Received: 4 January 2023 | Revised: 3 May 2023 | Accepted: 16 June 2023

DOI: 10.1002/we.2851

RESEARCH ARTICLE

WILEY

Data-driven modelling of turbine wake interactions and flow resistance in large wind farms

Andrew Kirby[1] | François-Xavier Briol[2] | Thomas D. Dunstan[3] | Takafumi Nishino[1]

[1]Department of Engineering Science, University of Oxford, Oxford, UK

[2]Department of Statistical Science, University College London, London, UK

[3]Informatics Lab, UK MetOffice, Exeter, UK

Correspondence
Andrew Kirby, Department of Engineering Science, University of Oxford, Oxford OX1 3PJ, UK.
Email: andrew.kirby@trinity.ox.ac.uk

Funding information
None reported.

Abstract

Turbine wake and local blockage effects are known to alter wind farm power production in two different ways: (1) by changing the wind speed locally in front of each turbine and (2) by changing the overall flow resistance in the farm and thus the so-called farm blockage effect. To better predict these effects with low computational costs, we develop data-driven emulators of the 'local' or 'internal' turbine thrust coefficient C_T^* as a function of turbine layout. We train the model using a multi-fidelity Gaussian process (GP) regression with a combination of low (engineering wake model) and high-fidelity (large eddy simulations) simulations of farms with different layouts and wind directions. A large set of low-fidelity data speeds up the learning process and the high-fidelity data ensures a high accuracy. The trained multi-fidelity GP model is shown to give more accurate predictions of C_T^* compared to a standard (single-fidelity) GP regression applied only to a limited set of high-fidelity data. We also use the multi-fidelity GP model of C_T^* with the two-scale momentum theory (Nishino & Dunstan 2020, J. Fluid Mech. 894, A2) to demonstrate that the model can be used to give fast and accurate predictions of large wind farm performance under various mesoscale atmospheric conditions. This new approach could be beneficial for improving annual energy production (AEP) calculations and farm optimization in the future.

KEYWORDS
blockage effects, Gaussian process, large eddy simulation, machine learning, turbine layout, wake effects

1 | INTRODUCTION

The installed capacity of wind energy is projected to increase rapidly in the next decades. A major challenge in the optimization of wind farm design is the accurate prediction of wind farm performance.[1] Existing wind farm models struggle to make accurate predictions of wind farm power production. This is partly because the 'global blockage effect' reduces the velocity upstream of large farms and hence the energy yield.[2] It remains unclear how global blockage should be modelled and this is the subject of a large-scale field campaign.[3]

Wind farms are typically modelled using engineering 'wake' models. These models predict the velocity deficit in the wakes behind turbines.[4,5] To account for interactions between multiple turbines, the wake velocity deficits are superposed.[6,7] Simple wake models can give predictions of wind farm performance with very low computational cost (10^{-3} CPU hours per simulation[1]). However, wake models do not account for the

response of the atmospheric boundary layer (ABL) to the wind farm, which is likely to be important for large wind farms.[8] It has been found that wake models compare poorly to large eddy simulations (LES) of large wind farms.[9]

Wind farms are also modelled in numerical weather prediction (NWP) models using farm parameterization schemes. In these parameterizations, farms are often modelled as a momentum sink and a source of turbulent kinetic energy.[10] Wake interactions cannot be adequately predicted using these schemes. A new scheme was proposed,[11] which uses a correction factor to model turbine interactions. More recently, data-driven approaches have been proposed[12] to model these effects in wind farm parameterizations.

Data-driven modelling of wind farm flows is a promising new approach.[13] Data from high-fidelity simulations with complex flow physics can be used to make predictions with low computational cost. Recent studies have applied machine learning techniques to data from a single turbine or from an existing wind farm. The data for these studies are from measurements,[14–17] LES[18] or Reynolds-Averaged Navier-Stokes (RANS) simulations.[19–21] A limitation of these approaches is that they are not generalizable to different turbine layouts unless they rely on wake superposition techniques to model farm flows. Another approach is modelling the effect of turbine layout using geometric parameters[17] or using the layout as a graph input to a neural network.[22,23] However, these alternative approaches may struggle to fully capture the complex two-way interaction with the ABL as it seems impractical to prepare a data set that covers the entire range of scales involved in wind farm flows.[1]

The problem of modelling wind farm flows can be split into 'internal' turbine-scale and 'external' farm-scale problems.[24] The 'internal' problem is to determine a 'local' or 'internal' turbine thrust coefficient, C_T^*, which represents the flow resistance inside a wind farm, that is, how the turbine thrust changes with wind speed within the farm. Nishino[25] proposed an analytical model for an upper limit of C_T^* by using an analogy to the classic Betz analysis. This analytical model is a function of turbine-scale induction factor but is independent of turbine layout and wind direction. Previous studies[8,24,25] showed that C_T^* is usually lower than the limit predicted by Nishino's model and can vary significantly with turbine layout due to wake and turbine blockage effects.

The aim of this study is to develop statistical emulators of C_T^* as a function of turbine layout and wind direction. The novelty of this approach is that we are modelling the effect of turbine wake interactions on C_T^* rather than turbine power. Both turbine-scale flows (e.g., wake effects) and farm-scale flows (e.g., farm blockage and mesoscale atmospheric response) affect turbine power within a farm. Therefore, to create an emulator of turbine power, either (1) a very large set of expensive data such as finite-size wind farm LES is needed, which covers a range of large-scale atmospheric conditions or (2) the model would not be generalizable to different mesoscale atmospheric responses. An emulator of C_T^* is however applicable to different atmospheric responses modelled separately, following the concept of the two-scale momentum theory.[8,24]

In Section 2, we give the definitions of key wind farm parameters in the two-scale momentum theory.[24] Section 3 summarizes the methodology of the LES and wake model simulations, followed by the machine learning approaches to develop the emulators in Section 4. In Section 5, we present the results from the trained emulators. These results are discussed in Section 6 and concluding remarks are given in Section 7.

2 | TWO-SCALE MOMENTUM THEORY

By considering the conservation of momentum for a control volume with and without a large wind farm over the land or sea surface, the following non-dimensional farm momentum (NDFM) equation can be derived[24]:

$$C_T^* \frac{\lambda}{C_{f0}} \beta^2 + \beta^\gamma = M \tag{1}$$

where β is the farm wind-speed reduction factor defined as $\beta \equiv U_F/U_{F0}$ (with U_F defined as the average wind speed in the nominal wind farm-layer of height H_F, and U_{F0} is the farm-layer-averaged speed without the wind farm present); λ is the array density defined as $\lambda \equiv nA/S_F$ (where n is the number of turbines in the farm, A is the rotor swept area and S_F is the farm footprint area); C_T^* is the internal turbine thrust coefficient defined as $C_T^* \equiv \sum_{i=1}^{n} T_i / \frac{1}{2}\rho U_F^2 nA$ (where T_i is thrust of turbine i in the farm and ρ is the air density); C_{f0} is the natural friction coefficient of the surface defined as $C_{f0} \equiv \langle \tau_{w0} \rangle / \frac{1}{2}\rho U_{F0}^2$ (where τ_{w0} is the bottom shear stress without the farm present); γ is the bottom friction exponent defined as $\gamma \equiv \log_\beta(\langle \tau_w \rangle / \tau_{w0})$ (where $\langle \tau_w \rangle$ is the bottom shear stress averaged across the farm); M is the momentum availability factor defined as

$$M = \frac{\text{Momentum supplied by the atmosphere to the farm site } \textbf{with} \text{ turbines}}{\text{Momentum supplied by the atmosphere to the farm site } \textbf{without} \text{ turbines}}. \tag{2}$$

noting that this includes pressure gradient forcing, Coriolis force, net injection of streamwise momentum through top and side boundaries and time-dependent changes in streamwise velocity.[24] The height of the farm-layer, H_F, is used to define the reference velocities U_F and U_{F0}. Equation (1) is valid so long as the same of H_F is used for both the internal and external problem. H_F is typically between $2H_{hub}$ and $3H_{hub}$[8] (where H_{hub} is the turbine hub-height) and in this study we use a fixed definition of $H_F = 2.5H_{hub}$.

Patel[26] used an NWP model to demonstrate that, for most cases, M varied almost linearly with β (for a realistic range of β between 0.8 and 1). Therefore, M can be approximated by

$$M = 1 + \zeta(1-\beta) \tag{3}$$

where ζ is the 'momentum response' factor or 'wind extractability' factor. Patel[26] found ζ to be time-dependent and vary between 5 and 25 for a typical offshore site (note that $\zeta = 0$ corresponds to the case where momentum available to the farm site is assumed to be fixed, i.e., $M = 1$).

Nishino[25] proposed an analytical model for C_T^* given by

$$C_T^* = 4\alpha(1-\alpha) = \frac{16C_T'}{(4+C_T')^2} \tag{4}$$

where α is the turbine-scale wind speed reduction factor defined as $\alpha \equiv U_T/U_F$ (U_T is the streamwise velocity averaged over the rotor swept area) and $C_T' \equiv T/\frac{1}{2}\rho U_T^2 A$ is a turbine resistance coefficient describing the turbine operating conditions.

For a given farm configuration at a farm site (i.e., for given set of C_T^*, λ, C_{f0}, γ and ζ), the farm wind-speed reduction factor β can be calculated using Equation (1). The (farm-averaged) power coefficient C_p is defined as $C_p \equiv \sum_{i=1}^{n} P_i/\frac{1}{2}\rho U_{F0}^3 nA$ (P_i is power of turbine i in the farm). Using the calculated value of β, C_p can be calculated by using the expression,

$$C_p = \beta^3 C_p^* \tag{5}$$

where C_p^* is the (farm-averaged) 'local' or 'internal' turbine power coefficient defined as $C_p^* \equiv \sum_{i=1}^{n} P_i/\frac{1}{2}\rho U_F^3 nA$.

3 | WIND FARM SIMULATIONS

In this study, we model wind farms as arrays of actuator discs (or aerodynamically ideal turbines operating below the rated wind speed). This is because, in real wind farms, the effects of turbine wake interactions on the farm performance are most significant when they operate below the rated wind speed. The 'internal' thrust coefficient C_T^* is an important wind farm parameter, which includes the effect of turbine interactions (including both wake and local blockage effects). In this study, we will be modelling the effect of turbine layout on C_T^* for aligned turbine layouts with various wind directions and a fixed turbine resistance of $C_T' = 1.33$. We chose $C_T' = 1.33$ because it leads to a turbine induction factor of 1/4, which is close to a typical value for modern large wind turbines. As such we will be considering

$$C_T^* = f(S_x, S_y, \theta) \tag{6}$$

where S_x is the turbine spacing in the x direction, S_y is the turbine spacing in the y direction and θ is the wind direction relative to the x direction (see Figure 1A). However, the true function C_T^* cannot be easily evaluated so we will instead investigate C_T^* using computer codes. One computer code we will use is LES (see Section 3.1) to estimate C_T^*

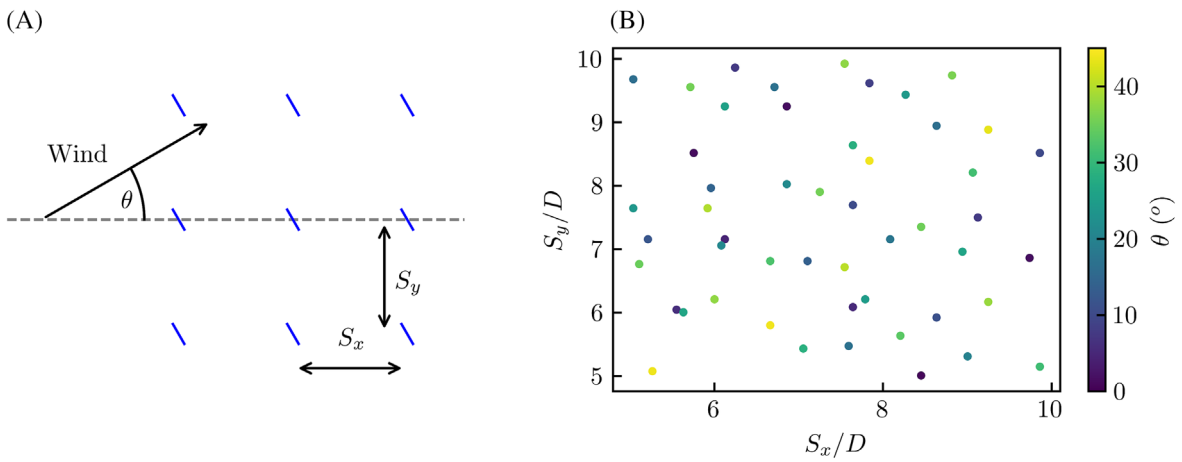

FIGURE 1 Design of numerical experiments: (A) input parameters and (B) maximin design of LES.

$$C^*_{T,LES} = f_{LES}(S_x, S_y, \theta). \tag{7}$$

We assume that the function f_{LES} is close to the true function f because of the accuracy of LES to model wind farm flows. We will also use a wake model (see Section 3.2) to provide cheap approximations of C^*_T according to

$$C^*_{T,wake} = f_{wake}(S_x, S_y, \theta). \tag{8}$$

Engineering problems are often investigated using complex computer models. Evaluating the output of such computer models for a given input can be very computationally expensive. Therefore, a common objective is to create a cheap statistical model of the expensive computer model; this is commonly known as emulation of computer models.[27,28] In this study, we aim to develop a statistical emulator, which can cheaply emulate f_{LES}.

The emulators will only be valid for aligned layouts of wind turbines and for a given turbine resistance (here we use $C'_T = 1.33$). We consider the input parameters for a realistic range of turbine spacings[1]: $S_x \in [5D, 10D]$, $S_y \in [5D, 10D]$ and $\theta \in [0^o, 45^o]$ where D is the diameter of the turbine rotor swept area. In this study, D is set as 100 m and the turbine hub height is also 100 m. We only need to consider wind directions of $\theta \in [0^o, 45^o]$ because of symmetry in the aligned turbine layouts. If θ is negative then the turbine layout given by (S_x, S_y, θ) is exactly the same as $(S_x, S_y, -\theta)$. When $\theta > 45^\circ$, then (S_x, S_y, θ) and $(S_y, S_x, 90^o - \theta)$ give identical layouts.

In this study, we build several emulators to predict f_{LES}. The models are trained using data from low-fidelity (wake model) and high fidelity (LES) wind farm simulations. One evaluation of $C^*_{T,wake}$ takes approximately 130 s on a single CPU and $C^*_{T,LES}$ requires around 400 CPU hours on a supercomputer. We use a space filling maximin design[29,30] to select training points in the parameter space. The maximin algorithm selects points, which maximizes the minimum distance to other points and to the boundaries. This provides a good coverage of the domain, which ensures that the emulators can give good predictions across the whole of the domain.[31] Figure 1B shows the LES training points in the parameter space.

3.1 | Large eddy simulations

This study uses the data from 50 high-fidelity (LES) simulations of wind farms published in a previous study.[8] Here, we give a brief summary of the LES methodology. The LES models a neutrally stratified atmospheric boundary layer over a periodic array of actuator discs, which face the wind direction θ and exert uniform thrust. The resolution is 24.5 m in the horizontal directions (4 points across the rotor diameter) and 7.87 m in the vertical. This is a coarse horizontal resolution; however, using a correction factor for the turbine thrust[32] makes the $C^*_{T,LES}$ values insensitive to horizontal resolution.[8] For all simulations, the vertical domain size was fixed at 1 km and the horizontal extent varied with turbine layout but was at least 3.14 km. The horizontal boundary conditions were periodic (essentially an infinitely-large wind farm). The bottom boundary used a no-slip condition with the value of eddy viscosity specified following the Monin-Obukhov similarity theory for a surface roughness length of $z_0 = 1 \times 10^{-4}$m. The top boundary had a slip condition with zero vertical velocity. The flow was driven by a pressure gradient forcing, which was constant and in the direction θ throughout the domain. Figure 2 shows the instantaneous and time-averaged hub height velocities from one wind farm LES. See the original paper[8] for further details of the LES.

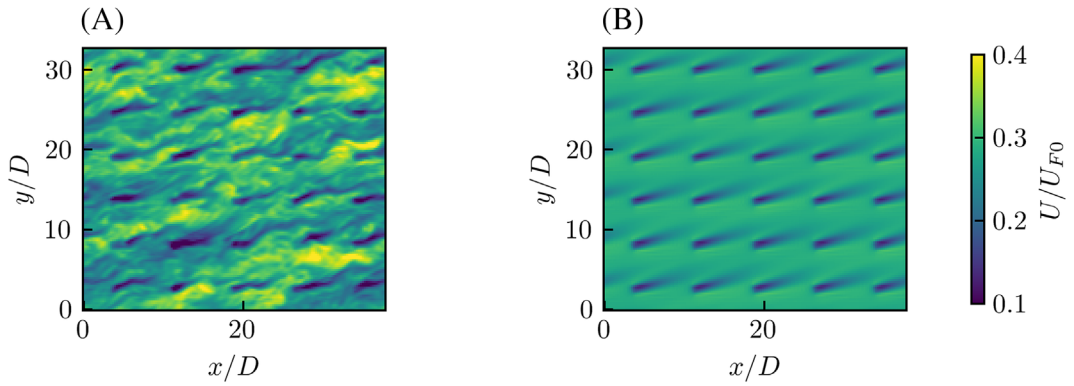

FIGURE 2 LES (A) instantaneous and (B) time-averaged flow fields over a periodic turbine array ($S_x/D = 7.59$, $S_y/D = 5.47$ and $\theta = 37.6^\circ$).

3.2 | Wake model simulations

Wake models are a cheap low-fidelity approach to modelling wind farm aerodynamics compared to expensive high-fidelity LES simulations.[1] We use the wake model proposed by Niayafar and Porté-Agel[33] to evaluate $C^*_{T,wake}$ as a cheap approximation of C^*_T. We use the Python package PyWake[34] to implement the wake model. The turbine thrust coefficient C_T is needed as an input for the wake model. We use the value of C^*_T predicted by Equation (4) as the value of C_T. For the turbine operating conditions used in this study ($C'_T = 1.33$), the wake model has C_T equal to 0.75 for all turbines. To model actuator discs, we consider a hypothetical turbine, which has a constant C_T for all wind speeds. We calculate $C^*_{T,wake}$ for a single turbine at the back of a large farm (marked X in Figure 3). The farm simulated using the wake model is 10 km long in the streamwise direction and 4 km long in the cross-streamwise direction. The farm size was chosen so that C^*_T no longer varied with increasing farm size. The wake growth parameter is calculated using $k^* = 0.38I + 0.004$ where I is the local streamwise turbulence intensity. The local streamwise turbulence intensity is estimated using the model proposed by Crespo and Hernández.[35] The background turbulence intensity (TI) is set as a typical value of 10%.

The velocity incident to the turbine is calculated by averaging the velocity across the disc area. We use a 4×3 cartesian grid with Gaussian quadrature coordinates and weights on the disc to average the velocity. The disc-averaged velocity, U_T is then calculated by multiplying the averaged incident velocity by $(1-a)$ where a is the turbine induction factor set by the value of C'_T (using the expression $a = C'_T/(4+C'_T)$). To calculate the farm-average velocity, U_F, we average the velocity across a volume around the single turbine. The volume has dimensions of S_y in the y direction, S_x in the x direction and 250 m in the z direction (the height of the nominal farm layer used in the previous LES study[8]). To calculate the average velocity, we discretize the volume into 200 points in the horizontal directions and 20 points in the vertical. This was sufficient for the calculation of $C^*_{T,wake}$ to not vary with further discretization. Figure 3 shows an example of the farm layout for the wake model simulations.

4 | MACHINE LEARNING METHODOLOGY

4.1 | Gaussian process regression

We will use Gaussian process (GP) regression[36] to build statistical emulators of f_{LES}. A Gaussian process is a stochastic process $g \sim \mathcal{GP}(m,k)$ described by a mean function $m(v) = \mathbb{E}[g(v)]$ and a covariance function $k(v,v') = \mathbb{E}[(g(v)-m(v))(g(v')-m(v'))]$. In our case, $v = (S_x, S_y, \theta)$. We will use such a stochastic process as a model of f_{LES}, the true mapping from v to $C^*_{T,LES}$. Each realization from this process will therefore be a function, which could plausibly represent this mapping. The mean function represents the expected output value at an input $v = (S_x, S_y, \theta)$. The covariance function gives the covariance between output values at v and v'. Examples of covariance functions include squared exponential, rational quadratic and periodic functions.[36] Different covariance functions will give differently shaped GPs. For example, the squared exponential covariance function will give very smooth GPs whereas the periodic function will give GPs with a periodic structure. Other types of structure, for example, symmetry, can also be encoded in the covariance function. Therefore, the expected shape (e.g., smoothness) of the expected relationship and any properties (e.g., discontinuities or symmetries) need to be considered when choosing a covariance function for GP regression.

Let $V = (v_1, \ldots, v_n)^T$ be a collection of design points then $m_V = (m(v_1), \ldots, m(v_n))^T$ is the mean vector and $k_{VV} = (k(v_i, v_j))$ is the covariance matrix. We will start by positing a GP model with mean m and covariance k (called the 'prior GP'), then condition this GP on LES observations; the

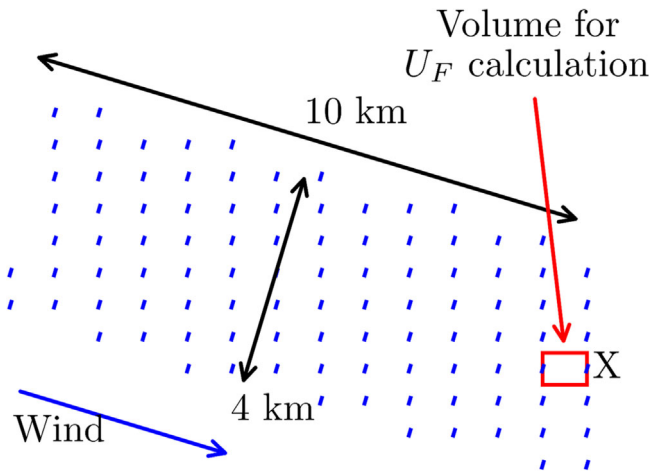

FIGURE 3 Example of wind farm layout for wake model simulations.

outcome is a new GP (called the 'posterior GP'). This gives the posterior distribution $g|V, C^*_{T,LES} \sim \mathcal{GP}(\overline{m}_{\sigma^2}, \overline{k}_{\sigma^2})$. $\overline{m}_{\sigma^2}$ is the posterior mean function given by $\overline{m}_{\sigma^2}(v) = m(v) + k_{vV}(k_{VV} + \sigma^2 I_{n\times n})^{-1}(C^*_{T,LES} - m_V)$ where $k_{vV} = (k(v, v_1), \ldots, k(v, v_n))$ and $I_{n\times n}$ is the identity matrix of size n. The posterior mean function $\overline{m}_{\sigma^2}$ is used to make predictions at $v = (S_x, S_y, \theta)$. The posterior covariance function $\overline{k}_{\sigma^2}$ quantifies the uncertainty in our prediction at $v = (S_x, S_y, \theta)$. The posterior covariance function is given by $\overline{k}_{\sigma^2}(v, v') = k(v, v') - k_{vV}(k_{VV} + \sigma^2 I_{n\times n})^{-1} k_{Vv'}$.

Often in GP regression a zero prior mean is used. However, using an informative prior mean can improve the accuracy of the trained model. By using a prior mean, many of the trends in f_{LES} can be incorporated into our model prior to making expensive evaluations of $C^*_{T,LES}$. Therefore, after training our model will likely better describe the true relationship between S_x, S_y, θ and f_{LES}. In this study, we will use both $C^*_{T,wake}$ and the analytical model of C^*_T as the prior mean for the standard GP regression. For the wake model prior mean, we also vary the specified ambient TI input parameter.

We expect f_{LES} to be a smooth function of input variables S_x, S_y and θ, and to vary more rapidly with θ than S_x or S_y. Therefore, we will use an anisotropic squared-exponential covariance function,

$$k(v, v') = \sigma_f^2 \exp\left(-\frac{(S_x - S'_x)^2}{2l_1^2}\right) \exp\left(-\frac{(S_y - S'_y)^2}{2l_2^2}\right) \exp\left(-\frac{(\theta - \theta')^2}{2l_3^2}\right) \tag{9}$$

where $\sigma_f^2 > 0$ is the signal variance hyperparameter and $l_i > 0$ is the lengthscale hyperparameter for each dimension. This is also called an ARD (automatic relevance detection) kernel. If we consider $v = v'$ then we can see that σ_f^2 determines the variance of $g(v)$. Therefore, σ_f^2 determines the prior uncertainty the model has about the value of $g(v)$. As the lengthscale hyperparameter l_i gets smaller then $k(v, v')$ decreases (for $v \neq v'$). Equally if l_i increases then $k(v, v')$ will also increase. A GP with a small l_i will therefore vary more rapidly across the parameter space in the ith dimension.

Due to numerical issues associated with the matrix inversion/linear system solve operations in the formulae for the posterior GP, it is common to add a nugget $\sigma^2 > 0$ to the kernel matrix. The hyperparameters σ_f^2 and l_i are selected automatically during the fitting process by maximizing the log marginal likelihood.[36] This approach selects the model that maximizes the fit to the data.

Figure 4 shows the impact of the hyperparameters in an example GP regression setting (using the squared exponential covariance function). The mean function and 95% credible interval (±1.96 times the standard deviation) prior to fitting are shown in Figure 4A with 3 GPs drawn from

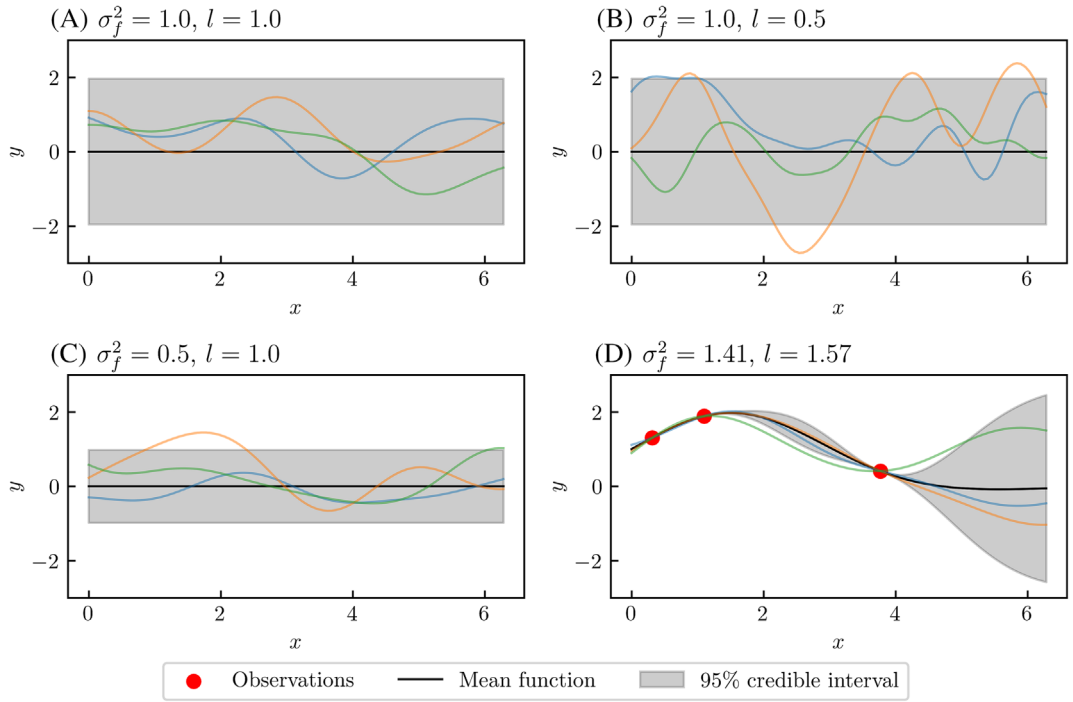

FIGURE 4 Demonstration of basic GP regression: Panel (A) shows the prior mean and covariance function prior to fitting with 3 GPs drawn from the distribution shown in color; panel (B) shows the effect of decreasing the lengthscale hyperparameter; panel (C) shows the effect of variance hyperparameter; and panel (D) shows the posterior mean and covariance functions.

the distribution (colored lines). The effect of decreasing the lengthscale hyperparameter l_i is shown in Figure 4B. The prior mean and 95% credible interval are unchanged however the example GPs drawn vary more rapidly because of the shorter lengthscale. Figure 4C shows the same setup as Figure 4A but with a smaller value of σ_f^2. The example GPs still vary slowly but the magnitude of the variations is now smaller. Figure 4D shows the GPs conditioned on observations with hyperparameters selected by maximizing the log marginal likelihood.

4.2 | Non-linear multi-fidelity Gaussian process regression

In many applications there are several computational models available. These models can have varying accuracies and computational costs. The models that are more computationally expensive typically give more accurate predictions. The GP regression framework can be extended to combine information from low and high-fidelity models.[37] This type of modelling uses the low-fidelity observations to speed up the learning process and the high-fidelity observations to ensure accuracy. In our scenario, we will combine evaluations of from a low-fidelity ($C^*_{T,wake}$) and a high-fidelity ($C^*_{T,LES}$) model. Note that for the multi-fidelity models in this study we set the ambient TI to 10% for the wake model and use a zero prior mean. We will keep the number of high-fidelity training points fixed at 50 and we will vary the number of low-fidelity training points used.

We combine information from our high and low-fidelity models using a nonlinear information fusion algorithm.[38] The framework is based on the autoregressive multi-fidelity scheme given by:

$$g_{high}(v) = \rho(g_{low}(v)) + \delta(v) \tag{10}$$

where $g_{low}(v)$ is a model with a GP denoted f_{wake} and $g_{high}(v)$ is a model with a GP denoted f_{LES}. ρ is a model with a GP that maps the low-fidelity output to the high-fidelity output and $\delta(v)$ is a model with a GP, which is a bias term. The non-linear multi-fidelity framework can learn non-linear space-dependent correlations between models of different accuracies. To reduce the computational cost and complexity of implementation, the autoregressive scheme given by Equation (10) is simplified. Firstly, the GP prior $g_{low}(v)$ is replaced by the GP posterior $g_{low,*}(v)$ and secondly the GPs ρ and δ are assumed to be independent. Equation (10) can then be summarized as

$$g_{high}(v) = h_{high}(v, g_{low,*}(v)) \tag{11}$$

where h_{high} is a model with a GP that has both v and $g_{low,*}(v)$ as inputs. More details of h_{high} and the implementation of the multi-fidelity framework are given in Perdikaris et al.[38]

Figure 5 shows an example of how a multi-fidelity GP can outperform a standard GP regression. We implement the non-linear multi-fidelity framework using the 'emukit' package.[39] We first maximize the log marginal likelihood whilst keeping the Gaussian noise variance fixed at a low value of 1×10^{-6}. The fitting process is then repeated whilst allowing the Gaussian noise variance to be optimized too. This is to prevent a high noise local optima from being selected.

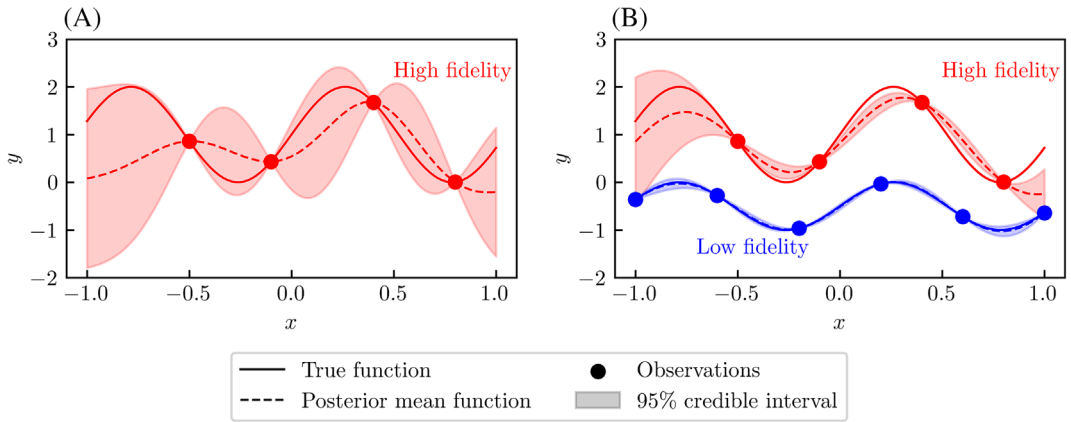

FIGURE 5 Demonstration of (A) basic GP regression and (B) multi-fidelity GP regression. In this example, $f(x) = 1 + sin(6x)$ for the high-fidelity data and $f(x) = -0.5 + 0.5sin(6x)$ for the low-fidelity data.

5 | RESULTS

In this study, we build various statistical emulators of f_{LES} using different techniques and compare the performance. A summary of the techniques is shown in the list below:

1. Standard Gaussian process regression (see Section 4.1)
   a. GP-analytical-prior: Gaussian process using analytical model (Equation 4) prior mean
   b. GP-wake-TI10-prior: Gaussian process using wake model (Section 3.2) with ambient TI=10% prior mean
   c. GP-wake-TI1-prior: Gaussian process using wake model with ambient TI=1% prior mean
   d. GP-wake-TI5-prior: Gaussian process using wake model with ambient TI=5% prior mean
   e. GP-wake-TI15-prior: Gaussian process using wake model with ambient TI=15% prior mean
2. Non-linear multi-fidelity Gaussian process regression (see Section 4.2)
   a. MF-GP-nlow500: multi-fidelity Gaussian process using 500 low-fidelity training points
   b. MF-GP-nlow250: multi-fidelity Gaussian process using 250 low-fidelity training points
   c. MF-GP-nlow1000: multi-fidelity Gaussian process using 1000 low-fidelity training points

The code and data used to produce the results in this section is available open-access at the following GitHub repository: https://github.com/AndrewKirby2/ctstar_statistical_model.

5.1 | Performance of standard GP regression

We first assessed the accuracy of the standard GP models (Section 4.1) by performing leave-one-out cross-validation (LOOCV). This is a method of estimating the accuracy of a statistical model when making predictions on data not used to train the model. We trained our model on 49 of the 50 training points and then calculated the prediction accuracy for the single high-fidelity data point, which is excluded from the training set. This is then repeated for all data points in turn, and we took the average accuracy as an estimate of the model test accuracy. The standard GP models were implemented using the 'GPy' package.[40]

The standard GP gave accurate predictions of f_{LES} with average errors of less than 2%. Table 1 shows the accuracy of the standard GP models compared to the analytical and wake models. We calculated the errors by using the expression $|\overline{m}_{\sigma^2} - C^*_{T,LES}|/0.75$ where $\overline{m}_{\sigma^2}$ is the posterior mean function of the emulator. The reference value for C^*_T of 0.75 was chosen because this is the prediction from the analytical model. Both GP models give similar maximum errors of approximately 6%. Using the wake model as a prior mean gave a lower mean absolute error of 1.26%. The GP models reduced the average prediction error and significantly reduced the maximum error compared to the wake model and analytical model of C^*_T.

The model GP-wake-TI10-prior has a high degree of confidence when making predictions in regions of the parameter space. Figure 6 shows the square root of the posterior covariance function $\overline{k}_{\sigma^2}$, which quantities the uncertainty of the emulator. The uncertainty is uniform throughout the parameter space with regions of slightly higher uncertainty at $\theta = 0^\circ$ and 45°.

We also assessed the sensitivity of the model accuracy to the ambient TI used in the wake model prior mean. Figure 7 shows the impact of ambient TI on the wake model prior mean and the fitted GP model. Increasing the ambient TI increased the value of $C^*_{T,wake}$. This is because of the enhanced wake recovery behind wind turbines. Increasing the ambient TI in the wake model results in $C^*_{T,wake}$ overpredicting $C^*_{T,LES}$. The MAE from the LOOCV procedure for each fitted GP is shown in the bottom right corner.

The fitted GPs became more accurate when the wake model ambient TI was increased. Increasing the ambient TI for the wake model causes the wakes to recover faster. The wakes become shorter in the streamwise direction and wider in the spanwise direction. As such, $C^*_{T,wake}$ becomes less sensitive to the turbine layout. When an ambient TI of 1% and 5% is used for the wake model, $C^*_{T,wake}$ is more sensitive to turbine layout than $C^*_{T,LES}$ (Figure 7A,B). When the ambient TI is increased to 10% and above, the relationship between $C^*_{T,wake}$ and $C^*_{T,LES}$ becomes simpler (Figure 7C,D). This seems to explain why the fitted GPs become more accurate.

TABLE 1 Accuracy of models for C^*_T prediction.

Model	MAE (%)	Maximum error (%)
GP-analytical-prior	1.87	6.09
GP-wake-TI10-prior	1.26	6.11
Analytical model	5.26	22.0
Wake model (TI=10%)	4.60	9.28

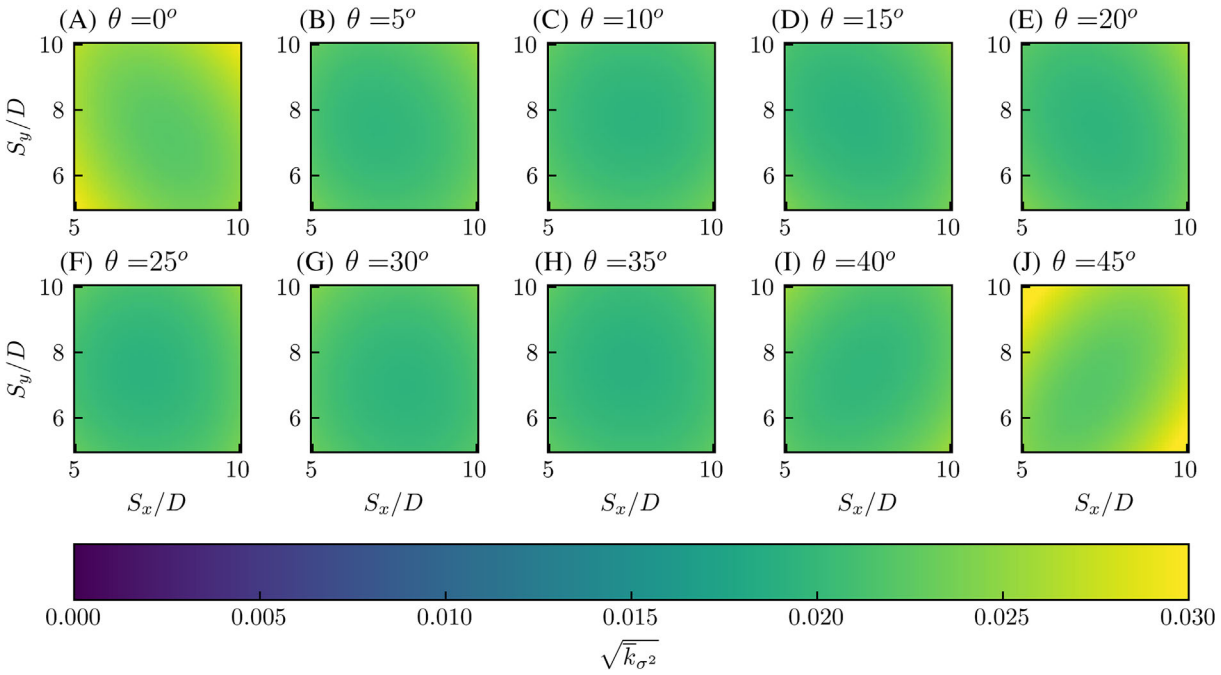

FIGURE 6 Posterior variance function of GP-wake-TI10-prior model.

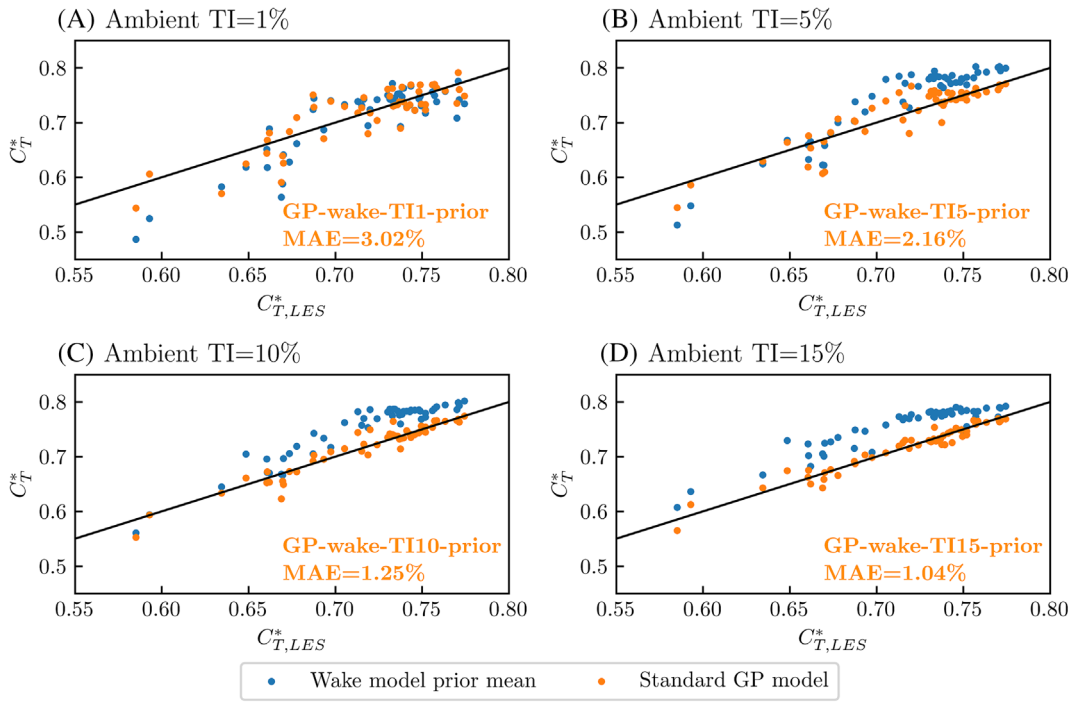

FIGURE 7 Sensitivity of fitted GP models to the ambient TI chosen for wake model prior means.

5.2 | Performance of non-linear multi-fidelity GP regression

We then assessed the accuracy of the multi-fidelity GP models (Section 4.2). All models used the 50 high-fidelity ($C^*_{T,LES}$) training points and a varying number of low-fidelity ($C^*_{T,wake}$) training points (using an ambient TI of 10% for $C^*_{T,wake}$). The results from LOOCV are shown in Table 2. For the LOOCV, we train our model on 49 out of the 50 high-fidelity data points and all low-fidelity data points. Then we average the error in predicting

the high-fidelity data point left of the training set and repeat this in turn for data points. Increasing the number of low-fidelity training points from 250 to 500 reduced the mean and maximum error. However, increasing this to 1000 low-fidelity training points did not increase accuracy and increased the fitting and prediction time. This is because the number of high-fidelity training points is fixed. There is a threshold where the model of the relationship between f_{LES} and f_{wake}, denoted ρ, limits the final accuracy of the emulator of f_{LES}.

The posterior mean $\overline{m}_{\sigma^2}$ of $g_{low}(v)$ is an emulator of f_{wake} and $g_{high}(v)$ is an emulator of f_{LES}. Figure 8 gives the predictions from the posterior mean of $g_{high}(v)$ (for MF-GP-nlow500). The lowest $\overline{m}_{\sigma^2}$ values were for a wind direction of $\theta = 0^\circ$. $\overline{m}_{\sigma^2}$ increased rapidly with θ reaching a maximum of slightly over 0.75 at $\theta = 10^\circ$. For large values of θ (above $\theta = 25^\circ$), there were local minima in $\overline{m}_{\sigma^2}$, which appear in Figure 8 as diagonal strips of low $\overline{m}_{\sigma^2}$ values. The main diagonal strip occurs along the line of $S_y = S_x \tan(\theta)$. There are two smaller strips either side of with positions given by $S_y = 2\tan(\theta)$ and $S_y = 0.5\tan(\theta)$ (this is discussed further in Section 6).

The uncertainty the model MF-GP-nlow500 has in predicting f_{LES} is shown in Figure 9. The model uncertainty is uniform throughout the parameter space with slightly higher values at $\theta = 0^\circ$ and 45°. Compared to the posterior variance of GP-wake-TI10-prior (shown in Figure 6) the uncertainty is lower. By incorporating information from $C^*_{T,wake}$, the multi-fidelity GP model has more confidence about predicting f_{LES}.

The prediction errors from the LOOCV (for MF-GP-nlow500) are shown in Figure 10. The box plot of prediction errors in Figure 10A shows that this model had no significant bias whereas both the wake and analytical models systemically overestimated $C^*_{T,LES}$. Figure 10B–D shows that for the statistical model there appears to be no part of the parameter space that had larger errors.

The multi-fidelity approach used in this study builds a statistical model of both the low-fidelity (f_{wake}) and high-fidelity (f_{LES}) model. We can use the posterior means of $g_{low}(v)$ and $g_{high}(v)$ to see the differences between the wake model and LES. The posterior mean for both models are shown in Figure 11. For the wake model, the change in $\overline{m}_{\sigma^2}$ with θ is greater than for the LES (especially between $\theta = 0^\circ$ and 10°). For larger values of θ, there is a larger difference in $\overline{m}_{\sigma^2}$ between waked and unwaked layouts for the low-fidelity model compared to the high-fidelity one. This suggests than the wake model is more sensitive to changes in wind directions than the LES.

TABLE 2 Performance of the multi-fidelity Gaussian process models.

Model	MAE (%)	Maximum error (%)	Training time (s)	Prediction time (s)
MF-GP-nlow250	1.46	7.12	6.15	0.00157
MF-GP-nlow500	0.828	3.75	9.73	0.00167
MF-GP-nlow1000	0.866	3.55	26.8	0.00236

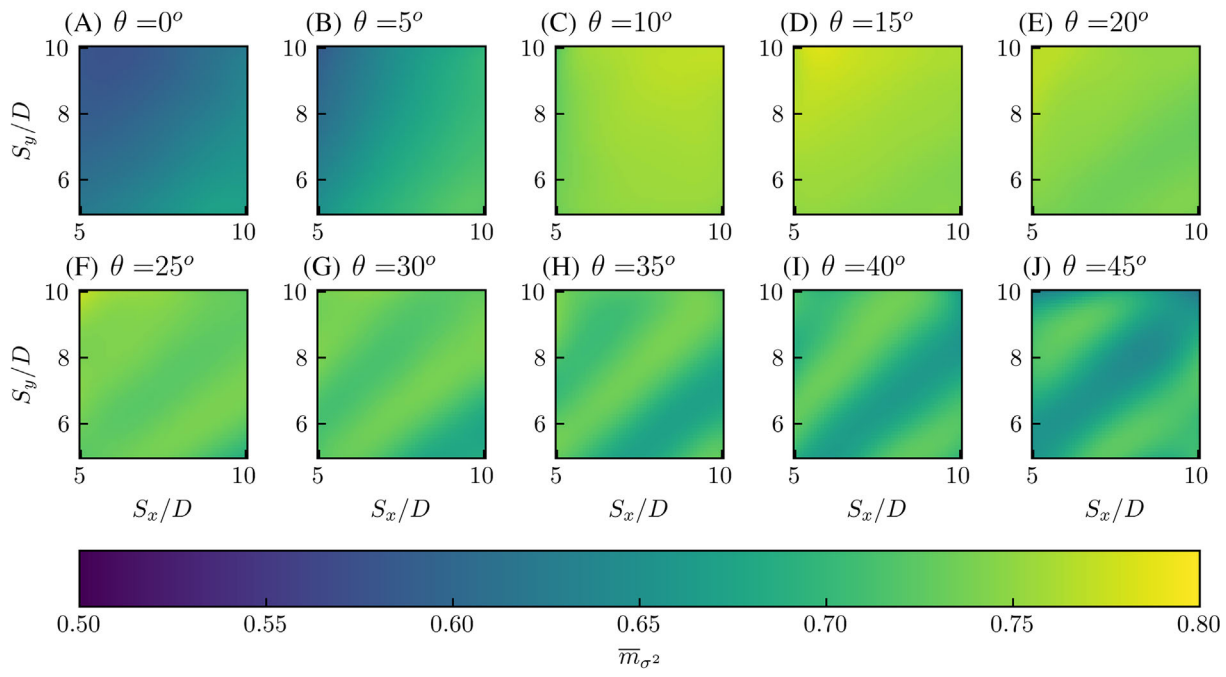

FIGURE 8 Posterior mean function for $g_{high}(v)$ of MF-GP-nlow500.

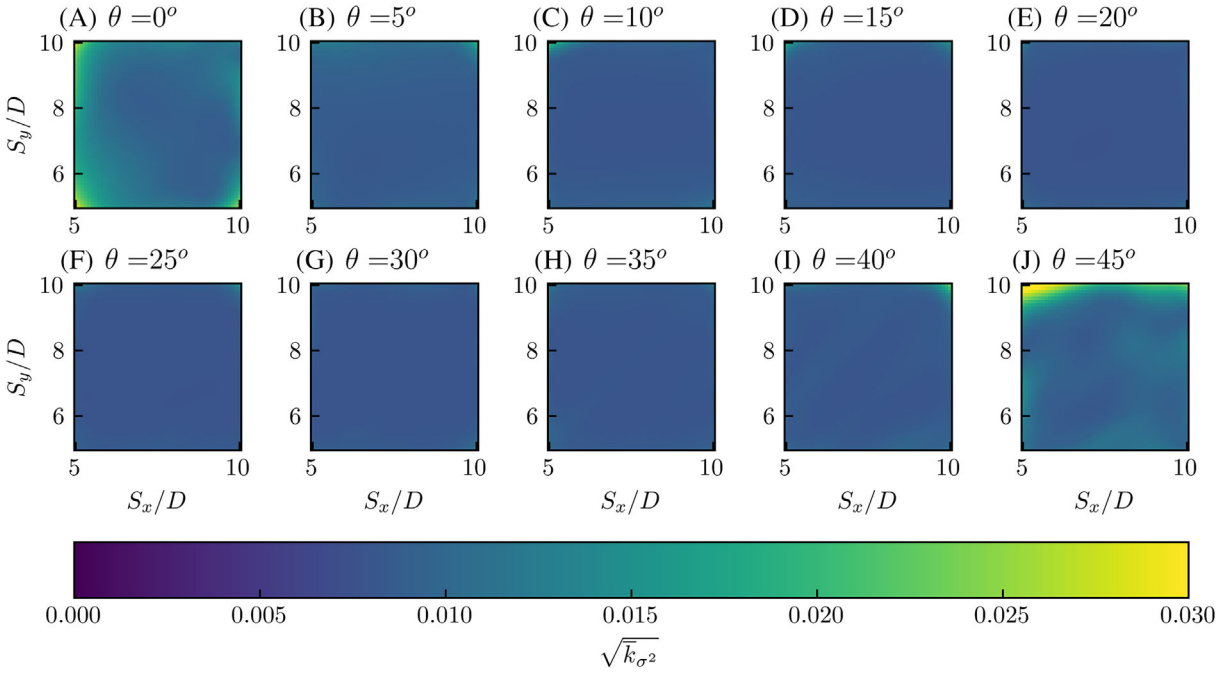

FIGURE 9 Posterior variance function for $g_{high}(v)$ of MF-GP-nlow500.

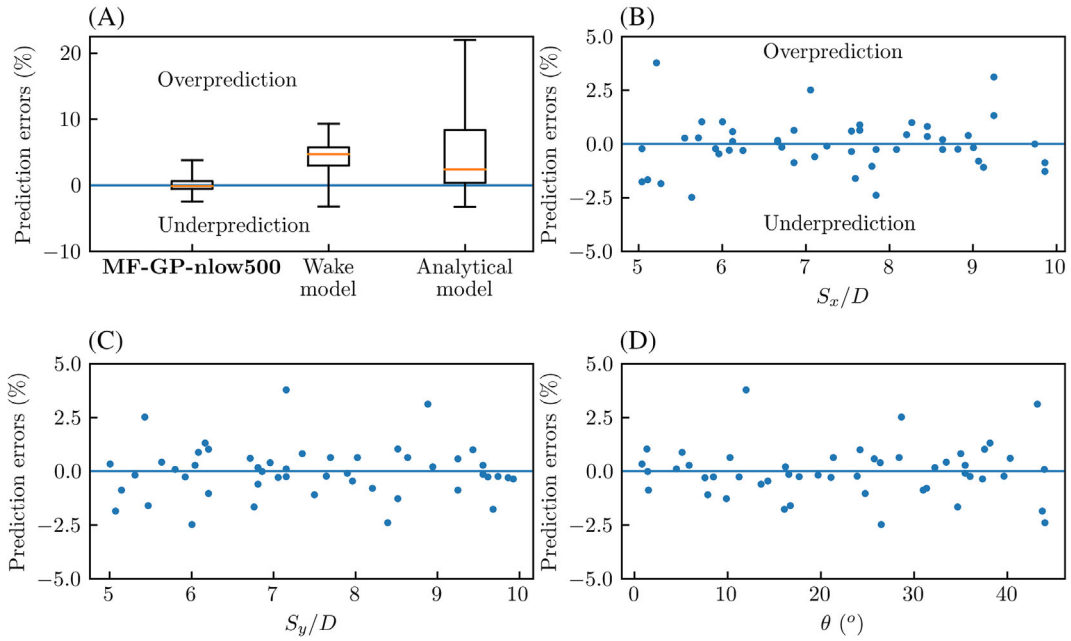

FIGURE 10 Comparison of LOOCV prediction errors (%) for different models (A) and LOOCV prediction error (%) of MF-GP-nlow500 against input parameters (B) S_x/D, (C) S_y/D, and (D) $\theta(^o)$. Note that for the box plot in (A) the orange line is the median LOOCV error and the box is the interquartile range of LOOCV error.

5.3 | Prediction of wind farm performance

We use the predicted values of $C^*_{T,LES}$ from the emulators to predict the power output of wind farms under various mesoscale atmospheric conditions, following the concept of the two-scale momentum theory. We predict the (farm-averaged) turbine power coefficient C_p using $C^*_{T,LES}$ predictions from MF-GP-nlow500. We call this prediction of farm performance $C_{p,model}$. Firstly, we use the $C^*_{T,LES}$ prediction from the LOOCV procedure as C^*_T in Equation (1) to calculate β for a given value of wind extractability ζ. We substitute this value of β into the expression $C_p = \beta^3 C_T^{*\frac{3}{2}} C_{T'}^{-\frac{1}{2}}$

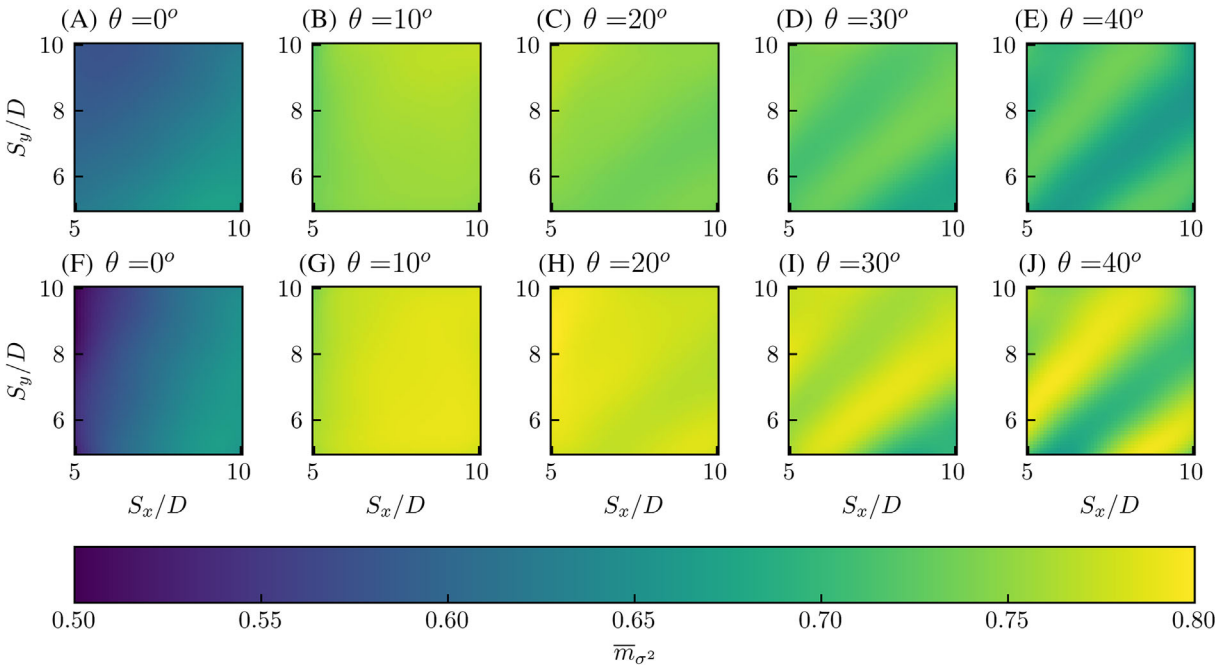

FIGURE 11 Posterior mean function of MF-GP-nlow500 for different values of θ for (A) to (E) $g_{high}(v)$ and (F) to (J) $g_{low}(v)$.

(which is only valid for actuator discs) to calculate $C_{p,model}$. We compare the value of $C_{p,model}$ with the turbine power coefficient recorded in the LES, $C_{p,LES}$. The effect of the coarse LES resolution on turbine thrust (and hence also ABL response and C_p) has already been corrected.[8] The LES was performed with periodic horizontal boundary conditions and a fixed momentum supply, that is, $\zeta = 0$. However, the $C_{p,LES}$ has also been adjusted for a given ζ by scaling the velocity fields assuming Reynolds number independence.[8]

Similarly, the analytical model of C_T^* can be used to give a theoretical prediction of wind farm performance called $C_{p,Nishino}$,[8] which is given by

$$C_{p,Nishino} = \frac{64C_T'}{(4+C_T')^3}\left[\frac{-\zeta+\sqrt{\zeta^2+4\left(\frac{16C_T'}{(4+C_T')^2}\frac{\lambda}{C_{f0}}+1\right)(1+\zeta)}}{2\left(\frac{16C_T'}{(4+C_T')^2}\frac{\lambda}{C_{f0}}+1\right)}\right]^3. \tag{12}$$

We will compare the accuracy of both $C_{p,model}$ and $C_{p,Nishino}$ in predicting $C_{p,LES}$.

Both $C_{p,model}$ and $C_{p,LES}$ are shown in Figure 12 for a realistic range of wind extractability factors, along with the results from $C_{p,Nishino}$ (Equation 12). $C_{p,Nishino}$ provides an approximate upper limit of farm-averaged C_p as it predicts very well the effects of array density and large-scale atmospheric response. The statistical model accurately predicts the effect of turbine layout on farm performance, which becomes more important with larger ζ values. As ζ increases, there is a larger difference between $C_{p,LES}$ and $C_{p,Nishino}$. Also, $C_{p,model}$ becomes slightly less accurate when ζ increases.

Table 3 shows the average prediction errors of $C_{p,model}$ and $C_{p,Nishino}$. We quantified the mean absolute error using two different reference powers. Using $C_{p,LES}$ as the reference power, $C_{p,Nishino}$ had an error of around 5% and the error increases with ζ. The mean absolute error of $C_{p,model}$ was typically less than 1.5% and this decreased slightly as ζ increases (due to the reference power $C_{p,LES}$ increasing). We also use the power of an isolated ideal turbine, $C_{p,Betz}$, as a reference power. $C_{p,Betz}$ is calculated using the actuator disc theory with the expression $C_{p,Betz} = 64C_T'/(4+C_T')^3$ (note that in this study $C_T' = 1.33$, and hence, $C_{p,Betz} = 0.563$). In this case, the mean absolute error increased with ζ for both $C_{p,model}$ and $C_{p,Nishino}$. However, the average prediction error of $C_{p,model}$ remained below 0.65%.

6 | DISCUSSION

Data-driven modelling of the internal turbine thrust coefficient C_T^* is a novel approach to modelling turbine wake interactions. Data-driven models of wind farm performance typically focus on predicting the power output, which, however, depends on flow physics across a wide range of scales.

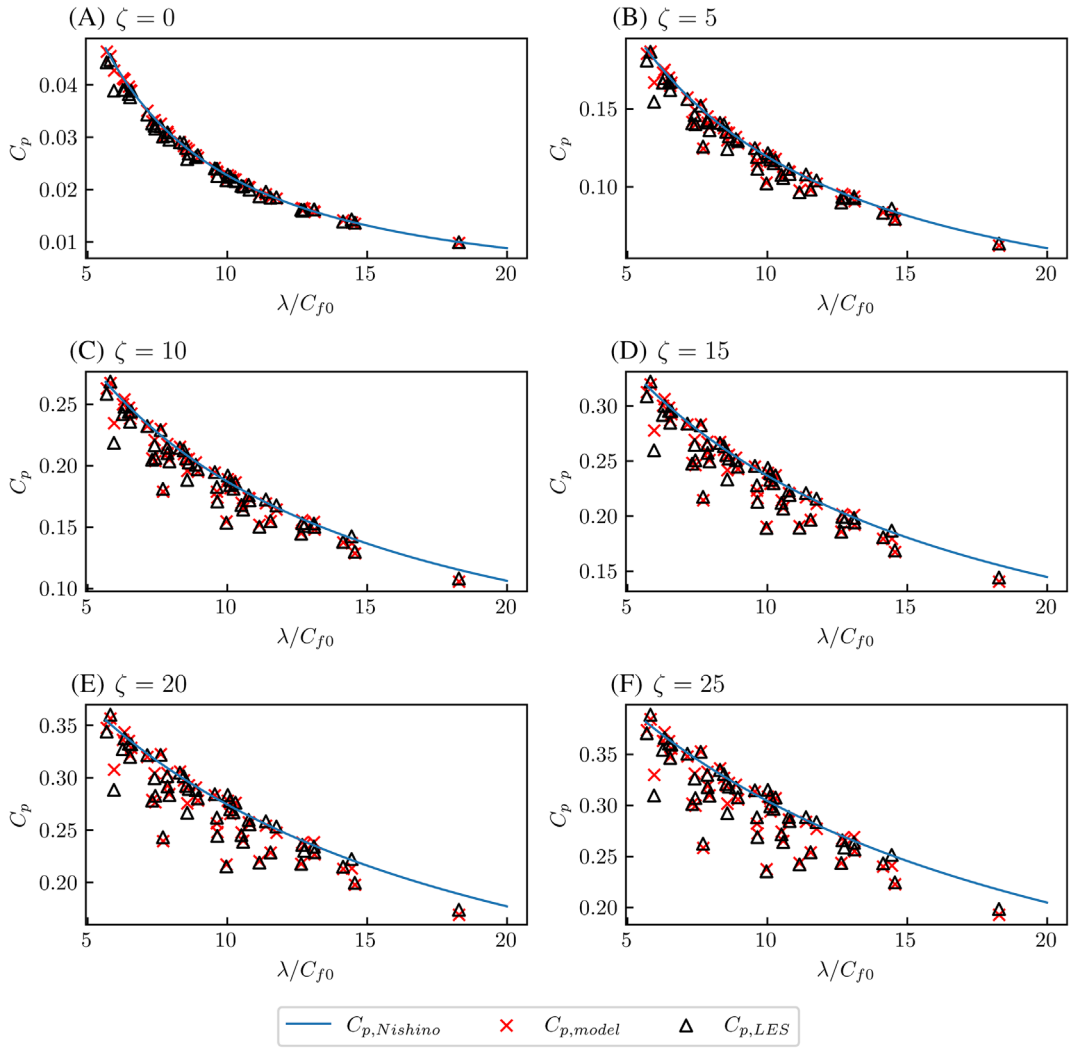

FIGURE 12 Comparison of C_p predictions with LES results for a realistic range of ζ values.

TABLE 3 Comparison of models for C_p prediction.

$\frac{1}{50}\sum_{i=1}^{50}\lvert C_{p,i} - C_{p,LES}\rvert / C_{p,LES}$			$\frac{1}{50}\sum_{i=1}^{50}\lvert C_{p,i} - C_{p,LES}\rvert / C_{p,Betz}$		
ζ	$C_{p,Nishino}$	$C_{p,model}$	ζ	$C_{p,Nishino}$	$C_{p,model}$
0	2.82%	2.15%	0	0.142%	0.108%
5	4.38%	1.48%	5	0.954%	0.338%
10	5.16%	1.35%	10	1.67%	0.459%
15	5.66%	1.30%	15	2.24%	0.542%
20	6.02%	1.26%	20	2.72%	0.601%
25	6.30%	1.24%	25	3.11%	0.648%

Current data-driven approaches are either not generalizable to different atmospheric responses, or would require a very large set of expensive training data, such as finite-size wind farm LES data. Data-driven models of C_T^* captures the effects of turbine wake interactions, whilst also being applicable to different atmospheric responses (following the concept of the two-scale momentum theory).

The statistical emulator of C_T^* developed in this study was able to predict the farm power C_p of Kirby et. al.[8] with an average error of less than 0.65%. The high accuracy and very low computational cost of this approach shows the potential of this approach for modelling turbine wake interactions. It has several advantages over traditional approaches using the superposition of wake models. Information from turbulence-resolving LES is included, which ensures a high accuracy. It will also be more advantageous as wind farms become larger because wake models struggle to

capture the complex multi-scale flows physics, which are important for large farms. The statistical model of C_T^* may therefore allow fast and accurate predictions of wind farm performance.

All emulators developed in this study gave substantially better predictions of $C_{T,LES}^*$ compared to the analytical and wake models. Both the mean and maximum prediction errors were reduced by the emulators. The standard GP regression approach had a mean prediction error of 1.26% and maximum error of approximately 6%. The accuracy depends on the size of the LES data set and could be further decreased with a larger training set. The multi-fidelity GP approach gave more accurate predictions of $C_{T,LES}^*$ compared to the standard GP regression. This is because non-linear information fusion algorithm has incorporated information from many low-fidelity data points to improve the emulator of the high-fidelity (LES) model. This approach has the advantage that, unlike the standard GP regression approach, it is not necessary to evaluate the prior mean before making a prediction. Therefore, to predict C_T^* it is only necessary to evaluate the posterior mean of the high-fidelity emulator for a specific turbine layout.

The shape of the posterior mean in Figure 8 gives insights into the physics of turbine wake interactions. This is because $C_{T,LES}^*$ is low when a layout has a high degree of turbine wake interactions. For the turbine operating conditions used, $C_{T,LES}^*$ is close to 0.75 when a layout has a small degree of wake interactions. Figure 8A shows $C_{T,LES}^*$ when the wind direction is perfectly aligned with the rows of turbines ($\theta = 0$). This gives wind farms with a high degree of wake interactions that results in low $C_{T,LES}^*$ values. For $\theta = 0^\circ$, increasing S_x/D increases C_T^* because there is a larger streamwise distance between turbines for the wakes to recover. When the cross-streamwise spacing (S_y/D) is increased the degree of wake interactions increases, that is, $C_{T,LES}^*$ decreases. This is because there is a lower array density, which results in a lower turbulence intensity within the farm and hence slower wake recovery. Yang[41] found that increasing the cross-streamwise spacing in infinitely-large wind farms increased the power of individual turbines and concluded that this was due to reduced wake interactions. However, the increase in turbine power found by Yang[41] may be also explained by a faster farm-averaged wind speed caused by a reduced array density rather than reduced wake interactions.

When the wind direction θ increases, $C_{T,LES}^*$ increases to a maximum of just over 0.75 at $\theta = 10^\circ$ (Figure 8C). This result agrees qualitatively with another study[42] in which it was found that the maximum farm power was produced by an intermediate wind direction. When θ increases above 20° regions of low $C_{T,LES}^*$ appear diagonally (see Figure 8F–J). The regions of low $C_{T,LES}^*$ are centered on the surfaces given by $S_y = 2S_x\tan(\theta)$, $S_y = S_x\tan(\theta)$ and $S_y = 0.5S_x\tan(\theta)$. These regions correspond to turbines being aligned along different axes throughout the farm (see Figure 13). There are longer streamwise distance between turbines for these arrangements (compared to $\theta = 0^\circ$) and so the $C_{T,LES}^*$ values are higher than for $\theta = 0^\circ$.

The accuracy of the statistical emulators could be further improved in future studies. Both the standard and multi-fidelity GP models can be improved by adding more evaluations of $C_{T,LES}^*$. From Table 2, the accuracy of the multi-fidelity GP models did not improve once we used more than 500 $C_{T,wake}^*$ evaluations. This shows that the error in predicting $C_{T,LES}^*$ for MF-GP-nlow500 is not due to the model of f_{wake}. Instead, the error arises from the learnt relationship between f_{wake} and f_{LES}.

The statistical emulators developed are not applicable to all wind farms because of the limited nature of our data set. A limitation of the developed model is that it is only applicable to farms with perfectly aligned layouts. It should also be noted that our model was trained on data from simulations of a neutrally stratified boundary layer. Therefore, a larger LES data set with an extended parameter space would be required to account for the effect of atmospheric stability on wake interactions and the resulting C_T^*. Another limitation of our model is that it assumes all turbines have the same resistance coefficient C_T'. It is likely that this condition can be strictly satisfied only in the fully developed region of a large farm where the wind speed does not change in the streamwise or cross-streamwise directions.

Although we considered only actuator discs in this study for demonstration, the proposed approach using a data-driven model of C_T* can be applied to power prediction of real turbines as well in future studies. In this study, we calculate $C_{p,model}$ using the expression $C_{p,model} = \beta^3 C_T^{*\frac{3}{2}} C_T'^{-\frac{1}{2}}$. This assumes that the relationship between C_p^* and C_T' is given by $C_p^* = C_T^{*\frac{3}{2}} C_T'^{-\frac{1}{2}}$, which is only valid for actuator discs. For real turbines, the relationship between C_p^* and C_T' can be calculated using BEM theory[43] according to the turbine design and operating conditions (noting that the turbine induction factor can still be estimated as $a = C_T'/(4 + C_T')$). $C_{p,model}$ can then be calculated using Equation (5) with β found using Equation (1).

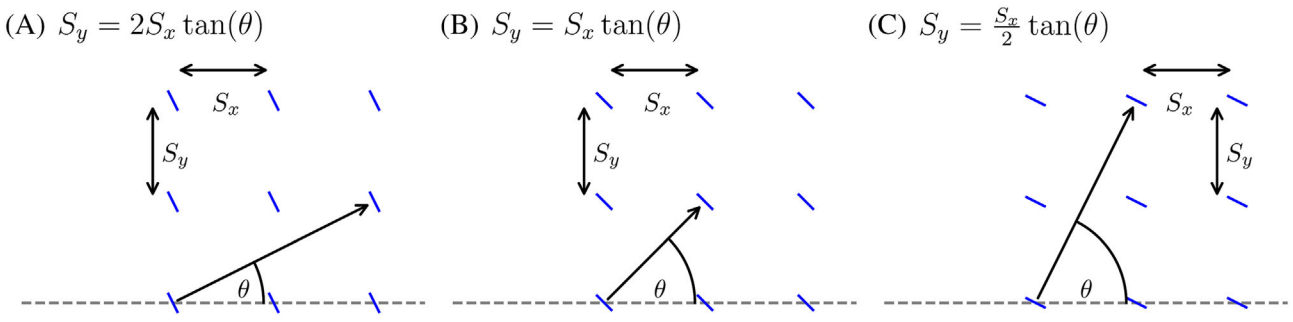

FIGURE 13 Alignment of turbines for different combinations of S_x, S_y and θ.

However, for a data-driven model of C_T^* to be applicable to real turbines, it will be necessary to model the impact of a variable C_T' rather than assuming a fixed C_T' value as in this study.

7 | CONCLUSIONS

In this study, we proposed a new data-driven approach to modelling turbine wake interactions and resulting flow resistance in large wind farms. We developed statistical emulators of the farm-internal turbine thrust coefficient $C_{T,LES}^*$ as a function of turbine layout and wind direction. C_T^* represents the flow resistance within a wind farm and reflects the characteristics of the turbine-scale flows including wake and turbine blockage effects. We developed several emulators using both standard GP regression and multi-fidelity GP regression. The standard GP was trained using data from 50 infinitely-large wind farm LES (and using a low-fidelity wake model as a prior mean). The multi-fidelity GP was trained using data from both LES and wake model simulations. We estimated the test accuracy of the model by performing leave-one-out cross-validation and assessed the error in predicting $C_{T,LES}^*$. All emulators had a mean test error of less than 2% for predicting $C_{T,LES}^*$. The multi-fidelity GP gave the best performance with a mean prediction error of 0.849% and maximum prediction error of 3.78% with no bias for under or over-prediction. This is low compared to the mean error of the wake model (4.60%) and analytical C_T^* model (5.26%), which both had a bias for overpredicting $C_{T,LES}^*$.

We used an emulator of $C_{T,LES}^*$ to make predictions of wind farm performance under various mesoscale atmospheric conditions (characterized by the wind extractability factor ζ) using the two-scale momentum theory.[24] Our predictions of farm power production had an average error of less than 1.5% under realistic wind extractability scenarios compared to the LES. When the error in power prediction is expressed relative to the power of an isolated ideal turbine the average prediction error is less than 0.7%. We also used a previously proposed analytical model of C_T^* [25] to predict farm power output with an average error of less than 3.5% (with the power of an isolated turbine as the reference power). The analytical model correctly predicts the trends in farm performance with array density under different scenarios of large-scale atmospheric response, although it tends to overpredict the power where turbine wake interactions are important. Using statistical emulators of C_T^* is a new approach to modelling turbine wake interactions and flow resistance within large wind farms. The approach can be extended in future studies by increasing the size of the training data set, for example, to account for the effects of C_T' and atmospheric stability conditions on C_T^*. The very low computational cost and high accuracy of the model could be beneficial for future wind farm optimization.

AUTHOR CONTRIBUTIONS

Takafumi Nishino derived the theory. Andrew Kirby and Thomas D. Dunstan performed the simulations. François-Xavier Briol provided assistance and guidance for the machine learning methodology. Andrew Kirby wrote the paper with corrections from Takafumi Nishino, François-Xavier Briol and Thomas D. Dunstan.

ACKNOWLEDGEMENTS

The first author (AK) acknowledges the NERC-Oxford Doctoral Training Partnership in Environmental Research (NE/S007474/1) for funding and training.

CONFLICT OF INTEREST STATEMENT

The authors report no conflict of interest.

PEER REVIEW

The peer review history for this article is available at https://www.webofscience.com/api/gateway/wos/peer-review/10.1002/we.2851.

DATA AVAILABILITY STATEMENT

The data and code that support the findings of this study are openly available at https://github.com/AndrewKirby2/ctstar_statistical_model. This includes the results from the wind farm LES and wake model simulations. The repository also includes the code for the results presented in Sections 5.1,5.2 and 5.3.

ORCID

Andrew Kirby https://orcid.org/0000-0001-8389-1619
François-Xavier Briol https://orcid.org/0000-0002-0181-2559
Takafumi Nishino https://orcid.org/0000-0001-6306-7702

REFERENCES

1. Porté-Agel F, Bastankhah M, Shamsoddin S. Wind-turbine and wind-farm flows: a review. *Bound-Layer Meteorol*. 2020;174:1-59. doi:10.1007/s10546-019-00473-0
2. Bleeg J, Purcell M, Ruisi R, Traiger E. Wind farm blockage and the consequences of neglecting its impact on energy production. *Energies*. 2018;11:1609.
3. Carbon Trust. Global blockage effect in offshore wind (globe). 2022. Accessed July 11, 2022. https://www.carbontrust.com/our-projects/large-scale-rd-projects-offshore-wind/global-blockage-effect-in-offshore-wind-globe
4. Jensen NO. A note on wind generator interaction. Risø-M-2411 RisøNational Laboratory Roskilde; 1983.
5. Bastankhah M, Porté-Agel F. A new analytical model for wind-turbine wakes. *Renew Energy*. 2014;70:116-123.
6. Katic I, Hojstrup J, Jensen NO. A simple model for cluster efficiency. In: Proceedings of the European wind energy association conference and exhibition, Rome, Italy; 1986:407-409.
7. Zong H, Porté-Agel F. A momentum-conserving wake superposition method for wind farm power prediction. *J Fluid Mech*. 2020;889:A8.
8. Kirby A, Nishino T, Dunstan TD. Two-scale interaction of wake and blockage effects in large wind farms. *J Fluid Mech*. 2022;953:A39.
9. Stevens RJAM, Gayme DF, Meneveau C. Effects of turbine spacing on the power output of extended wind-farms. *Wind Energy*. 2016;19:359-370.
10. Fitch AC, Olson JB, Lundquist JK, Dudhia J, Gupta AK, Michalakes J, Barstad I. Local and mesoscale impacts of wind farms as parameterized in a mesoscale NWP model. *Month Weather Rev*. 2012;140:3017-3038.
11. Abkar M, Porté-Agel F. A new wind-farm parameterization for large-scale atmospheric models. *J Renew Sustain Energy*. 2015;7:013121.
12. Pan Y, Archer CL. A hybrid wind-farm parametrization for mesoscale and climate models. *Bound-Layer Meteorol*. 2018;168:469-495. doi:10.1007/s10546-018-0351-9
13. Zehtabiyan-Rezaie N, Iosifidis A, Abkar M. Data-driven fluid mechanics of wind farms: a review. *J Renew Sustain Energy*. 2022;14:32703. doi:10.1063/5.0091980
14. Renganathan SA, Maulik R, Letizia S, Iungo GV. Data-driven wind turbine wake modeling via probabilistic machine learning. *Neural Comput Appl*. 2022;34:6171-6186. doi:10.1007/s00521-021-06799-6
15. Optis M, Perr-Sauer J. The importance of atmospheric turbulence and stability in machine-learning models of wind farm power production. *Renew Sustain Energy Rev*. 2019;112:27-41.
16. Japar F, Mathew S, Narayanaswamy B, Lim CM, Hazra J. Estimating the wake losses in large wind farms: a machine learning approach. In: ISGT 2014 IEEE; 2014:1-5.
17. Yan C, Pan Y, Archer CL. A general method to estimate wind farm power using artificial neural networks. *Wind Energy*. 2019;22:1421-1432.
18. Zhang J, Zhao X. Wind farm wake modeling based on deep convolutional conditional generative adversarial network. *Energy*. 2022;238:121747.
19. Wilson B, Wakes S, Mayo M. Surrogate modeling a computational fluid dynamics-based wind turbine wake simulation using machine learning. In: 2017 IEEE Symposium Series on Computational Intelligence (SSCI); 2017:1-8.
20. Ti Z, Deng XW, Yang H. Wake modeling of wind turbines using machine learning. *Appl Energy*. 2020;257:114025.
21. Ti Z, Deng XW, Zhang M. Artificial neural networks based wake model for power prediction of wind farm. *Renew Energy*. 2021;172:618-631.
22. Park J, Park J. Physics-induced graph neural network: an application to wind-farm power estimation. *Energy*. 2019;187:115883.
23. Bleeg J. A graph neural network surrogate model for the prediction of turbine interaction loss. *J Phys: Confer Ser*. 2020;1618:062054.
24. Nishino T, Dunstan TD. Two-scale momentum theory for time-dependent modelling of large wind farms. *J Fluid Mech*. 2020;894:A2.
25. Nishino T. Two-scale momentum theory for very large wind farms. *J Phys: Confer Ser*. 2016;753:032054.
26. Patel K, Dunstan TD, Nishino T. Time-dependent upper limits to the performance of large wind farms due to mesoscale atmospheric response. *Energies*. 2021;14:6437.
27. Sacks J, Welch WJ, Mitchell TJ, Wynn HP. Design and analysis of computer experiments. *Stat Sci*. 1989;4:409-423.
28. Currin C, Mitchell T, Morris M, Ylvisaker D. Bayesian prediction of deterministic functions, with applications to the design and analysis of computer experiments. *J Am Stat Ass*. 1991;86:953-963.
29. Johnson ME, Moore LM, Ylvisaker D. Minimax and maximin distance designs. *J Stat Plan Inference*. 1990;26:131-148.
30. Santner TJ, Williams BJ, Notz W. *The design and analysis of computer experiments*. Second; 2018.
31. Wynne G, Briol FX, Girolami M. Convergence guarantees for Gaussian process means with misspecified likelihoods and smoothness. *J Mach Learn Res*. 2021;22:5468-5507.
32. Shapiro CR, Gayme DF, Meneveau C. Filtered actuator disks: theory and application to wind turbine models in large eddy simulation. *Wind Energy*. 2019;22:1414-1420.
33. Niayifar A, Porté-Agel F. Analytical modeling of wind farms: a new approach for power prediction. *Energies*. 2016;9:741.
34. Pedersen MM, van der Laan P, Friis-Mller M, Rinker J, Rthor P-E. Dtuwindenergy/pywake: Pywake; 2021.
35. Crespo A, Hernndez J. Turbulence characteristics in wind-turbine wakes. *J Wind Eng Ind Aerodyn*. 1996;61:71-85.
36. Rasmussen CE, Williams CKI. *Gaussian processes for machine learning*. the MIT Press; 2018.
37. Peherstorfer B, Willcox K, Gunzburger M. Survey of multifidelity methods in uncertainty propagation, inference, and optimization. *SIAM Rev*. 2018;60:550-591.
38. Perdikaris P, Raissi M, Damianou A, Lawrence ND, Karniadakis GE. Nonlinear information fusion algorithms for data-efficient multi-fidelity modelling. *Proc Royal Soc A: Math, Phys Eng Sci*. 2017;473:20160751.
39. Paleyes A, Pullin M, Mahsereci M, Lawrence N, Gonzlez J. Emulation of physical processes with emukit; 2019.
40. GPy. GPy: a Gaussian process framework in python. 2012. http://github.com/SheffieldML/GPy
41. Yang X, Kang S, Sotiropoulos F. Computational study and modeling of turbine spacing effects infinite aligned wind farms. *Phys Fluids*. 2012;24:11510.
42. Stevens RJAM, Gayme DF, Meneveau C. Large eddy simulation studies of the effects of alignment and wind farm length. *J Renew Sustain Energy*. 2014;6:023105.
43. Nishino T, Hunter W. Tuning turbine rotor design for very large wind farms. *Proc R Soc A: Math, Phys Eng Sci*. 2018;474(2220):1-20.

How to cite this article: Kirby A, Briol F-X, Dunstan TD, Nishino T. Data-driven modelling of turbine wake interactions and flow resistance in large wind farms. *Wind Energy*. 2023;1-17. doi:10.1002/we.2851

APPENDIX A: ESTIMATED HYPERPARAMETERS

In this section, we present the estimated hyperparameters for each model in Section 5. The hyperparameters for the standard GP models are shown in Table A1. Unlike the LOOCV procedure in Section 5, the models are fitted to all 50 LES data points $C^*_{T,LES}$. The hyperparameter values in Table A1 will therefore be slightly different but very similar to the models in Section 5. Equation (9) shows the meaning of each of the hyperparameters. For GP-analytical-prior, l_3 is the smallest lengthscale. This suggests that $C^*_{T,LES}$ is more sensitive to wind direction than turbine spacing.

The kernel for the multi-fidelity GP models is shown in Equation (A1). The kernel is the product and sum of different anisotropic squared-exponential covariance functions. Table A2 gives the estimated hyperparameters for the multi-fidelity GP models. For Table A2, all 50 LES data points have been used for the fitting process.

$$k_{high}(v,v';\theta) = k_{high,\rho}(v,v';\theta_{high,\rho}) \cdot k_{high,f}(g_{low,*}(v), g_{low,*}(v');\theta_{high,f}) + k_{high,\delta}(v,v';\theta_{high,\delta}) \tag{A1}$$

TABLE A1 Estimated hyperparameters for standard GP regression models.

Hyperparameter	GP-analytical-prior	GP-wake-TI1-prior	GP-wake-TI5-prior	GP-wake-TI10-prior	GP-wake-TI15-prior
σ_f^2	5.25×10^{-3}	1.22×10^{-3}	1.27×10^{-3}	9.09×10^{-4}	1.19×10^{-3}
l_1	5.27	2.51	1.74×10^{3}	1.03×10^{4}	5.06
l_2	4.90	1.26×10^{3}	1.94×10^{3}	3.57	5.91
l_3	6.16×10^{-1}	2.02×10^{-1}	4.55×10^{-1}	2.03	6.25×10^{-1}
σ^2	1.47×10^{-4}	4.06×10^{-4}	3.83×10^{-4}	1.28×10^{-4}	4.10×10^{-5}

TABLE A2 Estimated hyperparameters for multi-fidelity GP regression models.

Hyperparameter		MF-GP-nlow250	MF-GP-nlow500	MF-GP-nlow100
k_{low}	σ_f^2	2.09×10^{-1}	9.51×10^{-2}	9.70×10^{-2}
	l_1	3.31	1.42	1.40
	l_2	2.51	1.18	1.10
	l_3	7.31×10^{-1}	5.12×10^{-1}	5.07×10^{-1}
$k_{high,\rho}$	σ_f^2	2.22×10^{-5}	7.91×10^{-6}	2.34×10^{-5}
	l_1	3.77	3.92×10^{4}	1.23×10^{1}
	l_2	6.29	6.94	9.52
	l_3	9.25	10.0	6.66
$k_{high,f}$	σ_f^2	2.82×10^{2}	2.00×10^{3}	4.89×10^{2}
	l	1.15×10^{-1}	1.96×10^{-1}	1.89×10^{-1}
$k_{high,\delta}$	σ_f^2	5.16×10^{-1}	5.42×10^{-1}	5.00×10^{-1}
	l_1	1.18×10^{4}	8.76×10^{4}	2.33×10^{4}
	l_2	1.45×10^{4}	1.17×10^{5}	2.58×10^{4}
	l_3	1.01×10^{4}	5.11×10^{5}	4.93×10^{3}
σ_{low}^2		1.09×10^{-4}	2.75×10^{-6}	1.84×10^{-6}
σ_{high}^2		5.59×10^{-4}	5.46×10^{-5}	4.42×10^{-5}