Proceedings of the 2025 Winter Simulation Conference
E. Azar, A. Djanatliev, A. Harper, C. Kogler, V. Ramamohan, A. Anagnostou, and S. J. E. Taylor, eds.

DYNAMIC CALIBRATION OF DIGITAL TWIN VIA STOCHASTIC SIMULATION: A WIND ENERGY CASE STUDY

Yongseok Jeon[1], Sara Shashaani[1], Eunshin Byon[2], and Pranav Jain[1]

[1]Fitts Dept. of Industrial and Systems Eng., North Carolina State University, Raleigh, NC, USA
[2]Industrial and Operations Eng., University of Michigan, Ann Arbor, MI, USA

ABSTRACT

This study presents an approach to dynamically calibrate a digital twin to support decision-making in systems operating under uncertainty. The framework integrates uncertainty by turning a physics-based model into a stochastic simulation, where independent variables that represent environmental conditions may be nonstationary whereas target variables are conditionally stationary. Calibration itself is formulated as a simulation optimization problem that we solve using a root-finding strategy. As a case study, we apply the framework to the prediction of short-term power deficit, known as the wake effect, in wind farms using real-world data and demonstrate the robustness of the proposed framework. Besides advancing the digital twin research, the presented methodology is expected to impact wind farm wake steering strategy by enabling accurate short-term wake effect prediction.

1 INTRODUCTION

Stochastic simulation research is undergoing a fascinating evolution in the age of digital twins. One basic research need for digital twins is the ability to exploit the continuous ingestion of data to adjust (calibrate) the model rapidly and robustly for more accurate near-term predictions. In other words, dynamic calibration of simulation models to characterize the evolving behavior of the underlying real-world environment with an influx of data is a growing challenge we address in this paper.

1.1 Motivating Examples

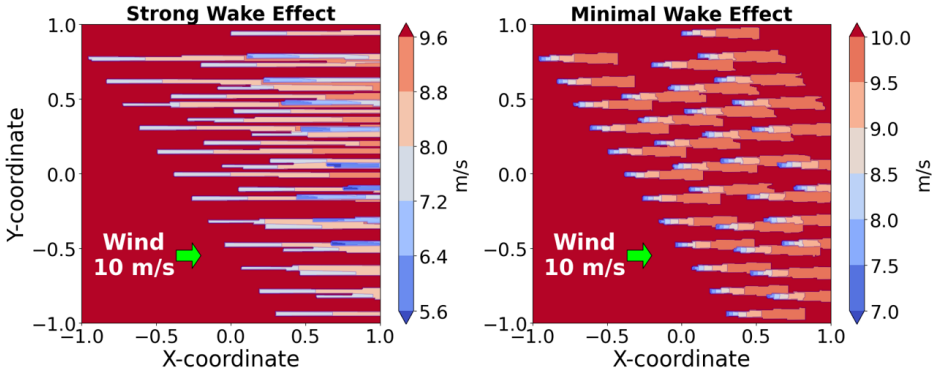

Figure 1: Simulated wind speed deficit in a wind farm under strong and minimal wake conditions using the FLORIS model available at https://nrel.github.io/floris/.

Our motivating example and case study in this paper involves calibration of wake effect in wind turbines. Wake effect is a aerodynamic phenomenon in which airflow passes through the upstream turbines leads to

a wind speed deficit and increased turbulence for the downstream turbines. This aerodynamic interaction compromises the downstream turbines' power generation and often induces increased mechanical loads. Accurately understanding the wake effect is crucial for optimizing wind farm performance, ensuring turbine reliability, and designing efficient wind farm layout (Shakoor et al. 2016).

Figure 1 illustrates the impact of different wake scenarios in a wind farm of multiple turbines, when the wake effect is modeled by the famous Jensen wake model (Jensen 1983). Figure 2 depicts the relationship between the wake parameter, optimal rotor angle offsets, and the resulting achievable power generation. In strong wake condition, the turbines adopt larger yaw offsets to deflect the wake effect and mitigate downstream power losses. In contrast, under weaker wake conditions, the wind power deficit recovers more rapidly, and the optimal rotor angle align more closer to the wind direction. Such decision-making in wind farm is known as wake steering, where rotor angles are strategically adjusted to optimize power generation by managing wake interactions.

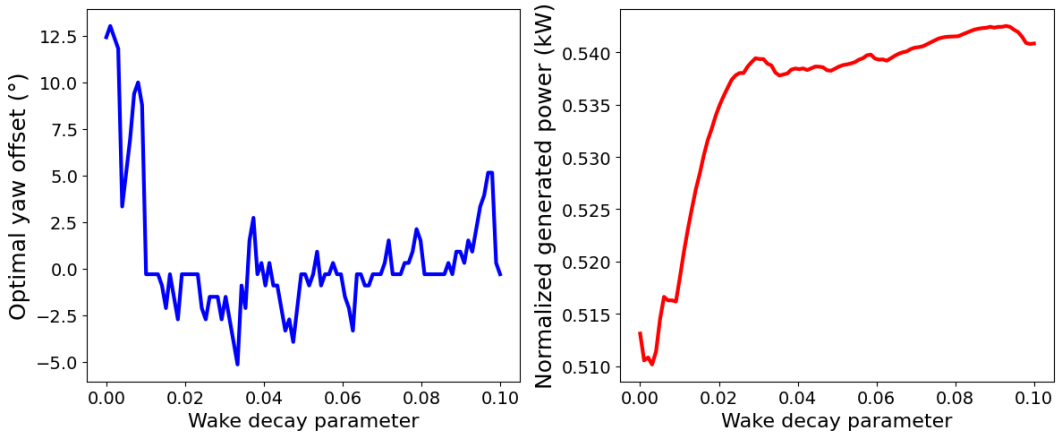

Figure 2: Impact of wake decay on optimal rotor angle (left) and the resulting power output (right).

1.2 Existing Work

In practice, dynamically implementing wake steering strategies is challenging. Traditionally, wake effects have mostly been analyzed with their long-term effects (statically), focusing on optimizing a global wake parameter by aligning the outputs of the model to the historical data through empirical minimization or using point predictions via machine-learning (Jain et al. 2023; Park et al. 2022). While long-term wake analysis is valuable in layout design or evaluating overall efficiency, it is less suited for the real-time operational decisions. A recent study modeled inter-turbine delays to refine short-term wake dynamics and inform control strategies (Xiao et al. 2025). However, the delayed wake effect is deterministically computed using fixed time intervals based on current conditions, rather than through a fully predictive or stochastic framework. Indeed a large portion of dynamic calibration literature is based on deterministic models or fixed-delay assumptions, such as moving average prediction or by deliberate selection of window size (Wang et al. 2025; Staid et al. 2015). These methods do not fully account for the inherent uncertainties present in the collected data due to weather variability, unknown turbine configurations, and measurement errors. A framework based in the stochastic simulation that explicitly accounts for uncertainty in the environment appears to be more suited. Although, there have been recent studies where stochastic simulation is used to generate the power output scenarios (Zhang et al. 2024), quantify the risk of instability (Vicente et al. 2017) and spatial optimization of the wind farm (Katikas et al. 2024), its use in the analysis of wake effects remains relatively unexplored. A critical challenge in the use of simulation to accurately predict the correct wake under the upcoming weather conditions lies in the nonstationary nature of the wind conditions and its interaction with the stochastic behavior of the turbines. We focus on this aspect in our proposed work.

1.3 Our Contribution

We present a stochastic simulation-based digital twin calibration framework that blends physics-based, data-driven, and probabilistic models to account for evolving and uncertain data. We divide the inputs to the calibration problem (which are covariates and target variable in the dataset) into nonstationary and conditionally stationary random variates, to which we fit appropriate probabilistic input models to provide a data generator for the simulation optimization problem. While we present the framework as tailored to our motivating example, with a weather emulator, a wind farm simulator, and a physics-based wake model to capture turbine responses and interactions, it is generalizable to other digital twin applications.

Our paper is organized as follow. In Section 2, we formally state and formulate the problem, introduce a dual reformulation, and list key assumptions. We then present the proposed stochastic-simulation based dynamic wake calibration in Section 3, detailing the design of the weather emulator, wind farm simulator, and the calibration process. In Section 4, we provide experimental results using real-world wind farm data to evaluate the wake prediction performance. Lastly, in Section 5, we conclude with final remarks and discuss the broader implications of our framework in enabling proactive wind farm control strategies.

2 PROBLEM STATEMENT AND PROPOSED FRAMEWORK

We start by establishing the necessary definitions and assumptions. Let $x^{\text{real}} \in \mathcal{X} \subset \mathbb{R}^{m_x}$ represent the m_x-dimensional vector of independent variables (often representing environmental conditions in digital twins). Let $y^{\text{real}} \in \mathcal{Y} \subset \mathbb{R}^{m_y}$ be the corresponding m_y-dimensional vector of target variables. Consider $\theta \in \Theta$ as the calibration parameter of the digital twin output $h : \mathcal{X} \times \Theta \to \mathcal{Y}$. Assuming we have T data points $\{(x_t^{\text{real}}, y_t^{\text{real}}),\ t = 1, 2, \cdots, T\}$ collected over time at our disposal, define a (positive) *discrepancy* function $d_+ : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_+$ such as absolute or mean squared error. Then, the classical (static) calibration problem involves

$$\min_{\theta \in \Theta} \ \frac{1}{T} \sum_{t=1}^{T} d_+(y_t^{\text{real}}, h(x_t^{\text{real}}, \theta)).$$

Since the independent variables are largely time-varying and affect the accuracy of a digital twin, it is desirable to adjust the calibration parameter θ_t for the newest incoming data $(x_t^{\text{real}}, y_t^{\text{real}})$ at time t, which can be the solution to $\min_{\theta \in \Theta_t}\ d_+(y_t^{\text{real}}, h(x_t^{\text{real}}, \theta))$, with $\Theta_t = \Theta$. This calibration decision has to be made before we receive data from the system at time t. Therefore, the original problem changes to

$$\min_{\theta \in \Theta_t} \ d_+(y_t^{\text{pred}}, h(x_t^{\text{pred}}, \theta)), \tag{1}$$

where $(x_t^{\text{pred}}, y_t^{\text{pred}})$ are the predictions of what we would expect to see at time t. The easiest choice for the predicted values is $(x_{t-1}^{\text{real}}, y_{t-1}^{\text{real}})$. Other approaches aiming to increase the predictive accuracy of $(x_t^{\text{pred}}, y_t^{\text{pred}})$ use moving averages or machine learning techniques that provide point predictions.

2.1 Regularization

In time-varying environments, the collected data $(x_t^{\text{real}}, y_t^{\text{real}})$ itself is subject to noise, and calibrating the system in this condition, even with the knowledge of future conditions, can lead to unrealistic parametric evolution. Robust calibration would be less sensitive to such fluctuations that are misguided by noise in the data (as in our case study). One strategy here is to add regularization terms to the objective, or impose constraints that force the changes in the parameter to be more gradual. Regularization may prove ineffective when the real system changes are significant impacting the scale of the target variables from time to time. Therefore, we focus instead on constraining the feasible space to $\Theta_t = \{\theta \in \Theta :\ |\theta_{t-1}^{\text{real}} - \theta| \leq \eta\}$ with an allowable change threshold of $\eta > 0$. Here, we have access to x_{t-1}^{real} and y_{t-1}^{real}, which allows us to compute $\theta_{t-1}^{\text{real}} = \arg\min_{\theta \in \Theta} d_+(y_{t-1}^{\text{real}}, h(x_{t-1}^{\text{real}}, \theta))$. However, since $\theta_{t-1}^{\text{real}}$ itself is affected by the noise in the data,

the earlier concerns would cause challenges in picking η as well. We therefore consider the dual version of the optimization problem (1) as

$$\min_{\theta\in\Theta} \ |\theta_{t-1}^{\text{real}} - \theta| \text{ s.t. } d_+(y_t^{\text{pred}}, h(x_t^{\text{pred}}, \theta)) \leq \kappa_{t,\alpha}^{\text{pred}}, \tag{2}$$

where $\kappa_{t,\alpha}$ can be set based on the range of the discrepancy function controlled by $\alpha \in [0,1]$. Define $g_t^{\text{real}}(\theta) := d_+(y_t^{\text{real}}, h(x_t^{\text{real}}, \theta))$ with $g_{t,\min}^{\text{real}} = \min_{\theta\in\Theta} g_t^{\text{real}}(\theta)$ and $g_{t,\max}^{\text{real}} = \max_{\theta\in\Theta} g_t^{\text{real}}(\theta)$. Then the predicted version of $\kappa_{t,\alpha}^{\text{real}} = \alpha(g_{t,\max}^{\text{real}} - g_{t,\min}^{\text{real}}) + g_{t,\min}^{\text{real}}$, which ensures that

$$\frac{g_t^{\text{real}}(\theta) - g_{t,\min}^{\text{real}}}{g_{t,\max}^{\text{real}} - g_{t,\min}^{\text{real}}} \leq \alpha,$$

would yield the right-hand-side in (2). Notably, α governs how tightly the calibration is expected to follow past trends. A lower α allows the calibration to be more responsive to rapid fluctuations in the observed data. In contrast, a higher α promotes smoother (regularized) and more gradual calibration. Figure 3 illustrates how α influences the resulting parameter trajectories in the wake calibration context.

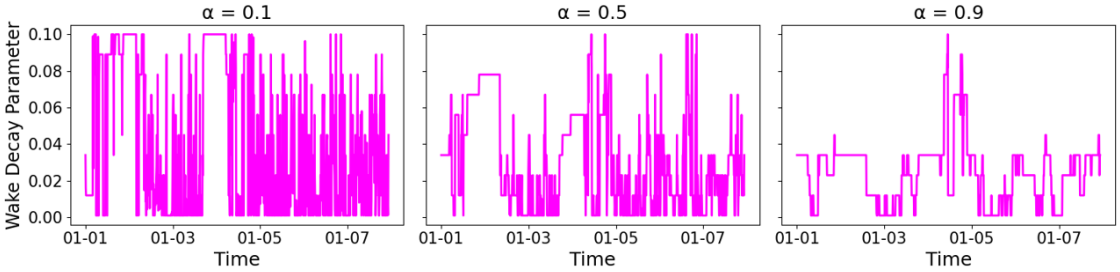

Figure 3: Wake parameter evolution trends with different α settings.

2.2 Stochastic Simulation

In our approach, we turn the calibration problem into a simulation-optimization problem by using probabilistic predictions or a distribution for random vectors (X_t, Y_t), denoted by $\mathbb{P}_{X_t,Y_t}$, i.e.,

$$\min_{\theta\in\Theta} \ |\theta_{t-1}^{\text{real}} - \theta| \quad \text{s.t. } \mathbb{E}_{X_t,Y_t}\left[d_+(Y_t, h(X_t,\theta))\right] \leq \kappa_{t,\alpha}^{\text{sim}}(\mathbb{P}_{X_t,Y_t}). \tag{3}$$

Here, the probability distribution generates all variables jointly, capturing dependencies and relationships across all variables. Note, in (3), the constraint's right-hand-side now needs to be computed with respect to $\mathbb{P}_{X_t,Y_t}$ as well. This poses a computational challenge as we need to compute the minimum and maximum of $\mathbb{E}_{X_t,Y_t}[d_+(Y_t, h(X_t,\theta))]$, adding two more simulation-optimization problems besides the main one.

Importantly, in digital twins the independent variables (e.g., weather or environmental characteristics) can have a nonstationary behavior, but the behavior of the target variable (e.g., turbine output) given a realization of independent variable may not be nonstationary. We assume, therefore, that the conditional distribution of target $\mathbb{P}_{Y_t|X_t}$ can be modeled separately to use $\mathbb{P}_{Y_t|X_t}\mathbb{P}_{X_t}$ as an alternative to $\mathbb{P}_{X_t,Y_t}$ in (3).

Remark 1 Why would the target variable given X_t be random? In the wind farm case study for example, even if we know X_t (weather conditions), one could expect a distribution of Y_t (generated power) that varies for each turbine due to the rotor angle in that turbine. The collected $(x_t^{\text{real}}, y_t^{\text{real}})$ does not provide the information of under which rotor angle such power was generated, introducing another source of uncertainty.

We, therefore, propose a nested simulation, where in the outer level, X_t is simulated using a nonstationary probability model, and given its realized value, the inner level produces Y_t from a conditional probability

distribution function that accounts for uncertainty about configurations in the system from which the data was collected:

$$\min_{\theta\in\Theta} \ |\theta_{t-1}^{\text{real}} - \theta| \text{ s.t. } \underbrace{\mathbb{E}_{X_t}\left[\mathbb{E}_{Y_t|X_t}\left[d_+(Y_t, h(X_t,\theta))\right]\right]}_{:=g_t^{\text{sim}}(\theta)} \leq \kappa_{t,\alpha}^{\text{sim}}(\mathbb{P}_{Y_t|X_t}\mathbb{P}_{X_t}). \tag{4}$$

Remark 2 We hypothesize that fitting a time-varying distribution to the target variables may be an overshoot and result in biases due to artificial correlations. In Figure 6, we justify this hypothesis in a side experiment for our case study by illustrating worse calibrations under a joint input model.

Remark 3 While Y_t is assumed to be nonstationary given X_t, it may still contain auto-correlation. In our application, we found that the generated power exhibits temporal-momentum effects (e.g., rotor inertia in wind turbines), which renders Y_t also time-varying. This time-varying behavior can be treated separately by making the current output dependent not only on the current environmental conditions but also on the system's recent output. In other words, we still consider Y_t stationary given X_t, yet introduce an auto-correlation structure by using y_{t-1}^{real} in specifying the conditional probability model, i.e., $\mathbb{P}_{Y_t|X_t} = \mathbb{P}_{Y_t|X_t,y_{t-1}^{\text{real}}}$.

3 DYNAMIC WAKE EFFECT PREDICTION WITH STOCHASTIC SIMULATION

In the wake calibration study, x_t^{real} contains m_x important summary statistics (i.e., mean and standard deviation) of key wind variables such as wind speed, wind direction and turbulence intensity observed from a met mast. Let $y_t^{\text{real}} \in \mathbb{R}^{m_y}$ be the corresponding m_y-dimensional vector of turbine outputs at time-t, where $y_{t,i}^{\text{real}} \in \mathbb{R}$ is the observed power output of turbine i. We assume the wake effect follows a parameterized physics-based model for turbine i, represented as $[\mathcal{J}(x_t^{\text{real}};\theta)]_i$ such as the Jensen's wake model (Jensen 1983). We define θ_t^{real} and θ_t^{sim} as the true and calibrated (solution to problem (4)) wake parameter at time t, in the $\Theta = [0.001, 0.1]$ domain. Note, θ_t^{sim} does not necessarily equate θ_t^{real}. To estimate the stochastic constraint in (4), we generate s realizations from a weather emulator $\mathcal{W} = \mathbb{P}_{X_t}$ and for each weather realization, we simulate r turbine responses using the wind farm simulator $\mathcal{M} = \mathbb{P}_{Y_t|X_t,y_{t-1}^{\text{real}}}$ to account for uncertainties. Modeling of these probability distributions is described in Section 3.1. Figure 4 provides an overview of this framework.

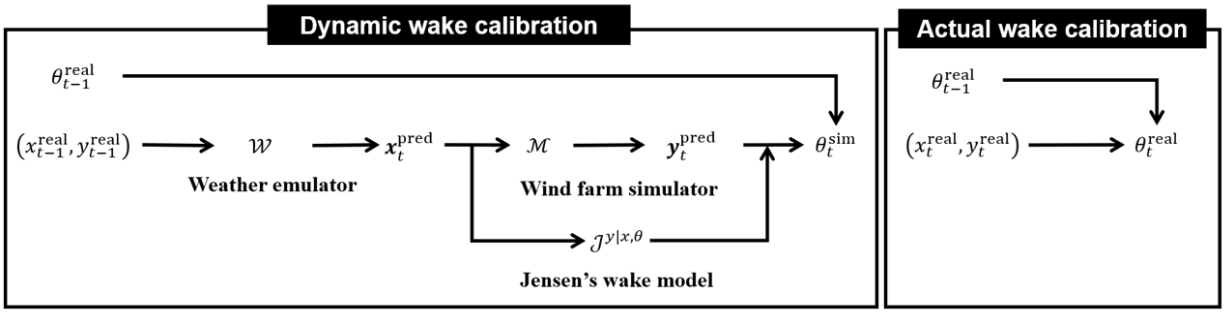

Figure 4: Proposed dynamic wake calibration framework and actual wake calibration process.

Suppose we use an L_1 discrepancy metric, $d_+(a,b) = m_y^{-1}\sum_{i=1}^{m_y}|a_i - b_i|,\ \forall a,b \in \mathbb{R}^{m_y}$. Then our estimate of $g_t^{\text{sim}}(\theta)$ is computed as

$$\hat{g}_t^{\text{sim}}(\theta; s, r) = \frac{1}{s}\sum_{j=1}^{s}\frac{1}{r}\sum_{k=1}^{r}\frac{1}{m_y}\sum_{i=1}^{m_y}\left|Y_{t,i,j,k} - [\mathcal{J}(X_{t,j},\theta)]_i\right|, \tag{5}$$

where $[\mathcal{J}(X_{t,j},\theta)]_i$ is the predicted output of turbine-i under weather realization-j and $Y_{t,i,j,k}$ is the simulated output of turbine-i of scenario-j in the k-th replication from $\mathbb{P}_{Y_t|X_{t,j},y_{t-1}^{\text{real}}}$.

The minimum and maximum of $\hat{g}_t^{\text{sim}}(\theta; s, r)$ would then be used to approximate the right-hand-side value in (4). Solving this optimization problem poses significant computational challenges. First, evaluating the physics-based model $\mathcal{J}$ across the forecast scenarios incurs high simulation costs and is particularly unsuitable in wind farm domain where fast and frequent responses are essential for operational decision-making. Second, verifying feasibility of a candidate wake parameter adds another layer of difficulty, as the threshold $\kappa_{t,\alpha}^{\text{sim}}$ itself must be estimated through excessive simulation runs. To address these computational challenges, we propose calibration with Bayesian optimization, where design points of θ can help estimate the right-hand-side constant $\kappa_{t,\alpha}^{\text{sim}}$ and our formerly developed root-finding strategy (Jeon and Shashaani 2024) can accelerate the computation. To facilitate root-finding, we would need to adjust the discrepancy metric, which we will discuss in Section 3.2.

3.1 Input Models—Fitting Probability Models to X_t and $Y_t|X_t, y_{t-1}^{\text{real}}$

In the wind farm context, dependencies among weather variables are hard to justify. Each weather feature represents a summary statistic of a distinct wind property. To apply multivariate modeling effectively, one must construct a covariance structure that precisely captures the temporal dynamics of each feature while preserving their underlying physical dependence. This is particularly challenging when the physical relationships between variables are not fully characterized or are nonstationary. We address this challenge by assuming independence between each weather variable (which our case study data justified) and fitting a probability model to each based on its own temporal behavior.

To model the temporal evolution of wind features, we use Gaussian process regressor (GPR). At time $t-1$, separate GPR for each weather variable is trained using the recent observations $\mathbf{x}_{\text{train}}$ to produce a probabilistic distribution (forecast) for time t. We compare the following two approaches for GPR training:

1. Time-index GPR models each weather variable independently as a function of time. At time $t-1$, we construct a GPR for the ℓ-th weather variable over time, denoted $f_{t-1,\ell}^{\text{time}}$, with the ℓ-th weather variable $x_{t',\ell} = f_{t-1,\ell}^{\text{time}}(t'), \forall\ t' \leq t-1,\ \ell \in \{1,\ldots,d\}$. Time-index GPR is trained using the previous n time steps information, i.e., $\mathbf{x}_{\text{train}} = \{(t-n, x_{t-n,\ell}), \ldots, (t-1, x_{t-1,\ell})\}$. The resulting GPR is then used to generate a probabilistic forecast for $X_{t,\ell}$ with input t.
2. Lagged-input GPR models each weather variable as a function of its recent temporal trends. Let n' denote the size of the lagged window, where each training input consists of previous n' values of the variable, and the corresponding target is the subsequent value. At time $t-1$, we fit a GPR for the ℓ-th weather variable on its lagged window, denoted $f_{t-1,\ell}^{\text{lag}}$. The model is trained using the training set $\mathbf{x}_{\text{train}} = \{((x_{\tau-n',\ell}, \ldots, x_{\tau-1,\ell}), x_{\tau,\ell}) |\ \tau \in \{t-n+n', \ldots, t-1\}\}$. To construct the predictive distribution of $x_{t,\ell}$, $(x_{t-n',\ell}, \ldots, x_{t-1,\ell})$ is used as an input. Importantly, we set $n' \leq n-2$ to avoid training GPR on a single-pair training set when $n' = n-1$.

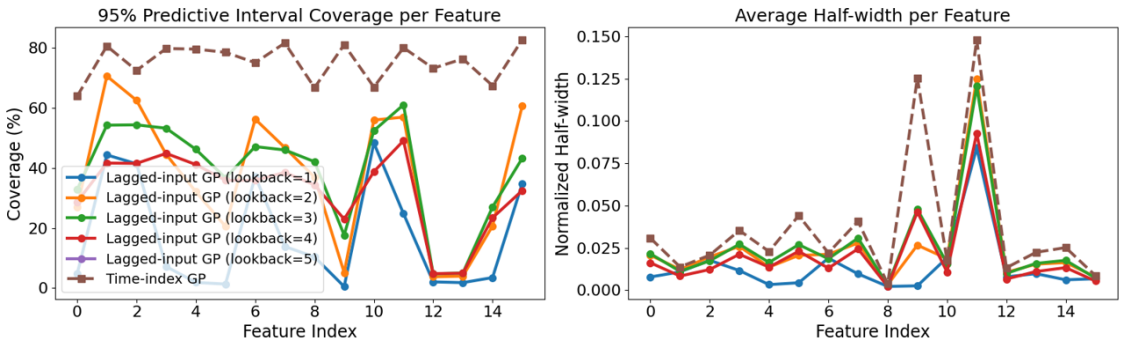

Figure 5: Prediction interval coverage and half-width using $n = 6$ time steps for training.

Figure 5 illustrates the PI coverage and half-width under $n = 6$ time steps (i.e. one-hour), based on the experiment using a real-world, one-year dataset to evaluate the forecasting performance of both approaches. In this experiment, at each prediction time, the constructed GPR provides a predictive distribution for the ℓ-th weather variable as $X_{t,\ell} \sim \mathcal{N}(\mu_{t,\ell}, \sigma^2_{t,\ell})$, where $\mu_{t,\ell}$ and $\sigma^2_{t,\ell}$ are its predictive mean and variance at time t, respectively. The corresponding $(1-\alpha)$-level prediction interval (PI) is then expressed as

$$\mathrm{PI}^{(1-\alpha)}_{t,\ell} = \left[\mu_{t,\ell} - z_{1-\alpha/2}\sigma_{t,\ell},\ \mu_{t,\ell} + z_{1-\alpha/2}\sigma_{t,\ell}\right].$$

The experimental results suggest that the time-indexed GPR achieves higher coverage while maintaining comparable half-width. Therefore, we use this approach for modeling the weather emulator $\mathcal{W}$.

We now describe the wind farm simulator $\mathcal{M}$ that is the conditional probability distribution of power outputs for each turbine given the weather characteristics and the most recent output. Given the 10-minute resolution of the dataset, turbine outputs at consecutive time steps are expected to be highly-correlated, making temporal features more capable of capturing short-term dynamics. We adopt ordinary least squares (OLS) regression (Staum 2009) using the actual weather covariates x^{real}_t along with each turbine's recent output $y^{\text{real}}_{t-1,i}$ as predictors. Recall that using $y^{\text{real}}_{t-1,i}$ helps to account for the absence of turbine-level control settings (e.g., individual rotor angles) in our study, as it implicitly reflects the impact of recent control decisions that are likely to persist over a short time interval. Accordingly, this relationship is expressed as

$$y^{\text{real}}_{t,i} = \mathbf{z}^\top_{t,i}\beta_i + \epsilon_i, \quad \epsilon_i \sim \mathcal{N}(0, \sigma^2_i), \tag{6}$$

where $\mathbf{z}_{t,i} = [x^{\text{real}}_t,\ y^{\text{real}}_{t-1,i}]^\top \in \mathbb{R}^{m_x+1}$ is the input vector. The term ϵ_i represents the residual variability due to the unobserved factors and inherent stochastic behavior of the turbine outputs. Due to lack of detailed information about these effects, we model ϵ_i as an independent Gaussian noise. The fixed model coefficients $\beta_i \in \mathbb{R}^{m_x+1}$ can be estimated through

$$\hat{\beta}_i = (\mathbf{Z}^\top_{T,i}\mathbf{Z}_{T,i})^{-1}\mathbf{Z}^\top_{T,i}\mathbf{y}_{T,i}$$

in which $\mathbf{z}_{T,i} \in \mathbb{R}^{T\times(m_x+1)}$ is the input matrix, T is the size of the training set and $\mathbf{y}_{T,i} \in \mathbb{R}^T$ is the vector of recent turbine outputs over the training set. The residual noise variance is estimated as

$$\hat{\sigma}^2_i = \frac{1}{T-(m_x+1)}\sum_{k=1}^{T}\left(y_{T,i,k} - \mathbf{z}^\top_{T,i,k}\hat{\beta}_i\right)^2$$

where $y_{T,i,k}$ is the k-th element of $\mathbf{y}_{T,i}$. To capture the stochastic variability in turbine responses under fixed weather conditions, we simulate r sub-scenarios of turbine outputs for each generated weather realization. Hence, for i-th turbine with j-th weather realization $X_{t,j}$, r-turbine outputs are generated as

$$Y_{t,i,j,k} = \mathbf{z}^\top_{t,i}\hat{\beta}_i + \hat{\epsilon}_{i,k}, \quad \hat{\epsilon}_{i,k} \sim \mathcal{N}(0, \hat{\sigma}^2_i), \quad k = 1, \ldots, r,$$

where $\mathbf{Z}_{t,i,j} = [X_{t,j}, y^{\text{real}}_{t-1,i}]^\top \in \mathbb{R}^{m_x+1}$ is the input vector.

3.2 Calibration with Root-Finding of a Simulation Metamodel via Bayesian Optimization

The crux of the proposed framework is calibrating the wake decay parameter by aligning the output of the physics-based model $\mathcal{J}$ under s copies of X_t each with r copies of the corresponding power output Y_t. As discussed earlier, directly solving the problem in (4) is computationally challenging, as each evaluation requires running up to sr simulations, and most importantly, the threshold value $\kappa^{\text{sim}}_{t,\alpha}$ is not known in advance. For a practically efficient solution method, we decompose the problem into two stages. In the first stage, we solve the primal optimization without constraints—counterpart of (1) but as a simulation

optimization, i.e., $\min_{\theta\in\Theta} g_t^{\text{sim}}(\theta)$ as defined in (4). This setup allows us to explore a wide range of candidate wake parameters without performing repeated feasibility checks. Exploring this broader set of candidates is still important, as even infeasible parameters may contribute to identifying better candidates with lower discrepancy and those with worst-case discrepancy that can determine the $\kappa_{t,\alpha}^{\text{sim}}$ threshold. Hence, we first minimize $\hat{g}_t^{\text{sim}}(\theta; s, r)$ as defined in (5).

Since directly minimizing this objective is computationally expensive due to many replications per evaluation, we use a root-finding simulation metamodel approach that we previously proposed in (Jeon and Shashaani 2024). Bayesian optimization with root-finding reduces the search space leading to faster convergence and fewer simulation runs. This metamodeling approach is particularly well-suited for our setting, as the aforementioned infeasible candidates can naturally be leveraged when constructing the metamodel.

However, a challenge here is to change the discrepancy function to produce signed values that enable us to find the *root* as the optimal θ, rather than one that minimizes the positivized discrepancy. To this end, we first use a lower bound approximation for (5) computed by

$$\hat{g}_t^{\text{approx}}(\theta; s, r) = \left| \frac{1}{s}\sum_{j=1}^{s}\frac{1}{r}\sum_{k=1}^{r}\frac{1}{m_y}\sum_{i=1}^{m_y} Y_{t,i,j,k} - \left[\mathcal{J}(X_{t,j}, \theta)\right]_i \right|. \tag{7}$$

This approximation measures the discrepancy of the averaged values rather than and L_1 norm. The same approximation could be valid for other norms by the triangle inequality. Note that this quantity estimates $g_t^{\text{approx}}(\theta) = d_+(\mathbb{E}_{X_t}[\mathbb{E}_{Y_t|X_t}[Y_t]], \mathbb{E}_{X_t}[\mathbb{E}_{Y_t|X_t}[h(X_t,\theta)]])$ instead of $g_t^{\text{sim}}(\theta)$ as defined in (4). The immediate result of using $g_t^{\text{approx}}(\theta)$ instead of $g_t^{\text{sim}}(\theta)$ is that we know this function is minimized at the root of the signed average discrepancy. In other words, we estimate the approximate θ_t with $\hat{\theta}_t^{\text{approx}}$ which satisfies

$$\hat{g}_t^{\text{approx-r}}(\hat{\theta}_t^{\text{approx}}; s, r) := \frac{1}{s}\sum_{j=1}^{s}\frac{1}{r}\sum_{k=1}^{r}\frac{1}{m_y}\sum_{i=1}^{m_y} Y_{t,i,j,k} - \left[\mathcal{J}(X_{t,j}, \hat{\theta}_t^{\text{approx}})\right]_i = 0.$$

The root-finding via BO starts by evaluating a set of candidate design points $\{\theta^{(1)}, \ldots, \theta^{(e)}\}$ with their corresponding signed discrepancy values $\{\hat{g}_t^{\text{approx-r}}(\theta^{(p)}; s, r),\ p = 1, \cdots, e\}$ to construct the initial metamodel. It then proceeds by optimizing a root-finding-based acquisition function until termination leveraging a reduced search space for the roots; see (Jeon and Shashaani 2024) for more details.

In the second stage, we need to identify feasibility of the solution to the first stage. We approximate the $g_{t,\text{min}}^{\text{sim}}$ and $g_{t,\text{min}}^{\text{sim}}$ with empirical estimates as

$$\hat{g}_{t,\text{min}}^{\text{sim}} \approx \min_{1\le p\le e} |\hat{g}_t^{\text{approx-r}}(\theta^{(p)}; s, r)|, \quad \text{and} \quad \hat{g}_{t,\text{max}}^{\text{sim}} \approx \max_{1\le p\le e} |\hat{g}_t^{\text{approx-r}}(\theta^{(p)}; s, r)|,$$

and the estimated discrepancy threshold as $\hat{\kappa}_{t,\alpha}^{\text{sim}} = \hat{g}_{t,\text{min}}^{\text{sim}} + \alpha\left(\hat{g}_{t,\text{max}}^{\text{sim}} - \hat{g}_{t,\text{min}}^{\text{sim}}\right)$, which is used to identify the feasible set as

$$\hat{\Theta}_t = \left\{\theta \mid \hat{g}_t^{\text{approx}}(\theta; s, r) \le \hat{\kappa}_{t,\alpha}^{\text{sim}}\right\}.$$

Of all the θ's evaluated during BO, we select the candidate that is closest to the previous $\theta_{t-1}^{\text{real}}$ as $\hat{\theta}_t^{\text{sim}} = \arg\min_{\theta\in\hat{\Theta}_t} |\theta_{t-1}^{\text{real}} - \theta|$. Finally, as these empirical estimates are obtained from a limited set of evaluated candidates, we enforce an additional regularizing step to mitigate potential prediction errors, moderating the influence of uncertain forecasts, ensuring that the update is not overly sensitive to potentially noisy or inaccurate predictions. We apply the following smoothing

$$\theta_t^{\text{sim}} = \gamma\hat{\theta}_t^{\text{sim}} + (1-\gamma)\theta_{t-1}^{\text{real}}, \tag{8}$$

where the smoothing parameter $\gamma \in [0, 1]$ controls the degree of information gain from the predicted values. A lower γ would lead to more conservative updates, tightening the parameter evolution.

3.3 Root-Finding in *Rootless* Instances

It is important to note that root-finding formulation does not always guarantee the existence of a root. This situation arises when the discrepancy between the model output and the target remains persistently positive (or negative) across all parameters, i.e., $\hat{g}_t^{\text{approx-r}}(\theta; s, r) > 0 \ (< 0), \ \forall \theta \in \Theta$. One reason for this incident is insufficient choice of r and s; we leave this investigation for future research. Another reason for this type of mismatch is the presence of unknown influences that are not captured by the digital twin ($\mathcal{J}$). In these cases, we prove that the root-finding approach reverts to solving the original minimization problem. To show this, we examine the limiting behavior of the upper confidence bound (UCB) acquisition function case that was primarily used in our study. We write the predictive distribution for $\hat{g}_t^{\text{approx-r}}(\theta; s, r)$ after l-evaluations, i.e., l design points, obtained through simulation metamodeling as

$$\hat{G}_l(\theta; s, r) \sim \mathcal{N}(\mu_l(\theta; s, r), \sigma_l^2(\theta; s, r)),$$

where $\mu_l(\theta; s, r)$ and $\sigma_l(\theta; s, r)$ are the predictive mean and standard deviation using the fixed replication sizes r and s. The UCB acquisition for the minimization and root-finding are defined, respectively, as

$$\text{UCB}(\theta; s, r) = \mu_l(\theta; s, r) + \lambda\sigma_l(\theta; s, r), \quad \text{UCB}_{\text{RF}}(\theta; s, r) = |\mu_l(\theta; s, r)| + \lambda\sigma_l(\theta; s, r),$$

where $\lambda > 0$ is a user-specified parameter controlling the exploration-exploitation trade-off. With this, we make the following assumptions:

Assumption 1 $\hat{g}_t^{\text{approx-r}}(\theta; s, r)$ is strictly positive, i.e., $\hat{g}_t^{\text{approx-r}}(\theta; s, r) > 0$, for all $\theta \in \Theta$.

Assumption 2 At each iteration, the global optimum is attained when optimizing the acquisition function.

Assumption 3 As l increases, $\sigma_l(\theta; s, r) < \infty$ at any $\theta \in \Theta$.

We then obtain the following theorem:

Theorem 1 Under Assumptions 1, 2, and 3,

$$\left|\text{argmin}_{\theta\in\Theta}\text{UCB}_{\text{RF}}(\theta; s, r) - \text{argmin}_{\theta\in\Theta}\text{UCB}(\theta; s, r)\right| \to 0 \text{ as } s, r \to \infty.$$

We explain a sketch of this proof here. Under Assumption 1, since the metamodel is trained on strictly positive discrepancy values, the predictive mean $\mu_l(\theta; s, r)$ also remains strictly positive in the limit, i.e., $\mu_l(\theta; s, r) > 0$ for all $\theta \in \Theta$ as $l \to \infty$. This observation together with Assumption 3 implies $|\mu_l(\theta; s, r)| = \mu_l(\theta; s, r)$; hence

$$\text{UCB}_{\text{RF}}(\theta; s, r) = \mu_l(\theta; s, r) + \lambda\sigma_l(\theta; s, r) = \text{UCB}(\theta; s, r), \ \forall \theta \in \Theta.$$

Then, Assumption 2, which guarantees the global optimum at each iteration, concludes the proof. For the opposite case in Assumption 1, where the discrepancy function is strictly negative, the same reasoning applies. The interpretation of this theorem is that when using UCB_{RF}, we do not need to know in advance whether $\hat{g}_t^{\text{approx-r}}(\theta; s, r)$ has a root at all, as after sufficient number of observations, UCB_{RF} is expected to make that same decisions as standard UCB. It is also worth emphasizing that the assumptions made in this analysis remain reasonable even when only a small number of evaluations are available. This is because, under Assumption 1, all observed evaluations will also be strictly positive (negative), and the predictive mean $\mu_l(\theta; s, r)$ will remain predominantly positive (negative) across Θ, even with limited number of observations, with a relatively smooth kernel in the metamodel. Similarly, Assumption 3 also holds in practice since the predictive variance of the GPR remains finite for any $\theta \in \Theta$ as long as the parameter space is compact and designs are sufficiently spread out.

4 NUMERICAL EXPERIMENT

We now present our experimental results of the proposed framework. The weather emulator $\mathcal{W}$ and wind farm simulator $\mathcal{J}$ are both trained using a one-year dataset of an existing wind farm and dynamic wake

prediction task is conducted on the dataset of the following year (for propitiatory reasons we leave out the year and other unimportant details in this paper). The simulator $\mathcal{J}$ is implemented using FLORIS (open source model by NREL), and configured with the precise physical layout and specifications of our case study. In the wind farm simulator $\mathcal{J}$, the relationships between the wind speed and both power outputs and thrust coefficients are embedded with fitted splines from the training data. Weather emulator $\mathcal{W}$ is a time-indexed GPR using $n = 6$ time steps. Table 1 presents the set of weather variables used in this study. For each variable, both the mean and standard deviation are computed at a 10-minute resolution, and thereby $m_x = 16$ independent GPR models are trained at each time step. To properly handle the circular nature of the wind directions, directional variables are decomposed into their sine and cosine components. Each GPR is based on the rbf kernel with a length-scale between $[0.1, 10]$. Once fitted, the weather emulator $\mathcal{W}$ is used to generate $s = 100$ weather realizations at each prediction stage.

Table 1: Normalized weather variables and corresponding average coefficients of 36 OLS models.

Variable Type Description	**Unit**	**Coefficient for Mean Variable**	**Coefficient for Std.dev Variable**
Horizontal wind speed	m/s	0.028	0.035
Vertical wind speed	m/s	0.020	−0.007
Wind speed	m/s	0.003	0.012
Wind direction measurement 1	°	−0.005	−0.003
Wind direction measurement 2	°	0.005	−0.015
Ambient temperature	°C	0.007	0.001
Relative humidity	%	0.003	0.000
Precipitation	mm	0.004	−0.017
Latest turbine output	kW	0.974	

The wind farm simulator $\mathcal{M}$ is implemented with OLS regression model as discussed in (6) with $r = 10$. The model uses the 16 weather variables from $\mathcal{W}$ and y_{t-1}^{real}, resulting in a total of 17 predictors. The average OLS coefficients across 36 models are summarized in Table 1. The estimated average residual standard deviation is $\hat{\sigma} = 0.055$. We use UCB_{RF} acquisition function and GPR as a metamodel for the calibration task. To initialize the metamodel, 10 evenly-spaced values from $\Theta = [0.001, 0.1]$ were sampled and fitted, and then 10 additional evaluations were processed. To evaluate the performance of the proposed dynamic wake prediction framework, we compare it against two benchmark approaches:

- **Static wake**: The wake decay parameter is fixed to a constant value across all time steps. We use the representative constant 0.04, selected based on the prior study (Jain et al. 2023).
- **Dynamic deterministic wake**: This approach estimates the future wake parameter based solely on the most recent single observation $(x_{t-1}^{\text{real}}, y_{t-1}^{\text{real}})$, without simulating or forecasting future uncertainty.

Table 2: Normalized test loss (RMSE) comparison under different levels of wake variability ($\gamma = 0.1$).

Method	$\alpha = 0.1$ **(Rapid)**	$\alpha = 0.5$ **(Moderate)**	$\alpha = 0.9$ **(Steady)**
Static wake	0.0368	0.0304	0.0258
Dynamic deterministic wake	0.0358	0.0215	0.0115
Dynamic stochastic wake	**0.0347**	**0.0211**	**0.0113**

In this experiment, we set the smoothing parameter to $\gamma = 0.1$ in (8) and evaluate the framework under three different settings of $\alpha = [0.1, 0.5, 0.9]$. These α values represent different levels of variability in the wake decay behavior; $\alpha = 0.1$ corresponds to rapidly-changing wake trends, $\alpha = 0.5$ to moderately evolving, and $\alpha = 0.9$ to relatively steady, gradually changing wake decay trends. The test loss is evaluated using

RMSE score. The experimental results, summarized in Table 2, show the dynamic wake prediction method consistently achieves the lowest test loss across all α scenarios. The performance gap is most significant in settings where the wake decay parameter evolves more rapidly (i.e., $\alpha = 0.1$). This is particularly significant, as highly dynamic settings pose a real challenge for methods based on static assumptions or on a single current weather information, which tend to struggle to adapt to evolving trends.

Additionally, we conducted a side-experiment to demonstrate the effectiveness of our proposed framework in comparison with a jointly modeled approach for approximating $\mathbb{P}_{X_t,Y_t}$ using multi-output gaussian process regression (MOGP) as discussed in Section 2. Recall that in our nested framework, we have explicitly modeled the dependence between the weather covariates and turbine outputs through $\mathcal{M}$, while avoiding unnecessary assumptions about inter-variance correlations among weather features. In contrast, MOGP jointly models all weather covariates and turbine outputs as a multivariate function, relying on the estimated cross-covariance structure to capture dependencies.

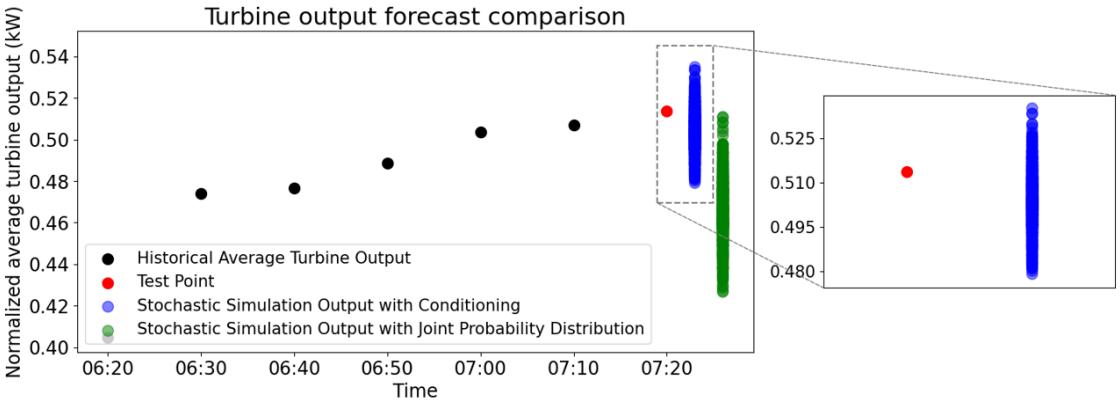

Figure 6: Comparison of turbine output forecasts using the proposed nested framework and joint modeling approach (MOGP) with $n = 6$ training time steps.

As depicted in Figure 6, when the number of training time steps is relatively small compared to the output dimension $m_x + m_y$, the model tends to prioritize the inter-variable dependencies over temporal structure, which results in negative bias and larger variance in predictions. To handle this, using MOGP would require a substantially large number of training time steps n as to balance the cross-covariance and temporal behavior. However, increasing n in MOGP significantly increases the computational complexity, which scales as $O(n^3(m_x + m_y)^3)$. In contrast, our proposed framework is well suited with smaller values of n with a reduced complexity of $O(m_x n^3)$. Furthermore, under full parallelization across the m_x independent GPRs, the overall complexity can further reduce to $O(n^3)$.

5 CONCLUSION

In this study, we proposed a dynamic framework for digital twin calibration that uses a nested stochastic simulation to produce robust calibrated parameters, particularly suited in rapidly evolving system. Our framework is informed by a case study for wake effect prediction to forecast the future wake behavior of the wind farm. Experimental results show that the proposed framework consistently produces lower prediction errors across various weather variability scenarios, which is particularly meaningful in highly dynamic conditions where wake decay parameters change abruptly over time. Future work will be dedicated to addressing our simplifying assumptions in dependence structure of X, aggregation of Y, smoothing, and the fixed number of replications. Lastly, we are interested in exploring alternative objective functions within the calibration process, such as the Conditional Value at Risk (CVaR). In such highly-dynamic weather environments, it is often preferable to adopt more conservative predictions to mitigate the risk of extreme prediction errors. This naturally casts the calibration problem into a broader simulation optimization

framework, where state-of-the-art simulation optimization methods such as ASTRO-DF (Shashaani et al. 2018) or stochastic kriging (Staum 2009) can be effectively leveraged.

ACKNOWLEDGMENTS

We acknowledge support by the National Science Foundation under Grants CMMI-2226347 and 2226348.

REFERENCES

Jain, P., S. Shashaani, and E. Byon. 2023. "Wake Effect Parameter Calibration with Large-Scale Field Operational Data Using Stochastic Optimization". *Applied Energy* 347:121426.

Jensen, N. O. 1983. "A Note on Wind Generator Interaction". Technical Report RISO-M-2411, Risø National Laboratory, Roskilde, Denmark.

Jeon, Y., and S. Shashaani. 2024. "Calibrating Digital Twins via Bayesian Optimization with a Root Finding Strategy". In *2024 Winter Simulation Conference (WSC)*, 335–346 https://doi.org/10.1109/WSC63780.2024.10838781.

Katikas, L., T. Kontos, P. Dimitriadis, and M. Kavouras. 2024. "A Raster-Based Multi-Objective Spatial Optimization Framework for Offshore Wind Farm Site-Prospecting". *ISPRS International Journal of Geo-Information* 13(11):409.

Park, R.-J., J.-H. Kim, B. Yoo, M. Yoon, and S. Jung. 2022. "Verification of Prediction Method Based on Machine Learning Under Wake Effect Using Real-Time Digital Simulator". *Energies* 15(24):9475.

Shakoor, R., M. Y. Hassan, A. Raheem, and Y.-K. Wu. 2016. "Wake Effect Modeling: A Review of Wind Farm Layout Optimization Using Jensen's Model". *Renewable and Sustainable Energy Reviews* 58:1048–1059.

Shashaani, S., F. S. Hashemi, and R. Pasupathy. 2018. "ASTRO-DF: A Class of Adaptive Sampling Trust-Region Algorithms for Derivative-Free Stochastic Optimization". *SIAM Journal on Optimization* 28(4):3145–3176.

Staid, A., P. Pinson, and S. D. Guikema. 2015. "Probabilistic Maximum-Value Wind Prediction for Offshore Environments". *Wind Energy* 18(10):1725–1738.

Staum, J. 2009. "Better Simulation Metamodeling: The Why, What, and How of Stochastic Kriging". In *2009 Winter Simulation Conference (WSC)*, 119–133 https://doi.org/10.1109/WSC.2009.5429320.

Vicente, W. B., R. Caire, and N. Hadjsaid. 2017, July. "Stochastic Simulations and Stability Analysis for Determining Maximum Wind Power Penetration in an Island Network". In *2017 IEEE Power & Energy Society General Meeting*, 1–5. Chicago, IL, USA.

Wang, D., M. Xu, Z. Guangming, F. Luo, J. Gao, and Y. Chen. 2025. "Enhancing Wind Power Forecasting Accuracy Through LSTM with Adaptive Wind Speed Calibration (C-LSTM)". *Scientific Reports* 15(1):5352.

Xiao, J., P. Wang, S. Huang, Q. Luo, W. Chen, and J. Wei. 2025. "Virtuality-Reality Combination Control for Wind Farm Maximum Power Generation with Wake Model Dynamic Calibration". *IEEE Transactions on Sustainable Energy* 16(2):1007–1020.

Zhang, X., D. Li, and X. Fu. 2024. "A Novel Wasserstein Generative Adversarial Network for Stochastic Wind Power Output Scenario Generation". *IET Renewable Power Generation* 18(16):3731–3742.

AUTHOR BIOGRAPHIES

YONGSEOK JEON is a Ph.D. student in the Edward P. Fitts Department of Industrial and Systems Engineering at North Carolina State University. His research interests include simulation and optimization. His e-mail address is yjeon@ncsu.edu.

SARA SHASHAANI is an Assistant Professor and Bowman Faculty Scholar in the Edward P. Fitts Department of Industrial and System Engineering at North Carolina State University. Her research interests include simulation optimization and uncertainty quantification in data-driven modeling and prediction. She is a co-creator of SimOpt. Her email address is sshasha2@ncsu.edu and her homepage is https://shashaani.wordpress.ncsu.edu/.

EUNSHIN BYON is a Professor and Richard Wilson Faculty Scholar in the Department of Industrial and Operations Engineering at the University of Michigan. Her research interests include data science, energy and sustainability, and quality and reliability engineering. Her email address is ebyon@umich.edu. Her website is https://ebyon.engin.umich.edu.

PRANAV JAIN recently completed his Ph.D. in Industrial Engineering at NC State University and is currently a Visiting Scholar North Carolina State University. He specializes in developing scalable algorithms and predictive models to improve operational efficiency. His email address is pjain23@ncsu.edu.