Reliability aware multi-objective predictive control for wind farm based on machine learning and heuristic optimizations

Xiuxing Yin[a], Xiaowei Zhao[a], Jin Lin[b] and Aris Karcanias[c]

[a] School of Engineering, the University of Warwick, Coventry CV4 7AL, U.K. (e-mail: x.yin.2@warwick.ac.uk)

[b]Department of Electrical Engineering, Tsinghua University, Beijing, 100084, P. R. China. (e-mail: linjin@tsinghua.edu.cn).

[c]FTI Consulting, London EC1A 4HD, U. K. (e-mail: aris.karcanias@fticonsulting.com)

Corresponding author: X. Zhao (e-mail: xiaowei.zhao@warwick.ac.uk)

Abstract:

In this paper, a reliability aware multi-objective predictive control strategy for wind farm based on machine learning and heuristic optimizations is proposed. A wind farm model with wake interactions and the actuator health informed wind farm reliability model are constructed. The wind farm model is then represented by training a relevance vector machine (RVM), with lower computational cost and higher efficiency. Then, based on the RVM model, a reliability aware multi-objective predictive control approach for the wind farm is readily designed and implemented by using five typical state of the art meta-heuristic evolutionary algorithms including the third evolution step of generalized differential evolution (GDE3), the multi-objective evolutionary algorithm based on decomposition (MOEA/D), the multi-objective particle swarm optimization (MOPSO), the multi-objective grasshopper optimization algorithm (MOGOA), and the non-dominated sorting genetic algorithm III (NSGA-III). The computational experimental results using the FLOw Redirection and Induction in Steady-state (FLORIS) and under different inflow wind speeds and directions demonstrate that the relative accuracy of the RVM model is more than 97%, and that the proposed control algorithm can largely reduce thrust loads (by around 20% on average) and improve the wind farm reliability while maintaining similar level of power production in comparison with a conventional predictive control approach. In addition, the proposed control method allows a trade-off between these objectives and its computational load can be properly reduced.

Key words:

Wind farm reliability; Predictive control; Relevance vector machine; Evolutionary algorithms.

1. Introduction

1.1. *Background and motivation*

The wind turbines under a wind farm are subject to unavoidable failure rates caused by the highly intermittent and inherently stochastic nature of the wind and environment [1]. Their failures result in increased operation and maintenance (O&M) costs, and consequently, increased cost of total energy production. The O&M currently consist of a considerable portion of the total wind energy costs, in particularly for the offshore case (up to 30%) [2]. As a result, the reliability is particularly important for wind farm operations, particularly for the offshore case. The structural loads caused by incoming wind are a major contributor to the wind turbine structural failures. The control and actuation (such as the electric/hydraulic pitch, yaw and torque control components) components in wind turbines account for more than 65% of the total failures [3]. In order to improve the wind farm's power generation and reliability while reducing O&M costs, it is necessary and imperative to design and implement efficient control approaches to achieve high energy capture efficiency and low maintenance costs [4].

On the other hand, the wind energy industry is also a data rich sector with a large amount of data generated every day due to the fast development of network technology and computing power [5]. These big data can be used to optimize operations of the wind farm by providing efficient and effective decisions. Meanwhile, artificial intelligence (AI) technology continues to advance, which offers great potential for optimizing the daily wind farm operations. Therefore, the big-data driven AI approaches see a great opportunity to be employed to the automatic wind farm operations due to their numerous tangible benefits such as increased system efficiency, stability and reliability [6].

1.2. *Literature review*

Recent years, the optimal control for improving wind farm operations and reliability has become a hot topic. In [7], an optimization framework was presented to allow the optimization of turbulent wind-farm boundary layers, where gradient-based optimization methods were used with the aim of increasing the total farm energy extraction. In [8], the game theory and cooperative control were used to optimize energy production of wind farms. In [9], the wind farm was modeled as a nonlinear steady-state model and a distributed optimization was employed as a global optimization framework to control all the turbines. The use of a cooperative wind farm control approach to improve the power production of a wind farm was described in [10]. A decentralized model-free approach was presented in [11] for wind farm power optimization with only limited information sharing among neighbor wind turbines. A wind farm control strategy was presented in [12] to optimize the yaw settings of wind turbines by taking into account the wake effects. A wind farm optimization strategy under uncertainty was formulated and solved in [13] to optimally steer the wakes in the presence of yaw angle uncertainty.

The afore-mentioned wind farm optimization/control designs relied highly on explicit information from the mathematical model of wind farm, whose results may be suboptimal for such a complicated, distributed and uncertain system. The data-driven model-free optimization approaches may provide a better solution. To the best knowledge of the authors, there are very limited researches in this area. The feasibility of a data-driven coordinated control approach, Bayesian Ascent (BA) algorithm, was explored in [14] to maximize the total wind farm power production. In [15], the cooperative wind farm control was studied to maximize the total wind farm power generation by incorporating a strategy

that regulates the trust region into the Bayesian optimization framework.

Nevertheless, the above data-driven approaches still highly depend on an analytical wind farm power function and are tailored to specific experimental studies using scaled wind turbines. The employed BA algorithm is also composed of online learning and optimization phases. The BA algorithm approximates the target function using Gaussian process regression in the learning phase, and determines the next sampling point to improve the target value in the optimization phase. Therefore, this BA approach is naturally flawed with limited regression capability and control flexibility considering volatile wind directions, and inevitably involves additional large computational burden. Besides, the aforementioned approaches generally use simple single distribution optimization while the reliability aspects of wind farms have not been taken into account.

1.3. *Contributions of the work*

The present paper aims to develop a reliability aware multi-objective predictive control approach for wind farm based on machine learning and heuristic optimizations. The high-fidelity wind farm model is represented by training the RVM with low cost and high efficiency. Consequently, a multi-objective predictive control strategy is conducted based on the trained RVM to maximize the power generation and reliability of wind farm, and simultaneously minimize its thrust force (thus reducing maintenance costs and maximizing lifetime of the wind turbines). Five typical state of the art meta-heuristic evolutionary optimization algorithms including the third evolution step of generalized differential evolution (GDE3), the multi-objective evolutionary algorithm based on decomposition (MOEA/D), the multi-objective particle swarm optimization (MOPSO), the multi-objective grasshopper optimization algorithm (MOGOA), and the non-dominated sorting

genetic algorithm III (NSGA-III) are used in the proposed control method which is then tested and compared with a conventional predictive control method. The extensive computational experiments based on FLORIS are conducted to verify the effectiveness of the proposed control.

The RVM model and its advanced version, hybrid RVM model, have been employed in the recent research including wind speed/power predictions, reference evapotranspiration forecasting and prognostics. In [16], a wind speed prediction approach was designed by using the hybrid model of wavelet decomposition and artificial bee colony algorithm-based RVM model. In [17], a hybrid RVM model was presented to predict future daily reference evapotranspiration. In [18], a hybrid RVM wind power probabilistic forecasting model was designed by using five kernel functions, and its forecasting performance was demonstrated. In [19], a RVM model was used to predict the short-term power output from raw data of a wind farm based on the differential empirical mode decomposition. In [20], a hybrid prognostic scheme combining the RVM and particle filter was presented for uncertainty assessment. However, the above-mentioned recent applications of the RVM or the hybrid RVM models mainly focused on the prediction or forecasting, while control designs based on RVM model, especially the wind farm predictive control, have not been investigated.

The main novelty and contributions of the paper are highlighted as follows:

(a) Distinguished from the previous approaches that rely on detailed analytical wind farm model, the proposed data driven RVM model framework is computationally efficient, highly accurate and simple in multi-objective wind farm modeling.

(b) The proposed multi-objective predictive control framework not only can maximize the wind farm power generation, but also can reduce wind farm thrust loads and

simultaneously improve the wind farm reliability. It allows a trade-off between these three objectives. The predictive control performances can be readily ensured through large-scale realistic data, which indicates good potential in practical applications.

(c) Different from other existing approaches, the proposed multi-objective predictive control approach can be well utilized to find a set of non-dominated optimal trade-off solutions among several competing objective functions by explicitly considering the effects of different constraints in control inputs. It is also characterized by decoupled prediction and predictive control, higher flexible extendibility and universal regression capabilities in terms of inputs, outputs and short time window.

2. Wind farm modelling

The output power and thrust force are two key parameters in evaluating power generation performance and reliability of a wind farm, respectively. In addition, the reliability is also significantly influenced by the failure rates of the turbine actuators. Therefore, this section focuses on developing a wind farm model by considering the thrust force and actuator health informed wind farm reliability.

2.1. *The wind farm model with wake interactions*

For a wind farm consisting of N wind turbines denoted by the set $D = \{1, 2, \ldots, N\}$, its model is highly characterized by turbine wake interactions which can be described by a multi-zone model [12]. In the multi-zone model, three wake zones (near-wake zone, far-wake zone, and mixing-wake zone) are defined to model the effects of partial wake overlap and wake velocity profile, especially in yawed conditions. For a two-turbine model as shown in Fig. 1, the effective wind velocity at the downstream turbine j can be modelled by combining the effects of the wake zone of each upstream turbine i.

$$V_j = V_\infty \cos(\varphi - 270)\left(1 - 2\sqrt{\sum_{i \in D: X_i < X_j}\left(a_i \sum_{q=1}^{3}\left(\frac{D_i}{D_i + 2k_e m_{U,q}(\gamma_i)(X_j - X_i)}\right)^2 \min\left(\frac{A_{i,j,q}^{ol}}{A_j}, 1\right)\right)^2}\right) \quad (1)$$

where V_∞ is the free stream inflow wind speed, φ is the angle of wind direction as shown in Fig. 1, $A_{i,j,q}^{ol}, q = 1, 2, 3$ is the overlapping areas of the three wake zones with the turbine rotor (q = 1, 2, 3 corresponding to the three wake overlap zones), X_i and X_j are respectively the x-axis locations of the turbines i and j, A_j is the rotor area of the downstream turbine j, D_i, a_i and γ_i are respectively the rotor diameter, axial induction factor and yaw offset of the turbine i, k_e is a wake coefficient defining both wake expansion and wake recovery.

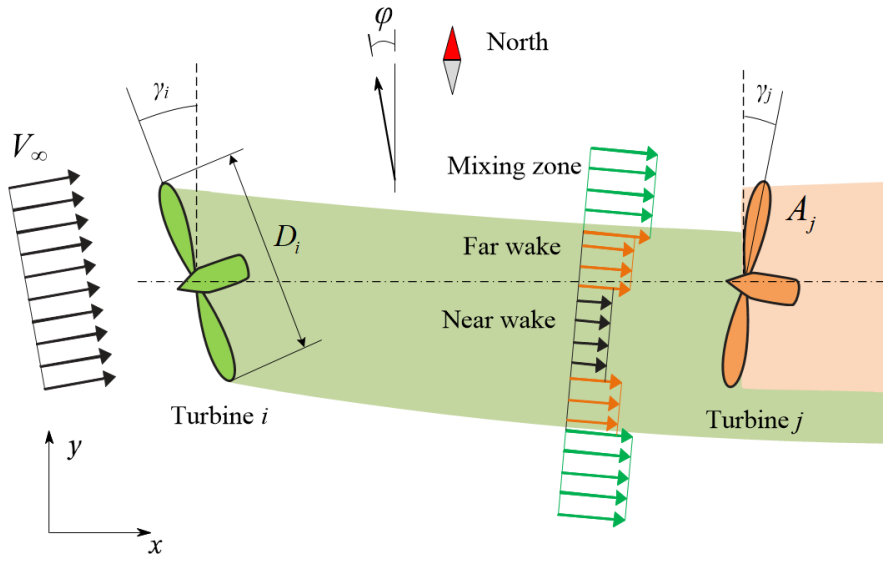

Fig. 1 The wake expansion model of the turbines i and j (top view). In the reference frame (x, y), the x-axis points downwind along the free stream inflow direction, the y-axis is orthogonal to the x-axis along the crosswind direction. The inflow wind direction is defined with respect to the north direction.

The $m_{U,q}(\gamma_i)$ in Eq. (1) is defined as

$$m_{U,q}(\gamma_i) = \frac{M_{U,q}}{\cos\left(a_U + b_U \gamma_i(t)\right)}, q = 1,2,3 \tag{2}$$

where $M_{U,q}$ is a tuned scaling factor, a_U and b_U denote tuned model parameters.

As shown in Eqs. (1) and (2), the effective wind velocity at the downstream turbine is not only determined by the inflow wind speed and direction, but also significantly influenced by the operating parameters (including the induction factors and yaw angles) of the upstream wind turbines. This illustrates the wake interactions within the wind farm.

The wind farm power is an aggregation of the individual wind turbine power outputs:

$$P_{avg} = \frac{1}{2N}\sum_{j=1}^{N} \rho A_j \cos^{1.88}(\gamma_j) V_j^3 C_{pj}(\lambda_j, \beta_j) \tag{3}$$

where P_{avg} is the averaged wind farm power output, ρ is the air density, and C_{pj} is the power coefficient of a typical turbine *j* in the wind farm.

The wind farm thrust force is given by

$$F_{avg} = \frac{1}{2N}\sum_{i=1}^{N} \rho A_j \cos^{1.88}(\gamma_j) V_j^2 C_{Tj}(\lambda_j, \beta_j) \tag{4}$$

where F_{avg} is the averaged thrust force, and C_{Tj} is the thrust coefficient.

The power coefficient C_{Pj} is a function of the tip speed ratio λ_j and the blade pitch angle β_j:

$$\begin{cases} \lambda_j = \dfrac{\omega_j R_j}{V_j(t)}; \\ C_{Pj}(\beta_j, \lambda_j) = 0.5176\left(\dfrac{116}{\bar{\lambda}_j} - 0.4\beta_j - 5\right)^{-\frac{21}{\bar{\lambda}_j}} + 0.0068\lambda_j; \\ \bar{\lambda}_j = \dfrac{1}{\lambda_j + 0.08\beta_j} - \dfrac{0.035}{\beta_j^3 + 1}. \end{cases} \tag{5}$$

where R_j is the rotor radius, and ω_j is the rotor speed.

The power and thrust coefficients C_{Pj} and C_{Tj} can also be represented as

$$\begin{cases} C_{Pj}(a_j) = 4a_j(1-a_j)^2; \\ C_{Tj}(a_j) = 4a_j(1-a_j). \end{cases} \tag{6}$$

where a_j is the axial induction factor of the turbine j.

By solving Eq. (6), one obtains the relationship between C_{Pj} and C_{Tj} as follows

$$C_{Pj} = \frac{C_{Tj}\left(1-\sqrt{1-C_{Tj}}\right)}{2}. \tag{7}$$

As shown in Eq. (7), the turbine thrust coefficient is directly related to the power coefficient which is described in Eq. (5). Therefore, by observing Eqs. (5) and (7), it is obvious that the thrust coefficient C_{Tj} can also be determined by the tip speed ratio λ_j and the blade pitch angle β_j.

For a typical turbine j within a wind farm, the turbine rotor torque is defined as [10]

$$T_{Wj} = \frac{\pi\rho R_j^5 \omega_j^2}{2\lambda_j^3}\cos^{1.88}(\gamma_j)C_{Pj}(\beta_j, \lambda_j) \tag{8}$$

Considering the turbine j equipped with gearbox transmission and a generator, one can obtain the drivetrain dynamics as [21]

$$J_{tj}\dot{\omega}_j = T_{Wj}(\beta_j, \lambda_j) - D_{tj}\omega_j - n_{gj}T_{ej} \tag{9}$$

where J_{tj} is the rotor inertia, D_{tj} is the external damping of the rotor, n_{gj} is the gear ratio, and T_{ej} is the generator (electromagnetic) torque.

As shown in Eq. (9), the turbine rotor speed ω_j can be controlled by regulating the generator torque which is equivalent to the generator current control through a power converter located between the generator and grid [21]. Since the rotor speed ω_j is directly

related to the tip speed ratio λ_j, the power and thrust coefficients C_{Pj} and C_{Tj} as shown in Eqs. (5) and (7), it is obvious that the power and thrust coefficients C_{Pj} and C_{Tj} can also be regulated by using the generator torque or the generator current. Therefore, due to the direct relationships among C_{Pj}, C_{Tj}, P_{avg}, and F_{avg} as shown in Eqs. (3) and (4), the averaged wind farm power and thrust force can also be regulated by using the generator torque or the generator current.

Then, by considering Eqs. (1)-(9), the averaged wind farm power and thrust force can be determined based on the inflow wind speed and direction, and turbine control parameters including the yaw angle inputs, the blade pitch angles, and the generator torque inputs. As a result, considering the time delay of wake propagation within the wind farm, the wind farm power and thrust force can be determined as a vector-valued function with sampling time interval delay as these control inputs under certain inflow wind speed and direction. Hence,

$$\boldsymbol{y}(k+1)=\left[y_1(k+1),y_2(k+1)\right]=\left[P_{avg}(k+1),F_{avg}(k+1)\right]=\boldsymbol{f}\left(\boldsymbol{u}_1(k),\boldsymbol{u}_2(k)...\boldsymbol{u}_j(k)..,\boldsymbol{u}_N(k)\right) \quad (10)$$

where $\boldsymbol{f}(\cdot)$ is the vector valued nonlinear function of the averaged power and thrust force, $\boldsymbol{u}_j=(\beta_j,\gamma_j,T_{ej})$ is the control actuation for the turbine *j*, and *k* is the sampling time instant.

2.2. *Actuator health informed wind farm reliability model*

In order to improve wind turbine efficiency and performance, the actuators for the yaw angle, blade pitch angle and generator torque in a modern wind turbine are actively regulated to generate control actions. However, the increasing use of these control efforts will deteriorate the actuator health or reliability, and will also negatively affect the overall reliability of the wind farm. Therefore, in order to guarantee the reliability, it is rational to

take the actuator health into consideration and establish the wind turbine/farm reliability model based on the loss of effectiveness of control actuators.

The reliability of the control actuators can be established based on their failure rates under degraded functional conditions, and the trend of actuator degradation according to the variations of the operating conditions needs also to be considered. Commonly, the exponential form for actuator reliability estimation is used and is directly related to the actuator control input [22].

For the *i* th actuator in the *j* th wind turbine, the actuator reliability can be modelled as a conditional probability [23]. Therefore,

$$\begin{cases} R_i(t) = \exp(-\eta_i t); \\ \eta_i = \eta_i^0 \exp\left[\dfrac{\int_0^t u_i^2(\tau)\mathrm{d}\tau}{t(u_{i\max} - u_{i\min})} \right]. \end{cases} \tag{11}$$

where $R_i(t)$ denotes the reliability of the *i* th actuator until the current instant time *t*, η_i^0 represents the nominal failure rate or baseline failure rate, $u_i(t)$ is the control effort at time *t*, $u_{i\max}$ and $u_{i\min}$ are respectively the minimum and maximum allowed control effort for the *i* th actuator.

By assuming that all the components for the turbine *j* are mutually independent, the turbine reliability can be determined by the reliability of its components. Therefore,

$$R_{Tj}(t) = 1 - \prod_{i=1}^{m} \left(1 - R_i(t)\right) \tag{12}$$

where $R_{Tj}(t)$ denotes the reliability for the *j* th wind turbine, and *m* denotes the number of actuators installed in the *j* th wind turbine.

Based on Eqs. (11) and (12), the overall reliability of the wind farm composed by *N* wind

turbines can be obtained as follows [24]

$$R_F(t) = 1 - \prod_{j=1}^{N} \left(1 - R_{Tj}(t)\right) \tag{13}$$

where $R_F(t)$ denotes the wind farm reliability at the time *t*.

By using $t=kT_s$ in discrete form, the Eq. (11) can be re-written as

$$\begin{cases} R_i(k) = \exp\left[-k \cdot T_s \cdot \eta_i(k)\right]; \\ \eta_i(k) = \eta_i^0 \exp\left[\dfrac{\sum_{\kappa=0}^{k} u_i^2(\kappa)}{k \cdot (u_{i\max} - u_{i\min})} \right]. \end{cases} \tag{14}$$

where T_s is the sampling time interval.

Accordingly, the wind turbine reliability model (12) and the wind farm reliability model (13) can be respectively transformed into the discrete time forms as $R_{Tj}(k)$ and $R_F(k)$ by using Eq. (14), which will facilitate the predictive control design in the following sections.

As shown in Eqs. (11)~(14), the wind turbine/farm reliability exhibits an inverse relationship with the control efforts delivered by the actuators. Therefore, in order to reduce the O&M costs, it is desirable to integrate actuator health information in the wind farm control and minimize the degradation rate of the most sensitive actuators such as yaw, pitch and generator torque control actuators.

3. The RVM modelling of wind farm

Although the wind farm model can be represented by Eq. (10), it is generally very difficult to derive an analytical expression due to the complex aerodynamic interactions. Also, it is rather difficult to explicitly represent stochastic meteorological conditions including wind speeds, directions and turbulence by deterministic analytical models. Even though the wind farm model can be represented by a simple and quick multi-zone model

[12], it is difficult to use such a model to fully represent the complex aerodynamic interactions under various meteorological conditions. On the other hand, the actual wind farm operations are highly characterized by large-scale realistic data, which indicates very good potential of using a surrogate model to accurately represent wind farm performances by learning the realistic data.

Therefore, rather than deriving a detailed analytical model for a wind farm, a machine learning model is trained as the surrogate model based on sample data to represent the model in Eq. (10), thereby reaping the benefits of both the lower cost and higher efficiency of wind farm modelling. The machine learning model is established based on the RVM, which is a Bayesian sparse kernel technique for regression using kernel functions [25].

3.1. *The RVM*

Considering Eq. (10) and denoting $\boldsymbol{u}=[\boldsymbol{u}_1(k),\boldsymbol{u}_2(k)...\boldsymbol{u}_j(k)..,\boldsymbol{u}_N(k)]$, $\boldsymbol{y}=\boldsymbol{y}(k+1)$, the wind farm model is designed as $\boldsymbol{y}=\boldsymbol{f}\left(\boldsymbol{u}\right)$, which can be readily represented by using the RVM. Therefore, for a given training dataset of n control input vectors $\boldsymbol{u}$ and the corresponding target output vector $\boldsymbol{y}$, the RVM can be trained to predict the target $\boldsymbol{y}$ as $\hat{\boldsymbol{y}}$. By assuming the target output includes zero-mean Gaussian noise with variance σ^2, the probability of prediction error ε is a Gaussian distribution with zero mean and variance σ^2.

The RVM model is designed as [26]

$$\begin{cases} y_i=\hat{y}_i+\varepsilon_i; \\ \hat{y}_i=w_0+\sum_{j=1}^{n} w_i K\left(\boldsymbol{u}_i,\boldsymbol{u}_j\right)=\boldsymbol{w}\phi(\boldsymbol{u}_i). \end{cases} \tag{15}$$

where y_i and $\hat{y}_i$ are respectively the i th target output and prediction, $\boldsymbol{w}=\left[w_0,...,w_n\right]$

denotes the weight vector, $\phi(\boldsymbol{u}_i)=\left[1,K\left(\boldsymbol{u}_i,\boldsymbol{u}_1\right),...,K\left(\boldsymbol{u}_i,\boldsymbol{u}_n\right)\right]^{\mathrm{T}}$ and $K\left(\boldsymbol{u}_i,\boldsymbol{u}_j\right)$ denotes the kernel function.

The likelihood function or probability of the target output $\boldsymbol{y}$ can be established as [26]

$$p\left(\boldsymbol{y}\,|\,\boldsymbol{w},\sigma^2\right)=(2\pi)^{-\frac{n}{2}}\sigma^{-n}\exp\left\{-\frac{\left\|\boldsymbol{y}-\hat{\boldsymbol{y}}\right\|}{2\sigma^2}\right\} \tag{16}$$

where $\boldsymbol{\Phi}=\left[\phi(\boldsymbol{u}_1),...,\phi(\boldsymbol{u}_n)\right]^{\mathrm{T}}$ and

$$\hat{\boldsymbol{y}}=\boldsymbol{\Phi}\boldsymbol{w}^{\mathrm{T}} \tag{17}$$

In order to avoid overfitting, a prior zero-mean Gaussian distribution is defined to complement Eq. (16) over the weight vector. Therefore,

$$p\left(\boldsymbol{w}\,|\,\boldsymbol{\alpha}\right)=(2\pi)^{-\frac{n}{2}}\prod_{\ell=1}^{n}\alpha_\ell^{\frac{1}{2}}\exp\left(-\frac{\alpha_\ell w_\ell^2}{2}\right) \tag{18}$$

where $\boldsymbol{\alpha}=\left[\alpha_0,...,\alpha_n\right]^{\mathrm{T}}$ is a hyper-parameter vector for controlling the deviation of $\boldsymbol{w}$ from zero.

By using the Bayes' rule [27], it is easy to express the posterior probability over $\boldsymbol{w}$ as

$$p\left(\boldsymbol{w}\,|\,\boldsymbol{y},\boldsymbol{\alpha},\sigma^2\right)=\frac{p\left(\boldsymbol{y}\,|\,\boldsymbol{w},\sigma^2\right)p\left(\boldsymbol{w}\,|\,\boldsymbol{\alpha}\right)}{p\left(\boldsymbol{y}\,|\,\boldsymbol{\alpha},\sigma^2\right)}=N\left(\boldsymbol{\Omega}\,|\,\boldsymbol{\mu},\boldsymbol{\Sigma}\right) \tag{19}$$

where $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ are respectively mean and covariance given as

$$\begin{cases}\boldsymbol{\Omega}=(\boldsymbol{A}+\sigma^{-2}\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi})^{-1};\\ \boldsymbol{\mu}=\boldsymbol{w}=\sigma^{-2}\boldsymbol{\Sigma}\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{y}.\end{cases} \tag{20}$$

where $\boldsymbol{A}=\mathrm{diag}\left(\alpha_0,...,\alpha_n\right)$ and $\boldsymbol{\mu}$ is the estimate of $\boldsymbol{w}$.

The vector $\boldsymbol{\alpha}$ can be estimated based on the sparse Bayesian learning that is formulated

as the local maximization logarithm $L(\boldsymbol{\alpha})$ with respect to $\boldsymbol{\alpha}$ as follows [26]

$$L(\boldsymbol{\alpha}) = -\frac{\left[n\ln 2\pi + \ln\left|\sigma^2\boldsymbol{I} + \boldsymbol{\Phi}\boldsymbol{A}\boldsymbol{\Phi}^{\mathrm{T}}\right| + \boldsymbol{y}^{\mathrm{T}}\left(\sigma^2\boldsymbol{I} + \boldsymbol{\Phi}\boldsymbol{A}\boldsymbol{\Phi}^{\mathrm{T}}\right)^{-1}\boldsymbol{y}\right]}{2} \tag{21}$$

The most probable value of the vector $\boldsymbol{\alpha}$ can be obtained by maximizing Eq. (21) with respect to $\boldsymbol{\alpha}$. Then, by substituting the value of $\boldsymbol{\alpha}$ into Eq. (20), $\boldsymbol{A}$, $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ can be obtained accordingly.

Therefore, the RVM model of the wind farm in Eq. (15) can be re-formulated as follows

$$\hat{y}_i = \mu_0 + \sum_{j=1}^{n} \mu_i \exp\left(-\frac{\left\| \boldsymbol{u}_i - \boldsymbol{u}_j \right\|}{\sigma^2} \right) \tag{22}$$

where the radial basis function $K\left(\boldsymbol{u}_i, \boldsymbol{u}_j\right) = \exp\left(-\frac{\left\| \boldsymbol{u}_i - \boldsymbol{u}_j \right\|}{\sigma^2} \right)$ is used in Eq. (22) as the kernel function.

As described in the above Eqs. (15)~(22), the RVM model produces the wind farm predictions in a probabilistic manner and the extreme sparsity of the RVM makes the predictions highly efficient. Unlike the point estimates in support vector machine (SVM), the RVM typically provides a sparser solution and the number of support vectors in the RVM grows linearly with the size of the training dataset. Therefore, the RVM avoids the principal limitations of the SVM, and typically leads to much sparser model and correspondingly faster prediction performance [28].

3.2. *The RVM modelling procedure*

Based on the above Eqs. (15)~(22), the RVM wind farm model can be constructed in two phases: a parameter selection phase and a training phase. The parameter selection phase is used to select necessary RVM parameters such as the best training set sizes. The

RVM model will then be trained and derived based on the following steps:

(1). Classify the input/output data of a wind farm into training, validation and test sets.

(2). Choose suitable RVM parameters and train the RVM for the averaged power output and thrust force using the training set.

(3). Evaluate the trained RVM model using the validation set and calculate the mean absolute percentage errors.

(4). Determine the best trained RVM model in terms of the minimum mean absolute percentage forecasting error.

4. The multi-objective predictive control

Based on the data-driven RVM model of the wind farm in section 3, the multi-objective predictive control for the wind farm can be readily designed. Therefore, in this section, the control objectives, the control formulation and the control solution are presented to detail the control procedure, which ultimately aims to maximize the profits from the wind farm and simultaneously improve the wind farm reliability.

4.1. *The control objectives*

For the multi-objective predictive control problem, three objective functions are first defined, which are equivalent to controlling the set-points for each wind turbine to maximize the power production of the whole wind farm while minimizing the mechanical thrust loads and improving the actuator health informed wind farm reliability (section 2.2.).

The control objectives are defined as follows:

(a). Maximizing the averaged wind farm power production

$$\text{obj1} = \sum_{h=1}^{H} \hat{y}_1(k+h \mid k) \tag{23}$$

where $\hat{y}_1(k+h|k)$ denotes the predicted wind farm power at the time step $k+h$ based on the

information available at the time step k by using the RVM model in section 3, H is the prediction horizon.

(b). Minimizing the averaged wind farm thrust loads

$$\text{obj2} = \sum_{h=1}^{H} \hat{y}_2(k+h \mid k) \tag{24}$$

where $\hat{y}_2(k+h|k)$ denotes the predicted wind farm thrust at the time step $k+h$ based on the information available at the time step k by using the RVM model in section 3.

(c). Maximizing the actuator health informed wind farm reliability (section 2.2.)

$$\text{obj3} = \sum_{h=0}^{H-1} R_F(k+h \mid k) \tag{25}$$

where $R_F(k+h|k)$ denotes the future wind farm reliability at the time step $k+h$ based on the information available at the time step k by using the wind farm reliability model in section 2.2.

The above three control objectives consist of the wind farm output predictions that can be computed recursively from the RVM model. The objectives are naturally contradictory to each other and hence the predictive control aims to find a trade-off solution to satisfy the requirements of all the above control objectives.

4.2. *The control formulation*

By considering the aforementioned three control objectives, the multi-objective predictive control problem of the approach can be formulated as follows.

$$\begin{aligned} &\min \{-\text{obj1}, \text{obj2}, -\text{obj3}\} \\ &\text{s.t. the RVM model in Eq. (22) and} \\ &\begin{cases} \boldsymbol{u}_{\min} \le \boldsymbol{u}(k+h \mid k) \le \boldsymbol{u}_{\max} \\ \Delta\boldsymbol{u}_{\min} \le \Delta\boldsymbol{u}(k+h \mid k) \le \Delta\boldsymbol{u}_{\max} \end{cases} \\ &h = 0, \ldots, H-1. \end{aligned} \tag{26}$$

where $\boldsymbol{u}_{\min}$ and $\boldsymbol{u}_{\max}$ are respectively the minimum and maximum values of the control inputs, $\Delta\boldsymbol{u}_{\min}$ and $\Delta\boldsymbol{u}_{\max}$ are respectively the minimum and maximum values of the variations of the control inputs.

As shown in Eq. (26), the predictive control is designed to maximize the wind farm production and reliability while minimizing the thrust loads. However, it is not easy to find the solutions that satisfy Eq. (26) since the control objective functions are not analytically known a priori. Therefore, by considering the conflicting objectives and the data driven RVM wind farm model, meta-heuristic evolutionary algorithms are designed to tackle this control problem. Unlike traditional optimization techniques based on explicit and rigorous mathematical representations and initial values, evolutionary algorithms are model-free and data-driven approaches which are able to find global optimal solutions [29]. By treating the optimization model in Eq. (26) as a black-box problem and obtaining objective variable feedback from the optimization model, the evolutionary algorithms can provide a set of compromised and alternative Pareto front solutions that trade off the objectives [30]. By using ranking and selection in the population of the Pareto-based techniques, the non-dominated, non-inferior or Pareto-optimal solutions can be generated directly by the evolutionary algorithms.

Then, the optimal solution can be obtained from the Pareto front solutions based on the meta-heuristic evolutionary optimizations. The optimal solution can be chosen as

$$\begin{cases} \boldsymbol{u}^*(k) = \arg\max\left\{\dfrac{\text{obj1}\times\text{obj3}}{\text{obj2}}\right\} \\ \boldsymbol{u}^*(k) = \begin{bmatrix} \boldsymbol{u}^*(k\,|\,k) \\ \boldsymbol{u}^*(k+1\,|\,k) \\ \vdots \\ \boldsymbol{u}^*(k+H-1\,|\,k) \end{bmatrix} \end{cases} \tag{27}$$

where $\boldsymbol{u}^*(k)$ denotes the optimal solution.

As shown in Eq. (27), the optimal solution is chosen from the Pareto set in order to maximize obj1, obj2 and simultaneously minimize the obj3. The first element of the optimal solution $\boldsymbol{u}^*(k|k)$ is applied to the wind farm to achieve the control objectives at the time step *k*.

4.3. *The evolutionary algorithms for control solution*

In order to solve the aforementioned predictive control problem, five typical state of the art meta-heuristic evolutionary algorithms are tested and compared to find the optimal control settings for each wind turbine within the wind farm. They are respectively the third evolution step of generalized differential evolution (GDE3), the multi-objective evolutionary algorithm based on decomposition (MOEA/D), the multi-objective particle swarm optimization (MOPSO), the multi-objective grasshopper optimization algorithm (MOGOA), and the non-dominated sorting genetic algorithm III (NSGA-III). These algorithms are generally inspired by reactions, biological activities and communication mechanisms in nature without relying on the derivations of a problem and gradient descent to find the global optimum.

The above five state of the art meta-heuristic evolutionary algorithms are selected because they are popular, flexible, multidimensional, and multi-stage parallel approaches, which have gradient-free and local optima avoidance mechanisms. Examples of other recent multi-objective optimization algorithms include [47]-[61]. There is no need to calculate derivative of the search space when using these algorithms. They only need the input and output information, which makes them highly flexible and suitable for solving the data-driven predictive control problem for wind farm. By using these five algorithms

to solve the wind farm predictive control problem in section 4.2 and comparing their performances simultaneously, this work aims to find and select the a suitable algorithm with the highest computational efficiency and accuracy so that it can be employed to solve the data-driven wind farm control problems in real world applications.

These five meta-heuristics algorithms belong to the family of stochastic optimization techniques and benefit from random operators, which are used to avoid local optima solutions. This mechanism ensures that the global optima solutions can be readily and accurately found by using the algorithms in finite time. In addition, by using the five meta-heuristics algorithms, the handling of the box constraints in Eq. (26) can be automatically achieved by only creating feasible solutions within the search space limited by the box constraints. The solutions that violate the constrained search space are infeasible and will not be adopted. This advantage can guarantee that the set of constraints (the box constraints on control input variables $\boldsymbol{u}$) in Eq. (26) are well handled by using these algorithms.

4.3.1. *GDE3*

GDE3 is an extension of traditional Differential Evolution (DE) for constrained multi-objective optimization using the Pareto approach [31]. In GDE3, non-dominated sorting with pruning of non-dominated solutions and a growing population are used to decrease the population size at the end of each generation, which makes the method more stable and improves the obtained diversity. The selection rule of the basic DE is modified in GDE3 such that the old vector in the next generation is replaced by the trial vector that weakly dominates the old vector in constraint violation space. There are no mechanism or sorting of non-dominated vectors in GDE3 for maintaining the extent and distribution of the solution. The number of needed function evaluations is reduced by the constraint handling

method in GDE3 and the optimal solutions are found based on the crowdedness. We refer [31] for more details about GDE3.

4.3.2. *MOEA/D*

MOEA/D is a generic, simple yet efficient multi-objective optimization approach based on decomposition. In MOEA/D, the multi-objective optimization problem is explicitly decomposed into a number of scalar optimization sub-problems which are solved simultaneously by evolving a population of solutions [32]. Each sub-problem is optimized by using the current information from its neighboring sub-problems and the neighborhood relations of the sub-problems are defined by using the distances between their aggregation coefficient vectors. The issues of diversity maintenance and fitness assignment encountered in non-decomposition optimization approaches become easier to handle in the MOEA/D due to the optimization of scalar problems. The MOEA/D uses a small population to produce a small number of very evenly distributed solutions and thus has relatively low computational complexity at each generation. It is very natural to incorporate the scalar optimization methods and objective normalization techniques into MOEA/D for coping with disparately scaled objectives. We refer [32] for more details.

4.3.3. *MOPSO*

The particle swarm optimization (PSO) is a global meta-heuristic approach inspired by the choreography of a bird flock and is designed based on swarm intelligence. The PSO needs fewer parameters and is easy to implement as compared with the other meta-heuristics algorithms. Due to the high speed of convergence of the algorithm in single-objective optimization, the PSO has been extended to deal with multi-objective optimization problems, called MOPSO.

In this paper, MOPSO [33] is used to strengthen the optimization ability in multi-objective wind farm predictive control. The algorithm improves the exploratory capabilities of PSO by introducing a range-varying mutation operator and adding a constraint-handling mechanism, which then considerably improves the exploratory capabilities of the original algorithm in [34].

In addition, an external repository is used to keep a historical record of the non-dominated particles with respect to the used repository. The external repository mainly consists of an archive controller and an adaptive grid. The archive controller is used to decide whether a certain particle solution should be added to the archive or not while the grid is used to produce well-distributed Pareto fronts. A prominent merit of the above MOPSO approach is the exceptionally low computational requirement and fast convergence speed, which makes it suitable for solving the above wind farm control problem.

4.3.4. *MOGOA*

MOGOA is inspired from the navigation of grass hopper swarms in nature and is used to find the set of best non-dominated solutions by simulating the swarming behavior of grasshoppers [35]. An individual grasshopper is modelled from the aspects of attraction force, repulsion force, and comfort zone. The position of the grasshoppers represents a possible solution of the optimization problem and mainly includes three components: social interaction, impact of gravitational force, and wind advection [36], [37].

The evolution of the movement of grasshopper position in (5) leads to the projection of Pareto optimal solutions in the search space, which are then stored in a Pareto optimal front set. In order to form the set, a special adaptive mechanism called roulette wheel is utilized

to select the target that leads to the grasshoppers towards the Pareto optimal front set. In case of premature convergence, the solutions with crowded neighborhood are deliberately removed to reduce the crowded regions and add new solutions in the less populated regions. The MOGOA algorithm can be generally implemented by using the "unified framework" by Padhye et al. [38], [39].

The MOGOA algorithm has advantages in smoothly balancing exploration and exploitation with very fast convergence speed. These characteristics make the MOGOA algorithm appropriate to deal with the multi-objective predictive control of the wind farm with a large amount of distributed wind turbines.

4.3.5. *NSGA-III*

The newly developed third version of the non-dominated sorting genetic algorithm (NSGA-III) is extended from the well-known NSGA-II by Jan and Deb [40], [41]. The NSGA-III is specially designed to deal with multi-objective optimization problem by using a reference point approach and can obtain a uniform distribution of Pareto solutions with relatively low computation complexity.

Compared with NSGA-II, NSGA-III has significant changes in the selection operator and population diversity, maintained by supplying and adaptively updating a number of well-spread reference points [42]. In NSGA-III, the non-domination level is selected to construct a new population from the combined parent and offspring population, thereby enabling the preservation of elite members of the parent population. Then, the combined population is sorted according to different non-domination levels and the solutions that will maximize the diversity of the combined population are chosen. Different from NSGA-II that uses a niche-preservation operator to calculate the crowding distance for every last

level member [43], the solutions of NSGA-III have larger crowding distances and the crowding distance operator is improved. In NSGA-III, the diversity in the solutions is ensured by a predefined set of reference points that can either be predefined in a structured manner or supplied preferentially by the user. The ideal solution is determined by identifying the minimum value of each objective function. NSGA-III adaptively maintains a diversity in the search space spanned by the population members since the extreme points are used in the normalization procedure and the hyper-plane creation from the start.

5. Results and discussions

In this section, extensive computational experiments of yaw angle control are conducted by using a novel internal parametric model for wake effects called the FLORIS. Here the simulation setup comprises of 4 × 5 MW wind turbines as shown in Fig. 2. A few different test cases have been employed to evaluate the performances of the proposed control method under different operational scenarios including different wind speeds and directions. Consequently, comparative validations are conducted by comparing the proposed control approach with a conventional single-objective differential evolution (DE) algorithm for maximizing the wind farm power capture under the same operational conditions.

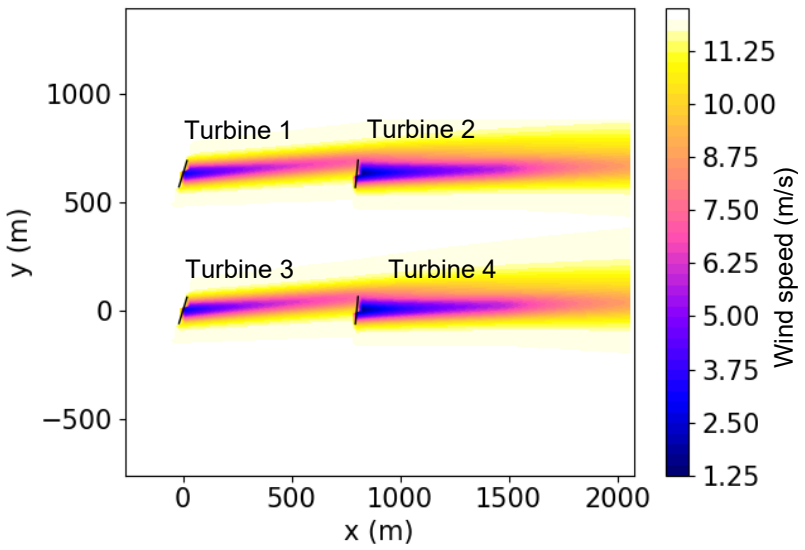

Fig. 2 The configuration of the wind farm in FLORIS. The inflow wind comes from the left.

5.1. *Description of the data set and preprocessing*

The data set for training and testing the RVM wind farm model in Fig. 2 was provided by using the FLORIS which implements a 3 dimensional version of the Jensen model, the curl model, and the Gaussian wake model [44], [45]. The input and output data samples are respectively the yaw angles of the four wind turbines, and the averaged power and thrust force of the wind farm as illustrated in Eq. (10). The inflow wind speeds and directions have substantial effects on the wind farm behaviors and therefore 20 different operation regimes are generated. The 10 different wind speed cases are generated by varying the inflow wind speed from 5 m/s to 16 m/s under the constant wind direction of 270° while the 10 different wind direction cases are generated by varying the inflow wind direction from 180° to 360° under the constant and rated wind speed of 11.5 m/s. The yaw angle control inputs are randomly generated within the range of -20° and 20° for the 20 different scenarios. The generated data are also contaminated with random noise to further

test the robustness of the RVM model. The whole dataset contains 1.6×10^5 data samples and is divided into training and testing subsets. The training subset has 1.2×10^5 data samples and the testing subset contains 0.8×10^5 data samples such that a proper division of the dataset for training and testing is retained to avoid overfitting.

Then, the RVM wind farm models were trained based on the above training sample dataset to represent control oriented wind farm models. They include an averaged power output model and an averaged thrust force model while the control inputs are wind turbine yaw angle settings of the four wind turbines. The regression functions of the RVM are used in these models and the radial basis function is used as the kernel function. For the wind farm reliability model in section 2.2, the nominal failure rate or baseline failure rate is set as 0.15.

The following metrics are used to measure the accuracy of the trained RVM models.

The mean absolute percentage error (MAPE)

$$\mathrm{MAPE}(y_i)=\frac{1}{M}\sum_{i=1}^{M}\left|\frac{y_i-\hat{y}_i}{y_i}\right|\times100\% \tag{28}$$

The root mean square error (RMSE)

$$\mathrm{RMSE}(y_i)=\left(\frac{1}{M}\sum_{i=1}^{M}\left|y_i-\hat{y}_i\right|^2\right)^{1/2} \tag{29}$$

where y_i is the real value, $\hat{y}_i$ is the predicted value and M is the total number of data samples.

The five typical evolutionary algorithms are implemented based on the RVM models in the proposed predictive control. The parameters of all the five algorithms are carefully tuned to achieve a tradeoff between control effectiveness and computational complexity.

The parameters of the five meta-heuristic algorithms for the specific wind farm control

problem can be tuned by using the design of experiments as a tool to define a parameter search space in which a reasonable wide range of the parameters to be tuned is first defined. Then, by evaluating the behaviors of the algorithms with respect to different settings of the parameters within the range, the parameter settings that lead to the better performances of the algorithms are selected and the range is narrowed accordingly. Based on the results and by conducting the design of experiments repeatedly, the suitable parameters can be identified. The interested readers can refer to [46] for more details about more fine tuning of the parameters.

The parameter settings of the algorithms are listed as follows:

(a) GDE3. The population size and maximum number of generations are all set as 200.

(b) MOEA/D. The population size and maximum number of generations are all set as 200. The normal boundary weights are generated by using the weight generator. The number of outer divisions is set as 12.

(c) MOPSO. The population size, the repository size and maximum number of generations are all set as 200. The inertia weight is set as 0.4. The individual confidence factor and the swarm confidence factor are both set as 2. The number of grids in each dimension is set as 20, the maximum velocity in percentage is set as 5 and the uniform mutation percentage is set as 0.5.

(d) MOGOA. The population size is set as 200, the maximum number of iterations is set as 200 and the maximum archive size is set as 100.

(e) NSGA-III. The number of division is set as 10, the maximum number of iterations is set as 200 and the population size is set as 200. The crossover and mutation percentages are both set as 0.5, and the mutation rate is set as 0.02.

5.2. *The RVM modelling performances*

The RVM modelling results of the wind farm power and thrust are shown in Figs. 3 and 4, respectively. As shown in Fig. 3, most of the predicted power points are located around the exact line (the dotted light green line) while only a small portion of the predictions have some deviations from the exact line. The MAPE and RMSE of the wind farm power modelling are calculated based on Eqs. (28) and (29) as 2.643% and 0.093 , which means that the relative accuracy of the RVM model is around 97.35% (100%-MAPE).

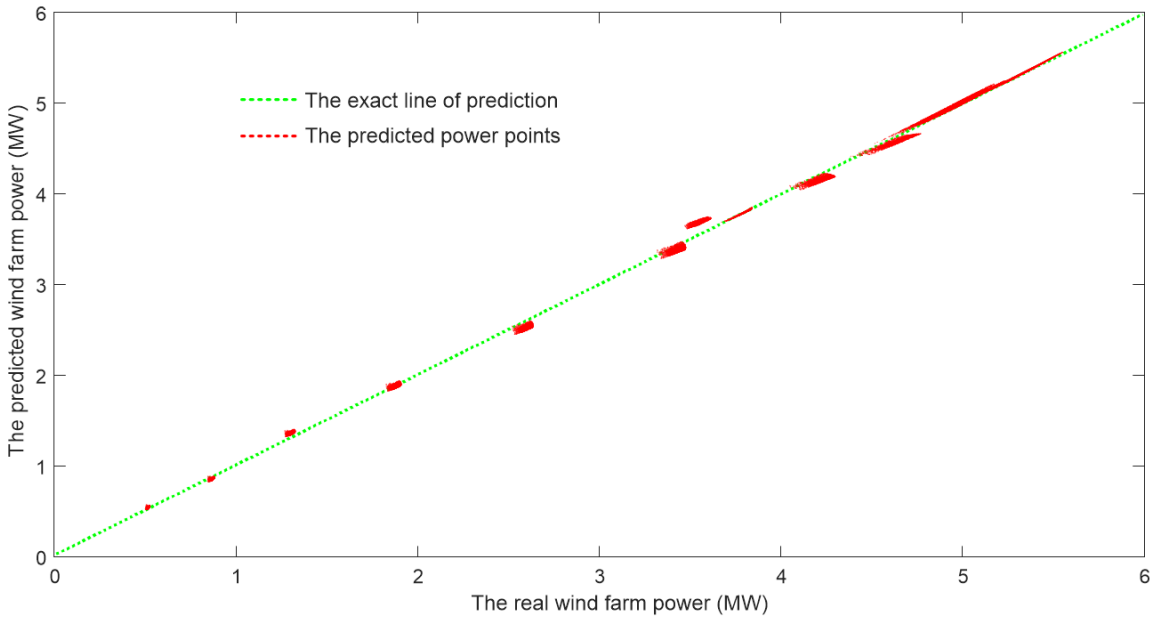

Fig. 3 The RVM modelling results of the averaged wind farm power

As illustrated in Fig. 4, the predicted wind farm thrust points from the RVM model are almost aligned with the exact line (the dashed light green line) while only very few points have some deviations from the exact line. The MAPE and RMSE of the RVM modelling of the wind farm thrust are 2.385% and 0.0134 respectively, which indicates that the RVM modelling accuracy is more than 97% and the trained RVM model can be readily used to construct the machine learning model for the wind farm.

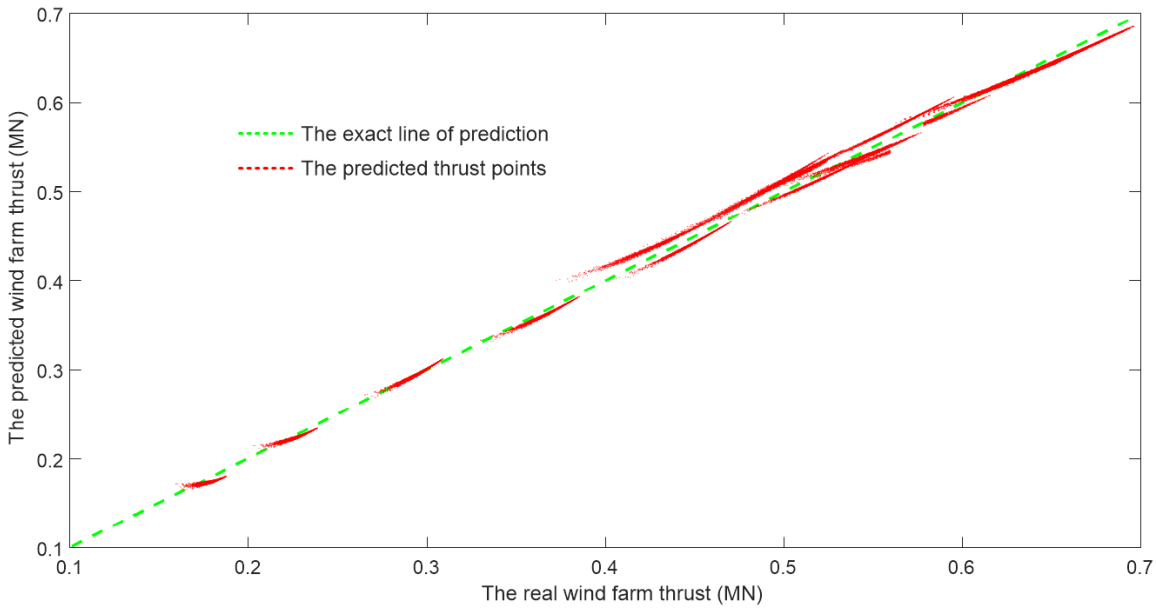

Fig. 4 The RVM modelling results of the averaged wind farm thrust

5.3. *The multi-objective predictive control performances*

Based on the RVM models for wind farm power and thrust predictions, the proposed multi-objective predictive control has been designed and implemented. The results of the proposed control are presented in this section in comparisons with a conventional predictive control method based on a single-objective DE algorithm which only maximizes the wind farm power generation.

5.3.1. *The control performances under different wind speeds*

The comparisons of the two control methods under varying wind speeds from 6 m/s to 15 m/s and the constant wind direction of 270° are shown in Figs. 5-7. As shown in Fig. 5, the wind farm power can be maintained at almost the same level with both control methods and the maximum reduction of 11% occurs at 15 m/s when using the proposed control method. It's also noted that the wind farm thrust load can be obviously reduced by using the proposed control approach compared to the conventional control method and the maximum reduction of 21.9% occurs at 13 m/s as shown in Fig. 6. The overall trend of the

wind farm reliability is improved by using the proposed control approach and the largest improvement of 1.6866% is achieved as shown in Fig. 7. Therefore, the proposed control approach has obvious advantages in maintaining relatively same level of wind farm power production, reducing thrust loads and improving wind farm reliability in comparison with the conventional control approach.

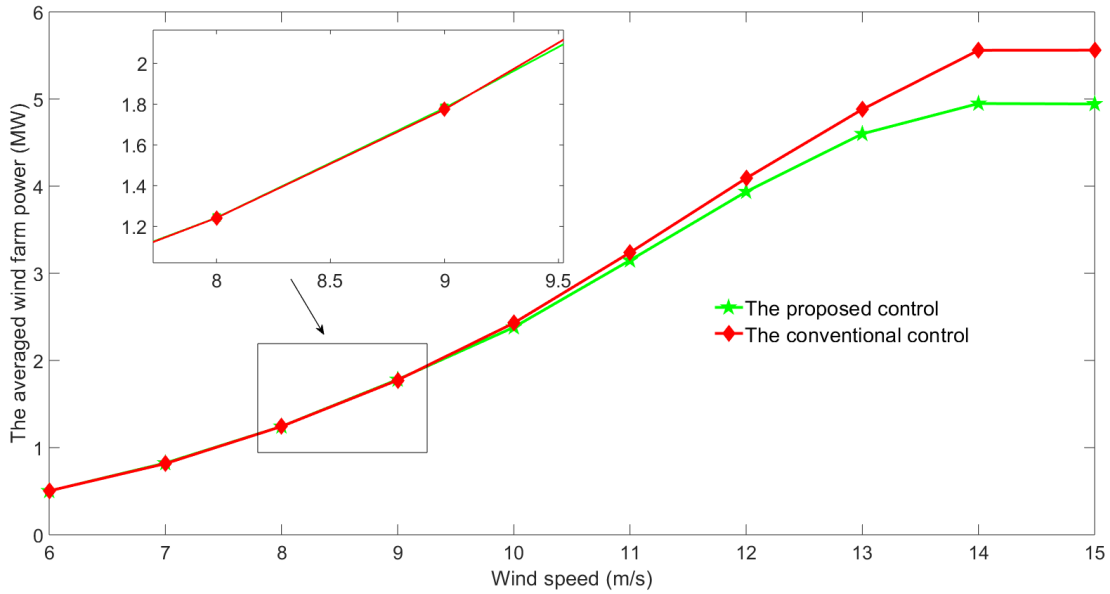

Fig. 5 The wind farm power generations under different wind speeds

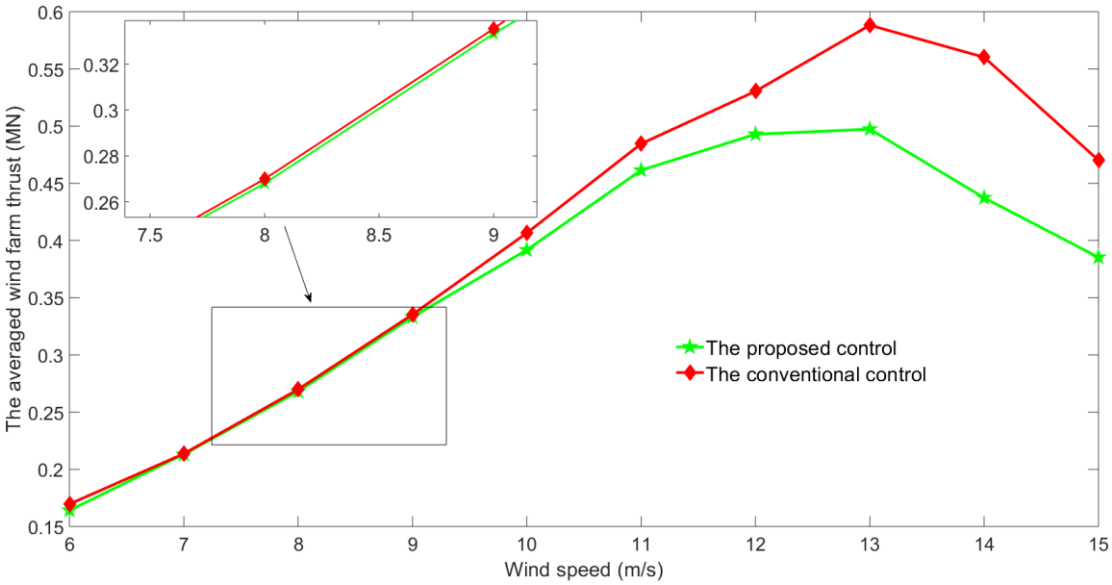

Fig. 6 The wind farm thrust generations under different wind speeds

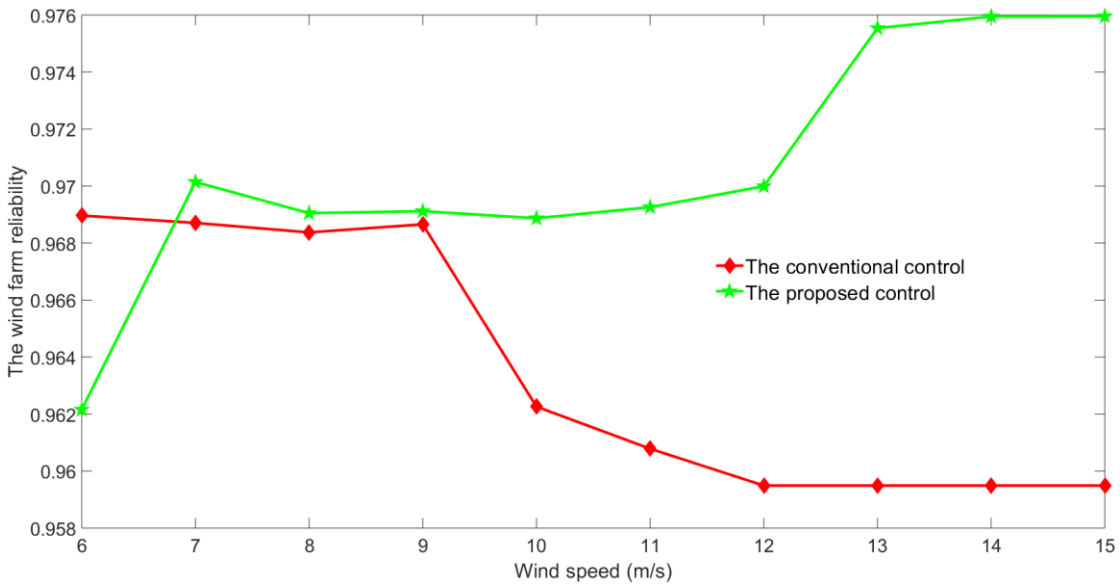

Fig. 7 The wind farm reliability under different wind speeds

As shown in Fig. 8, the yaw angle variations of the four turbines with the proposed control are generally more moderate than that with the conventional control. This result can be attributed to the third objective function in improving wind farm reliability by avoiding too aggressive yaw angle control inputs. The yaw angle control inputs are actually determined based on the trade-off between the three objectives in the proposed control whereas the conventional control determines the yaw angles only based on the wind farm power generation optimization.

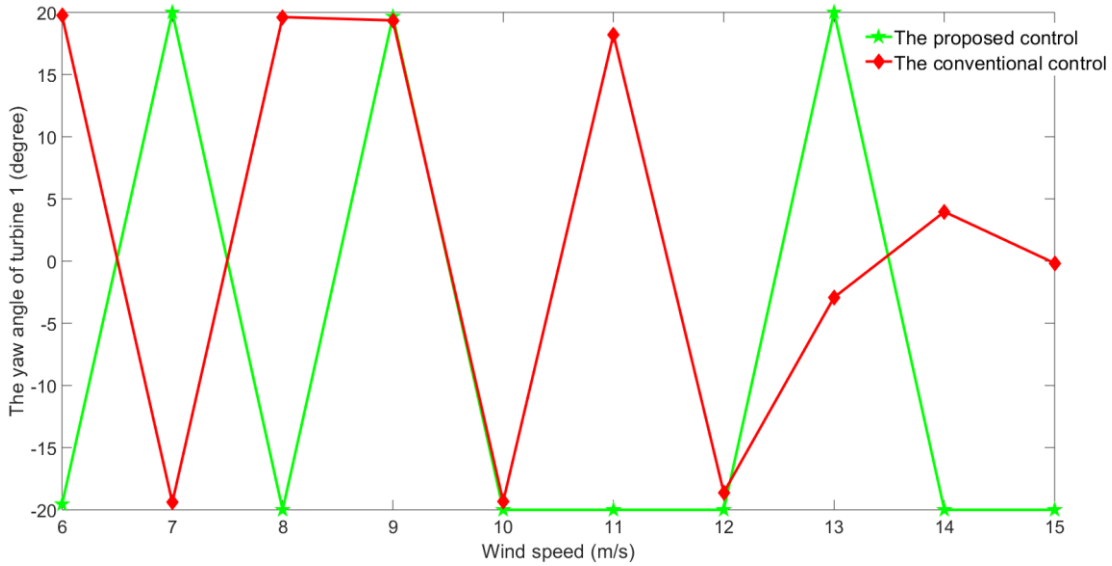

Fig. 8 (a) The yaw angle variations of the turbine 1 under different wind speeds

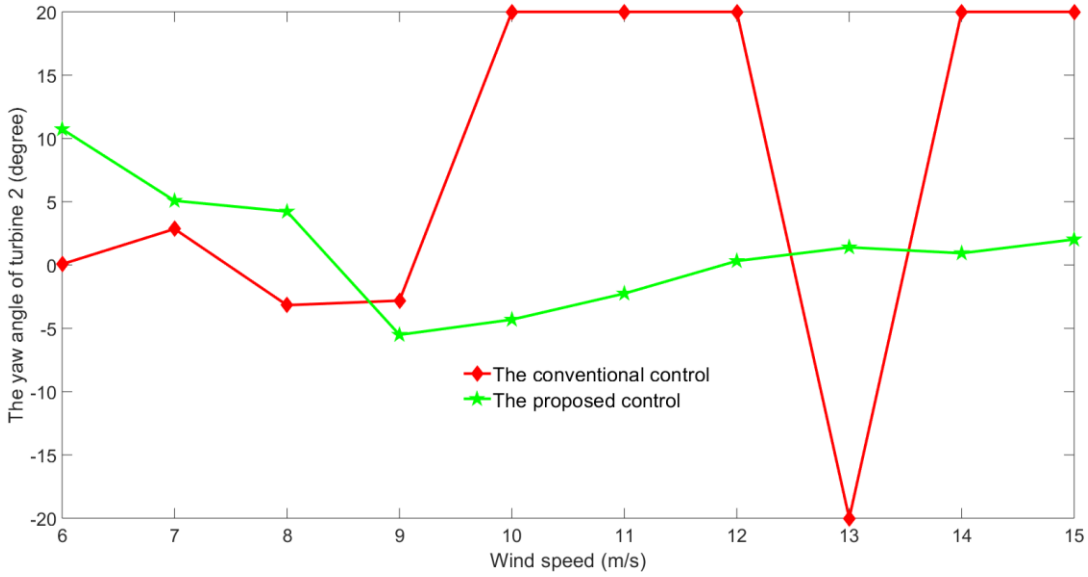

Fig. 8 (b) The yaw angle variations of the turbine 2 under different wind speeds

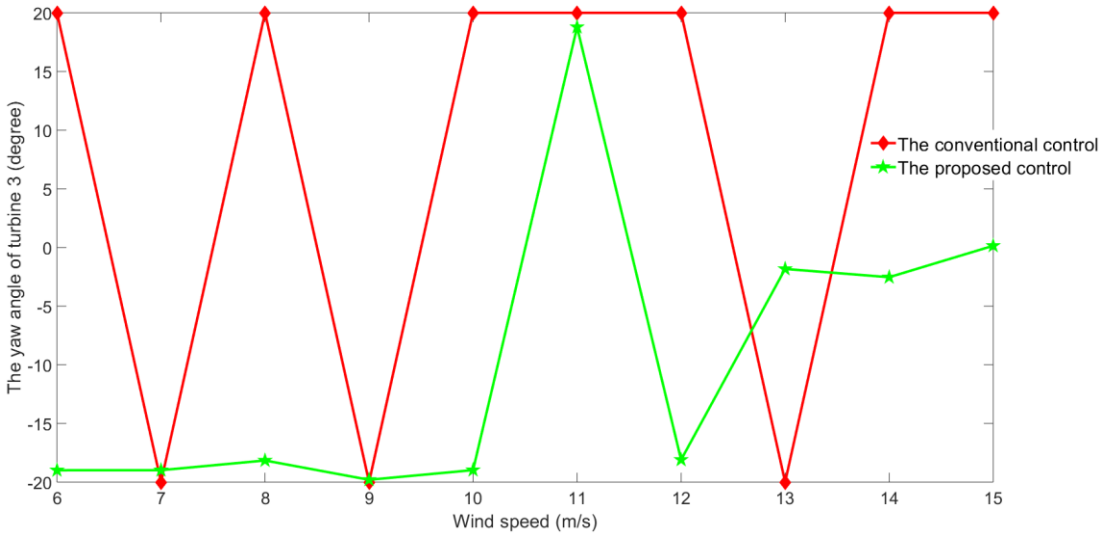

Fig. 8 (c) The yaw angle variations of the turbine 3 under different wind speeds

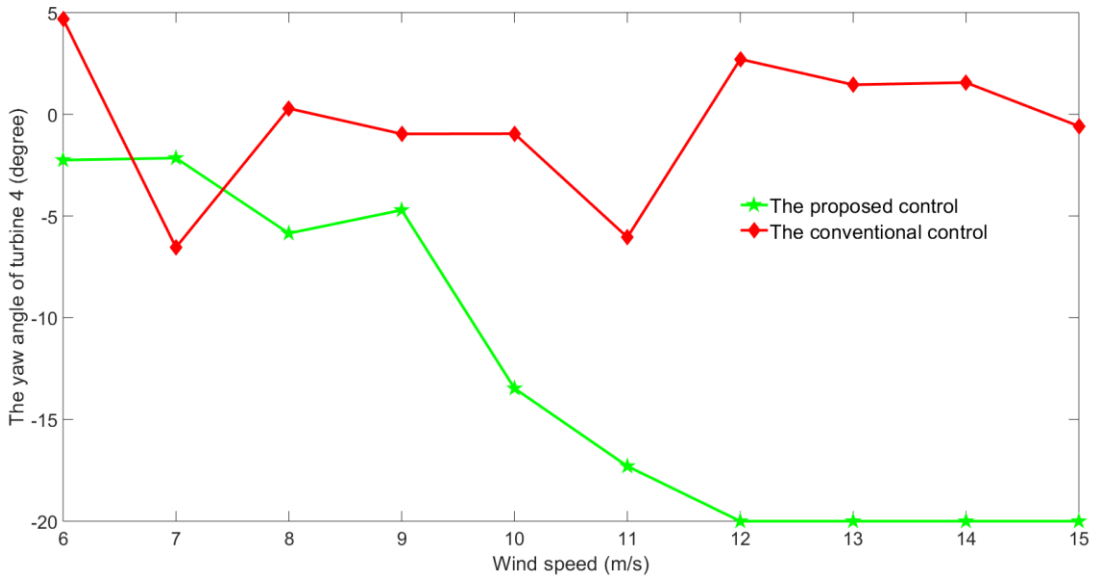

Fig. 8 (d) The yaw angle variations of the turbine 4 under different wind speeds

5.3.2. *The control performances under different wind directions*

Figs. 9 to 11 below show the results of the two control methods under the same wind speed of 11.5 m/s but different wind directions varying from 180° to 270°. As shown in Fig. 9, the wind farm power with the proposed control method can be kept close to that with the conventional control method, and the maximum reduction does not exceed 11% while the wind farm thrust load is clearly reduced by using the proposed method in

comparison with the conventional method and the maximum reduction is around 20%. The wind farm reliability is also clearly improved by around 1.7% and maintained around 98% by using the proposed control than that using the conventional control method. These results are highly consistent with those under the varying wind speed conditions and demonstrate that the proposed control method is able to provide very competitive results in terms of the trade-off between the wind power generation, reliability improvement, and mechanical load reduction, in comparison with the conventional control.

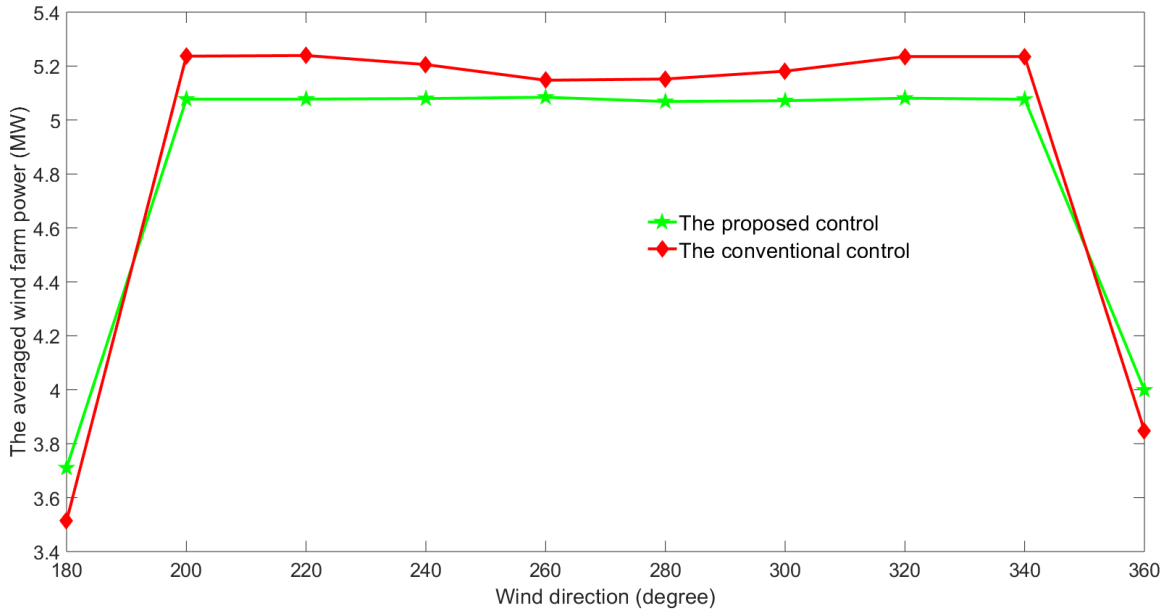

Fig. 9 The wind farm power generations under different wind directions

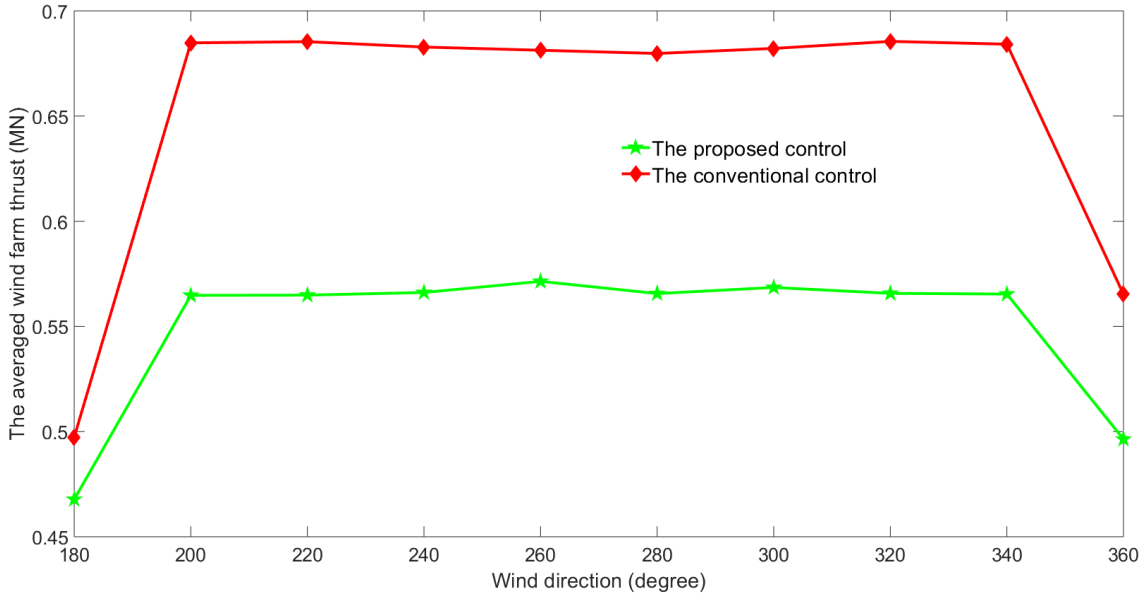

Fig. 10 The wind farm thrust generations under different wind directions

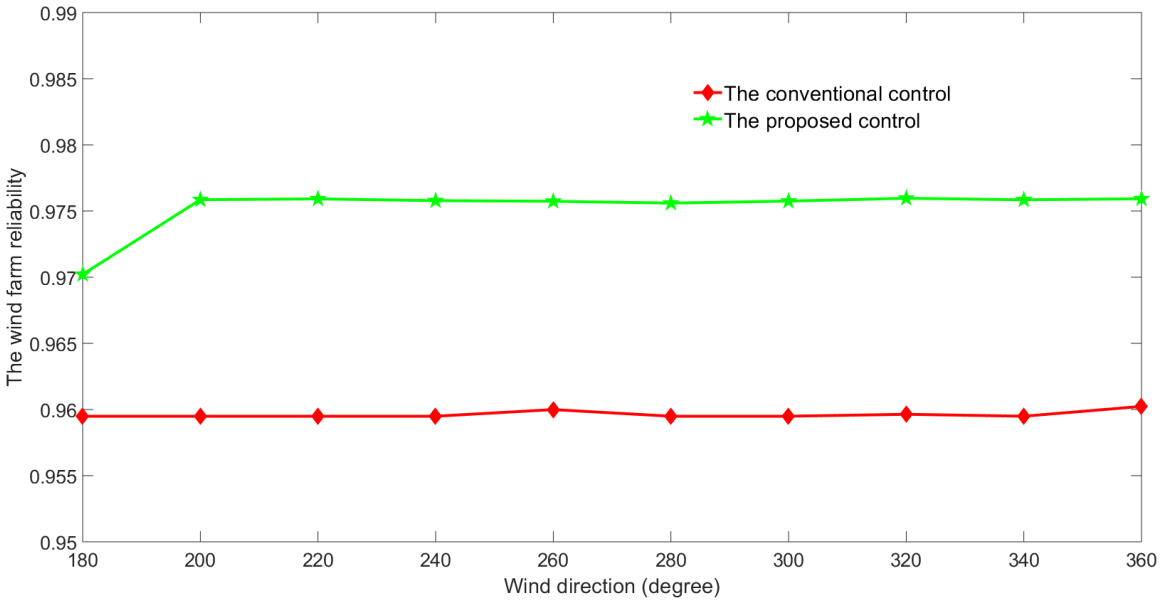

Fig. 11 The wind farm reliability under different wind directions

The comparisons of the yaw angle control inputs of the two control methods are presented in Fig. 12. As shown in this figure, the trend of the yaw angle inputs of the four turbines under varying wind directions also agree well with those under varying wind speeds in Fig. 8. The yaw angle inputs from the proposed control method vary more

smoothly than that from the conventional control method. The results are obtained by considering the third objective of improving wind farm reliability and reducing more aggressive yaw angle inputs since wind farm reliability will be highly deteriorated by the aggressive use of the yaw mechanisms.

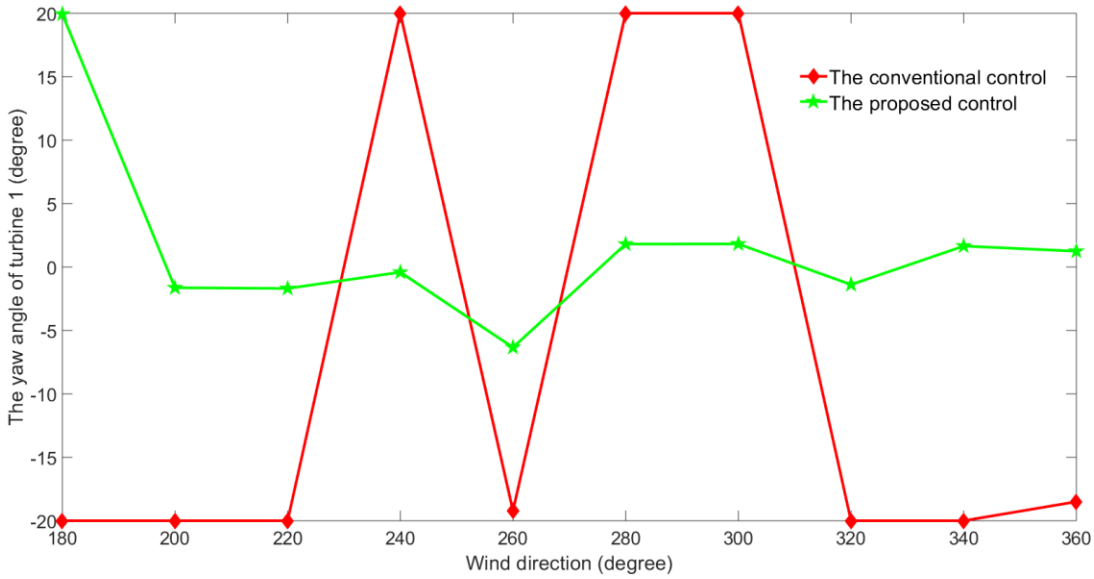

Fig. 12 (a) The yaw angle variations of the turbine 1 under different wind directions

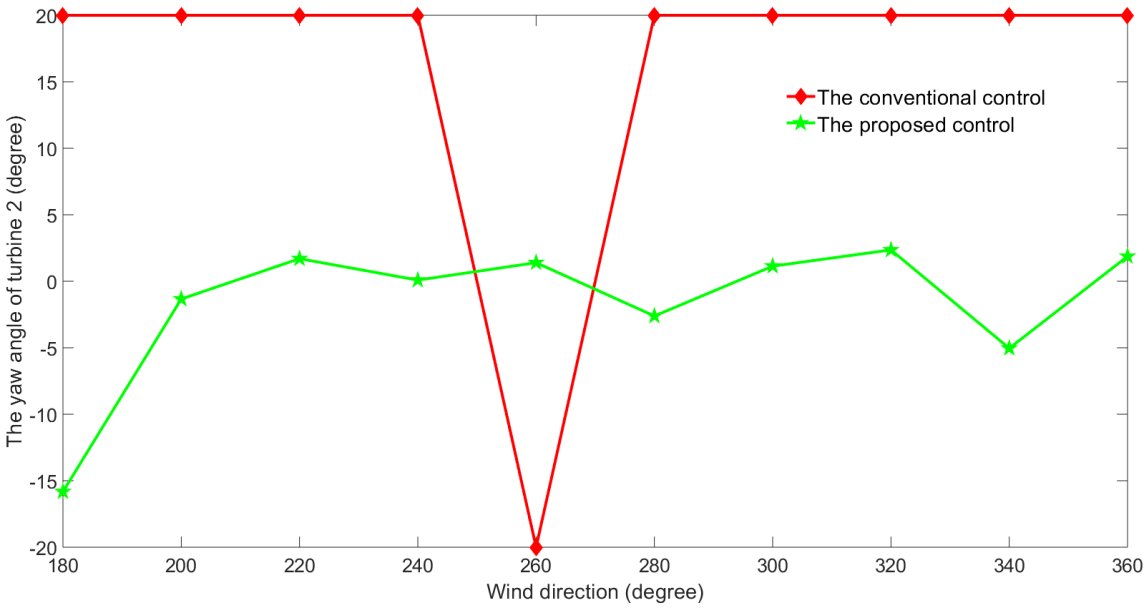

Fig. 12 (b) The yaw angle variations of the turbine 2 under different wind directions

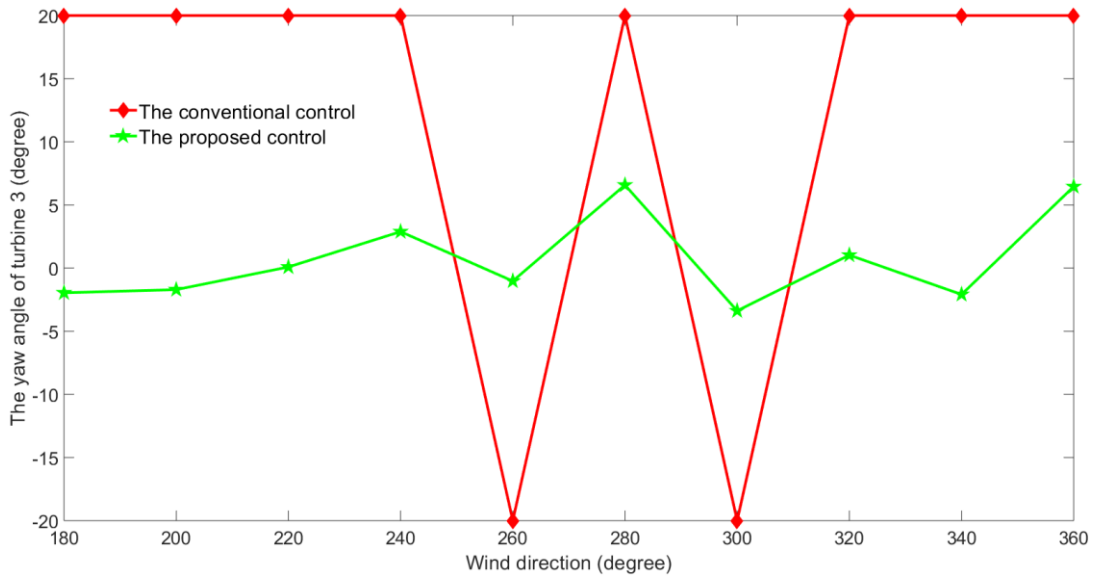

Fig. 12 (c) The yaw angle variations of the turbine 3 under different wind directions

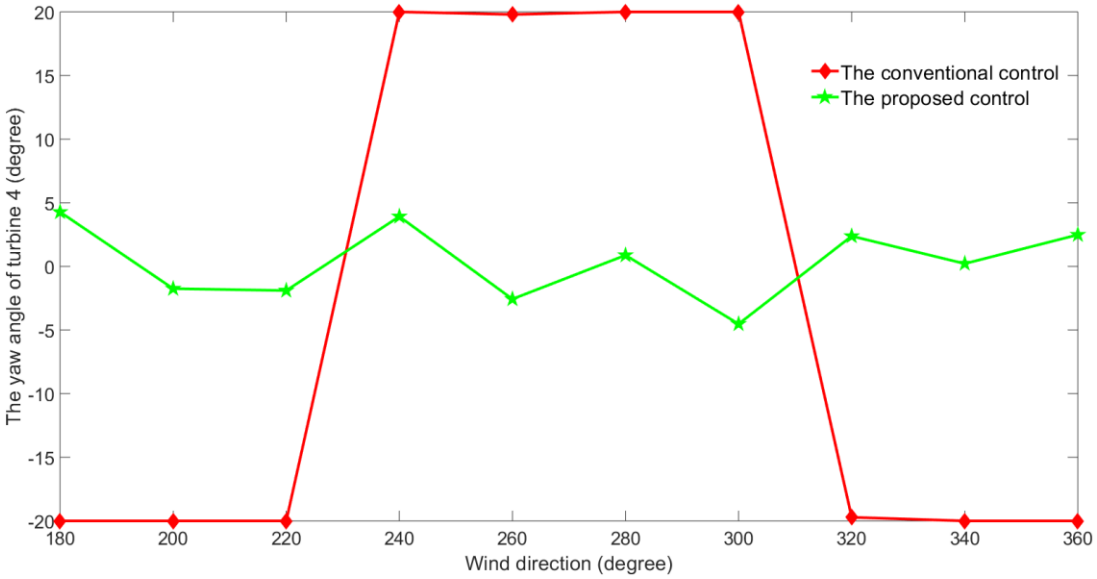

Fig. 12 (d) The yaw angle variations of the turbine 4 under different wind directions

5.3.3. *Performances of the five evolutionary algorithms*

The five evolutionary algorithms employed in this paper can all generate consistent distribution of the Pareto front solutions of the optimization problem in Eq. (26) (Fig. 15) and have no obvious differences in statistical performance measures. The main performance difference among them is the computational load that is the key in

determining the performance of the proposed control method.

The computational load or the used CPU times of the five evolutionary algorithms have been calculated at each time step. As shown in Figs. 13 and 14, GDE3 needs the longest CPU time of around 20s or more to achieve the proposed control and the NSGA-III needs around 5s to accomplish the optimization at one time step. On the other hand, the CPU times of MOEA/D and MOGOA are both around 1s while MOPSO can achieve the predictive control in less than 1s in general. Therefore, GDE3 has the largest computational load, and NSGA-III, MOEA/D and MOGOA have intermediate computational loads while MOPSO is much more computationally efficient. Considering that the sampling time interval of the wind farm control is around 1~2s in practice, it is not recommended to use GDE3 or NSGA-III in the predictive control while the MOPSO is readily applicable for the proposed predictive control.

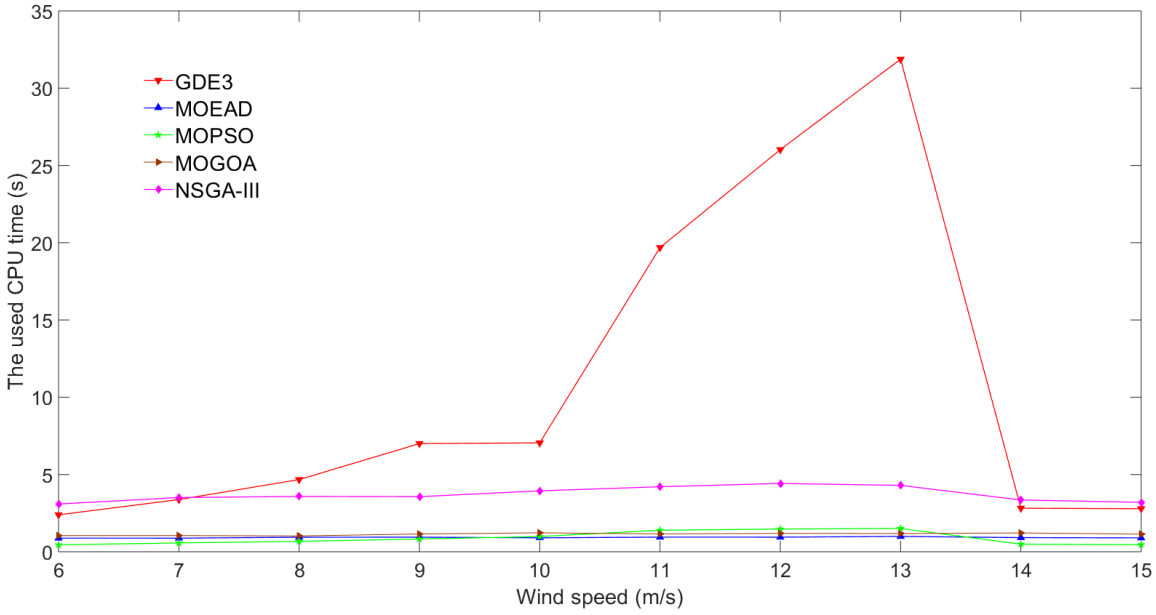

Fig. 13 The CPU times of the five algorithms under different wind speeds at one time step

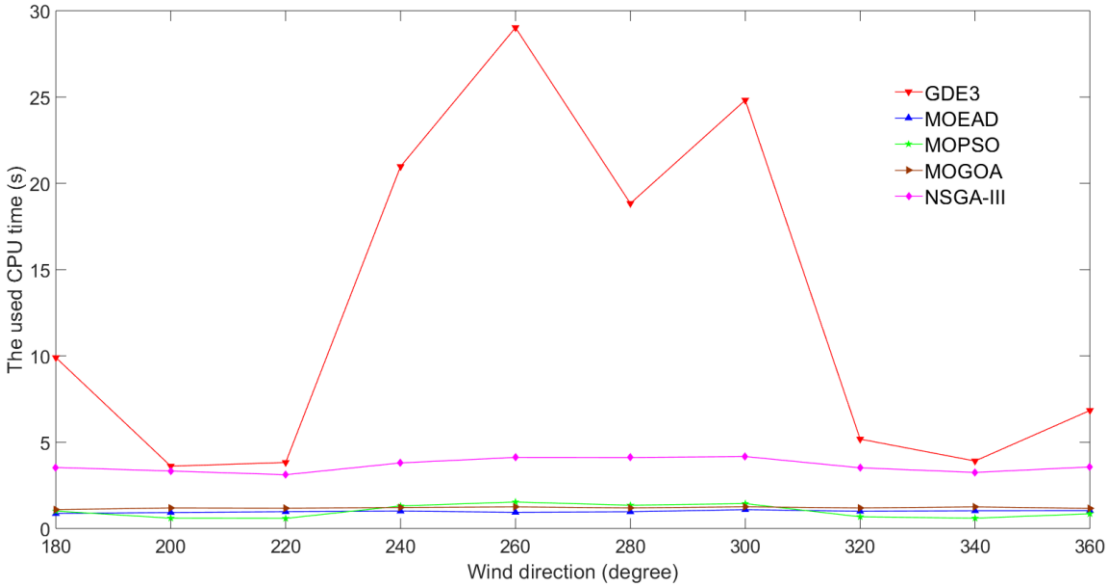

Fig. 14 The CPU times of the five algorithms under different wind directions at one time step

Fig. 15 shows the typical Pareto front of the five algorithms under the wind speed of 6 m/s and wind direction of 270°. As the figure shows, the five evolutionary algorithms can all generate consistent distribution of the Pareto front solutions of the optimization problem in Eq. (26). The solutions from GDE3, MOEA/D, MOPSO and NSGA-III distribute continuously and uniformly along the Pareto frontier while the solutions from MOGOA are scattered. MOPSO offers much better and continuous solutions than other algorithms. The optimal solution can be obtained from the Pareto front by using Eq. (27) and the optimal solutions of the five algorithms converge to the same point that is the selected solution of the control problem (see Fig. 15). Actually, the five evolutionary algorithms offer more options in choosing or regulating yaw angle settings for a wind farm for maximizing wind farm power production, improving farm reliability and simultaneously minimizing farm thrust loads than the conventional single-objective predictive control.

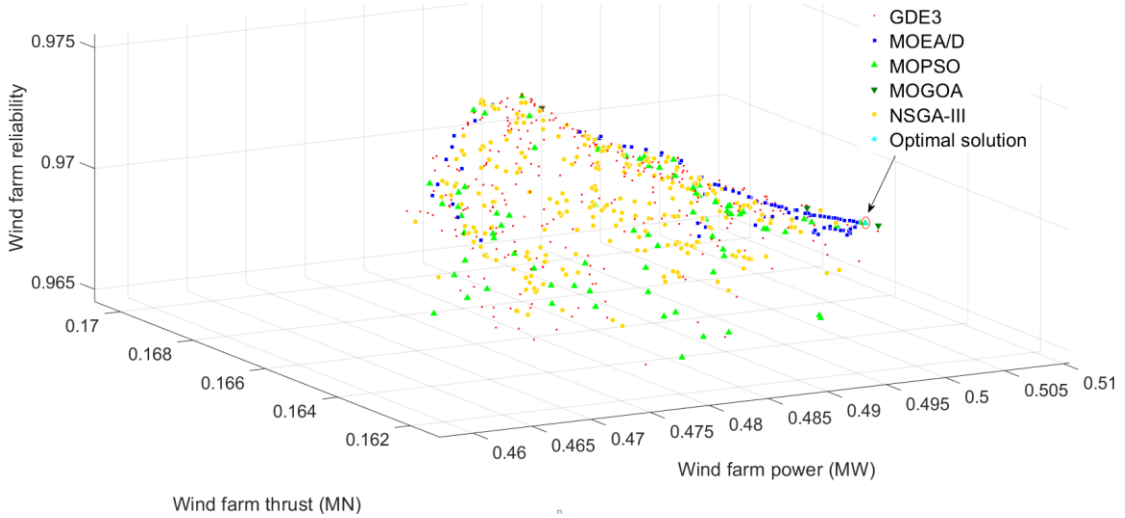

Fig. 15 The Pareto front of the five algorithms under the wind speed of 6 m/s and wind direction of 270°

Fig. 16 illustrates the evolutions of the three objectives during the computing step by using the MOPSO algorithm. As the figure shows, the objective functions 1 and 3 both increase steadily and converge to the relatively stable values during the computing steps. The objective function 2 also decreases steadily to a relatively stable value during the computing steps. The results indicate that the MOPSO algorithm is capable of finding the converged optimal solutions of Eq. (26) within small time interval of around 0.5 s such that the reliability aware multi-objective wind farm control task is readily fulfilled.

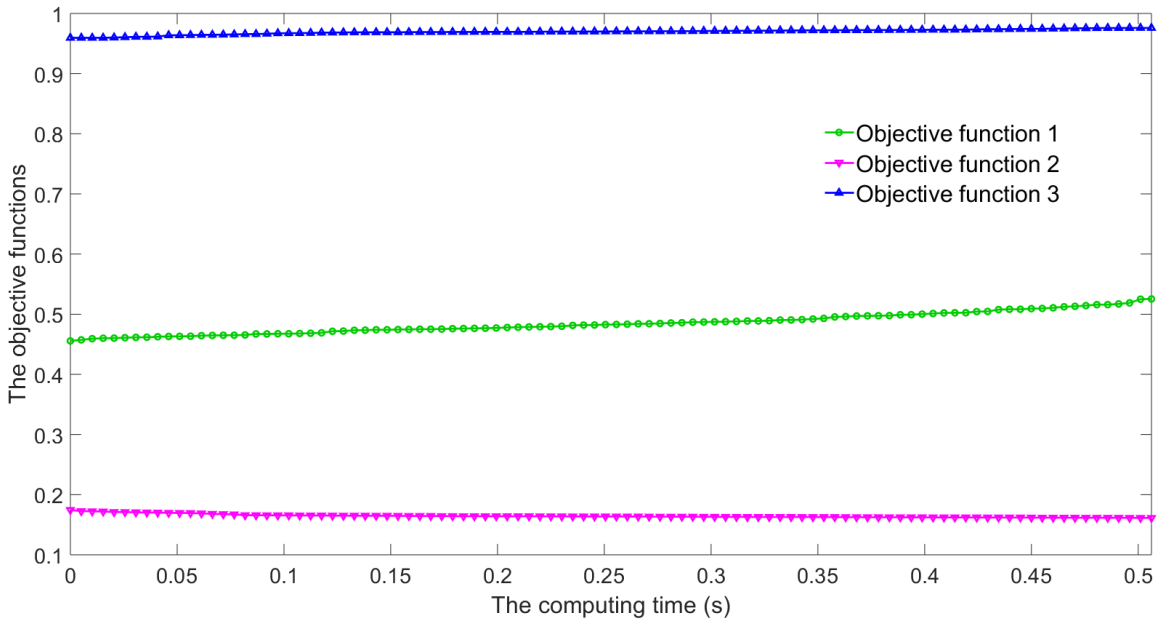

Fig. 16 The trends of the three objective functions in the proposed control under the wind speed of 6 m/s and wind direction of 270°

6. Conclusion

This paper has developed the reliability aware multi-objective predictive control approach for a wind farm based on machine learning and heuristic optimizations. The RVM wind farm model has been built based on large sample data from FLORIS, and the actuator health informed wind farm reliability model has also been constructed. Based on the trained RVM model, a multi-objective predictive control approach has been designed and implemented to maximize the wind farm power generation and reliability as well as simultaneously minimizing the wind farm thrust loads by determining the coordinated yaw angle control actions for each wind turbine. It allows a trade-off between all these three objectives. The five typical state of the art meta-heuristic evolutionary algorithms have been tested and compared to find the optimal control settings for each wind turbine within the wind farm. Extensive computational experiments have been conducted by using the FLORIS, and the proposed approaches have been validated by using different sets of large

data samples under different wind speeds and directions in comparison with a conventional predictive control algorithm. The validation results have demonstrated that the developed control is able to maintain the wind farm power close to that by using the conventional control method, while the wind farm thrust load is clearly reduced by using the proposed method in comparison with the conventional method (around 20%). The wind farm reliability index is also clearly improved by around 1.7% and maintained around 98% by using the proposed control than that from the conventional control method. In addition, the MOPSO algorithm used in the proposed approach is quite computationally efficient and has high potential in practical applications.

Acknowledgement

This work is supported by a joint UK-China Offshore Renewable Energy research scheme by the UK Engineering and Physical Sciences Research Council (grant number: EP/R007470/1) and the Natural Science Foundation of China (grant number: 5171101179).

References

[1] Tavner P. Offshore wind turbines: reliability. Availability and Maintenance. The Institution of Engineering and Technology, London, UK, 2012.

[2] Fischer K, Besnard F, Bertling L. Reliability-centered maintenance for wind turbines based on statistical analysis and practical experience. IEEE Transactions on energy conversion, 2011, 27(1): 184-195.

[3] Pérez J M P, Márquez F P G, Tobias A, et al. Wind turbine reliability analysis . Renewable and Sustainable Energy Reviews, 2013, 23: 463-472.

[4] Zhixin W, Chuanwen J, Qian A, et al. The key technology of offshore wind farm and its new development in China. Renewable and Sustainable energy reviews, 2009, 13(1): 216-222.

[5] Muyeen S M, Takahashi R, Tamura J. Operation and control of HVDC-connected offshore wind farm. IEEE Transactions on Sustainable Energy, 2010, 1(1): 30-37.

[6] Sagiroglu S, Sinanc D. Big data: A review. Collaboration Technologies and Systems (CTS), 2013 International Conference on. IEEE, 2013: 42-47.

[7] Meyers J, Munters W, Goit J. A framework for optimization of turbulent wind-farm boundary layers and application to optimal control of wind-farm energy extraction. American Control Conference (ACC), 2016. IEEE, 2016: 519-524.

[8] Marden J R, Ruben S D, Pao L Y. A model-free approach to wind farm control using game theoretic methods. IEEE Transactions on Control Systems Technology, 2013, 21(4): 1207-1214.

[9] J. Annoni, C. Bay, T. Taylor, L. Pao, P. Fleming and K. Johnson. Efficient Optimization of Large Wind Farms for Real-Time Control. 2018 Annual American Control Conference (ACC), Milwaukee, WI, 2018, pp. 6200-6205.

[10] Park J, Law K H. Cooperative wind turbine control for maximizing wind farm power using sequential convex programming. Energy Conversion and Management, 2015, 101: 295-316.

[11] Zhong S, Wang X. Decentralized Model-Free Wind Farm Control via Discrete Adaptive Filtering Methods. IEEE Transactions on Smart Grid, 2018, 9(4): 2529-2540.

[12] Gebraad P M O, Teeuwisse F W, Van Wingerden J W, et al. Wind plant power optimization through yaw control using a parametric model for wake effects—a CFD simulation study. Wind Energy, 2016, 19(1): 95-114.

[13] Quick J, Annoni J, King R, et al. Optimization under uncertainty for wake steering strategies. Journal of Physics: Conference Series. IOP Publishing, 2017, 854(1): 012036.

[14] Park J, Law K H. A data-driven, cooperative wind farm control to maximize the total power production. Applied Energy, 2016, 165: 151-165.

[15] Park J, Law K H. Bayesian Ascent: A Data-Driven Optimization Scheme for Real-Time Control with Application to Wind Farm Power Maximization. IEEE Trans. Contr. Sys. Techn., 2016, 24(5): 1655-1668.

[16] Fei S, He Y. Wind speed prediction using the hybrid model of wavelet decomposition and artificial bee colony algorithm-based relevance vector machine . International Journal of Electrical Power & Energy Systems, 2015, 73: 625-631.

[17] Bachour R, Maslova I, Ticlavilca A M, et al. Wavelet-multivariate relevance vector machine hybrid model for forecasting daily evapotranspiration. Stochastic environmental research and risk assessment, 2016, 30(1): 103-117.

[18] Liu Y, Zhang H, Yan J, et al. Hybrid relevance vector machine model for wind power forecasting. 2015.

[19] Bao Y, Wang H, Wang B. Short-term wind power prediction using differential EMD and relevance vector machine. Neural Computing and Applications, 2014, 25(2): 283-289.

[20] Chang Y, Fang H. A hybrid prognostic method for system degradation based on particle filter and relevance vector machine. Reliability Engineering & System Safety, 2019, 186: 51-63.

[21] Mérida J, Aguilar L T, Dávila J. Analysis and synthesis of sliding mode control for large scale variable speed wind turbine for power optimization. Renewable Energy, 2014, 71: 715-728.

[22] Salazar J C, Weber P, Nejjari F, et al. System reliability aware model predictive control framework. Reliability Engineering & System Safety, 2017, 167: 663-672.

[23] Khelassi A, Theilliol D, Weber P, et al. Fault-tolerant control design with respect to actuator health degradation: An LMI approach. 2011 IEEE International Conference on Control Applications (CCA). IEEE, 2011: 983-988.

[24] Khelassi A, Theilliol D, Weber P. Reconfigurability analysis for reliable fault-tolerant control design. International Journal of Applied Mathematics and Computer Science, 2011, 21(3): 431-439.

[25] Tipping M E. Sparse Bayesian learning and the relevance vector machine. Journal of machine learning research, 2001, 1(Jun): 211-244.

[26] Tipping M E, Faul A C. Fast marginal likelihood maximisation for sparse Bayesian models. AISTATS. 2003.

[27] Lin Y. Support vector machines and the Bayes rule in classification. Data Mining and Knowledge Discovery, 2002, 6(3): 259-275.

[28] Caesarendra W, Widodo A, Yang B S. Application of relevance vector machine and logistic regression for machine degradation assessment. Mechanical Systems and Signal Processing, 2010, 24(4): 1161-1171.

[29] Marler R T, Arora J S. Survey of multi-objective optimization methods for engineering. Structural and multidisciplinary optimization, 2004, 26(6): 369-395.

[30] Cui Y, Geng Z, Zhu Q, et al. Multi-objective optimization methods and application in energy saving. Energy, 2017, 125: 681-704.

[31] Kukkonen S, Lampinen J. GDE3: The third evolution step of generalized differential evolution. 2005 IEEE congress on evolutionary computation. IEEE, 2005, 1: 443-450.

[32] Zhang Q, Li H. MOEA/D: A multiobjective evolutionary algorithm based on decomposition. IEEE Transactions on evolutionary computation, 2007, 11(6): 712-731.

[33] Coello C A C, Pulido G T, Lechuga M S. Handling multiple objectives with particle swarm optimization. IEEE Transactions on evolutionary computation, 2004, 8(3): 256-279.

[34] C. A. Coello Coello and M. S. Lechuga, MOPSO: A proposal for multiple objective particle swarm optimization, in Proc. Congr. Evolutionary Computation (CEC'2002), Honolulu, HI, May 2002, 1, 1051–1056.

[35] Mirjalili S Z, Mirjalili S, Saremi S, et al. Grasshopper optimization algorithm for multi-objective optimization problems. Applied Intelligence, 2018, 48(4): 805-820.

[36] Topaz C M, Bernoff A J, Logan S, et al. A model for rolling swarms of locusts. The European Physical Journal Special Topics, 2008, 157(1): 93-109.

[37] Saremi S, Mirjalili S, Lewis A. Grasshopper optimisation algorithm: theory and application. Advances in Engineering Software, 2017, 105: 30-47.

[38] Deb K, Padhye N. Enhancing performance of particle swarm optimization through an algorithmic link with genetic algorithms. Computational Optimization and Applications, 2014, 57(3): 761-794.

[39] Padhye N, Bhardawaj P, Deb K. Improving differential evolution through a unified approach. Journal of Global Optimization, 2013, 55(4): 771-799.

[40] Deb K, Jain H. An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: Solving problems with box constraints. IEEE Trans. Evolutionary Computation, 2014, 18(4): 577-601.

[41] Jain H, Deb K. An Evolutionary Many-Objective Optimization Algorithm Using Reference-Point Based Nondominated Sorting Approach, Part II: Handling Constraints and Extending to an Adaptive Approach. IEEE Trans. Evolutionary Computation, 2014, 18(4): 602-622.

[42] Deb K, Pratap A, Agarwal S, et al. A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE transactions on evolutionary computation, 2002, 6(2): 182-197.

[43] Deb K, Agrawal S, Pratap A, et al. A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. International Conference on Parallel Problem Solving From Nature. Springer, Berlin, Heidelberg, 2000: 849-858.

[44] Dilip, D. and Porté-Agel, F.: Wind Turbine Wake Mitigation through Blade Pitch Offset, Energies, 10, 757, 2017.

[45] Abkar, M. and Porté-Agel, F.: Influence of atmospheric stability on wind-turbine wakes: A large-eddy simulation study, Physics of Fluids, 27, 035 104, 2015.

[46] Barbosa E B M, Senne E L F. Improving the fine-tuning of metaheuristics: an approach combining design of experiments and racing algorithms. Journal of Optimization, 2017, 2017.

[47] Got A, Moussaoui A, Zouache D. A guided population archive whale optimization algorithm for solving multiobjective optimization problems. Expert Systems with Applications, 2020, 141: 112972.

[48] Liang J, Xu W, Yue C, et al. Multimodal multiobjective optimization with differential evolution. Swarm and evolutionary computation, 2019, 44: 1028-1059.

[49] Liu T, Jiao L, Ma W, et al. A new quantum-behaved particle swarm optimization based on cultural evolution mechanism for multiobjective problems. Knowledge-Based Systems, 2016, 101: 90-99.

[50] Ayala H V H, Keller P, de Fátima Morais M, et al. Design of heat exchangers using a novel multiobjective free search differential evolution paradigm. Applied Thermal Engineering, 2016, 94: 170-177.

[51] Zhang Z. Fast multiobjective immune optimization approach solving multiobjective interval number programming. Swarm and Evolutionary Computation, 2019, 51: 100578.

[52] Bora T C, Mariani V C, dos Santos Coelho L. Multi-objective optimization of the environmental-economic dispatch with reinforcement learning based on non-dominated sorting genetic algorithm. Applied Thermal Engineering, 2019, 146: 688-700.

[53] Lacerda A S M, Batista L S. KDT-MOEA: A multiobjective optimization framework based on KD trees. Information Sciences, 2019, 503: 200-218.

[54] Neto J X V, Junior E J G, Moreno S R, et al. Wind turbine blade geometry design based on multi-objective optimization using metaheuristics. Energy, 2018, 162: 645-658.

[55] Cao L, Xu L, Goodman E D, et al. Decomposition-based evolutionary dynamic multiobjective optimization using a difference model. Applied Soft Computing, 2019, 76: 473-490.

[56] Mirjalili S, Saremi S, Mirjalili S M, et al. Multi-objective grey wolf optimizer: a novel algorithm for multi-criterion optimization. Expert Systems with Applications, 2016, 47: 106-119.

[57] Zhang H, Sun J, Liu T, et al. Balancing exploration and exploitation in multiobjective evolutionary optimization. Information Sciences, 2019, 497: 129-148.

[58] Liu X, Du Y, Jiang M, et al. Multiobjective Particle Swarm Optimization Based on Network Embedding for Complex Network Community Detection. IEEE Transactions on Computational Social Systems, 2020.

[59] Rambabu R, Vadakkepat P, Tan K C, et al. A Mixture-of-Experts Prediction Framework for Evolutionary Dynamic Multiobjective Optimization. IEEE transactions on cybernetics, 2019.

[60] Gaudrie D, Le Riche R, Picheny V, et al. Targeting solutions in Bayesian multi-objective optimization: sequential and batch versions. Annals of Mathematics and Artificial Intelligence, 2020, 88(1): 187-212.

[61] Barba-González C, Nebro A J, García-Nieto J, et al. A multi-objective interactive dynamic particle swarm optimizer. Progress in Artificial Intelligence, 2019: 1-11.