

Article

A Data-Driven, Cooperative Approach for Wind Farm Control: A Wind Tunnel Experimentation

Jinkyoo Park ^{1,*}, Soon-Duck Kwon ² and Kincho Law ³

¹ Department of Industrial and Systems Engineering, Korea Advanced Institute of Science and Technology, Daejeon 34141, Korea

² Department of Civil Engineering, Chonbuk National University, Jeonju 5896, Korea; sdkwon@chonbuk.edu

³ Department of Civil and Environmental Engineering, Stanford University, Stanford, CA 94305, USA; law@stanford.edu

* Correspondence: jinkyoo.park@kaist.ac.kr; Tel.: +82-42-350-3133

Received: 23 March 2017; Accepted: 16 June 2017; Published: 27 June 2017

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 between the wind turbines in a wind farm. Because of wake interference, such greedy control strategy can significantly lower the power production of the downstream wind turbines and, thus, reduce the overall wind farm power production. As an alternative to the greedy control strategy, we study a cooperative wind farm control strategy that determines and executes the optimum coordinated control actions for maximizing the total wind farm power production. To determine the optimum coordinated control actions of the wind turbines, we employ a data-driven optimization method that seeks to find the optimum control actions using only the power measurement data collected from the wind turbines in a wind farm. In particular, we employ the Bayesian Ascent (BA) algorithm, 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.

Keywords: wind farm control; Bayesian Ascent algorithm; wind tunnel experiment; data-driven optimization

1. Introduction

As modern wind turbines now allow for adjusting the blade angle, the yaw angle and the generator torque, this study investigates control strategies for maximizing the total power production of a wind farm. Conventionally, every individual wind turbine in a wind farm is operated to maximize its power production through adjusting its own operational conditions, i.e., control actions. Under this greedy control strategy, the wake formed by the upstream wind turbine would potentially lower the power production of the downstream wind turbines due to reduced wind speed and increased turbulence intensity inside the wake. The wake interaction among wind turbines has been experimentally studied [1–5]. Realizing that the interactions among the wind turbines can affect their power production, cooperative strategies that can coordinate the control actions to actively manipulate the wake interference pattern can increase the total power production of a wind farm. For example, the joint set of the induction factors for the wind turbines in a wind farm has been employed to optimize and to increase the total energy production of the wind farm [6]. In addition, the joint set of yaw-offset

angles of the wind turbines in a wind farm has also been used to maximize the total wind farm power production [7–9].

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 a mathematical optimization method. For example, Park and Law formulated the cooperative wind farm control problem using the wind farm power function calibrated using computational fluid dynamics (CFD) simulation data, and proposed to solve the control problem using sequential convex programming [10]. 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 [7,11], point tracking method [6], and closed loop wake detection controller [12] have been proposed to determine the optimum control actions using only 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 such model free control approaches strongly depends on the rate of convergence to an optimum and the robustness to deal with noisy data in optimizing the target system. In other words, a data driven control or optimization method should identify the optimum operational condition of a target system using a small set of measurement data that may be corrupted with noise.

Recently, efforts have been made to optimize a target system using scarce data by exploiting the expressivity of non-parametric regression models. For example, Bayesian Optimization (BO) iteratively determines the optimum of target system through a sequence of learning and sampling steps [13,14]. At each iteration, BO approximates the input and the output relationship of a target system using Gaussian Process (GP) regression (learning) and uses the approximated model to determine the next inputs that improve the target values (sampling). Park shows that using analytical wind arm power function for 5 wind turbines, the BA algorithm requires 50 iterations while the Genetic Algorithm (GA) requires 1000 iterations to reach a similar output [15,16]. Conducting an excessive number of trial actions is not desired for real implementation. To further expedite the convergence for finding the optimum control actions, Park and Law have developed the Bayesian Ascent (BA) method that combines the strengths of Bayesian Optimization and gradient-free trust region methods. BA adapts the strategy of regulating the optimization scope, as used in the trust region method, into a BO framework to ensure that the algorithm can monotonically increase a target value [17]. With 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 algorithm to determine the optimum coordinated control actions for wind farm power production by exploiting the power measurement data from the wind turbines. Numerical simulations using an analytically derived power function have demonstrated the potential of the BA algorithm for the control problem [17]. In this work, we focus on the implementation of the BA algorithm for real-time control of 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 optimum coordinated control actions for wind turbines (i.e., the optimum cooperative control strategy) using only the power measurement data. We employ the Bayesian Ascent algorithm to the scaled wind farm with different configurations and wind directions to gain insight on cooperative control using the BA algorithm.

This paper is organized as follows: First, the cooperative wind farm control problem is formulated using the concepts of game theory. The procedure of deriving the optimum control actions for the cooperative wind farm control using the Bayesian Ascent algorithm is discussed. The wind tunnel experimental setups for employing the cooperative control strategy with the BA algorithm are then

described in details. The results of the experimental studies are then presented. The paper concludes with a summary and a discussion on future works.

2. Formulation of Wind Farm Power Maximization Problem

Control strategies for wind farm power maximization depend on how the interactions among the wind turbines are modelled. The conventional approach is to have each wind turbine maximize its own power production without taking into consideration the effect on the other wind turbines. This greedy control strategy is analogous to a non-cooperative game. On the other hand, if the wind turbines cooperate with a common goal of maximizing the total wind farm power production, the control problem can be posed as a cooperative game. This section briefly describes the wind farm power maximization problem in the framework of the cooperative and non-cooperative games. For the formulation, we denote $f_i(\mathbf{x})$ as the power production of wind turbine i under the control actions $\mathbf{x} = (x_1, \dots, x_i, \dots, x_N)$ of the N wind turbines in the wind farm. The control action x_i of wind turbine i may include adjustments on the yaw angle, blade angle or the generator (i.e., x_i can be either a vector or a scalar depending on the number of control inputs for wind turbine i).

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

$$x_i^* = \operatorname{argmax}_{x_i} f_i(\mathbf{x}) = \operatorname{argmax}_{x_i} f_i(x_i, \mathbf{x}_{-i}) \text{ for } i = 1, \dots, N \quad (1)$$

in which wind turbine i maximizes its own power production $f_i(\mathbf{x})$ with respect to its own control action x_i regardless of the control actions $\mathbf{x}_{-i} = (x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_N)$ of the other wind turbines. For a non-cooperative game, there exists an equilibrium point $\mathbf{x}^* = (x_1^*, \dots, x_N^*)$, called the Nash equilibrium, that satisfies [18]:

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

In other words, if all wind turbines except wind turbine i hold the Nash equilibrium actions $\mathbf{x}_{-i}^*$, the control action of wind turbine i deviated from x_i^* will decrease its own power production according to Equation (1), which is the incentive for all the wind turbines to hold the Nash equilibrium strategy $\mathbf{x}^* = (x_1^*, \dots, x_N^*)$. The operational condition that individual wind turbine maximizes its own power production reflects a Nash equilibrium [17].

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

$$\begin{aligned} \operatorname{maximize}_{\mathbf{x}} f(\mathbf{x}) &\triangleq \sum_{i=1}^N f_i(\mathbf{x}) \\ \text{subject to } \mathbf{x}^l &\leq \mathbf{x} \leq \mathbf{x}^u \end{aligned} \quad (3)$$

where $f(\mathbf{x}) \triangleq \sum_{i=1}^N f_i(\mathbf{x})$ is defined as the wind farm power function. Furthermore, $\mathbf{x}^l$ and $\mathbf{x}^u$ are, respectively, the lower and the upper bounds on the wind turbine control actions for the N wind turbines. While Equation (1) is a decentralized control problem, Equation (3) is a centralized control problem in that the control actions $\mathbf{x}$ of all the wind turbines in the wind farm are determined simultaneously.

3. Bayesian Ascent Method

For real-time, data-driven wind farm control, it is imperative that the control algorithm is designed to improve the total power production of the target wind farm by executing as few trial actions (and observing the corresponding responses) as possible. To achieve this goal, the Bayesian Ascent method is developed by incorporating specific strategies to regulate the search region for the control actions into the Bayesian Optimization framework [17]. The following briefly describes the optimization

procedure using the BA algorithm, which will be used to solve the cooperative control problem posted in Equation (3).

In the data driven approach, the function $f(x)$ is assumed to relate the control actions x of all the wind turbines in a wind farm and the total wind farm power production $f = f(x)$. The expression for $f(x)$ is not known; only the power measurement y is obtained and used to infer about the target function $f(x)$. Figure 1 illustrates the procedure for solving the cooperative control problem using the BA algorithm. The BA algorithm iteratively searches the optimum control actions for all the wind turbines by executing a series of trial control actions for all the wind turbines and observing the corresponding wind farm power production. In other words, the learning about the target wind farm power function and the optimization procedure to select the next control actions for all the wind turbines are interwoven. Specifically, as shown in Figure 1, the following three steps are conducted at the n th iteration:

- **Learning:** the unknown wind farm power function $f(x)$ is modelled by Gaussian Process regression using the historical data $\mathcal{D}^n = \{(x^1, y^1), \dots, (x^i, y^i), \dots, (x^n, y^n)\}$, where $x^i = (x_1^i, \dots, x_N^i)$ is the control actions for all the N wind turbines in a wind farm at the i th iteration, and y^i is the corresponding total wind farm power output (i.e., the sum of the power outputs from all the N wind turbines) measured at the i th iteration. GP regression can model the target function $f(x)$ probabilistically and thus can quantify the uncertainty in the target value. Uncertainty quantification plays a crucial role when we determining the next control input online. In addition, as a non-parametric regression model, GP regression can model complex target function without assuming the basis function of the target function while using a small number of data points.
- **Optimization:** the next trial control actions x^{n+1} for all the N wind turbines are determined.
- **Observation:** the next trial control actions x^{n+1} are executed and the corresponding total wind farm power output y^{n+1} is obtained. The pair of the input and the output (x^{n+1}, y^{n+1}) will then be used to update the regression model for the target function $f(x)$ in the next iteration.

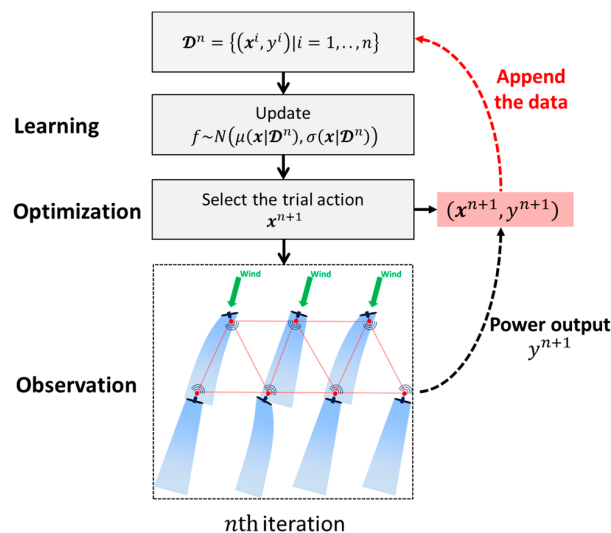


Figure 1. Bayesian optimization for wind farm power maximization problem.

The BA algorithm attempts to improve the total wind farm power production by executing the least number of trial actions.

3.1. Learning

In the learning phase of the n th iteration for BA, we have the n different combinations of control actions $x^{1:n} = \{x^1, \dots, x^n\}$ executed and the corresponding n observations on the total wind farm

power production $\mathbf{y}^{1:n} = \{y^1, \dots, y^n\}$. Using this historical data $\mathbf{D}^n = \{(\mathbf{x}^i, y^i) | i = 1, \dots, n\}$, the unknown target wind farm power function $f(\mathbf{x})$ is modelled using a Gaussian Process (GP) regression.

In GP regression, the true function values $\mathbf{f}^{1:n} = \{f^1, \dots, f^n\} = \{f(\mathbf{x}^1), \dots, f(\mathbf{x}^n)\}$ corresponding to the n sequence of trial actions $\mathbf{x}^{1:n} = \{\mathbf{x}^1, \dots, \mathbf{x}^n\}$ are treated as latent random variables. The unknown function values $\mathbf{f}^{1:n}$ are inferred from the observed sequence of wind farm power outputs $\mathbf{y}^{1:n} = \{y^1, \dots, y^n\}$. GP regression puts prior on the function values $\mathbf{f}^{1:n}$ using Gaussian Process as $p(\mathbf{f}^{1:n}) = GP(m(\cdot), k(\cdot, \cdot))$, where $m(\cdot)$ and $k(\cdot, \cdot)$ are, respectively, the mean and the covariance function [19]. In addition, GP regression assumes that the observation of wind farm power output y^i is measured with Gaussian noise, i.e., $y^i = f^i + \epsilon^i$ with $\epsilon^i \sim N(0, \sigma_\epsilon^2)$, where σ_ϵ^2 is the variance for the noisy component that is assumed to exist in the observation. Based on the prior and the noise (i.e., observation) model, the total wind farm power output $f(=f(\mathbf{x}))$ for the unseen control actions $\mathbf{x}$ and the observed wind farm power outputs $\mathbf{y}^{1:n} = \{y^1, \dots, y^n\}$ follow a multivariate Gaussian distribution [19]:

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

where $\mathbf{K}$ is the covariance matrix (kernel matrix) whose (i, j) th entry is $\mathbf{K}_{ij} = k(\mathbf{x}^i, \mathbf{x}^j)$, and $\mathbf{k}^T = (k(\mathbf{x}^1, \mathbf{x}), \dots, k(\mathbf{x}^n, \mathbf{x}))$. The value of the covariance function $k(\mathbf{x}^i, \mathbf{x}^j)$ quantifies the similarity between the two control actions $\mathbf{x}^i$ and $\mathbf{x}^j$; the more the two control actions differ, the closer the value of the covariance becomes zero, meaning that they are not correlated in terms of their function values. The noise variance σ_ϵ^2 in Equation (4) is estimated to quantify the level of noise in the observed wind farm power outputs $\mathbf{y}^{1:n} = \{y^1, \dots, y^n\}$.

We use a squared exponential covariance function whose evaluation between the two control inputs $\mathbf{x}^i$ and $\mathbf{x}^j$ is expressed as [20,21]:

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

which is described by the hyper-parameters σ_s and $\boldsymbol{\lambda}$. We have used this covariance function because we believe that the target wind farm power function smoothly varies with the control inputs. The term σ_s^2 is referred to as the signal variance that quantifies the overall magnitude of the covariance value. The hyper-parameters $\boldsymbol{\lambda} = (\lambda_1, \dots, \lambda_N)$ is referred to as the characteristic length scales to quantify the relevancy of the input features in $\mathbf{x}$ for predicting the response y . The noise variance σ_ϵ^2 in Equation (4) is often treated together with σ_s and $\boldsymbol{\lambda}$ as hyperparameters, i.e., $\boldsymbol{\theta} = \{\sigma_s, \boldsymbol{\lambda}, \sigma_\epsilon\}$. The optimum values for the hyper-parameters are determined as ones maximizing the marginal log-likelihood of the training data $\mathbf{D}^n = \{(\mathbf{x}^i, y^i) | i = 1, \dots, n\}$ as [19]:

$$\begin{aligned} \boldsymbol{\theta}^* &= \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \log p(\mathbf{y}^{1:n} | \mathbf{X}^n, \boldsymbol{\theta}) \\ &= \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \left(-\frac{1}{2}(\mathbf{y}^{1:n})^T (\mathbf{K} + \sigma_\epsilon^2 \mathbf{I})^{-1} \mathbf{y}^{1:n} - \frac{1}{2} \log |\mathbf{K} + \sigma_\epsilon^2 \mathbf{I}| - \frac{n}{2} \log 2\pi \right) \end{aligned} \quad (6)$$

With the hyper parameters optimized, the posterior distribution on the wind farm power output f for the unseen control actions $\mathbf{x}$ for all the wind turbines given the historical data $\mathbf{D}^n = \{(\mathbf{x}^i, y^i) | i = 1, \dots, n\}$ can be expressed as an 1-D Gaussian distribution $f \sim N(\mu(\mathbf{x} | \mathbf{D}^n), \sigma^2(\mathbf{x} | \mathbf{D}^n))$ with the mean and variance functions expressed, respectively, as [19]:

$$\mu(\mathbf{x} | \mathbf{D}^n) = \mathbf{k}^T (\mathbf{K} + \sigma_\epsilon^2 \mathbf{I})^{-1} \mathbf{y}^{1:n} \quad (7)$$

$$\sigma^2(\mathbf{x} | \mathbf{D}^n) = k(\mathbf{x}, \mathbf{x}) - \mathbf{k}^T (\mathbf{K} + \sigma_\epsilon^2 \mathbf{I})^{-1} \mathbf{k} \quad (8)$$

Here, $\mu(x|D^n)$ and $\sigma^2(x|D^n)$ are treated as the functions for evaluating, respectively, the mean and the variance of the hidden wind farm power output f corresponding to the unseen control input x . Thus, the unknown target function $f(x)$ is probabilistically represented as $f \sim N(\mu(x|D^n), \sigma^2(x|D^n))$.

3.2. Optimization

In the optimization phase of n th iteration for BA, the next control actions x^{n+1} for all the wind turbines are determined to learn more about the unknown target wind farm power function as well as to improve the wind farm power output at the same time. In Bayesian Optimization, the next control input is being selected as one maximizing an acquisition function that incorporates both the aspects of exploration and the exploitation [13,14]. Likewise, the BA algorithm selects the next set of control actions as one that maximizes the expected improvement $E[\max\{0, f - f^{max}\}]$, the acquisition function that has been proposed by Mockus et al. [21]. Additionally, the BA algorithm imposes the trust region on the scope of next set of control actions to ensure that the control actions x^{n+1} is chosen near the best control actions $x^{max} = (x_1^{max}, \dots, x_N^{max})$ observed so far in an attempt to monotonically increase the total wind farm power output. This optimization phase of BA algorithm is posed as a constrained optimization problem described as [17]:

$$\begin{aligned} & \underset{x}{\text{maximize}} \ E[\max\{0, f(x) - f^{max}\} | D^n] \\ & \text{subject to } x^l \leq x \leq x^u \\ & \quad x \in T \triangleq \{x \mid \|x_i - x_i^{max}\|_2 < \tau_i \text{ for } i = 1, \dots, m\} \end{aligned} \quad (9)$$

The BA algorithm aims to find the input x that further improve the target function $f(x)$, which is modelled as a random variable, comparing to the maximum target output f^{max} estimated so far. The trust region T is defined as a hypercube with its center being x^{max} that produces the maximum wind farm power output $f^{max} = \max_{\{x \in x^{1:n}\}} \mu(x|D^n)$. The i th component τ_i of $\tau = (\tau_1, \dots, \tau_i, \dots, \tau_N)$ determines the range where the i th component x_i of $x = (x_1, \dots, x_i, \dots, x_N)$ is being chosen next. Thus, the vector τ controls the overall size of the hypercube trust region where the exploration takes place. The strategy employed in the BA algorithm is similar to imposing a trust region constraint in mathematical optimization [22].

3.3. Observation

In the observation phase, the selected control actions x^{n+1} for all the wind turbines are executed and the corresponding wind farm power output y^{n+1} is observed. The new data point collected (x^{n+1}, y^{n+1}) is then appended to the historical data set as $D^{n+1} = \{(x^i, y^i) \mid i = 1, \dots, n+1\}$, which is then used to update the regression model for the target wind farm power function $f(x)$ in the learning phase of the next iteration.

Algorithm 1 summarizes the BA algorithm [17]. In addition to expedite the convergence rates, the BA algorithm adjusts the size of the trust region depending on the improvement observed in the wind farm power output. Let's denote the solution of Equation (9) as x^{n+1} . The BA algorithm checks whether x^{n+1} sufficiently improves the target wind farm power production. With the measured wind farm power output y^{n+1} , if the observed increase $y^{n+1} - f^{max}$ with respect to the previously estimated maximum target wind farm power f^{max} is larger than a certain threshold ratio γ of the average increase $(1/n)(f^{max} - y^1)$, where y^1 is the initial measurement of the wind farm power output, the control actions x^{n+1} will be updated as the best control actions x^{max} observed so far, and the trust region is expanded as $\tau^{n+1} = \beta \tau^n$, with $\beta > 1$ to expedite the convergence rate. Otherwise, the trust region will be reset as $\tau^{n+1} = \tau^1$, where τ^1 is the size of initial trust region. Imposing a trust region and adjusting its size are to ensure a monotonic increase in the wind farm power production and gradual convergence to an optimum with a high probability.

Note that there is not explicitly defined convergence criterion. The algorithm will continuously suggest the best control inputs, while adjusting the hyper-parameters for the GP regression model with newly observed data. The update of the data-driven control policy based on the Bayesian framework is one of the major advantages of the BA algorithm.

Algorithm 1. Bayesian Ascent (BA) algorithm.

Choose $\mathbf{x}^1$ and τ^1 (initial trust region size) and observe y^1

Repeat until convergence, $n = 1, 2, 3 \dots$

1: Optimize the hyper parameters:

$$\theta^* = \underset{\theta}{\operatorname{argmax}} \log p(\mathbf{y}^{1:n} | \theta)$$

2: Construct a GP regression to approximate $f(\mathbf{x})$:

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

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

3: Select the next input by solving:

$$\mathbf{x}^{n+1} = \underset{\mathbf{x} \in A \cap T}{\operatorname{argmax}} \mathbb{E}[\max\{0, f(\mathbf{x}) - f^{max}\} | \mathbf{D}^n]$$

$$\text{where } A := \left\{ \mathbf{x} \mid \mathbf{x}^l \leq \mathbf{x} \leq \mathbf{x}^u \right\}$$

$$T := \left\{ \mathbf{x} \mid \|x_i - x_i^{max}\| < \tau_i^n \text{ for } i = 1, \dots, N \right\}$$

4: Append the data $\mathbf{D}^{n+1} = \left\{ (\mathbf{x}^i, y^i) \mid i = 1, \dots, n+1 \right\}$

5: Update the size of trust region:

6: **if** $y^{n+1} - f^{max} \geq \gamma(1/n)(f^{max} - y^1)$ **then**

7: $\tau^{n+1} = \beta \tau^n$, ($\beta > 1$)

8: **else**

9: $\tau^{n+1} = \tau^1$

10: **end if**

4. Wind Tunnel Experimental Setups

Wind tunnel experiments with 6 scaled wind turbines are designed to validate (1) the effectiveness of the cooperative wind farm control strategy for improving total wind farm power; and (2) the capability of the BA algorithm for finding the optimum coordinated control actions using only the power measurement data observed from the wind turbines. This section describes the experimental setups and procedure.

4.1. Wind Turbine Model and Control Mechanisms

The scaled wind turbine, 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, ROBOTIS INC, Lake Forest, CA, USA). As shown in Figure 2b, 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 correspond to 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 2b, the yaw angle is controlled by the same type of servomotor through a mechanical gear system. With a 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. The angles of the two servomotors are then defined as the control actions of each wind turbine. The range of the first servomotor (for blade pitch control) is from 0° to 70° and the range of the second servomotor (for yaw control) is -40° to 40° , which are incorporated to the constraint on the control inputs $\mathbf{x}^l \leq \mathbf{x} \leq \mathbf{x}^u$. An AC generator, shown in Figure 2b, is used to convert the mechanical energy into electrical energy.

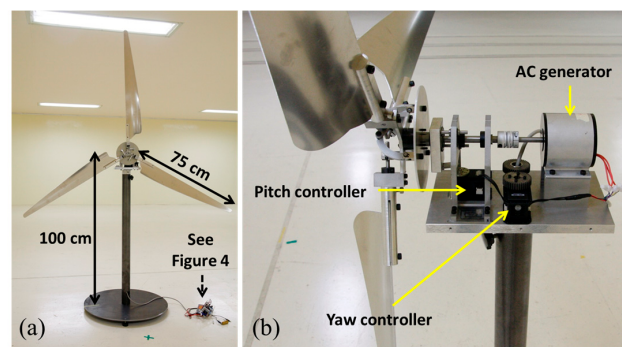


Figure 2. Scaled wind turbine model, (a) dimensions of the wind turbine; (b) wind turbine control controller.

Figure 3 shows the local control unit designed to measure the electrical power output from the wind turbine and to execute the control actions to adjust the blade and yaw angles 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, ROBOTIS INC, Lake Forest, CA, USA) continuously samples the instantaneous power and compute the average power (using a moving average technique). The microcontroller then transmits the computed average power over a one-minute time interval to the central control unit, shown in Figure 4, via the XBee radio. The central controller collects the averaged powers from the local units and determines the next control actions by executing the BA algorithm. The control actions determined are then wirelessly transmitted to each wind turbine through the central XBee node shown in Figure 4b. Upon receiving the new control actions, the local control unit changes the yaw and the blade pitch angles. After one minute of stabilizing time, the time that is required for the disturbed wake to be stabilized after the changes in the operational condition of the wind turbines, each local control unit starts to measure the power production by the wind turbine again.

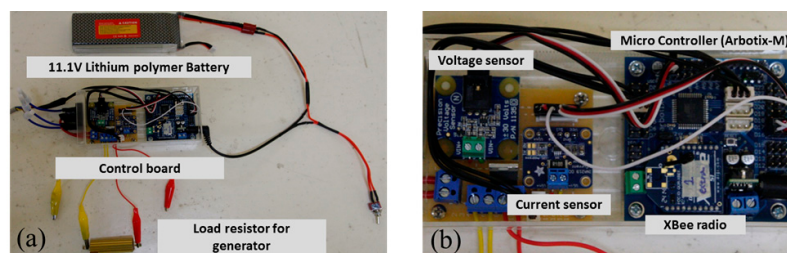


Figure 3. Local node controlling servomotors and measuring the power output, (a) Configuration of electrical control board; (b) Control board and power monitoring circuit.

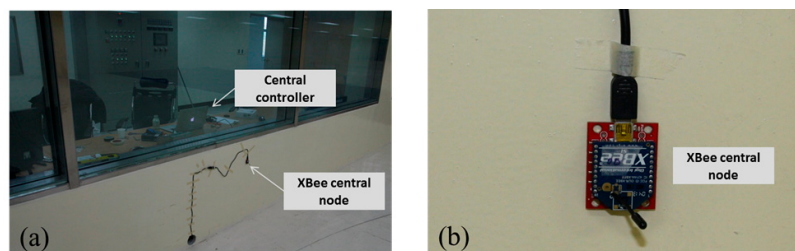


Figure 4. Central node hosting the Bayesian Ascent (BA) algorithm, (a) Location of central controller; (b) Xbee central node.

4.2. Wind Tunnel Experiments

Figure 5 shows six scaled wind turbines arranged in the test section of the KOCED 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 increases to 2.5 m at the end of the test section allowing longitudinal pressure gradient to be zero. Due to the varying height of the test section, the wind speed also varies along the test section. A constant wind speed of 4 m/s (measured at a distance of 32 m from the front 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 5 is one example of the wind farm configurations used in the experimental studies.

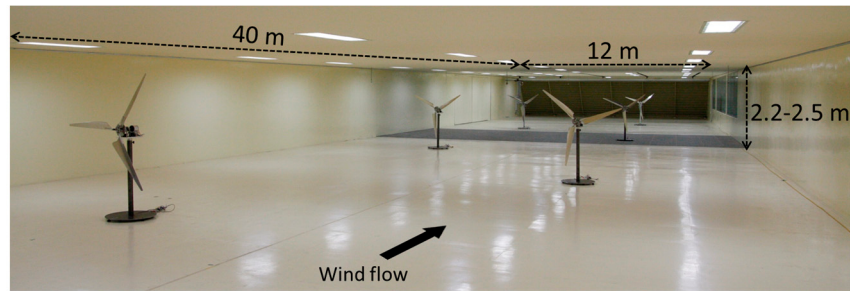


Figure 5. Wind tunnel section.

Cooperative control using the BA algorithm is applied to the scaled wind farm with different arrangements of the wind turbines and different wind conditions. For each arrangement, the yaw angles of the wind turbines are initially set to be perpendicular to the wind direction, which corresponds to the optimum control actions for a wind turbine to maximize its own power production. In addition, for comparison of performance efficiency, we measure two power parameters by a single wind turbine:

- 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 of all the wind turbines is that the wind flow conditions (i.e., wind speed and turbulence intensity) are different depending on the location in the wind turbine. The measured power P_i normalized by P_i^F then represents the power efficiency for the 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 cooperative control results for comparing the relative improvement over the greedy control strategy.
- 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 as P_1^F . For a downstream wind turbine i , $i = 2, \dots, N$, P_i^G is individually determined by iteratively changing the blade pitch angle of wind turbine i when 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 $\mathbf{x}^G = (x_1^G, \dots, x_N^G)$ that produces the greedy maximum power $\mathbf{P}^G = (P_1^G, \dots, P_N^G)$, the BA algorithm proceeds to find the optimum coordinated control actions $\mathbf{x}^C = (x_1^C, \dots, x_N^C)$ that maximizes the sum of powers produced by the wind turbines. Denoting the powers produced by the coordinated actions $\mathbf{x}^C$ as $\mathbf{P}^C = (P_1^C, \dots, P_N^C)$, the wind farm power efficiency for the cooperative control strategy is then computed as $\sum_{i=1}^N P_i^C / \sum_{i=1}^N P_i^F$.

5. Wind Tunnel Experiment Results

For different numbers of wind turbines and wind directions, we study the effectiveness of cooperative control by the BA algorithm for finding the optimum coordinated control actions. The improvement in the total wind farm power by the cooperative control strategy can be quantified as the relative gain $\left(\sum_{i=1}^N P_i^C - \sum_{i=1}^N P_i^G\right) / \sum_{i=1}^N P_i^G$, when 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. Starting initially with the greedy control strategy, the goal of BA algorithm is to improve the wind farm power production by cooperatively adjusting the wind turbines for a common gain.

5.1. Scenario 1: Effects of the Number of Wind Turbines

Scenario 1 is designed to study the influence of the number of wind turbines (i.e., dimension of the optimization variables) on the effectiveness of the cooperative control and the convergence rate of the BA algorithm. We consider 2, 3 and 4 wind turbines (i.e., $N = 2, 3$ and 4) when the wind turbines are initially facing the wind direction $\theta^W = 0^\circ$, as shown in Figure 6. For each case, the last downstream wind turbine operates at its greedy control actions since the deviation from the greedy control actions would only decrease power production of the wind turbine.

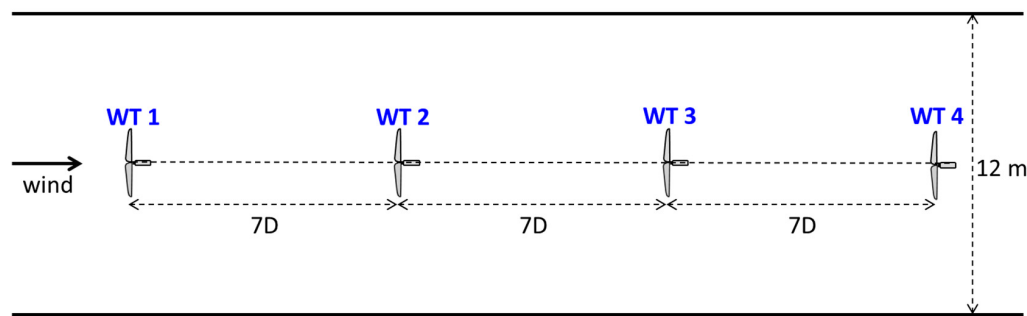


Figure 6. Influence of the number of wind turbines.

Figure 7 shows the trajectories of individual power efficiency P_i / P_i^F and the associated control actions of the wind turbines as the BA algorithm executes the cooperative actions. In addition, Figure 8 shows the improvement in the total wind farm power efficiency obtained by the BA algorithm compared to the efficiency of the greedy control strategy. In terms of the effectiveness of the cooperative control, the following trends are observed:

- As shown in Figure 7, for cooperative control, the wind turbines collectively adjust their control actions determined by the BA algorithm to increase the total wind farm power production.
- As shown in Figure 7, the cooperative control actions lower the power production for the first upstream wind turbine but significantly increase the power production of the downstream wind turbines. As a result, power produced by the wind turbines are more evenly distributed after the coordination, which could be beneficial for maintaining the wind turbines since the electrical and structural components experience similar level of loads.
- As the number of wind turbines increases, as shown in Figure 8, the initial (greedy) wind farm power efficiency decreases because more downstream wind turbines are affected by wake interference. Furthermore, as the number of wind turbines increases, the relative improvement by the cooperative control actions increases.

In terms of the capability of the BA algorithm to improve the target wind farm power, the following trends are observed:

- As shown in Figure 8, BA increases the wind farm power efficiency almost monotonically by gradually changing the control actions of the wind turbine.
- As the number of the wind turbine considered for cooperative control increases, more iterations are required for the BA algorithm to converge on the (local) optimum.

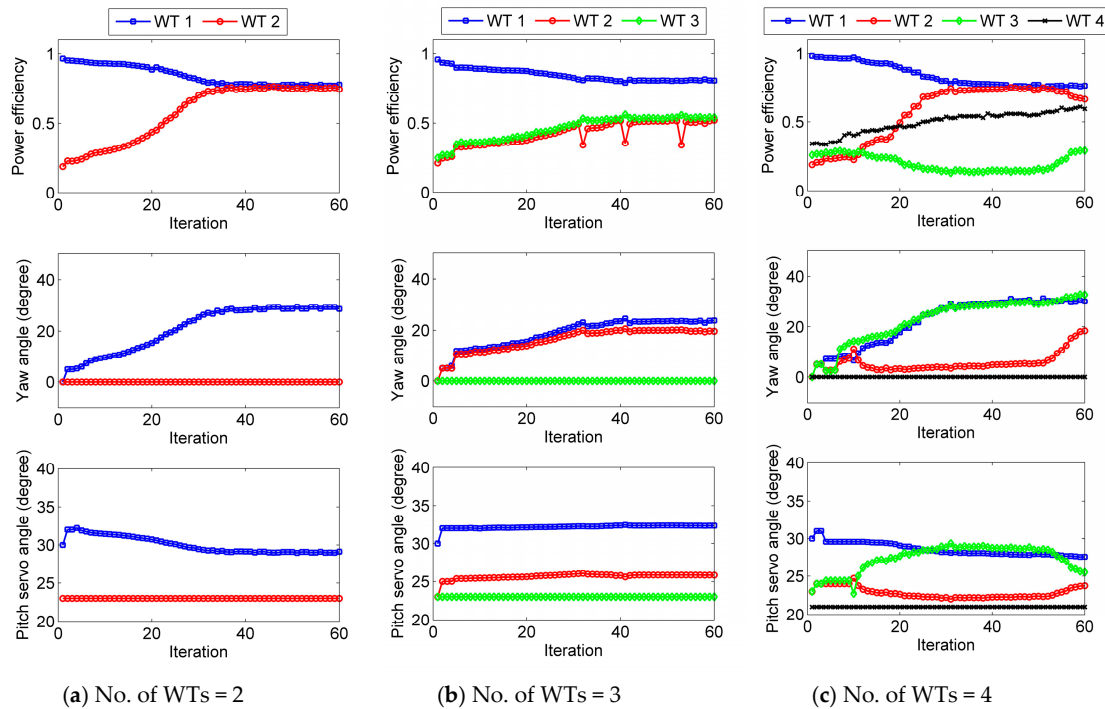


Figure 7. Control actions and power efficiencies for different number of wind turbines (WTs).

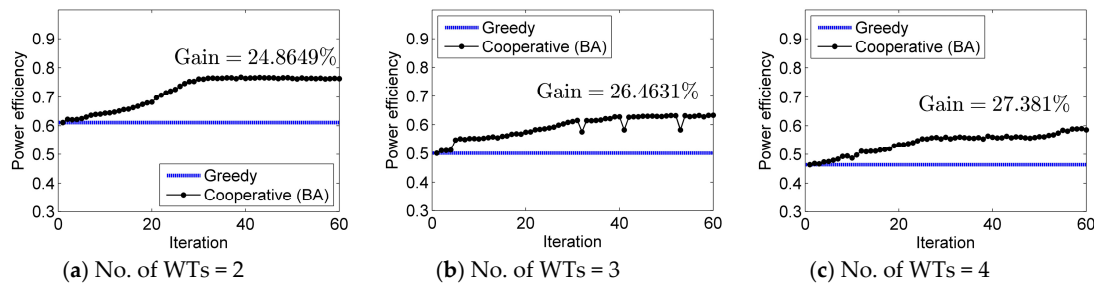


Figure 8. Improvement on power production using cooperative control for different number of wind turbines (WTs).

5.2. Scenario 2: Influences of the Wind Direction

Scenario 2 is designed to study the influence of the wind direction on the effectiveness of the cooperative control and the convergence rate of the BA algorithm. Since the wind flow direction is fixed in the wind tunnel, as shown in Figure 9, the wind direction is emulated by changing the relative locations of the wind turbines. The yaw angles of the wind turbines are initially set to be perpendicular to the wind direction. The BA algorithm is employed to optimize the coordinated control actions of 4 wind turbines for three cases of wind directions, $\theta^W = 0^\circ$, $\theta^W = 3^\circ$ and $\theta^W = 6^\circ$. For each case, the last downstream wind turbine operates at its greedy control actions since any deviation from the greedy control actions only decreases the power production of the wind turbine.

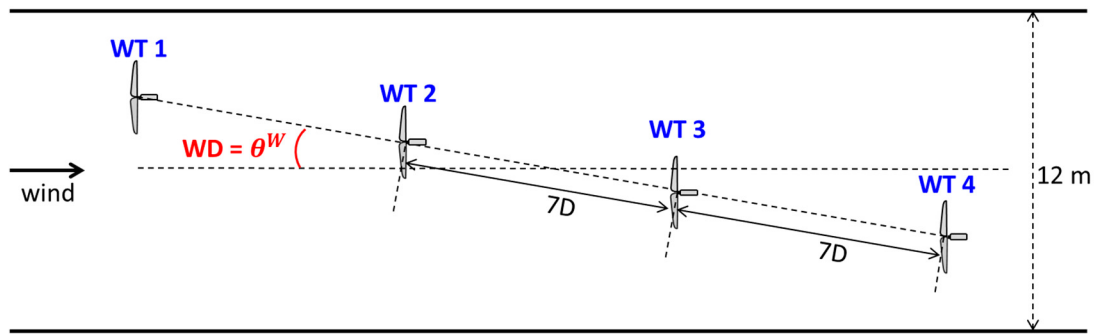


Figure 9. Influence of the number of wind turbines.

In terms of the effectiveness of the cooperative control, the following trends are observed:

- As shown in Figure 10a, when the wind direction θ^W is set at 0° where the wake is perfectly aligned with the wind turbine array, the initial power production by the downstream wind turbines, WT 2, WT 3 and WT 4, is only a small fraction of the power produced by the upstream wind turbine in the front. As the wind direction θ^W deviates from 0° , the downstream wind turbines are affected less by the wakes formed by the upstream wind turbines and produce more power initially, as shown in Figure 10b,c.
- As shown in Figure 11, the wind farm power efficiency with the greedy control strategy increases as the wind direction deviates from 0° . For example, when $\theta^W = 6^\circ$, the wind farm power efficiency is above 80% even before executing the cooperative control strategy. When the cooperative control strategy is employed, the wind farm power efficiency further increases. Note that the differences between the optimum power output and the initial (greedy) power production of the four wind turbines become smaller as the wind direction θ^W increases.

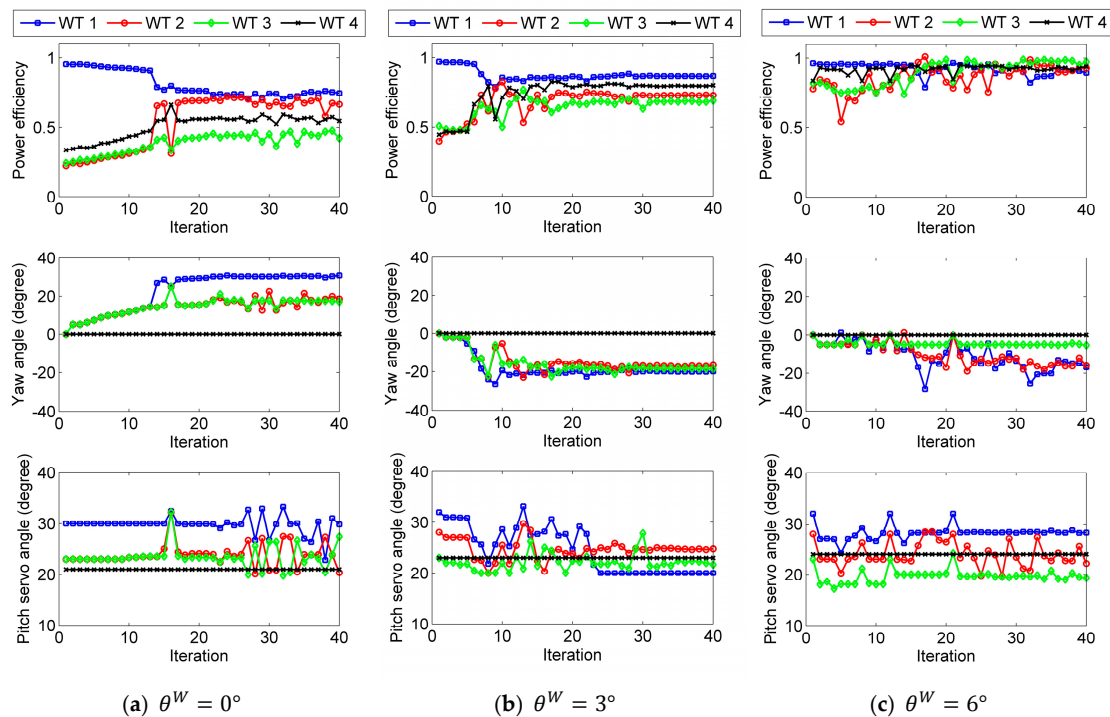


Figure 10. Control actions and power efficiencies for different wind directions.

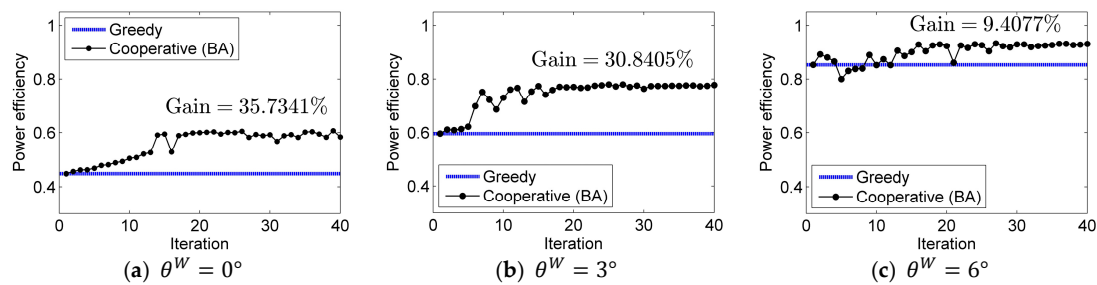


Figure 11. Improvement on power production using cooperative control for different wind directions.

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

- As in Scenario 1, the BA algorithm effectively increases the wind farm power efficiency using a small number of trial actions.
- When $\theta^W = 0^\circ$, the direction of yaw offset angle is not critical to improve the total wind farm power. However, when $\theta^W = 3^\circ$ and $\theta^W = 6^\circ$, only the yaw offset angle in counterclockwise (negative) can increase the wind powers of the downstream wind turbines due to asymmetric configuration of the wind turbine layout. If the yaw-offset angle of the upstream wind turbine is imposed in the opposite direction, the wake generated by the upstream wind turbine will be skewed toward the downstream wind turbines, which will reduce the powers of the downstream wind turbines.
- When $\theta^W = 6^\circ$, more iterations are required for the BA algorithm to converge on the local optimum. This is because the effects of changing control actions on the total wind farm power is not significant as for the cases $\theta^W = 0^\circ$ and 3° . Such small increases in the control effects make it difficult for the BA algorithm to reliably estimate the ascending direction.

5.3. Scenario 3: Grid Wind Farm Layout

Scenario 3 is designed to study the effectiveness of the cooperative control and the BA algorithm when the wind turbines are arranged in a grid configuration. As shown in Figure 12, a total of 6 wind turbines are arranged in two lines separated by a lateral distance $2H$. The downstream inter distance between the two wind turbines is fixed at $7D$. The BA algorithm is employed to maximize the total power production for 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 WT 1–5, are optimized by the BA algorithm. The BA algorithm converges slower for $H = 3$ possibly because, in this case, the influence of the control actions on the other wind turbines are weaker comparing to the other cases.

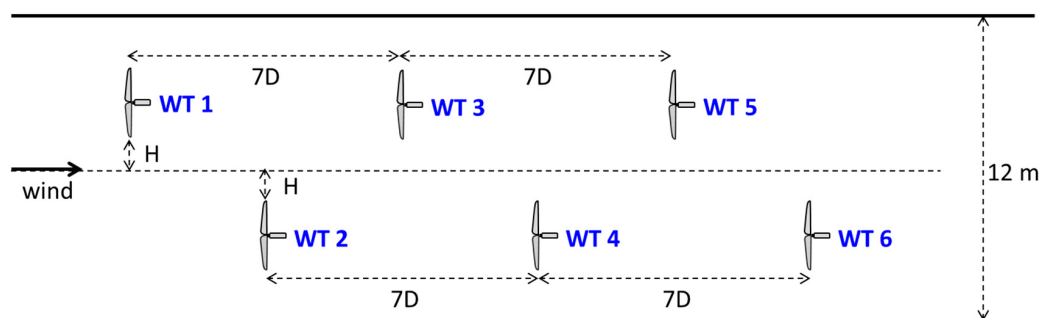


Figure 12. Influence of the number of wind turbines.

For the three cases considered, Figure 13 shows the trajectories of the control actions and the power efficiencies of the 6 wind turbines. Figure 14 shows the improvements for the total wind farm power efficiency obtained by executing the BA algorithm. In terms of the effectiveness of the cooperative control, the following trends are observed:

- As shown in Figure 13, the two upstream wind turbines WT 1 and WT 2, shown in Figure 12, produce the powers that are comparable to the maximum free stream powers. When the cooperative control is employed, these two upstream wind turbines offset their yaw angles the most (compared to other wind turbines) so that the power production of the downstream wind turbines, and thus the total power production, increases.
- It is interesting to note that WT 1 and WT 3 offset the yaw angles in a clockwise direction while WT 2 and WT 4 offset in counter clockwise direction such that the wakes are diverted away from the downstream wind turbines.

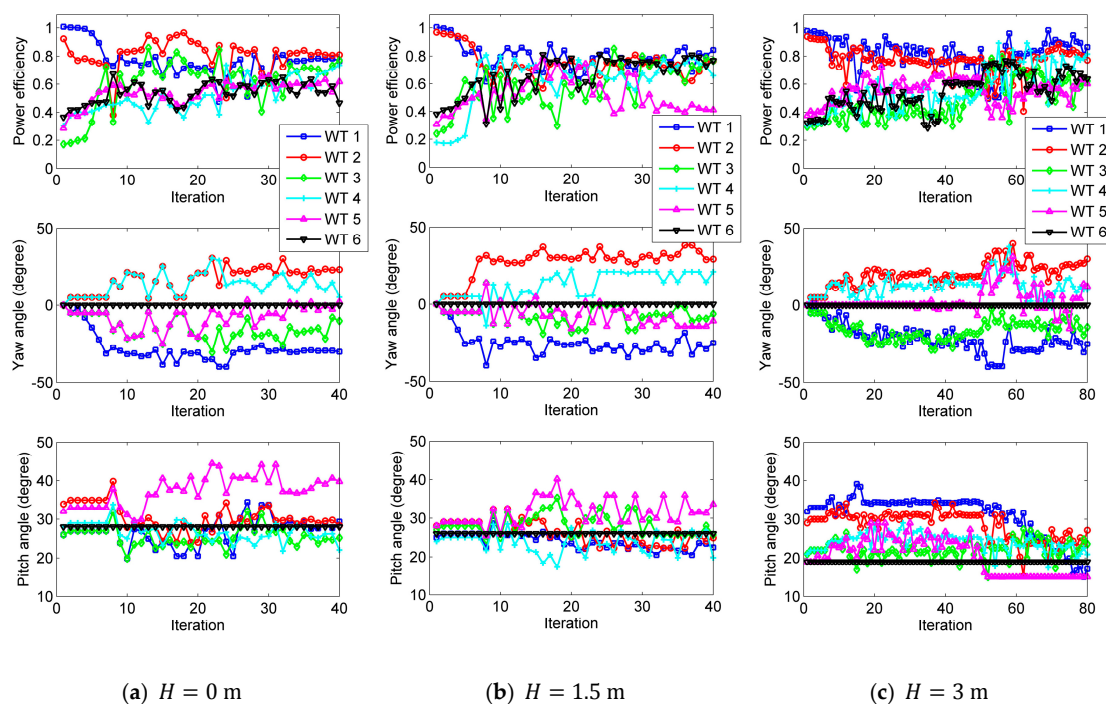


Figure 13. Control actions and power efficiencies for different wind directions.

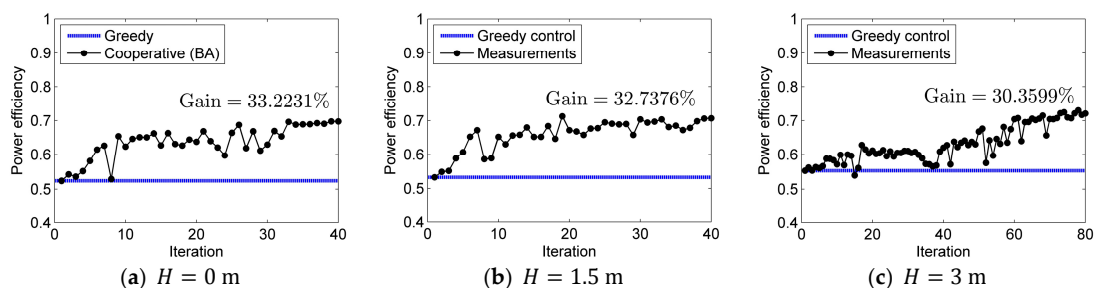


Figure 14. Improvement on power production using cooperative control for different wind directions.

In terms the capability of the BA algorithm, the following trends are observed:

- As shown in the Figure 14, the lateral distance does not dramatically influence the wind farm power efficiencies for both the greedy and the cooperative control strategies. This result implies

that the cooperative control strategy would still be effective for a wind farm where wind turbines are placed closely together.

- When $H = 3$ m, the convergence rate of the BA algorithm is slower than the other cases. The BA algorithm requires a large number of trial actions initially before finally converging to the local optimum.

6. Discussion and Conclusions

Power production of a wind farm often significantly decreases due to wake interference among the wind turbines. To mitigate the wake interference and thus to increase the total wind farm power production, we have formulated the cooperative wind farm control problem. In this approach, the optimum coordinated control actions that maximize the total wind farm power is determined and executed. As a method to determine the optimum coordinated control actions, we use the Bayesian Ascent algorithm previously developed by the authors [17]. As a data driven optimization algorithm, the BA 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 cooperative control in improving the total wind farm power; and (2) the capability of the BA algorithm to find the optimum coordinated control actions using only the input and the resultant power measurement data.

For the cooperative control strategy, the upstream wind turbine acts to deflect the wake trajectory away from the downstream wind turbines. As a result, the downstream wind turbines produce more power, which leads to higher power output for the wind farm as a whole. The improvements in power production depend on the wind farm configurations and the wind direction. Relative to the greedy control strategy, the improvement in the total wind farm power by the cooperative control strategy is greater when a large number of wind turbines are considered, and the wind direction is aligned to the wind turbine array. In other words, when the influence of the wake on the downstream wind turbines is severe, cooperative control can achieve significant improvement in the total wind farm power. This observation implies that cooperative control can be potentially beneficial when a large number of wind turbines are placed in a limited wind farm area.

The BA algorithm is employed to determine optimum coordinated control actions for the wind turbines that maximize the total power production. Due to the trust region constraints, the BA algorithm almost monotonically increases the target wind farm power by gradually chaining the control actions of the wind turbines. The required time for the BA algorithm to converge depends on the size of a target wind farm. More wind turbines require the BA algorithm to model a higher dimensional target function with data; thus, it requires a longer time to converge on the optimal operational conditions. In addition, the time interval between the successive trial actions also increases with the size of a wind farm. For example, if the wind speed is 10 m/s and there are 5 wind turbines with a space of 500 m, the algorithm needs to wait at least 200 seconds for the modified wind flow to reach the last wind turbine. In addition to this traveling time an additional time may be required to compute the average power output.

Although it takes time to converge on the unknown optimal operational point, the BA algorithm continuously and almost monotonically increases the wind farm power output, thus eliminating or minimizing the loss of energy production during the learning process. It should be noted that the true optimum is generally unknown for cooperative control, the role of the BA algorithm is not to identify the exact optimum, but to improve the wind farm power as quickly as possible compared to the initial strategy.

Lastly, for the wind tunnel test set up, the wind conditions do not change during the execution of the BA algorithm for finding the optimum coordinated control actions. In a real wind farm, the wind speed and wind direction are constantly changing. In future study, we will explore two strategies to employ the BA algorithm under varying wind conditions. First is to run multiple BA algorithms

in parallel, each of which is designated to maximize the total wind farm power for a certain wind condition; each BA algorithm can be freely interrupted and reinitiated during the variation of wind conditions because the GP regression model can memorize the learned target function. Another approach is to treat the wind conditions (i.e., wind speed and wind direction) as random variables and to construct a GP regression model on both the action (wind turbine control inputs) and the context (wind conditions). The kernel functions for the wind turbine control action space and for the wind conditions space can be constructed separately and combined to a generalized kernel function for the paired variables between the wind turbine control actions and wind conditions. To involve experimental design to test and validate the control strategies is also among future challenges.

Author Contributions: Jinkyoo Park designed the algorithm for data-driven wind farm control; Jinkyoo Park and Soon-Duck Kwon designed the scaled wind turbines and conducted experiments; Jinkyoo Park and Kincho Law analyzed and wrote the paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Adaramola, M.S.; Krogstad, P.A. Experimental investigation of wake effects on wind turbine performance. *Renew. Energy* **2011**, *36*, 2078–2086. [\[CrossRef\]](#)
- Dahlberg, J.; Medici, D. Potential improvement of wind turbine array efficiency by active wake control. *Proc. Eur. Wind Energy Conf.* **2013**, 65–84.
- Medici, D. Experimental Studies of Wind Turbine Wakes: Power Optimization and Meandering. Ph.D. Thesis, KTH Royal Institute of Technology, Stockholm, Sweden, 2005.
- Marathe, N.; Swift, A.; Hirth, B.; Walker, R.; Schroeder, J. Characterizing power performance and wake of a wind turbine under yaw and blade pitch. *Wind Energy* **2015**, *19*, 963–978. [\[CrossRef\]](#)
- Campagnolo, F.; Petrović, V.; Bottasso, C.L.; Croce, A. Wind tunnel testing of wake control strategies. In Proceedings of the American Control Conference, Boston, MA, USA, 6–8 July 2016; pp. 513–518.
- Gebraad, P.M.O.; Wingerden, J.W. Maximum power-point tracking control for wind farms. *Wind Energy* **2015**, *18*, 429–447. [\[CrossRef\]](#)
- Gebraad, P.M.O.; Teeuwisse, F.; Wingerden, J.; Fleming, P.; Ruden, S.; Marden, J.; Pao, L. Wind plant power optimization through yaw control using a parametric model for wake effects—A CFD simulation study. *Wind Energy* **2016**, *1*, 95–114. [\[CrossRef\]](#)
- Fleming, P.A.; Gebraad, P.M.O.; Lee, S.; Wingerden, J.W.; Johnson, K.; Churchfield, M.; Moriarty, P. Simulation comparison of wake mitigation control strategies for a two-turbine case. *Wind Energy* **2014**, *18*, 2135–2143. [\[CrossRef\]](#)
- Fleming, P.A.; Gebraad, P.M.O.; Lee, S.; van Wingerden, J.W.; Johnson, K.; Churchfield, M.; Michalakes, J.; Spalart, P.; Moriarty, P. Evaluating techniques for redirecting turbine wakes using SOWFA. *Renew. Energy* **2014**, *70*, 211–218. [\[CrossRef\]](#)
- Park, J.; Law, K. Cooperative wind turbine control for maximizing wind farm power using sequential convex programming. *Energy Convers. Manag.* **2015**, *101*, 295–316. [\[CrossRef\]](#)
- Marden, J.; Ruben, S.; Pao, L. A model-free approach to wind farm control using game theoretic methods. *IEEE Trans. Control Syst. Technol.* **2013**, *21*, 1207–1214. [\[CrossRef\]](#)
- Campagnolo, F.; Petrović, V.; Schreiber, J.; Nanos, E.M.; Croce, A.; Bottasso, C.L. Wind tunnel testing of a closed-loop wake detection controller for wind farm power maximization. *J. Phys. Conf. Ser.* **2016**, *753*, 032006. [\[CrossRef\]](#)
- Bronchu, E.; Cora, M.V.; Freitas, N. *A Tutorial on Bayesian Optimization of Expensive Cost Functions with Application to Active User Modelling and Hierarchical Reinforcement Learning*; Department of Computer Science, University of British Columbia: Vancouver, BC, Canada, 2013.
- Osborne, M. Bayesian Gaussian Process for Sequential Prediction, Optimization and Quadrature. Ph.D. Thesis, Department of Computer Science, University of Oxford, Oxford, UK, 2010.
- Park, J. Data-Driven Cooperative Wind Farm Control. Ph.D. Thesis, Stanford University, Stanford, CA, USA, 2016.

16. Jones, D.; Schonlau, M.; Welch, W. Efficient global optimization of expensive black-box functions. *J. Glob. Optim.* **1998**, *13*, 455–492. [[CrossRef](#)]
17. Park, J.; Law, K. A Bayesian optimization approach for wind farm power maximization. In Proceedings of the Smart Sensor Phenomena, Technology, Networks, and Systems Integration 2015, San Diego, CA, USA, 8 March 2015.
18. Nash, J. Equilibrium points in n-person games. *Proc. Natl. Acad. Sci. USA* **1950**, *36*, 48–49. [[CrossRef](#)] [[PubMed](#)]
19. Rasmussen, C.E.; Williams, C.K. *Gaussian Processes for Machine Learning*; MIT Press: Cambridge, MA, USA, 2006.
20. Neal, R.M. *Bayesian Learning for Neural Networks*; Springer: New York, NY, USA, 1996.
21. Mockus, J.; Fretitas, A.; Castelanous, J.A. *Toward Global Optimization*; North-Holland: Amsterdam, NY, USA, 1978.
22. Nocedal, J.; Wright, S. *Numerical Optimization*; Springer: New York, NY, USA, 2000.



© 2017 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).