Model-Guided Learning for Wind Farm Power Optimization

Zhiwei Xu, *Graduate Student Member, IEEE*, Bing Chu, Hua Geng, *Fellow, IEEE*, and Xiaohong Nian

***Abstract*— In a wind farm, the interactions between turbines caused by wakes can significantly reduce the power output of the wind farm. Accurately modelling the interactions is challenging due to the highly complex nature of the wakes and this limits the performance of model-based wind farm power optimization methods. There are also data-driven approaches, which do not require a system model. However, they generally require a large number of measurement data and the convergence speed can be slow. To address these limitations, this paper proposes a model-guided learning method for wind farm to improve its power output by leveraging the knowledge of the available simplified power generation model and learning from the real-time power generation data. The proposed method can quickly increase the power output of the wind farm, guarantee implemented control actions to satisfy the control constraints of all turbines, and have the ability to find the optimal solution of the power optimization problem. The presented method is then extended to deal with time-varying wind conditions using a hierarchical framework. Simulation results indicate that the proposed scheme can efficiently improve the power output of the wind farm in different wind conditions compared with some benchmarks. It shows a power efficiency gain of 2.4% over greedy policy and 1.0% than model-based gradient method in given complex wind conditions, which are substantial improvements in the performance for the considered wind farm power optimization problem.**

***Index Terms*— Wind farm, wake interactions, power optimization, model uncertainties, cooperative control.**

I. Introduction

RENEWABLE energy plays a vital role in mitigating the climate change, environmental pollution and increasing electricity demands. Wind energy is one of the most environmental friendly and cost-competitive renewable energy sources, and its utilization is fast-growing in an unprecedented rapid pace [1]. In 2019, there was 60.4GW global new wind power installation—an increase of 19% compared to that of 2018. The total capacity has risen up to 651GW with a growth of 10% compared to 2018 [2]. Wind power has met 15% of the EU's power demand on average and would meet more than 50% by 2050 as currently planned [3]. With this promising trend, advanced wind farm control strategies are becoming increasingly important to efficiently utilize the wind energy, increase power extraction and develop more profitable wind farms [4], [5].

In the wind farm, the wind turbines are often placed together to reduce installation, operation and maintenance costs [6]. This leads to the problem that the wakes generated by upstream wind turbines may not fully recover before arriving at downstream wind turbines. Then the power output of the downwind turbines is likely to be significantly degraded due to reduced wind speeds inside the wakes [7], which results in lower power output of the whole wind farm. In practice, greedy policy is widely applied, where each turbine aims to maximize its own power output. However, it neglects the wake interactions among the turbines and thus often leads to suboptimal wind farm power output [8]–[10]. Experimental results indicate that using greedy policy total wind farm power loss can surpass 30% under some worst case scenarios [11]. Therefore, an important research topic on wind farm is about how to mitigate the effect of wake interactions among the turbines to maximize the power output of wind farm through cooperative control between turbines.

Various control strategies for wind farm have been proposed, mainly including model-based methods, data-driven methods and hybrid methods. Most model-based methods optimize the power output of wind farm by utilizing the analytical power generation models of the wind farm via dynamic programming approach [12], steepest descent method [13], particle swarm optimization algorithm [14], *et al*. These methods have fast convergence speed due to the availability of analytical power generation models. However, their merits on improving the power output of wind farm can be limited as obtaining accurate models to be used in optimization reflecting the actual aerodynamics of the wakes can be difficult due to their very complex nature. Additionally, the optimization methods based on Computational Fluid Dynamics (CFD) models are presented, e.g. conjugate gradient method [15] with large eddy simulation [16]. However, the use of CFD simulations requires significant computational cost that is usually not satisfied in practice even if it improves the model accuracy [17].

Data-driven methods have also received great attentions, which aim to maximize the power output of wind farm without models and only use power generation data from wind farm. These methods include safe experimentation dynamics (SED) [8], gradient-based method [18], discrete adaptive filtering

This work was supported by National Natural Science Foundation of China (NSFC) under Grant U2166601, Grant U2066602, Grant 52061635102 and Grant 62173347. *(Corresponding author: Hua Geng.)*

Z. Xu and H. Geng are with the Department of Automation, Tsinghua University, and Beijing National Research Center for Information Science and Technology, Beijing 100084, China (e-mail: xuzw18@mails.tsinghua.edu.cn; genghua@tsinghua.edu.cn).

B. Chu is with the School of Electronics and Computer Science, University of Southampton, Southampton SO171BJ, U.K. (e-mail: B.Chu@soton.ac.uk).

X. Nian is with the School of Automation, Central South University, Changsha 410075, China (e-mail: xhnian@csu.edu.cn).

algorithms [19], game theoretic approaches [20], stochastic projected simplex method [21], *et al*. Although the data-driven methods do not use the power generation model of wind farm, they usually achieve an optimum at the cost of a large number of measurement data and thus the convergence speed is slow.

To address the above problems, some hybrid methods have been recently proposed, solving the wind farm power optimization problem using the inaccurate power generation models and real-time measurement data. These methods mainly include two categories. The key point of one category is to calibrate the inaccurate wind farm power generation models using measurement data and then determine control action based on the calibrated models. In [22]–[24], the parameters of the inaccurate models are identified using measurement data to improve model accuracy and wind farm power performance. This often requires that the structures of the inaccurate models are correct, which however cannot be satisfied easily due to highly complex nature of the wakes. In [25], a modifier adaptation approach is proposed, in which model mismatch is identified by Gaussian process regression and measurements. Although this can improve the inaccurate model and the wind farm power generation performance, but it is not trivial to analyze the convergence of the approach. The key idea of another category is to utilize the inaccurate models to assist the optimization processes of data-driven methods, especially in initial optimization stage. In particular, a novel knowledge-assisted reinforcement learning method is proposed by combining the analytical model with reinforcement learning framework to maximize the power output of wind farm [26]. Note that the convergence analysis of the above method is also challenging.

The contributions of this paper are summarized as follows: (a) A novel model-guided learning (MGL) method is developed for wind farm power optimization problem; (b) The convergence properties of the proposed method are rigorously analyzed; (c) The presented method is further extended to handle time-varying wind conditions using a hierarchical power optimization framework; (d) The effectiveness of the proposed method is evaluated through realistic simulation study using real wind data. The presented method belongs to the second category of the above hybrid methods. It can rapidly improve the power output of wind farm, benefiting from the available albeit simplified analytical power generation models—the models can reflect some key features of actual wind farm and thus can generally provide a satisfactory search direction for the developed method in the early optimization stage. Meanwhile, the proposed method has the ability to find the optimal solution of the power optimization problem by learning from real-time power generation data and thus can compensate for the effect of the model uncertainties on the wind farm power performance. Precisely because of above merits, the presented method can efficiently improve the power output of wind farm.

The rest of this paper is organized as follows. In Section II, the power generation model of wind farm and its power optimization problem are formulated. In Section III, a model-guided learning method is developed for wind farm power optimization under fixed wind direction and its convergence properties are analyzed. A hierarchical wind farm power optimization scheme is then developed using the proposed method to deal with time-varying wind direction in Section IV. Section V presents numerical results using real wind farm data to verify the performance of the proposed scheme. Finally, Section VI gives the conclusion and possible directions for future research.

II. Wind Farm Power Optimization Problem

In this section, the power generation model of wind farm is introduced and then the wind farm power optimization problem is described.

A. Power Model

A wind farm consisting of n wind turbines is considered. Let $N = \{1, 2, \cdots, n\}$ denote the set of all turbines. For simplicity, the blade disk planes of all turbines are supposed to be perpendicular to wind direction. The control action of turbine $i \in N$ is selected as its axial induction factor (AIF) u_i. The AIF denotes the wind velocity reduction over rotor plane and can be adjusted by the blade pitch and generator torque (standard inputs) of turbine. The feasible domain of the u_i is denoted by the set $\mathcal{U}_i = \{u_i \,|u_{i,min} \le u_i \le u_{i,max}\}$. The $u_{i,min}$ and $u_{i,max}$ denote the lower bound and upper bound of the u_i, respectively. The joint AIF of all turbines is expressed by the tuple $u = (u_1, \cdots, u_n)$, whose admissible set is represented by $\mathcal{U} = \mathcal{U}_1 \times \cdots \times \mathcal{U}_n$ and $\times$ is the Cartesian product.

The aggregate wind velocity $V_i(\{u_j\}_{j\in\mathcal{N}_i}, V_\infty, \theta)$ at an arbitrary turbine i can be expressed as follows:

$$V_i(\{u_j\}_{j\in\mathcal{N}_i}, V_\infty, \theta) = V_\infty \left(1 - \delta V_i(\{u_j\}_{j\in\mathcal{N}_i}, \theta)\right), \quad (1)$$

where $\mathcal{N}_i$ denotes the set of upstream turbines that are coupled with turbine i by wakes, V_∞ is free-stream wind speed, θ represents wind direction, $\delta V_i(\{u_j\}_{j\in\mathcal{N}_i}, \theta)$ is the wind speed deficit at turbine i quantifying the reduction of the wind speed in the wakes. The power generated by turbine i can be modeled as

$$P_i(u_i; \{u_j\}_{j\in\mathcal{N}_i}, V_\infty, \theta) = K_{p,i}(u_i) V_i(\{u_j\}_{j\in\mathcal{N}_i}, V_\infty, \theta)^3, \quad (2)$$

where $K_{p,i}(u_i) = (1/2)\rho A_i C_{p,i}(u_i)$, ρ is the air density, A_i is the disk area generated by the blade of turbine i, and $C_{p,i}(u_i)$ is the power coefficient defined as

$$C_{p,i}(u_i) = 4u_i(1 - u_i)^2. \quad (3)$$

The total power output of the wind farm is defined as the sum of the power generated by all turbines:

$$P(u; V_\infty, \theta) = \sum_{i=1}^{n} P_i(u_i; \{u_j\}_{j\in\mathcal{N}_i}, V_\infty, \theta). \quad (4)$$

The goal of the wake interaction modelling is to identify the wind speed deficit $\delta V_i(\{u_j\}_{j\in\mathcal{N}_i}, \theta)$ in (1). However, accurately modelling it is extremely challenging due to the highly complex characteristics of wakes, e.g. deflection and dependence on environment parameters [9]. In this paper, the Park model [27], a simplified yet very popular wake model, is used to approximately describe the wake interactions among

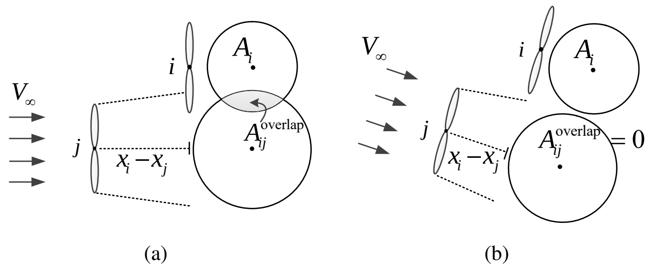


Fig. 1. Two-turbine wake interaction examples.

the turbines. According to the Park model, the wind speed deficit can be described as follows:

$$\delta V_i(\{u_j\}_{j\in\mathcal{N}_i}, \theta) = 2\sqrt{\sum_{j\in\mathcal{N}_i}\left(u_j X_{ij}^2 \frac{A_{ij}^{\text{overlap}}(\theta)}{A_i}\right)^2}, \quad (5)$$

where

$$X_{ij} = \frac{D_j}{D_j + 2\kappa\left(x_i(\theta) - x_j(\theta)\right)},$$

D_j is the diameter of the blade rotation disk of turbine j, κ is the roughness coefficient that defines the slope of wake expansion, $x_i(\theta)$ and $x_j(\theta)$ denote respectively the distances of turbine i and j from a common vertex along wind direction θ, $A_{ij}^{\text{overlap}}(\theta)$ is the part area of the A_i that overlaps with the wake created by turbine j and is related with the θ. To illustrate this, consider the wake interaction examples given in Fig. 1, where the direction that the arrow points to represents the wind direction and between top and bottom dotted lines shows the wake area of turbine j. From Fig. 1(a) to Fig. 1(b), it can be observed that the turbine distance $x_i - x_j$ and the A_{ij}^{overlap} vary with changes of the wind direction, which results in different wake interaction patterns among the turbines.

As mentioned earlier, it is very challenging to obtain an accurate description of the wind farm power generation model $P(u; V_\infty, \theta)$ in (4), largely due to the difficulties in modeling the wake interactions. Only an approximate nominal power generation model $\bar{P}(u; V_\infty, \theta)$ can be obtained (using e.g. the wake interaction model (5) based on the Park model). We describe the relationship as follows:

$$P(u; V_\infty, \theta) = \bar{P}(u; V_\infty, \theta) + \Delta P(u; V_\infty, \theta), \quad (6)$$

where $\Delta P(u; V_\infty, \theta)$ represents the integration of the model uncertainties or mismatches related to both inaccurate parameters and unmodeled characteristics. For the nominal model $\bar{P}(u; V_\infty, \theta)$ based on the Park model, the roughness coefficient κ is one of the inaccurate parameters and wake deflection is one of the unmodeled characteristics.

Remark 1: In this paper, the wind speed deficit $\delta V_i(\{u_j\}_{j\in\mathcal{N}_i}, \theta)$ in (1) is approximated by (5) using the Park model. The obtained results in this paper still hold for other wake interaction models.

B. Problem Formulation

The wind farm power optimization problem can be formulated as finding the optimal joint AIF to maximize the power output of the wind farm, i.e.,

$$u_{opt} = \arg\max_{u\in\mathcal{U}} P(u; V_\infty, \theta). \quad (7)$$

It is worth mentioning that traditionally, most model-based methods are designed based on the nominal model $\bar{P}(u; V_\infty, \theta)$, which may not guarantee a satisfactory power generation performance due to the existence of model uncertainties $\Delta P(u; V_\infty, \theta)$. In addition, it is possible that the time-varying wind direction θ leads to different wake interaction patterns among the turbines and thus the optimal action of the problem (7) depends on wind direction θ. Hence, an efficient power optimization method needs to be capable of compensating for the effect of model uncertainties as well as dealing with varying wind conditions.

To evaluate the performance of different power optimization schemes, it is convenient in practice to use the power efficiency of wind farm [7] [28] defined below:

$$\eta(u; \theta) = (1/n)\sum\nolimits_{i=1}^{n} \eta_i(u_i; \{u_j\}_{j\in\mathcal{N}_i}, \theta), \quad (8)$$

where $\eta_i(u_i; \{u_j\}_{j\in\mathcal{N}_i}, \theta) = P_i/P_i^*$ defines the power efficiency of turbine i, $P_i^* = (1/2)\rho A_i C_{p,\max} V_\infty^3$ models the power output of turbine i without wake interactions, $C_{p,\max}$ denotes the maximum power coefficient calculated by (3) with $u_i = 1/3$. From (1) and (2), we have

$$\eta_i(u_i; \{u_j\}_{j\in\mathcal{N}_i}, \theta) = \frac{C_{p,i}(u_i)\left(1 - \delta V_i(\{u_j\}_{j\in\mathcal{N}_i}, \theta)\right)^3}{C_{p,\max}}. \quad (9)$$

Without loss of generality, suppose all turbines are identical and then $P_i^* = P_j^*$. Let $P^* = P_j^*$. The (8) can be rewritten as

$$\eta(u; \theta) = \frac{1}{nP^*}\sum\nolimits_{i=1}^{n} P_i. \quad (10)$$

According to (4), (6) and (10), we further get

$$\eta(u; \theta) = \bar{\eta}(u; \theta) + \Delta\eta(u; \theta), \quad (11)$$

where $\bar{\eta}(u; \theta) = \bar{P}/(nP^*)$ is the nominal model for wind farm power efficiency and $\Delta\eta(u; \theta) = \Delta P/(nP^*)$ denotes the integration of model uncertainties.

The power efficiency optimization problem of the wind farm can now be defined as

$$u_{opt} = \arg\max_{u\in\mathcal{U}} \eta(u; \theta). \quad (12)$$

Remark 2: It can be seen from (8) that the $\eta(u; \theta)$ is the normalization of the wind farm power output and the base value is the power output nP^* of the wind farm without wake interactions. Therefore, the maximization of the wind farm power output can be guaranteed by maximizing $\eta(u; \theta)$. In other words, the problem (7) and the problem (12) are equivalent.

Remark 3: It can be easily derived from (2) and (3) that the $u_i = 1/3$ for turbine $i \in N$ is the optimal control action to maximize its power output and is thus called greedy policy. However, as mentioned earlier, the greedy policy might not be optimal for problem (7) to maximize the total power output of wind farm due to the wake interactions between turbines.

Algorithm 1: Model-guided Learning Method for Wind Farm Power Optimization under Constant Wind Direction

Initialization:

$u_b^0 \in \mathcal{U}$, $0 < \beta_1^0 \le 1$, $0 \le \beta_2^0 \le 1$, $0 < \varepsilon_1 < 1$, $\delta > 0$,
$0 < \varepsilon_2 < 1$, $\eta_b^0 = \eta(u_b^0)$, $0 < \mu_1 < 1$, $\mu_2 > 0$, $0 < \mu_3 < 1$

For $k = 0, 1, \cdots$

Step 1: Action Update

$u^{k+1} = \prod_{\mathcal{U}}(u_b^k + \beta_1^k \Delta u_g^k + \beta_2^k \Delta u_s^k)$

$\Delta u_g^k = \nabla \bar{\eta}(u_b^k)$

$\Delta u_s^k = \begin{cases} \Delta u_{sl}^k & \text{with probability } \varepsilon_1 \\ \Delta u_{sg}^k & \text{with probability } 1-\varepsilon_1 \end{cases}$

$\Delta u_{sl}^k = (\bar{\omega}_i)_{1\times n}$, where $\bar{\omega}_i \in [-\delta, \delta]$ is chosen uniformly

$\Delta u_{sg}^k = (\omega_i)_{1\times n}$, where

$\omega_i = \begin{cases} \hat{\omega}_i & \text{with probability } \varepsilon_2 \\ 0 & \text{with with probability } 1-\varepsilon_2 \end{cases}$

$\hat{\omega}_i \in [u_{i,\min} - u_{b,i}^k,\ u_{i,\max} - u_{b,i}^k]$ is chosen uniformly

$u_{b,i}^k$ is the i_{th} component of the u_b^k

Step 2: Action Evaluation

Send action u^{k+1} to actual wind farm

Obtain power efficiency η^{k+1}

Step 3: Baseline Update

$u_b^{k+1} = \begin{cases} u^{k+1} & \text{If } \eta_b^k \le \eta^{k+1} \\ u_b^k & \text{else} \end{cases}$

$\eta_b^{k+1} = \begin{cases} \eta^{k+1} & \text{If } \eta_b^k \le \eta^{k+1} \\ \eta_b^k & \text{else} \end{cases}$

Step 4: Parameter Update

$\beta_1^{k+1} = \mu_1 \beta_1^k$

$\beta_2^{k+1} = \min\{1, \frac{1}{1+e^{-(k-\mu_2)}} + \mu_3\}$

III. Model-guided Learning for Constant Wind Direction

In this section, a model-guided learning method is proposed for the wind farm power efficiency optimization problem with fixed wind direction. In Section IV, time-varying wind direction will be considered.

A. Description and Interpretation of the Algorithm

It is assumed that the wind direction θ is constant. Then the optimal solution of the problem (12) stays fixed and the problem can be equivalently formulated as

$$u_{opt} = \arg\max_{u \in \mathcal{U}} \eta(u), \tag{13}$$

which is a nonlinear optimization problem that has inaccurate system model $\bar{\eta}(u)$ and bound constraint $u \in \mathcal{U}$.

Based on the projected gradient method [29], [30], the action update formula of the problem (13) can be given as follows:

$$u^{k+1} = \prod_{\mathcal{U}} \left(u^k + \beta_1^k \nabla \eta(u^k) \right). \tag{14}$$

where u^{k+1} is a new iteration point generated at iteration k, $\prod_{\mathcal{U}}(\bullet)$ denotes the Euclidean projection operator onto the $\mathcal{U}$, β_1^k is a parameter, and $\nabla \eta(u^k)$ denotes the gradient of the $\eta(u)$ at u^k. Note that it is extremely challenging to obtain the accurate gradient information $\nabla \eta(u^k)$ in (14) due to the difficulties in accurately modeling the wake interactions among the turbines. This results in that the action update formula (14) can not be directly carried out. From (11), we can obtain

$$\nabla \eta(u^k) = \nabla \bar{\eta}(u^k) + \nabla \left(\Delta \eta(u^k) \right). \tag{15}$$

Intuitively, the $\nabla \eta(u^k)$ can be approximated by using the gradient $\nabla \bar{\eta}(u^k)$ of the nominal model $\bar{\eta}(u)$ at u^k and thus an alternative for (14) can be run. However, the (15) clearly indicates that $\nabla \bar{\eta}(u^k)$ is different from real $\nabla \eta(u^k)$ because of the model uncertainties $\Delta \eta(u)$. This means that the use of the $\nabla \bar{\eta}(u^k)$ is likely to generate a wrong search direction at iteration point u^k, especially when the u^k is close to the optimal action u_{opt}, which hinders access to optimum and thus limits the power generation performance of wind farm. To address this problem, the key idea of SED [8] is introduced, i.e., how to obtain optimal solution using real-time measurement data. Then a model-guided learning method is developed for the problem (13), as shown in Algorithm 1.

In Step 1 of Algorithm 1, the action update is designed by using base action u_b^k, approximated gradient direction Δu_g^k based on nominal model, as well as data-driven random search direction Δu_s^k. The Δu_s^k is local search direction Δu_{sl}^k with probability ε_1 and global search direction Δu_{sg}^k with probability $1 - \varepsilon_1$. The use of Δu_{sl}^k achieves the local exploration around the u_b^k and Δu_{sg}^k guarantees the exploration for whole action space by exploiting u_b^k. Additionally, the Euclidean projection operator $\Pi_{\mathcal{U}}(\bullet)$ is used to guarantee the new action u^{k+1} satisfy the control constraints of all turbines. In Step 2, the action u^{k+1} is evaluated by actual wind farm to obtain corresponding power generation data.

From Step 3 of Algorithm 1, it can be noticed that the u_b^k has higher or equal power efficiency than the u^k. Then in Step 1 of Algorithm 1, the u_b^k is a better baseline for calculating u^{k+1} than the u^k used in (14), whose application can accelerate the convergence speed of the algorithm. Meanwhile, the use of the u_b^k can prevent Algorithm 1 from iterating continuously along the wrong search direction as the u_b^k is updated only when new action u^{k+1} shows higher or equal power efficiency (See Step 3). The Δu_g^k can quickly improve the power output of wind farm as an efficient nominal model $\bar{\eta}(u)$ can commonly provide a good search direction that has an acute angle with the correct direction induced by accurate model when the u^k is far away from optimal action u_{opt}.

In Step 4, parameter β_1^k is gradually decreased and β_2^k is monotonically increasing with the increase of k. This means that in the early optimization stage, the action update of Step 1 is performed based on u_b^k mainly along the gradient direction Δu_g^k of nominal model. With the decrease of β_1^k and increase of β_2^k, the action update will be performed based on u_b^k mainly along the random search direction Δu_s^k. And the Δu_s^k is selected as global search direction Δu_{sg}^k (generated based on the idea of SED) with probability $1 - \varepsilon_1$. This can guarantee optimum as k tends to ∞ as shown in following Theorem 1.

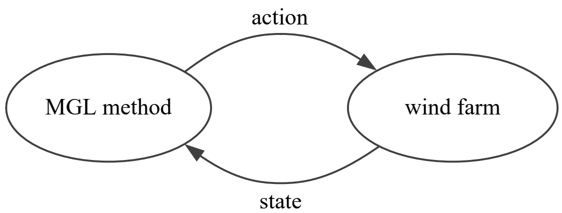


Fig. 2. Interaction between MGL method and wind farm.

Remark 4: The optimum point of inaccurate model can be directly chosen as initial point for data-driven methods, which might be a bad choice when the accuracy of the model is low. The proposed method can avoid this problem for wind farm by initializing base action with greedy policy. Note that the gradient estimated from measurements can provide a good search direction but generally guarantee a Karush-Kuhn-Tucker (KKT) point for the optimization problem with structural model mismatch. Although random search cannot contribute to a good search direction, it has ability to find optimal solution by exploration. As the inaccurate model can commonly ensure a good benchmark action, the random search is used in Algorithm 1 to achieve optimum.

Remark 5: Note that the Algorithm 1 integrates the key ideas of the projected gradient method [29], [30] and SED [8]. The projected gradient method is a model-based method and SED is a data-driven learning algorithm. The inaccurate model can commonly provide a good search direction for the action update of the Algorithm 1 (especially in the early optimization process), which can be taken as a guidance for the algorithm. Therefore, the Algorithm 1 is termed model-guided learning (MGL) method, where the learning features are reflected in Step 1 and 3.

Remark 6: The proposed MGL method is an online learning method. As shown in Fig. 2, the MGL method solves the wind farm power optimization problem by interacting with real wind farm. In each interaction, the MGL method updates its baselines by using measured state data from real wind farm. Meanwhile, it decides control action and sends the action to the wind farm. The wind farm runs the receiving control action and then feeds the corresponding state data back to the MGL method to obtain next action. On the other hand, the greedy policy remains constant for all the times (iterations).

Remark 7: Although the MGL method is proposed in this paper for the wind farm power efficiency optimization problem with fixed wind direction, it can solve a nonlinear optimization problem that has inaccurate system model and bound constraint.

B. Convergence Properties and Implementation of the Algorithm

The convergence property of the Algorithm 1 is given in the following theorem:

Theorem 1: Consider the application of the Algorithm 1 to the problem (13). The optimal solution u_{opt} can be found with probability 1.

Proof: See the Appendix A. ■

The above theorem shows that even if only an approximate model is available, the proposed algorithm can still achieve the optimal wind farm power output, which implies that the model uncertainties can be compensated by the proposed algorithm and thus the robustness can be achieved. Note that this is attractive in practice. Furthermore, the following corollary will indicate that under some conditions, the power efficiency of wind farm monotonically improves at the initial stage.

Corollary 1: Assume the $\nabla\eta(u)$ is Lipschitz continuous with positive constant $L > 0$, i.e., for any $u', u'' \in \mathcal{U}$,

$$\|\nabla\eta(u') - \nabla\eta(u'')\| \le L\|u' - u''\|. \tag{16}$$

If there is a positive integer $K > 0$ such that $k \le K$,

$$\left(\nabla\eta(u_b^k) - \frac{L}{2}G^k\nabla\bar{\eta}(u_b^k)\right)^T G^k\nabla\bar{\eta}(u_b^k) \ge 0, \tag{17}$$

then for any $k \le K$, the proposed Algorithm 1 has following property that the power efficiency improves monotonically:

$$\eta(u^{k+1}) \ge \eta(u^k), \tag{18}$$

where $G^k = diag(g_1^k, \cdots, g_n^k)$,

$$g_i^k = \begin{cases} \frac{u_{i,min} - u_{b,i}^k}{\beta_1^k \Delta u_{g,i}^k}, & \text{If } \hat{u}_i^k < u_{i,min}, \\ 1, & \text{If } u_{i,min} \le \hat{u}_i^k \le u_{i,max}, \\ \frac{u_{i,max} - u_{b,i}^k}{\beta_1^k \Delta u_{g,i}^k}, & \text{If } u_{i,max} < \hat{u}_i^k, \end{cases} \tag{19}$$

$\Delta u_{g,i}^k$ is the i_{th} component of the Δu_g^k, $\hat{u}_i^k = u_{b,i}^k + \beta_1^k \Delta u_{g,i}^k$, $i = 1, \cdots, n$.

Proof: See the Appendix B. ■

Remark 8: There are two assumptions in Corollary 1. One requires that the gradient $\nabla\eta(u)$ of wind farm power efficiency model $\eta(u)$ is Lipschitz continuous for fixed wind direction, i.e., (16). According to (8), the assumption (16) essentially requires that the sum of the power generation models of all turbines should be Lipschitz continuous for fixed wind condition. Another assumption requires that the gradients of $\eta(u)$ and $\bar{\eta}(u)$ should satisfy (17) to monotonically improve the power generation performance of wind farm in initial finite iterations. Note that the above assumptions are technical assumptions needed to prove the Corollary 1 and impose no restrictions on how the turbines should be built or controlled. Practically, the assumption (16) is trivially satisfied for any physical systems (such as the turbines); The assumption (17) can be checked during the running of Algorithm 1, the failure of which will only affect the monotonic convergence property, but the proposed algorithm will eventually converge as shown in Theorem 1, which is of ultimate importance.

Note that the inaccurate system model $\bar{\eta}(u)$ has a complex form so that it can be difficult to directly calculate its gradient analytically in Algorithm 1. The central-difference formula in [31] can be used to estimate the partial derivative of $\bar{\eta}(u)$ with respect to the i_{th} variable u_i, namely

$$\frac{\partial\bar{\eta}(u)}{\partial u_i} \approx \frac{\bar{\eta}(u + \varepsilon e_i) - \bar{\eta}(u - \varepsilon e_i)}{2\varepsilon}, \tag{20}$$

where ε is a small positive scalar and e_i is the i_{th} unit vector, whose elements are all 0 except for a 1 in the i_{th}

position, $i = 1, \cdots, n$. According to [31], the estimation error in (20) is $o(\varepsilon^2)$, which means that the small ε can guarantee a good estimation accuracy. Furtherly, the gradient $\nabla\bar{\eta}(u)$ is approximated by $\nabla\bar{\eta}(u) = (\frac{\partial\bar{\eta}(u)}{\partial u_1}, \cdots, \frac{\partial\bar{\eta}(u)}{\partial u_n})$.

Note that the $u \in \mathcal{U}$ in (13) is a bound constraint. Then the projection $\prod_{\mathcal{U}}(u)$ in Step 1 of Algorithm 1 can be calculated componentwise as

$$\left[\prod\nolimits_{\mathcal{U}}(u)\right]_i = \begin{cases} u_{i,min}, & \text{If } u_i < u_{i,min}, \\ u_i, & \text{If } u_{i,min} \leq u_i \leq u_{i,max}, \\ u_{i,max}, & \text{If } u_{i,max} < u_i, \end{cases} \tag{21}$$

which keeps u^{k+1} in its feasible domain $\mathcal{U}$.

IV. Hierarchical Power Optimization Scheme for Time-varying Wind Direction

In this section, a hierarchical power optimization scheme is developed for wind farm to handle time-varying wind direction using the proposed model-guided learning method.

A. Dividing the Wind Direction Interval into Subintervals

Firstly, the whole wind direction interval is divided into a finite number of sub-intervals, during which the wind farm power efficiency is insensitive to the changes of wind direction. Hence for each wind direction sub-interval, the coupling strength among the turbines is similar under the different wind directions belonging to the sub-interval and thus only one wake interaction pattern is required to be considered in optimization.

To do this, the historical power generation data of actual wind farm with greedy policy should be obtained, which can be easily achieved due to the wide application of the policy in practice. Then the power efficiency $\eta(u;\theta)$ of the wind farm under all wind directions can be computed by (8). Note that the wakes are the inherent characteristic of wind farm. Therefore, the power efficiency of the wind farm with greedy policy can reflect the coupling strength between turbines even if the policy does not consider the wakes. According to the obtained $\eta(u;\theta)$, the entire range of wind direction θ can be divided into a certain number of m sub-intervals denoted by $\Theta_1, \cdots, \Theta_m$. The $\Theta_j = [\theta_{j,\min}, \theta_{j,\max})$ denotes the j_{th} sub-interval, where $\theta_{j,\min}$ and $\theta_{j,\max}$ represent the lower bound and upper bound of the sub-interval Θ_j respectively, $j = 1, \cdots, m$. For any $\theta_1, \theta_2 \in \Theta_j$, it is required that

$$|\eta(u;\theta_1) - \eta(u;\theta_2)| \leq \varsigma, \tag{22}$$

where $\varsigma \in [0, 1)$ is a small positive constant. The selected ς should ensure that the $\eta(u;\theta)$ has only minor changes for the changes of θ in sub-interval Θ_j, $j = 1, \cdots, m$. Therefore, it can be assumed that only one wake interaction pattern exists for each divided wind direction sub-interval.

The power efficiency optimization sub-problem of wind farm for sub-interval Θ_j is defined as

$$u_{opt,j} = \arg\max_{u\in\mathcal{U}} \eta^j(u), \tag{23}$$

where $\eta^j(u)$ denotes the power efficiency function of wind farm under $\theta \in \Theta_j$, $j = 1, \cdots, m$. Define

$$\alpha_j(\theta) = \begin{cases} 1 & \text{If } \theta \in \Theta_j, \\ 0 & \text{else.} \end{cases} \tag{24}$$

Then the power efficiency optimization problem (12) of wind farm can be reformulated as

$$u_{opt} = \arg\max_{u\in\mathcal{U}} \sum\nolimits_{j=1}^{m} \alpha_j(\theta)\eta^j(u). \tag{25}$$

The above (25) indicates that the problem (12) can be denoted as the sum of m sub-problems defined in the wind direction sub-intervals. Each sub-problem can be approximately considered as a wind farm power optimization problem with fixed wind direction that is considered in Section III as its wake interaction pattern is almost invariable in the corresponding wind direction sub-interval.

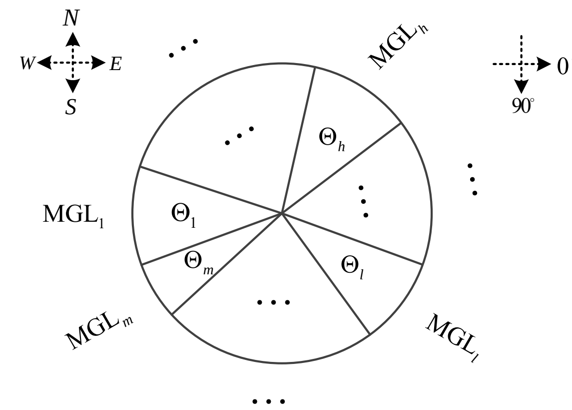


Fig. 3. Hierarchical framework of wind farm power optimization scheme.

Algorithm 2: Model-guided Learning Policy for Wind Farm Power Optimization under Time-varying Wind Direction

---

Initialization:
  Measure initial wind direction θ_0
For $t = 0, 1, \cdots$
  Step 1: Action Update
    If $\theta_t \in \Theta_j$, $j \in M$:
      Decide action u_t by MGL$_j$
  Step 2: Action Evaluation
    Send action u_t to actual wind farm
    Obtain data $V_{\infty,t+1}$, θ_{t+1}, and P_{t+1} from the wind farm
  Step 3: Policy Update
    If $\theta_t \in \Theta_j$ and $\theta_{t+1} \in \Theta_j$, $j \in M$:
      Update the baselines and parameters of MGL$_j$

B. Hierarchical Power Optimization Scheme

The m copies of Algorithm 1, denoted by MGL$_j$, $j = 1, \cdots, m$, are introduced to handle time-varying wind direction. Let each copy only focus on one sub-problem defined in (23). As shown in Fig. 3, MGL$_h$ is used to solve the wind farm power efficiency optimization sub-problem defined in wind direction sub-interval Θ_h. Then a hierarchical power

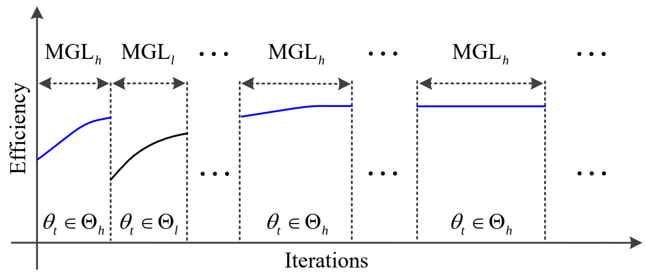


Fig. 4. Work flowchart of wind farm power optimization scheme.

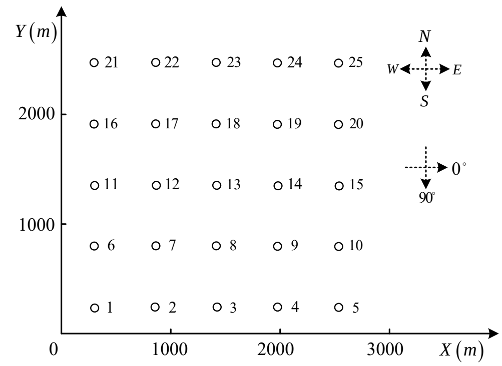


Fig. 5. Layout of 25-turbine wind farm.

optimization scheme is proposed for the problem (25) as shown in Algorithm 2. It is called MGL policy due to the use of previously proposed model-guided learning method for constant wind direction. In each iteration of Algorithm 2, the action u_t is firstly given by copy MGL$_j$ for new wind direction θ_t if $\theta_t \in \Theta_j$, where $j \in M$ and $M = \{1, 2, \cdots, m\}$. Note that wind direction θ_t only belongs to one of the divided sub-intervals and thus only one copy of Algorithm 1 would be run in Step 1 for the θ_t. In Step 2, the action u_t is sent to actual wind farm and the corresponding state data ($V_{\infty,t+1}$, θ_{t+1}, and P_{t+1}) is obtained from the wind farm. If the successive wind directions θ_t and θ_{t+1} belong to same wind direction sub-interval, the wake interactions among the turbines do not change in this sampling period. Then based on the sampled data, the calculated wind farm power efficiency can reflect the performance of the action u_t for θ_t if wind speed also has no change in the sampling period. This implies that the action u_t is efficiently evaluated, and thus the baselines and parameters of the corresponding copy would be updated in Step 3.

The work flowchart of the proposed MGL policy is shown in Fig. 4. The policy can make good use of the learned knowledge. As shown in Fig. 4, the MGL$_h$ is run to obtain the $u_{opt,h}$ in $\theta_t \in \Theta_h$; The MGL$_h$ is terminated and MGL$_l$ would be started with the switch of wind direction θ_t from sub-interval Θ_h to Θ_l; The MGL$_h$ would be reactivated and continue seeking $u_{opt,h}$ based on the previous learned knowledge while the sub-interval Θ_h is visited again. This would be beneficial to enhance the convergence speed of algorithms and thus quickly improve the power generation performance of wind farm in time-varying wind conditions.

Remark 9: The Step 2 of Algorithm 2 implies that the proposed MGL policy is also an online learning method and solves the wind farm power optimization problem under time-varying wind direction by interacting with actual wind farm. At each iteration of Algorithm 2, the sub-interval that wind direction θ_t belongs to is found first and then the corresponding copy of Algorithm 1 is carried out. This accounts for the hierarchical idea.

Remark 10: To handle varying wind direction, a Bayesian Ascent algorithm is employed for each wind direction in [28]. However, this method may show low optimization efficiency for stochastically time-varying wind direction in a large interval. In this paper, based on the power efficiency data of wind farm with greedy policy under different wind directions, the whole wind direction interval is divided into a finite number of sub-intervals, and then for each sub-interval, one proposed algorithm is employed, which can improve the exploitation rate of wind farm power generation data and thus is beneficial to quickly improve the power generation performance of the wind farm.

Remark 11: The measurement uncertainty of wind direction is not considered in this paper but it might make the control action jump from one sub-problem to another. In this case, the running control actions can still guarantee a better power performance for wind farm as the optimal control actions for turbines have no large change when the changes of wind direction are small. Additionally, this problem can be mitigated by using some methods. For example, multiple wind directions are measured simultaneously in each iteration and their average value is selected as the input of the proposed policy.

Remark 12: In this paper, the changes of wind speed are not specifically considered. It can be easily found that the wind farm power efficiency is independent of wind speed from (8) and (9). Therefore, it is not required that the wind speed stays the same for the continuously multiple iterations of the algorithm when the wind farm power efficiency is used to evaluate the action performance. The sampled data is feasible for wind farm power optimization as long as wind direction lies in one sub-interval and wind speed has no change in given sampling period.

V. Simulation Results

In this section, two simulation examples are given to illustrate the performance of the MGL policy. The first example is performed in simple wind conditions. The second example uses more realistic and complex wind conditions from a real wind farm.

The wind farm with 25 turbines shown in Fig. 5 is considered, where the spacing between adjacent turbine pair is $560m$. All turbines are identical and their diameters are $126m$. The roughness coefficient κ is 0.025. The air density ρ is $1.225kg/m^3$. The upstream wind speed V_∞ is set as $8m/s$. A common feasible domain $\mathcal{U}_i = \{u_i | 0.1 \leq u_i \leq 0.33\}$ is selected for the control action u_i of turbine $i \in N$, which is sufficient to verify the performance of different control policies [19]. The $\bar{P}(u; V_\infty, \theta)$ based on Park model is assumed as approximate nominal wind farm power generation model with key features. The FLORIS model developed in [32] is used

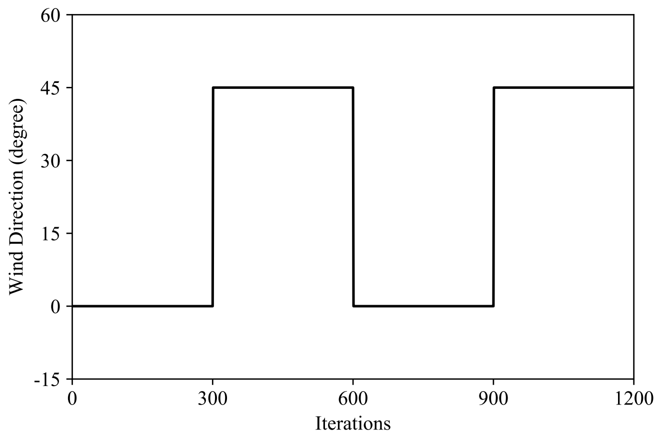


Fig. 6. Simple wind conditions.

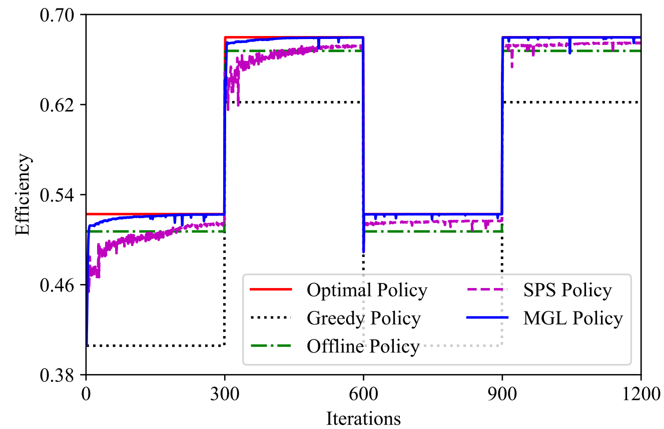


Fig. 7. Trajectories of power efficiency in simple wind conditions.

to simulate the accurate wind farm power generation model $P(u;V_\infty,\theta)$, which includes the wake interactions among the turbines and is widely used in many references to evaluate the effectiveness of different wind farm power optimization schemes. The FLORIS model is a combination of Park model, Jiménez model for wake deflection and further modifications to better model the wake velocity profile. Note that the wind direction is a parameter of the FLORIS model, and thus the wind farm with different wind directions can be simulated by inputting different wind directions to the model.

The parameters of Algorithm 1 are set as $\beta_1^0=1$, $\beta_2^0=0$, $\varepsilon_1=0.95$, $\delta=0.006$, $\varepsilon_2=0.05$, $\mu_1=0.9$, $\mu_2=15$, $\mu_3=1e-5$, respectively. The base action u_b^0 is initialized by greedy policy $u=(0.33,\cdots,0.33)$. To demonstrate the advantages of the proposed MGL policy, the following optimization methods are selected: (a) Greedy policy, which is usually used as a benchmark to assess the power performance of wind farm under different control policies; (b) Offline policy (i.e., model-based gradient method), which is obtained by using nominal model $\bar{P}(u;V_\infty,\theta)$ via gradient ascent method and thus belongs to traditional model-based methods; (c) Stochastic projected simplex (SPS) policy, which is proposed to solve the wind farm power optimization problem in a data-driven manner [21] and shows better power generation performance than the SED benchmark [8], and thus is used here for comparison; (d) Optimal policy, which is derived by using simulated accurate model $P(u;V_\infty,\theta)$ based on gradient ascent method, and is unknown for a real wind farm since the accurate model $P(u;V_\infty,\theta)$ cannot be easily obtained.

A. Simple Wind Conditions Example

As shown in Fig. 6, this example assumes that the wind direction changes slowly in an angle set of $\{0^\circ,45^\circ\}$. The MGL policy is made up of the 2 copies of Algorithm 1 for the angle set. For comparison purpose, the wind farm power optimizations are performed based on our proposed MGL policy and the above selected policies. Fig. 7 shows the power efficiency trajectories of the wind farm under different control policies. Fig. 8 gives the control action trajectories of turbine 11, 12, 13, and 14 in the wind farm.

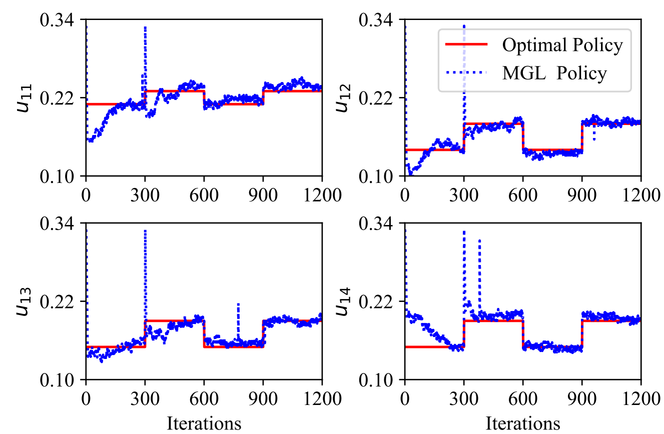


Fig. 8. Trajectories of control actions in simple wind conditions.

TABLE I
THE PERCENTAGE OF POWER EFFICIENCY IMPROVEMENT OF WIND FARM UNDER MGL POLICY COMPARED WITH BENCHMARK POLICIES

Wind direction	Greedy policy	Offline policy	SPS policy
$\theta=0^\circ$	28.7%	3.0%	1.1%
$\theta=45^\circ$	9.2%	1.7%	0.7%

From Fig. 7, it can be seen that the power efficiency of wind farm with our MGL policy at θ=0° and θ=45° approaches the optimal values while other policies could not. This illustrates that the MGL policy can compensate the effect of model uncertainties on wind farm power generation performance compared with the method only based on analytical power generation model, i.e., offline policy. Furthermore, Fig. 7 shows that the MGL policy has much faster convergence speed than the purely data-driven SPS policy that is one of best performing policies among existing designs, mainly because the available model provides a relatively reliable search direction for the copies of Algorithm 1 in the early optimization stage and thus accelerates the convergence speeds of the copies. Additionally, it can be observed that the proposed MGL policy can adapt to time-varying wind conditions and monotonically improves the power efficiency of wind farm during the initial stage, which

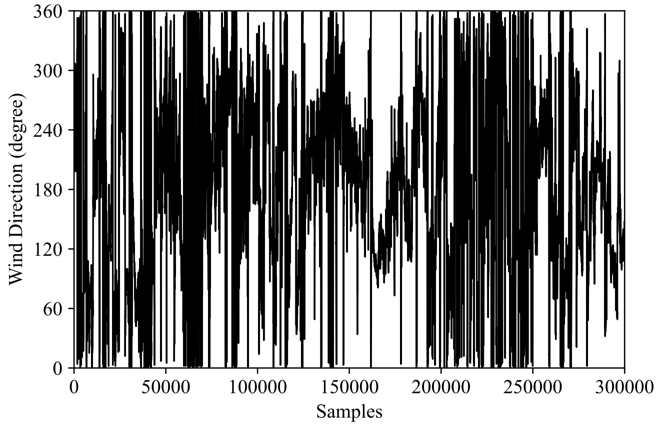


Fig. 9. Complex wind conditions.

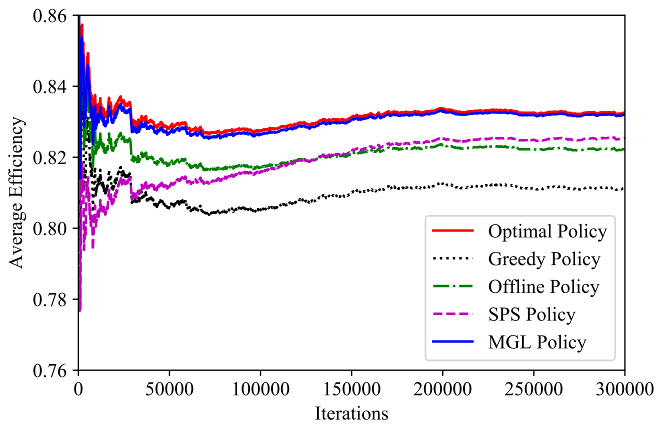


Fig. 10. Trajectories of average power efficiency in complex wind conditions.

verifies the expectation from the Corollary 1. In a quantitative manner, Table I shows that the proposed policy efficiently improves the power efficiency of wind farm compared with other all policies. It can be concluded that the proposed policy can efficiently improve the power output of wind farm in simple wind conditions. It is also worth mentioning that in the considered application, this level of improvement represents a substantial performance improvement.

In each iteration, the average computation time taken by the proposed policy is about 0.12s with a PC of Intel Core i7-8700 CPU @ 3.20GHz, 32.GB RAM, and NVIDIA GeForce RTX 2070, and thus it can be negligible. Fig. 8 shows that the proposed policy can guarantee the feasibility of the resulting control actions for all turbines, where it can be clearly seen that the control actions of some turbines make great changes to achieve the optimum compared with the greedy policy that is widely applied in practice.

B. Complex Wind Conditions Example

In this simulation, more complex wind conditions are considered. The 10-minute wind direction statistics of Anholt offshore wind farm can be accessed in [33]. As the number of the wind direction statistics is small and sample interval (10 minutes) is big, the two copies of each wind direction of the statistics are interpolated behind the wind direction. Then the 3.3-minute wind data shown in Fig. 9 is generated and used in this example to verify the performance of the policies under real/complex wind conditions. The greedy policy is applied to the simulated accurate model and then the corresponding power generation data is obtained for $\theta \in [0°, 360°)$. The data is taken as the historical data from actual wind farm. Then the power efficiency of the wind farm can be computed by (8). The constant ς is selected as 0.02. Based on (22), the entire wind direction interval is divided into the 203 sub-intervals. This means that there are 203 sub-problems and thus the 203 copies of the Algorithm 1 would be run in MGL policy to solve the wind farm power optimization problem under time-varying wind direction. Each copy is allowed to run 300 iterations, after which its base action will be implemented when the corresponding wind direction sub-interval is visited. Fig. 10 shows the average power efficiency of the wind farm in complex wind conditions.

It can be observed that the proposed MGL policy shows more superior performance than greedy policy, offline policy, and SPS policy. At iteration $k \in (250000, 300000]$, the average power efficiency of the wind farm under MGL policy reaches the 99.9% of optimum, whose average improvement rates are respectively about 2.5%, 1.2%, and 0.8% compared with above three policies, representing significant power production improvement. Therefore, the proposed MGL policy can efficiently improve the power generation performance of the wind farm in complex wind conditions.

VI. Conclusion

In this paper, a model-guided learning method is developed for wind farm to mitigate the effect of wake interactions among the turbines and maximize the power output of the wind farm. It is extremely challenging to accurately model the interactions due to the highly complex nature of the wakes and thus only an approximate wind farm power generation model can be obtained in practice. The proposed method can effectively compensate for the uncertainties of the model due to the efficient exploitation for real-time power generation data and also shows the fast convergence speed by using the knowledge of the available model. A hierarchical wind farm power optimization scheme is then proposed to handle time-varying wind conditions using the proposed model-guided learning method.

Simulation results are presented to demonstrate the effectiveness of the proposed scheme in different wind conditions. The results in simple wind conditions example show that the proposed scheme can quickly improve the power output of wind farm, compensate the effect of model uncertainties on wind farm power generation performance and adapt to time-varying wind conditions. The complex wind conditions example further verifies the performance of the proposed scheme in more realistic environment conditions, where the real wind direction data are used.

There are a number of topics to further investigate. In this paper, the control framework is centralized and may not be appropriate for very large-scale wind farm power optimization due to scalability and reliability issues. Based on available analytical model and real-time power generation data, the distributed power optimization of large-scale wind farm is a key area of our future research. Modifier adaptation schemes can also solve the optimization problem with structural plant-model mismatch by introducing correction terms for the cost and constraint functions [34]. Then wind farm power optimization using modifier adaptation will be another key area of our future research. For the slowly changing wind conditions considered in this paper, the aerodynamics of wind turbines and the transients in the changes of wind conditions are ignored. Incorporating turbine aerodynamics and the transients in the changes of wind conditions into the design will be a focus of our future research. A completely novel concept, i.e., dynamic individual pitch control (DIPC), is proposed in [35] and the simulations show that it can be effective at increasing wake recovery and improving the power production of wind farm. Therefore, the optimal control settings for the DIPC will be also a point of attention for our future research.

Appendix I
Proof of Theorem 1

In Step 4 of Algorithm 1, we notice that

$$\beta_1^{k+1} = (\mu_1)\beta_1^k = (\mu_1)^2\beta_1^{k-1} = \cdots = (\mu_1)^{k+1}\beta_1^0. \tag{26}$$

As $0 < \mu_1 < 1$, $0 < \beta_1^0 \le 1$, the sequence $\{\beta_1^k\}$ is strictly monotonically decreasing with the increasing of k. Meanwhile, the $\beta_1^{k+1} > 0$ holds for any k. Clearly, the sequence $\{\beta_1^k\}$ is convergent by using monotone bounded theorem and the limit of $\{\beta_1^k\}$ is 0. Therefore, there is a positive integer K_1 such that at $k > K_1$, the term $\beta_1^k \Delta u_g^k$ in Step 1 of Algorithm 1 can be ignored as $\|\Delta u_g^k\|$ is bounded. Then

$$u^{k+1} = u_b^k + \beta_2^k \Delta u_{sg}^k \tag{27}$$

if Δu_s^k is selected as Δu_{sg}^k at $k > K_1$ with probability $1-\varepsilon_1$.

Let

$$K_2 = \max\{K_1,\ \mu_2 - \log\frac{\mu_3}{1-\mu_3}\}. \tag{28}$$

From Step 4 of Algorithm 1, $\beta_2^{k+1} = 1$ at $k > K_2$. This means that once Δu_s^k is chosen as Δu_{sg}^k at $k > K_2$ with probability $1-\varepsilon_1$, the whole action space $\mathcal{U}$ would be explored randomly by u^{k+1}.

Assume $u_{opt} = (u_1^*, \cdots, u_n^*)$ is the optimal solution and $u_{opt} \in \mathcal{U}$. Let

$$\Delta u_{ob}^k = (u_1^* - u_{b,1}^k, \cdots, u_n^* - u_{b,n}^k). \tag{29}$$

The event that the Δu_{sg}^k is selected as Δu_{ob}^k at $k > K_2$ and thus u^{k+1} constitutes the optimal solution u_{opt} in (27) occurs with at least probability

$$(1-\varepsilon_1)(\frac{\varepsilon_2}{|\mathcal{U}_1|})\cdots(\frac{\varepsilon_2}{|\mathcal{U}_n|}), \tag{30}$$

where $|\mathcal{U}_i|$ denotes the cardinality of the action set $\mathcal{U}_i$ of turbine i. Therefore, the optimal solution u_{opt} will eventually be found with probability 1 for any $0 < \varepsilon_1 < 1$ and $0 < \varepsilon_2 < 1$.

Appendix II
Proof of Corollary 1

Define $\Delta u_b^k = u^{k+1} - u_b^k$. For any $k \le K$, we have

$$\begin{aligned}\eta(u^{k+1}) &= \eta(u_b^k) + \int_0^1 \nabla\eta(u_b^k + \tau\Delta u_b^k)^T \Delta u_b^k \mathrm{d}\tau \\ &= \eta(u_b^k) + \nabla\eta(u_b^k)^T \Delta u_b^k \\ &\quad + \int_0^1 \left(\nabla\eta(u_b^k + \tau\Delta u_b^k) - \nabla\eta(u_b^k)\right)^T \Delta u_b^k \mathrm{d}\tau.\end{aligned} \tag{31}$$

Then

$$\begin{aligned}&\left|\eta(u^{k+1}) - \eta(u_b^k) - \nabla\eta(u_b^k)^T \Delta u_b^k\right| \\ &\le \int_0^1 \left|\left(\nabla\eta(u_b^k + \tau\Delta u_b^k) - \nabla\eta(u_b^k)\right)^T \Delta u_b^k\right| \mathrm{d}\tau \\ &\le \int_0^1 \|\nabla\eta(u_b^k + \tau\Delta u_b^k) - \nabla\eta(u_b^k)\| \|\Delta u_b^k\| \mathrm{d}\tau,\end{aligned} \tag{32}$$

where the triangle inequality is used. By (32) and (16), we get

$$\begin{aligned}&\left|\eta(u^{k+1}) - \eta(u_b^k) - \nabla\eta(u_b^k)^T \Delta u_b^k\right| \\ &\le \int_0^1 L\|u_b^k + \tau\Delta u_b^k - u_b^k\| \|\Delta u_b^k\| \mathrm{d}\tau \\ &= \int_0^1 L\tau\|\Delta u_b^k\|^2 \mathrm{d}\tau \\ &= \frac{L}{2}\|\Delta u_b^k\|^2.\end{aligned} \tag{33}$$

Hence,

$$\eta(u^{k+1}) \ge \eta(u_b^k) + \nabla\eta(u_b^k)^T \Delta u_b^k - \frac{L}{2}\|\Delta u_b^k\|^2. \tag{34}$$

The $\beta_2^k = o(\beta_1^k)$ holds for any $k \le K$ if μ_2 and μ_3 are appropriately selected. Then for any $k \le K$, the term $\beta_2^k \Delta u_s^k$ can be ignored as $\|\Delta u_s^k\|$ is bounded. Thus, in Step 1 of Algorithm 1,

$$u^{k+1} = \prod\nolimits_{\mathcal{U}} (u_b^k + \beta_1^k \Delta u_g^k) \tag{35}$$

for any $k \le K$ by selecting appropriate μ_2 and μ_3. From (35), (21) and (19), we get

$$u^{k+1} = u_b^k + \beta_1^k G^k \Delta u_g^k. \tag{36}$$

Based on (34) and (36), we obtain

$$\begin{aligned}\eta(u^{k+1}) &\ge \eta(u_b^k) + \nabla\eta(u_b^k)^T (u_b^k + \beta_1^k G^k \Delta u_g^k - u_b^k) \\ &\quad - \frac{L}{2}\|u_b^k + \beta_1^k G^k \Delta u_g^k - u_b^k\|^2 \\ &= \eta(u_b^k) + \beta_1^k \nabla\eta(u_b^k)^T G^k \nabla\bar{\eta}(u_b^k) \\ &\quad - \frac{L}{2}(\beta_1^k)^2 \left(G^k \nabla\bar{\eta}(u_b^k)\right)^T G^k \nabla\bar{\eta}(u_b^k).\end{aligned} \tag{37}$$

As $0 < \beta_1^k \leq 1$, we get by (37)

$$\begin{aligned} \eta(u^{k+1}) &\geq \eta(u_b^k) + \beta_1^k \nabla\eta(u_b^k)^T G^k \nabla\bar{\eta}(u_b^k) \\ &\quad - \frac{L}{2}\beta_1^k \left(G^k \nabla\bar{\eta}(u_b^k)\right)^T G^k \nabla\bar{\eta}(u_b^k) \\ &= \eta(u_b^k) + \beta_1^k \mathcal{H}^k \\ &\geq \eta(u_b^k), \end{aligned} \tag{38}$$

where $\mathcal{H}^k = \left(\nabla\eta(u_b^k) - \frac{L}{2}G^k\nabla\bar{\eta}(u_b^k)\right)^T G^k \nabla\bar{\eta}(u_b^k)$ and the (17) is used. Then for any $1 \leq k \leq K+1$,

$$\eta(u^k) \geq \eta(u_b^{k-1}). \tag{39}$$

According to Step 3 of Algorithm 1, $\eta(u_b^k) = \eta(u^k)$. Therefore using (38), we obtain

$$\eta(u^{k+1}) \geq \eta(u^k). \tag{40}$$

Namely, the (18) holds.

Acknowledgment

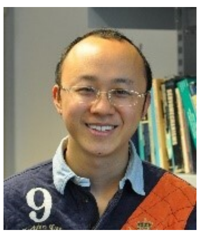
The authors would like to thank Yueqing Zhang (the School of Electronics and Computer Science, University of Southampton) for her valuable advice to the paper, Ørsted for their real wind data, the editor and anonymous reviewers for their insightful comments and suggestions.

References

[1] S. Vásquez, M. Kinnaert and R. Pintelon, "Active fault diagnosis on a hydraulic pitch system based on frequency-domain identification," *IEEE Trans. Control. Syst. Technol.*, vol. 27, no. 2, pp. 663–678, Mar. 2019.
[2] Global Wind Energy Council, "Global wind report 2019," Accessed on: Jan 20, 2021, [Online]. Available: https://gwec.net/global-wind-report-2019/.
[3] Wind Europe, "Wind can be the cornerstone of Europe's energy system," Accessed on: Jan 15, 2021, [Online]. Available: https://windeurope.org/about-wind/wind-energy-today/.
[4] H. Liao *et al.*, "Active power dispatch optimization for offshore wind farms considering fatigue distribution," *Renew. Energy*, vol. 151, pp. 1173–1185, Nov. 2020.
[5] M. Vali, V. Petrovic, L. Y. Pao, and M. Kühn, "Model predictive active power control for optimal structural load equalization in waked wind farms," *IEEE Trans. Control. Syst. Technol.*, vol. 30, no. 1, pp. 30–44, Jan. 2022.
[6] T. Ahmad *et al.*, "Field implementation and trial of coordinated control of wind farms," *IEEE Trans. Sustain. Energy*, vol. 9, no. 3, pp. 1169–1176, Jul. 2018.
[7] J. Park, K. H. Law, "Bayesian Ascent: A data-driven optimization scheme for real-time control with application to wind farm power maximization," *IEEE Trans. Control. Syst. Technol.*, vol. 24, no. 5, pp. 1655–1668, Sep. 2016.
[8] J. R. Marden, S. D. Ruben, L. Y. Pao, "A model-free approach to wind farm control using game theoretic method," *IEEE Trans. Control. Syst. Technol.*, vol. 21, no. 4, pp. 1207–1214, Jul. 2013.
[9] S. Boersma *et al.*, "A tutorial on control-oriented modeling and control of wind farms," in *Proc. Amer. Control Conf.*, Seattle, USA, May. 2017, pp. 1–18.
[10] C. Kim, Y. Gui, and C. C. Chung, "Maximum power point tracking of a wind power plant with predictive gradient ascent method," *IEEE Trans. Sustain. Energy.*, vol. 8, no. 2, pp. 685–694, Apr. 2017.
[11] J. A. Dahlberg, "Assessment of the Lillgrund windfarm: Power performance, wake effects," Vatenfall Vindkraft AB, Sweden, Rep. 6_1 LG Pilot Report, Sep. 2009.
[12] E. Bitar, P. Seiler, "Coordinated control of a wind turbine array for power maximization," in *Proc. Amer. Control Conf.*, Washington, USA, June. 2013, pp. 2898–2904.
[13] J. Park, S. Kwon, K. H. Law, "Wind farm power maximization based on a cooperative static game approach," in *Proc. SPIE.*, San Diego, CA, USA, Apr. 2013.
[14] N. Gionfra, G. Sandou, H. Siguerdidjane, D. Faille, P. Loevenbruck, "Wind farm distributed PSO-based control for constrained power generation maximization," *Renew. Energy*, vol. 133, pp. 103–117, Apr. 2019.
[15] J. P. Goit, W. Munters, J. Meyers, "Optimal coordinated control of power extraction in LES of a wind farm with entrance effects," *Energies*, vol. 9, no. 3, pp. 2–20, Jan. 2016.
[16] Y. T. Wu, F. Porté-Agel, "Modeling turbine wakes and power losses within a wind farm using LES: An application to the Horns Rev offshore wind farm," *Renew. Energy*, vol. 75, pp. 945–955, Mar. 2015.
[17] A. C. Kheirabadi, R. Nagamune, "A quantitative review of wind farm control with the objective of wind farm power maximization," *J. Wind Eng. Ind. Aerodynam.*, vol. 192, pp. 45–73, Sep. 2019.
[18] P. M. O. Gebraad, J. W. van Wingerden, "Maximum power-point tracking control for wind farms," *Wind Energy*, vol. 18, pp. 429–447, Feb. 2015.
[19] S. Zhong, X. Wang, "Decentralized model-free wind farm control via discrete adaptive filtering methods," *IEEE Trans. Smart Grid*, vol. 9, no. 2, pp. 2529–2540, Jul. 2018.
[20] N. Deljouyi, A. Nobakhti, A. Abdolahi, "Wind farm power output optimization using cooperative control methods," *Wind Energy*, vol. 24, no. 5, pp. 502–514, Oct. 2020.
[21] Z. Xu, H. Geng and B. Chu, "A hierarchical data-driven wind farm power optimization approach using stochastic projected simplex method," *IEEE Trans. Smart Grid*, vol. 12, no. 4, pp. 3560–3569, Jul. 2021.
[22] B. M. Doekemeijer, S. Boersma, L. Y. Pao, T. Knudsen, and J. W. van Wingerden, "Online model calibration for a simplified LES model in pursuit of real-time closed-loop wind farm control," *Wind Energ. Sci.*, vol. 3, pp. 749–765, Oct. 2018.
[23] B. M. Doekemeijer, P. A. Fleming, J. W. Van Wingerden, "A tutorial on the synthesis and validation of a closed-loop wind farm controller using a steady-state surrogate model," in *Proc. Amer. Control Conf.*, New York, USA, Jul. 2019, pp. 2825–2836.
[24] B. M. Doekemeijer, D. van der Hoek, J. W. van Wingerden, "Closed-loop model-based wind farm control using FLORIS under time-varying inflow conditions," *Renew. Energy*, vol. 156, pp. 719–730, Apr. 2020.
[25] L. E. Andersson, L. Imsland, "Real-time optimization of wind farms using modifier adaptation and machine learning," *Wind Energ. Sci.*, vol. 5, pp. 885–896, Jul. 2020.
[26] H. Zhao, J. Zhao, J. Qiu, G. Liang, Z. Y. Dong, "Cooperative wind farm control with deep reinforcement learning and knowledge assisted learning," *IEEE Trans. Ind. Informat*, vol. 16, no. 11, pp. 6912–6921, Nov. 2020.
[27] I. Katic, J. Hojstrup, N. Jensen, "A simple model for cluster efficiency," in *Proc. Eur. Wind Energy Conf.*, Rome, Italy, 1986, pp. 407–410.
[28] J. Park, K. H. Law, "A data-driven, cooperative wind farm control to maximize the total power production," *Applied Energy*, vol. 165, pp. 151–165, Mar. 2016.
[29] A. A. Goldstein, "Convex programming in Hilbert space," *Bull. Amer. Math. Soc.*, vol. 70, no. 5, pp. 709–710, May. 1964.
[30] P. H. Calamai, J. J. Moré, "Projected gradient methods for linearly constrained problems," *Math. Program.*, vol. 39, no. 1, pp. 93–116, Mar. 1987.
[31] J. Nocedal and S. J. Wright, *Numerical optimization*, 2nd ed. New York: Springer, 2006.
[32] P. M. O. Gebraad *et al.*, "Wind plant power optimization through yaw control using a parametric model for wake effects—a CFD simulation study," *Wind Energy*, vol. 19, pp. 95–114, Dec. 2016.
[33] Ørsted, "Offshore wind data," Accessed on: Oct 31, 2020, [Online], Available: https://orsted.com/en/our-business/offshore-wind/wind-data.
[34] A. G. Marchetti, G. François, T. Faulwasser and D. Bonvin, "Modifier adaptation for real-time optimization—methods and applications," *Processes*, vol. 4, no. 55, pp. 1–35, Dec. 2016.
[35] J. A. Frederik, B. M. Doekemeijer, S. P. Mulders, J.-W, van Wingerden, "The helix approach: using dynamic individual pitch control to enhance wake mixing in wind farms," *Wind Energy*, vol. 23, pp. 1739–1751, Mar. 2020.

Zhiwei Xu (Graduate Student Member, IEEE) received the B.S. degree in College of Mathematics and Statistics from Northwest Normal University, Lanzhou, China, in 2015 and the M.S. degree in School of Information Science and Engineering from Central South University, Changsha, China, in 2018. He is currently pursuing the Ph.D. degree with the Department of Automation, Tsinghua University, Beijing, China.

His current research interests include online learning, data-driven optimization, model-based learning control, reinforcement learning, distributed optimization, and their applications to cooperative wind farm control.



Bing Chu received the B.Eng. degree in automation and the M.Sc. degree in control science and technology from Tsinghua University, Beijing, China, in 2004 and 2007, respectively, and the Ph.D. degree in automatic control and systems engineering from the University of Sheffield, Sheffield, U.K. in 2009. He was a Post-Doctoral Researcher with the University of Oxford, Oxford, U.K., from 2010 to 2012. He is currently an Associate Professor in Electronics and Computer Science with the University of Southampton, Southampton, U.K.

His current research interests include iterative learning and repetitive control, analysis and control of large-scale networked systems, applied optimization theory, and their applications to robotics, energy and sustainability, and next generation healthcare. Dr. Chu was a recipient of a number of awards including the Best Paper Prize of the 2012 United Kingdom Automatic Control Council International Conference on Control and Certificate of Merit for the 2010 IET Control and Automation Doctoral Dissertation Prize.

Hua Geng (S'07–M'10–SM'14–F'21) received the B.S. degree in electrical engineering from Huazhong University of Science and Technology, Wuhan, China, in 2003 and the Ph.D. degree in control theory and application from Tsinghua University, Beijing, China, in 2008. From 2008 to 2010, he was a Postdoctoral Research Fellow with the Department of Electrical and Computer Engineering, Ryerson University, Toronto, ON, Canada. He joined Automation Department of Tsinghua University in June 2010.

His current research interests include advanced control on power electronics and renewable energy conversion systems. He has authored more than 170 technical publications and holds more than 20 issued Chinese/US patents. He is the editors of IEEE Trans. on Energy Conversion and IEEE Trans. on Sustainable Energy, associate editors of IEEE Trans. on Industry Applications, IET Renewable Power Generation, Control Engineering Practice. He is an IEEE Fellow and an IET Fellow, Standing Director of China Power Supply Society (CPSS), vice chair of IEEE IAS Beijing Chapter.

Xiaohong Nian received the B.S. degree from Northwest Normal University, Lanzhou, China, in 1985, the M.S. degree from Shandong University, Jinan, China, in 1996, and the Ph.D. degree from Peking University, Beijing, China, in 1999. From 2004 to 2008, he was a Research Fellow with the Institute of Zhuzhou Electric Locomotive, Zhuzhou, China. He is currently a Professor with Central South University, Changsha, China.

His current research interests include coordinated control and optimization of complicated multiagent systems, unmanned aerial vehicles, converter technology, and control of tractive power supply systems in highspeed train. Dr. Nian is an Editor of Converter Technology and Electric Traction.