

Energy capture efficiency enhancement of wind turbines via stochastic model predictive yaw control based on intelligent scenarios generation

Dongran Song^a, Ziqun Li^a, Lei Wang^{b,*}, Fangjun Jin^b, Chaoneng Huang^a, E. Xia^a, Rizk M. Rizk-Allah^c, Jian Yang^a, Mei Su^a, Young Hoon Joo^d

^a School of Automation, Central South University, Changsha, Hunan, China

^b School of Automation, Chongqing University, Chongqing, China

^c Department of Basic Engineering Science, Faculty of Engineering, Menoufia University, Shebin El-Kom, Egypt

^d School of IT Information and Control Engineering, Kunsan National University, Kunsan, Republic of Korea

HIGHLIGHTS

- Stochastic model predictive yaw control strategy based on intelligent scenarios generation.
- New quasi-PID optimization problem for generating optimal scenarios.
- Novel co-evolution bonobo optimizer to solve the quasi-PID problem.
- Improving the energy capture efficiency by 0.26%–0.43% while reducing the yaw time ratio by 0.35%–1.58%.

ARTICLE INFO

Keywords:

Wind turbine
Yaw
Energy capture efficiency
Stochastic model predictive control
Intelligent scenarios generation
Co-evolutionary bonobo optimizer

ABSTRACT

Wind direction is random and time-varying, which is arduous to be accurately predicted. The yaw control based on the predicted wind direction is limited by the accuracy of the wind direction prediction, which leads to narrow improvement in the energy capture efficiency of the wind turbine (WT). For this issue, a Stochastic Model Predictive Yaw Control (SMPYC) strategy based on Intelligent Scenarios Generation (ISG) is proposed. Herein, in view of the uncertainty of wind direction prediction, the ISG method is proposed to generate scenarios that characterize it, then the yaw action optimized through the proposed scenario-based SMPYC is performed to improve the energy capture efficiency of WTs. Specifically, ISG creates an optimization problem from scenarios generation in each control period, and the co-evolution bonobo optimizer is improved to solve the optimal scenarios in real time for this high-dimensional multimodal problem. The proposed SMPYC based on ISG is tested using historical wind direction data, and its effectiveness and advantages under different accuracy of wind direction prediction are validated by the test results. The proposed SMPYC reduces the yaw time ratio by 0.35%–1.58% and improves the energy capture efficiency by 0.26%–0.43% in comparison with the baseline MPYC. For a 5 MW WT, the gained energy production could reach $1.14\text{--}1.88 \times 10^5$ kWh in a year, which corresponds to an additional annual profit of 68,000–110,000 yuan. Consequently, the proposed method is promising to enhance the energy capture efficiency and has important application value for reducing the cost of wind power.

1. Introduction

Carbon emissions have continually been a hot issue of global concern. Powered by the rapid progression of wind power, this problem is expected to be alleviated, and wind power, accordingly, has become a clean energy favored by many countries [1]. In order to expand the economic benefits of wind power generation, wind turbine (WT) is

constantly developing towards high-power and large-scale [2]. For large WTs, yaw refers to adjusting the angle between the rotor surface and the direction of the incoming wind by turning the nacelle position [3]. The misalignment of yaw will lead to greater energy loss and higher fatigue load, which will not only reduce the power generation efficiency, but also short the service life of WT, and thus the influence of yaw cannot be neglected [4]. Therefore, in order to improve the economic benefits of wind power generation through active yaw, more advanced control

* Corresponding author.

E-mail address: leiwang08@cqu.edu.cn (L. Wang).

<https://doi.org/10.1016/j.apenergy.2022.118773>

Received 28 December 2021; Received in revised form 29 January 2022; Accepted 14 February 2022

Available online 28 February 2022

0306-2619/© 2022 Elsevier Ltd. All rights reserved.

Nomenclature			
Acronyms			
WT	wind turbine	$rcpp$	parameter of BO
MPYC	model predictive yaw control	N_h	data set length
SMPYC	stochastic model predictive yaw control	MPC	model predictive control
BO	bonobo optimizer	SMPC	stochastic model predictive control
ISG-SMPYC	intelligent scenarios generation based stochastic model predictive yaw control	ISG	intelligent scenarios generation
PDF	probability density function	IC-BO	improved co-evolution bonobo optimizer
KDE	Kernel Density Estimation	LSTM-WDP	long and short-term memory neural network wind direction predictor
PP	positive phase	OCP	optimal control problem
STD	standard deviation	PSO	particle swarm optimization
Symbols		NP	negative phase
$\dot{\theta}_{np}$	yaw rate	BMPYC	baseline model predictive yaw control
θ_{wd}	wind direction	θ_{ye}	yaw error
T_s	control period	θ_{np}	nacelle position
QF	objective function	y_i	i -th optimal scenario
ζ	the proportion of yaw time to total time	ξ	reduction of energy capture
ω	weight factor of QF	m	prediction horizon
x	wind direction prediction error	p_i	probability corresponding to the i -th scenario
N_{kde}	length of the prediction error sequence	h	bandwidth of KDE
$\omega_1, \omega_2, \omega_3$	weight coefficients of ISG	N_{pdf}	probability density intervals
N_{pe}	window length of KDE	γ	joint distribution
N_{st}	length of prediction step	N_{sc}	length of scenarios sequence
L, Z, Q	subobjective function of ISG	W_{yw}	objective function of ISG
U_{up}	upper bounds of y	U_{low}	lower bounds of y
p_{bo}	p -th individual in BO	i_{bo}	i -th individual in BO
$tsgs_{factor}$	parameter of BO	$r_1 - r_6$	random number
		s_{cab}, s_{csb}	parameters of BO
		$p_{xgm_initial}$	parameter of BO
		N_p	prediction set length

strategies and methods are required.

The development of prediction methods [5,6] and technical methods (such as LiDAR) [7,8] makes it possible to achieve advanced control strategies. [8] applied LiDAR technology to assist the modeling of dynamic yaw misalignment, which confirms the superiority of utilizing LiDAR. In terms of prediction issues, numerous prediction methods have emerged with the assistance of machine learning [9] and artificial intelligence [10], including wind direction prediction [11–13], wind speed prediction [14–18] and wind power prediction [19,20]. Some of these methods used deep learning to establish an ultra-short-term prediction model of wind direction [11]; some predicted equivalent wind speed by extreme learning machines [14]; some constructed a GRU prediction model of wind speed [18]. Summarily, the current methods and techniques have been capable to obtain moderately accurate wind information.

On this basis, several predictive control methods have been gradually applied to WT yaw system. As a substantial method of predictive control, model predictive control (MPC) owns many applications in energy conservation, emission reduction and clean energy [21–23], which also shows excellent potential in WT yaw control [24–27]. [24] designed a finite set model predictive yaw control (MPYC) strategy based on the predicted wind direction, which degrades the yaw error, and [25] systematically compared the disparity between traditional logic control and MPYC, showed that MPYC performed better than logic control. On this basis, [26] established a multi-step MPYC, which made full use of the predicted wind direction information, while [27] further considered the influence of control period and proposed an adaptive MPYC to meliorate yaw performance. In general, these methods assume that the predicted wind direction is accurate. Unfortunately, with the current technical methodology, the prediction of wind direction cannot fully get rid of errors [28], thus the performance of these advanced control strategies might not be guaranteed.

Aimed at the possible random interference and uncertainty, a variety of coping methods have been proposed in other fields [29,30]. Stochastic model predictive control (SMPC) is capable of applying the random characteristics of noise to deal with model or external uncertainty, showing satisfactory control effects [31,32]. In [33], a scenario-based SMPC was established for energy storage scheduling optimization, and the solution speed was accelerated by optimality condition decomposition. [34] proposed a distributed SMPC for photovoltaic system to ensure efficient energy management. In terms of wind power generation, [35,36] established SMPC based on scenario tree on the basis of Markov chain prediction to be applied to wind energy conversion system; [37] modeled the predicted wind speed as Gaussian disturbance, and solved SMPC by quadratic programming method to increase power generation. Unfortunately, these proposed SMPCs almost all focus on the torque control of the WT, and little attention is paid to the yaw system. In addition, the key of scenario based SMPC is whether the generated scenarios can accurately describe the uncertainty of the system, and the high degree of randomness and uncertainty of wind prediction makes scenarios generation more strenuous.

It is gratifying that the advancement of artificial intelligence provides some promising novel thoughts for scenario generation. [38] directly generated renewable energy scenarios by GAN to avoid any prior knowledge; on this basis, [39] realized scenarios prediction by solving optimization problems, and provided reliable prediction scenarios for power system dispatching. Coincidentally, [40] proposed a conditional automatic variational encoder to generate scenarios for wind power photovoltaic output, which is convenient for power system scheduling. However, these methods are based on a large amount of data for offline training, which is arduous to directly apply to the generation of wind direction prediction scenarios: On the one hand, the wind direction prediction is applied to MPYC, requiring real-time online training; on the other hand, the yaw control time scale (in second level)

of WT is far less than the scheduling problem (in hour level), which requires more accurate scenarios. In summary, constructing SMPC for WT's yaw system and generating appropriate scenarios is an urgent problem to be solved, but related research is relatively lacking.

Gratifiedly, as intelligent algorithms grow by leaps and bounds, stochastic optimization [41], which can solves approximate solutions of complex optimization problems in a short time [42], is expected to solve these problems. Once the optimization problem of scenarios generation is constructed, it will be possible to generate optimal scenarios of SMPC for WTs. Based on this, this study proposes a Stochastic Model Predictive Yaw Control (SMPYC) method based on Intelligent Scenarios Generation (ISG) for the WT yaw system. The SMPYC could generate the optimal scenarios in real time in each control period, and implement the control decision based on these scenarios, thus improving energy capture and reducing yaw time. The main innovative contributions of the study are as follows:

- 1) The proposed method carefully constructs corresponding optimization problems for scenarios generation. This optimization problem is inspired by PID: full consideration is given to the learning of historical prediction error experience (I), the evaluation of current prediction error state (P), and the tolerance of future prediction error fluctuations (D), which is called quasi-PID optimization problem. Solving this problem can quickly and efficiently generate optimal scenarios, which has universal applicability and pertinence;
- 2) An improved Bonobo Optimizer (BO), called Improved Co-evolution BO (IC-BO), is proposed to solve the quasi-PID optimization problem. Based on the baseline BO [43], co-evolution strategy and metropolis criteria are introduced to effectively improve IC-BO's ability to solve high-dimensional multimodal optimization problems;
- 3) Based on the optimal scenarios, a finite set SMPYC is designed to achieve the deep enhancement of yaw performance. The controller includes a yaw error prediction model, an objective function, and a solver, which all are specially constructed for multiple scenarios.

The subdivision of this study is as follows: Section 2 details the proposed intelligent scenarios generation stochastic model predictive yaw control method (ISG-SMPYC), including SMPYC framework, the construction of scenarios generation optimization problem and the introduction of IC-BO; Section 3 verifies the effectiveness of the proposed method by simulation experiments. Section 4 summaries the conclusion.

2. Framework of ISG-SMPYC

Due to the prediction error of wind direction, the yaw time of the WT is increased and the energy capture efficiency is reduced under the baseline MPYC. The proposed ISG-SMPYC establishes a scenario-based SMPYC based on generating the optimal scenarios that characterize

the uncertainty of wind direction prediction through ISG, thereby improving the yaw performance of the WT when the predicted average wind direction (to make the expression concise, it is defined as the predicted wind direction in the whole paper) is inaccurate. The overall structure of ISG-SMPYC is shown in Fig. 1. It generates the optimal scenarios in real time in each control period, and then calculates the control action by optimal scenarios-based SMPYC to realize yaw control. In this section, the framework of SMPYC is first introduced, then the concept of intelligent scenarios generation, and finally the philosophy of IC-BO.

2.1. Construction of SMPYC

SMPYC is the intermediate link of ISG-SMPYC. The predicted wind direction and the optimal scenarios generated by ISG are used as the inputs of SMPYC, combined with the WT's nacelle position and other signals to determine the best yaw action. Like the basic MPC, the proposed SMPYC structure is mainly composed of three parts: prediction model, objective function and solver.

2.1.1. Yaw error prediction model of SMPYC

To construct the SMPYC yaw error prediction model, it is necessary to model the yaw motion and average wind direction prediction first. The object of this study is a large horizontal axis WT with large yaw torque and slow yaw action. Therefore, the yaw model adopts a rigid model with a fixed yaw rate of 0.5 deg/s [26]. The yaw actions include clockwise rotation (0.5 deg/s), standstill (0), and counterclockwise rotation (−0.5 deg/s), and there is action constraint between them:

$$\dot{\theta}_{np}(k+1) = \begin{cases} 0.5 \exists \dot{\theta}_{np}(k) = \{0, 0.5\} \\ 0 \exists \dot{\theta}_{np}(k) = \{-0.5, 0, 0.5\} \\ -0.5 \exists \dot{\theta}_{np}(k) = \{-0.5, 0\} \end{cases} \quad (1)$$

where $\dot{\theta}_{np}(k)$ is the yaw rate at k -th control period, which remains unchanged in each control period.

The average wind direction is predicted by a Long and short-Term Memory neural network Wind Direction Predictor (LSTM-WDP). LSTM-WDP first obtains the historical average wind direction sequence with N_h as the window length, and normalizes the sequence to establish a data set; then trains the LSTM and uses the trained model to estimate the future N_p data to establish a prediction set; finally, the prediction set is denormalized to obtain the predicted wind direction sequence, and the process is repeated in the next period. On this basis, the yaw error is calculated by the predicted wind direction and the yaw model:

$$\theta_{ye}(k+m|k) = \theta_{wd}(k+m|k) - \theta_{np}(k|k) - \sum_{i=1}^m \dot{\theta}_{np}(k+i) \cdot T_s \quad (2)$$

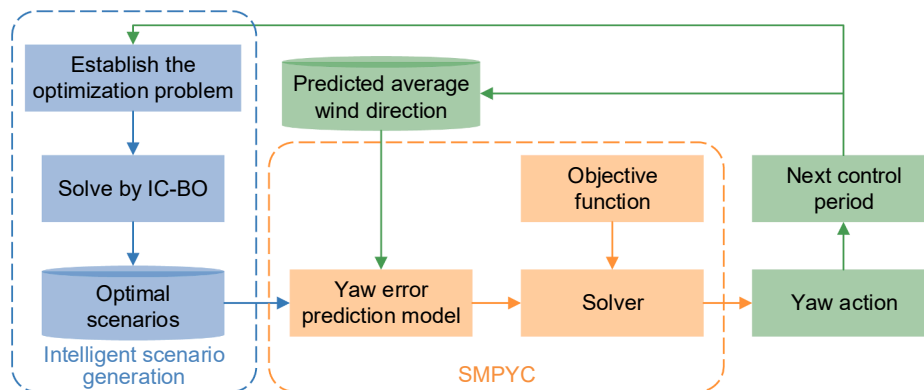


Fig. 1. The overall framework of ISG-SMPYC.

where m is the prediction step, $\theta_{ye}(k+m|k)$ and $\theta_{wd}(k+m|k)$ are the predicted yaw error and the predicted wind direction of $(k+m)$ -th period at k -th period, respectively, $\theta_{np}(k|k)$ is nacelle position at k -th period, T_s is control period.

It can be seen from Eq. (2) that the uncertainty of the average wind direction prediction directly affects the predicted yaw error, thereby influencing the control performance of MPYC. For SMPYC, the yaw error is calculated comprehensively based on the predicted wind direction under different scenarios. The yaw error prediction model under the i -th scenario is:

$$\theta_{ye,i} = \theta_{ye}(k+m|k) - y_i(k+m|k) \quad i \in [1, N_{sc}] \quad (3)$$

where $y_i(k+m|k)$ is the i -th optimal scenario of $(k+m)$ -th period at k -th period, N_{sc} is the total number of scenarios.

2.1.2. Objective function of SMPYC

The targets of SMPYC are reducing the yaw time ratio and improving the energy capture efficiency of WTs, which are contradictory. By adding a weight coefficient between the two goals, the objective function of SMPYC is expressed as:

$$QF = (1 - \omega) \cdot \xi + \omega \cdot \zeta$$

$$= (1 - \omega) \cdot \left(1 - \left(\sum_{n=k+1}^{k+m} \cos^2(\theta_{ye}(n)) \right) / m \right) + \omega \cdot \left(\left(\sum_{n=k+1}^{k+m} (|\dot{\theta}_{np}(n)| > 0) \right) / m \right) \quad (4)$$

where ξ is the reduction of energy capture caused by yaw error, ζ is the proportion of yaw time to total time. The QF calculated with different scenario is different. Assuming that the objective function calculated with $\theta_{ye,i}$ is denoted as $QF(i)$, the total objective function of SMPYC is the probability sum of $QF(i)$:

$$QF_{SMPYC} = \sum_{i=1}^{N_{sc}} p_i \cdot QF(i) \quad (5)$$

where p_i is the probability corresponding to the i -th scenario.

2.1.3. Solver of SMPYC

It can be seen from Eq. (1) that, when m is finite, Eq. (2) and Eq. (4) describe a finite set MPYC. Moreover, there are only three possible yaw actions, so enumeration is a fast and effective method [44]. Similarly,

SMPYC also solves the problem by enumeration method. Because the algorithm complexity increases in linear order, the multi-scenarios calculation will not significantly increase the time cost. SMPYC solver mainly consists of three steps:

- Step 1: Enumerate all possible yaw action combinations according to the current yaw action and m by Eq. (1);
- Step 2: Calculate the objective function QF of each yaw action combination under each scenario by Eqs. (2)–(4);
- Step 3: Calculate the SMPYC objective function QF_{SMPYC} by Eq. (5), and the first action in the yaw action combination corresponding to the minimum value is the yaw action of the next control period.

Taking $m = 2$, the current $\dot{\theta}_{np} = 0$ as an example, Fig. 2 shows the solution process of SMPYC.

2.2. Statement of scenario generation problem

Due to the randomness of wind direction and prediction errors, there are external uncertainties in the yaw system. By constructing a mathematical description model of uncertainty, it can be compensated in the

control to improve the yaw performance. Scenario generation is a method of describing external uncertainty, and its essence is a stochastic simulation technique [45]. However, due to the large randomness of short-term wind direction prediction and the obvious statistical difference of prediction errors at different times, it is hard to apply traditional scenario generation method. Aiming at this uncertainty problem with high randomness and large fluctuations in statistical characteristics, an ISG method is proposed to generate corresponding scenarios through intelligent optimization to compensate for the shortcomings of traditional methods.

2.2.1. Random scenario generation

The scenario generation method uses the Probability Density Function (PDF) of the error distribution to sample the independent and identically distributed prediction errors to generate a large number of scenarios, thereby constructing a scenario-based Optimal Control

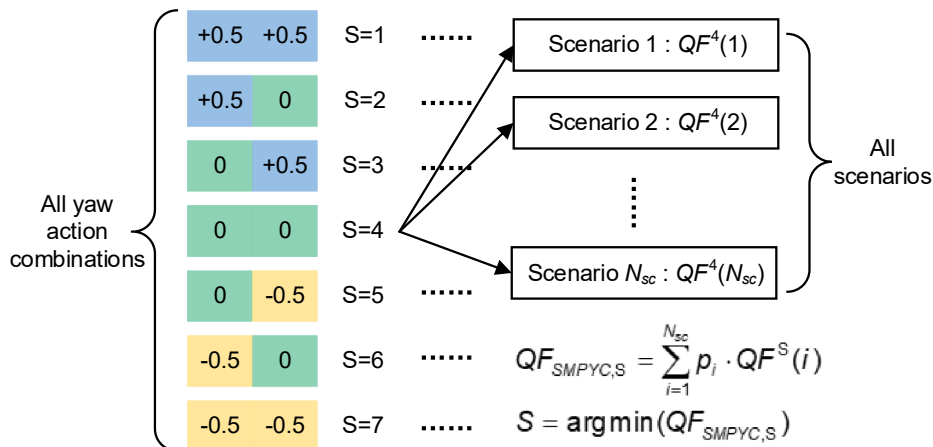


Fig. 2. The solution process of SMPYC.

Problem (OCP). When the number of scenarios is large enough, the solution of the scenario-based OCP is similar to the solution of the original OCP [45]. Therefore, the premise of scenario generation is to obtain the PDF of the average wind direction prediction error. Due to the high randomness of the wind direction, the PDFs at different times are different, and the type of PDF is also unknown. Kernel Density Estimation (KDE), as a non-parametric test method, does not use prior knowledge about the data distribution, so the statistical characteristics can be well estimated by the KDE method. For the historical average wind direction prediction error sequence $x_1, x_2, \dots, x_{N_{kde}}$, assuming that its sample points are independent and identically distributed, the PDF calculated by KDE is:

$$\hat{f}_h(x) = \frac{1}{N_{kde}} \sum_{i=1}^{N_{kde}} K_h(x - x_i) = \frac{1}{N_{kde}h} \sum_{i=1}^{N_{kde}} K\left(\frac{x - x_i}{h}\right) \quad (6)$$

where h is the bandwidth, N_{kde} is the length of the prediction error sequence, $K(\cdot)$ is the kernel function.

Considering that the prediction error of wind direction presents an approximate Gaussian distribution under a large sample, this study uses the Gaussian kernel function to establish the KDE of the prediction error:

$$K(x) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}x^2\right) \quad (7)$$

Through Eq. (6) and Eq. (7), the PDF of any average wind direction prediction error sequence can be obtained. Random scenario generation is realized on this basis. Fig. 3 shows the specific idea of random scenario generation.

Assuming that the statistical characteristics of wind direction prediction errors are constant under a large amount of data, this assumption is reasonable according to the law of large numbers. First obtain the average wind direction prediction error data, and calculate its PDF through KDE. Then discretize the obtained PDF into N_{pdf} probability density intervals with the same width, and calculate the probability value corresponding to each interval; finally, a certain number of sample points are generated by random sampling. The value pair $[y_i, p_i]$ formed by each sample point and the probability of corresponding interval is a randomly generated scenario.

2.2.2. The quasi-PID optimization problem

The above-mentioned random scenario generation method has a good effect when applied to power system dispatching or wind power-photovoltaic output problems [46]. However, there are two critical problems with traditional methods in yaw control:

- The time scale of the scheduling problem is large, and the fluctuation of random interference is small, so the generated scenario can well represent the uncertainty of the problem. The difference is that the wind direction changes drastically due to the influence of turbulence, and the period of yaw control is relatively small; therefore, the scenarios generated by traditional methods are difficult to accurately characterize the prediction errors.
- On the other hand, the yaw control system requires to generate scenarios online within a limited time. The time cost of generating a

large number of scenarios and reducing it is very high, so it can only be done offline. However, the offline generated scenarios will have the defects mentioned in the first point, because they cannot dynamically track the statistical characteristics of the prediction errors in real time.

The proposed ISG constructs the optimization problem of scenarios generation, and improves the BO to solve the optimization problem to realize the dynamic and rapid generation of scenarios. It updates the statistical characteristics of prediction errors in real time in each control period, improves the dynamic tracking ability of scenarios, and effectively avoids the first problem. At the same time, two major improvements to BO enhance the speed and accuracy of the solution, thereby realizing online scenarios generation.

The goal of scenarios generation is to approximate the external uncertainty with a greater probability and reduce the impact of prediction errors on the yaw control system. Inspired by the classical control method PID, ISG divides the goal of scenario generation into three parts similar to PID: proportional, integral, and derivative, so as to achieve the conversion of scenario generation problem to optimization problem. Specifically, the three parts can be summarized as follows: 1) The generated scenarios need to accurately reflect the statistical characteristics of historical prediction errors over a period of time; 2) The ability to accurately fit short-term prediction errors; 3) It has good wrapping properties for future prediction errors. The constructed scenarios generation optimization problem can be expressed as:

$$W_{yw} = \omega_1 L + \omega_2 Z + \omega_3 Q \quad (8)$$

$s.t. U_{low} < y < U_{up}$

where L , Z , and Q are the three sub-objectives mentioned above, respectively; ω_1 , ω_2 , and ω_3 are the weight coefficients corresponding to the three sub-objectives, respectively; y is the scenario value, that is, the optimization variable; U_{low} and U_{up} are the upper and lower bound constraints of the scenario value, respectively. Eq. (8) is the quasi-PID optimization problem, and its structure will be described in detail below:

Reflection for the statistical characteristics of historical prediction errors

One of the intelligent goals of scenarios generation is to have a good similarity to the statistical characteristics of wind direction prediction errors over a period. The statistical characteristics of historical prediction errors are a summary of historical experience, including the overall change characteristics of prediction errors. To summarize, this goal is to learn from historical experience and grasp the overall changes.

By calculating the distance between the two probability distributions, the degree of similarity between the generated scenarios and the historical prediction errors can be measured. Intelligent optimization usually covers the entire search space as much as possible for optimization. Therefore, it is very likely that there is no overlap between the distribution of the generated scenarios and the prediction errors. At this time, the KL divergence will lose its meaning and the JS divergence will degenerate into constant [47], and Wasserstein distance can still characterize the distance between the two distributions when there is no overlap. For the prediction errors probability distribution $\mathbb{P}_{pe}(x)$ and the scenarios generation probability distribution $\mathbb{P}_{sc}(y)$, the Wasserstein distance is:

$$W(\mathbb{P}_{pe}, \mathbb{P}_{sc}) = \inf_{\gamma \in \Pi(\mathbb{P}_{pe}, \mathbb{P}_{sc})} \mathbb{E}_{(x,y) \sim \gamma} [\|x - y\|] \quad (9)$$

where $\Pi(\mathbb{P}_{pe}, \mathbb{P}_{sc})$ represents the set of all possible joint distributions of $\mathbb{P}_{pe}(x)$ and $\mathbb{P}_{sc}(y)$; $\mathbb{E}_{(x,y) \sim \gamma} [\|x - y\|]$ represents the expectation of the distance between samples x and y under the joint distribution γ ; $\mathbb{P}_{pe}(x)$ is the PDF of the prediction errors sequence $[x_{t-N_{pe}}, x_{t-N_{pe}+1}, \dots, x_{t-1}]$ with a window length of N_{pe} at t -th control period that is calculated by KDE. The value range of N_{pe} is generally [100, 200] to ensure that sufficient historical wind direction features are extracted; $\mathbb{P}_{sc}(y)$ is the PDF of the

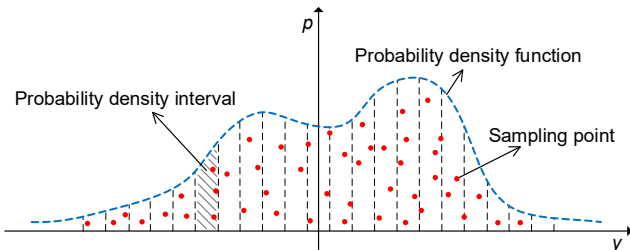


Fig. 3. The process of random scenario generation.

generated N_{sc} scenarios sequence $[y_1, y_2, \dots, y_{N_{sc}}]$ that calculated by KDE. The value of N_{sc} generally needs to be greater than 30 to ensure that all possible scenarios are covered. The goal of intelligent scenario generation is to minimize the Wasserstein distance, so the first goal can be expressed as:

$$L = \min \left(\sum_{i=1}^{N_{st}} W_i(\mathbb{P}_{pe}, \mathbb{P}_{sc}) \right) \quad (10)$$

where N_{st} is the length of prediction step, in this study its value is consistent with m .

Fitting short-term prediction errors

Another goal of intelligent scenario generation is to have a better ability to fit short-term wind direction prediction errors. The short-term wind direction prediction errors refer to the average wind direction prediction errors sequence of the previous N_{st} control periods at t -th control period $[x_{t-N_{st}}, x_{t-N_{st}+1}, \dots, x_{t-1}]$. This target reflects the situation of the prediction errors at the current moment, and it is an evaluation of the current state, making the scenarios generation quickly locate the area where the current prediction errors are located.

Quantify the degree of fit between the generated scenarios and the short-term prediction errors by calculating the absolute difference between the probability sum of the scenario and the short-term prediction errors sequence. The probability sum of scenarios is calculated as:

$$Z_{sc} = \sum_{i=1}^{N_{sc}} p_i y_i \quad (11)$$

where y_i and p_i are the value and the corresponding probability of the i -th scenario, respectively. On this basis, the goal of intelligent scenarios generation is to minimize the weighted sum of the absolute difference with the short-term prediction errors sequence, so the second goal can be expressed as:

$$Z = \min \left(\sum_{i=1}^{N_{st}} \left((N_{st} - i + 1) |Z_{sc}^i - x_{t-i}| \right) / \left(\sum_{j=1}^{N_{st}} j \right) \right) \quad (12)$$

where Z_{sc}^i is the corresponding scenarios probability sum at $(t-i)$ -th control period, x_{t-i} is the average wind direction prediction error at $(t-i)$ -th control period. The closer the absolute difference is to the current control period, the larger the corresponding weight.

Encapsulation of future prediction errors

At the same time, the scenarios generation needs to fully consider the uncertainty of future prediction errors, ensuring that the scenarios have a good encapsulation for future prediction errors. Therefore, this goal guarantees the robustness of scenarios generation.

For a certain scenario cluster, the encapsulation of scenarios generation is measured by calculating the difference between its overall upper and lower bounds. Its expression is:

$$Q_{sc} = \sup_{i \in N_{sc}} \sum_{n=1}^{N_{st}} y_i^n - \inf_{j \in N_{sc}} \sum_{m=1}^{N_{st}} y_j^m \quad (13)$$

where N_{st} is the number of scenario clusters, that is, starting from t -th control period, N_{st} scenario clusters are generated backwards. This value is consistent with the length of prediction step, so it is uniformly denoted by N_{st} . The goal of intelligent scenarios generation is to make its wrapping and robustness as good as possible, that is, Q_{sc} should be maximized. Considering that the first two goals are all seeking the minimum, the third goal can be expressed as:

$$Q = \min(1/Q_{sc}) \quad (14)$$

In summary, Eq. (8) converts the scenarios generation problem into an optimization problem. Among the three sub-goals, L is the learning of historical experience and the grasp of overall change, which makes the generated scenarios more in line with the statistical characteristics of

historical wind direction prediction errors. Z is the evaluation of the current wind direction prediction errors, which is convenient for scenarios generation to quickly locate the region where the current prediction errors is located; Q is the consideration of extreme cases and robustness of scenarios generation, which makes the generated scenarios have better encapsulation. By solving the optimization problem in each yaw control period, the corresponding optimal scenarios can be dynamically generated, and then ISG-SMPYC can be realized.

2.3. ISG solution method

For the real-time global optimization problem described in Eq. (8), the optimization variable $\mathbf{y}$ has $N_{st} \cdot N_{sc}$ dimensions and the objective function is also enormously complex. It needs to be solved quickly in each control period, so an efficient heuristic algorithm is required. BO is a new heuristic algorithm for real parameters proposed in recent years, which has been verified in multiple test functions. It has better performance than Particle Swarm Optimization (PSO), Differential Evolution and other algorithms, and has been applied to the field of renewable energy [48]. Due to the high dimensions and multimodality of this problem, co-evolution is introduced to improve BO to reduce dimensions and accelerate the convergence speed, moreover, the ability of BO is improved to jump out of local optimum by applying the metropolis criterion.

2.3.1. Philosophy of BO

The inspiration of BO comes from the social behavior and reproductive strategy of bonobos. The algorithm draws on the social strategy of bonobo's fission and fusion, which forms several subgroups of different sizes in the population to update, and reunites into a whole after each iteration. The population update strategy is similar to the four breeding behaviors of bonobos: promiscuous, restrictive mating, consort-ship, and extra-group. Different update strategies correspond to whether the population should strengthen exploration or exploitation. After each iteration, BO automatically updates the algorithm parameters according to the state of the current population to balance in exploration and exploitation.

Similar to other heuristic algorithms, the best individual in the population is called α_{bo} , which is also dominant. When the α_{bo} gets improved in iteration, the population is said to be in a Positive Phase (PP), and vice versa, it is called a Negative Phase (NP). As shown in Fig. 4, when the population is located in PP, other bonobos tend to approach α_{bo} , and the probability of promiscuous and restrictive mating is greater. At this time, the algorithm focuses on the exploitation of search space. When the population is located in NP, other bonobos will be far away from α_{bo} , and the probability of consort-ship and extra-group mating is greater. At this time, the algorithm focuses on the exploration of search space.

BO first determines the temporary subgroup corresponding to individual i_{bo} according to the temporary subgroup size factor $tsgs_{factor}$, and then selects individual p_{bo} in the temporary subgroup to mate with it.

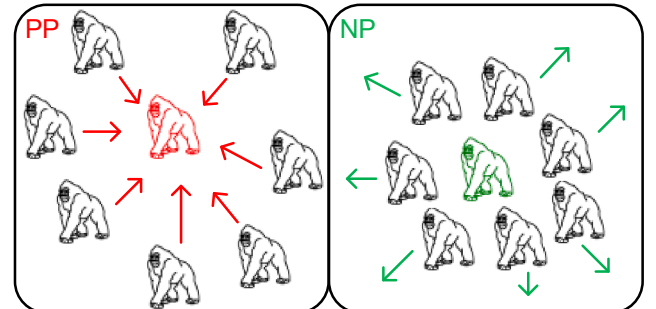


Fig. 4. The search tendency of bonobos in PP and NP phases.

The selection of p_{bo} follows two rules: 1) If the individuals in the subgroup are worse than i_{bo} , it is selected randomly; 2) if there are individuals better than i_{bo} in the subgroup, it selected randomly from these individuals. The formula for promiscuous and restrictive mating to update individuals is:

$$i_{bo}^j = i_{bo}^j + r_1 \cdot s_{cab} \cdot (\alpha_{bo}^j - i_{bo}^j) + (1 - r_1) \cdot s_{csb} \cdot f_{lag} \cdot (i_{bo}^j - p_{bo}^j) \quad (15)$$

where j is the dimension of the variable, r_1 is a random number between $[0,1]$, s_{cab} and s_{csb} are two corresponding inertia weights, respectively; f_{lag} is a flag bit, which is $+1$ for promiscuous, and p_{bo} is selected by rule 1; -1 for restrictive mating, and p_{bo} is selected by rule 2.

The formula for extra-group mating to update individuals is:

$$\begin{aligned} \beta_1 &= e^{(r_4^2 + r_4 - 2/r_4)} & \beta_2 &= e^{(-r_4^2 + 2r_4 - 2/r_4)} \\ i_{bo}^j &= i_{bo}^j + \beta_1 (V_{\max}^j - i_{bo}^j) & i_{bo}^j &= i_{bo}^j - \beta_2 (i_{bo}^j - V_{\min}^j) \\ i_{bo}^j &= i_{bo}^j - \beta_1 (i_{bo}^j - V_{\min}^j) & i_{bo}^j &= i_{bo}^j + \beta_2 (V_{\max}^j - i_{bo}^j) \end{aligned} \quad (16)$$

where r_4 is a random number between $(0,1]$, and $V_{\max}^j$ and $V_{\min}^j$ are the upper and lower limits of the j -th dimension of the variable, respectively. The formula for the consort-ship mating is:

$$i_{bo}^j = \begin{cases} i_{bo}^j + f_{lag} \cdot e^{-r_5} \cdot (i_{bo}^j - p_{bo}^j) & f_{lag} = 1 || r_6 < p_d \\ p_{bo}^j & otherwise \end{cases} \quad (17)$$

where r_5 and r_6 are random numbers between $[0,1]$.

2.3.2. Construction of IC-BO

The pseudo code for Improved Co-evolutionary BO algorithm is given in Algorithm 1. The two features of IC-BO can be summarized as follows:

- Co-evolutionary is used to reduce the dimensionality of this high-dimensional problem. On the one hand, the dimensionality is reduced and then the sub-populations are created to optimize the efficiency of internal competition, thereby accelerating the convergence speed of BO, and providing convenience for real-time optimization. On the other hand, the information exchange after each iteration enhances the cooperation of BO sub-groups, improves the local search ability of BO, and facilitates finding the optimal solution.
- In view of the multimodal of this problem, new individuals are screened by the metropolis criterion, so that BO has the ability to escape from the local optimum. Combined with the advantages of BO itself and co-evolution, IC-BO can quickly solve the high-dimensional complex optimization problem described in Eq. (8) in each yaw control period, and realize intelligent scenario generation.

Algorithm 1 IC-BO

```

1:  // Initialization #/
2:  Create and initialize  $N_{st}$  populations, each with  $N_{sc}$  dimensions. The  $j$ -th population is denoted as  $P_j$ ,  $j \in [1, N_{st}]$ . The individual best bonobo of  $i$ -th bonobo in  $j$ -th populations is denoted as  $P_{j,y_i}$ . The best bonobo of  $j$ -th populations is denoted as  $P_{j,y^*}$ .
3:  // Co-evolution and bonobos update #/
4:  repeat
5:    for each population  $j \in [1, N_{st}]$  do
6:      for each bonobo  $i \in j$  do
7:        if  $w_{yw}([P_{1,y^*}, P_{2,y^*}, \dots, j, \dots]) \leq w_{yw}([P_{1,y^*}, P_{2,y^*}, \dots, P_{j,y_i}, \dots])$ 
8:           $P_{j,y_i} = j$ 
9:        if  $w_{yw}([P_{1,y^*}, P_{2,y^*}, \dots, P_{j,y_i}, \dots]) \leq w_{yw}([P_{1,y^*}, P_{2,y^*}, \dots, P_{j,y^*}, \dots])$ 
10:          $P_{j,y^*} = P_{j,y_i}$ 
11:      end for
12:    Update bonobos in each population according to Eq. (15)–Eq. (17).
13:    Accept new bonobos by .
14:  end for
15:  Information exchange between populations.
16: until termination criterion is met

```

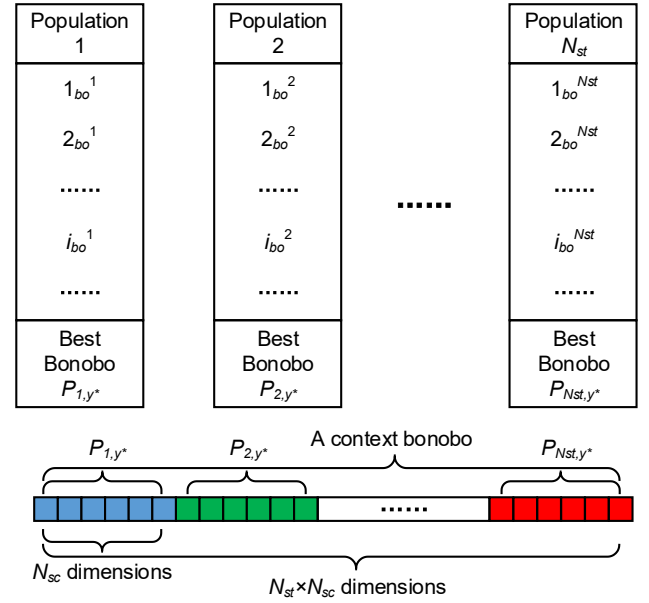


Fig. 5. Schematic diagram of co-evolution.

Co-evolution

Co-evolution employs cooperative behavior to improve the performance of heuristic algorithm, and uses multi-population technology to collaboratively optimize different parts of the optimization vector variables [49]. Although individual competition can improve its performance, appropriate cooperation is of positive significance for improving the search efficiency of the entire population. As shown in Fig. 5, co-evolution divides optimization variables into several parts according to dimensions, and each part corresponds to a subpopulation for evolution. For the optimization problem described in Eq. (8), the optimization vector y is highly dimensional and separable. Divide y into N_{st} parts according to N_{sc} , and the optimization of each part is carried out in their respective groups and exchanges information after each iteration.

Metropolis criterion

Due to the high dimensionality of the variables, the local optimum of the problem increases. Although the convergence speed and the optimization ability are accelerated by co-evaluation, the complexity of the optimization problem causes the population to fall into the local optimum and premature convergence. The metropolis criterion receives degraded solutions with a certain probability, thereby improving the algorithm's ability to escape the local optimum and avoid premature convergence [50]. For BO, the metropolis criterion can be expressed as:

$$p_{ac} = \begin{cases} 1 & w_{yw}(bo^k) - w_{yw}(bo^{k-1}) < 0 \\ e^{-k(w_{yw}(bo^k) - w_{yw}(bo^{k-1})) / T} & w_{yw}(bo^k) - w_{yw}(bo^{k-1}) \geq 0 \end{cases} \quad (18)$$

where k is the number of iterations, $w_{yw}(bo^k)$ and $w_{yw}(bo^{k-1})$ is the fitness value of the individual at the k and $k-1$ iterations, respectively, and T is a hyperparameter.

In summary, the proposed IC-BO algorithm is utilized to solve problem Eq. (8), and the optimal scenarios in each yaw control period can be obtained. Combining with the predicted wind direction as the input of SMPYC to control the WT yaw action, is the overall process of ISG-SMPYC. In the next section, the proposed ISG-SMPYC is validated.

3. Method validation and result discussion

In order to evaluate the proposed intelligent scenarios generation method and the performance of SMPYC, simulation experiments were carried out. Section 3.1 describes the experimental preparation and

Table 1
Parameter settings of the algorithm and controller.

Algorithm/ Controller	IC-BO	MPYC	SMPYC
Parameters	$rcpp = 0.0035$ $scab = 1.25$ $scsb = 1.3$ $tsgs_{factor_max} = 0.05$ $P_{xgm_initial} = 1/N_{sc}$	Sample period = 1 s Control period = 20 s Predict horizon(m) = 3	$N_{sc} = 30$ $N_{st} = m$ $\omega = 0.03$

discusses the uncertainty of wind direction prediction. Then the optimal results are discussed and analyzed in Section 3.2. Section 3.3 compares the control performance of MPYC and SMPYC. Finally, the SMPYC is further analyzed and discussed in Section 3.4.

3.1. Simulation preparation

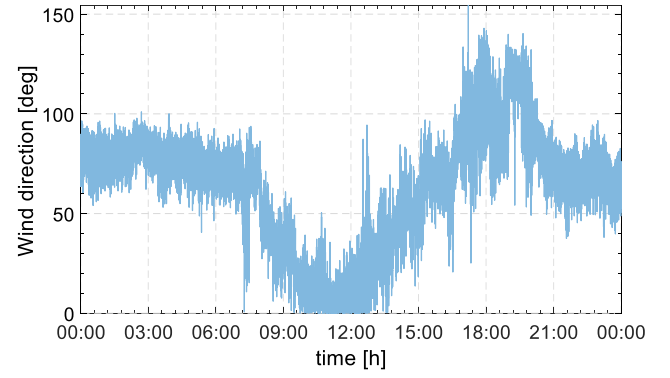
Based on the proposed method, the model is built in MATLAB/Simulink for simulation experiments to study the effectiveness of the method. The simulation test was carried out on a laptop (Intel(R) Core (TM) i7-6700HQ CPU @ 2.60 GHz with 16 GB RAM). The parameter setting of IC-BO and SMPYC is shown in Table 1.

The wind direction data used in the simulation comes from the real wind direction of a certain wind farm within a day with the sampling period being 1 s. The adjustable parameters of LSTM-WDP have a greater impact on the prediction accuracy, which should be selected carefully. After test, the settings of each parameter are shown in Table 2. From Table 2, m is the prediction step of SMPYC, Adam is a classic optimization algorithm, which is direct and efficient, with less parameter adjustment and fast convergence speed, and MSE is mean squared error. The entire network is trained through the Adam algorithm and the back-propagation rule, and the error cost function gradually goes back from the future time to the historical time. The entire LSTM network is optimized in the process, so that the output of the model is closer to the true value of the wind direction. It should be noted that because LSTM-WDP serves the MPYC system, it is actually a dynamic ultra-short-term timing prediction. As the control horizon progresses, the data set is continuously moved backwards, which dynamically predicts the wind direction. Therefore, N_h is chosen to be a small value.

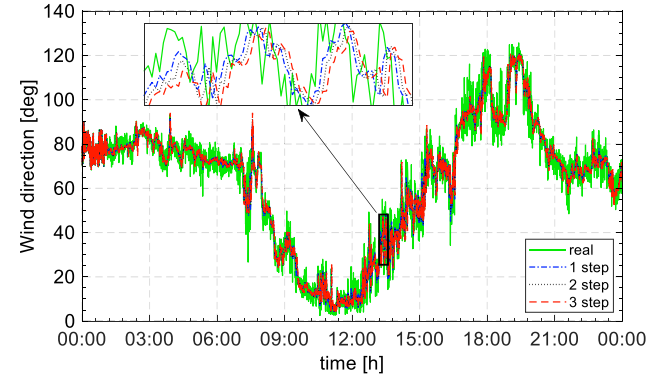
Fig. 6(a) shows the wind direction data used in the simulation. The input of SMPYC is the average wind direction of each control period, namely Fig. 6(b). The blue, grey and red curves are the average wind direction predicted by LSTM-WDP, and the green curve is the real average wind direction. Therefore, errors between the two can be seen. Fig. 7 shows the probability density distribution of the prediction error and the PDF fitted by KDE. It can be found that the error is approximately Gaussian distribution under large samples and tends to concentrate with the decrease of m , which is line with the fact that the smaller m is, the more accurate the prediction closer to the real wind direction is. On this basis, the effect of intelligent scenario generation is first discussed.

Table 2
The parameters of LSTM-WDP.

The parameters of LSTM-WDP	Value
Data set length N_h	200
Prediction set length N_p	m
Training set length	$N_h - N_p$
Test set length	N_p
Training method	Adam
Loss function	MSE
Hidden layer nodes	100
Training times	100
Initial learning rate	0.002
Drop factor	0.2



(a)



(b)

Fig. 6. 24-hour wind direction: (a) The actual wind direction; (b) The real average wind direction and the predicted average wind direction.

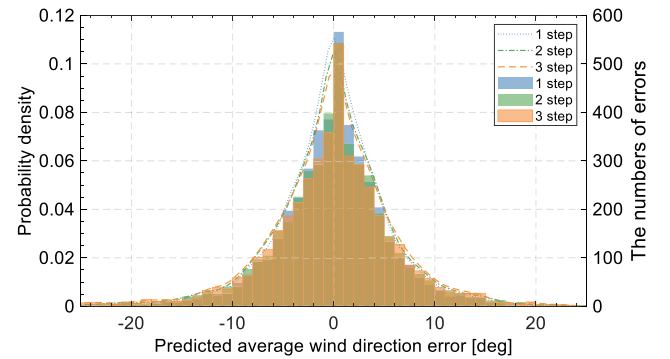


Fig. 7. Average wind direction error histogram and probability density curve by KDE.

3.2. Analysis of optimization results

Under the assumption that the PDF described in Fig. 7 is known, the results of ISG are discussed and analyzed. In reality, the statistical information of wind direction in the region where the wind turbine is located is known, so this assumption is reasonable. According to the PDF, the upper and lower limits of variables in Eq. (8) and the corresponding probabilities of each generation scenario can be determined. After testing, the values of ω_1 , ω_2 , and ω_3 are set to 1.25, 0.85 and 500, respectively. IC-BO obtains the optimal scenarios by solving Eq. (8) online in each control period. Fig. 8(a) and (b) show the optimal scenarios corresponding to 11:06:00 and 14:50:40, respectively. To facilitate the observation, the scenarios has been combined with the corresponding predicted wind direction. It can be seen from Fig. 8 that the scenarios of the two both possess good encapsulation for the real

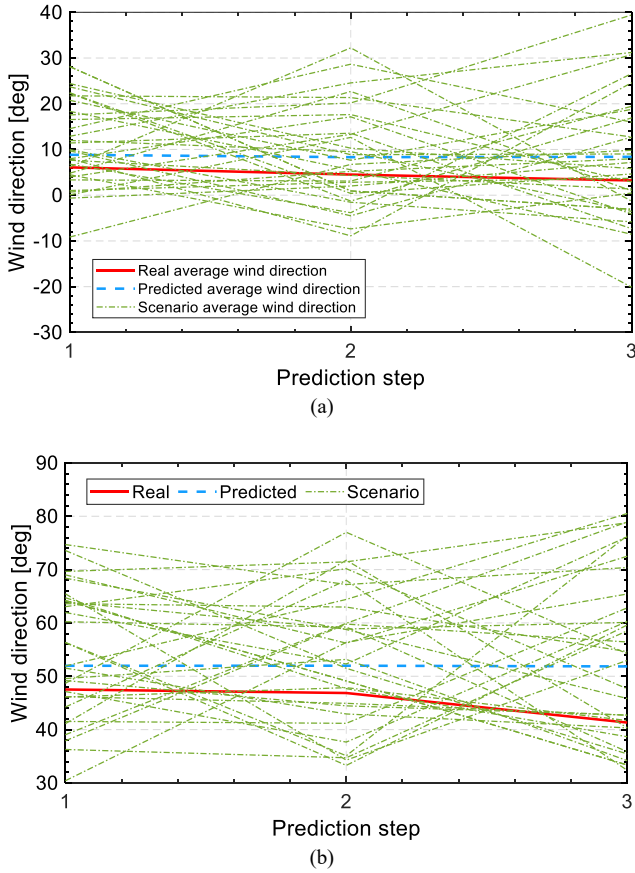


Fig. 8. The generated scenario diagram: (a) Generated at 11:06:00; (b) Generated at 14:50:40.

Table 3
Parameters of three algorithms.

Algorithms	IC-BO	BO	PSO
Parameters	$N = 150$ $T = 200$	$N = 150$ $T = 200$	$N = 150$ $T = 200$ $c_1 = 1.4955$ $c_2 = 1.4495$

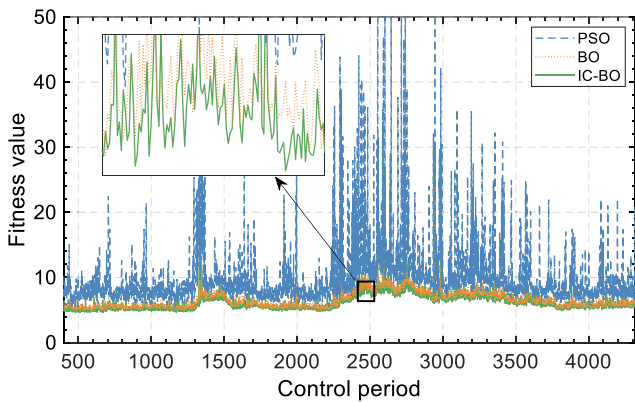


Fig. 9. Fitness value of each algorithm.

wind direction. Among them, the scenarios in Fig. 8(a) are aggregated, while those in Fig. 8(b) are dispersed. This is because in Fig. 8(a), the error between the predicted wind direction and the real wind direction at the corresponding time is slight. Thus, the generated scenarios only

need to capture less features to achieve compensation. By comparison, the error in Fig. 8(b) is apparent and the generated scenarios need more features to package it.

In order to verify the performance of IC-BO, IC-BO, BO and PSO are utilized to optimize the problem Eq. (8). The common parameters of each algorithm are set as shown in Table 3. N is the population size, T is the number of iterations, c_1 and c_2 are the acceleration coefficients of PSO, respectively.

Fig. 9 denotes the optimal fitness value of each algorithm in each control period during the whole simulation time. Overall, the fitness value obtained by IC-BO is the smallest; although the optimization result of BO is close to that of IC-BO, it can be observed from the partial amplification that the optimization result of IC-BO is smaller than that of BO. Additionally, the optimization results of PSO are the worst. It can be seen obviously that the results of PSO are quite different from those of IC-BO and BO, and there are more burrs. The above phenomena indicate that BO itself has better capability to handle high-dimensional and multimodal complex optimization problem, and the IC-BO has stronger global optimization ability on this basis. Contemporarily, many non-convergence phenomena emerge in PSO in many times of operation, and the processing ability of this optimization problem is poor. In addition, the overall fitness value of any algorithm is fluctuating, because the external conditions have changed in each optimization, and the optimal solution of the problem has also altered accordingly. Two large fluctuations of fitness value occurred near the number of control periods of 1300 and 2300, namely about 7:00 and 13:00, respectively. It can be found from Fig. 6(b) that the wind direction at these two nodes of time is producing severe fluctuations, and the error of predicted wind direction is also large, which makes the generation of scenarios demanding and the fitness value increased.

Furthermore, Table 4 shows the statistical result of each algorithm, where T_{run} is the time cost in each operation, and Fit is the optimal fitness value obtained from each operation. The mean and standard deviation of fitness value of IC-BO are the smallest, which are 6.257 and 1.420, respectively. BO followed by 6.606 and 1.520, respectively; PSO is the worst, 10.185 and 5.249, respectively. PSO has the shortest running time of 15.201 s; BO followed by 16.189 s; the running time of IC-BO is 17.104 s. The above statistical results further illustrate the advantages of IC-BO in solving this problem. Although the corresponding cost is the increase of time cost, the calculation cost is similar to that of the other two algorithms and is still less than the control period of SMPYC, which meets the requirements of real-time control. Overall, IC-BO is more efficient for Eq. (8).

3.3. Comparative study of SMPYC control performance

Whether the optimal scenarios generated by ISG can improve the performance of the yaw system under the condition of inaccurate wind direction prediction needs to be evaluated. For comparison, a baseline MPYC (BMPYC) was realized [26]. Fig. 10 shows the operation of two controllers in the wind direction shown in Fig. 6. Overall, the proposed SMPYC achieves 98.06% energy capture efficiency at 7.15% yaw time ratio, while BMPYC achieves 97.80% energy capture efficiency at 8.73% yaw time ratio. Compared with BMPYC, the proposed method reduces the yaw time ratio by 1.58% and improves the energy capture efficiency by 0.26%, which denotes that for a 5 MW WT, even at 30% available hourly rate per year, it can reduce the yaw time ratio and increase the power output by 34,164 kWh at the same time, and it can reach $1.14 \times$

Table 4
Statistical results of algorithms.

Algorithms	Mean(Fit)	STD(Fit)	Mean(T_{run})(s)	STD(T_{run})(s)
PSO	10.185	5.249	15.201	0.821
BO	6.606	1.520	16.189	0.951
IC-BO	6.257	1.420	17.104	1.290

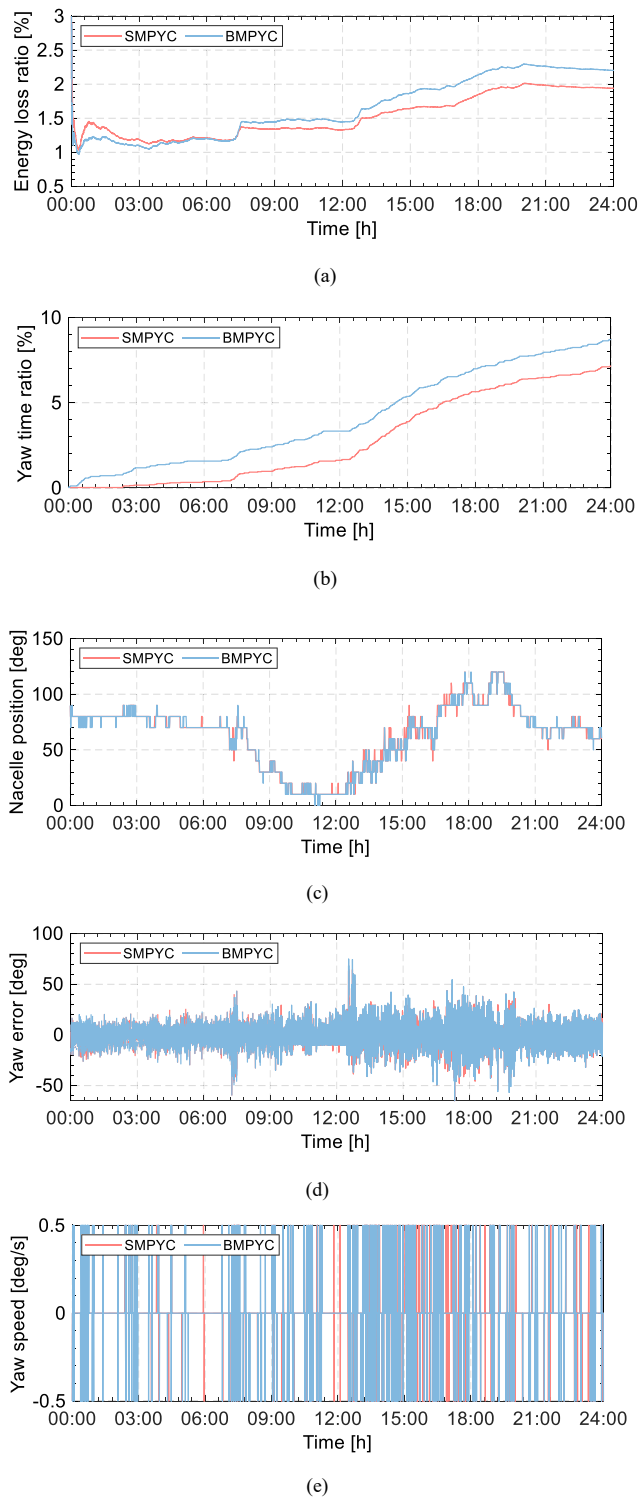


Fig. 10. Simulation results of controllers: (a) Energy loss ratio; (b) Yaw time ratio; (c) Nacelle position; (d) Yaw error; (e) Yaw speed.

Table 5

The performance of SMPYC by three scenario generation method.

Method	Yaw time ratio (%)	Energy loss ratio (%)
Monte Carlo Sampling	7.245	2.239
Latin Hypercube Sampling	7.245	2.233
Intelligent Scenario Generation	7.153	1.940

10⁵ kWh at full power state, which can fully demonstrate the effectiveness of the proposed SMPYC in improving yaw performance. It can be seen from Fig. 10(c)–(e) that the performance improvement of SMPYC is embodied in the overall decrease of the yaw error of the WT, which results from the fact that the optimal scenarios generated by the ISG method compensates for the influence of the prediction error to a certain extent. Consequently, it avoided unnecessary yaw actions, which not only decreases the yaw time, but also minimizes the energy capture loss caused by the wrong action.

In addition, different scenario generation methods can also affect the effect of SMPYC. The two random scenarios generation methods are realized to obtain comparison, the first is based on Monte Carlo sampling, and the second is based on Latin hypercube sampling [46]. Table 5 summarizes performance of three SMPYCs based on different scenarios generation methods. SMPYC based on ISG has the best control performance, and the other two random scenarios generation methods achieve 97.761% and 97.767% energy capture efficiency at 7.245% yaw time ratio, respectively. The control effect is very close and both are worse than that of ISG method. In addition, the yaw time ratio of SMPYC based on the two methods is lower than that of BMPYC, but its energy capture efficiency is slightly lower than that of BMPYC. The decrease of the yaw time ratio is at the cost of energy capture loss, which also indicates that SMPYC generated by random methods cannot bring substantial improvement to BMPYC. The reason for this phenomenon is that the fluctuation of wind direction is too large and the statistical characteristics in different time periods are disparate. The random scenarios generation cannot accurately represent the characteristics of the predicted wind direction error under each control period, and accordingly, the yaw performance cannot be improved. However, ISG compensates the error well and achieves better control effect by dynamically generating the optimal scenarios in each control period.

3.4. Depth analysis of SMPYC

The above results and discussions show that the proposed ISG-SMPYC has better control effect than BMPYC and other SMPYC based on random scenarios generation. Furthermore, the accuracy of predicted wind direction and the weight coefficient in the SMPYC objective function also have an impact on control effect. The former determines the performance improvement space of ISG-SMPYC; the latter determines the trade-off between the two objectives of SMPYC, thus influencing the yaw action.

The predicted wind direction with different statistical characteristics can be obtained by adjusting the parameters of LSTM-WDP. Table 6 shows the statistical characteristics of different predicted wind direction sequences, where the predicted wind direction No. 7 is the wind data used in the above discussion. The involved LSTM parameters include the data set length N_h and the training model, where N_h is set to four possible

Table 6

Statistical characteristics of predicted wind direction sequence under different parameters.

N_h	Training method	Feature of prediction error		Number
		Mean (deg)	STD (deg)	
50	Adam	−0.0573	7.1995	1
	SGDM	−0.0687	8.0236	2
	RMSProp	−0.0726	7.8762	3
100	Adam	−0.0797	6.4289	4
	SGDM	−0.0926	6.7326	5
	RMSProp	−0.0892	6.6233	6
200	Adam	−0.1150	6.0010	7
	SGDM	−0.1248	6.0080	8
	RMSProp	−0.1209	6.1394	9
300	Adam	−0.1287	5.9745	10
	SGDM	−0.1306	6.0030	11
	RMSProp	−0.1362	6.0100	12

values of 50, 100, 200, and 300, and the training models include Adam, SGDM, and RMSProp. From Table 6, as the N_h decreases, the Mean of the prediction error becomes smaller, but the STD becomes larger. This may be because the smaller dataset easily causes overfitting during training, which makes LSTM_WDP unable to capture fluctuations in wind direction, resulting in an increase in STD and a slight decrease in Mean. As the dataset grows, the LSTM_WDP could capture more data features, and the STD decreases, but the Mean also increases slightly with the fluctuation of the predicted value. On the other hand, compared with SGDM and RMSProp, the LSTM-WDP trained by Adam has better prediction performance and is optimal under all four dataset lengths. This may be because Adam essentially combines both SGDM and RMSProp and thus has better results.

Table 7 shows the simulation results of the 12 predicted wind direction sequences numbered in Table 6, where No. 7 is the situation in the preceding discussion. From Table 7, SMPYC can reduce the energy capture loss ratio by 0.26%–0.43% and the yaw time ratio by 0.35%–1.58% compared with BMPYC under different predicted wind direction sequences. The more inaccurate the predicted wind direction, the greater the room for improvement. In addition, the mean value of the predicted wind direction error has a small impact on the performance improvement of SMPYC, while the change of the STD has a greater impact. The larger the STD, the more enhancement of SMPYC. This may be because the mean value is small and fluctuates only in a small range, while the STD fluctuates greatly.

Fig. 11 shows the yaw time ratio and energy capture loss of SMPYC and BMPYC with different ω under predicted wind direction 7, where rBMPYC represents BMPYC implemented under real average wind direction and the arrow indicates the direction of ω increase. With the increase of ω , the three curves all show a trend of decreasing yaw time ratio and increasing energy capture loss ratio, and the change range of energy capture loss ratio is much smaller than that of yaw time ratio. This is because with the increase of ω , the proportion of yaw time ratio in the objective function Eq. (4) increases, and the controller tends to reduce the yaw action, which leads to the increase of energy capture loss and the decrease of yaw time. And the variation amplitude indicates that it is difficult to obtain the improvement of energy capture by adjusting ω , and a substantial increase in yaw time ratio is needed. On the other hand, SMPYC is overall located below BMPYC, while rBMPYC is overall located below SMPYC. This shows that SMPYC can capture more wind energy than BMPYC at the same yaw time ratio, and fully demonstrates the effectiveness of ISG for improving yaw performance. However, there is still a noticeable gap between the performance of SMPYC and rBMPYC, indicating that there might be improvement space.

4. Conclusion

In this study, a Stochastic Model Predictive Yaw Control based on Intelligent Scenarios Generation (ISG-SMPYC) is proposed, which can

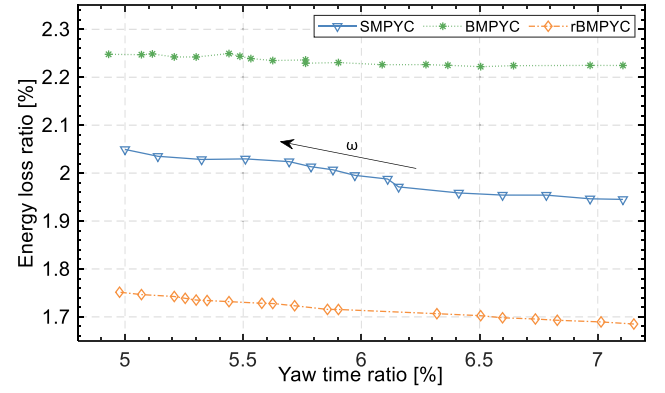


Fig. 11. Controller performance under different ω .

effectively improve the energy capture efficiency of WT when the predicted wind direction is inaccurate. The main conclusions encompass:

- ISG-SMPYC reduces the yaw time ratio by 1.58% and improves the energy capture efficiency by 0.26% compared with the baseline MPYC. For a 5 MW WT, the gained energy production could reach 1.14×10^5 kWh in a year.
- The proposed intelligent scenarios generation method (ISG) can characterize the uncertainty of predicted wind direction better than the conventional random scenarios generation method by creating the optimization problem of scenarios generation. SMPYC based on ISG reduces the yaw time rate by about 0.1 % and improves the energy capture efficiency by about 0.29 % compared with SMPYC based on random method.
- The improved IC-BO algorithm is more effective than the baseline BO and PSO for this high-dimensional multimodal optimization problem. It slightly increases the computational cost of about 1 s and 2 s, respectively compared with BO and PSO, and upgrades the average best fitness value by 5.6% and 38.6% respectively, which is also more stable.
- Predicted wind direction has a certain impact on the performance of SMPYC. Compared with BMPYC, SMPYC reduces the yaw time ratio by 0.35%–1.58% and improves the energy capture efficiency by 0.26%–0.43% under different predicted wind direction sequences. Correspondingly, the gained energy production could reach 1.14 – 1.88×10^5 kWh in a year. The more inaccurate the predicted wind direction is, the more performance enhancement of SMPYC.

With the large-scale development of wind energy and the popularization of wind prediction, large-scale wind farms generally have prediction capabilities and centralized control centers. By deploying the control algorithm in centers, scenarios can be generated in real time by ISG based on the predicted wind direction, then the yaw action

Table 7
Simulation results under different predicted wind direction sequences.

Prediction wind direction	Energy capture loss ratio (%)			Yaw time ratio (%)		
	SMPYC	BMPYC	Reduction	SMPYC	BMPYC	Reduction
1	1.936	2.293	0.357	9.306	10.116	0.810
2	1.968	2.400	0.432	9.884	10.231	0.347
3	1.952	2.368	0.416	9.514	10.046	0.532
4	2.001	2.326	0.325	9.815	10.532	0.717
5	1.996	2.308	0.312	9.931	10.578	0.647
6	1.998	2.326	0.328	9.884	10.370	0.486
7	1.940	2.200	0.260	7.150	8.730	1.580
8	2.033	2.331	0.298	8.843	9.792	0.949
9	2.021	2.324	0.303	8.241	9.375	1.134
10	1.962	2.219	0.257	7.940	8.500	0.560
11	1.998	2.259	0.261	8.519	9.028	0.509
12	2.003	2.296	0.293	8.819	9.375	0.556

commands will be calculated through SMPYC. Finally, the yaw action of the WT can be well controlled. In this way, the economic benefits of the wind farm can be improved without increasing the hardware cost. The future study will mainly unfold from the following two aspects:

- Although ISG-SMPYC has achieved relative satisfactory control results, there remains improvement room for the control performance. In the future, how to further improve the performance of SMPYC will be explored to further enhance the energy capture efficiency;
- The intelligent scenarios generation method has a certain universality. In the future, the maximum wind energy capture problem of WT based on this method will be considered under the inaccurate wind speed prediction.

CRedit authorship contribution statement

Dongran Song: Conceptualization, Methodology, Funding acquisition, Writing – review & editing. **Ziqun Li:** Methodology, Software, Writing – original draft, Visualization. **Lei Wang:** Supervision, Funding acquisition, Writing – review & editing, Visualization. **Fangjun Jin:** Visualization, Software. **Chaoneng Huang:** Resources, Formal analysis. **E. Xia:** Investigation, Data curation. **Rizk M. Rizk-Allah:** Writing – review & editing, Visualization. **Jian Yang:** Visualization, Investigation. **Mei Su:** Software, Investigation. **Young Hoon Joo:** Data curation, Formal analysis.

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.

Acknowledgement

This work is supported by the National Natural Science Foundation of China under Grant (51875058 and 61803393), the Natural Science Foundation of Hunan Province (No. 2020JJ4751), and the Innovation-Driven Project of Central South University (No. 2020CX031), the project of Science and Technology Assistance to Xinjiang (NO.2021E02039); Chongqing Technology Innovation and Application Development Special General Project (NO. cstc2021jscx-msxm0022) and Chongqing Technology Innovation and Application Development Special Key Project (NO.cstc2021jscx-gksb0102).

References

- [1] GWEC. Global Wind Report 2021. Global Wind Energy Council, Europe; 2021.
- [2] Azam A, Ahmed A, Wang H, Wang Y, Zhang Z. Knowledge structure and research progress in wind power generation (WPG) from 2005 to 2020 using CiteSpace based scientometric analysis. *J Clean Prod* 2021;295. Art. no. 126496.
- [3] Dong X, Wang Y, Wang H, Kong L. Optimised yaw model for adaptive wind direction characteristic based on a data-driven approach. *IET Renew Power Gener* 2021;15(1):237–50.
- [4] Friis Pedersen T, Gottschall J, Kristoffersen JR, Dahlberg J-Å. Yawing and performance of an offshore wind farm. (in eng), *Proceedings* 2011.
- [5] Chen Y, Zhang D. Theory-guided deep-learning for electrical load forecasting (TgDLF) via ensemble long short-term memory. *Adv Appl Energy* 2021;1:100004.
- [6] Wang Z, Hong T, Li H, Ann Piette M. Predicting city-scale daily electricity consumption using data-driven models. *Adv Appl Energy* 2021;2:100025.
- [7] Bakhshi R, Sandborn P. Maximizing the returns of LIDAR systems in wind farms for yaw error correction applications. *Wind Energy* 2020;23(6):1408–21.
- [8] Choi D, Shin W, Ko K, Rhee W. Static and Dynamic Yaw Misalignments of Wind Turbines and Machine Learning Based Correction Methods Using LiDAR Data. *IEEE Trans Sustainable Energy* 2019;10(2):971–82.
- [9] Suvana M, et al. A machine learning framework to quantify and assess the impact of COVID-19 on the power sector: An Indian context. *Adv Appl Energy* 2022;5: 100078.
- [10] Pinto G, Wang Z, Roy A, Hong T, Capozzoli A. Transfer learning for smart buildings: A critical review of algorithms, applications, and future perspectives. *Adv Appl Energy* 2022;5:100084.
- [11] Tang ZH, Zhao GN, Ouyang TH. Two-phase deep learning model for short-term wind direction forecasting. *Renewable Energy* 2021;173:1005–16.
- [12] Shen X, Fu X, Su Z. Screening methodology of correlated wind turbines for wind direction prediction based on yawing manoeuvre data. *IET Renew Power Gener* 2020;14(19):4112–20.
- [13] Ouyang T, Kusiak A, He Y. Predictive model of yaw error in a wind turbine. *Energy* 2017;123:119–30.
- [14] Deng X, et al. Sensorless effective wind speed estimation method based on unknown input disturbance observer and extreme learning machine. *Energy* 2019; 186.
- [15] Sasser C, Yu M, Delgado R. Improvement of wind power prediction from meteorological characterization with machine learning models. *Renewable Energy* Jan 2022;183:491–501.
- [16] Zhang Y, Zhao Y, Shen X, Zhang J. A comprehensive wind speed prediction system based on Monte Carlo and artificial intelligence algorithms. *Appl Energy* 2022;305. Art. no. 117815.
- [17] Li X, Qiu Y, Feng Y, Wang Z. Wind turbine power prediction considering wake effects with dual laser beam LiDAR measured yaw misalignment. *Appl Energy* 2021;299. Art. no. 117308.
- [18] Tang G, Wu Y, Li C, Wong PK, Xiao Z, An X. A Novel Wind Speed Interval Prediction Based on Error Prediction Method. *IEEE Trans Ind Inf* 2020;16(11): 6806–15.
- [19] Heinemann J, Kramer O. Machine learning ensembles for wind power prediction. *Renewable Energy* 2016;89:671–9.
- [20] Khosravi A, Koury RNN, Machado L, Pabon JGG. Prediction of wind speed and wind direction using artificial neural network, support vector regression and adaptive neuro-fuzzy inference system. *Sustainable Energy Technol Assess* 2018;25:146–60.
- [21] Yang S, et al. Model predictive control for integrated control of air-conditioning and mechanical ventilation, lighting and shading systems. *Appl Energy* 2021;297. Art. no. 117112.
- [22] Raman NS, Devaprasad K, Chen B, Ingle HA, Barooah P. Model predictive control for energy-efficient HVAC operation with humidity and latent heat considerations. *Appl Energy* 2020;279.
- [23] Song D, et al. Maximum wind energy extraction of large-scale wind turbines using nonlinear model predictive control via Yin-Yang grey wolf optimization algorithm. *Energy* 2021;221.
- [24] Song D, et al. Maximum power extraction for wind turbines through a novel yaw control solution using predicted wind directions. *Energy Convers Manage* 2018; 157:587–99.
- [25] Song D, Fan X, Yang J, Liu A, Chen S, Joo YH. Power extraction efficiency optimization of horizontal-axis wind turbines through optimizing control parameters of yaw control systems using an intelligent method. *Appl Energy* 2018; 224:267–79.
- [26] Song DR, et al. Model Predictive Control Using Multi-Step Prediction Model for Electrical Yaw System of Horizontal-Axis Wind Turbines. *IEEE Trans Sustainable Energy* 2019;10(4):2084–93.
- [27] Song DR, Chang Q, Zheng SY, Yang S, Yang J, Joo YH. Adaptive Model Predictive Control for Yaw System of Variable-speed Wind Turbines. *J Mod Power Syst Clean Energy* 2021;9(1):219–24.
- [28] Song DR, Yang J, Liu Y, Su M, Liu AF, Joo YH. Wind Direction Prediction for Yaw Control of Wind Turbines. *Int J Control Autom Syst* 2017;15(4):1720–8.
- [29] Mayne DQ. Model predictive control: Recent developments and future promise. *Automatica* 2014;50(12):2967–86.
- [30] Zhao N, You F. New York State's 100% renewable electricity transition planning under uncertainty using a data-driven multistage adaptive robust optimization approach with machine-learning. *Adv Appl Energy* 2021;2:100019.
- [31] Oldewurtel F, Jones CN, Parisio A, Morari M. Stochastic Model Predictive Control for Building Climate Control. *IEEE Trans Control Syst Technol* 2014;22(3): 1198–205.
- [32] Wei S, Gao X, Zhang Y, Li Y, Shen J, Li Z. An improved stochastic model predictive control operation strategy of integrated energy system based on a single-layer multi-timescale framework. *Energy* 2021;235. Art. no. 121320.
- [33] Zhu D, Hug G. Decomposed Stochastic Model Predictive Control for Optimal Dispatch of Storage and Generation. *IEEE Trans Smart Grid* 2014;5(4):2044–53.
- [34] Namba T, Funabiki S, Takaba K, Ieee. Microgrid Management with PV power prediction via stochastic distributed optimization. In: 58th Annual Conference of the Society-of-Instrument-and-Control-Engineers-of-Japan (SICE), Hiroshima, JAPAN, 2019, 2019, p. 1685–1691.
- [35] Zhou Y, Shi Y, Ieee. Scenario-based stochastic model predictive control for wind energy conversion system. In: 28th Chinese Control and Decision Conference, Yinchuan, PEOPLES R CHINA, 2016, 2016, p. 6222–6227.
- [36] Shi Y, Xiang X, Zhang Y, Sun D, Ieee. Design of Stochastic Model Predictive Control for Wind Energy Conversion System. In 14th International Workshop on Complex Systems and Networks (IWCSN), Doha, QATAR, 2017, 2017, p. 108–114.
- [37] Kou P, Liang DL, Gao L, Ieee. stochastic model predictive control for wind turbines with doubly fed induction generators. In: IEEE-Power-and-Energy-Society General Meeting (PESGM), Boston, MA, 2016, 2016.
- [38] Chen Y, Wang Y, Kirschen D, Zhang B. Model-free renewable scenario generation using generative adversarial networks. *IEEE Trans Power Syst* 2018;33(3): 3265–75.
- [39] Chen YZ, Wang XY, Zhang BS, Ieee. An unsupervised deep learning approach for scenario forecasts. In: Power Systems Computation Conference (PSCC), Dublin, IRELAND, 2018, 2018.
- [40] Shouxiang W, Haiwen C, Xiaoping L, Xin S. Conditional Variational Automatic Encoder Method for Stochastic Scenario Generation of Wind Power and Photovoltaic System. *Power Syst Technol* 2018;42:8.

- [41] Zakaria A, Ismail FB, Lipu MSH, Hannan MA. Uncertainty models for stochastic optimization in renewable energy applications. *Renewable Energy* 2020;145: 1543–71.
- [42] Dokeroglu T, Sevinc E, Kucukyilmaz T, Cosar A. A survey on new generation metaheuristic algorithms. *Comput Ind Eng* 2019;137.
- [43] Das AK, Pratihari DK, Ieee. A new bonobo optimizer (BO) for real-parameter optimization. In: *IEEE-Region-10 Symposium (TENSYP)*, Kolkata, INDIA, 2019, 2019, p. 108–113.
- [44] Yang J et al. Stochastic model predictive control for the yaw control system of horizontal-axis wind turbines. In: *15th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, Electr Network, 2020, 2020, p. 478–483.
- [45] Lan-Tao X, Lei X, Hong-Ye S. A comparative study on algorithms of robust and stochastic MPC for uncertain systems. *Acta Autom Sin* 2017;43(6):23.
- [46] Lei D, Mengxia L, Naishi C, Shixiong F, Tianjiao P. Coordinated Optimal Control of Distributed Energy Based on Stochastic Model Predictive Control. *Power System Technol* 2018;42(10):7.
- [47] Arjovsky M, Chintala S, Bottou L. Wasserstein generative adversarial networks. In: *34th International Conference on Machine Learning*, Sydney, AUSTRALIA, 2017, vol. 70, 2017.
- [48] Abdelghany RY, Kamel S, Sultan HM, Khorasy A, Elsayed SK, Ahmed M. Development of an improved bonobo optimizer and its application for solar cell parameter estimation. *Sustainability* 2021;13(7). Art. no. 3863.
- [49] vandenBergh F, Engelbrecht AP. A cooperative approach to particle swarm optimization. *IEEE Trans Evol Comput* 2004;8(3):225–39.
- [50] Crosson E, Harrow AW, Ieee. Simulated quantum annealing can be exponentially faster than classical simulated annealing. In: *57th IEEE Annual Symposium on Foundations of Computer Science (FOCS)*, New Brunswick, NJ, 2016, 2016. p. 714–23.