



Quantum parallel multi-layer Monte Carlo optimization algorithm for controller parameters optimization of doubly-fed induction generator-based wind turbines



Kunlun Han, Tianwei Huang, Linfei Yin*

College of Electrical Engineering, Guangxi University, Nanning, Guangxi, 530004, China
Institute of Artificial Intelligence, Guangxi University, Nanning, Guangxi, 530004, China

ARTICLE INFO

Article history:

Received 5 January 2021
Received in revised form 2 August 2021
Accepted 8 August 2021
Available online 17 August 2021

Keywords:

Quantum parallel multi-layer Monte Carlo optimization algorithm
Qubit probability amplitude
Maximum power point tracking
Doubly-fed induction generator-based wind turbines

ABSTRACT

With larger searching space, intelligent algorithms could have insufficient global search capability and accuracy for optimization problems. This paper proposes a novel quantum parallel multi-layer Monte Carlo optimization algorithm (QPMMCOA) to optimize the rotor-side controller (RSC) parameters based on proportional–integral of a doubly-fed induction generator (DFIG) for achieving maximum power and improving generation efficiency. The QPMMCOA is proposed to find optimal solutions in an accurate small searching space. The QPMMCOA combines qubit probability amplitude with Monte Carlo random numbers to generate a diverse population and expand accurate search space. The optimization process of the QPMMCOA with strong global search ability and local development ability is divided into the rough search, precise search, and re-precise search. These three search processes are searched in the ever-shrinking feasible region. The QPMMCOA mainly seeks the optimized solution by continuously changing and narrowing the feasible region. The QPMMCOA is utilized to optimize a discontinuous step function and a multimodal function for confirming the efficacy and feasibility. With broader exploration and deeper development capabilities, the QPMMCOA can achieve the optimization result of the fitness function of the RSC is at least 0.51% lower than other algorithms. Compared with other algorithms, the QPMMCOA-based RSC can enhance the average power coefficient of the DFIG-based wind power system by at least 0.0028%.

© 2021 Elsevier B.V. All rights reserved.

1. Introduction

In the context of economic globalization, energy demand has increased substantially along with the development of industries and emerging enterprises [1]. However, the ever-decreasing reserves of non-renewable energy are in contradiction with the ever-expanding energy demand [2]. Therefore, with the energy crisis accompanied by environmental deterioration, the development and research of renewable energies attract more and more scholars and researchers [3]. Wind energy is one major energy of renewable energies [4]. Wind power generation plays an important role in adjusting energy structure, reducing environmental pollution, and solving the problems of electricity consumption in remote areas [5]. Thus, both developing and developed countries are vigorously developing wind power [6].

The method of generating power through a doubly-fed induction generator (DFIG) does not require strictly synchronous [7].

Consequently, when the output frequency of a DFIG is independent of rotor speed, the rigid connection between the electromechanical systems turns flexible [8]. Therefore, the DFIG-based wind power generation systems with no direct connection between rotor speed and frequency is a research hot spot and inevitable development trend in wind power generation [9]. Maximum power point tracking (MPPT) is one major task of the DFIG with low wind speed [10].

The control performance of the nonlinear DFIG control system requires strong robustness and dynamic characteristics [11]. The vector control method based on the proportional–integral (PI) loop has the advantages of an easy-to-implement control framework that eliminates the coupling relationship between active and reactive power [12]. Therefore, the PI framework combined with vector control has been widely adopted in industrial applications, such as wind power generation systems based on DFIG [13]. The MPPT can be realized by improving the control performance of the rotor-side controller (RSC) of a DFIG under a low real-life wind speed. However, the control process and the mathematical system model of the DFIG are very complicated [14]. The complexity of the DFIG-based wind turbines

* Corresponding author at: College of Electrical Engineering, Guangxi University, Nanning, Guangxi, 530004, China.

E-mail address: yinlinfei@163.com (L. Yin).

(DFIG-WTs) control system leads to difficulty in tuning the PI parameters on the RSC. The intelligent optimization algorithm can be employed to optimize the PI parameters of the RSC. Intelligent optimization methods have many advantages, such as not relying on models, not relying on expert experience, derivation-free mechanism, and local optima avoidance [15]. In recent years, intelligent optimization algorithms have been widely developed in various optimization fields [16]. In Ref. [17], a nature-inspired approach has been developed for controller parameters optimization of DFIG energy systems. Ref. [18] utilizes a genetic algorithm (GA) to solve the converter PI parameters optimization in the grid-connected DFIG. The optimized converter can improve the stability of the single unit, extend the service life of the single unit, and expand the stable area of the wind farm. Ref. [19] combines GA and particle swarm algorithm (PSO) for proportional–integral–derivative (PID) parameters; the PSO with a short calculation time and a simple iterative structure possesses an outstanding performance in simulation experiments. In Ref. [20], the controller parameters of the DFIG are selected by the PSO. The whale optimization algorithm (WOA) has been applied in Ref. [21] to optimize the resource allocation problem of wireless networks; the WOA gained certain applicability for various potential resource allocations in 5G wireless networks and other networks. Low-frequency torsional oscillations can be avoided by tuned PI controllers based on the WOA [22]. Based on the gray wolf optimizer, a grouped gray wolf optimizer (GGWO) has been proposed [23]; the proposed algorithm with an improved hierarchical cooperation mechanism of the grey wolves can effectively optimize the PI parameters and improve power extraction of DFIG systems. A method to tune the optimal PI parameters named the chemical reaction optimization algorithm has been proposed to ensure the MPPT of DFIG-WTs [24]. However, the compared algorithms (i.e., GA, PSO, WOA, and GGWO) may lead to premature convergence because of insufficient development and exploration capabilities. Therefore, the compared algorithms may not be readily obtained the optimal global RSC PI parameters of DFIG-WTs.

The Monte Carlo method has been applied in the controller parameters tuning of synchronous motor systems [25]. In Ref. [26], the Monte Carlo algorithm has been applied to the parameter optimization of the fin stabilization controller for achieving a satisfactory anti-rolling effect. The disadvantage of the simple Monte Carlo method is that the diversity of the population is insufficient during the optimization process. The simple Monte Carlo algorithm leads to deficiencies in optimization exploration and development. Certain quantum mechanics concepts have been quoted in optimization algorithms, such as qubits, quantum states, and quantum superposition states [27]; after applying the quantum mechanisms, the optimization algorithm can improve the diversity and global convergence of the population. After learning from related concepts in quantum mechanisms, this paper employs the qubit probability amplitude in quantum theory combined with the Monte Carlo method to construct a quantum parallel multi-layer Monte Carlo algorithm (QPMMCOA), which is utilized for optimizing the RSC PI parameters of DFIG-WTs. The novelty of the QPMMCOA can be listed as (i) different from the conventional quantum mechanics combined with the Monte Carlo method, the proposed algorithm draws on the concept of qubit probability amplitude and combines Monte Carlo random numbers to generate the optimization process population; (ii) the proposed algorithm adopts layer structure in the feasible region, continuously changes and reduces the feasible region to realize the optimization process; (iii) under rough search, precise search, and re-precise search, the proposed algorithm has strong global search capability and optimization accuracy. In this paper, the QPMMCOA is applied to search the optimal global PI parameters, which enable the controller to complete excellent control

performances. The QPMMCOA optimizes the PI parameters of the control system through numerical simulation. The QPMMCOA generates PI parameters with a certain regularity and randomness, substitutes the generated PI parameters into the control system, and runs. Different PI parameters produce different control responses to the control system. With the optimization of the QPMMCOA, the PI parameters with accurate control response are finally taken as the optimization result of the QPMMCOA. The control response based on the rotor angular velocity deviation and reactive power deviation can indirectly reflect the control effect of the PI parameters on the control system.

The main contributions of this paper are as follows.

- (1) A novel algorithm named QPMMCOA is proposed. The characteristic of the QPMMCOA can generate the population with diversity by combining the qubit probability amplitude with the Monte Carlo random number method. The diversity of the population in the optimization process can expand the search space and avoid premature phenomenon to a certain extent. The QPMMCOA is divided into rough search layers, precise search layers, and re-precise search layers in the optimization process, resulting in strong global and local search abilities. The effectiveness and feasibility of the proposed algorithm are verified by a discontinuous step function and a multimodal function.
- (2) Conventional intelligent algorithms (i.e., GA, PSO, WOA, and GGWO) have been applied to the optimization in the PI parameters of the RSC. In optimizing the PI parameters of the RSC, conventional intelligent algorithms may not obtain the optimal global parameters because of insufficient population diversity or inadequate exploration and development capability. However, the QPMMCOA can solve the challenge of optimizing the PI parameters globally.
- (3) The control performance of the RSC optimized by the QPMMCOA is stronger than other compared algorithms. The RSC optimized by the QPMMCOA has a small overshoot, low fluctuation, and satisfactory wind energy conversion efficiency. Therefore, the proposed algorithm allows a DFIG-based wind turbine to use wind energy and generate more electricity fully.

The following content is divided into four contents. A detailed introduction of the QPMMCOA is given in Section 2. The QPMMCOA for DFIG-WTs is described in Section 3. Compared with the other four algorithms (i.e., GA, PSO, WOA, and GGWO), the QPMMCOA is verified in Section 4. The conclusions are given in Section 5.

2. Quantum parallel multi-layer Monte Carlo optimization algorithm

A nested partition optimization method with the way of narrowing the feasible region can effectively solve the optimization problem [28]. The shrinking space technology completes the search work in a specific area of the feasible region by narrowing the constrained search space has been proposed by Aguirre et al. [29]. In the linear programming narrowing technique, the method of narrowing feasible regions has been adopted to realize the search for the solution [30]. In recent years, quantum mechanics concepts and laws such as qubits, quantum states, and quantum superposition states have aroused great attention and interest [31]. This paper combines the concept of quantum bits in quantum mechanics, the Monte Carlo method, and the idea of narrowing the feasible region method for proposing a QPMMCOA. The proposed algorithm implements the optimization process by continuously narrowing the feasible region. The QPMMCOA of narrowing the feasible region is closely related to

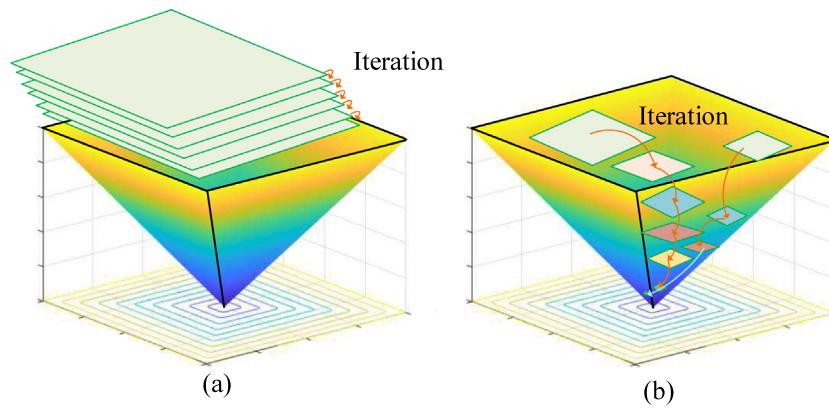


Fig. 1. Searching regions at each iteration: (a) other intelligent algorithms; (b) QPMMCOA.

the optimization point of the previous generation and belongs to the evolutionary type similar to the genetic algorithm. With more time searching in a small feasible region, the QPMMCOA can find the optimal global solution easier than other compared methods (Fig. 1). That is to say, the QPMMCOA keeps narrowing the search space of the optimal solution to approach the global optimal solution and then achieves more accurate optimization results.

2.1. Mathematical model of continuous function optimization

A point or vector is usually regarded as the solution for continuous n -dimensional space optimization problems [32]. A continuous optimization problem can be defined as

$$\begin{cases} \min f(x) = f(x_1, x_2, \dots, x_n) \\ a_k \leq x_k \leq b_k; k = 1, 2, \dots, n \end{cases} \quad (1)$$

where $[a_k, b_k]$ is the domain of the variable x_k ; f is the objective function; x is a point or a vector; n represents the spatial dimension. The value of f is usually applied to reflect the fitness value of a point or vector. The optimal solution can be the x which renders the $f(x)$ value optimal.

2.2. Qubit of quantum computing

Information in quantum computing can be represented through qubits. The state of the qubit is generally shown as [33]:

$$|\psi\rangle = \alpha|0\rangle + \mu|1\rangle, \quad (2)$$

where both α and μ mean the probability amplitudes of qubits; and $|\alpha|^2 + |\beta|^2 = 1$. Let $\alpha = \cos \theta$, $\mu = \sin \theta$; where θ denotes the phase. A qubit can be represented in the form of a probability amplitude $[\cos \theta \quad \sin \theta]^T$.

2.3. Population initialization

In the QPMMCOA, the qubit probability amplitude is directly applied as the initialization of the individuals in the population. The individuals in the population are randomly selected. The coding scheme of the population is expressed as

$$q_j = \begin{bmatrix} \left| \cos(\theta_{j1}) \quad \cos(\theta_{j2}) \right| \dots \left| \cos(\theta_{jn}) \right| \\ \left| \sin(\theta_{j1}) \quad \sin(\theta_{j2}) \right| \dots \left| \sin(\theta_{jn}) \right| \end{bmatrix}, j = 1, 2, \dots, m; \\ k = 1, 2, \dots, n \quad (3)$$

where $\theta_{jk} = 2\pi \cdot r$; r is a Monte Carlo random number from 0 to 1; m denotes the population size.

In Eq. (3), each random individual contains two positions during the traversal space. The searched space can be doubled with a constant population size. The search space can be doubled to avoid premature maturity and improve the global optimum. Two positions of each random individual are calculated as

$$\begin{cases} q_{j0} = (\cos(\theta_{j1}), \cos(\theta_{j2}), \dots, \cos(\theta_{jn})) \\ q_{j1} = (\sin(\theta_{j1}), \sin(\theta_{j2}), \dots, \sin(\theta_{jn})) \end{cases} \quad (4)$$

where q_{j0} represents the position of the $|0\rangle$ -state; q_{j1} represents the position of the $|1\rangle$ -state.

A solution space transformation is required to examine the pros and cons of each current location of the individual. The transformation of the solution space maps the two positions occupied by each individual from $I = [-1, 1]^n$ to solution spaces.

Assume that the domain of x_k is $[a_k, b_k]$ and the k th qubit on the individual q_j is $[\cos \theta_{jk} \quad \sin \theta_{jk}]^T$, the corresponding solution space variable is expressed as

$$\begin{bmatrix} x_{j0}^k \\ x_{j1}^k \end{bmatrix} = \begin{bmatrix} \frac{b_k - a_k}{2} & 0 \\ 0 & \frac{b_k - a_k}{2} \end{bmatrix} \begin{bmatrix} \cos \theta_{jk} \\ \sin \theta_{jk} \end{bmatrix} + \begin{bmatrix} \frac{b_k + a_k}{2} \\ \frac{b_k + a_k}{2} \end{bmatrix}. \quad (5)$$

Therefore, the two solutions in the optimization issue can correspond to an individual in the population.

2.4. Implementation steps of quantum parallel multi-layer Monte Carlo optimization algorithm

The execution process of the QPMMCOA has three steps: rough search, precise search, and re-precise search (Fig. 2); the implementation steps of QPMMCOA in Fig. 2 are based on $n = 2$, that is, the feasible regions are two-dimensional space. The total number of layers of the proposed algorithm is M ; the number of rough search layers is $M_1 = [k_1 \cdot M]$, the number of precise search layers is $M_2 = [k_2 \cdot M]$, the number of re-precise search layers is $M_3 = [k_3 \cdot M]$; where $k_1 + k_2 + k_3 = 1$.

2.4.1. Rough search

When the QPMMCOA enters the rough search ($L \leq M_1$, where L represents current layer or current number of iterations), first randomly searches for m points in the initial feasible region that are combined with f to calculate the fitness values (the initial feasible region of the first layer ($L = 1$) is the feasible region of the problem), and then save the first u optimized points (Fig. 2). A new feasible region is defined based on the optimal point among

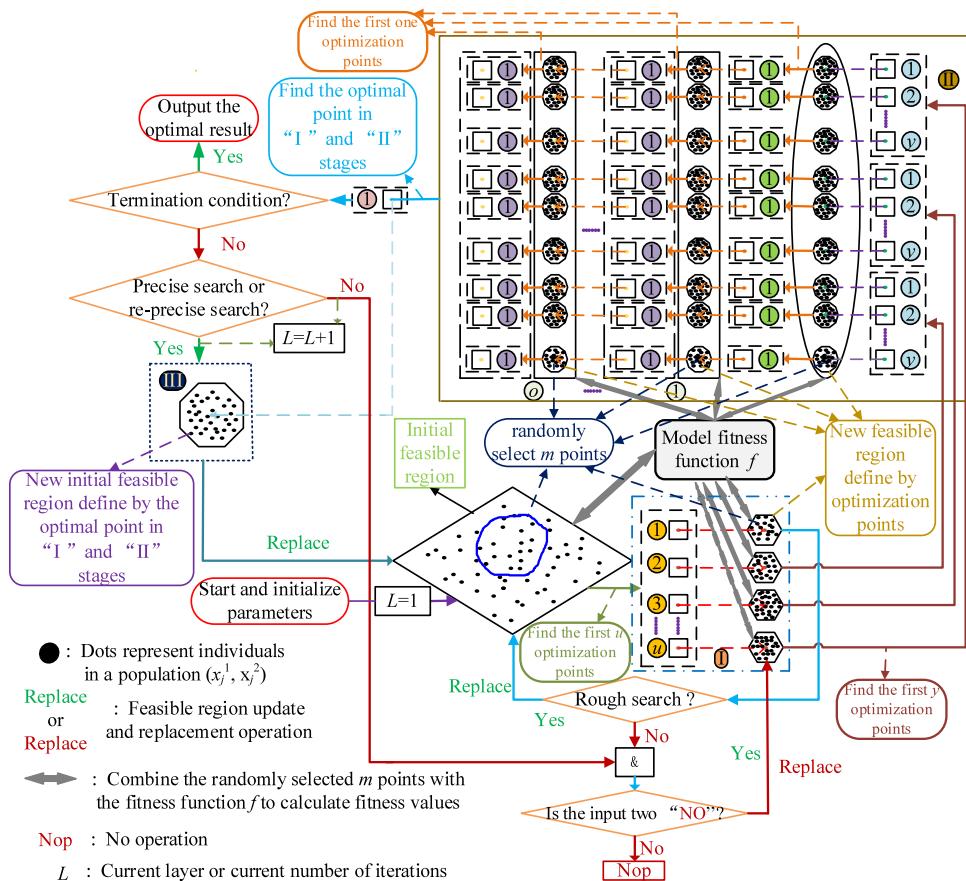


Fig. 2. Framework of QPMMCOA.

u points and regarded as the initial feasible region of the next iteration process in rough search process. In stage "I" in Fig. 2, the $u - 1$ new feasible regions are defined around the remaining $u - 1$ points saved in the initial feasible region. Then, the $u - 1$ new feasible regions are entered the search of "II" stage; the search process of the $u - 1$ feasible regions is similar (stage "II" in Fig. 2). The search process for one of the $u - 1$ new feasible regions is: (i) randomly select m points that are combined with f to calculate the fitness values in the feasible region, and then save the first y optimized points; (ii) define y feasible regions according to these y optimized points; (iii) randomly pick m points that are combined with f to calculate the fitness values in these y feasible regions separately, and then save the optimal one point of each feasible region; (iv) according to the optimal point of each feasible region, define y new feasible regions again, randomly select m points that are combined with f to calculate the fitness values separately again, and then save the optimal point of these new feasible regions separately again (this process is looped o times). After completing all searches in the "II" stage, save the optimal point of the entire search stage, and then the optimal point is the optimization result of the current layer.

In the rough search process of the QPMMCOA. Start from the second layer, the initial feasible region of each layer has the following characteristics (stage "I" in Fig. 2): (i) the initial feasible region of the current layer is related to the optimal point among the u saved points in the initial feasible region of the previous layer; (ii) a new feasible region is defined around the optimal point among the u saved points in the initial feasible region of the previous layer; (iii) the new feasible region is regarded as the initial feasible region of the current layer.

The main purpose of the rough search is to find a rough optimized solution in the feasible region of the problem. In the rough search, the initial feasible region of each layer is associated with the optimal point among the u saved points in the initial feasible region of the previous layer; the definition method of the initial feasible region leads to the proposed algorithm have a global search performance and avoid falling into the local optimum quickly. Multiple optimization points are employed to form the new feasible regions in the rough search process and increase the global search capability and local development capability.

The pseudocode for the rough search is listed in Algorithm 1.

In Algorithm 1, u is the number of optimization points saved in the initial feasible region (stage "I" in Fig. 2); y is the number of optimization points saved in the new feasible region (stage "II" in Fig. 2); o is the number of loops in optimization search process (stage "II" in Fig. 2). Variables L_{Bk} and U_{Bk} are employed to save the lower and upper bounds of the variables in the feasible region of the optimized problem; L_{Bk} and U_{Bk} are employed to define the initial or new feasible regions (reduced feasible regions) in the optimization process, and are employed to limit the initial or new feasible region in the feasible region of the original problem; with the increase of L , that is, the increase of the number of iterations, the range of the initial or new feasible regions of each layer is randomly shrunk. Variables $c_{\text{shrink},1}$, $c_{\text{shrink},2}$, and $c_{\text{shrink},3}$ represent the reduction factors when defining the initial or new feasible regions; after testing and adjustment, they are set to 4, 8, and 10 respectively in this article.

2.4.2. Precise search

In the precise search, the search process is roughly similar to that of the rough search. One aspect that is different from

Algorithm 1. Optimization procedure of rough search.

```

1: Initialize optimization parameters for rough search:  $m$ ,  $u$ ,  $y$ ,  $o$ ,  $M$ ,  $M_2 = k_2 \cdot M$ ,  $L_{Bk} = a_k$ ,  $U_{Bk} = b_k$ ,
 $M_1 = k_1 \cdot M$ ,  $ii = 2$ ,  $kk = 1$ ,  $aa = 1$ ,  $cc = 1$ ,  $M_3 = M - M_1 - M_2$ ,  $L = 1$ .
2: for the number of layers of optimization algorithm  $L = 1 : M_1$  do
3: The solutions obtained by initializing Eq. (5) are taken into Eq. (1) to obtain the fitness values of each
individual in the population.
4: The fitness values are employed to sort in ascending order. Then, save the first  $u$  fitness values and
their corresponding solutions ( $x_e^k$ ,  $e = 1, 2, \dots, u$ ).
5: for  $ii = 2 : u$  do
6:    $a_k = \max\{x_{ii}^k - ((U_{Bk} - L_{Bk}) / c_{shrink,1} - ((U_{Bk} - L_{Bk}) / c_{shrink,1}) \cdot L / M_1), L_{Bk}\}$ 
    $b_k = \min\{x_{ii}^k + ((U_{Bk} - L_{Bk}) / c_{shrink,1} - ((U_{Bk} - L_{Bk}) / c_{shrink,1}) \cdot L / M_1), U_{Bk}\}$ 
7: Combine Step 6 with Eq. (5) to generate a new population. Combine the new population with Eq. (1)
to achieve the new fitness values population.
8: Sort fitness values of Step 7 from small to large. Then, save the first  $y$  values and their corresponding
solutions ( $x_{jj+y-(ii-2)}^k$ ,  $jj = 1, 2, \dots, y$ ).
9: end for
10: for  $kk = 1 : y \cdot (u-1)$  do
11:    $a_k = \max\{x_{kk}^k - ((U_{Bk} - L_{Bk}) / c_{shrink,2} - ((U_{Bk} - L_{Bk}) / c_{shrink,2}) \cdot L / M_1), L_{Bk}\}$ 
    $b_k = \min\{x_{kk}^k + ((U_{Bk} - L_{Bk}) / c_{shrink,2} - ((U_{Bk} - L_{Bk}) / c_{shrink,2}) \cdot L / M_1), U_{Bk}\}$ 
12: Combine Step 11 with Eq. (5) to achieve a new population. Combine the new population with Eq.
(1) to achieve the new fitness values population.
13: Sort fitness values of Step 12 from small to large. Then, save the smallest fitness value and
corresponding solution of the smallest fitness value ( $x_{kk}^k$ ).
14: end for
15: for  $aa = 1 : o$  do
16:   for  $cc = 1 : y \cdot (u-1)$  do
17:      $a_k = \max\{x_{cc}^k - ((U_{Bk} - L_{Bk}) / (c_{shrink,3} + aa) - ((U_{Bk} - L_{Bk}) / (c_{shrink,3} + aa)) \cdot L / M_1), L_{Bk}\}$ 
      $b_k = \min\{x_{cc}^k + ((U_{Bk} - L_{Bk}) / (c_{shrink,3} + aa) - ((U_{Bk} - L_{Bk}) / (c_{shrink,3} + aa)) \cdot L / M_1), U_{Bk}\}$ 
18:     Combine Step 17 with Eq. (5) to achieve a new population. Combine the new population with Eq.
(1) to achieve the new fitness values population.
19:     Sort fitness values of Step 18 from small to large. Then, save the smallest fitness value and
corresponding solution of the smallest fitness value ( $x_{cc}^k$ ).
20:   end for
21: end for
22: Combine  $x_i^k$  in Step 4 with Step 6 to obtain  $(a_k, b_k)$ . Then, combine  $(a_k, b_k)$  with Eq. (5) to generate
a new population.
23: end for
24: Finally, find the smallest fitness value in Step 2 to Step 23 and obtain the solution  $x_R^k$  of the smallest
fitness value. The solution  $x_R^k$  is the result of a rough search.

```

the rough search is that the initial feasible regions are defined differently. The initial feasible region in each layer of the precise search is related to the optimal point found in the search of the "I" and "II" stages of the previous layer (Fig. 2). The new initial feasible region is obtained with certain rules through the optimal point found in the search of the "I" and "II" stages in the previous layer; the new initial feasible region is regarded as the initial feasible region of the current layer, as shown in stage "III" in Fig. 2. Another aspect is that the stage "I" in Fig. 2 has changed. The new feasible region generated based on the optimal point among the u saved points in the initial feasible region is no longer replaced the initial feasible region, while the new feasible region generated based on the u th optimized point is replaced. The last aspect is that the rules for defining feasible regions are different.

The meaning of the precise search can perform a more precise local search based on the optimized solution of the rough search. Meanwhile, precise search defines different feasible regions for searching in the entire search process according to different optimization points. The purpose of different feasible regions can avoid searching only near one point and falling into the local optimum during the precise search.

The pseudocode for the precise search is listed in Algorithm 2.

2.4.3. Re-precise search

The search process of the re-precise search is similar to that of precise search, except for the different ways of defining initial and

new feasible regions. The purpose of the re-precise search can re-optimize the optimized solution obtained from the precise search, improve the accuracy of the solution, avoid the local optimum. The re-precise search ensures that the QPMMCOA has satisfactory global search performance.

The pseudocode for the re-precise search is listed in Algorithm 3.

Eqs. (6) and (7) are given in Box I.

The two piecewise functions defined in Eqs. (6)–(7) are employed to define the initial or new feasible regions in the re-precise search process. Two piecewise functions are defined to increase the accuracy of the optimization results; the constants in the two piecewise functions are obtained from multiple simulation tests; these constants do not need to be re-adjusted in the optimization problem of this paper.

3. Structure of wind turbines based on doubly-fed induction generators

Through enhancing RSC performances: (i) the rotation speed of the rotor is adjusted in real-time when the wind speed changes; (ii) the optimized controller allows the rotor speed to follow the optimal rotor speed as much as possible; (iii) the power generation efficiency of the DFIG-WT is improved; the operating conditions of the DFIG-WT are optimized.

Algorithm 2. Optimization procedure of precise search.

1: Initialize optimization parameters for rough search: m , u , y , o , M , $ii=1$, $kk=1$, $aa=1$, $cc=1$, $L=M_1+M_2+1$, $x_{\text{Mid}}^k = x_p^k$.

2: **for** the number of layers of optimization algorithm $L=M_1+1:M_1+M_2$ **do**

3: $a_k = \max\{x_{\text{Mid}}^k - (\frac{(U_{\text{Bk}} - L_{\text{Bk}})}{c_{\text{shrink},1}} - (\frac{(U_{\text{Bk}} - L_{\text{Bk}})}{c_{\text{shrink},1}}) \cdot \frac{(L-M_1)}{M_2}), L_{\text{Bk}}\}$
 $b_k = \min\{x_{\text{Mid}}^k + (\frac{(U_{\text{Bk}} - L_{\text{Bk}})}{c_{\text{shrink},1}} - (\frac{(U_{\text{Bk}} - L_{\text{Bk}})}{c_{\text{shrink},1}}) \cdot \frac{(L-M_1)}{M_2}), U_{\text{Bk}}\}$

4: Combine Step 3 with Eq. (5) to generate a population. Combine the obtained population with Eq. (1) for obtaining the fitness value of each individual in the population.

5: Ascending sort fitness values and corresponding solutions (x_e^k , $e=1, 2, \dots, u$).

6: **for** $ii=1:u-1$ **do**

7: $a_k = \max\{x_{ii}^k - (\frac{(U_{\text{Bk}} - L_{\text{Bk}})}{c_{\text{shrink},2}} - (\frac{(U_{\text{Bk}} - L_{\text{Bk}})}{c_{\text{shrink},2}}) \cdot \frac{(L-M_1)}{M_2}), L_{\text{Bk}}\}$
 $b_k = \min\{x_{ii}^k + (\frac{(U_{\text{Bk}} - L_{\text{Bk}})}{c_{\text{shrink},2}} - (\frac{(U_{\text{Bk}} - L_{\text{Bk}})}{c_{\text{shrink},2}}) \cdot \frac{(L-M_1)}{M_2}), U_{\text{Bk}}\}$

8: Combine Step 7 with Eq. (5) to generate a new population. Combine the new population with Eq. (1) to achieve the new fitness values population.

9: Sort fitness values of Step 8 from small to large. Then, save the first y values and their corresponding solutions ($x_{jj+y:(ii-2)}^k$, $jj=1, 2, \dots, y$).

10: **end for**

11: **for** $kk=1:y \cdot (u-1)$ **do**

12: $a_k = \max\{x_{kk}^k - (\frac{(U_{\text{Bk}} - L_{\text{Bk}})}{c_{\text{shrink},2}} - (\frac{(U_{\text{Bk}} - L_{\text{Bk}})}{c_{\text{shrink},2}}) \cdot \frac{(L-M_1)}{M_2}), L_{\text{Bk}}\}$
 $b_k = \min\{x_{kk}^k + (\frac{(U_{\text{Bk}} - L_{\text{Bk}})}{c_{\text{shrink},2}} - (\frac{(U_{\text{Bk}} - L_{\text{Bk}})}{c_{\text{shrink},2}}) \cdot \frac{(L-M_1)}{M_2}), U_{\text{Bk}}\}$

13: Combine Step 12 with Eq. (5) to achieve a new population. Combine the new population with Eq. (1) to achieve the new fitness values population.

14: Sort fitness values of Step 13 from small to large. Then, save the smallest fitness value and corresponding solution of the smallest fitness value (x_{kk}^k).

15: **end for**

16: **for** $aa=1:o$ **do**

17: **for** $cc=1:y \cdot (u-1)$ **do**

18: $a_k = \max\{x_{cc}^k - (\frac{(U_{\text{Bk}} - L_{\text{Bk}})}{c_{\text{shrink},3}} - (\frac{(U_{\text{Bk}} - L_{\text{Bk}})}{c_{\text{shrink},3}}) \cdot \frac{(L-M_1)}{M_2}), L_{\text{Bk}}\}$
 $b_k = \min\{x_{cc}^k + (\frac{(U_{\text{Bk}} - L_{\text{Bk}})}{c_{\text{shrink},3}} - (\frac{(U_{\text{Bk}} - L_{\text{Bk}})}{c_{\text{shrink},3}}) \cdot \frac{(L-M_1)}{M_2}), U_{\text{Bk}}\}$

19: Combine Step 18 with Eq. (5) to achieve a new population. Combine the new population with Eq. (1) to achieve the new fitness values population.

20: Sort fitness values of Step 19 from small to large. Then, save the smallest fitness value and corresponding solution of the smallest fitness value (x_{cc}^k).

21: **end for**

22: **end for**

23: Find the smallest fitness value in Step 1 to Step 22 and obtain the solution x_p^k of the smallest fitness value. Let $x_{\text{Mid}}^k = x_p^k$.

24: **end for**

25: Finally, the solution x_p^k is the result of the precise search.

3.1. Related principles of wind turbines

The mechanical power P_m generated via a wind turbine is given as [34]:

$$P_m = \frac{1}{2} \rho \pi R^2 C_p(\beta, \lambda) v_{\text{wind}}^3, \quad (8)$$

$$\lambda = R\omega_m/v_{\text{wind}}, \quad (9)$$

$$T_m = \frac{P_m}{\omega_m}, \quad (10)$$

where the air density is represented by ρ ; R represents the radius of the rotating blade of the wind turbine; C_p represents the wind energy conversion factor; v_{wind} indicates the wind speed; ω_m denotes the mechanical angular speed; λ , β and T_m indicate tip speed ratio, pitch angle, and mechanical torque, respectively.

The C_p can be derived from λ and β as

$$C_p(\lambda, \beta) = 0.5176 \left(\frac{116}{\lambda_b} - 0.4\beta - 5 \right) e^{-\frac{21}{\lambda_b}} + 0.0068\lambda, \quad (11)$$

$$\frac{1}{\lambda_b} = \frac{1}{\lambda + 0.08\beta} - \frac{0.035}{\beta^3 + 1}. \quad (12)$$

3.2. Motor model of doubly-fed induction generators

The mathematical model of the DFIG is given as [35]:

$$\begin{cases} u_{ds} = (-R_s - L_s p)i_{ds} + \omega_{sy} L_s i_{qs} + L_m p i_{dr} - \omega_{sy} L_m i_{qr} \\ u_{qs} = -\omega_{sy} L_s i_{ds} - (R_s + L_s p)i_{qs} + \omega_{sy} L_m i_{dr} + L_m p i_{qr}, \\ u_{dr} = -L_m p i_{ds} + \omega_{sr} L_m i_{qs} + (R_r + L_r p)i_{dr} - \omega_{sr} L_r i_{qr} \\ u_{qr} = -\omega_{sr} L_m i_{ds} - L_m p i_{qs} + \omega_{sr} L_r i_{dr} + (R_r + L_r p)i_{qr} \end{cases} \quad (13)$$

$$T_e = n_p L_m (i_{qs} i_{dr} - i_{ds} i_{qr}), \quad (14)$$

$$Q_s = u_{qs} i_{ds} - u_{ds} i_{qs} \quad (15)$$

where p is a differential operator; u_{ds} and u_{dr} are stator and rotor direct-axis voltages, respectively; u_{qs} and u_{qr} are stator and rotor quadrature-axis voltages, respectively; i_{ds} and i_{dr} are stator and rotor direct-axis currents, respectively; i_{qs} and i_{qr} are stator

Algorithm 3. Optimization procedure of re-precise search.

1: Initialize optimization parameters for rough search: m , u , y , o , M , $ii=1$, $kk=1$, $aa=1$, $cc=1$, $L=M_1+M_2+1$, $x_{\text{Mid}}^k = x_p^k$.

2: **for** the number of layers of optimization algorithm $L=M_1+M_2+1:M$ **do**

3: $a_k = \max\{x_{\text{Mid}}^k - P_{\text{rp}} \left(\frac{x_{\text{Mid}}^k}{10^{J(L)}} \right), L_{\text{Bk}}\}$, $b_k = \min\{x_{\text{Mid}}^k + P_{\text{rp}} \left(\frac{x_{\text{Mid}}^k}{10^{J(L)}} \right), U_{\text{Bk}}\}$.

4: Combine Step 3 with Eq. (5) to generate a population. Combine the obtained population with Eq. (1) for obtaining the fitness value of each individual in the population.

5: Ascending sort fitness values and corresponding solutions (x_e^k , $e=1, 2, \dots, u$).

6: **for** $ii=1:u-1$ **do**

7: $a_k = \max\{x_{ii}^k - P_{\text{rp}} \left(\frac{x_{ii}^k}{10^{J(L)}} \right), L_{\text{Bk}}\}$, $b_k = \min\{x_{ii}^k + P_{\text{rp}} \left(\frac{x_{ii}^k}{10^{J(L)}} \right), U_{\text{Bk}}\}$.

8: Combine Step 7 with Eq. (5) to generate a new population. Combine the new population with Eq. (1) to achieve the new fitness values population.

9: Sort fitness values of Step 8 from small to large. Then, save the first y values and their corresponding solutions ($x_{jj+y-(ii-2)}^k$, $jj=1, 2, \dots, y$).

10: **end for**

11: **for** $kk=1:y \cdot (u-1)$ **do**

12: $a_k = \max\{x_{kk}^k - P_{\text{rp}} \left(\frac{x_{kk}^k}{10^{J(L)}} \right), L_{\text{Bk}}\}$, $b_k = \min\{x_{kk}^k + P_{\text{rp}} \left(\frac{x_{kk}^k}{10^{J(L)}} \right), U_{\text{Bk}}\}$.

13: Combine Step 12 with Eq. (5) to achieve a new population. Combine the new population with Eq. (1) to achieve the new fitness values population.

14: Sort fitness values of Step 13 from small to large. Then, save the smallest fitness value and corresponding solution of the smallest fitness value (x_{kk}^k).

15: **end for**

16: **for** $aa=1:o$ **do**

17: **for** $cc=1:y \cdot (u-1)$ **do**

18: $a_k = \max\{x_{cc}^k - P_{\text{rp}} \left(\frac{x_{cc}^k}{10^{J(L)}} \right), L_{\text{Bk}}\}$, $b_k = \min\{x_{cc}^k + P_{\text{rp}} \left(\frac{x_{cc}^k}{10^{J(L)}} \right), U_{\text{Bk}}\}$, $J(L)$ and P_{rp} are calculated as Eq. (6) and Eq. (7), respectively.

19: Combine Step 18 with Eq. (5) to achieve a new population. Combine the new population with Eq. (1) to achieve the new fitness values population.

20: Sort fitness values of Step 19 from small to large. Then, save the smallest fitness value and corresponding solution of the smallest fitness value (x_{cc}^k).

21: **end for**

22: **end for**

23: Find the smallest fitness value in Step 1 to Step 22 and obtain the solution x_{RP}^k of the smallest fitness value. Let $x_{\text{Mid}}^k = x_{\text{RP}}^k$.

24: **end for**

25: Finally, the solution x_{RP}^k is the result of the re-precise search. The solution obtained after the re-precise search is the solution obtained by QPMMCOA.

and rotor quadrature-axis currents, respectively; R_s and R_r are stator and rotor resistances, respectively; L_s and L_r are stator and rotor self-inductances, respectively; mutual inductance is L_m ; ω_{sy} and ω_{sr} are synchronous and slip angular velocities of the motor, respectively; n_p denotes the number of pole pairs of the motor; T_e represents the electromagnetic torque of the generator; Q_s is the reactive power.

3.3. Shaft system model

The shaft system can usually be replaced by the concentrated inertia constant H_m . The concentrated inertia constant H_m can be obtained from the concentrated inertia constant H_t of the wind turbine and the concentrated inertia constant H_g of the generator [36]:

$$H_m = H_t + H_g, \quad (16)$$

The electromagnetic dynamic equation can be described as:

$$\frac{d\omega_m}{dt} = \frac{1}{2H_m}(T_m - T_e - D\omega_m). \quad (17)$$

where D is concentrated inertia damping.

3.4. Quantum parallel multi-layer Monte Carlo optimization algorithm-based control structure for doubly-fed induction generator

The QPMMCOA optimizes the RSC PI parameters of DFIG-WTs. The structure of utilizing QPMMCOA for the parameter optimization of the RSC of DFIG-WTs is given in Fig. 3. Relevant symbols are expressed as

$$\begin{cases} s = \frac{\omega_{sy} - \omega_r}{\omega_{sy}}, \sigma = 1 - \frac{L_m^2}{L_s L_r}, i_{ms} = \frac{v_{qs} - R_s i_{qs}}{\omega_{sy} L_m} \\ v_{qr2} = s \omega_{sy} \left(\sigma L_r i_{dr} + \frac{L_m^2 i_{ms}}{L_s} \right), \\ v_{dr2} = -s \omega_{sy} \sigma L_r i_{qr} \end{cases} \quad (18)$$

where σ is the leakage coefficient; ω_r means rotor electrical angular velocity, $\omega_r = \omega_m$; both v_{qr2} and v_{dr2} represent the compensation voltages.

To effectively optimize the PI parameters of the RSC, a fitness function needs to be found. In the DFIG-WT, the voltage stability is related to the reactive power Q_s stability and the rotor angular velocity error [37]. The relationship between the fitness function and the PI parameters is the implicit function. The DFIG-WT based on optimized RSC obtains the fitness function value. The fitness

$$J(L) = \begin{cases} 0 & [M \cdot (1 - k3)] < L \leq [M \cdot (1 - k3) + M \cdot k3 \cdot 0.142] \\ 1 & [M \cdot (1 - k3) + M \cdot k3 \cdot 0.142] < L \leq [M \cdot (1 - k3) + M \cdot k3 \cdot 0.284] \\ 2 & [M \cdot (1 - k3) + M \cdot k3 \cdot 0.284] < L \leq [M \cdot (1 - k3) + M \cdot k3 \cdot 0.432] \\ 3 & [M \cdot (1 - k3) + M \cdot k3 \cdot 0.432] < L \leq [M \cdot (1 - k3) + M \cdot k3 \cdot 0.574] \\ 4 & [M \cdot (1 - k3) + M \cdot k3 \cdot 0.574] < L \leq [M \cdot (1 - k3) + M \cdot k3 \cdot 0.716] \\ 5 & [M \cdot (1 - k3) + M \cdot k3 \cdot 0.716] < L \leq [M \cdot (1 - k3) + M \cdot k3 \cdot 0.0858] \\ 6 & [M \cdot (1 - k3) + M \cdot k3 \cdot 0.0858] < L \leq [M] \end{cases} \quad (6)$$

$$P_{rp}(|x|) = \begin{cases} 250 \cdot r & (1100 \leq |x|); \quad 50 \cdot r \quad (120 \leq |x| < 1100); \quad 13 \cdot r \quad (100 \leq |x| < 120) \\ 10 \cdot r & (60 \leq |x| < 100); \quad 6 \cdot r \quad (20 \leq |x| < 60); \quad 3 \cdot r \quad (8 \leq |x| < 20) \\ 1 \cdot r & (1 \leq |x| < 8); \quad 5 \times 10^{-1} \quad (10^{-1} \leq |x| < 1) \\ 5 \times 10^{-2} & (10^{-2} \leq |x| < 10^{-1}); \quad 5 \times 10^{-3} \quad (10^{-3} \leq |x| < 10^{-2}) \\ 5 \times 10^{-4} & (10^{-4} \leq |x| < 10^{-3}); \quad 5 \times 10^{-5} \quad (10^{-5} \leq |x| < 10^{-4}) \\ 5 \times 10^{-6} & (10^{-6} \leq |x| < 10^{-5}); \quad |x|/2 \quad (|x| < 10^{-7}) \end{cases} \quad (7)$$

Box I.

function can be described as

$$f_{\min}(\mathbf{x}) = \int_0^T (w_1 \cdot |\omega_r - \omega_r^*| + (1 - w_1) \cdot |Q_s - Q_s^*|) dt, \quad (19)$$

subject to

$$\begin{cases} K_{pi\min} < K_{pi} < K_{pi\max} \\ K_{li\min} < K_{li} < K_{li\max} \\ v_{wind\min} < v_{wind} < v_{wind\max}, \quad i = 1, 2, 3, 4. \\ v_{s\min} < v_s < v_{s\max} \\ Q_{s\min} < Q_s < Q_{s\max} \end{cases} \quad (20)$$

where both K_{pi} and K_{li} are the controller parameters; w_1 represents the weight of the rotor angular velocity error of the fitness function, $w_1 \in [0.5, 0.9]$. The value ranges of K_{pi} and K_{li} are $K_{p1} \in [100, 1000]$, $K_{l1} \in [0, 60]$, $K_{p2} \in [0, 0.02]$, $K_{l2} \in [0, 0.007]$, $K_{p3} \in [20, 1100]$, $K_{l3} \in [0, 120]$, $K_{p4} \in [1, 5]$ and $K_{l4} \in [0, 0.01]$, respectively. Variable T represents the operating time of the DFIG-WT. The system parameters appear through the per unit (p.u.) system. Wind speed v_{wind} , grid voltage v_s , and reactive power Q_s are configured as [8, 12] m/s, [0.2, 1.0] p.u. and [-1.0, 1.0] p.u., respectively.

The optimized parameter in this paper is the PI parameters of the RSC. Corresponding to the independent variables in Eq. (1), in this paper, $x_1 = K_{p1}$, $x_2 = K_{l1}$, $x_3 = K_{p2}$, $x_4 = K_{l2}$, $x_5 = K_{p3}$, $x_6 = K_{l3}$, $x_7 = K_{p4}$, $x_8 = K_{l4}$. When the optimization algorithm randomly takes a set of PI parameters (x_1, x_2, \dots, x_8) in the feasible region, sets this set of parameters in the RSC of the system, and finally runs the DFIG-WT simulation system to get the fitness value of Eq. (19).

In the QPMMCOA, eight parameters need to be set: the intermediate process parameters u , y , and o , the maximum number of layers M , the number of rough search layers M_1 , the number of precise search layers M_2 , the number of re-precise search layers M_3 , the number of population m . Appropriate parameter setting can alleviate the contradiction between the program running time and the effectiveness of the solution. After many simulation experiments, the QPMMCOA can obtain a satisfactory solution when $u = 3$, $y = 2$, $o = 1$, $M = 100$, $M_1 = 0.08 \cdot M$,

$M_2 = 0.2 \cdot M$, $M_3 = 0.72 \cdot M$, $m = 16$. The whole execution steps of the QPMMCOA for optimizing the RSC parameters are presented in Fig. 4.

The parameters of DFIG are configured as: $P_{rated} = 5$ MW, $R_s = 0.0005$ p.u., $L_m = 4$ p.u., $R_r = 1.1R_s$, $L_s = 1.01L_m$, $L_r = 1.005L_r$ [23]. The wind turbine parameters are: $\rho = 1.225$ kg/m³, $R = 58.6$ m, $v_{wind\max} = 12$ m/s, $\lambda_{opt} = 6.325$, $D = 0.05$ p.u., $H_m = 4.4$ s.

4. Case studies

4.1. Case study of two benchmark functions

This paper selects a discontinuous step function and a multimodal function to verify the effectiveness of the QPMMCOA. The discontinuous step function can be employed to test the effectiveness and convergence of the proposed algorithm [38]. The multimodal function can be utilized for testing the ability to avoid the local optimum [39]. This paper selects four comparison algorithms: GA, PSO, WOA, and GGWO.

The expression of the two functions is expressed as

$$\begin{cases} \min f_1(x) = - \sum_{k=1}^{n-1} [100(x_{k+1} - x_k^2)^2 + (x_k - 1)^2] \\ a_k \leq x_k \leq b_k; k = 1, 2, \dots, n; \\ a_k = -30, b_k = 30, n = 30. \end{cases} \quad (21)$$

$$\begin{cases} \min f_2(x) = - \sum_{k=1}^n (x_k \sin(\sqrt{|x_k|})) \\ a_k \leq x_k \leq b_k; k = 1, 2, \dots, n; \\ a_k = -500, b_k = 500, n = 30. \end{cases} \quad (22)$$

The minimum values $f_1(x)$ and $f_2(x)$ are 0 and -12569.50 with solutions $x_k = 1$ and $x_k = 420.9687$, respectively. In the discontinuous step and multimodal functions (i.e., $f_1(x)$ and $f_2(x)$). The parameters of GA, PSO, WOA, GGWO, and QPMMCOA are given in Table 1, where P_{mu} represents the probability of mutation; P_{cro} denotes cross probability; ω represents inertia weight; c_2 and c_3 are both acceleration factors; a_1 and a_w both are iterative attenuation coefficients of the PSO; b_w denotes logarithmic spiral

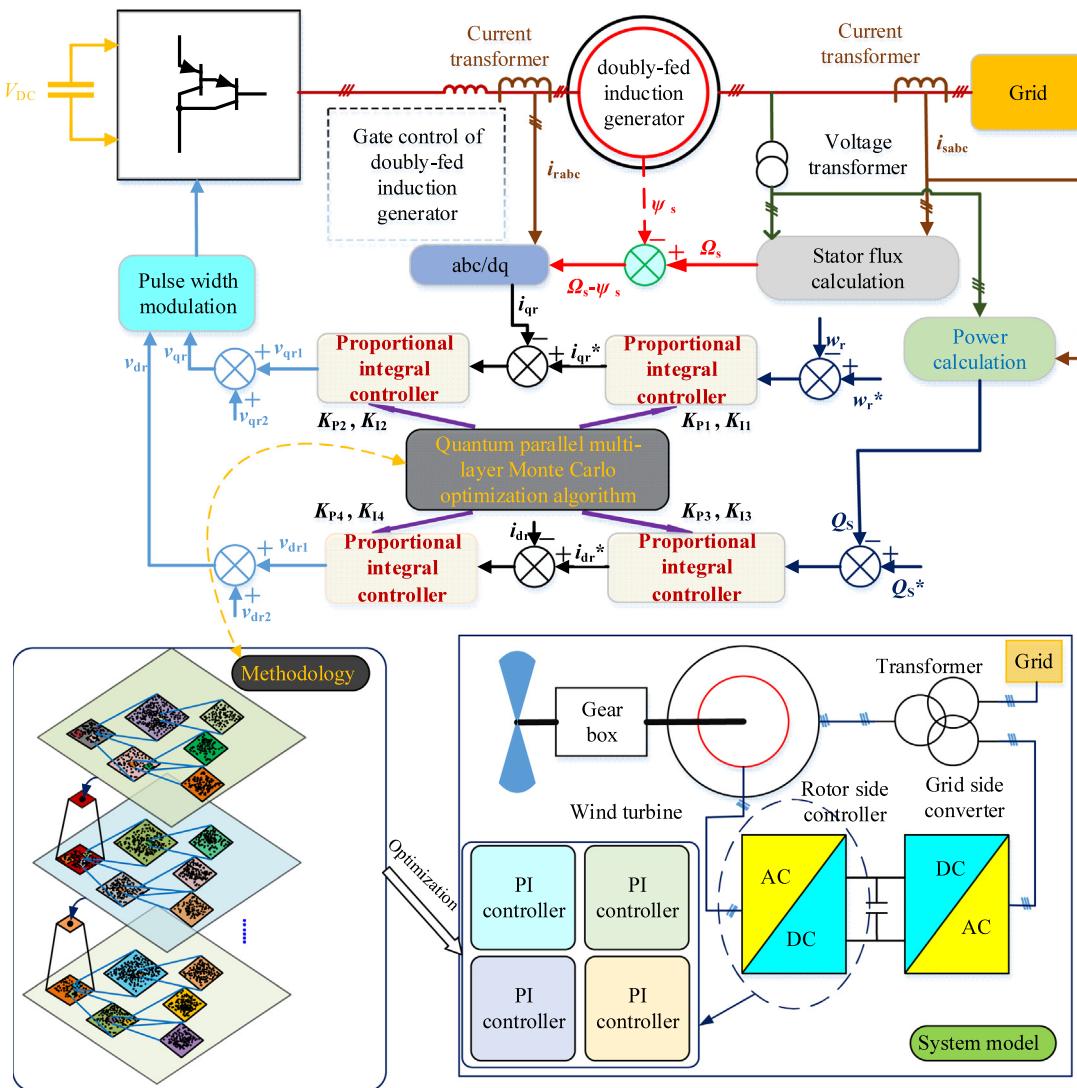


Fig. 3. Structure of QPMMCOA for DFIG-WTs.

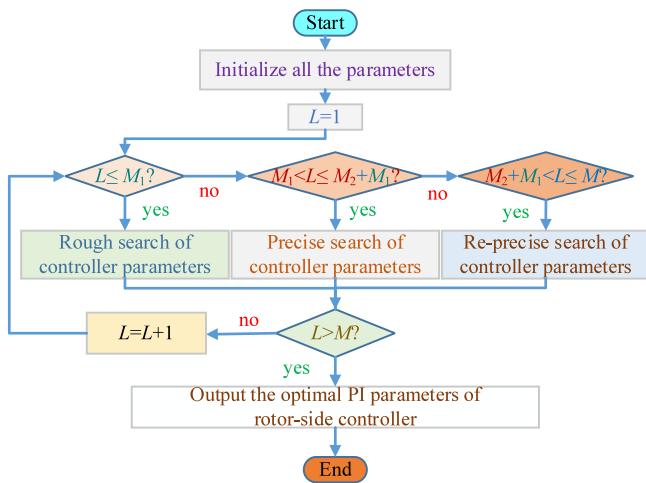


Fig. 4. Overall execution procedure of PI parameters of RSC optimized by QPMMCOA.

shape constant; l_w represents a random number between -1 and 1 ; k_α , k_θ and k_δ are leading factors.

Functions $f_1(x)$ and $f_2(x)$ are optimized by the five optimization algorithms mentioned above. To alleviate the inevitable contingency for the optimization, each algorithm has run 30 times when $f_1(x)$ and $f_2(x)$ are optimized. After each algorithm has been run 30 times, the median results obtained by each algorithm are selected for comparison. The median results of these five algorithms of their fitness convergence curves are shown in Fig. 5. In Fig. 5(a), the QPMMCOA is only slightly worse in convergence speed than the WOA. In optimizing the minimum $f_1(x)$, the optimization result of the QPMMCOA is closest to zero. In Fig. 5(b), the convergence speeds of the QPMMCOA in these five algorithms rank fourth. The QPMMCOA achieves the minimum $f_2(x)$. The QPMMCOA is satisfactory in terms of the convergence speed and the optimization of the minimum $f_1(x)$ and $f_2(x)$ (Fig. 5). Therefore, the effectiveness, convergence, and ability to escape from a local optimum of the developed algorithm have been verified.

The fourth column of the statistical results in Table 2 indicates that the optimization result of the QPMMCOA is closer to the theoretical value than other algorithms when optimizing the two benchmark functions. The statistical results show that the QPMMCOA has optimal local search capabilities and leads to more accurate optimization results. When optimizing the two benchmark functions, the statistical results show that the QPMMCOA

Table 1

Algorithm parameters setting for optimizing two benchmark functions.

Algorithm	Dimension	Iterations	Maximum number of evaluations	Other parameters
GA	30	500	165 000	$P_{\text{mu}} = 0.01, P_{\text{cro}} = 0.5$
PSO	30	500	165 000	$c_2 = 1.5, c_3 = 1.5, \omega = [0.4, 0.9]$
WOA	30	500	165 000	$l_w \in [-1, 1], b_w = 1, a_w \in [2, 0]$
GGWO	30	500	165 000	$a_1 \in [2, 0], k_\alpha = 0.3, k_\theta = 0.3, k_\delta = 0.4$
QPMMCOA	30	500	165 000	$k_1 = 0.3, k_2 = 0.4, k_3 = 0.3, m = 30 u = 3, y = 2, o = 1$

Table 2

Comparison of fitness indices of five algorithms in two benchmark functions.

Algorithm	Function	Average value (f_{av})	Error ($f_{\text{av}} - f_{\text{min}}$)	Standard deviation	Calculation time (s)
GA	$f_1(x)$	10.9472	10.9472	33.8536	14.9727
PSO	$f_1(x)$	19.7036	19.7036	20.2055	2.0043
WOA	$f_1(x)$	25.4200	25.4200	0.3841	0.5879
GGWO	$f_1(x)$	25.8569	25.8569	0.3026	1.1330
QPMMCOA	$f_1(x)$	2.1923	2.1923	1.6259	1.4303
GA	$f_2(x)$	-533.3937	12 036.1063	0.003449	3.2669
PSO	$f_2(x)$	-1514.0295	11 055.4705	351.0610	2.1123
WOA	$f_2(x)$	-10 397.6633	2171.8367	685.4382	0.5765
GGWO	$f_2(x)$	-7450.6997	5118.8003	1144.7724	1.2127
QPMMCOA	$f_2(x)$	-10 930.3084	1639.1916	760.6598	1.2303

Table 3

Parameter setting for optimizing Eq. (23).

Algorithm	Dimension	Iterations	Maximum number of evaluations	Other parameters
GA	8	100	8800	$P_{\text{mu}} = 0.01, P_{\text{cro}} = 0.5$
PSO	8	100	8800	$c_2 = 1.5, c_3 = 1.5, \omega = [0.4, 0.9]$
WOA	8	100	8800	$l_w \in [-1, 1], b_w = 1, a_w \in [2, 0]$
GGWO	8	100	8800	$a_1 \in [2, 0], k_\alpha = 0.3, k_\theta = 0.3, k_\delta = 0.4$
QPMMCOA	8	100	8800	$k_1 = 0.08, k_2 = 0.72, k_3 = 0.3, m = 8 u = 3, y = 2, o = 1$

is smaller and closer to the theoretical value with optimal global search capabilities than other algorithms. Therefore, the proposed algorithm has a more satisfactory global search ability and certain local search performance than other algorithms. Various indicators in Table 2 show that the QPMMCOA can be utilized to solve the minimum optimization problem credibly and effectively.

4.2. Case studies of proportional–integral parameters optimization of wind turbines

4.2.1. Parameters optimization of rotor-side controller of doubly-fed induction generators

In this section, the QPMMCOA is utilized to optimize the PI parameters of the RSC under three cases. The fitness function applied in the optimization is the sum value of Eq. (19) under three conditions.

$$f_{\text{min}}(\mathbf{x}) = \sum_{\text{case}=1}^3 \int_0^T (w_1 \cdot |\omega_r - \omega_r^*| + (1 - w_1) \cdot |Q_s - Q_s^*|) dt. \quad (23)$$

where $c_{\text{ase}} = 1$ means the step-change of wind speed; $c_{\text{ase}} = 2$ means randomly changing wind; $c_{\text{ase}} = 3$ means grid voltage drop; $w_1 = 0.7$; $Q_s^* = 0$.

The comparison algorithms with the QPMMCOA in this section are GA, PSO, WOA, GGWO. The parameters of these compared methods are shown in Table 3.

After all the algorithms (i.e., GA, PSO, WOA, GGWO, and QPMMCOA) optimize the fitness function Eq. (23) 10 times, the statistical results (Fig. 6(a)) show that the optimization result of the fitness function Eq. (23) obtained by the QPMMCOA is smaller than the other four algorithms. The minimum fitness value of each algorithm is selected from the statistical results. Then, compare the iterative convergence curve of each algorithm (Fig. 6(b)). The value of the fitness function Eq. (23) optimized by

Table 4

Statistical fitness values obtained by five algorithms in 10 runs.

Algorithm	Maximum	Minimum	Mean	Standard deviation	Calculation time (s)
GA	0.6696	0.3900	0.4937	0.08266	2.1024×10^4
PSO	0.3879	0.3531	0.3664	0.01314	2.5705×10^4
WOA	0.9433	0.3735	0.5278	0.1920	2.1203×10^4
GGWO	0.4022	0.3604	0.3774	0.01228	2.2605×10^4
QPMMCOA	0.3538	0.3513	0.3521	0.0007776	1.9838 $\times 10^4$

the QPMMCOA converges to a smaller value than the other four algorithms. The statistic results in Table 4 clearly show that the QPMMCOA has great advantages over the other four algorithms in optimizing the performance of fitness function Eq. (23). Table 5 shows the optimized PI parameters by each method and standard values selected from Ref. [23].

Different wind conditions are tested one by one. The test method is conducted separately under the conditions of each case in Eq. (23). Four indicators reflecting the control performance of the RSC are selected: the rotor angular velocity error w_{err} , the power coefficient C_p , the active power P , and the deviation of reactive power Q_{err} . Table 6 shows the average absolute value (AAV) indices of the different cases obtained by each method; Table 7 shows the performance indices include integral of absolute error (IAE), integral of squared error (ISE), integral time-weighted absolute error (ITAE), and integral of time multiplied by squared error (ITSE). Here $IAE_h = \int_0^T |h^* - h| dt$ and h^* is the reference value of the variable h ; $ISE_h = \int_0^T |h^* - h|^2 dt$; $ITAE_h = \int_0^T t|h^* - h| dt$; $ITSE_h = \int_0^T t|h^* - h|^2 dt$; $AAV_{h_1} = \frac{\int_0^T h_1 dt}{T}$; $h \in \{w_{\text{err}}, Q_{\text{err}}\}$ and $h_1 \in \{C_p, P\}$. The configured simulation time is set to 25 s.

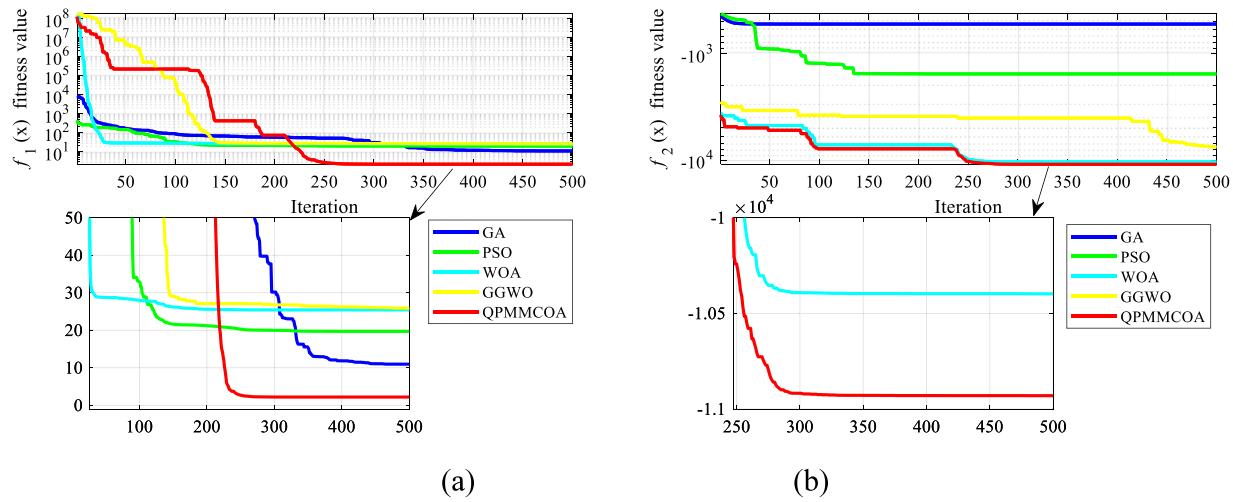


Fig. 5. Test results obtained by five algorithms in $f_1(x)$ and $f_2(x)$: (a) fitness convergence curve comparison of $f_1(x)$; (b) fitness convergence curve comparison of $f_2(x)$.

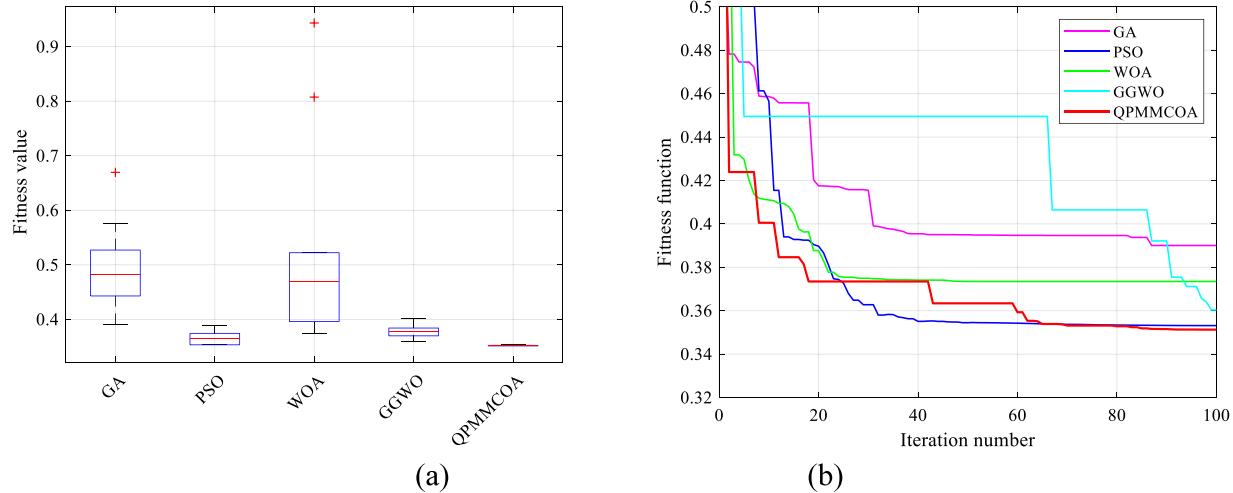


Fig. 6. Convergence of fitness function acquired through five algorithms and distribution statistics of 10 runs: (a) statistical results; (b) fitness convergence curve comparison.

Table 5

PI parameters obtained by five algorithms and standard PI parameters.

Algorithm	K_{P1}	K_{I1}	K_{P2}	K_{I2}	K_{P3}	K_{I3}	K_{P4}	K_{I4}
GA	850.6069	35.2565	0.008552	0.002448	39.8231	39.1373	0.6366	0.005779
PSO	999.8634	53.8381	0.006508	0.002319	6.6720	32.2307	1.2612	0.007615
WOA	1000.0000	60.0000	0.007747	0.002218	166.4266	120.0000	0.1581	0.001471
GGWO	1000.0000	51.6285	0.007703	0.002252	26.8934	40.3920	0.9812	0.005648
QPMMCOA	999.9960	59.9527	0.006402	0.002329	7.1131	32.2694	1.3570	0.007702
Par-form-Re. [23]	844.6700	39.9900	0.009830	0.003120	117.1900	34.9200	0.0231	0.002600

4.2.2. Comparison of control effects between optimized proportional-integral parameters and standard proportional-integral parameters

The seventh row in Table 5 is a set of unoptimized PI parameters randomly selected in the feasible region. The unoptimized PI parameters are tested together with the optimized PI parameters under different cases.

Stepped wind conditions reflect the abrupt change of wind speed, starting at 8 m/s with three successive step changes to simulate the sudden wind speed change in stages (Fig. 7(a)). The random wind condition is reflected via the randomly changing wind (Fig. 7(b)). The grid voltage drops by 20% from the nominal value at 625 ms (Fig. 7(c)).

Figs. 8, 9, and 10 show that the optimized controller in Ref. [23] fluctuates greatly in the control reactive power deviation. Fig. 10

shows that the optimized controller selected in Ref. [23] has worse control performance when the grid voltage drops than the controller based on other algorithms in this paper. Therefore, the PI parameters need to be adjusted and optimized. Therefore, the optimized controller parameters in Ref. [23] cannot achieve good reactive power control performance and stable control performance when the grid voltage drops.

4.2.3. Test of optimized proportional-integral parameters under step-change of wind speed

In the control process of the RSC optimized by the QPMMCOA, most of the w_{err} of the rotor recovers to zero at the fastest speed when the overshoot of the w_{err} of the rotor occurs (Fig. 11(a)); meanwhile, the overshoot of w_{err} in Table 8 indicates that the PI

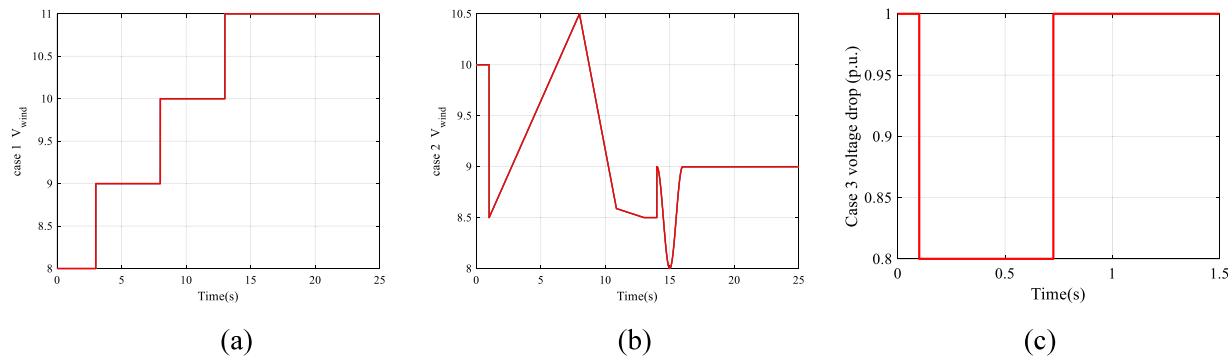


Fig. 7. Curves of wind speed or voltage drop: (a) step-change of wind speed; (b) randomly changing wind; (c) voltage drops.

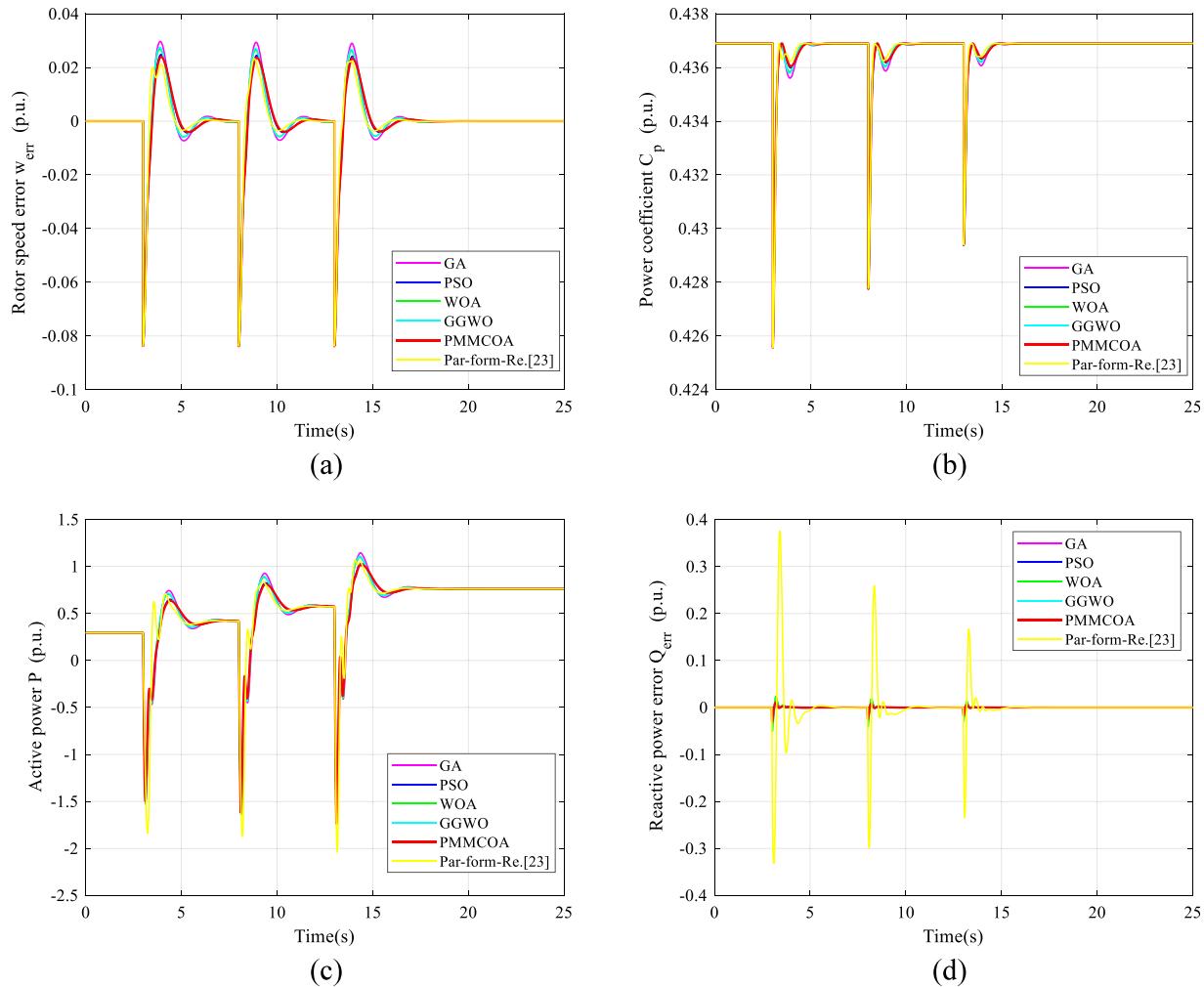


Fig. 8. Response under step-change of wind conditions: (a) rotor angular velocity error w_{err} ; (b) power coefficient C_p ; (c) active power P ; (d) deviation of reactive power Q_{err} .

parameters optimized by the QPMMCOA lead a smaller overshoot than other compared algorithms. In the whole control process of the RSC, the IAE of the w_{err} optimized by the QPMMCOA is the minimum (**Table 7**). Therefore, the RSC optimized by the QPMMCOA has satisfactory control performance in control the w_{err} .

Fig. 11(b) shows that when the PI parameters are found by the QPMMCOA, 75% of the C_p during the entire control process is higher than or equal to the C_p of the other four algorithms; the

controller optimized by the QPMMCOA causes the same undershoot of C_p as other compared algorithms (**Table 8**). In **Table 6**, the QPMMCOA-based RSC achieves the index of the C_p rank firsts. Therefore, the QPMMCOA enables the RSC to have the first-highest C_p under smooth and stable control conditions compared with other algorithms.

Fig. 11(c) and (d) show the control process of the P and the Q_{err} . The QPMMCOA-based RSC achieves the overshoots in the control of the Q_{err} with the third ranks (**Table 8**); similarly, the QPMMCOA appears that the dynamic process is smoother than

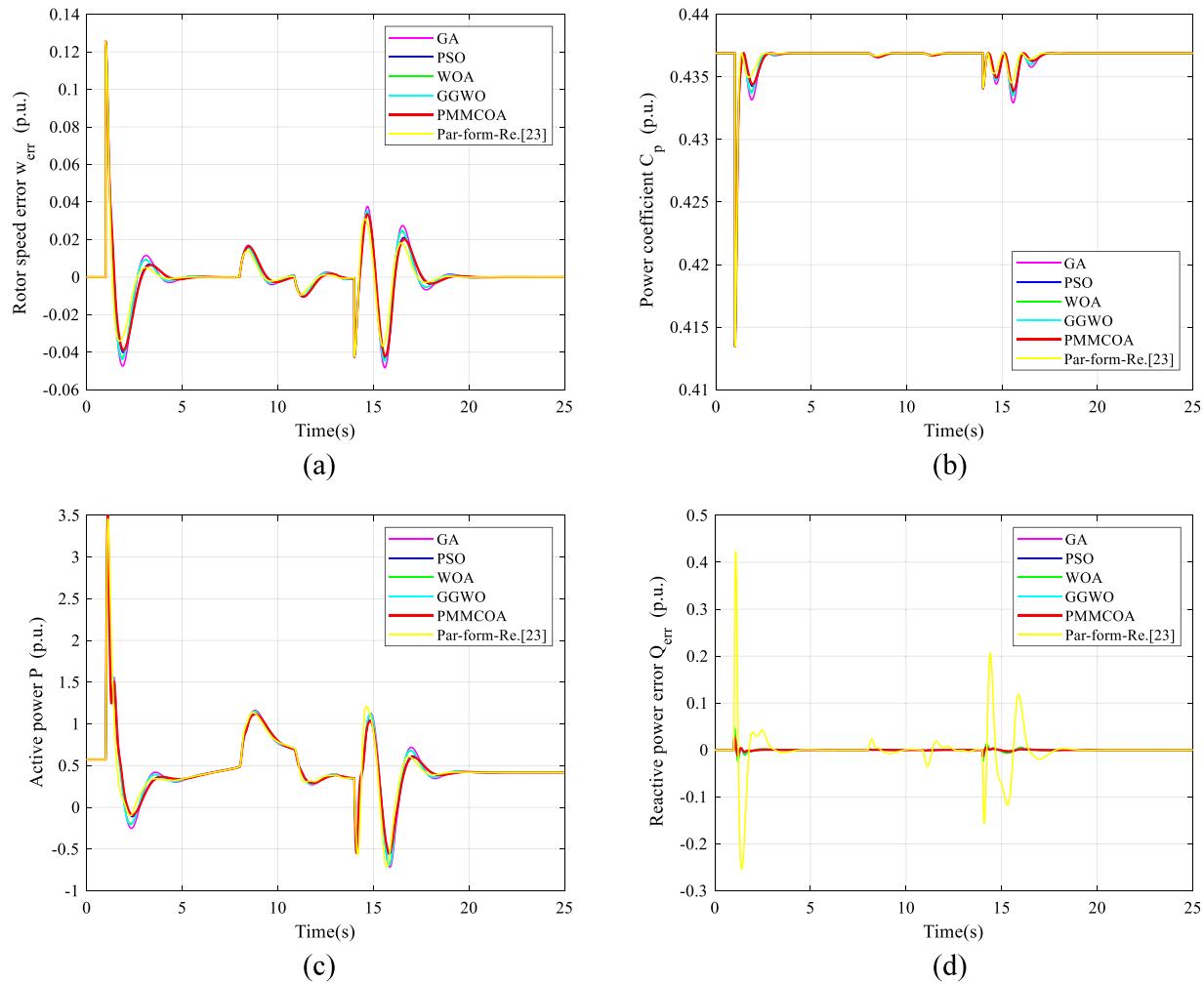


Fig. 9. Response under randomly changing wind conditions: (a) rotor angular velocity error w_{err} ; (b) power coefficient C_p ; (c) active power P ; (d) deviation of reactive power Q_{err} .

other algorithms in the adjustment of active power. In Table 7, the RSC optimized by the QPMMCOA has a smaller IAE of the Q_{err} than the other four algorithms. Therefore, the RSC optimized by the QPMMCOA can smoothly adjust active and reactive power under the step-change of wind speed conditions.

Table 7 shows the performance indices after the RSC parameters are optimized by various algorithms. The IAE, ISE, and ITAE indicators show that the controller parameters optimized by the QPMMCOA possess satisfactory transient response, small transient response oscillations, and fast response speed compared with other algorithms. The parameters optimized by the QPMMCOA offer the $ITSE_{w_{err}}$ index of the system minimum compared with other algorithms; therefore, the controller parameters optimized by the QPMMCOA can control large deviations and shorten adjustment time.

4.2.4. Test of optimized proportional-integral parameters under randomly changing wind

Fig. 12(a) shows that the w_{err} has minimum fluctuation in the entire control process when the QPMMCOA optimizes the parameters of the RSC (Table 8). In Table 7, the IAE of the w_{err} shows that the control effect of the RSC optimized by the QPMMCOA is significantly higher than the other four algorithms.

In Fig. 12(b), the C_p is higher than or equal to the other four algorithms at 90% when the controller parameters are optimized by the QPMMCOA; meanwhile, the QPMMCOA-based RSC causes the

same undershoot of C_p as other compared algorithms (Table 8). In the whole control process, the mean of the C_p shows that the C_p of the DFIG-WT system control by the QPMMCOA-based RSC is higher than that of the other four algorithms (Table 6). Therefore, the QPMMCOA can extract more wind energy in randomly changing wind conditions than other algorithms.

Fig. 12(c) and (d) show the control process of the P and the Q_{err} ; in the control of the P , the QPMMCOA has a slightly smaller fluctuation than the other four algorithms. In the control of the Q_{err} , the QPMMCOA is more stable than the other four algorithms; meanwhile, the RSC optimized by the QPMMCOA leads the overshoot of the Q_{err} with the second rank (Table 8). The RSC optimized by the QPMMCOA has a smaller IAE of the Q_{err} than the other four algorithms (Table 7). From the perspective of the P and the Q_{err} , the QPMMCOA provides the RSC with smoother control performance than other algorithms.

The IAE, ISE, ITAE, and ITSE indicators show that the PI parameters optimized by the QPMMCOA allow the control performance of the RSC stronger than other algorithms.

4.2.5. Test of optimized proportional-integral parameters under grid voltage drop

The controller parameters optimized by the QPMMCOA possess the smallest overshoot of the w_{err} and restore the w_{err} to zero at the fastest speed (Fig. 13(a) and Table 8). In Fig. 13(b), the C_p is significantly higher than the other four algorithms in the entire

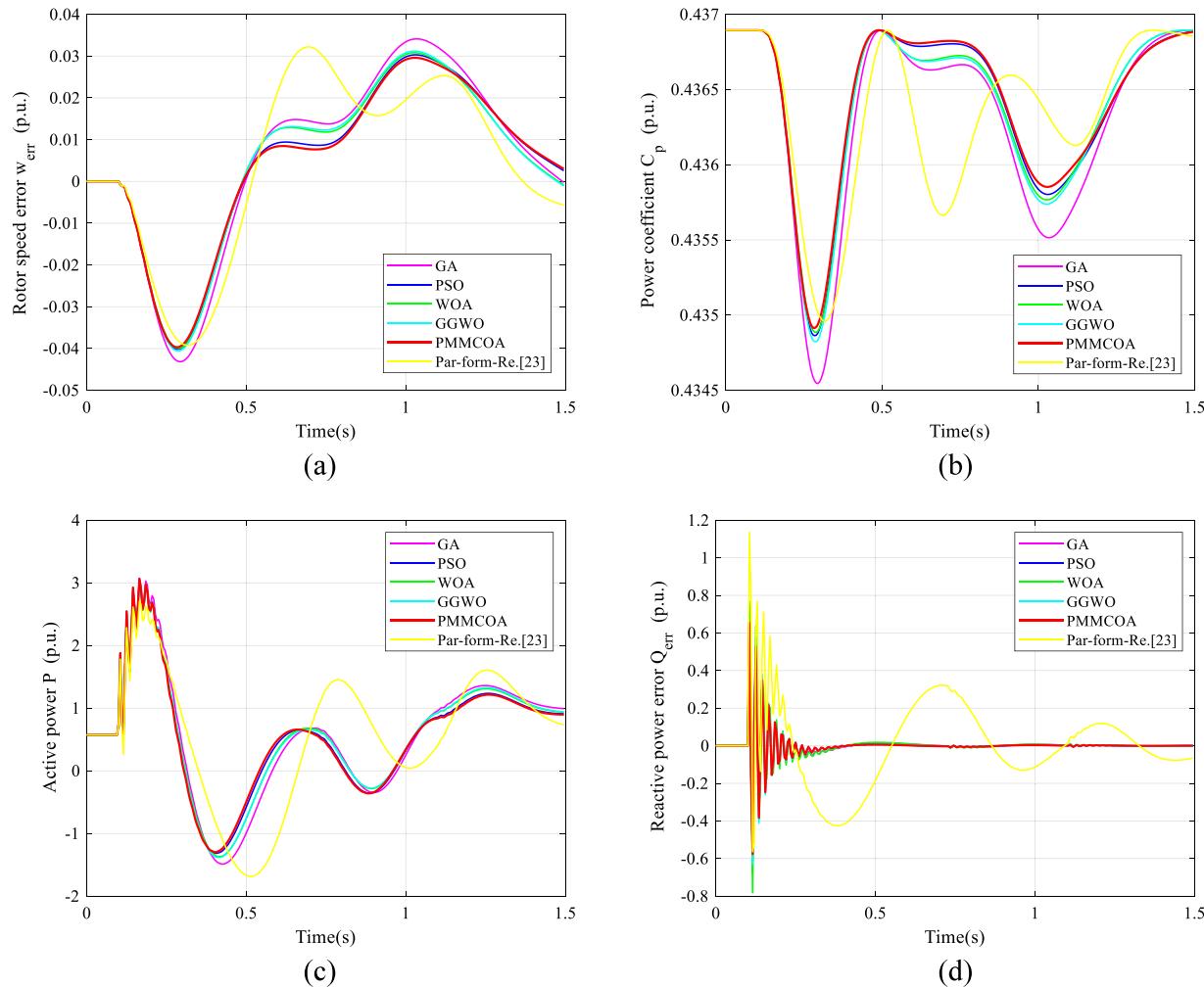


Fig. 10. Response under voltage drops conditions: (a) rotor angular velocity error w_{err} ; (b) power coefficient C_p ; (c) active power P ; (d) deviation of reactive power Q_{err} .

control process when utilizing the controller parameters found by the QPMMCOA. In Fig. 13(c), (d), and Table 8, the QPMMCOA based RSC can maintain the P with relatively minimal fluctuations and overshoot of the P ; the QPMMCOA maintains high-quality control ability and small overshoot in the control of the Q_{err} . The indicators in Tables 6 and 7 show that the RSC optimized by the QPMMCOA can achieve satisfactory control performance after the grid voltage drops suddenly. The response shows that the QPMMCOA achieves the highest rate of restoring rotor speed and reactive power, the highest power coefficient, and the smoothest active power compared with other algorithms. Therefore, the QPMMCOA can quickly and effectively repair the system interference caused by the voltage drop and enhance practicality and reliability.

Case 3 shows that the PI parameters optimized by the QPMMCOA lead the $IAE_{Q_{err}}$, $IAE_{\omega_{err}}$, $ITAE_{Q_{err}}$, $ISE_{\omega_{err}}$, and $ITSE_{\omega_{err}}$ indicators to the lowest, compared to other algorithms (Table 7). Therefore, the PI parameters optimized by the QPMMCOA can achieve RSC a satisfactory control effect under the grid voltage drops.

4.2.6. Summary in results of proportional-integral parameters optimization in rotor-side controller

Compared with other algorithms, the QPMMCOA has broader exploration and deeper development capabilities for optimizing the fitness function of the RSC. The fitness function value of the controller optimized by the QPMMCOA is 0.3513, which is 11.02%

Table 6

Average absolute value indices obtained by five algorithms under three cases.

Cases	Indices	GA	PSO	WOA	GGWO	QPMMCOA
1	AAV_{C_p}	0.4366502	0.4366821	0.4366821	0.4366764	0.4366886
1	AAV_p	0.6198162	0.6108063	0.6188051	0.6196442	0.6093519
2	AAV_{C_p}	0.4364991	0.4365619	0.4365580	0.4365474	0.4365735
2	AAV_p	0.5083271	0.4953898	0.5042813	0.5054043	0.4934627
3	AAV_{C_p}	0.4362971	0.4364141	0.4364033	0.4363890	0.4364330
3	AAV_p	0.9250898	0.8531823	0.8797524	0.8842650	0.8425008

lower than the GA, 0.51% lower than the PSO, 6.32% lower than the WOA, and 2.59% lower than the GGWO. The QPMMCOA has the following features.

- (1) The diversity of the population of the QPMMCOA can improve through the population coding method combining the qubit probability amplitude and Monte Carlo random number. With the diversity of the population, the QPMMCOA can avoid prematurely falling into the local optimum.
- (2) The QPMMCOA adopts a staged optimization mechanism with the rough search, precise search, and re-precise search. The rough search can find a rough optimized solution in the feasible region of the problem. The precise search can perform a more precise local search based on the optimized solution of the rough search. The re-precise search can re-optimize the optimized solution obtained

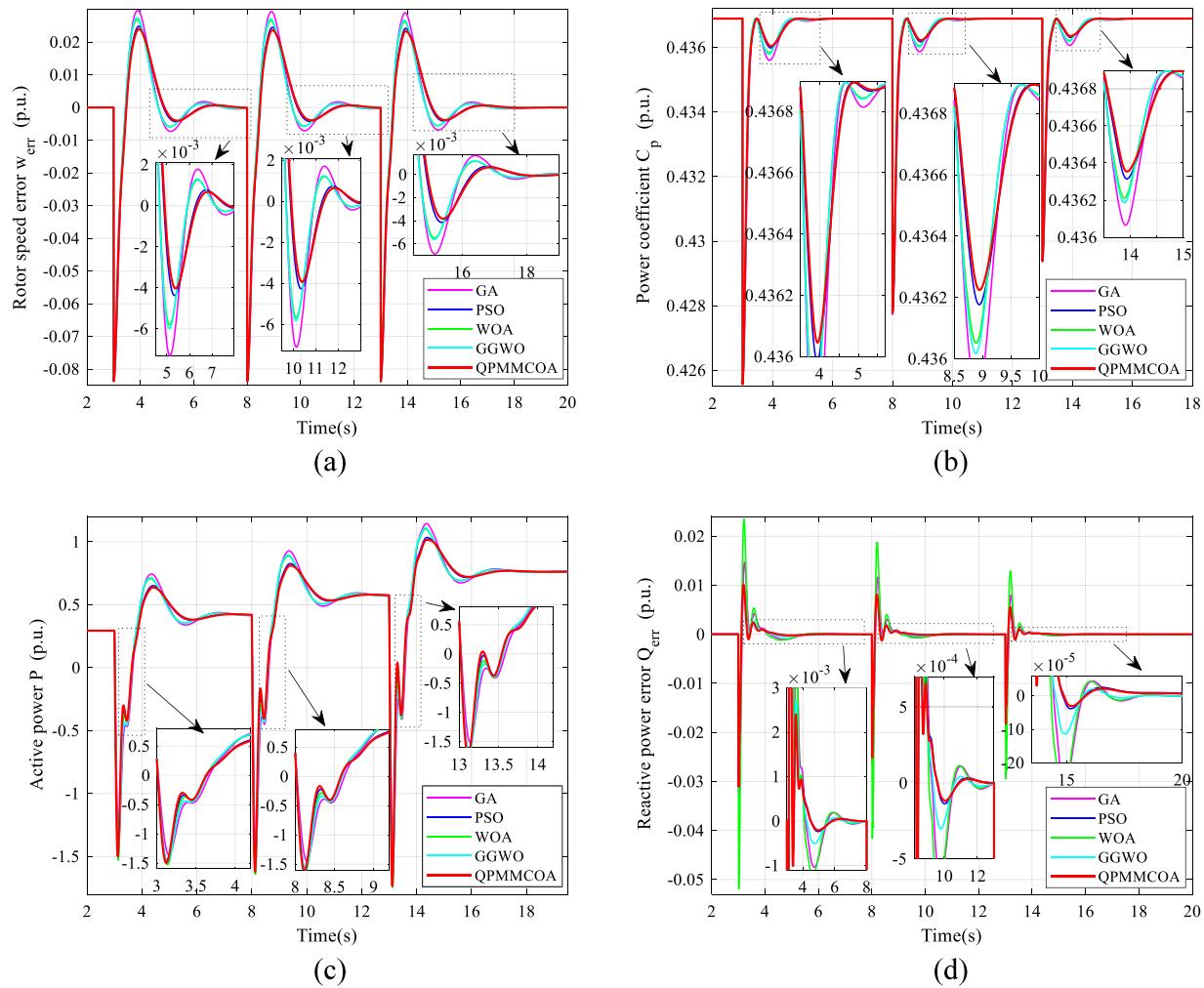


Fig. 11. Response under Case 1 and optimized controller parameters: (a) rotor angular velocity error w_{err} ; (b) power coefficient C_p ; (c) active power P ; (d) deviation of reactive power Q_{err} .

Table 7
Performance indices of optimized RSC by five algorithms under three cases.

Cases	Indices	GA	PSO	WOA	GGWO	QPMMCOA
1	$IAE_{Q_{err}}$	3.2766627	2.0946619	4.2763572	2.0760596	1.9955321
1	$IAE_{\omega_{err}}$	28.2469734	25.3217812	25.1929930	25.5834264	24.8701219
1	$ITAE_{Q_{err}}$	23.4292441	14.9675243	30.4967069	14.7850365	14.2696692
1	$ITAE_{\omega_{err}}$	250.4084326	223.2465061	221.9016026	225.1666235	219.2559081
1	$ISE_{Q_{err}}$	0.03987412	0.02201025	0.08138481	0.01921502	0.02100871
1	$ISE_{\omega_{err}}$	0.9506035	0.8237092	0.8247028	0.8469938	0.7981803
1	$ITSE_{Q_{err}}$	0.2464577	0.1376404	0.5010196	0.1174601	0.1303798
1	$ITSE_{\omega_{err}}$	7.9355323	6.8468934	6.8501191	7.0375037	6.6319270
2	$IAE_{Q_{err}}$	4.3412116	2.5634435	5.2829062	2.6682885	2.3738732
2	$IAE_{\omega_{err}}$	36.485433	32.859119	32.623765	33.157763	32.294474
2	$ITAE_{Q_{err}}$	38.7396729	21.2086755	45.4732412	23.3797612	19.3735319
2	$ITAE_{\omega_{err}}$	343.4782295	305.0085350	305.4994751	310.7003375	299.1311748
2	$ISE_{Q_{err}}$	0.03296422	0.01669132	0.06090712	0.01474572	0.01537835
2	$ISE_{\omega_{err}}$	1.2438577	1.0507357	1.0590407	1.0920680	1.0150227
2	$ITSE_{Q_{err}}$	0.1792101	0.07427466	0.3015433	0.07564139	0.06721576
2	$ITSE_{\omega_{err}}$	8.8699441	7.1783467	7.3582326	7.6151264	6.8915291
3	$IAE_{Q_{err}}$	6.9678101	5.6412573	7.9648438	5.9621641	5.5059274
3	$IAE_{\omega_{err}}$	5.5086367	4.8798568	4.9555279	5.0243814	4.7640095
3	$ITAE_{Q_{err}}$	1.7941463	1.2452089	2.0718883	1.3201959	1.1990620
3	$ITAE_{\omega_{err}}$	4.1353675	3.7263579	3.6869366	3.7341909	3.6503524
3	$ISE_{Q_{err}}$	1.9617930	1.6154048	2.4280749	1.7565396	1.5741473
3	$ISE_{\omega_{err}}$	0.1469427	0.1180267	0.1207491	0.1242591	0.1133530
3	$ITSE_{Q_{err}}$	0.2565251	0.2076877	0.3195502	0.2276714	0.2023794
3	$ITSE_{\omega_{err}}$	0.10371053	0.08329483	0.08402461	0.08640407	0.08000667

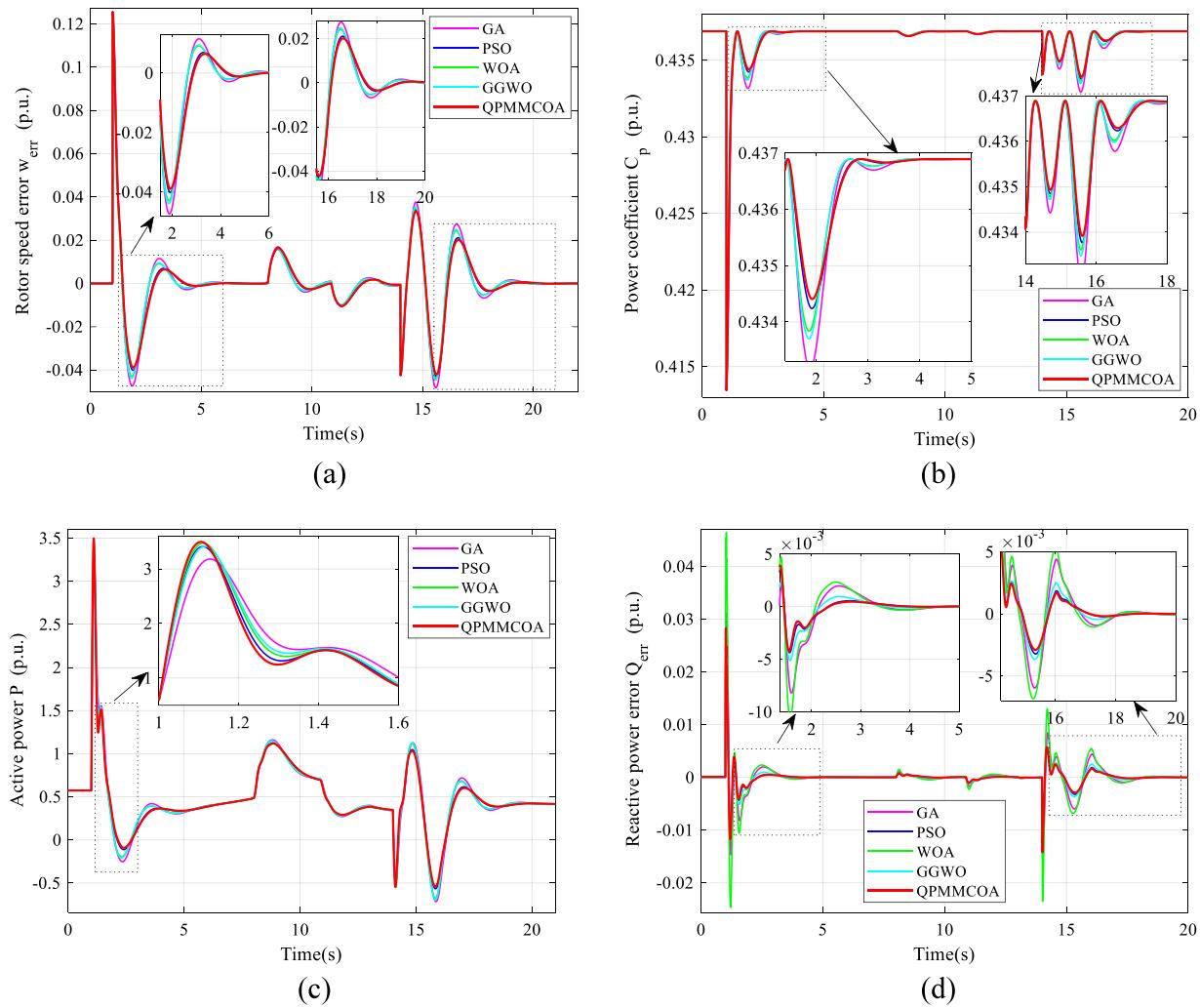


Fig. 12. Response under Case 2 and optimized controller parameters: (a) rotor angular velocity error w_{err} ; (b) power coefficient C_p ; (c) active power P ; (d) deviation of reactive power Q_{err} .

Table 8

Overshoot and undershoot indices of optimized RSC by five algorithms under three cases.

Cases	Indices	GA (%)	PSO (%)	WOA (%)	GGWO (%)	QPMMCOA (%)
1	Overshoot of ω_{err}	2.9705	2.4841	2.6849	2.7478	2.4004
1	Undershoot of C_p	2.5912	2.5912	2.5912	2.5912	2.5912
1	Overshoot of P	50.1497	35.5395	44.5667	45.6272	33.1819
1	Overshoot of Q_{err}	1.4683	0.9923	2.3467	0.9948	1.0149
2	Overshoot of ω_{err}	12.5478	12.5478	12.5478	12.5478	12.5478
2	Undershoot of C_p	5.3651	5.3651	5.3651	5.3651	5.3651
2	Overshoot of P	660.5338	715.6713	730.2398	716.2108	736.2468
2	Overshoot of Q_{err}	3.1143	2.8465	4.6393	2.3198	2.8283
3	Overshoot of ω_{err}	3.4122	3.0306	3.0805	3.1199	2.9595
3	Undershoot of C_p	0.5377	0.4649	0.4598	0.4734	0.4531
3	Overshoot of P	226.4635156	225.2926391	217.0386206	225.1083771	225.2771947
3	Overshoot of Q_{err}	74.6149	66.8821	76.9128	64.8638	65.4957

from the precise search. These three search processes constantly change and narrow the feasible region for the optimized process. The constantly changing feasible regions are closely related to the smallest optimization point and other sub-minimum optimization points in the search process. Under the optimization mechanism of the QPMMCOA, the probability of achieving a small fitness value is enhanced. Meanwhile, compared with other algorithms, the QPMMCOA has a stronger ability to avoid the local optimum.

Under three conditions, Table 6 shows that the average power coefficient of the DFIG-WT obtained by the QPMMCOA-based RSC is 0.4365650, which is 0.019% higher than the GA, 0.0028% higher than the PSO, 0.0039% higher than the WOA, and 0.0063% higher than the GGWO. Therefore, compared with other algorithms, the RSC optimized by the QPMMCOA can effectively improve the wind energy conversion efficiency of the DFIG-WT. Fig. 7 shows that the QPMMCOA possesses stronger global convergence stability and reliability than GA and PSO. Compared with other algorithms, the IAE, ISE, ITAE, and ITSE indicators in Table 7 show that the controller parameters optimized by the QPMMCOA

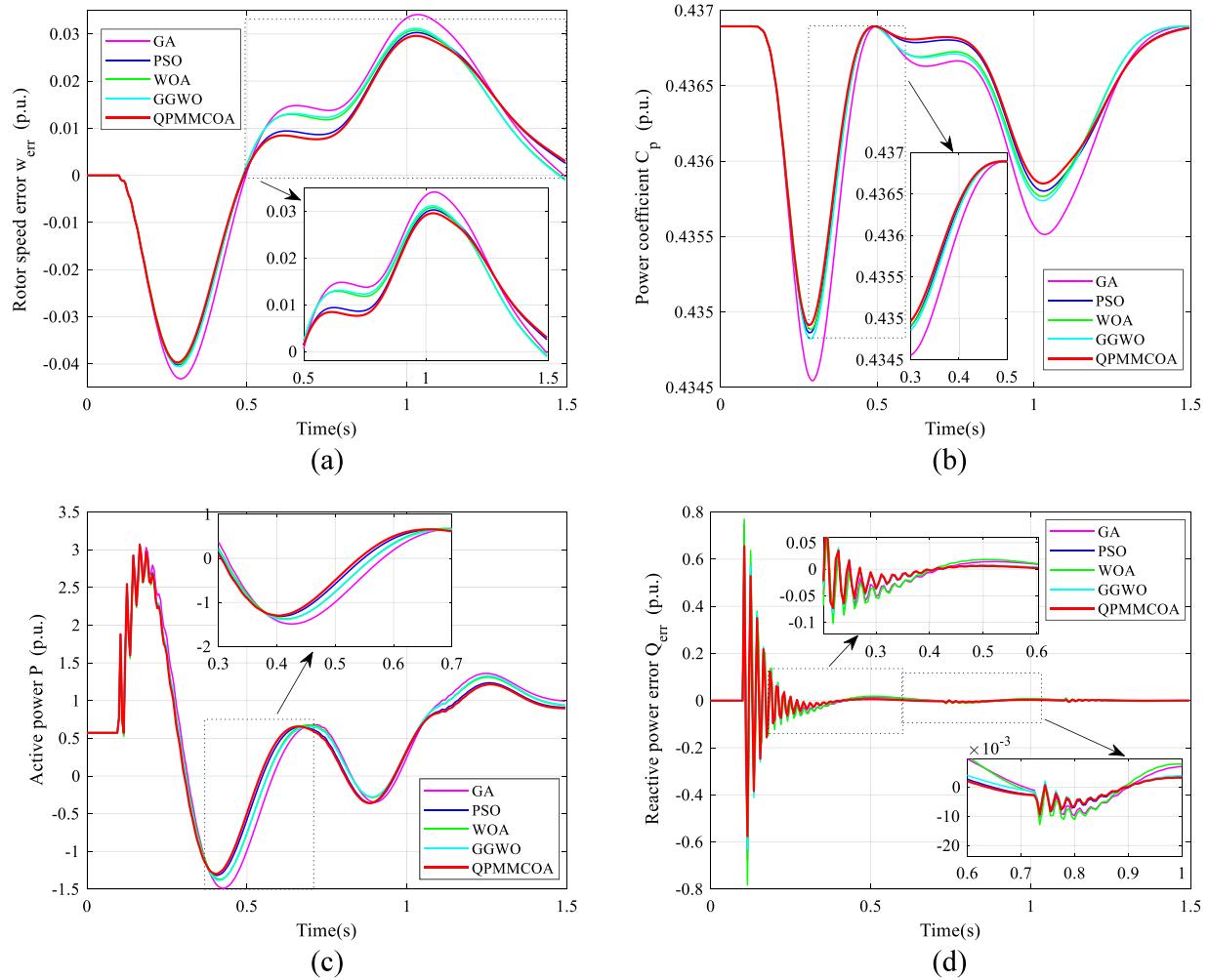


Fig. 13. Response under Case 3 optimized controller parameters: (a) rotor angular velocity error w_{err} ; (b) power coefficient C_p ; (c) active power P ; (d) deviation of reactive power Q_{err} .

possess more satisfactory transient response, smaller transient response oscillations, and faster response speed.

5. Conclusions

This article develops a QPMMCOA for optimizing PI parameters of the RSC of DFIG-WTs. Thus, wind energy is utilized in an optimal state as much as possible, and the MPPT is achieved. The QPMMCOA utilizes the concept of qubits in quantum mechanics to form the coding method of the population. Therefore, the individuals can occupy two positions in the space when the population size is constant. The population coding method in the algorithm can generate a population with diverse individuals in the optimization search process. A population with diverse individuals can reduce the possibility of falling into a local optimum as much as possible in the search process; therefore, the global search ability of the algorithm can be improved. In addition to the coding method, the search process of the QPMMCOA is divided into three steps: rough search, precise search, and re-precise search. These three search processes approach the optimal solution with multiple ever-shrinking feasible regions. Therefore, the QPMMCOA can improve global search capability and can achieve a deeper search capability. The applications of two different standard test functions show that the QPMMCOA is a credible and plausible way of disentangling optimization problems. Then, the QPMMCOA is applied to the PI parameters optimization process

of the RSC of DFIG-WTs. The optimization results show that the control performance of the RSC optimized by the QPMMCOA is higher than the other four algorithms (i.e., GA, PSO, WOA, and GGWO). The MPPT of a DFIG-WT is achieved under different wind types when the QPMMCOA optimizes the PI parameters of the RSC. The RSC optimized by the QPMMCOA ultimately improves the adoption of wind energy and the electricity supply efficiency of the DFIG-WT.

Through the design and analysis of this paper, the QPMMCOA is applied to the optimization of PI parameters; then, the control effect of the RSC is very satisfactory. However, the limitations of this paper are listed as follows.

- (1) The proposed algorithm is employed to realize the optimization of PI parameters of RSC of DFIG-WTs under the environment of MATLAB/Simulink. Although the new algorithm can achieve satisfactory control of the internal operating process of DFIG-WTs, the operating conditions of the DFIG-WT are still an ideal process. Therefore, the verification of the proposed algorithm in the actual measurement is necessary.
- (2) Other real-life conditions could be added to increase the diversity of wind samples. The optimized PI parameters of the controller under various real-life wind samples could increase the control performance of the RSC of DFIG-WTs.
- (3) The optimization time of the QPMMCOA can be shortened by changing the definition method of the feasible variable

region, changing the stopping conditions of the algorithm, and applying servers or other equipment to implement parallel computing. The parallel calculation could be employed rather than a single-step calculation after randomly selecting a population each time.

The QPMMCOA could be applied to more complex optimization problems, distributed optimization problems, and multi-objective optimization problems. The improved QPMMCOA could be utilized to solve distributed multi-objective optimization problems.

CRediT authorship contribution statement

Kunlun Han: Data curation, Software, Validation. **Tianwei Huang:** Data curation, Software, Validation, Writing - original draft. **Linfei Yin:** Conceptualization, Supervision, Methodology, Writing - review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the National Natural Science Foundation of Guangxi Province, China under Grant. AD19245001 and 2020GXNSFBA159025.

Appendix. Nomenclature

Variables

a_k :	Lower limits of variable x_k
a_1, a_w :	Iterative attenuation coefficients
α :	Probability amplitudes of qubits
aa :	Logic variable applied in the loop
b_w :	Logarithmic spiral shape constant
b_k :	Upper limits of variable x_k
β :	Pitch angle of wind turbine
$c_{\text{shrink},1}, c_{\text{shrink},2}, c_{\text{shrink},3}$:	Reduction factors
cc :	Logic variable applied in the loop
c_2, c_3 :	Acceleration factors
C_p :	Wind energy conversion factor
θ_{jk} :	Random phase from 0 to 2π and there are a total of j multiplied by k groups
D :	Damping of concentrated inertia system
H_g :	Concentrated inertia constant of generator
H_t :	Concentrated inertia constant of wind turbine
H_m :	Concentrated inertia constant of shaft system
i_{ds}, i_{dr} :	Direct-axis current components of stator and rotor
i_{qs}, i_{qr} :	Quadrature-axis current components of stator and rotor

ii :	Logic variable applied in loop
k_1, k_2, k_3 :	Rough search layer coefficient, precise search layer coefficient and re-precise search layer coefficient
$k_\alpha, k_\theta, k_\delta$:	Leading factors
kk :	Logic variables used in loop
K_{Pi}, K_{Ii} :	Controller parameters
l_w :	Random number between -1 and 1
L :	Represents current layer
L_{Bk} :	Lower limits of changeable feasible region in optimization process
L_s :	Self-induction of stator
L_r :	Self-induction of rotor
L_m :	Mutual inductance
λ :	Tip speed ratio
m :	Size of population
μ :	Probability amplitudes of qubits
M, M_1, M_2, M_3 :	Numbers of layers, rough search layers, precise search layers, and re-precise search layers
n :	Dimensionality of variable
n_p :	Number of pole pairs of motor
o :	Number of loops in optimization search process of each layer
ω :	Inertia weight
ω_m :	Mechanical angular speed
ω_{sy} :	Synchronous angular velocity of motor
ω_{sr} :	Slip angular velocity of motor
ω_r :	Electrical angular velocity of motor rotor
ω_r^* :	Reference value of rotor angular speed
p :	Differential operator
P :	Active power
P_{mu} :	Probability of mutation
P_{cro} :	Probability of cross probability
q_{j0}, q_{j1} :	Position of $ 0\rangle$ -state and position of $ 1\rangle$ -state
Q_s :	Reactive power
Q_s^* :	Reference value of reactive power
Q_{err} :	Deviation of reactive power
r :	Monte Carlo random number between 0 and 1
ρ :	Air density
R :	Radius of rotating blade of wind turbine
R_r :	Resistance of rotor
R_s :	Resistance of stator
σ :	Leakage coefficient
T :	Operating time
T_e :	Electromagnetic torque of generator
T_m :	Mechanical torque
u :	Number of optimization points saved in initial feasible region
u_{ds}, u_{dr} :	Direct-axis voltage components of stator and rotor
u_{qs}, u_{qr} :	Quadrature-axis voltage components of stator and rotor
U_{Bk} :	Upper limits of changeable feasible region in optimization process
v_{wind} :	Wind speed
v_{qr2}, v_{dr2} :	Compensation voltages
v_s :	Grid voltage
v_r :	Rotor-side voltage
v_{dr}, v_{qr} :	Control signals
w_1 :	Weight of the rotor angular velocity error in fitness function
w_{err} :	Rotor angular velocity error

$x:$	A point or a vector
$x_{j0}^k, x_{j1}^k:$	0)-state solutions and 1)-state solutions of problem
$x_1, x_2, \dots, x_n:$	An n -dimensional variable of a function
$y:$	Number of optimization points saved in optimization search process of each layer

Abbreviations

AAV:	Average absolute value
DFIG:	Doubly-fed induction generator
DFIG-WTs:	DFIG-based wind turbines
GA:	Genetic algorithm
GGWO:	Grouped gray wolf optimizer
IAE:	Integral of absolute error
ISE:	Integral of squared error
ITAE:	integral time-weighted absolute error
ITSE:	Integral of time multiplied by squared error
MPPT:	Maximum power point tracking
PI:	Proportional-integral
PID:	Proportional-integral-derivative
PSO:	Particle swarm algorithm
QPMMCOA:	Quantum parallel multi-layer Monte Carlo algorithm
RSC:	Rotor-side controller
WOA:	Whale optimization algorithm

References

- [1] A. Mohammed, Z. Li, A.O. Arowolo, H. Su, X. Deng, O. Najmuddin, Y. Zhang, Driving factors of CO₂ emissions and nexus with economic growth, development and human health in the Top Ten emitting countries, *Resour. Conserv. Recy.* 148 (2019) 157–169.
- [2] P.K. Ray, A. Mohanty, A robust firefly-swarm hybrid optimization for frequency control in wind/PV/FC based microgrid, *Appl. Soft Comput.* 85 (2019) 105823.
- [3] M.M. Rahman, E. Velayutham, Renewable and non-renewable energy consumption-economic growth nexus: new evidence from South Asia, *Renew. Energy* 147 (2020) 399–408.
- [4] W.-j. Niu, Z.-k. Feng, C.-t. Cheng, X.-y. Wu, A parallel multi-objective particle swarm optimization for cascade hydropower reservoir operation in southwest China, *Appl. Soft Comput.* 70 (2018) 562–575.
- [5] Z. Wang, W. Wang, C. Liu, Z. Wang, Y. Hou, Probabilistic forecast for multiple wind farms based on regular vine copulas, *IEEE Trans. Power Syst.* 33 (1) (2018) 578–589.
- [6] Z. Cheng, J. Wang, A new combined model based on multi-objective salp swarm optimization for wind speed forecasting, *Appl. Soft Comput.* (2020) 106294.
- [7] R. Khezri, A. Oshnoei, S. Oshnoei, H. Bevrani, S. Muyeen, An intelligent coordinator design for GCSC and AGC in a two-area hybrid power system, *Appl. Soft Comput.* 76 (2019) 491–504.
- [8] H. Wang, Y. Liu, B. Zhou, N. Voropai, G. Cao, Y. Jia, E. Barakhtenko, Advanced adaptive frequency support scheme for DFIG under cyber uncertainty, *Renew. Energy* 161 (2020) 98–109.
- [9] Z.Y. Xue, K.S. Xiahou, M.S. Li, T.Y. Ji, Q.H. Wu, Diagnosis of multiple open-circuit switch faults based on long short-term memory network for DFIG-based wind turbine systems, *IEEE J. Emerg. Sel. Top. Power Electron.* 8 (3) (2020) 2600–2610, <http://dx.doi.org/10.1109/JESTPE.2019.2908981>.
- [10] X. Peng, W. Yao, C. Yan, J. Wen, S. Cheng, Two-stage variable proportion coefficient based frequency support of grid-connected DFIG-WTs, *IEEE Trans. Power Syst.* 35 (2) (2020) 962–974.
- [11] D.E.R. Conde, A. Lunardi, A.J. Sguarezi Filho, Current control for DFIG systems under distorted voltage using predictive repetitive control, *IEEE J. Emerg. Sel. Top. Power Electron.* 14 (8) (2020) 1–10, <http://dx.doi.org/10.1109/JESTPE.2020.3025503>.
- [12] A. Giannakis, A. Karlis, Y.L. Karnavas, A combined control strategy of a DFIG based on a sensorless power control through modified phase-locked loop and fuzzy logic controllers, *Renew. Energy* 121 (2018) 489–501.
- [13] M. Rostami, S.M. Madani, S. Ademi, Sensorless closed-loop voltage and frequency control of stand-alone DFIGs introducing direct flux-vector control, *IEEE Trans. Ind. Electron.* 67 (7) (2020) 6078–6088.
- [14] H. Mahvash, S.A. Taher, M. Rahimi, M. Shahidehpour, DFIG performance improvement in grid connected mode by using fractional order PI controller, *Int. J. Electr. Power Syst.* 96 (2018) 398–411.
- [15] A. Zakaria, F.B. Ismail, M.H. Lipu, M.A. Hannan, Uncertainty models for stochastic optimization in renewable energy applications, *Renew. Energy* 145 (2020) 1543–1571.
- [16] I. Ahmadianfar, Z. Khajeh, S.-A. Asghari-Pari, X. Chu, Developing optimal policies for reservoir systems using a multi-strategy optimization algorithm, *Appl. Soft Comput.* 80 (2019) 888–903.
- [17] Hale Bakir, Adel Merabet, Rupak Kanti Dhar, Ahmet Afsin Kulaksiz, Experimental evaluation of water cycle technique for control parameters optimization of double-fed induction generator-based wind turbine, *Eng. Sci. Technol.* 24 (4) (2021) 890–898.
- [18] A. Chen, D. Xie, D. Zhang, C. Gu, K. Wang, PI parameter tuning of converters for sub-synchronous interactions existing in grid-connected DFIG wind turbines, *IEEE Trans. Power Electron.* 34 (7) (2019) 6345–6355.
- [19] E.D.P. Puchta, H.V. Siqueira, M.d.S. Kaster, Optimization tools based on metaheuristics for performance enhancement in a gaussian adaptive PID controller, *IEEE Trans. Cybern.* 50 (3) (2020) 1185–1194.
- [20] Ravi Patel, Faizal Hafiz, Akshya Swain, Abhishek Ukil, Nonlinear rotor side converter control of DFIG based wind energy system, *Electr. Power Syst. Res.* 198 (2021) 107358, <http://dx.doi.org/10.1016/j.epsr.2021.107358>.
- [21] Q. Pham, S. Mirjalili, N. Kumar, M. Alazab, W. Hwang, Whale optimization algorithm with applications to resource allocation in wireless networks, *IEEE Trans. Veh. Technol.* 69 (4) (2020) 4285–4297.
- [22] Rajeev Kumar, Rajveer Singh, Haroon Ashfaq, Sudhir Kumar Singh, Manoj Badoni, Power system stability enhancement by damping and control of sub-synchronous torsional oscillations using whale optimization algorithm based Type-2 wind turbines, *ISA Trans.* 108 (2021) 240–256.
- [23] B. Yang, X. Zhang, T. Yu, H. Shu, Z. Fang, Grouped grey wolf optimizer for maximum power point tracking of doubly-fed induction generator based wind turbine, *Energy Convers. Manage.* 133 (2017) 427–443.
- [24] C.N. Cong, R. Rodriguez-Jorge, N.N. Ba, C.T. Trong, N.N. Anh, Design of optimal PI controllers using the chemical reaction optimization algorithm for indirect power control of a DFIG model with MPPT, in: Workshops of the International Conference on Advanced Information Networking and Applications, Springer, 2020, pp. 1250–1260.
- [25] W. Zheng, W. Liu, W. Dong, Z. Deng, Y. Yang, Monte-Carlo tuning of PMSM servo system based on active-disturbance rejection controller, in: 2017 36th Chinese Control Conference (CCC), 2017, pp. 6090–6093, <http://dx.doi.org/10.23919/ChiCC.2017.8028326>.
- [26] L. Liang, P. Zhao, S. Zhang, Roll reduction control during ship turns using fin stabilizers with PID controller based on Monte Carlo optimization, in: 2018 IEEE International Conference on Mechatronics and Automation (ICMA), 2018, pp. 749–754, <http://dx.doi.org/10.1109/ICMA.2018.8484680>.
- [27] Y. Xia, Z.-k. Feng, W.-j. Niu, H. Qin, Z.-q. Jiang, J.-z. Zhou, Simplex quantum-behaved particle swarm optimization algorithm with application to ecological operation of cascade hydropower reservoirs, *Appl. Soft Comput.* 84 (2019) 105715.
- [28] L. Shi, S. Ólafsson, Nested partitions method for stochastic optimization, *Methodol. Comput. Appl. Probab.* 2 (3) (2000) 271–291.
- [29] A.H. Aguirre, S.B. Rionda, C.A. Coello, G.L. Lizárraga, E.M. Montes, Handling constraints using multiobjective optimization concepts, *Internat. J. Numer. Methods Engrg.* 59 (15) (2004) 1989–2017.
- [30] K. Yamamura, K. Suda, N. Tamura, LP narrowing: A new strategy for finding all solutions of nonlinear equations, *Appl. Math. Comput.* 215 (1) (2009) 405–413.
- [31] A. Anupam, R. Gupta, A. Naeemi, N. JafariNaimi, Particle in a box: An experiential environment for learning introductory quantum mechanics, *IEEE Trans. Educ.* 61 (1) (2018) 29–37.
- [32] Y. Yang, J. Liu, S. Tan, A multi-objective evolutionary algorithm for steady-state constrained multi-objective optimization problems, *Appl. Soft Comput.* (2020) 107042.
- [33] V. Akshay, H. Philathong, M. Morales, J. Biamonte, Reachability deficits in quantum approximate optimization, *Phys. Rev. Lett.* 124 (9) (2020).
- [34] J. Hussain, M.K. Mishra, An efficient wind speed computation method using sliding mode observers in wind energy conversion system control applications, *IEEE Trans. Ind. Appl.* 56 (1) (2020) 730–739.
- [35] A.J. Sguarezi Filho, A.L. de Oliveira, L.L. Rodrigues, E.C.M. Costa, R.V. Jacomini, A robust finite control set applied to the DFIG power control, *IEEE J. Emerg. Sel. Top. Power Electron.* 6 (4) (2018) 1692–1698, <http://dx.doi.org/10.1109/JESTPE.2018.2833474>.
- [36] C. Zhang, D. Ke, Y. Sun, C.Y. Chung, J. Xu, F. Shen, Coordinated supplementary damping control of DFIG and PSS to suppress inter-area oscillations with optimally controlled plant dynamics, *IEEE Trans. Sustain. Energy* 9 (2) (2018) 780–791.
- [37] P.P. Biswas, P.N. Suganthan, R. Mallipeddi, G.A. Amaralunga, Optimal reactive power dispatch with uncertainties in load demand and renewable energy sources adopting scenario-based approach, *Appl. Soft Comput.* 75 (2019) 616–632.
- [38] C.-F. Wang, W.-X. Song, A novel firefly algorithm based on gender difference and its convergence, *Appl. Soft Comput.* 80 (2019) 107–124.
- [39] K. Chen, F. Zhou, L. Yin, S. Wang, Y. Wang, F. Wan, A hybrid particle swarm optimizer with sine cosine acceleration coefficients, *Inform. Sci.* 422 (2018) 218–241.