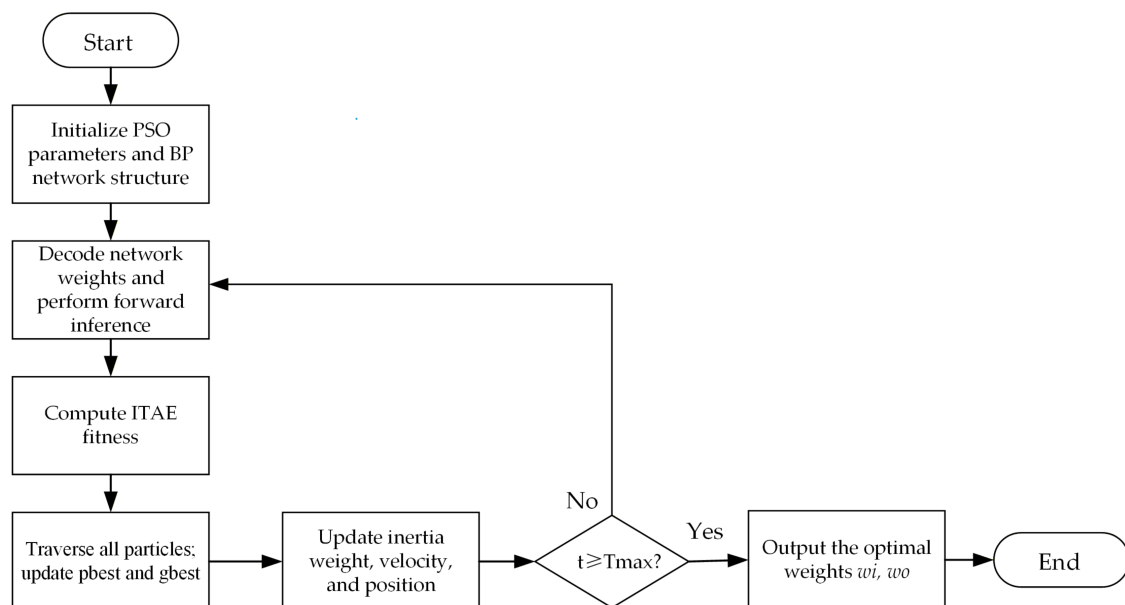


Figure 7. Flowchart of the hybrid PSO-BPNN optimization algorithm. Rounded rectangles indicate the start and end points, rectangles denote processing steps, the diamond represents the termination decision, and arrows indicate the process flow.

4. Simulation and Experimental Verification

4.1. Equivalent-Model Verification

Based on the MATLAB R2021b (The MathWorks, Inc., Natick, MA, USA) simulation platform and taking the inner-current-loop plant model established in Equation (5) as the optimization target, the PSO-BPNN-PI closed-loop simulation system shown in Figure 8 is constructed. On this equivalent plant, the PSO algorithm performs offline global optimization of the initial weights of the BPNN, so that the trained network learns mapping from the current-error state to the inner-loop PI parameters at a low computational cost. The equivalent model thus serves only as the training and optimization target for the offline PSO-BPNN procedure, rather than as a substitute for the full switching model. In the experimental design, three control strategies—the conventional PI controller, the BPNN-PI controller, and the proposed PSO-BPNN-PI controller—are implemented and compared under the same plant and simulation conditions. The unit-step response test is used to evaluate the dynamic performance of each algorithm in terms of the overshoot, rise time, and settling time; the sudden-load-disturbance and parameter-perturbation tests are used to verify the disturbance-rejection capability and robustness of the system.

The controlled plant in this study is the inner current loop of the Vienna rectifier. In this subsection, the Vienna rectifier is represented by a simulation model rather than commercially sourced physical equipment. The overall control system adopts a dual-closed-loop structure consisting of an outer voltage loop and an inner current loop, in which the outer voltage loop uses conventional PI control to maintain a stable DC-bus

voltage, while the inner current loop uses the proposed PSO-BPNN-PI control strategy. The main simulation parameters of the system are listed in Table 1.

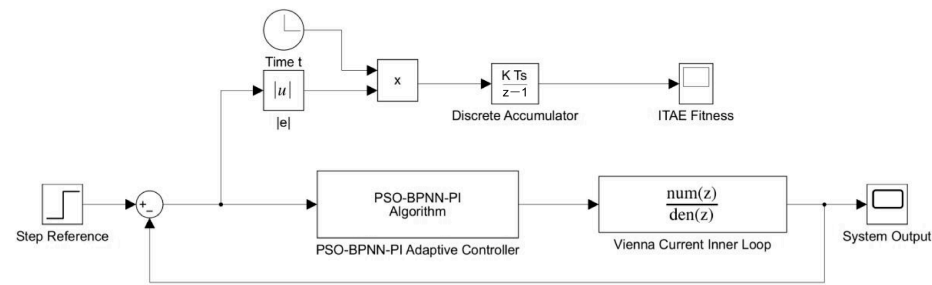


Figure 8. PSO-BPNN-PI closed-loop simulation system.

Table 1. System and main optimization parameters.

Parameter	Value	Parameter	Value
Input-side parasitic resistance R	$0.005 \, \Omega$	Switching frequency f_s	10 kHz
Input-side filter inductance L	1.8 mH	Conventional PI gains	$K_P = 16, K_i = 10$
PSO settings	$D = 25, N = 40, Iter_{max} = 50, w_{max} = 0.9, w_{min} = 0.4, c_1 = c_2 = 1.45$		
BPNN settings	$3\text{-}5\text{-}2, \eta = 2 \times 10^{-5}, \alpha = 0, K_P \in [10, 24], K_i \in [6, 18]$		
Fitness and seeds	ITAE, Seed: 2026		

The proportional and integral gains of the conventional PI current controller were determined empirically. Starting from typical inner-current-loop PI values used in comparable three-phase PWM/Vienna rectifier simulations, the gains were refined in the full switching model under rated operating conditions until the current loop exhibited a stable, well-damped response with acceptable DC-bus voltage fluctuation and input-current THD. The resulting baseline gains, $K_P = 16, K_i = 10$, were then held fixed across all comparative simulations to provide a common and fair baseline.

The iterative convergence curves of the two optimization algorithms are compared in Figure 9. The two algorithms exhibit a clear difference in PI-parameter tuning: the PSO algorithm reaches a stable fitness value after about 25 iterations, with a final steady-state value of approximately 4.4×10^{-3} , whereas the PSO-BPNN strategy converges within only 17 iterations, with a final steady-state value of approximately 1.88×10^{-3} . These results indicate that the PSO-BPNN strategy provides faster convergence and better fitness minimization for PI-parameter tuning under the adopted optimization objective. It should be noted that the standalone PSO method is used only as an optimization benchmark in the fitness convergence analysis, whereas the subsequent closed-loop control comparison focuses on PI, BPNN-PI, and PSO-BPNN-PI to evaluate the contribution of neural-network-based PI tuning and PSO-assisted initialization.

To evaluate the parameter-optimization performance of the PSO-BPNN-tuned PI algorithm, the conventional PI and BPNN-PI algorithms are added for comparison. A simulation model driven by a unit-step input signal is run according to the transfer function and the PI control principle, and the results are shown in Figure 10 and Table 2.

As clearly shown by the comparison, under the same plant and control conditions, the overshoot of the PSO-BPNN-PI strategy is only 3.35%. Compared with the other two methods, its overshoot is smaller, which helps prevent excessive system fluctuation and detuning. Its settling time is also short, and the system performance is markedly improved. As shown in Figure 10b, the PSO-BPNN-tuned gains K_P and K_i start from their initial values and converge to steady values within the transient stage. This gain-mapping behavior contributes to the reduced overshoot observed in Figure 10a.

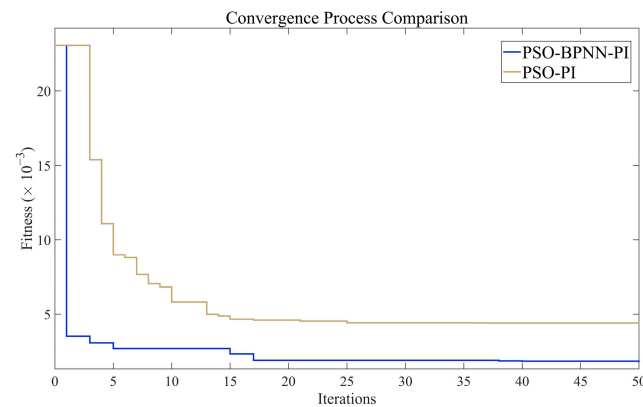


Figure 9. Comparison of the iterative fitness (ITAE) convergence curves.

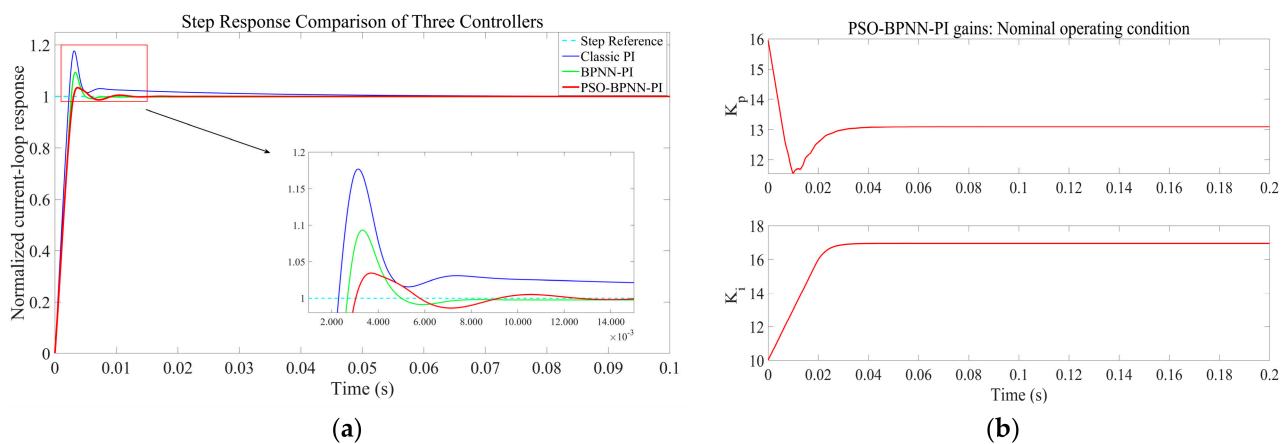


Figure 10. Unit-step response of the equivalent current-loop model: (a) normalized current-loop response; (b) PSO-BPNN-PI gain trajectories.

Table 2. Analysis of the simulation results.

Control Method	Overshoot (%)	Rise Time (ms)	Settling Time (ms)
Conventional PI	17.3	2	46
BPNN-PI	9.33	2.2	15
PSO-BPNN-PI	3.35	2.3	16

Figure 11 illustrates the robustness of the system under the combined condition of parameter perturbation and command step change. At $t = 0.1$ s, two disturbances are applied simultaneously: the AC-side inductance L is doubled to emulate parameter perturbation, and the reference is stepped from 1.0 to 1.5 to emulate a command change. The results show that the conventional PI produces an overshoot exceeding 10%; the BPNN-PI exhibits lag and overshoot, while the PSO-BPNN-PI strategy achieves the smallest overshoot with a settling time comparable to the other two, demonstrating strong robustness under the coexistence of parameter uncertainty and command change. As shown in Figure 11b, when the reference step and inductance perturbation occur at $t = 0.10$ s, the PSO-BPNN-PI controller updates K_p and K_i toward new bounded values. This gain variation helps explain the improved transient response in Figure 11a.

Figure 12 shows the system's response to a step disturbance during steady-state operation. All three controllers undergo a transient undershoot and then recover: the conventional PI exhibits the deepest undershoot and slowest recovery, the BPNN-PI is intermediate, and the proposed PSO-BPNN-PI recovers fastest with the shallowest undershoot and no secondary oscillation. Since all three contain integral action and thus

reach zero steady-state error, the difference lies in the recovery speed obtained by the PSO-BPNN-tuned gains, whose time evolution is shown in Figure 12b.

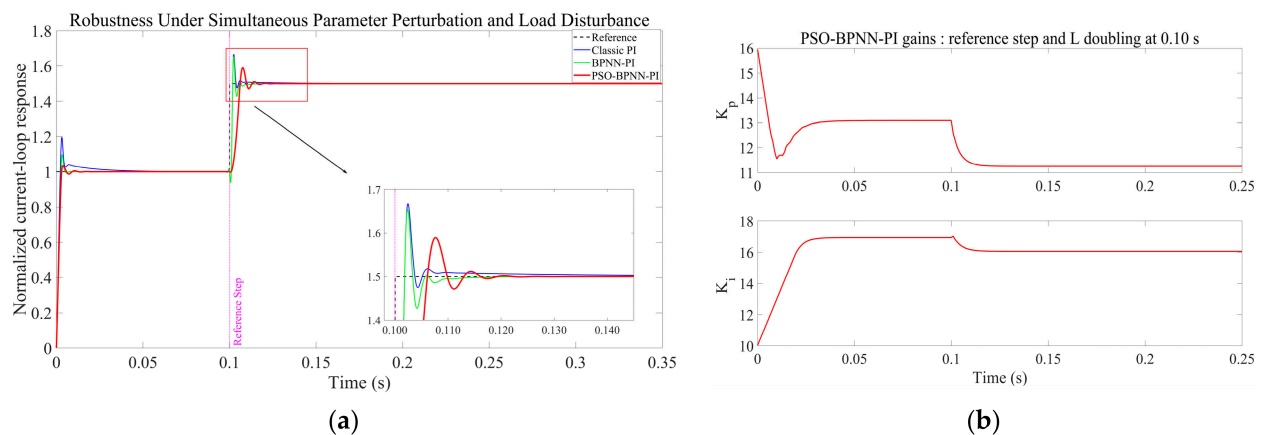


Figure 11. Response under reference step and inductance perturbation: (a) normalized current-loop response; (b) PSO-BPNN-PI gain trajectories.

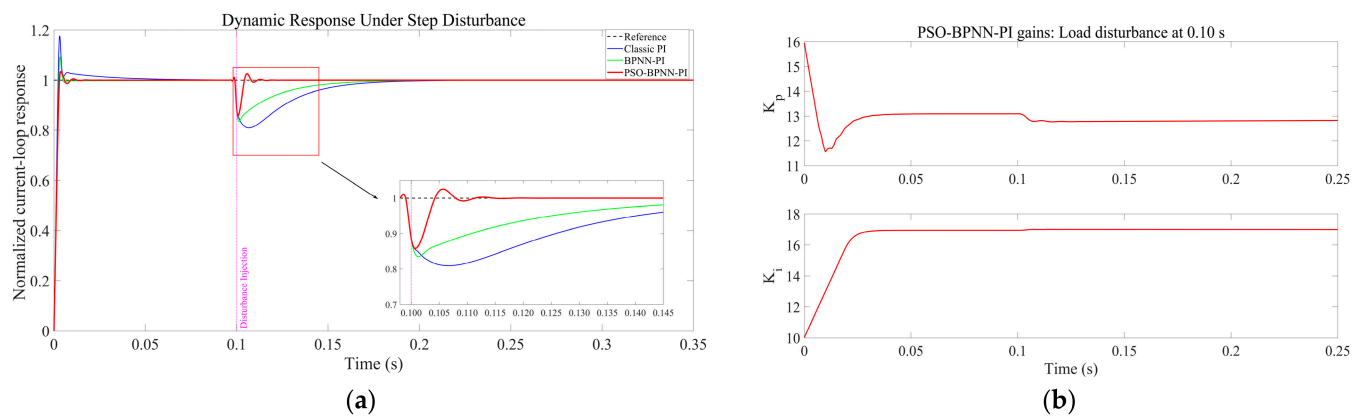


Figure 12. Response under load disturbance: (a) normalized current-loop response; (b) PSO-BPNN-PI gain trajectories.

4.2. Switching-Model Verification

To further verify the strategy under realistic switching conditions, a full switching model of the three-phase Vienna rectifier is built in MATLAB/Simulink R2021b, including the power devices, SVPWM modulation with neutral-point balancing, PLL, and VOC-based cascaded control. The main circuit simulation parameters are listed in Table 3. Under the same main-circuit parameters, SVPWM strategy, and sampling frequency, the conventional PI, BPNN-PI, and PSO-BPNN-PI controllers are first compared in the steady-state regime (Table 4).

Table 3. Simulation parameters of the Vienna rectifier system.

Parameter	Value	Parameter	Value
Phase RMS voltage	220 V	Grid frequency f	50 Hz
Input-side filter inductance L	1.8 mH	Input-side parasitic resistance R	0.005 Ω
DC-side voltage-divider capacitors C_1, C_2	2200 μF each	Switching frequency f_s	10 kHz
Rated load resistance R_L	37.5 Ω	Rated DC-bus voltage	750 V
Rated power	15 kW	CPU control period	100 μs
FPGA simulation step	1 μs	--	--

Table 4. Steady-state indicator. The input-current THD is calculated from the 2nd to 50th harmonics with respect to the 50 Hz fundamental component.

Controller	U_{dc} Settling (s)	THD (%)	Neutral-Point Difference (V)
Classic PI	0.0697	0.347	0.190
BPNN-PI	0.0701	0.338	0.189
PSO-BPNN-PI	0.0692	0.221	0.191

Since the steady-state comparison mainly verifies stable operation under rated conditions rather than disturbance-rejection capability, the LADRC baseline is not included in the steady-state comparison but is added to the dynamic disturbance tests, including parameter perturbation and the sudden load change, where disturbance-rejection capability is the key metric. LADRC is a less model-dependent robust controller that lumps unmodeled dynamics, parameter mismatch, dq -axis cross-coupling, and external disturbances into a total disturbance, which is estimated by a linear extended state observer and compensated in the control law. Its parameters were selected according to the bandwidth-parameterization principle and refined in the full switching model, with the observer bandwidth chosen conservatively to limit switching-ripple and noise amplification. The final values are b_0 scaling factor = 0.75, $\omega_c = 3000$ rad/s, and $\omega_0 = 6000$ rad/s.

During start-up and the transition to the rated operating condition, the system responds normally and satisfies the steady-state criteria. All three controllers raise the DC-bus voltage to 750 V and the d -axis current to its rated value; under steady-state conditions, their signal trajectories almost completely coincide, and the steady-state indices are given in Table 4. The settling time, grid-side power factor, and neutral-point voltage difference of the three algorithms are nearly identical.

A sudden load increase was emulated by injecting a 4 A current step from a parallel current source on the DC side at $t = 0.5$ s. The experimental results are shown in Table 5. To quantitatively evaluate the current-tracking performance, the maximum d -axis tracking error, the integral of absolute error (IAE), and the input-current THD are adopted as evaluation indicators. The IAE is defined as follows:

$$IAE = \int_0^{t_f} |e(t)| dt \approx \sum_k |e(k)| T_s, \quad (24)$$

where $e(k) = i_d^*(k) - i_d(k)$ is the d -axis current-tracking error, T_s is the sampling period, and t_f denotes the evaluation time window.

Table 5. Current tracking and input-current THD under dc-side load step.

Controller	Max i_d Error (A)	i_d IAE (A·s)	THD (%)
Classic PI	0.428	0.013	0.311
BPNN-PI	0.414	0.012	0.309
LADRC	0.347	0.012	0.408
PSO-BPNN-PI	0.360	0.011	0.232

Among the four controllers, the PSO-BPNN-PI achieves the lowest tracking IAE and input-current THD; the LADRC attains a slightly smaller maximum d -axis error owing to its disturbance observer, but at a higher input-current THD than the PSO-BPNN-PI.

A repeated-initialization check was further conducted for the Table 5 load-step case using five independent random seeds. Compared with BPNN-PI, PSO-BPNN-PI reduced the input-current THD in all five trials ($0.265 \pm 0.021\%$ vs. $0.328 \pm 0.008\%$) and improved the current-tracking indices in four of the five trials. Although one less-favorable initialization increased the mean and standard deviation of the peak tracking error, the median

maximum d-axis current error, id-axis IAE, and THD of PSO-BPNN-PI remained lower than those of BPNN-PI, indicating improved typical performance. These results confirm that random initialization can affect the final controller quality; therefore, repeated offline validation with full-switching-model verification is recommended before fixed-parameter HIL deployment.

To assess the sensitivity to filter-parameter mismatch, the four controllers for the rated $L = 1.8$ mH were tested with the actual filter inductance set to 1.8, 3.6, and 5.4 mH, while the same excitation was applied in all cases. Table 6 reports the d -axis tracking IAE and the input-current THD.

Table 6. Current tracking and power-quality indicators under filter-inductance parameter mismatch.

Controller	L (mH)	i_d IAE (A·s)	THD (%)
Classic PI	1.8	0.023	0.40
BPNN-PI	1.8	0.016	0.41
LADRC	1.8	0.015	0.43
PSO-BPNN-PI	1.8	0.014	0.29
Classic PI	3.6	0.022	0.455
BPNN-PI	3.6	0.017	0.498
LADRC	3.6	0.014	0.437
PSO-BPNN-PI	3.6	0.012	0.345
Classic PI	5.4	0.023	0.561
BPNN-PI	5.4	0.018	0.538
LADRC	5.4	0.017	0.540
PSO-BPNN-PI	5.4	0.013	0.417

To further verify the robustness of the proposed control method under parameter mismatch and load disturbance, the actual filter inductance is set to 5.4 mH, i.e., three times the rated design value of 1.8 mH. At $t = 0.5$ s, a 4 A current-sink branch is connected in parallel on the DC side to emulate a sudden increase in load power. For all four controllers, the outer-loop voltage control, the SVPWM modulation scheme, the neutral-point voltage-balancing strategy, and the simulation parameters are kept identical, with only the structure of the inner-current-loop controller being changed. The current tracking error results are shown in Figure 13.

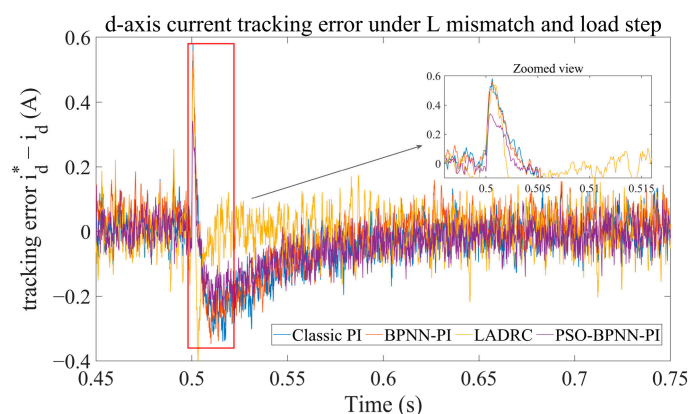


Figure 13. d -Axis current tracking error under filter-inductance mismatch and dc-side load step.

As shown in Figure 13 and Table 7, the combined disturbance of the inductance mismatch and sudden load increase causes noticeable transient tracking errors in all four controllers. The conventional PI and BPNN-PI exhibit relatively larger error fluctuations because their disturbance-compensation capability is limited under severe parameter variation. LADRC shows good error-attenuation capability owing to its extended state

observer, but its observer-based structure is more sensitive to switching ripple and measurement noise, resulting in a relatively higher input-current THD. In contrast, the proposed PSO-BPNN-PI controller achieves the smallest maximum tracking error and the lowest input-current THD among the four controllers, while maintaining a relatively low IAE, indicating a more balanced performance between transient current tracking and input-current quality. These results indicate that PSO-assisted initialization improves the parameter-tuning accuracy of the BPNN and enhances the robustness of the inner current loop under severe inductance mismatch and DC-side load disturbance.

Table 7. Current tracking and power-quality indicators under severe filter-inductance mismatch and DC-side load disturbance.

Controller	Max i_d Error (A)	i_d IAE (A·s)	THD (%)
Classic PI	0.581	0.0132	0.462
BPNN-PI	0.558	0.0120	0.448
LADRC	0.536	0.007	0.514
PSO-BPNN-PI	0.341	0.0107	0.419

It should be emphasized that the PSO-BPNN-PI controller optimized using the equivalent model in Equation (5) is directly applied to the full switching model without re-training. The results in Tables 5–7 show that the proposed controller maintains balanced performance under the dc-side load step, filter-inductance mismatch, and combined-disturbance conditions, indicating that the equivalent model captures the dominant low-frequency dynamics of the inner current loop and that the learned error-to-PI-parameter mapping can be transferred to the full switching model for the 750 V, 15 kW Vienna rectifier studied here. Nevertheless, such transferability is subject to the modeling assumptions. If the voltage level, power rating, filter inductance, sampling frequency, or modulation strategy changes significantly, the mismatch between the equivalent model and the full switching model may increase, and the controller should be re-trained or re-tuned accordingly. Therefore, the equivalent model is used only as an efficient offline optimization tool, while the full switching simulation and HIL experiment are used to assess the transferability and engineering applicability of the optimized controller.

4.3. Architecture and Construction of the HIL Platform

To further verify the feasibility of deploying the offline-optimized PI parameters obtained from the proposed PSO-BPNN tuning strategy in a real hardware system and to bridge the gap between offline simulation performed using MATLAB/Simulink R2021b and practical engineering implementation, an FPGA-based Hardware-in-the-Loop (HIL) platform is built, and a semi-physical experimental verification of the Vienna rectifier control system is carried out. The platform adopts the EasyGo PXIe-200 chassis (Wuhan Senmu Leishi Technology Co., Ltd., Wuhan, China) as its core hardware and uses EasyGo DeskSim 2024 (version 24.2.2.3) for model configuration and real-time execution, as shown in Figure 14.

The platform employs a heterogeneous computing architecture that combines a high-frequency CPU with a low-level FPGA. However, in the practical engineering application of the Vienna rectifier, due to the high-frequency switching period and the real-time multiply-accumulate computing bottleneck of the low-level digital signal processor (DSP/FPGA), running the complete forward and backward propagation of the BP neural network within an extremely short control cycle carries the risk of computation timeout. To balance the sophistication of the control algorithm with the realizability of the physical hardware, an offline-optimization, online-fixed-parameter deployment scheme is adopted in the FPGA-based HIL experiment. That is, the complete PSO-BPNN-tuned PI control algorithm is

run on the host computer to perform in-depth offline optimization under various typical operating conditions, and a fixed set of PI parameters is extracted from the offline-optimized controller and downloaded to the FPGA hardware execution unit. Consequently, the HIL experiment validates the engineering feasibility of this deployed operating point under realistic hardware delays, whereas the online adaptive inference of the BPNN—exercised in the simulations of Sections 4.1 and 4.2—is not activated on the FPGA in the present implementation. Accordingly, the computational burden of the proposed strategy is mainly concentrated in the offline optimization stage. The PSO-BPNN optimization does not consume real-time control resources. In the full switching simulations, BPNN-PI and PSO-BPNN-PI use the same compact 3-5-2 network for forward PI-gain inferences, so their per-cycle computational burdens are comparable and only slightly higher than that of the conventional PI controller. In the FPGA-based HIL implementation, a fixed set of offline-optimized PI parameters is deployed; therefore, the online computational burden is essentially the same as that of the conventional PI controller.



Figure 14. Real-time simulation system.

The software architecture of the HIL system is shown in Figure 15. The control system module, including the UI control interface and the core control logic, is deployed on the CPU node and executes the control law using the offline-optimized PI parameters obtained by the PSO-BPNN algorithm. The CPU also performs the SVPWM modulation and sends the corresponding switching commands to the FPGA. The physical main-circuit topology model of the Vienna rectifier is compiled into real-time simulation components and hosted on the FPGA node, where it runs with a fixed simulation step of 1 μ s. The CPU-side control algorithm is executed with a control period of 100 μ s. At the closed-loop interaction level, voltage and current feedback signals are calculated by the FPGA plant model and transmitted to the CPU for monitoring and control computation. The CPU then generates the gate-driving commands and sends them back to the FPGA ports to control the switching actions of the rectifier model, thereby forming the HIL closed-loop test chain.

In the HIL experiment, the control performance of the offline-optimized PI parameters obtained by the PSO-BPNN-tuned PI control algorithm is verified in a physical hardware environment. To ensure consistency between the real-time verification and the switching-model simulation, the Vienna rectifier parameters used in the HIL platform are kept the same as those listed in Table 3. The real-time output response waveforms captured by the host-computer oscilloscope are shown in the following figures.

Figure 16 shows the DC-bus voltage waveform of the system. After an overshoot of about 15% during the start-up transient, the total DC-side voltage stabilizes near the rated value of 750 V within a short time, with a small ripple. The upper and lower voltage-divider capacitor voltages both remain at about 375 V and maintain good voltage-balancing characteristics.

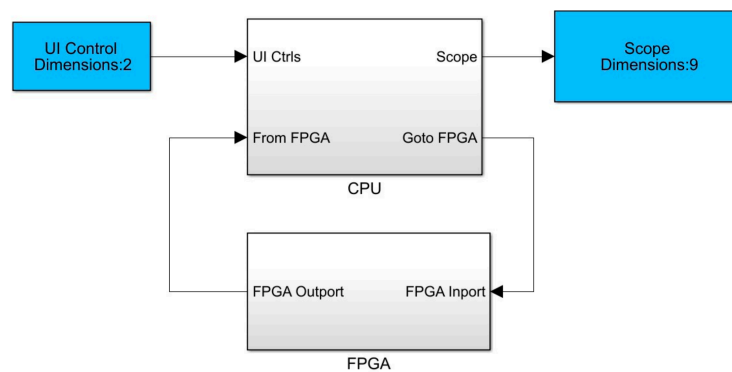


Figure 15. Software architecture of the HIL system. The colors follow the control software's default settings and have no additional physical meaning. The arrows indicate the directions of signal transmission between the modules.

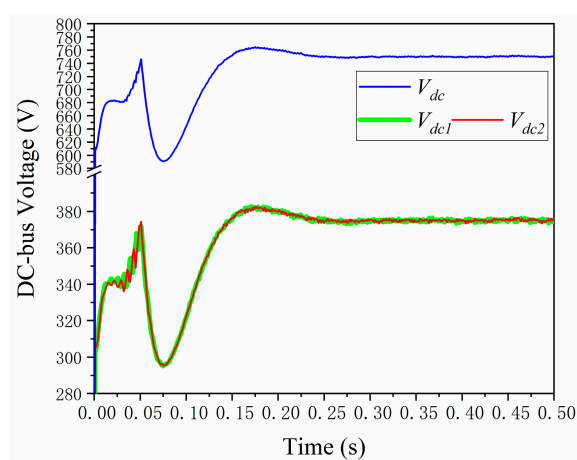


Figure 16. DC-bus voltage.

Figure 17 shows the waveforms of the phase-A grid voltage and input current during steady-state operation. The current remains in phase with the voltage, indicating that the system achieves unity-power-factor operation under the proposed control strategy, with the reactive power being approximately zero.

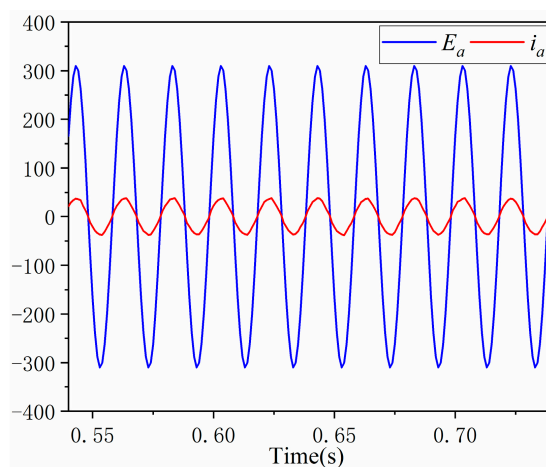


Figure 17. Grid voltage and input current.

Figure 18a,b show the three-phase input voltage and current waveforms under steady-state conditions. The three-phase voltages are standard symmetric sinusoids with an amplitude of about 311 V and successive phase intervals of 120° , while the three-phase currents are symmetrically sinusoidal with an amplitude of about 35 A, exhibiting low waveform distortion and good three-phase balance.

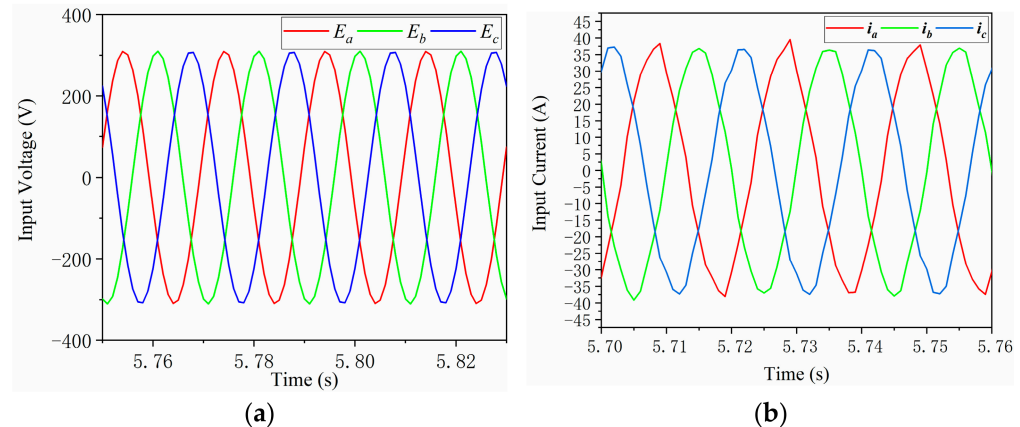


Figure 18. Steady-state three-phase waveforms of the Vienna rectifier: (a) three-phase voltages; (b) three-phase currents.

Figure 19a,b show the dynamic response of the system before and after a parallel load is suddenly applied on the DC side. At the instant of the load change, the DC-bus voltage exhibits a transient drop, and the two divider-capacitor voltages drop and recover synchronously while maintaining good voltage balance throughout. After the load change, the three-phase current amplitude increases accordingly as the power demand rises, with a smooth transition and no obvious distortion or oscillation.

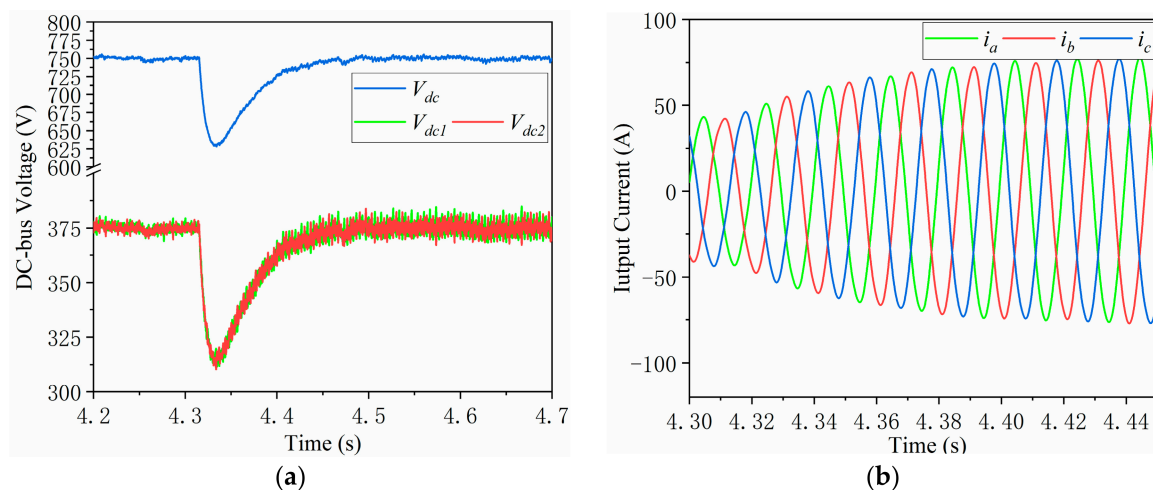


Figure 19. Dynamic response under a suddenly applied parallel load: (a) DC-bus voltage V_{dc} and the upper/lower divider-capacitor voltages V_{dc1} , V_{dc2} ; (b) three-phase input currents i_a , i_b , i_c .

Figure 20 shows the phase-A grid voltage and input current before and after a parallel load is suddenly applied. After the load change, the current amplitude increases accordingly as the power demand rises, while the current remains in phase with the grid voltage throughout the transient, indicating that the system maintains unity-power-factor operation under load disturbance.

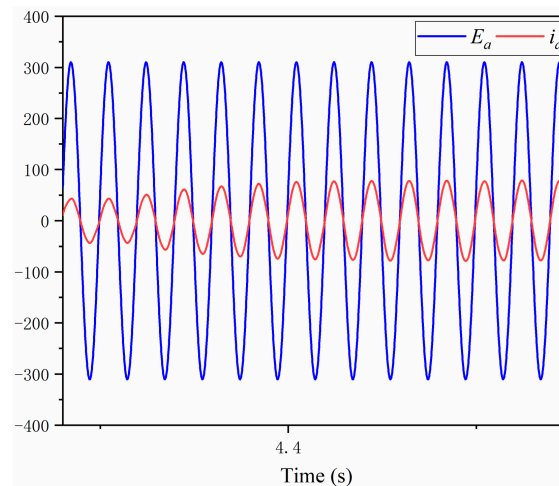


Figure 20. Phase-A grid voltage and input current under a suddenly applied parallel load.

5. Conclusions

To improve the current-tracking performance of the Vienna rectifier under transient conditions, such as sudden load changes and command steps, this study proposes a PI parameter-tuning strategy based on a BP neural network optimized via particle swarm optimization (PSO-BPNN-PI). By combining the global search capability of PSO with the nonlinear mapping capability of the BP neural network, the method effectively improves both the tuning accuracy of the inner-loop PI parameters and the dynamic control performance of the system.

To overcome the sensitivity of the conventional BP neural network to initial weights and its tendency to fall into local minima, PSO is introduced into the optimization of the network weights. Meanwhile, considering that the inner current loop of the Vienna rectifier contains multiple dynamic elements such as the digital sampling delay, the PWM modulation delay, and the inductor-branch impedance, an equivalent second-order plant transfer function is established to provide a clear optimization target for the algorithm. Theoretical analysis and simulation results show that this equivalent model captures the dominant low-frequency dynamics of the inner current loop and can serve as an effective target for offline PI-parameter optimization, while the resulting controller is further validated using the full switching model and the HIL platform.

In the proposed PSO-BPNN-PI strategy, PSO performs global optimization of the BP network and converges to a steady state within only 17 iterations, which is effectively faster than direct PSO-based PI parameter tuning. The simulation and HIL experimental results show that the method not only reduces the step-response overshoot to 3.35% while maintaining a short settling time of 16 ms, but also exhibits fast recovery and competitive steady-state accuracy with a balanced overall performance under complex conditions, such as parameter perturbation, a command change, and load disturbance. The HIL experiment further confirms that the offline-optimized PI parameters achieve stable tracking of the 750 V DC-bus voltage and unity-power-factor operation under realistic hardware-delay conditions, indicating the method's potential for rectification applications with large load fluctuations.

From an engineering perspective, the improvements offered by the proposed method are practically meaningful: the reduced input-current THD helps improve the grid-side current quality, and the automated PSO-BPNN tuning reduces the reliance on repeated manual trial-and-error, improving the reproducibility of the parameter-tuning process. The additional complexity is concentrated in the offline optimization stage, whereas the online computational burden, particularly in the HIL deployment with fixed parameters, remains

essentially the same as that of a conventional PI controller. The proposed method is therefore most attractive for Vienna-rectifier applications with frequent load changes, significant parameter uncertainty, or stringent current-quality requirements, while a conventional PI controller remains a simple and effective choice for fixed operating conditions or extremely resource-constrained platforms.

Although the proposed algorithm improves the control performance, the HIL experiment adopts an offline-optimization and online fixed-parameter-deployment scheme due to the strict real-time constraints and limited multiply-accumulate resources of the FPGA-based platform. Therefore, the real-time adaptive potential of online BPNN inference has not yet been fully exploited in the present hardware implementation. In addition, it should be noted that the validation in this work is limited to a 750 V, 15 kW Vienna rectifier, and the broader applicability of the proposed method to other voltage levels, power ratings, and rectifier topologies has not been experimentally verified and remains as future work. Future work will therefore focus on the lightweight and fixed-point implementation of the neural network to reduce the computational burden and enable the online real-time tuning of PI parameters. Experimental validation across a wider range of voltage levels, power ratings, and rectifier topologies will also be conducted to further assess its tracking performance and robustness in broader industrial application scenarios.

Author Contributions: Conceptualization, Z.B.; methodology, S.C.; software, S.C.; validation, S.C.; formal analysis, S.C.; data curation, S.C.; investigation, J.C.; writing—original draft preparation, S.C.; writing—review and editing, J.C. and Z.B.; visualization, S.C.; supervision, J.C.; project administration, J.C.; funding acquisition, Z.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China, grant number 62503156, and the Doctoral Research Startup Fund of Hubei University of Technology, grant number 4301/00347.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding authors.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

AC	Alternating current
BPNN	Back-propagation neural network
CPU	Central processing unit
DC	Direct current
DSP	Digital signal processor
FPGA	Field-programmable gate array
HIL	Hardware-in-the-loop
ITAE	Integral of time-weighted absolute error
IAE	Integral of Absolute Error
PI	Proportional–integral
PSO	Particle swarm optimization
PWM	Pulse-width modulation
SVPWM	Space-vector pulse-width modulation
VOC	Voltage-oriented control

References

1. Singh, B.; Singh, B.N.; Chandra, A.; Al-Haddad, K.; Pandey, A.; Kothari, D.P. A review of three-phase improved power quality AC-DC converters. *IEEE Trans. Ind. Electron.* **2004**, *51*, 641–660. [\[CrossRef\]](#)
2. Rodriguez, J.R.; Dixon, J.W.; Espinoza, J.R.; Pontt, J.; Lezana, P. PWM regenerative rectifiers: State of the art. *IEEE Trans. Ind. Electron.* **2005**, *52*, 5–22. [\[CrossRef\]](#)
3. Kolar, J.W.; Zach, F.C. A novel three-phase utility interface minimizing line current harmonics of high-power telecommunications rectifier modules. In *Proceedings of the INTELEC '94, Vancouver, BC, Canada, 30 October–3 November 1994*; IEEE: New York, NY, USA, 1994; pp. 367–374. [\[CrossRef\]](#)
4. Kolar, J.W.; Ertl, H.; Zach, F.C. Design and experimental investigation of a three-phase high power density high efficiency unity power factor PWM (VIENNA) rectifier employing a novel integrated power semiconductor module. In *Proceedings of the APEC '96, San Jose, CA, USA, 3–7 March 1996*; IEEE: New York, NY, USA, 1996; Volume 2, pp. 514–523. [\[CrossRef\]](#)
5. Kolar, J.W.; Friedli, T. The Essence of Three-Phase PFC Rectifier Systems—Part I. *IEEE Trans. Power Electron.* **2013**, *28*, 176–198. [\[CrossRef\]](#)
6. Bharaneedharan, B.; Suresh, P.; Elumalai, P.V.; Asif, M. Energy-efficient Vienna rectifier for electric vehicle battery charging stations. *Results Eng.* **2024**, *23*, 102671. [\[CrossRef\]](#)
7. Mansouri, A.; El Magri, A.; Lajouad, R.; Giri, F. Control design and multimode power management of WECS connected to HVDC transmission line through a Vienna rectifier. *Int. J. Electr. Power Energy Syst.* **2024**, *155*, 109563. [\[CrossRef\]](#)
8. Jin, N.; Zhang, L.; Chen, H.; Zhou, K.; Wu, X. Design of a novel double closed loop system of the Vienna rectifier in the pre-stage of UPS. *Int. J. Power Electron.* **2023**, *17*, 18–39. [\[CrossRef\]](#)
9. Wang, T.; Chen, S.; Li, X.; Zhang, J.; Ma, J. Direct Power Control of Vienna Rectifier Based on Fractional Order Sliding Mode Control. *World Electr. Veh. J.* **2024**, *15*, 543. [\[CrossRef\]](#)
10. Borase, R.P.; Maghade, D.K.; Sondkar, S.Y.; Pawar, S.N. A review of PID control, tuning methods and applications. *Int. J. Dyn. Control* **2021**, *9*, 818–827. [\[CrossRef\]](#)
11. Sun, Z.; Jin, W.; Wu, F.; Han, Q.; Guan, K.; Ren, J. A novel low-complexity model predictive control for Vienna rectifier. *Int. J. Circuit Theory Appl.* **2023**, *51*, 5136–5153. [\[CrossRef\]](#)
12. Feng, X.; Tao, Y.; Cui, X.; Shao, K.; Wang, Y. Sliding mode and predictive current control strategy of the three-phase Vienna rectifier. *J. Power Electron.* **2020**, *20*, 743–753. [\[CrossRef\]](#)
13. Meng, F.; Liu, J.; Wang, J.; Liu, Y.; Wang, L. Vienna rectifier control strategy based on improved sliding mode active disturbance rejection. *J. Power Electron.* **2025**, *25*, 1041–1053. [\[CrossRef\]](#)
14. Li, J.; Wang, M.; Wang, J.; Zhang, Y.; Yang, D.; Wang, J.; Zhao, Y. Passivity-based control with active disturbance rejection control of Vienna rectifier under unbalanced grid conditions. *IEEE Access* **2020**, *8*, 76082–76092. [\[CrossRef\]](#)
15. Tian, Y.; Yuan, H.; Zhu, W.; Li, X.; Li, Y. Sliding mode control of Vienna rectifier under unbalanced weak power grid. *IEEE Access* **2024**, *12*, 39095–39109. [\[CrossRef\]](#)
16. Aissa, O.; Moulahoum, S.; Colak, I.; Babes, B.; Kabache, N. Design and real time implementation of three-phase three switches three levels Vienna rectifier based on intelligent controllers. *Appl. Soft Comput.* **2017**, *56*, 158–172. [\[CrossRef\]](#)
17. Kennedy, J.; Eberhart, R. Particle swarm optimization. In *Proceedings of the ICNN'95—International Conference on Neural Networks, Perth, WA, Australia, 27 November–1 December 1995*; IEEE: New York, NY, USA, 1995; pp. 1942–1948. [\[CrossRef\]](#)
18. Yadav, S.K.; Patel, A.; Mathur, H.D. PSO-Based Online PI Tuning of UPQC-DG in Real-Time. *IEEE Open J. Power Electron.* **2024**, *5*, 1419–1431. [\[CrossRef\]](#)
19. Roslan, M.F.; Al-Shetwi, A.Q.; Hannan, M.A.; Ker, P.J.; Zain, A.W.M. Particle swarm optimization algorithm-based PI inverter controller for a grid-connected PV system. *PLoS ONE* **2020**, *15*, e0243581. [\[CrossRef\]](#) [\[PubMed\]](#)
20. Wang, J.; Li, M.; Jiang, W.; Huang, Y.; Lin, R. A Design of FPGA-Based Neural Network PID Controller for Motion Control System. *Sensors* **2022**, *22*, 889. [\[CrossRef\]](#) [\[PubMed\]](#)
21. Wang, Y.; Liu, J.; Li, R.; Suo, X.; Lu, E. Application of PSO-BPNN-PID Controller in Nutrient Solution EC Precise Control System: Applied Research. *Sensors* **2022**, *22*, 5515. [\[CrossRef\]](#) [\[PubMed\]](#)
22. Zhou, Y.; He, X.; Shao, F.; Zhang, X. Research on the Optimization of the PID Control Method for an EOD Robotic Manipulator Using the PSO Algorithm for BP Neural Networks. *Actuators* **2024**, *13*, 386. [\[CrossRef\]](#)
23. Li, Y. Particle swarm optimization based neural network automatic controller for stability steering control of four-wheel drive electric vehicle. *Front. Mech. Eng.* **2024**, *10*, 1378175. [\[CrossRef\]](#)
24. Gong, Z.; Tuo, P.; Zheng, C.; Wu, X. Design and Analysis of the Model Predictive Control Implemented by the ANN Technique for MMC-Based Rectifier with Improved Grid Adaptability. *IEEE Trans. Power Electron.* **2024**, *39*, 12306–12322. [\[CrossRef\]](#)
25. Ahmed, H.; Çelik, D. Sliding mode based adaptive linear neuron proportional resonant control of Vienna rectifier for performance improvement of electric vehicle charging system. *J. Power Sources* **2022**, *542*, 231788. [\[CrossRef\]](#)
26. Okilly, A.H.; Kim, C.; Kim, D.-W.; Baek, J. Control and modeling framework for balanced operation and electro-thermal analysis in three-level T-type neutral point clamped inverters. *Energies* **2025**, *18*, 5587. [\[CrossRef\]](#)

27. Aiello, G.; Cacciato, M.; Scarcella, G.; Scelba, G.; Gennaro, F.; Aiello, N. Real-time emulation of a three-phase Vienna rectifier with DC voltage control and power factor correction. *Electr. Eng.* **2020**, *102*, 97–106. [[CrossRef](#)]
28. Wang, J.; Wang, G.; Wang, Q.; Zhang, Z.; Li, B.; Shao, H. Neutral Point Voltage Fluctuation Suppression for Electrolytic Capacitorless Vienna Rectifiers Based on Optimal Duty Cycle Model Predictive Control. *IEEE Trans. Power Electron.* **2024**, *39*, 13262–13273. [[CrossRef](#)]
29. Costa, I.D.L.; Brandao, D.I.; Matakas Junior, L.; Simões, M.G.; Morais, L.M.F. Analysis of Stationary- and Synchronous-Reference Frames for Three-Phase Three-Wire Grid-Connected Converter AC Current Regulators. *Energies* **2021**, *14*, 8348. [[CrossRef](#)]
30. Eberlein, S.; Rudion, K. Impact of Inner Control Loops on Small-Signal Stability and Model-Order Reduction of Grid-Forming Converters. *IEEE Trans. Smart Grid* **2023**, *14*, 2812–2824. [[CrossRef](#)]
31. Zou, Y.; Zhang, L.; Xing, Y.; Zhang, Z.; Zhao, H.; Ge, H. Generalized Clarke Transformation and Enhanced Dual-Loop Control Scheme for Three-Phase PWM Converters Under the Unbalanced Utility Grid. *IEEE Trans. Power Electron.* **2022**, *37*, 8935–8947. [[CrossRef](#)]
32. Lai, R.; Wang, F.; Burgos, R.; Boroyevich, D.; Jiang, D.; Zhang, D. Average modeling and control design for VIENNA-type rectifiers considering the DC-link voltage balance. *IEEE Trans. Power Electron.* **2009**, *24*, 2509–2522. [[CrossRef](#)]
33. Wang, Y.; Li, Y.; Guo, X.; Huang, S. Hybrid Space Vector PWM Strategy for Three-Phase VIENNA Rectifiers. *Sensors* **2022**, *22*, 6607. [[CrossRef](#)] [[PubMed](#)]
34. Rajendran, G.; Vaithilingam, C.A.; Misron, N.; Naidu, K.; Ahmed, M.R. Voltage Oriented Controller Based Vienna Rectifier for Electric Vehicle Charging Stations. *IEEE Access* **2021**, *9*, 50798–50809. [[CrossRef](#)]
35. Nussbaumer, T.; Heldwein, M.L.; Gong, G.; Round, S.D.; Kolar, J.W. Comparison of Prediction Techniques to Compensate Time Delays Caused by Digital Control of a Three-Phase Buck-Type PWM Rectifier System. *IEEE Trans. Ind. Electron.* **2008**, *55*, 791–799. [[CrossRef](#)]
36. Pan, D.; Ruan, X.; Bao, C.; Li, W.; Wang, X. Capacitor-current-feedback active damping with reduced computation delay for improving robustness of LCL-type grid-connected inverter. *IEEE Trans. Power Electron.* **2014**, *29*, 3414–3427. [[CrossRef](#)]
37. Shami, T.M.; El-Saleh, A.A.; Alswaitti, M.; Al-Tashi, Q.; Summakieh, M.A.; Mirjalili, S. Particle Swarm Optimization: A Comprehensive Survey. *IEEE Access* **2022**, *10*, 10031–10061. [[CrossRef](#)]
38. Shi, Y.; Eberhart, R. A Modified Particle Swarm Optimizer. In *Proceedings of the IEEE International Conference on Evolutionary Computation (ICEC), Anchorage, AK, USA, 4–9 May 1998*; IEEE: New York, NY, USA, 1998; pp. 69–73. [[CrossRef](#)]
39. Rumelhart, D.E.; Hinton, G.E.; Williams, R.J. Learning representations by back-propagating errors. *Nature* **1986**, *323*, 533–536. [[CrossRef](#)]
40. Wang, Q.; Xi, H.; Deng, F.; Cheng, M.; Buja, G. Design and analysis of genetic algorithm and BP neural network based PID control for boost converter applied in renewable power generations. *IET Renew. Power Gener.* **2022**, *16*, 1336–1344. [[CrossRef](#)]
41. Hasan, M.M.; Rana, M.S.; Tabassum, F.; Pota, H.R.; Roni, M.H.K. Optimizing the initial weights of a PID neural network controller for voltage stabilization of microgrids using a PEO-GA algorithm. *Appl. Soft Comput.* **2023**, *147*, 110771. [[CrossRef](#)]
42. Dhanasekaran, B.; Kaliannan, J.; Baskaran, A.; Dey, N.; Tavares, J.M.R.S. Load Frequency Control Assessment of a PSO-PID Controller for a Standalone Multi-Source Power System. *Technologies* **2023**, *11*, 22. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.