A Data-Driven Approach for Cooperative Wind Farm Control

Jinkyoo Park, Soon-Duck Kwon, and Kincho H. Law

Abstract—**This paper discusses a data-driven, cooperative control strategy to maximize wind farm power production. Conventionally, every wind turbine in a wind farm is operated to maximize its own power production without taking into account the interactions among the wind turbines in a wind farm. Such greedy control strategy, when an upstream wind turbine attempts to maximize its power production, can significantly lower the power productions of the downstream wind turbines and, thus, reduces the overall wind farm power production. As an alternative, we propose a cooperative wind farm control strategy that determines and executes the optimum coordinated control actions that maximize the total wind farm power production. To determine the optimum coordinated control actions of the wind turbines, we employ Bayesian Ascent (BA), a probabilistic optimization method constructed based on Gaussian Process regression and the trust region concept. Wind tunnel experiments using 6 scaled wind turbine models are conducted to assess (1) the effectiveness of the cooperative control strategy in improving the power production, and (2) the efficiency of the BA algorithm in determining the optimum control actions of the wind turbines using only the input control actions and the output power measurement data.**

I. Introduction

Conventionally, a wind turbine in a wind farm is operated individually to maximize its own power production by adjusting its operational conditions, i.e., control actions. Under this greedy control strategy, the wake formed by the upstream wind turbine would potentially lower the power productions of the downstream wind turbines due to reduced wind speed and increased turbulence intensity inside the wake. Realizing that the interactions among the wind turbines can affect their power productions, this study investigates a cooperative strategy to coordinate the control actions that influence the wake interference pattern with the objective of maximizing the total power production of a wind farm.

To determine the optimum coordinated control actions of wind turbines, various approaches have been proposed. One approach is to formulate the wind farm control problem mathematically using an analytically derived wind farm power function, and to determine the optimum control actions by solving the formulated problem using an optimization scheme. For example, Park and Law formulated the cooperative wind farm control problem using the wind farm power function calibrated with CFD simulation data and proposed to solve the control problem using sequential quadratic programming [1]. As an alternative to constructing a wind farm power function, model-free optimization approaches have also been attempted to determine the control actions for maximizing the wind farm power. For examples, game theoretic search [2-3] and point tracking method [4] have been proposed to determine the optimum control actions using only the wind farm power output data. For model-free methods, the strategy is to iteratively find better control actions by executing trial actions and observing the consequent power outputs. The success of the model free control approaches strongly depends on the rate of convergence to an optimum and the robustness in handling noisy data.

Recently, efforts have been made to optimize a target system using scarce data by exploiting the expressivity of non-parametric regression model. For example, Bayesian Optimization (BO) iteratively determines the optimum of a target system through a sequence of learning and sampling steps [5-6]. At each iterate, BO approximates the input and the output relationship of the target system using Gaussian Process (GP) regression (learning) and uses the approximated model to determine the next set of inputs (sampling) that improve the target values. Park and Law have developed the Bayesian Ascent (BA) method that combines the strengths of the Bayesian Optimization (BO) and the gradient-free trust region approach [7]. BA adapts the strategy of regulating the optimization scope, as used in the Trust Region method, into the Bayesian Optimization (BO) framework to ensure that the algorithm can monotonically increase a target value. With the trust region constraint imposed on the sampling procedure, BA tends to increase the target value and results in rapid convergence towards the optimum.

In this study, we employ the Bayesian Ascent method to determine the optimum coordinated control actions for wind farm power production by exploiting the power measurement data obtained from the wind turbines. Numerical simulations using an analytically derived power function have demonstrated the potential of the BA algorithm for the control problem [8]. In this work, we focus on the implementation of the BA method for real-time control on a physical system. We conduct experimental wind tunnel studies using 6 scaled wind turbines to investigate (1) the effectiveness of the cooperative control strategy in terms of increasing the total wind farm power production, and (2) the feasibility of using the Bayesian Ascent (BA) algorithm in terms of deriving the optimum coordinated control actions for wind turbines (i.e., the optimum cooperative control strategy) using only power measurement data. We employ the Bayesian Ascent algorithm to the scaled wind farm with

Jinkyoo Park, PhD student, Department of Civil and Environmental Engineering, Stanford University, 473 Via Ortega, Stanford, CA 94305 USA (phone: 512-917-7751; e-mail: jkpark11@stanford.edu).

Soon-Duck Kwon, Professor of Civil Engineering and Director of KOECED Wind Tunnel Laboratory, Chonbuk National University, South Korea (e-mail: sdkwon@chonbuk.ac.kr).

Kinch H. Law, Professor, Department of Civil and Environmental Engineering, Stanford University, 473 Via Ortega, Stanford, CA 94305 USA (e-mail: law@stanford.edu).

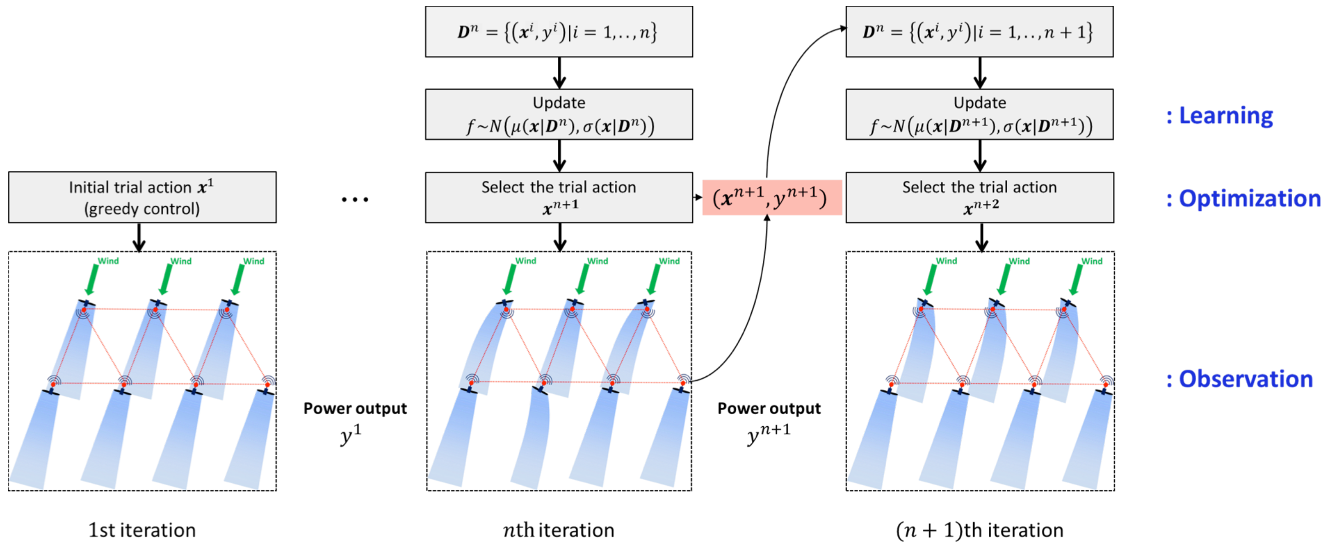


Figure 1. Bayesian optimization for wind farm power maximization problem

different configurations and wind directions to gain insight on the cooperative control using the BA algorithm.

II. Formulation of Cooperative Wind Farm Control

When each wind turbine tries to maximize its own objective function without considering the objectives of other wind turbines, the non-cooperative wind farm control problem can be formulated as

$$x_i^* = \underset{x_i}{\operatorname{argmax}} f_i(\boldsymbol{x}) = \underset{x_i}{\operatorname{argmax}} f_i(x_i, \boldsymbol{x}_{-i}) \tag{1}$$

in which wind turbine i maximizes its own power $f_i(\boldsymbol{x})$ with respect to its own control actions x_i regardless of the control actions $\boldsymbol{x}_{-i} = (x_1, \dots, x_{i-1}, x_{i+1}, \dots x_N)$ of other wind turbines. For a non-cooperative game, there exists an equilibrium point $\boldsymbol{x}^* = (x_1^*, \dots, x_N^*)$, called the Nash equilibrium, that satisfies [9].

$$f_i(x_i^*, \boldsymbol{x}_{-i}^*) \geq f_i(x_i, \boldsymbol{x}_{-i}^*) \text{ for } i = 1, \dots, N \tag{2}$$

In other words, if all agents except agent i hold the Nash equilibrium actions $\boldsymbol{x}_{-i}^*$, the action of agent i deviated from x_i^* will decrease its own objective function according to Eq. (1), which is the incentive for all the agents to hold the Nash equilibrium strategy $\boldsymbol{x}^* = (x_1^*, \dots, x_N^*)$. The operational condition that individual wind turbine maximizes its own power reflects the Nash equilibrium actions [1].

When all wind turbines coordinate their control actions to achieve the common goal of maximizing the wind farm power production, the cooperative wind farm control problem can be formulated as

$$\boldsymbol{x}^* = \underset{x}{\operatorname{argmax}} f(\boldsymbol{x}) \triangleq \sum_{i=1}^{N} f_i(\boldsymbol{x}) \tag{3}$$

where $f(\boldsymbol{x}) = \sum_{i=1}^{N} f_i(\boldsymbol{x})$ denotes as the total wind farm power function. While Eq. (1) is a decentralized control problem, Eq. (3) is a centralized control problem in that the control actions $\boldsymbol{x}$ of the entire wind turbines are determined simultaneously.

III. Bayesian Ascent Algorithm

For real-time, data-driven control, it is imperative that the control algorithm is designed to improve the target value using as few measurement data as possible. To achieve this goal, the Bayesian Ascent (BA) method is developed by incorporating into the Bayesian Optimization (BO) framework strategies to regulate the search region [7]. The following briefly describes the optimization procedure of the BA method, which will be used to solve the cooperative control problem posted in Eq. (3). Figure 1 illustrates the procedure of solving the cooperative control problem using the BA algorithm. BA algorithm iteratively searches the optimum control actions by executing a series of trial actions and observing the responses (total power productions) corresponding to the selected and executed control actions. Each iteration of the BA algorithm consists of a learning, an optimization and an observation phase.

A. Learning Phase

In the learning phase of the nth iteration, using the collected inputs $\boldsymbol{x}^{1:n} = \{\boldsymbol{x}^1, \dots, \boldsymbol{x}^n\}$ and the observed output data $\boldsymbol{y}^{1:n} = \{y^1, \dots, y^n\}$, the unknown objective function $f(\boldsymbol{x})$ is modelled using a Gaussian Process (GP) regression. Based on the prior and the observation model, the value $f^* = f(\boldsymbol{x}^*)$ of the target function for the unseen input $\boldsymbol{x}^*$ and the observed outputs $\boldsymbol{y}^{1:n} = \{y^1, \dots, y^n\}$ are assumed to follow a multivariate Gaussian distribution [10]:

$$\begin{bmatrix} \boldsymbol{y}^{1:n} \\ f^* \end{bmatrix} \sim N\left(\boldsymbol{0}, \begin{bmatrix} \mathbf{K} + \sigma_\epsilon^2 \mathbf{I} & \boldsymbol{k} \\ \boldsymbol{k}^T & k(\boldsymbol{x}^*, \boldsymbol{x}^*) \end{bmatrix}\right) \tag{4}$$

where $\mathbf{K}$ is the covariance matrix (kernel matrix) whose (i,j)th entry is defined as $\mathbf{K}_{ij} = k(\boldsymbol{x}^i, \boldsymbol{x}^j)$, and $\boldsymbol{k}^T = \left(k(\boldsymbol{x}^1, \boldsymbol{x}^*), \dots, k(\boldsymbol{x}^n, \boldsymbol{x}^*)\right)$. The value of the covariance function $k(\boldsymbol{x}^i, \boldsymbol{x}^j)$ quantifies the similarity between the two input vectors $\boldsymbol{x}^i$ and $\boldsymbol{x}^j$. We use a squared exponential covariance function whose evaluation between two input vectors $\boldsymbol{x}^i$ and $\boldsymbol{x}^j$ is expressed as [11]:

$$k(\boldsymbol{x}^i, \boldsymbol{x}^j) = \sigma_s^2 \exp\left(-\frac{1}{2}(\boldsymbol{x}^i - \boldsymbol{x}^j)^T \mathrm{diag}(\boldsymbol{\lambda})^{-2}(\boldsymbol{x}^i - \boldsymbol{x}^j)\right) \tag{5}$$

where σ_s and $\boldsymbol{\lambda}$ are termed hyper-parameters. The term σ_s^2 is referred to as the signal variance that quantifies the overall magnitude of the covariance value. With hyper parameters optimized, the posterior distribution on the response f^* for the unseen input $\boldsymbol{x}^*$ given the historical data $\boldsymbol{D}^n = \{(\boldsymbol{x}^i, y^i) | i = 1, \dots, n\}$ can be expressed as an 1-D Gaussian distribution $f^* \sim N\left(\mu(\boldsymbol{x}^*|\boldsymbol{D}^n), \sigma^2(\boldsymbol{x}^*|\boldsymbol{D}^n)\right)$ with the mean and variance functions expressed, respectively, as [10]:

$$\mu(\boldsymbol{x}^*|\boldsymbol{D}^n) = \boldsymbol{k}^T(\mathbf{K} + \sigma_\epsilon^2 \mathbf{I})^{-1} \boldsymbol{y}^{1:n} \tag{6}$$

$$\sigma^2(\boldsymbol{x}^*|\boldsymbol{D}^n) = k(\boldsymbol{x}^*, \boldsymbol{x}^*) - \boldsymbol{k}^T(\mathbf{K} + \sigma_\epsilon^2 \mathbf{I})^{-1} \boldsymbol{k} \tag{7}$$

Here, $\mu(\boldsymbol{x}^*|\boldsymbol{D}^n)$ and $\sigma^2(\boldsymbol{x}^*|\boldsymbol{D}^n)$ are used as the functions for evaluating, respectively, the mean and the variance of the hidden function output f^* corresponding to the unseen input data $\boldsymbol{x}^*$.

B. Optimization (Sampling)

In the nth iteration of the optimization phase, the next input $\boldsymbol{x}^{n+1}$ is determined by exploiting the learned target function $f(\boldsymbol{x})$ in order to *learn* more about the target function as well as to *improve* the target value at the same time. In Bayesian Optimization (BO), the next sampling point is being selected as one maximizing an acquisition function that incorporates both the aspects of exploration and the exploitation [5-6]. Likewise, the BA algorithm selects the next input as one that maximizes the expected improvement $\mathrm{E}[\max\{0, f(\boldsymbol{x}) - f^{max}\}]$, the acquisition function that has been proposed by [12]. Additionally, the BA algorithm imposes a trust region on the scope of the next sampling to ensure that the next input $\boldsymbol{x}^{n+1}$ is chosen near the best input observed so far in an attempt to monotonically increase the target value. This optimization phase of BA algorithm is posed as a constrained optimization problem described in [7]:

$$\begin{aligned} &\max_{\boldsymbol{x}}.\ \mathrm{E}[\max\{0, f(\boldsymbol{x}) - f^{max}\} \,|\boldsymbol{D}^n] \\ &\text{s.t. } \boldsymbol{x} \in \boldsymbol{T} \triangleq \{\boldsymbol{x} |\ \|x_i - x_i^{max}\|_2 < \tau_i \ \text{ for } \ i = 1, \dots, N\} \end{aligned} \tag{8}$$

where the trust region $\boldsymbol{T}$ is defined as a hypercube with its center being $\boldsymbol{x}^{max}$ that produces the maximum target value f^{max} with the historical data $\boldsymbol{D}^n$. The strategy employed in the BA algorithm is similar to imposing a trust region constraint in mathematical optimization [13]. The ith component τ_i of $\boldsymbol{\tau} = (\tau_1, \dots, \tau_i, \dots, \tau_N)$ determines the range where the ith component x_i of $\boldsymbol{x} = (x_1, \dots, x_i, \dots, x_N)$ is being sampled next. Thus, the vector $\boldsymbol{\tau}$ controls the overall size of the hypercube trust region where the exploration takes place.

C. Observation

In the observation phase, the selected control input $\boldsymbol{x}^{n+1}$ is executed and the corresponding output y^{n+1} is observed. The collected new data point $(\boldsymbol{x}^{n+1}, y^{n+1})$ is then appended to the historical data set as $\boldsymbol{D}^{n+1} = \{(\boldsymbol{x}^i, y^i) | i = 1, \dots, n+1\}$, which is then used to update the target function $f(\boldsymbol{x})$ and re-optimize the hyper-parameters in the learning phase of the next iteration. In addition, to expedite the convergence rates, BA adjusts the size of the trust region depending on the improvement in the target value.

IV. Wind Tunnel Experimental Setups

This section describes an experimental study to validate (1) the effectiveness of the cooperative wind farm control strategy for improving the total wind farm power and (2) the efficiency of the BA algorithm for finding the optimum coordinated control actions using only the power measurement data.

A. Scaled Wind Turbine

The scaled wind turbine model, shown in Figure 2, is made of three aluminum blades with a length of 70 cm. The rotor diameter is 150 cm. The tower is made of a steel tube with a height of 100 cm. The blade pitch angles are controlled by a servomotor (Dynamixel-64T). As shown in Figure 2(b), the rotation of the servomotor is transformed into a linear motion to rotate the blade angles through a mechanical linkage. The rotation angles of the servomotor range from 0° to 70° which convert the blade pitch angles varying from 0° to 20° (albeit they are not related in a linear fashion). We use the rotation of the servomotor, instead of the actual blade pitch angle, as the control variable for optimization. The rotational change of the servomotor is easy to track using the encoder in the servomotor, which is also used to acknowledge the executed control actions. As shown in Figure 2(b), the yaw angle is controlled by the same type of servomotor through a mechanical gear system. With an one-to-one gear ratio, the rotational angle of the servomotor is the same as the actual rotation of the yaw of the wind turbine. An AC generator, shown in Figure 2(b), is used to convert the mechanical energy into electrical energy.

B. Control Board

Figure 3 shows the circuit board designed to measure the electrical power output from the wind turbine and to execute the control actions to adjust the blade pitch and the yaw angle of the wind turbine. The AC voltage output from the generator is converted into DC voltage by the rectifier. The rectified voltage and the associated current flowing through the load resistance are then measured using voltage and current sensors, from which the instantaneous power is computed. The microcontroller (Arbotix-M) continuously samples the instantaneous power and compute the average power (using a moving average technique). The microcontroller then transmits the computed average power to the central node (laptop computer) through the XBee radio module every 2 minutes. The BA method processes the average power collected from the wind turbines in the central node and determines the next control actions. The determined

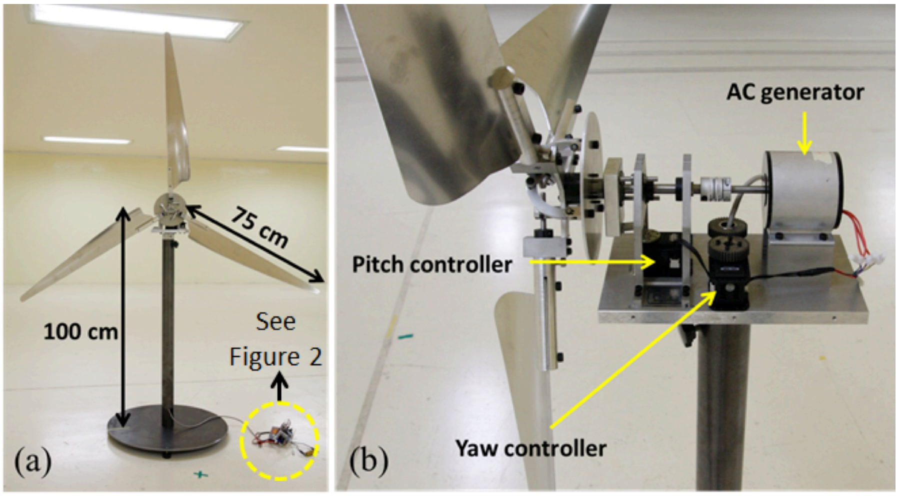


Figure 2. Scaled wind turbine model

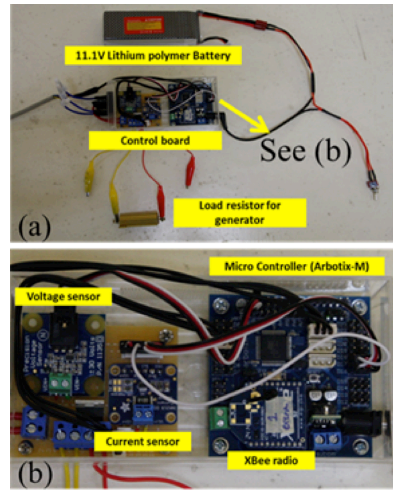


Figure 3. Control board

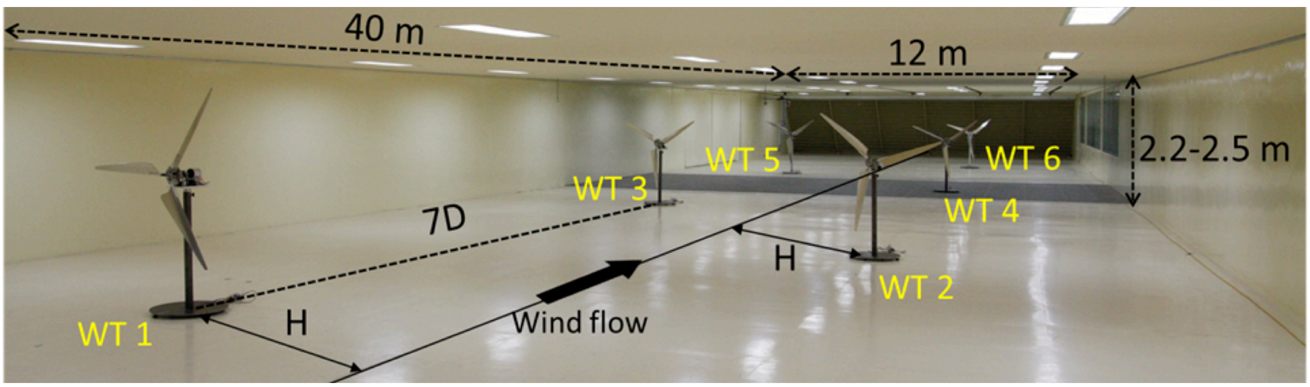


Figure 4. Wind tunnel section

control actions are then wirelessly transmitted to the microcontroller to change the blade pitch and the yaw angle in the wind turbine.

C. Wind Tunnel Laboratory

Figure 4 shows the 6-scaled wind turbines arranged in the test section of the KOCED's Wind Tunnel, located at Chonbuk National University in Korea. The wind tunnel test section is 12 m wide and 40 m long. The height of the test section is 2.2 m at the front and continuously increases until reaching 2.5 m at the end of the test section. Due to the varying height of the test section, the wind speed varies depending on the locations in the test section. The constant wind speed of 4 m/s (measured at 32m downside of the test section) is used throughout the experiments to ensure that the scaled wind turbines are operated safely without having excessive vibrations. The layout shown in Figure 4 is one example of the wind farm configuration used in the experimental studies.

D. Experimental Procedure

The effectiveness of the cooperative control with the BA algorithm is experimentally investigated by applying it to the scaled wind farm with different wind farm configurations and wind conditions. For each case, the wind turbines are placed at the designated locations, and the yaw angles of the wind turbines are set to be perpendicular to the wind direction, i.e., the initial yaw offset angle is always zero that is the optimum control actions for maximizing its own power production. In addition, to evaluate the performance of the cooperative control approach and the BA algorithm in terms of efficiency, two wind turbine powers, as defined below, are measured before employing the cooperative control strategy with the BA algorithm:

- P_i^F: Freestream maximum power of wind turbine i that can be produced at a given location when there is no wake interference. P_i^F for $i = 1, \dots, N$ is individually determined by iteratively changing the pitch angle of wind turbine i located at its designated position in the wind tunnel. The reason why we measure P_i^F for all wind turbines is that the wind flow conditions (i.e., wind speed and turbulence intensity) are different depending on the location of the wind turbine. The measured power P_i normalized by P_i^F then represents the power efficiency for wind turbine i. The total wind farm power is computed $\sum_{i=1}^{N} P_i^F$, where N is the number of wind turbines considered. The maximum total wind farm power will be used to normalize the results to show the relative improvement.
- P_i^G: Greedy maximum power of wind turbine i that can be produced at a given location when the upstream wind

turbines are producing their maximum powers. P_1^G for the first upstream wind turbine is same with P_1^F . For $i = 2, \dots, N, P_i^G$ is individually determined by iteratively changing the blade pitch angle of wind turbine i as the upstream wind turbines are operated with their greedy control strategy. The wind farm power efficiency for the greedy control strategy is then computed as $\sum_{i=1}^N P_i^G / \sum_{i=1}^N P_i^F$.

Starting from the identified greedy control strategy $\boldsymbol{x}^G = (x_1^G, \dots, x_N^G)$ that produces the greedy maximum power $\boldsymbol{P}^G = (P_1^G, \dots, P_N^G)$, the BA algorithm proceeds to find the optimum coordinated control actions $\boldsymbol{x}^C = (x_1^C, \dots, x_N^C)$ that maximizes the sum of the wind turbine powers $\sum_{i=1}^N P_i^C$ where $\boldsymbol{P}^C = (P_1^C, \dots, P_N^C)$ corresponds to wind turbine powers produced using $\boldsymbol{x}^C$. The wind farm power efficiency for the cooperative control strategy is then computed as $\sum_{i=1}^N P_i^C / \sum_{i=1}^N P_i^F$.

V. RESULTS

We study the effectiveness of the cooperation control and the capability of the BA algorithm for finding the optimum coordinated control actions. The improvement in the total wind farm power by the cooperative control strategy is quantified as a gain $(\sum_{i=1}^N P_i^C - \sum_{i=1}^N P_i^G) / \sum_{i=1}^N P_i^G$, which reflects the effectiveness of the cooperative control strategy comparing to the greedy control strategy. The convergence rate to the maximum cooperative power $\sum_{i=1}^N P_i^C$ reflects the performance of the BA algorithm. It should be noted that the exact optimum strategy for the cooperative control is not known. The goal of BA algorithm is to rapidly improve the wind farm power production compared to the initial greedy control strategy.

As shown in Figure 4, a total of 6 wind turbines are arranged in two lines separated by a lateral distance $2H$. The downstream inter distance between two wind turbines is fixed at $7D$ where D is the rotor diameter of a wind turbine. The BA algorithm is employed to three different wind farm configurations with $H = 0$ m, 1.5 m and 3 m. For each case, the blade pitch angle and the yaw offset angle of WT 6 is fixed at its greedy control strategy because WT 6 is the last wind turbine in the array. In total, 10 control variables, yaw and the blade pitch (servo) angles for wind turbines WT1~5, are optimized by the BA algorithm.

For the three cases, Figure 5 shows the trajectories of the control actions and the power efficiencies of the 6 wind turbines with the iterations of the BA algorithm. Figure 6 shows the improvements in the total wind farm power efficiency by the BA algorithm. In terms of the effectiveness of the cooperative control, the following trends are observed:

- As shown in Figure 5, the two upstream wind turbines WT 1 and WT 2 produce the powers that are comparable to the maximum free stream powers for the three cases, $H = 0$ m, 1.5 m and 3 m. When the cooperative control is employed, these two upstream wind turbines offset their yaw angles the largest (when comparing to other wind turbines) to increase the power productions of the downstream wind turbines as well as the total power production.
- While the yaw offset angles for WT 1 and WT 3 change in clockwise direction, WT 2 and WT 4 change in the counterclockwise direction. This result causes the wake to divert away from the downstream wind turbines. The blade pitch servo angles for the 5 wind turbines are also concurrently adjusted, but their common trend is a little bit difficult to observe because its influence on both their own powers and the powers of other wind turbines are not as strong as the influence of the yaw offset angles.

In terms of the effects of the wind direction on the capability of the BA algorithm, the following trends are observed:

- As shown in Figure 6, the lateral distance H does not dramatically influence the wind farm power efficiencies for both the greedy and the cooperative control strategies (compare the initial and the final wind farm power efficiencies for three cases). This implies that the cooperative control strategy would still be effective to a wind farm with its wind turbines being densely populated.
- When $H = 3$ m , the convergence rate of the BA algorithm is slower than the other cases. After a few explorations in the initial iterations, the BA algorithm improves the target wind farm power efficiency as much as the other cases.

VI. CONCLUSION

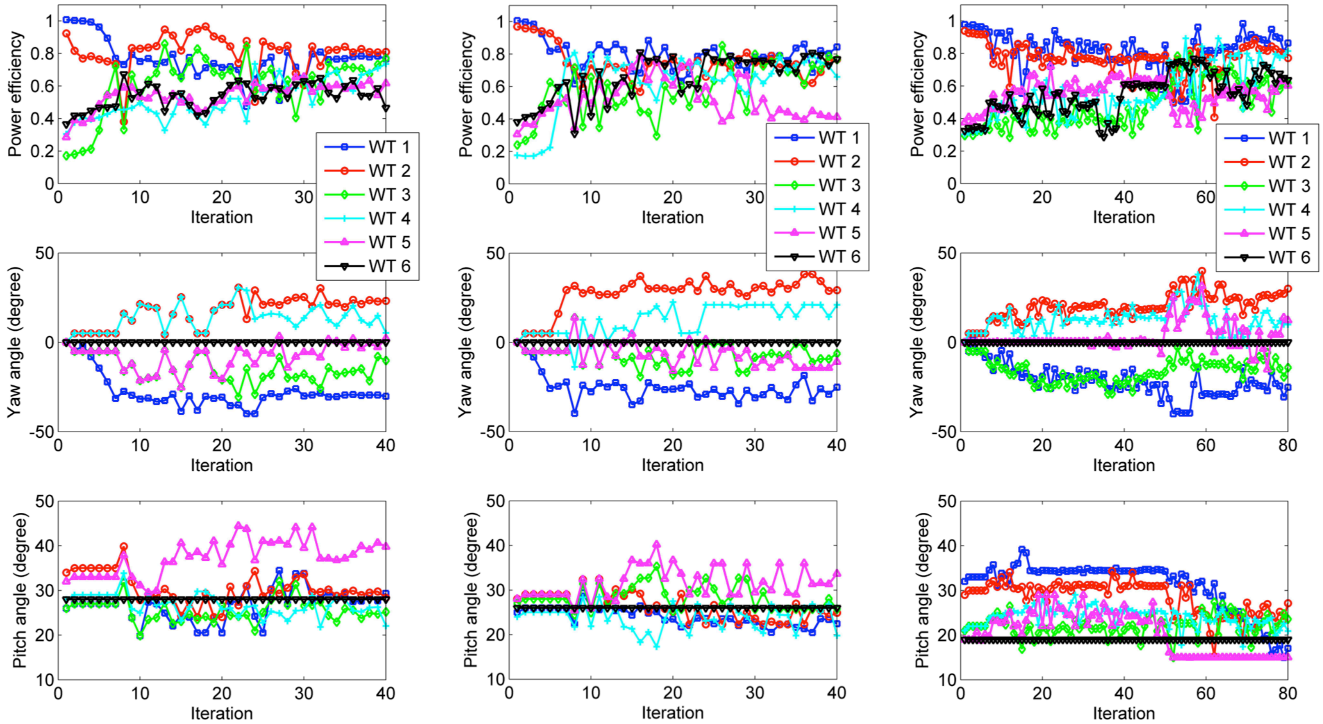
The data driven BA optimization algorithm iteratively finds the optimum of the target system by using the input (control actions of the wind turbines) and the output (power measurements from the wind turbines) data collected from the target (wind farm). Using 6 scaled wind turbines in a wind tunnel laboratory, this study experimentally investigates (1) the effectiveness of the cooperative control in improving the total wind farm power, and (2) the capability of BA algorithm in finding the optimum coordinated control actions using only the input and the resultant power measurement data. The BA algorithm is employed to determine optimum coordinated control actions of the wind turbines that maximize the total power production. Due to the trust region constraints, the BA algorithm increases almost monotonically the target wind farm power by gradually chaining the control actions of the wind turbines. Finally, it is generally unknown what the true optimum is for the cooperative control problem. The role of BA algorithm is not to identify the exact optimum, but to improve the wind farm power as quickly as possible compared to the initial conventional control strategy.

REFERENCES

[1] J. Park, and K. Law, "Cooperative wind turbine control for maximizing wind farm power using sequential convex programming," *Energy Conversion and Management*, 101, pp. 295-316, 2015a.

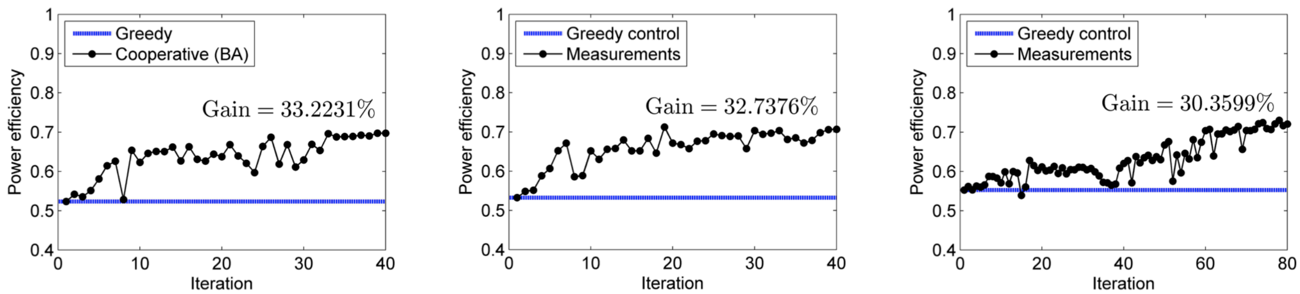
[2] J. Marden, S. Ruben, and L. Pao, "A model-free approach to wind farm control using game theoretic methods," *IEEE Transactions on Control Systems Technology*, 21(4), pp. 1207-1214, 2013

[3] P.M.O. Gebraad, F. Teeuwisse, J. W. Wingerden, P. Fleming, S. Ruden, J. Marden, and L. Pao, "Wind plant power optimization through yaw control using a parametric model for wake effects-a CFD simulation study," *Wind Energy*, DOI: 10.1002/we.1822, 2014.



(a) $H = 0$ m (b) $H = 1.5$ m (c) $H = 3$ m

Figure 5. Control actions and power efficiencies for different wind directions.



(a) $H = 0$ m (b) $H = 1.5$ m (c) $H = 3$ m

Figure 6. Improvement on power production using cooperative control for different wind directions.

[4] P.M.O. Gebraad, and J.W. Wingerden, "M aximum power-point tracking control for wind farms," *Wind Energy*, 18(3), pp. 429-447, 2015.

[5] E. Bronchu, M.V. Cora, and N. Freitas, *A tutorial on Bayesian optimization of expensive cost functions with application to active user modelling and hierarchical reinforcement learning*. Tech. Rep., Department of Computer Science, University of British Columbia, 2010.

[6] M. Osborne, *Bayesian Gaussian process for sequential prediction, optimisation and quadrature*. PhD dissertation, Department of Computer Science, University of Oxford, 2010.

[7] J. Park, and K. Law, "A Bayesian optimization approach for wind farm power maximization," *Proceedings of the SPIE Smart Structures/NDE Conference.* San Diego, CA, USA, 2015b.

[8] J. Park, and K. Law, "A data-driven, cooperative wind farm control to maximize the total power production" *Applied Energy*, submitted for publication.

[9] J. Nash, "Equilibrium points in n-person games," *Proceedings of the National Academy of Sciences*, 36(1), pp. 48–49, 1950.

[10] C.E. Rasmussen, and C.K. Williams, *Gaussian processes for machine learning*, MIT Press, Cambridge, MA, 2006.

[11] R.M. Neal, *Bayesian learning for neural networks*. Springer-Verlag, New York, NY, 1996.

[12] J. Mockus, A. Fretitas, and J.A. Castelanous, *Toward global optimization*. North-Holland, Amsterdam, 1978.

[13] J. Nocedal, and S. Wright, *Numerical Optimization. Springer*, New York, NY, 2000.