Wind Energ. Sci., 11, 2845–2868, 2026
https://doi.org/10.5194/wes-11-2845-2026

eawe european academy of wind energy
Open Access WIND ENERGY SCIENCE

Validation of RANS-calibrated engineering models and ANN-based surrogate for wind farm flow simulation and layout optimization

Jens Peter Schøler, Ernestas Simutis, M. Paul van der Laan, Julian Quick, and Pierre-Elouan Réthoré

DTU Wind and Energy Systems, Frederiksborgvej 399, 4000 Roskilde, Denmark

Correspondence: Jens Peter Schøler (jpsch@dtu.dk) and Pierre-Elouan Réthoré (pire@dtu.dk)

Received: 21 January 2026 – Discussion started: 13 February 2026
Revised: 18 June 2026 – Accepted: 13 July 2026 – Published: 7 August 2026

Abstract. Accurate yet efficient wake modeling is essential for wind farm layout optimization (WFLO). Wind turbine wakes are disturbed regions of flow behind a wind turbine, characterized by lower mean wind speeds and higher turbulence, which reduce downstream power production and increase structural loading. This study compares an artificial neural network (ANN)-based surrogate trained on Reynolds-averaged Navier–Stokes (RANS) data with two representative engineering wake models based on the TurbOPark and super-Gaussian formulations. The work includes recalibration of the engineering models, a systematic flow simulation study across varying turbine counts and spacings, and WFLO benchmarks validated against RANS-based annual energy production (AEP). Results show that the ANN surrogate achieves the lowest RMSE and MAPE across all scenarios in flow estimation, albeit at a higher computational cost. In WFLO, the TurbOPark-based model produced the highest RANS-validated AEP layouts, despite having lower predictive accuracy, suggesting that optimization complexity influences outcomes. Blockage modeling increased computational cost without improving accuracy.

1 Introduction

A key challenge when designing a wind farm is determining the optimal placement of wind turbines. This decision must consider various factors, including nearby communities, local wind conditions, electrical infrastructure, seabed conditions, and other relevant considerations. The interaction between turbines through wake effects is particularly important, as it can significantly reduce energy output and, in turn, impact project economic viability. While wake effects are the dominant turbine-to-turbine interaction, the induction of the turbines also produces a wind-farm-scale blockage effect that slows the inflow upstream and can result in power losses at the leading turbines (Bleeg et al., 2018). This upstream slowdown has been observed in offshore lidar measurements (Schneemann et al., 2021) and grows in relevance for the large turbine counts of modern offshore wind farms, where the induction of the individual turbines aggregates into a combined wind-farm-scale blockage effect (Nygaard et al., 2020).

To obtain the best achievable placement of the turbines, wind farm layout optimization (WFLO) can be applied. WFLO utilizes numerical optimization to minimize or maximize a given cost function, typically a variation of energy production or energy cost, while adhering to predefined constraints (see, e.g., Herbert-Acero et al., 2014; Baker et al., 2019; Bortolotti et al., 2022). WFLO algorithms can be broadly grouped into gradient-based and gradient-free methods. The present work focuses on gradient-based optimization. Gradient-free approaches nonetheless remain an area of interest and continue to advance, ranging from classical evolutionary genetic algorithms (Mosetti et al., 1994) and particle swarm optimization (Pookpunt and Ongsakul, 2013) to recent hybrids such as the reinforcement learning-enhanced genetic algorithm of Dong et al. (2026). Regardless of the optimization algorithm applied in WFLO, it is essential for flow models to account for wake effects. Existing approaches can be broadly classified into two categories: engineering wake models, which construct the wind farm flow field by superposing individual wakes, and computational fluid dynamics

(CFD) models, which resolve the underlying physics through numerical simulation.

The first engineering model, proposed by Jensen (1983) and often referred to as the Park model, was primarily concerned with representing the wake-center velocity and employed a top-hat shape to do so. This approximation initially worked well but became increasingly problematic as the turbine and wind farm sizes increased. The breakdown of turbulent structures leads to the wake becoming self-similar and converging toward a Gaussian-like shape. For this reason, later variations typically rely on a Gaussian profile to model the wake deficits. The area where the wake is self-similar is referred to as the far wake. The location downstream of the turbine where this occurs depends on both atmospheric and terrain conditions. Dar et al. (2019) investigated this with LESs and showed that self-similarity occurs between 1–3 rotor diameters (D) in complex terrain and between 3–10 rotor diameters for flat terrain. The different areas of the flow around a wind turbine are labeled and illustrated in Fig. 1.

In the literature, engineering models and CFD methods are often referred to as low- and high-fidelity models, respectively. Engineering models are much faster: $\sim 10^4\times$ faster than Reynolds-averaged Navier–Stokes (RANS) (van der Laan et al., 2015a) and $\sim 10^7\times$ faster than large-eddy simulation (LES) (Porté-Agel et al., 2020), making CFD too slow for WFLO, where the flow must be evaluated many times during the optimization process. In general, high-fidelity models are considered more accurate. However, under certain circumstances, engineering models can outperform high-fidelity models, e.g., within limited domains, such as the far wake, when operating near a calibration point, or when considering integrated quantities, such as power. However, design decisions and the calibration domain constrain their upper limits of accuracy.

Wake models can be split into two categories. The first comprises analytically solved models, which typically assume a self-similar far-wake profile. Alternatively, there is a group of low-fidelity models that require solving but do not rely on self-similarity profiles (e.g., Ainslie, 1988; Ott et al., 2011; Klemmer and Howland, 2025). The class of self-similar, analytically solved wake models dominates WFLO practice and is the focus of the present study. In WFLO, a minimum inter-turbine separation is typically imposed to limit fatigue loads; because this constraint keeps downstream turbines out of the near wake, engineering wake models applied in WFLO have typically been developed and calibrated with the far wake as the design target.

Several analytically solved engineering models exist and can loosely be categorized into families: Gaussian formulations (Bastankhah and Porté-Agel, 2014; Niayifar and Porté-Agel, 2016), a super-Gaussian variation that bridges top-hat near-wake and Gaussian far-wake behavior (Blondel and Cathelain, 2020), and the turbulence-optimized park (TurbOPark) models (Nygaard et al., 2020, 2022). Special formulations have also been developed to account for the wake of turbines operating with a yaw offset, which is particularly useful for wake-steering scenarios (Bastankhah and Porté-Agel, 2016; Martínez-Tossas et al., 2019). The breadth of existing models is wide, and the models mentioned here are an important subset, but many more exist for different applications. At the farm scale, overlapping wakes have further motivated momentum-conserving superposition schemes (Zong and Porté-Agel, 2020; Lanzilao and Meyers, 2022) beyond the classical linear and quadratic sums (Porté-Agel et al., 2020). In the present study, recalibration has been performed on two engineering model configurations using the `PyWake` (Pedersen et al., 2023) framework.

An alternative to classical models is the use of data-driven surrogate models. Surrogates are models that seek to approximate a higher-fidelity model at a fraction of the cost. In wake modeling, surrogates are typically trained with RANS or LES data. Zehtabiyan-Rezaie et al. (2022) have conducted a review of the field, encompassing various types of artificial neural network (ANN)-based wake surrogates, as well as classic machine learning (ML) techniques such as proper orthogonal decomposition (POD) and dynamic mode decomposition (DMD). Some notable works include Ti et al. (2021) and Yang and Deng (2023), who trained an ANN-based RANS surrogate using multiple parallel neural networks, each corresponding to a specific grid location in the turbine flow. A limitation of this approach is that the surrogate cannot be used with automatic differentiation in gradient-based WFLO or any optimization with continuous turbine coordinates, as these models are undefined outside the considered grid points. To address this, we have previously developed ANN-based RANS surrogates with continuous definitions that support automatic differentiation (Schøler et al., 2023; Pish et al., 2024). One limitation of training ANN-based surrogates is the need for extensive data to cover cases across various inflow types and operating conditions. Schøler et al. (2024) investigated physics-informed neural networks (PINNs) as a means to reduce data requirements, ultimately finding that a hybrid data-and-physics loss was too costly given the gains. Gafoor et al. (2025) demonstrated the possibility of an RANS PINN surrogate trained solely with physics information. Together, current literature indicates a significant potential in ANN-based RANS surrogates. From the related field of dynamic surrogate wake models, a couple of interesting works are listed to situate the current work better in a wider context. Dynamic models are typically motivated by a desire to improve wind farm control and mitigate loads. The work of Zhang and Zhao (2021) used a physics-informed reconstruction of unsteady wake fields from lidar measurements, the predictive and stochastic reduced-order model (PS-ROM) of wake dynamics by Andersen and Leon (2022), the PS-ROM derivative work on global POD modes by Moreno et al. (2025), and real-time physics-guided frameworks (Li et al., 2024a). Most recently, Liu et al. (2026) proposed PhyWakeNet, a hybrid physics and data-driven model that captures the time-averaged deficit, wake meandering,

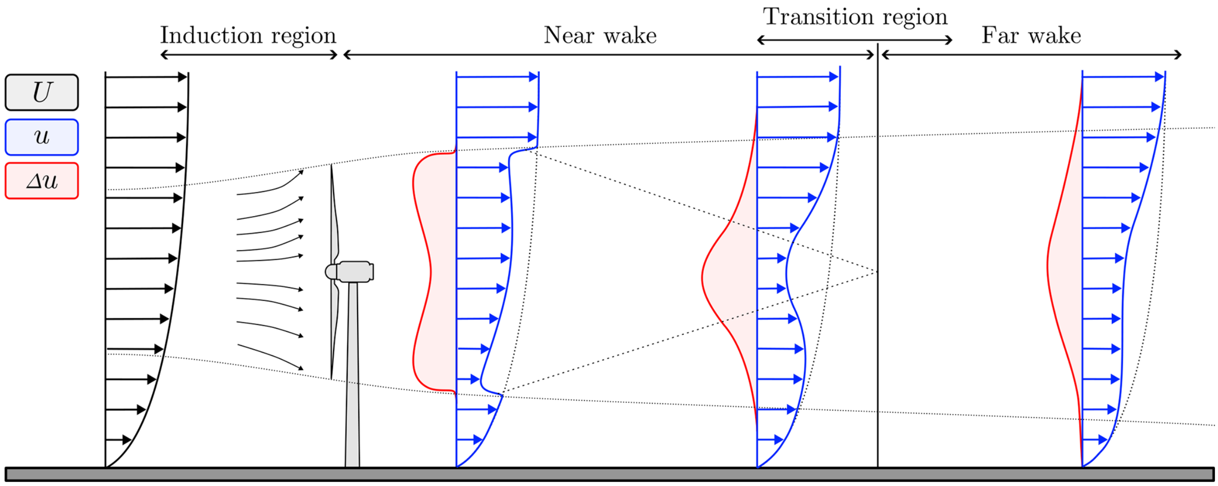

Figure 1. Simplified illustration of the flow around a wind turbine, depicting the upstream induction zone and the downstream wake, which is separated into the near wake, transition region, and far wake. The illustration includes the effective wind speed and the velocity deficit, illustrating the influence of the turbine on the background flow.

and small-scale turbulence of an unsteady wake using separate but integrated model components.

While numerous studies on ANN wake surrogates hypothesize that these models can outperform traditional engineering approaches in flow simulation and WFLO accuracy, numeric validation studies remain limited. Anagnostopoulos and Piggott (2022) trained an ANN surrogate using data from an engineering model and applied it to both yaw optimization for a 15-turbine array and layout optimization for a 6-turbine configuration. Similarly, Sun and Yang (2023) developed an ANN surrogate trained on data from an engineering model and conducted a WFLO study incorporating hub-height optimization for a 30-turbine wind farm. Yang et al. (2022) and Yang and Deng (2023) demonstrated the ANN-RANS surrogates developed by Ti et al. (2021) in yaw optimization of a five-turbine row and in a WFLO re-powering study of the Horns Rev 1 offshore wind farm, respectively. However, neither study validated their results against CFD simulations or against supervisory control and data acquisition (SCADA) data. In contrast, Li et al. (2024b) introduced a novel RANS-based ANN surrogate trained using a generative adversarial network (GAN). Their study included flow simulation via the superposition of single-wake models applied to the Horns Rev 1 wind farm, with power production results compared against RANS, time-averaged LES, and SCADA data, though only for a limited range of inflow cases. Likewise, Liu et al. (2025) trained a convolutional neural network (CNN) surrogate using time-averaged LES data and validated it using time-averaged LES simulations from a row of five turbines with varying inter-turbine spacings.

In summary, the existing validation studies in the literature have been limited by surrogate training and validation using low-fidelity engineering model data, WFLO without validation, flow studies with few turbines or limited variation in the studied farm layouts, and restricted inflow conditions. In this work, systematic experiments are conducted to validate a pre-developed RANS-ANN for flow simulation and WFLO. The contributions of this work include the following two main items, a flow study and a WFLO study:

1a. recalibration of engineering models to a RANS-based single-wake dataset;

1b. comparative study investigating the accuracy of an ANN wake model and two popular engineering model setups for wind farm flow simulation with a varying number of wind turbines, including a cost–benefit analysis of accuracy, memory consumption, and computational cost;

2. WFLO validation study of the ANN wake model against two engineering model setups and three different optimization algorithms, as well as validation of the optimized layouts with RANS-based annual energy production (AEP) estimates.

The paper consists of Sect. 2, which outlines the studies conducted and the methods used. In Sect. 3, the results of the studies are presented and discussed. In Sect. 4, the conclusions of the study are summarized, and suggestions for future work are presented.

2 Methodology

In this section, the methods used in the paper are presented, and the experiments conducted are described.

2.1 Reynolds-averaged Navier–Stokes (RANS)

RANS is a steady-state CFD model that solves the mean flow by modeling all turbulence scales. As RANS represents the

Table 1. Parameters for `PyWakeEllipSys-flatbox` mesh considered during simulations.

Parameter	Description	Considered values
Z_{grid}	Outer flow domain height	$25D$
R_{grid}	Outer flow domain extend	$500D$
M_{N}	Northern inner-domain length	$3D$
M_{E}	Eastern inner-domain length	$20D$
M_{S}	Southern inner-domain length	$3D$
M_{W}	Western inner-domain length	$5D$
Z_M	Inner-box height	$3D$
Δs	Grid spacing	$D/8$

time-averaged wake, it does not resolve unsteady features such as wake meandering, whose net effect on the mean velocity deficit and wake recovery is instead represented implicitly through a calibrated closure model, here the k–ε–f_P (van der Laan et al., 2015a, b).

The inflow represents a logarithmic profile, where the ambient turbulence intensity (TI) based on the turbulent kinetic energy at hub height is set by the roughness length. For the WFLO AEP study, the TI is set to 6 %, yielding an offshore roughness length of 2.5×10^{-4} m. The turbines are modeled by actuator disks (ADs) (Troldborg et al., 2015). The AD thrust and tangential force distributions utilize an analytical Joukowsky rotor model, as proposed by Sørensen et al. (2020) but recalibrated by van der Laan et al. (2020) for improved performance under veer and shear. The RANS simulations are performed with the `PyWakeEllipsys` code (DTU Wind and Energy Systems, 2023), which wraps the `EllipSys3D` finite-volume flow solver (Michelsen, 1992; Sørensen, 1994).

A Cartesian mesh with inner and outer grids was used, as illustrated in Fig. 2. The mesh is fixed with the inflow always directed from west to east, and different wind directions are realized by rotating the wind farm layout rather than the grid. Generic mesh parameters are introduced in Fig. 2, and their specific values are listed in Table 1. The asymmetric extents, $M_{\mathrm{E}} = 20D$ downstream, $M_{\mathrm{N}} = M_{\mathrm{S}} = 3D$ lateral, and $M_{\mathrm{W}} = 5D$ upstream, follow from this orientation: only the downstream direction requires a large buffer to capture the far wake.

The WFLO RANS AEP simulations are performed for all wind directions and wind speeds with intervals of 5° and $1\,\mathrm{m\,s^{-1}}$, respectively, yielding 1584 inflow cases. The simulations are run consecutively, where each new wind speed is obtained by scaling the turbine controllers according to Reynolds number similarity (van der Laan et al., 2019), while the inflow is kept constant. This reduces the total number of required iterations because only local changes need to be recalculated, as discussed by van der Laan et al. (2022). Furthermore, the wind speed cases are simulated from low to high, and the wind speed cases above the wind farm rated wind speed are skipped, which reduce the total number of flow cases by about 40 %. Finally, a relatively low convergence level of 10^{-4} is applied, which reduces the computational effort by an order of magnitude, while the convergence error in terms of AEP is 0.02 %, as shown in van der Laan et al. (2022). A single AEP simulation with this RANS setup takes between 28–33 h employing 1000 cores using the Sophia high-performance computer (HPC) (Technical University of Denmark, 2019), which is equipped with first-generation AMD EPYC 7351 cores (released in 2017). Tests of PyWakeEllipSys RANS simulations on a different HPC with more modern fourth-generation AMD Genoa-X cores (released in 2023) shows a speed-up of a factor of 2.

2.2 Engineering models

In this work, the `PyWake` engineering model framework (Pedersen et al., 2023) was used as the engine to run wind farm simulations. PyWake is a highly efficient framework that provides a high degree of abstraction for the components commonly found in low-fidelity wind farm models. The vast number of module combinations means only a subset can be considered at a time. We have chosen to consider a configuration based on the Turbulence Optimized Park model (TurbOPark) (Nygaard et al., 2020, 2022). The configuration is altered slightly and is therefore generally referred to as the TP model to avoid confusion with the original TurbOPark (Nygaard et al., 2022). The second alternative considered is based on a configuration submitted to the American WAKE Experiment (AWAKEN) (Bodini et al., 2026), which utilizes the super-Gaussian model proposed by Blondel and Cathelain (2020) and is therefore referred to as the SG model.

Regardless of the wake model used to represent the wind farm flow field, wake superposition is required. This is typically performed using either linear or quadratic methods. In the present work, linear superposition is applied for velocity deficits (Δu) in both configurations, while a quadratic max sum is used for added turbulence intensity (I_{a}) in the SG configuration. An overview of superposition methods is provided by Porté-Agel et al. (2020).

For readers already familiar with engineering wind farm models, a summary of the two configurations is shown in Table 2, while the models are presented in greater detail in the remainder of the subsection for readers who are unfamiliar with them. Where possible, the PyWake default superposition model has been used. In the TP configuration, however, the wake and blockage models require different superposition methods: the original TurbOPark model superimposes the wake deficits quadratically (Nygaard et al., 2022), but a quadratic (squared) sum would not preserve the sign of the blockage-induced speed-ups (Forsting et al., 2023). Linear superposition is therefore applied to the velocity deficits so that the wake and blockage contributions can be combined under a single operator.

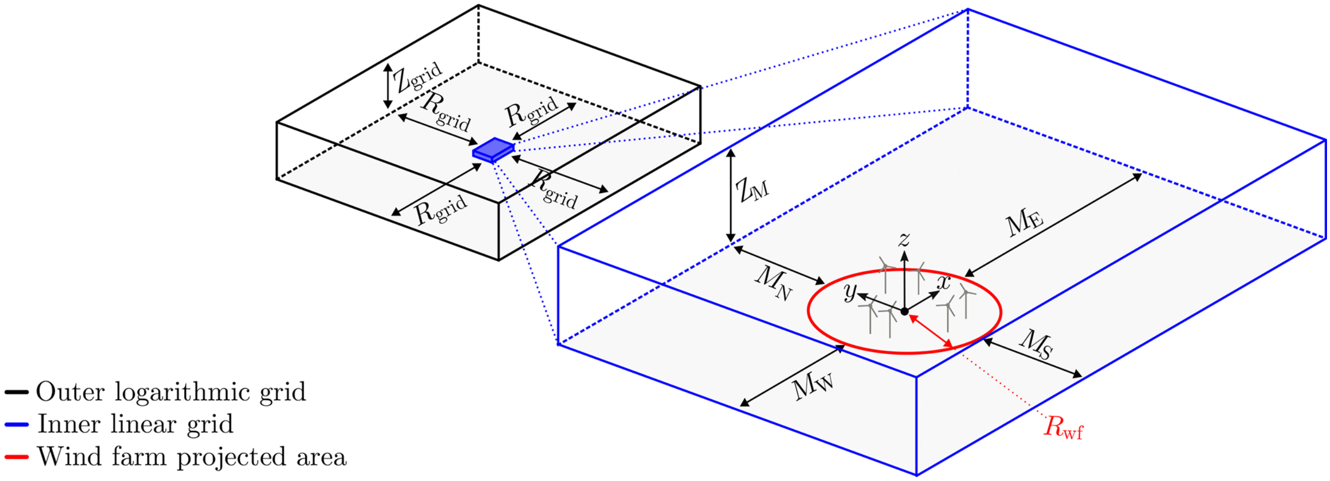

Figure 2. Illustration of utilized `PyWakeEllipsys-flatbox` mesh.

Table 2. Comparison of `PyWake` configurations (with/without blockage). Values flanked by dashes indicate shared settings across both blockage configurations. A single dash (–) indicates that the component is not used.

	SG		TP	
Component	w/o blockage	with blockage	w/o blockage	with blockage
Wake model	– Blondel and Cathelain (2020) –		– Nygaard et al. (2022) –	
Blockage model	–	Forsting et al. (2023) [a]	–	Forsting et al. (2023) [a]
Turbulence model	– Crespo and Hernández (1996) –		–	Crespo and Hernández (1996)
Superposition (Δu)	– Linear –		– Linear –[b]	
Superposition (I_a)	– Quad. max sum –		–	Quad. max sum
Wind farm model	Prop. downwind	All2All iter.	Prop. downwind	All2All iter.
Ground model	– Mirror –		– Mirror –	
Rotor averaging	– Center –		– Gaussian overlap –	

[a] Updated version of Troldborg and Meyer Forsting (2017).
[b] In the original TurbOPark paper, quadratic superposition was employed; however, this approach is incompatible with the blockage model (Forsting et al., 2023).

Turbo Park (TP) model

The TP model uses the updated Turbulence Optimized Park (TurbOPark) model (Nygaard et al., 2022), a Gaussian wake model derived from the classic Gaussian wake model by (Bastankhah and Porté-Agel, 2014). The wake model is shown in Eq. (1):

$$\frac{\Delta u(x)}{U} = C(x)\exp\left(-\frac{r^2}{2\sigma_w^2(x)}\right), \tag{1a}$$

$$C(x) = 1 - \sqrt{1 - \frac{C_\mathrm{T}}{8(\sigma_w(x)/D)^2}}, \tag{1b}$$

where $\Delta u(x)$ is the wake deficit at the downstream position x, U is the free-stream velocity, $C(x)$ is the peak velocity deficit at the wake centerline, r is the radial distance from the wake center, C_T is the coefficient of thrust, and $\sigma_w(x)$ is the variable wake expansion factor, which is the major difference between the TurbOPark implementation and the default Gaussian deficit model which uses a constant wake expansion factor. The expression for the wake expansion factor $\sigma_w(x)$ is implemented using the Frandsen (2005) turbulence model, which is why the TurbOPark model does not require a standalone added-turbulence model. $\sigma_w(x)$ is shown in Eq. (2) parameterized with α and β from Frandsen (2005):

$$\frac{\sigma_w(x)}{D} = \varepsilon + \frac{AI_0}{\beta}\cdot\left(\sqrt{(\alpha+\beta x/D)^2+1} - \sqrt{1+\alpha^2} - \ln\left[\frac{\left(\sqrt{(\alpha+\beta x/D)^2+1}+1\right)\alpha}{\left(\sqrt{1+\alpha^2}+1\right)(\alpha+\beta x/D)}\right]\right), \tag{2a}$$

$$\alpha := c_{I,1}I_0,\ \ \beta := \frac{c_{I,2}I_0}{\sqrt{C_\mathrm{T}}} \tag{2b}$$

$$\varepsilon := c_\varepsilon\sqrt{\frac{1}{2}\frac{1+\sqrt{1-\widehat{C}_\mathrm{T}}}{\sqrt{1-\widehat{C}_\mathrm{T}}}},$$

$$\text{where } \widehat{C}_\mathrm{T} := \min\left(C_\mathrm{T},\, C_{\mathrm{T,lim}}\right), \tag{2c}$$

where I_0 is the ambient TI, A is a wake expansion calibration parameter, $c_{I,1}$ and $c_{I,2}$ are calibration parameters from the Frandsen (2005) model, and ε is an initial characteristic wake width; i.e., $\sigma_w(0)/D = \varepsilon$. Here, it is important to note that the engineering wake models operate under an assumption that $I_k = I_u$, where $I_k = \sqrt{2/3 \cdot k}/U$ is the TI derived from turbulent kinetic energy, and $I_u = \sigma_u/U$ is the streamwise TI based on the standard deviation (SD) of the flow in the streamwise direction σ_u. This differs from the more physically accurate relationship $I_k \approx 0.8 \cdot I_u$, derived from standard atmospheric turbulence ratios (Panofsky and Dutton, 1984). However, this simplification is not expected to alter the main conclusions, as the recalibrated models exhibit similar overall trends.

Super-Gaussian (SG) model

The SG model, as previously mentioned, uses the super-Gaussian model proposed by Blondel and Cathelain (2020); here, we present the simplified analytical version. The model unifies the observation that the far wake can be accurately modeled as a Gaussian shape, whereas the near wake more closely resembles a top-hat shape. By augmenting the super-Gaussian shape parameter n, the velocity deficit takes on different forms; i.e., when $n = 2$, it becomes a regular Gaussian shape, while for $n > 2$ the shape gradually becomes more top-hat-like.

$$\frac{\Delta u(x)}{U} = C(x) \exp\left(-\frac{(r/D)^n}{2(\sigma(x)/D)^2}\right) \tag{3a}$$

$$C(x) = 2^{2/n-1} - \sqrt{2^{4/n-2} - \frac{nC_\mathrm{T}}{16\Gamma(2/n)(\sigma(x)/D)^{4/n}}} \tag{3b}$$

$$\frac{\sigma(x)}{D} = (a_s I + b_s) x/D + c_s \sqrt{\frac{1}{2}\frac{1+\sqrt{1-C_\mathrm{T}}}{\sqrt{1-C_\mathrm{T}}}} \tag{3c}$$

$$n \approx a_f \exp\left(b_f \cdot x/D\right) + c_f, \quad a_f := 3.11 \tag{3d}$$

Here Γ is the gamma function, and I is the effective TI. Equation (3d) shows the exponential function that Blondel and Cathelain (2020) created to avoid having to solve for n and is the key change to enable fast evaluations of the flow. The effective TI is evaluated using the added-TI (I_a) model proposed by Crespo and Hernández (1996) and the quadratic sum. The Crespo and Hernández (1996) model is shown in Eq. (4). To evaluate the axial induction (a_m), the method by Madsen et al. (2020), as implemented in PyWake, is utilized, and a quadratic max summation is used for superposition:

$$I_\mathrm{a} = 0.73 a_\mathrm{m}^{0.8325} I_0^{-0.0325} (x/D)^{-0.32}, \tag{4a}$$

$$a_\mathrm{m} = 0.083 C_\mathrm{T}^3 + 0.0586 C_\mathrm{T}^2 + 0.2460 C_\mathrm{T}, \tag{4b}$$

$$I = \sqrt{I_0^2 + \max_j\left(I_{\mathrm{a},j}\right)^2}, \tag{4c}$$

where j is an index indicating a neighboring turbine.

Blockage

While wake effects are the primary drivers of flow interactions between turbines, turbine induction also plays a role. In PyWake, these effects are modeled using blockage models. When blockage is considered, the Self-Similarity Deficit model by Troldborg and Meyer Forsting (2017), as updated by Forsting et al. (2023), is employed:

$$\frac{\Delta u_b}{U} = a_0(x, C_\mathrm{T}) \nu(x) \operatorname{sech}^{c_\alpha}\left(c_\beta \frac{r}{r_{1/2}(x)}\right), \tag{5a}$$

$$\nu(x) = \left(1 + \frac{x/R}{\sqrt{1+(x/R)^2}}\right), \tag{5b}$$

where ν is the centerline induction, a_0 is the axial induction factor, R is the rotor radius, and $r_{1/2}(x)$ is a linear induction zone half radius, defined as

$$\frac{r_{1/2}(x)}{R} = c_\lambda \cdot (x/R) + c_\eta, \tag{5c}$$

$$a_0(x, C_\mathrm{T}) = \frac{1}{2}\left(1 - \sqrt{1 - \gamma(x, C_\mathrm{T}) \cdot C_\mathrm{T}}\right), \tag{5d}$$

where c_λ and c_η are tunable parameters introduced to account for the effects of wind farm blockage. In the updated blockage model (Forsting et al., 2023), the $\gamma(x, C_\mathrm{T})$ function was updated to gradually shift from a far-field formulation to a near-field formulation. The said γ function is introduced below, with an abstraction of the transition function reported as a $\delta(x)$ function:

$$\delta(x) = \begin{cases} 1 & \text{for } x/R < -6 \\ \frac{|\nu(x)-\nu|}{\nu(-6)-\nu(-1)} & \text{for } -6 \leq x/R \leq -1 \\ 0 & \text{for } -1 < x/R \end{cases} \tag{5e}$$

$$\begin{aligned} \gamma(x, C_\mathrm{T}) = \Big\{ & \delta(x), \\ & \times \left(-0.06489 \sin\left(\frac{C_\mathrm{T} - 0.4911}{-0.1577}\right) + 1.116\right) \\ & + (1 - \delta(x)) \cdot \left(C_\mathrm{T}^3 a_\mathrm{nf} + C_\mathrm{T}^2 b_\mathrm{nf} + C_\mathrm{T} c_\mathrm{nf} + d_\mathrm{nf}\right) \Big\}. \end{aligned} \tag{5f}$$

The near-field expressions are parameterized with a_nf, b_nf, c_nf, and d_nf. For the far-field parameters, the original values are used as reported in Eq. (5f).

2.3 Recalibration of deficit models with RANS data

To ensure that the deficit models considered are compared fairly, they have been recalibrated to a single-wake RANS dataset. To construct the dataset, a pre-existing RANS lookup table (LUT) is used, which consists of 3D fields of velocity deficit and wake-added TI for all combinations of $C_T \in [0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 0.923]$

and $I_0 \in [5, 10, 20, 30]$ %. The inflow represents a logarithmic profile where the roughness length is used to set I_0, as described in Sect. 2.1. The single-wake RANS dataset is constructed by linearly interpolating/extrapolating the RANS-LUT to wind speeds $U \in [4, 5, 6, 7, 8, 9, 10]\,\mathrm{m\,s^{-1}}$ and turbulence intensities $I_0 \in [5, 10, 15, 20, 25, 30, 35, 40]$ %, covering the modeling space. Similar RANS single-wake databases have been used to construct surrogate models in previous work (Criado Risco et al., 2023; van der Laan et al., 2026).

The Bayesian optimization framework proposed by Nogueira (2014) is used to minimize the L^2 norm between the hub-height velocity fields of the RANS data and the deficit models. For each flow case, an L^2 norm is calculated, and the mean of those is used as the optimization objective function. The deficit models are calibrated to account for velocities between 2–10 rotor diameters downstream of the source turbine and up to 2 rotor diameters in the cross-flow direction. The upstream deficit model is calibrated considering the region between 1–2 rotor diameters upstream of the source turbine and up to 2 rotor diameters away in the cross-flow direction. The optimization is initialized with 50 parameter configurations sampled from uniform distributions (detailed in Table 3). A set of parameters is denoted in vector form as $\boldsymbol{\phi}_i$ and a collection of them in matrix notation as $\boldsymbol{\Phi}$. A given set of parameters constitutes a realization of the wake deficit parameters shown in Table 3; the original parameters are not shown here but are included inside the results section with the recalibrated parameters in Table 7.

This implementation of Bayesian optimization uses Gaussian process (GP) regression. Here, the posterior predictive distribution for a test input $\boldsymbol{\phi}_*$ is given by

$$f(\boldsymbol{\phi}_*) \mid \boldsymbol{\Phi}, \boldsymbol{y} \sim \mathcal{N}\left(\mu_{\mathrm{GP}}(\boldsymbol{\phi}_*), \sigma^2_{\mathrm{GP}}(\boldsymbol{\phi}_*)\right), \tag{6a}$$

$$\mu_{\mathrm{GP}}(\boldsymbol{\phi}_*) = \boldsymbol{k}_*^\top \left(\mathbf{K} + \sigma_n^2 \mathbb{I}_n\right)^{-1} \boldsymbol{y}, \tag{6b}$$

$$\sigma^2_{\mathrm{GP}}(\boldsymbol{\phi}_*) = k(\boldsymbol{\phi}_*, \boldsymbol{\phi}_*) - \boldsymbol{k}_*^\top \left(\mathbf{K} + \sigma_n^2 \mathbb{I}_n\right)^{-1} \boldsymbol{k}_*, \tag{6c}$$

where $\mathcal{N}(\mu, \sigma^2)$ denotes a Gaussian distribution with mean μ and SD σ, $\boldsymbol{y}$ is the vector of observed outputs at the n training points, σ_n can be used to account for observation and model uncertainty but is just included here for numerical stability with $\sigma_n = 10^{-3}$, $\mathbb{I}_n$ is the $n \times n$ identity matrix, and μ_{GP} and σ_{GP} are the mean and SD obtained through GP regression. The covariance matrix $\mathbf{K}$ and test covariance vector $\boldsymbol{k}_*$ are constructed using a pre-specified kernel function. This framework uses a Matérn kernel for the GP regressor with smoothness parameter $\nu_{\mathrm{M}} = 2.5$ and length scale set to 1, which can therefore be removed as it only occurs in denominators; the Matérn kernel is shown in Eq. (7).

$$k(\boldsymbol{\phi}_i, \boldsymbol{\phi}_j) = \frac{1}{\Gamma(\nu_{\mathrm{M}})2^{\nu_{\mathrm{M}}-1}} \left(\sqrt{2\nu_{\mathrm{M}}}\|\boldsymbol{\phi}_i - \boldsymbol{\phi}_j\|_2\right)^{\nu_{\mathrm{M}}} \times K_{\nu_{\mathrm{M}}}\left(\sqrt{2\nu_{\mathrm{M}}}\|\boldsymbol{\phi}_i - \boldsymbol{\phi}_j\|_2\right) \tag{7}$$

Table 3. Parameter bounds for the considered wake models during the calibration process: the SG model (Blondel and Cathelain, 2020), the TurbOPark-based model (TP; Nygaard et al., 2022), and the blockage model (Forsting et al., 2023).

Model	Parameter	Lower	Upper
SG	a_s	0.001	0.5
	b_s	0.001	0.01
	c_s	0.001	0.5
	b_f	−2	1
	c_f	0.1	5
TP	A	0.001	0.5
	$c_{I,1}$	0.01	5
	$c_{I,2}$	0.01	5
	c_ϵ	0.01	3
	$C_{\mathrm{T,lim}}$	0.01	1
Blockage	c_α	0.05	3
	c_β	0.05	3
	c_λ	−2	2
	c_η	−2	2
	a_{nf}	−3	3
	b_{nf}	−3	3
	c_{nf}	−3	3
	d_{nf}	−3	3

Here $k(\boldsymbol{\phi}_i, \boldsymbol{\phi}_j)$ is the Matérn covariance between parameter sets $\boldsymbol{\phi}_i$ and $\boldsymbol{\phi}_j$, $K_{\nu_{\mathrm{M}}}$ is the modified Bessel function of the second kind, and Γ is the gamma function. The elements of the covariance matrix $\mathbf{K}$ are given by $K_{ij} = k(\boldsymbol{\phi}_i, \boldsymbol{\phi}_j)$, and the test vector is

$$\boldsymbol{k}_* = \begin{bmatrix} k(\boldsymbol{\phi}_1, \boldsymbol{\phi}_*) & k(\boldsymbol{\phi}_2, \boldsymbol{\phi}_*) & \cdots & k(\boldsymbol{\phi}_n, \boldsymbol{\phi}_*) \end{bmatrix}^\top. \tag{8}$$

To guide the selection of the next evaluation point, an acquisition function is required. This implementation uses the upper confidence bound (UCB) acquisition function to decide the next best set of parameters. In this implementation, an exploration parameter of $\kappa = 2.576$ is used for UCB, as shown in Eq. (9).

$$\mathrm{UCB}(\boldsymbol{\phi}_*) = g\left(f(\boldsymbol{\phi}_*) \mid \boldsymbol{\Phi}, \boldsymbol{y}\right) = \mu_{\mathrm{GP}}(\boldsymbol{\phi}_*) + \kappa\, \sigma_{\mathrm{GP}}(\boldsymbol{\phi}_*) \tag{9}$$

The Bayesian optimization is run for 200 sequential iterations. In each iteration, the GP model is fitted to all previously observed parameter sets, the UCB acquisition function is maximized to select the following parameter set $\boldsymbol{\phi}_*$ to evaluate, and the objective function is computed at this point. After all iterations, the parameters associated with the lowest error are reported.

2.4 Artificial neural network (ANN) surrogate

In this work, a pre-trained RANS-based wake surrogate developed by Pish et al. (2024), based on the work of Schøler et al. (2023), is employed for wind farm simulations and

Table 4. Architecture of the pre-trained RANS ANN surrogates for the velocity deficit (Δu) and added-turbulence (I_a) outputs.

Parameter	Δu model	I_a model
x range	$[-3D, 70D]$	$[-3D, 70D]$
y range	$[-6D, 6D]$	$[-6D, 6D]$
Neurons per layer	(70, 102, 102, 102)	(118, 118, 118, 118)
Activation (σ)	(tanh, $3 \times$ sigmoid)	($4 \times$ sigmoid)
Parameters	28 847	43 071

Table 5. Configuration used to deploy the ANN surrogates as wake models in PyWake.

Setting	Value
Superposition (Δu)	Linear
Superposition (I_a)	Linear
Wind farm model (without blockage)	Propagate downwind
Wind farm model (with blockage)	All2All iterative
Ground model	Mirror
Rotor averaging	Center

WFLO. The surrogate is based on a conventional ANN architecture, commonly referred to as a deep neural network, a fully connected feed-forward neural network, or a multilayer perceptron (MLP). The model takes the Cartesian position in space (x, y, z), the yaw offset from the inflow direction (γ), C_T, and the TI as inputs. While TI is considered during our experiments, yaw misalignment is not. Not addressing yaw misalignment was a conscious decision to limit the scope of the paper as it would have entailed a significant increase in the number of cases to evaluate with RANS.

In Eq. (10) the MLP is defined as a set of equations. Equation (10a) shows the input $\boldsymbol{s}$, Eq. (10b) shows the hidden layer l, and Eq. (10c) shows the output of the MLP $f_\theta(\boldsymbol{s})$.

$$\boldsymbol{h}^{(0)} = \boldsymbol{s} = \begin{bmatrix} x, & y, & z, & \gamma, & C_\mathrm{T}, & I \end{bmatrix}^\top \tag{10a}$$

$$\boldsymbol{h}^{(l)} = \sigma^{(l)}\left(\mathbf{W}^{(l)}\boldsymbol{h}^{(l-1)} + \boldsymbol{b}^{(l)}\right), \quad l = 1, 2, \ldots, L-1 \tag{10b}$$

$$f_\theta(\boldsymbol{s}) = \mathbf{W}^{(L)}\boldsymbol{h}^{(L-1)} + \boldsymbol{b}^{(L)} \tag{10c}$$

Here $\boldsymbol{h}^{(l)}$, $\sigma^{(l)}$, $\mathbf{W}^{(l)}$, and $\boldsymbol{b}^{(l)}$ are the activation, activation function, weights, and bias at layer l, respectively, and L is the total number of layers. The model is trained with a single-wake RANS dataset of the DTU 10 MW reference wind turbine (Bak et al., 2013). The architectures of the employed models are summarized in Table 4, and a brief overview of the engineering model configuration for the ANN is presented in Table 5. The configuration was adopted in accordance with Pish et al. (2024).

2.5 Wind farm flow study

To evaluate the accuracy of the three low-fidelity options – the two engineering models and the ANN surrogate – we conduct a systematic study examining the influence of turbine count and inter-turbine spacing. Because flow construction based on superposition inherently introduces errors that depend on these parameters, varying them enables a structured investigation of construction error and facilitates a direct comparison of the engineering models and the ANN. Additionally, the study examines how these factors impact memory requirements and computational costs.

To examine the effect of turbine count, six base layouts are considered with $n_\mathrm{wt} \in \{6, 10, 20, 30, 40, 50\}$. Figure 3 illustrates these configurations. To assess the impact of inter-turbine spacing, each base layout is scaled by a separation factor $s_\mathrm{wf} \in \{3D, 5D, 7D\}$, resulting in a total of 18 distinct layouts. Each layout is simulated using RANS, the recalibrated and original engineering models, and the ANN surrogate. By comparing these results, trends in model performance can be identified, and the effectiveness of the ANN surrogate can be evaluated.

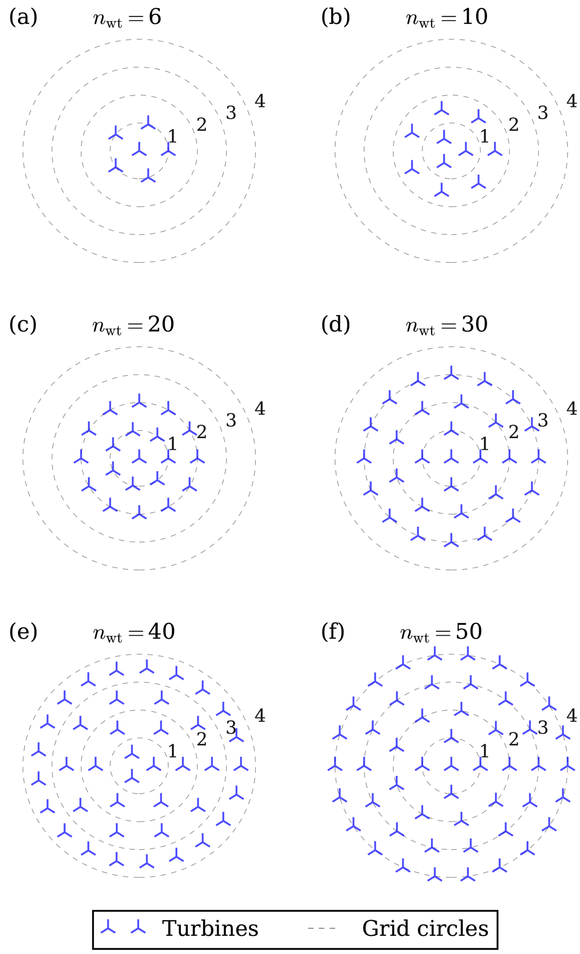

Figure 3. Circular turbine layouts evaluated in the wind farm flow study. The grid circles represent normalized radial positions, which are scaled by a separation factor $s_\mathrm{wf} \in \{3D, 5D, 7D\}$ to generate configurations with varying inter-turbine spacing.

In the flow study, each possible model-layout combination is evaluated under different inflow conditions consisting of wind speeds $U = 8.0\,\mathrm{m\,s^{-1}}$ and $14\,\mathrm{m\,s^{-1}}$, ambient turbulence intensities $I = 5\,\%$ and $10\,\%$, and wind direction angles θ ranging from 270 to 315° in increments of 3°.

2.6 Wind farm layout optimization (WFLO)

To compare the low-fidelity models in WFLO, optimization is performed using each of the available low-fidelity models, and the results are validated by estimating the annual energy production (AEP) with RANS simulation. A WFLO case is formulated with the objective of maximizing AEP, subject to a minimum turbine separation $s_{\min} = 2D$ and polygonal boundary $\mathcal{P}$:

$$\max_{\{(x_i,y_i)\}_{i=1}^{n_{\mathrm{wt}}}} \mathrm{AEP}\left(\{(x_i,y_i)\}_{i=1}^{n_{\mathrm{wt}}}\right)$$

$$\approx 8760 \sum_{i=1}^{n_{\mathrm{wt}}} \sum_{j=1}^{n_{\mathcal{U}}} \sum_{k=1}^{n_{\theta}} P_i(\{(x_m,y_m)\}_{m=1}^{n_{\mathrm{wt}}}, \mathcal{U}_j, \theta_k) \rho(\mathcal{U}_j, \theta_k)$$

$$\text{s.t.} \sqrt{(x_i - x_q)^2 + (y_i - y_q)^2} \geq s_{\min}, \quad \forall i \neq q$$

$$(x_i, y_i) \in \mathcal{P}, \quad \forall i = 1, \ldots, n_{\mathrm{wt}}, \tag{11}$$

where x_i, y_i, and P_i are coordinates and power produced at turbine i, 8760 is the number of hours in a year, and n_{wt} is the number of turbines. ρ is a probability mass function indicating the probability of seeing a given combination of wind speed bin $\mathcal{U}_j$ and binned wind direction θ_k. During optimization, the AEP is estimated with a wind direction bin for every 1° to avoid artificially unwaked areas which the optimization algorithms would exploit. However, during validation the bin discretization was decreased to only consider a bin for every 5° to conserve computational resources driven by the cost of RANS simulations, as discussed in Sect. 2.1.

The optimization is performed with the DTU-developed software `Topfarm2` (Pedersen et al., 2025). Three different gradient-based optimization algorithms are considered, as implemented in `Topfarm2`: sequential least squares quadratic programming (SLSQP) (Kraft, 1988; Virtanen et al., 2020), interior point optimizer (IPOPT) (Wächter and Biegler, 2006), and stochastic gradient descent (SGD) (Quick et al., 2023). In all cases, the gradients for the optimization are provided by the automatic differentiation included in PyWake.

The wind climate used in the layout study is taken from the IEA Wind 740-10-MW reference offshore wind plant (Kainz et al., 2024): the wind rose in Fig. 4b is adopted as the site inflow distribution. The boundary was designed for this study to accommodate many turbines inside a small area, while maintaining both sharp and soft edges. The RANS cost scales with the modeled area because the wind directions are modeled by rotating the farm. A circular boundary minimizes this area and is shown here as the smallest circle enclosing the farm (radius R_{wf} in Fig. 4a). The turbines are

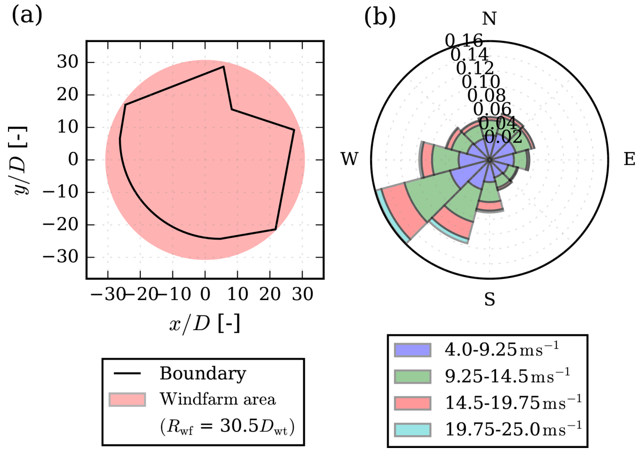

Figure 4. (a) Boundary constraint for the considered layout optimization. **(b)** Wind rose at the considered IEA 740-10-MW reference site.

modeled as the DTU 10 MW reference wind turbine (Bak et al., 2013). The initial and optimized coordinates for the best optimization runs are shown in Appendix B. To simplify both optimization and validation, the ambient TI was set to $I_0 = 6\,\%$ for all flow cases. The layout optimization procedure is performed for each low-fidelity model, both with and without a blockage model, and for the engineering models in their original and recalibrated forms, using multiple optimization algorithms and random initial layouts. For cases without blockage, all three optimization algorithms, SLSQP, IPOPT, and SGD, are used, each initialized from the same set of five randomly generated layouts to enable a consistent comparison across algorithms.

For cases including the blockage model, 10 random initial layouts are considered; however, only the SGD algorithm is used, as it is less memory-intensive than the other approaches due to batch sampling of the wind rose. From the resulting set of optimized layouts, the layout yielding the highest AEP is selected for each low-fidelity model and subsequently validated using RANS simulations, thereby limiting the total number of RANS cases to one per low-fidelity model.

2.7 Evaluation metrics

To evaluate the relative performance of the different models, quantitative measures are needed. In this work, we use the mean absolute percentage error (MAPE), the root mean square error (RMSE), the maximum absolute error ($E_{\max}$), and the error standard deviation (σ_E).

$$\mathrm{MAPE} = \frac{1}{N} \sum_{i=1}^{N} \left| \frac{u_i - u_i'}{U_i} \right| \cdot 100 \tag{12a}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} \left(u_i - u_i'\right)^2} \tag{12b}$$

Table 6. Parameters of considered optimization algorithms as used during studies with `TopFarm2`.

Parameter	SLSQP	IPOPT	SGD
Max iterations	1000	1000	2000
Convergence tolerance	1×10^{-3}MWh	1×10^{-2}	–
Initial learning rate	–	–	17.83 m
Momentum weight 1 (β_1)	–	–	0.1
Momentum weight 2 (β_1)	–	–	0.2
Final learning rate	–	–	0.01 m
Initial constant learning rate iterations	–	–	100
Early stopping ratio	–	–	0.1

$$E_{\max} = \max_i |u_i - u_i'| \tag{12c}$$

$$\sigma_E = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}\left((u_i - u_i') - \frac{1}{N}\sum_{j=1}^{N}(u_j - u_j')\right)^2} \tag{12d}$$

Here u_i is the target wind speed for observation i, u_i' is the predicted wind speed, and N is the total number of observations. In Eq. (12a) the denominator uses the free-stream velocity U_i rather than u_i, in order to avoid numerical issues in heavily waked regions, where low wind speeds can lead to disproportionately large relative errors. Conversely, this formulation prevents large errors in unwaked regions from being underrepresented in the overall metric.

For flow field comparisons, regions within 1 rotor diameter of the turbines are omitted, making the analysis domain (Ω) a subset of the whole flow domain (Ω_{flow}). The whole flow domain is the hub-height flow plane, with grid parameters provided in Table 1 and shown in Fig. 2.

$$\Omega = \left\{ (x,y) \in \Omega_{\text{flow}} \,\middle|\, \sqrt{(x-x_i)^2 + (y-y_i)^2} \geq D \quad \forall i = 1,2,\ldots,n_{\text{wt}} \right\} \tag{13}$$

Where this subset is used, the metric will be marked with an Ω subscript.

3 Results and discussion

This section presents and discusses the results of the three studies introduced in Sect. 2. First, the recalibration of the two engineering models against the RANS single-wake dataset is reported. Next, the wind farm flow study compares the ANN surrogate and both the recalibrated and the originally calibrated engineering models against RANS across varying turbine counts and inter-turbine spacings and assesses their computational cost and memory requirements. Finally, the WFLO study validates optimized layouts obtained with each of the low-fidelity models against RANS-based AEP calculations.

3.1 Recalibration of wake models with RANS data

The recalibrated model parameters are summarized in Table 7, where the original parameter values are also included. Overall, the changes are not dramatic: the sign of each parameter is preserved. However, several parameters do show substantial numerical differences.

For the SG model, most changes are relatively small, except for a_s, which increases from 0.17 to 0.33. This parameter governs the relationship between wake width and rotor thrust. Because the original calibration was based on a mix of experimental data, whereas the present work uses RANS data, different thrust and wake-width characteristics are expected; therefore, a noticeable change in a_s is expected.

For the TP model, the parameter $c_{I,2}$ changes markedly from 0.80 to approximately 3.15. The parameters $c_{I,1}$ and $c_{I,2}$ impact the auxiliary variables α and β in Nygaard et al. (2022), which together act as replacements for both the ambient TI and C_{T} in the wake expansion formulation. The initial values for $c_{I,1}$ and $c_{I,2}$ were used from Frandsen (2005) and the IEC (2010). The original value was derived from measurement data, whereas the present RANS simulations employ a fundamentally different turbulence formulation. Consequently, a substantial adjustment is required for consistency with the RANS flow. Note that the recalibrated $C_{\text{T,lim}}$ of the TP model lies below the maximum C_{T} of the DTU 10 MW thrust curve (0.9), so it acts as an active wake-shaping parameter at high thrust rather than only as a $C_{\text{T}} \geq 1$ safeguard. The SG model lacks an equivalent parameter, so the two configurations were not calibrated on a fully symmetric parameter set. Future comparisons should be mindful of this difference and treat the clip consistently across models. Furthermore, a lower value of both c_ϵ and $C_{\text{T,lim}}$ makes the initial wake narrower for high C_{T} values above the limit. This tends to compensate for the higher A value, which leads to greater wake spreading.

In the self-similarity deficit blockage model, the polynomial-fit parameters a_{nf}, b_{nf}, c_{nf}, and d_{nf} also vary considerably. Because these coefficients describe a polynomial rather than a directly interpretable physical relationship, the practical implications of their changes are difficult to interpret. The parameters c_β and c_λ, however,

Table 7. Original and recalibrated parameters for the considered wake and blockage models.

	SG (Blondel and Cathelain, 2020)				
	a_s	b_s	c_s	b_f	c_f
Orig.	0.1700	0.0050	0.2000	−0.6800	2.4100
Recalib.	0.3332	0.0048	0.1625	−0.4242	2.9034
	TP (Nygaard et al., 2022)				
	A	$c_{I,1}$	$c_{I,2}$	c_ϵ	$C_{\mathrm{T,\,lim}}$
Orig.	0.0400	1.5000	0.8000	0.2500	0.9990
Recalib.	0.2817	1.2016	3.1484	0.1421	0.7751
	Blockage (Forsting et al., 2023)				
	c_α	c_β	c_λ	c_η	
Orig.	0.8889	1.4142	−0.672	0.4897	
Recalib.	0.6208	2.6617	−0.5066	1.8932	
	a_{nf}	b_{nf}	c_{nf}	d_{nf}	
Orig.	−1.381	2.627	−1.524	1.3360	
Recalib.	−2.1531	2.0948	−2.7612	2.9041	

Table 8. Metrics before and after recalibration for the two considered models.

	SG (Blondel and Cathelain, 2020)			
	RMSE	$E_{\max}$	σ_E	MAPE
Orig.	0.0354	0.2224	0.0347	0.3460
Recalib.	0.0205	0.1588	0.0194	0.2005
	TP (Nygaard et al., 2022)			
	RMSE	$E_{\max}$	σ_E	MAPE
Orig.	0.0716	0.2679	0.0681	0.6733
Recalib.	0.0167	0.2850	0.0165	0.1563
	Blockage (Forsting et al., 2023)			
	RMSE	$E_{\max}$	σ_E	MAPE
Orig.	0.0039	0.0131	0.0023	0.0500
Recalib.	0.0019	0.0079	0.0018	0.0201

are more interpretable and show meaningful shifts. The parameter c_λ represents an offset in the linear rotor half-radius induction zone, introduced to correct for wind farm blockage effects – effects that are absent in the single-turbine RANS dataset used here. A significant change is therefore expected. The scaling factor c_β, which modulates the influence of the rotor half radius, is similarly affected.

In Table 8, error metrics are collected to compare results before and after recalibration. The error metrics in Table 8 show improvements across all measures for the SG and blockage models. For the TP model, most metrics also improve, except for the maximum absolute error. This exception was surprising, but as the remaining metrics improved, it is expected that this was due to a large error near the rotor, where the self-similar far-wake assumption underlying the Gaussian deficit does not yet hold (Blondel and Cathelain, 2020). This idea is supported by inspecting the spatial distribution of the mean RMSE (averaged over inflow conditions) in Fig. 5d, which indicates that the most significant errors occur immediately behind the turbine at $x = 2D$.

The overall takeaway from Fig. 5 is that the recalibrated models yield substantially improved performance, particularly outside the near wake. The TP model exhibits the greatest overall improvement, which aligns with expectations: the original TurbOPark model of Nygaard et al. (2020) was calibrated against wind farm flow data, encompassing both intra-farm and inter-farm wake interactions rather than isolated single wakes, and therefore requires the largest recalibration to perform well in the single-wake context evaluated here. By contrast, the original super-Gaussian model by Blondel and Cathelain (2020) was calibrated on single-wake data and requires comparatively little correction in this setting.

3.2 Wind farm flow study

The layouts introduced in Fig. 3 were evaluated with separation factors $s_{\mathrm{wf}} \in \{3D, 5D, 7D\}$. Flow simulations were performed using all the available low-fidelity models (SG model, TP model, and the ANN), with the engineering models assessed in both recalibrated and original forms. To distinguish the recalibrated models from the original models, a superscript (†) is used with the original models; e.g., SG† refers to the SG model with original parameters. Similarly, to distinguish between the models with and without blockage, a superscript (*) is used to indicate the inclusion of blockage; e.g., SG$^{\dagger *}$ indicates the original SG model with blockage.

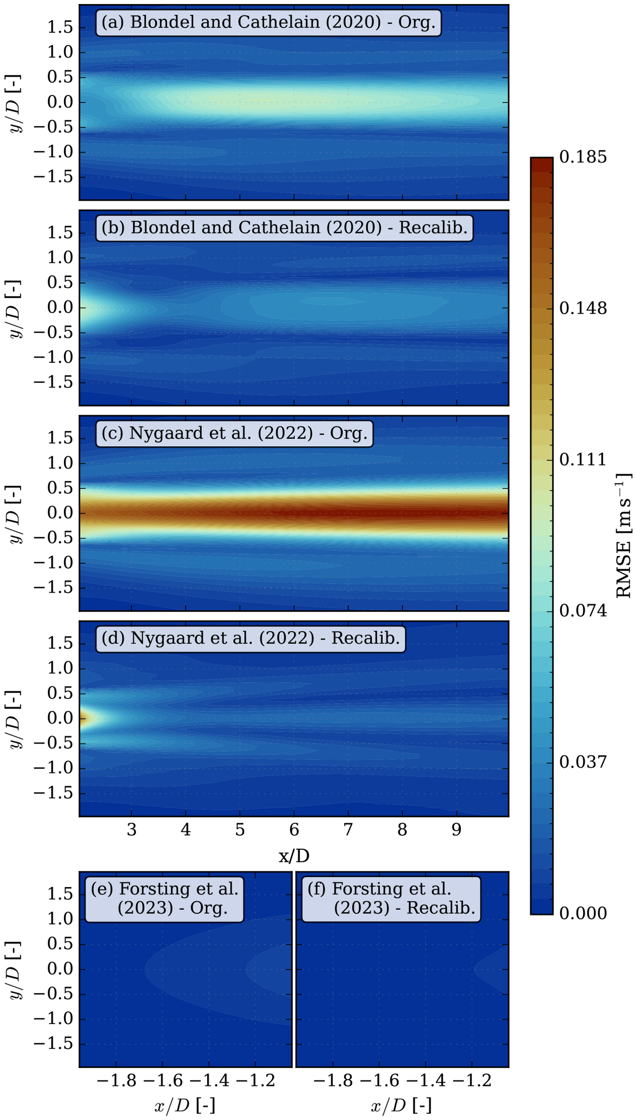

Figure 5. Wake and blockage model calibration RMSE error maps.

To quantify the accuracy of the lower-fidelity simulations, the RANS simulations described in Sect. 2.1 were used as the reference ground truth. A masked flow domain, defined in Eq. (13), was applied to exclude the region immediately surrounding each turbine, as this region is irrelevant for optimization and does not represent a feasible turbine spacing. Figure 6 presents MAPE_{Ω} and RMSE_{Ω} as functions of the number of turbines n_{wt}, with mean values indicated by solid lines and bootstrapped 95 % confidence intervals shown as shaded regions. The ensemble mean is constructed from the considered inflow conditions, including wind directions, wind speeds, and ambient TIs.

Figure 6 presents the results for the models without blockage; for completeness, the metrics with blockage are reported in Appendix A.

The results show that the error consistently decreases as the separation factor increases. This trend reflects the increasing complexity of the wake interactions at smaller inter-turbine distances. In these regimes, near-wake dynamics violate key assumptions in simplified wake models, particularly the superposition of individual wakes, which becomes progressively less valid as group effects strengthen (e.g., Andersen et al., 2014; Gunn et al., 2016). Simultaneously, the wake of a single turbine also behaves in a more complex manner in the near wake, further complicating the modeling of denser farms. Similarly, there is a shared trend of the error growing with the number of turbines, though at times it is subtle, which can be explained by the modeling error of superposing wakes, becoming increasingly problematic as the number of turbines increases. The trend in regard to the number of turbines is not as pronounced as for the separation factor; for interested readers, a version of Fig. 6 with a logarithmic axis is available in Appendix C. These trends between the error and the number of turbines and turbine density are observed for both the recalibrated and the original engineering models. It is also observed that the SG model with the lowest recalibration error is the one with the lowest error here, increasing confidence in the recalibration process.

Furthermore, calibration improves the engineering models across both configurations, reducing the mean error and the width of the confidence intervals, indicating improvements in accuracy and robustness across wind directions and inflow conditions. An exception occurs for the MAPE_{Ω} of the TP model at $s_{\mathrm{wf}} = 3D$, where calibration produces a lower mean error, but the confidence interval worsens. The reason is not immediately apparent, though it could be that the original model was intended for long-distance wakes and therefore produced consistent but poor results in the near wake, which would explain the poor performance.

Across all cases, the ANN-based surrogate outperforms the engineering models. Its superior performance is attributed to its ability to represent complex, spatially varying flow structures without relying on strong simplifying assumptions about wake behavior. This expressiveness enables the ANN to capture near-wake features with substantially higher fidelity, resulting in significantly lower errors than traditional engineering approaches.

In Fig. 7, the computational cost and memory consumption are plotted against the number of wind turbines. Figure 7a and b both show the computational cost, but in Fig. 7a the cost is expressed in CPU hours (CPUh), meaning that only models executed on the CPU can be compared directly. CPU hours are calculated by multiplying the wall time (in hours) by the number of CPU cores used during the simulation, allowing for a fair comparison. The nodes used have 32 cores. The engineering models can be run on a single core, unlike the ANN, which by default uses all available cores. Figure 7b, on the other hand, reports the real wall time in seconds, enabling direct comparison with the GPU-accelerated ANN. All simulations have been conducted on identical and

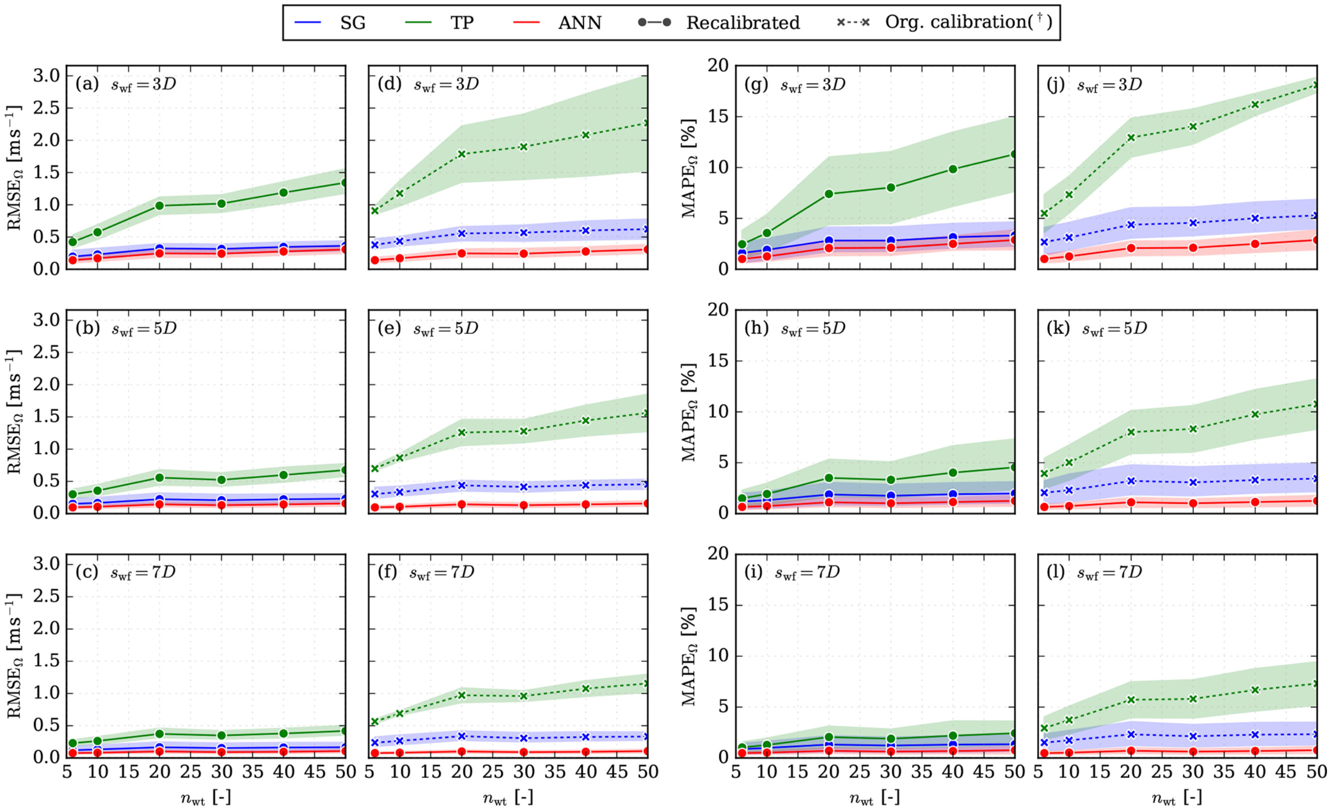

Figure 6. Flow study result metrics for an increasing number of wind turbines, without a blockage model. **(a–c)** RMSE_Ω with recalibrated wake models, **(d–f)** RMSE_Ω for the wake models with original parameters, **(g–i)** MAPE_Ω with recalibrated wake models, and **(j–l)** MAPE_Ω with wake models using the original parameters.

exclusive nodes on the DTU cluster Sophia (Technical University of Denmark, 2019).

All the results are reported with a 95 % confidence interval, based on an ensemble over the inflow conditions (wind direction, wind speed, and ambient TIs) and separation factors s_{wf}. The confidence is quite high, indicating that the variation in computational resource requirements with respect to layout density and inflow conditions is low. The results are shown on a semi-logarithmic plot, which likely contributes to the appearance of increasing confidence; however, it is also expected to be low.

Figure 7a shows that, in terms of CPUh, the fastest model is the SG model, followed by the TP model, with the ANN being the slowest. The ANN being the slowest is expected, as evaluating it requires multiple matrix multiplications, as described in Eq. (10), whereas the engineering models require far fewer computations. The TP model employs a Gaussian overlap for the rotor-averaging strategy, which requires additional flow evaluation and appears to increase its computational cost; however, this relative difference decreases as the number of turbines increases. In Fig. 7b, we see that the real-time performance of the ANN improves significantly when GPU acceleration is enabled, placing it roughly on par with the TP model. Without GPU acceleration, however, the computational cost of the ANN is too high for applications such as WFLO, where many repeated evaluations would otherwise become prohibitively expensive.

The memory requirements of the considered models are similar, as shown in Fig. 7c, except for the GPU-accelerated ANN, which requires more memory. However, as all models remain below 1 GB of RAM, this remains well within an acceptable range. The more pressing concern is the availability of GPU resources.

3.3 WFLO study

A series of WFLO problems was solved using the optimization algorithms listed in Table 6 for each engineering model, both recalibrated and original, as well as for the ANN model. All optimizations were executed on identical and exclusive nodes of the Sophia cluster (Technical University of Denmark, 2019) to ensure comparable runtime conditions. As PyWake at the time of writing does not support automatic GPU gradients, all optimizations have been performed using CPU resources only. Table 9 summarizes the performance of different combinations of wake models and optimization algorithms. For each algorithm within a given model, the random seed that achieved the highest expected AEP is reported.

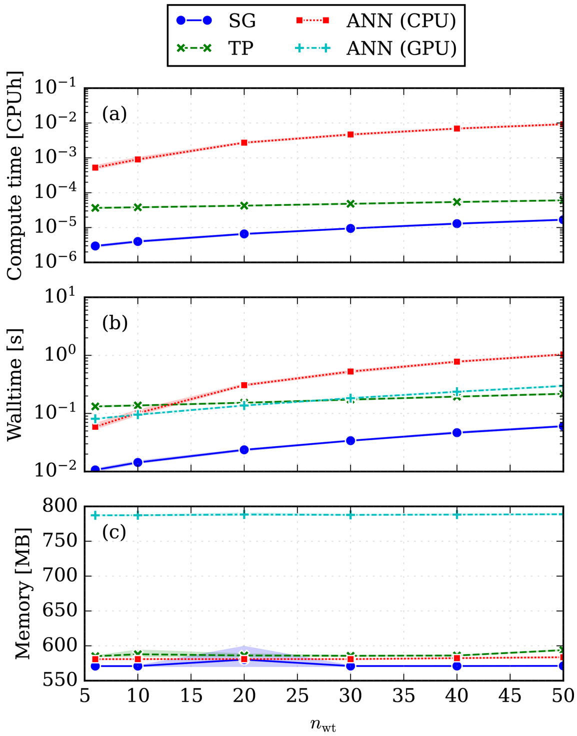

Figure 7. Average computational resource cost of a flow case for increasing wind farm sizes without inclusion of blockage. **(a)** Relative cost in CPU hours, **(b)** real-world cost in wall time (seconds), and **(c)** random access memory (RAM) consumption in MB.

Table 9. AEP of the optimized layouts produced from the best-performing initial seeds. Results shown in bold indicate the optimization algorithm that achieved the highest AEP, selected for RANS validation.

Method	Low-fidelity AEP [GWh]			
	w/o blockage			with blockage
	SLSQP	IPOPT	SGD	SGD
SG	3049	**3050**	3046	**3058**
SG†	**2972**	2966	2954	**2969**
TP	**2974**	2968	2956	**2958**
TP†	**2750**	2747	2721	**2739**
ANN	3041	**3042**	3040	**3047**

† Model with original parameters.

To aid interpretation, the AEP value for the best-performing algorithm in each model is highlighted in bold.

The results in Table 9 are separated into cases with and without blockage. Only the SGD algorithm was used for cases with blockage due to the substantially higher memory consumption of these models. The batch sampling inherent to SGD makes it computationally less expensive, making it the only approach feasible at the required wind-rose resolution. Consequently, SGD results are used throughout for cases with blockage. A visualization of the obtained layouts is available in Appendix B.

For AEP validation with RANS, one optimized layout was selected for each low-fidelity model, both with and without blockage, corresponding to the 10 results marked with bold in Table 9. Table 10 presents the RANS AEP alongside the corresponding low-fidelity AEP and an accuracy estimate, together with key optimization metrics for assessing computational expense. The best-performing metrics in Table 10 are highlighted in bold. The inclusion of blockage did not improve the accuracy of any model, despite its greater physical fidelity; however, it substantially increased the optimization cost by factors of between $\sim 6\times$ and $\sim 20\times$. The ANN was an exception, where the blockage case was cheaper; this could be because the IPOPT algorithm, which succeeded in the case without blockage, more aggressively explores the loss landscape than the SGD, which is known to be resistant to local minima.

The TP model produced the layout with the highest RANS AEP, despite being the least accurate in predicting RANS AEP from its low-fidelity estimate, excluding models using original parameters. This outcome is somewhat unexpected given that the recalibrated models and ANN were tuned against RANS data, making RANS the designated ground-truth reference in this study. By contrast, the SG model reproduced the RANS AEP most accurately, followed closely by the ANN, with TP trailing behind significantly. Hence, the ability to reproduce the RANS AEP accurately was not correlated with the ability to find the highest RANS AEP layout through optimization in this study.

One possible explanation for this behavior is that the issue arises as an artifact of the optimization process. To investigate this, we first examine whether, among the layouts previously studied in RANS, the layout obtained using each low-fidelity model represents the best layout that the model could theoretically produce. To test this hypothesis, a cross-comparison is performed in which the AEP is calculated using each low-fidelity model for all layouts evaluated with RANS. In Fig. 8, the results of this cross-comparison are presented as a heatmap showing the change in AEP (ΔAEP) relative to the default optimized model–layout pair. Entries with positive ΔAEP in a given model row therefore indicate that a better layout exists compared to the one obtained during optimization. As illustrated in Fig. 8, significant improvements in AEP can be achieved for both the SG model and the ANN without blockage, whereas for the TP model, no better layout is available. In cases with blockage, however, there is a single better layout for the TP* model: the one generated by the TP model without blockage. This difference is marginal and not systematic: TP* is the only blockage case for which a better layout exists, whereas, without blockage, better layouts were available for the SG model and the ANN. As all blockage configurations use the same flow model and optimization algorithm, this isolated difference cannot be firmly

Table 10. Comparison of RANS-validated AEP and optimization statistics for wind farm layouts generated using different wake models, evaluated with and without blockage. The best-performing values within each category are highlighted in bold.

Metric	w/o blockage					with blockage				
	SG	SG†	TP	TP†	ANN	SG*	SG†*	TP*	TP†*	ANN*
RANS AEP [GWh]	3083	3090	**3093**	3091	3080	**3092**	3091	3091	3088	3089
Low-fidelity AEP [GWh]	3050	2972	2974	2750	3042	3058	2969	2958	2739	3047
Low-fidelity acc. [% of RANS]	**98.92**	96.18	96.16	88.98	98.75	**98.9**	96.04	95.71	88.7	98.67
Opt. alg.	IPOPT	SLSQP	SLSQP	SLSQP	IPOPT	SGD	SGD	SGD	SGD	SGD
Wall time [h]	4.513	**0.4275**	0.6851	0.5117	159.1	28.29	10.17	13.62	**7.423**	92.54

† Model with original parameters. * Model with blockage effects included.

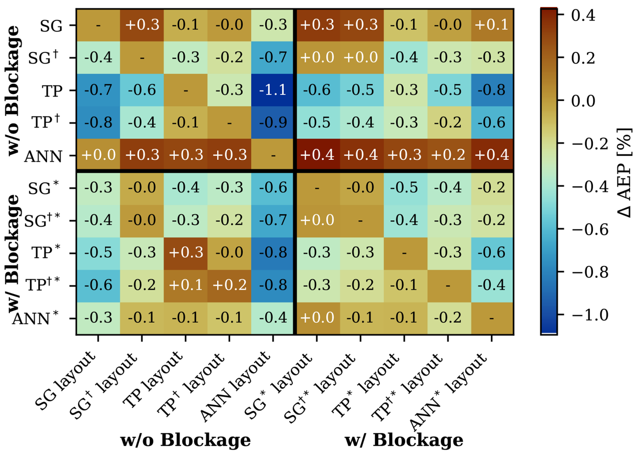

Figure 8. Cross-comparison of ΔAEP for layouts optimized by different low-fidelity models, showing changes relative to each model's default layout. Each row corresponds to a distinct model used to compute AEP. Each column corresponds to a distinct layout generated by optimizing with a selected model.

attributed to any one reason, and it is suggested that further research investigate this.

According to the data in Table 10, a trend appears indicating that the SG model and ANN produce more accurate AEP estimates than the TP model. To determine whether this trend holds more generally, the accuracy of the low-fidelity models is assessed by comparing their AEP estimates with the RANS AEP for all available layouts. This comparison is presented in Fig. 9, where additional aggregated metrics for each row have been included, along with a model podium to easily identify the models most closely matching RANS predictions for each layout. As shown in Fig. 9, the hierarchy of models in terms of AEP estimation accuracy places the TP model in third position. In contrast, the SG model and ANN compete for first place, with the SG model ultimately emerging as the most accurate. Both the SG model and the ANN are more advanced than the TP model. The SG model incorporates a transition from a top-hat shape in the near wake to a Gaussian profile in the far wake, addressing limitations of a fully Gaussian wake. In contrast, the ANN is substantially more expressive than the engineering models, with more than 28 000 tunable parameters compared to fewer than 10. While this analysis highlights differences in AEP estimation accuracy, the WFLO results reveal that accuracy alone does not guarantee optimal layouts. In fact, the recalibrated TP model produced the best layouts in our WFLO experiment, despite demonstrating the lowest accuracy among the recalibrated models.

The TP model achieved strong RANS AEP-validated performance despite having lower AEP accuracy than the SG and ANN models. This was the first indication we observed that AEP accuracy, while important, is not the only critical factor in achieving an optimal layout. Further investigation across all RANS-validated layouts confirmed this pattern: the SG and ANN models consistently showed higher AEP accuracy, yet they did not yield the best layouts during WFLO. These observations are based on experiments with a limited number of initial random seeds, so while the results indicate that the TP model performs better, further experiments would strengthen this conclusion. That said, since computational resources are inherently limited in WFLO applications, this represents a realistic scenario for ANN-based optimization, and the TP model demonstrates better performance even under these practical constraints.

This insight raises two questions: why might a less accurate model be advantageous during optimization, and is this behavior systematic? Although a definitive answer remains elusive, we hypothesize that increased model fidelity in wake representations induces greater multimodality in the objective function geometry. As turbine–turbine interactions and the resulting flow become more complex, the relationship between turbine placement and AEP becomes correspondingly more complex. This complexity amplifies both the number and the sharpness of local extrema, complicating the optimization problem and increasing the likelihood that gradient-based methods become trapped in local minima.

Multimodality has been examined in a recent paper by Poole (2025), which investigated its prevalence in circular farms and applied a smoothing strategy proposed by Thomas et al. (2017). Although limited to relatively small and sim-

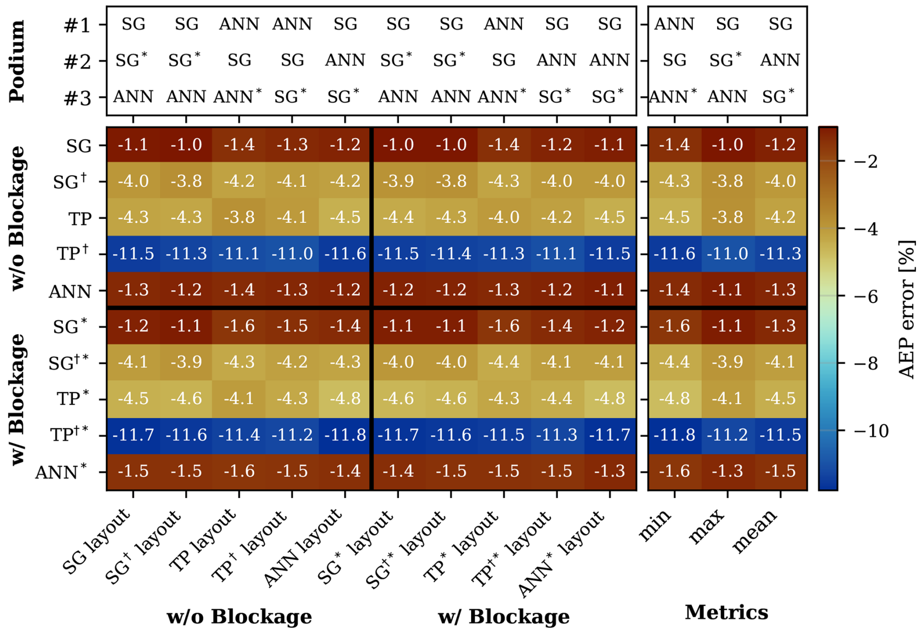

Figure 9. Comparison of AEP estimates from low-fidelity models against RANS for all layouts. Each row corresponds to a distinct model used to compute AEP. Each column corresponds to a distinct layout, which are the same layouts considered in Fig. 8. The colors show the relative difference between the RANS-computed AEP and model-computed AEP associated with each layout. Podium rows indicate the top three models that most closely matched RANS predictions for each layout, while additional columns report aggregated metrics.

ple farms, the study establishes a correlation between farm size and multimodality. Simultaneously, they demonstrated that the mitigation strategy could reduce the modality of the objective function's geometry.

Earlier work by Stanley and Ning (2019) suggested reparameterizing wind farm layouts using the boundary-grid (BG) approach, which places turbines along the farm boundary. This assumption does not generally hold, as it ignores electrical infrastructure during optimization. Nonetheless, reducing multimodality by reducing the number of design variables remains a sound approach, as it can preserve AEP accuracy while making local minima easier to avoid.

An alternative approach, aligned with our hypothesis that complex wake modeling increases multimodality, is to simplify flow models. Thomas and Ning (2018) developed a wake model with an adaptable spread, enabling artificially smeared wakes during early optimization. Smearing of wakes reduces locally unwaked areas and lowers the number of local minima. As convergence nears, the wake spread is gradually removed, allowing final optimization with the original wake model. Another method is the FLOWERS analytical AEP model by LoCascio et al. (2024), which further simplifies classical assumptions by considering a single-turbine operating condition, a constant wake expansion, a unified wind speed per directional bin, and a Fourier series approximation of wind directions. These simplifications enable an analytical AEP expression and significantly reduce the degree of multimodality.

To investigate this hypothesis, an experiment to visualize the multimodality of the objective function was conducted. Unfortunately, the admittedly limited experiment only showed a slight indication that the TP model produces lower multimodality. Therefore, the experiment has been documented but moved to Appendix D as further studies are required to reach a definitive conclusion on the matter.

4 Conclusions

This study systematically validated an ANN-based surrogate model against two representative engineering models for wind farm flow simulation and WFLO. The main findings can be summarized as follows:

1. *Recalibration of engineering models.* Bayesian optimization using RANS-based single-wake data significantly improved both engineering models, most notably in the near-wake region, reducing RMSE and MAPE relative to the original configurations.

2. *Flow simulation performance.* Across all tested layouts and inflow conditions, the ANN surrogate consistently outperformed the engineering models in $RMSE_{\Omega}$ and $MAPE_{\Omega}$, owing to its ability to represent complex spatial flow structures. This higher fidelity comes at a greater computational cost, though GPU acceleration yields an $\sim 8\times$ speed-up for larger farms.

3. *WFLO results and optimization behavior.* Despite lower AEP prediction accuracy, the recalibrated TP model produced the layouts with the highest RANS-validated AEP, whereas the more accurate SG model and ANN surrogate did not yield the best-performing layouts. This suggests that higher model fidelity may add complexity to the optimization landscape, hindering convergence of gradient-based algorithms, so that simpler models can occasionally yield layouts of higher realized AEP. This effect may be mitigated by considering more initial seeds, but given the cost of running the ANN model, this will likely be too costly in the current configuration. Including blockage effects did not improve AEP prediction accuracy or layout performance and substantially increased computational cost.

Further research should investigate the multimodality of the WFLO problem more thoroughly, particularly when using high-fidelity surrogates, including formally characterizing the convexity of the optimization landscapes associated with different wake models and conducting sensitivity analyses of optimal layout AEP with respect to wake model parameters.

Minimizing the impact of multimodality could be achieved by focusing on multifidelity by employing simplified wake models in early iterations to reduce the impact initially, followed by high-fidelity surrogates for final convergence. An investigation into whether ANN-based optimization can improve upon layouts generated by simpler models, such as the TP model, could further provide insight into the practical benefits of high-fidelity surrogates.

The engineering model configurations explored in this work represent only a small subset of the combinations available in the literature, and further expanding this scope would strengthen the study, including considering additional wake models, superposition methods, blockage models, added-TI models, rotor-averaging strategies, etc.

Finally, improving scalability with trust-region methods and extending validation to larger farms and broader inflow conditions will strengthen the applicability of surrogate-based optimization in real-world settings.

Appendix A: Flow study with blockage

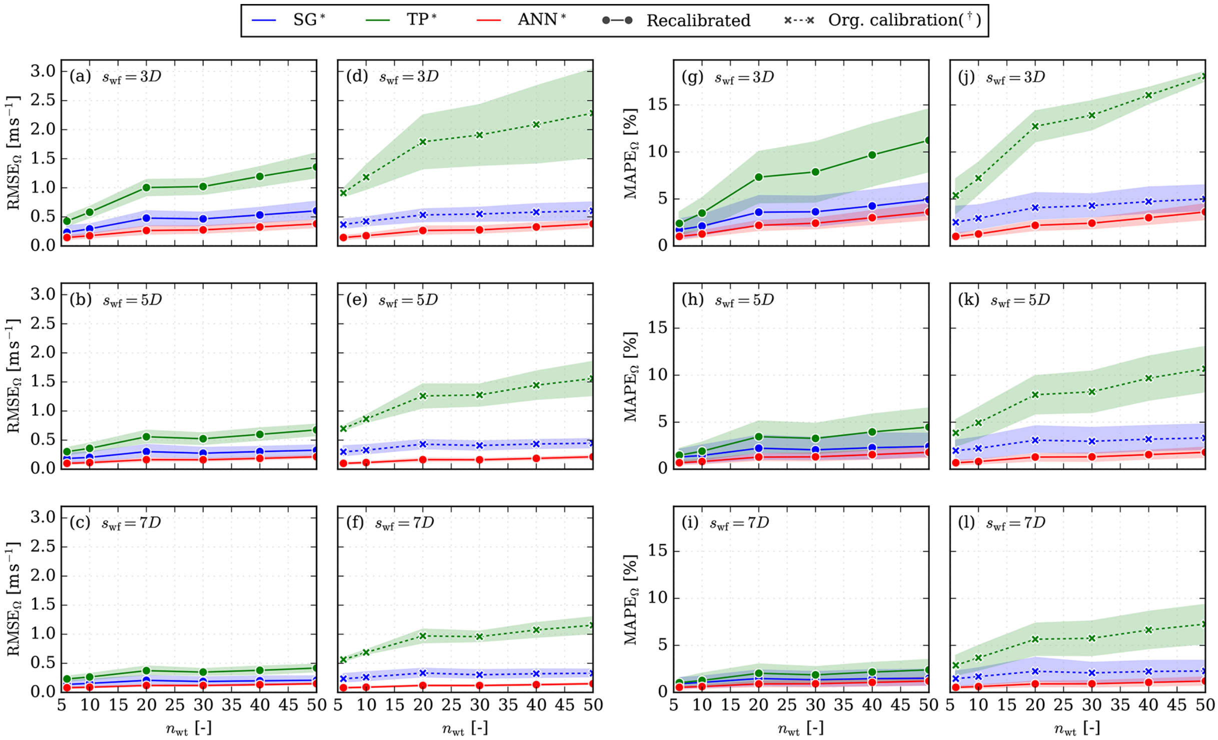

Figure A1. Flow study result metrics for an increasing number of wind turbines, with a blockage model. **(a–c)** RMSE with calibrated wake models, **(d–f)** RMSE with uncalibrated wake models, **(g–i)** MAPE with calibrated wake models, and **(j–l)** MAPE with uncalibrated wake models.

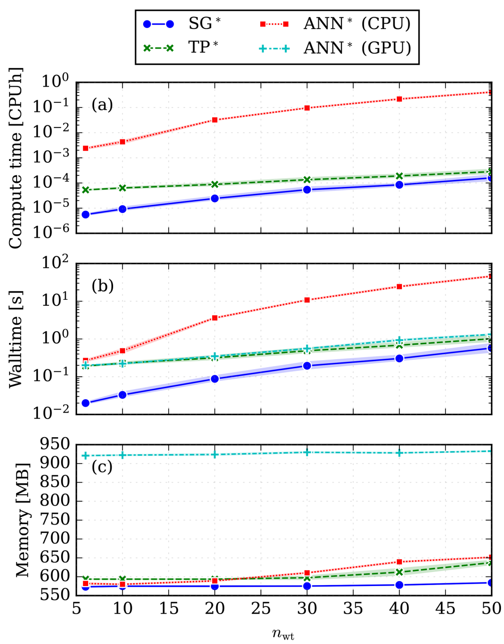

Figure A2. Average computational resource cost of a flow case for increasing wind farm sizes with inclusion of blockage. **(a)** Relative cost in CPU hours, **(b)** real-world cost in wall time (seconds), and **(c)** random access memory (RAM) consumption in MB.

Appendix B: Layouts obtained during WFLO

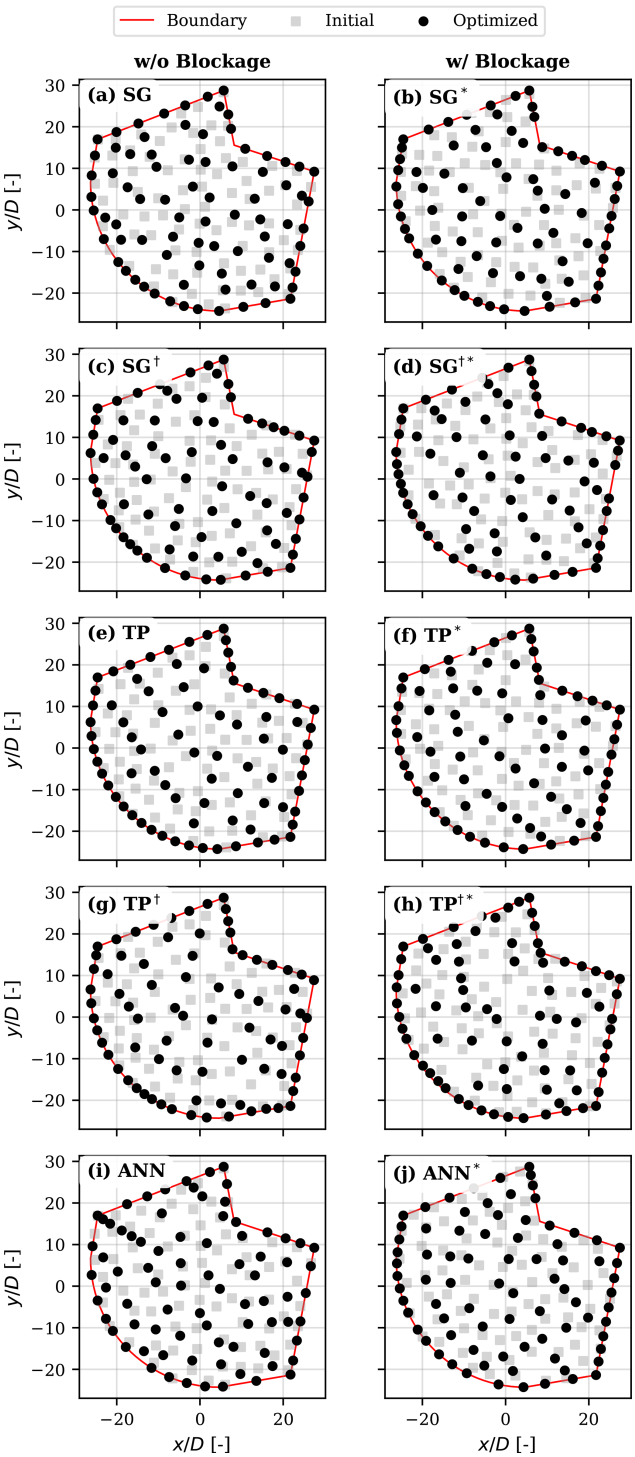

Figure B1. Initial and final layouts obtained through WFLO.

Appendix C: Flow study without blockage displayed on log scale

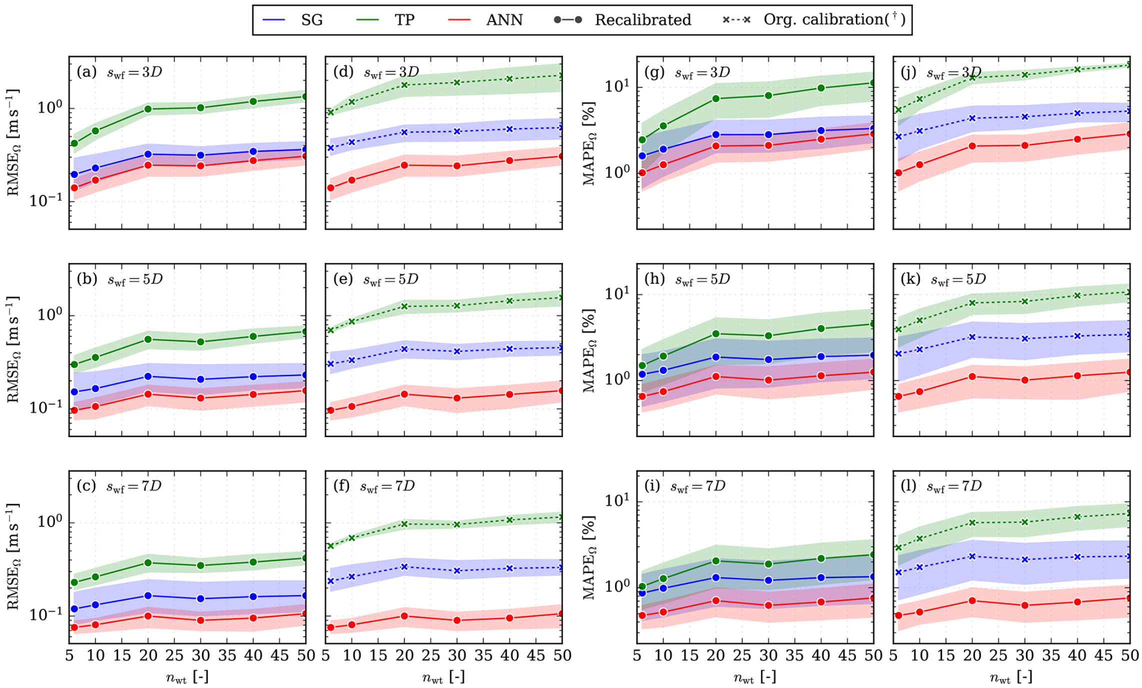

Figure C1. Flow study result metrics for an increasing number of wind turbines without a blockage model, displayed on a semi-logarithmic axis. Containing the same data as Fig. 6.

Appendix D: Multimodality experiment

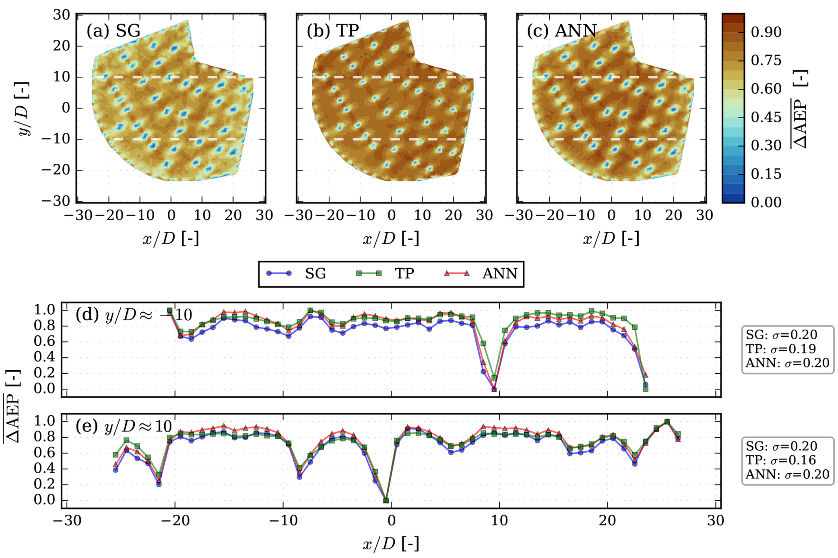

Figure D1. Objective function maps illustrating the normalized AEP improvement $\overline{\Delta\mathrm{AEP}}$ for the calibrated low-fidelity models without blockage: **(a)** SG model, **(b)** TP model, and **(c)** ANN model with a cross-plane profile for all models **(d)** at $y/D \approx -10$ and **(e)** at $y/D \approx 10$.

We investigated the hypothesis that the improved RANS AEP obtained using the TP model correlates with a lower degree of multimodality in the objective function. To do so, the SG, TP, and ANN models were employed to evaluate the improvement in AEP (ΔAEP) that could be achieved by adding a turbine to the best-performing layout obtained with the TP model by systematically assessing the addition of the turbine at different locations throughout the farm. In practice, a grid was constructed with a density of one grid point per D, and the AEP improvement (ΔAEP) was then evaluated for every possible location the additional turbine could be placed in the grid.

Because the ranges vary significantly, which obscures the perceived smoothness of the objective function map, ΔAEP is min–max normalized independently for each observation, placing all the results within a range of 0–1:

$$\overline{\Delta\mathrm{AEP}}(x, y) = \frac{\Delta\mathrm{AEP}(x, y) - \min_{x,y}(\Delta\mathrm{AEP})}{\max_{x,y}(\Delta\mathrm{AEP})}. \tag{D1}$$

The resulting objective function maps, illustrating $\overline{\Delta\mathrm{AEP}}$, are shown in Fig. D1. Additionally, two extra subplots are included, showing $\overline{\Delta\mathrm{AEP}}$ at $y \approx \pm 10D$ for all three models in one plot at a time.

Figure D1a–c illustrate that there is slightly less variability in the range of $\overline{\Delta\mathrm{AEP}}$ for the TP model compared to the other two models. In Fig. D1d and e, the cross-plane $\overline{\Delta\mathrm{AEP}}$ is visualized. Here, the differences between the models are subtle: the TP model shows marginally smoother variations, with fewer extreme peaks and valleys, compared to SG and ANN, but the overall patterns remain broadly similar. As this can be difficult to see visually, SD (σ) values are displayed in Fig. D1d and e to quantify the variability in each cross-section. Lower values of σ indicate smoother objective function landscapes. Here, the TP model consistently shows the lowest σ, supporting the observation of reduced multimodality.

Code and data availability. The code associated with this publication is publicly available at https://gitlab.windenergy.dtu.dk/surrogate-validation-study (last access: 3 August 2026). Because PyWakeEllipsys requires access to the closed-source Ellipsys3D solver, the flow data generated with it have been archived separately at https://doi.org/10.5281/zenodo.18305003. The RANS single-wake database used to calibrate the engineering wake models has been published previously and is available at https://gitlab.windenergy.dtu.dk/TOPFARM/pywake_ranslut/-/tree/v0.3 (last access: 3 August 2026).

Author contributions. JPS, ES, and PER conceived the research; ES programmed the ANN model to the PyWake interface and ran the PyWake simulations and TopFarm optimizations; MPvdL simulated the RANS data; JQ performed the engineering model recalibration; JPS created layouts, performed post-processing, carried out plotting, and analyzed the results; JPS and PER supervised the work; JPS wrote the original draft; and JPS, ES, JQ, MPvdL, and PER reviewed the draft.

Competing interests. The contact author has declared that none of the authors has any competing interests.

Disclaimer. Publisher's note: Copernicus Publications remains neutral with regard to jurisdictional claims made in the text, published maps, institutional affiliations, or any other geographical representation in this paper. The authors bear the ultimate responsibility for providing appropriate place names. Views expressed in the text are those of the authors and do not necessarily reflect the views of the publisher.

Acknowledgements. In preparing this paper and the associated code, the following AI tools were used for proofreading, code development, and auto-completion: ChatGPT, Claude, Grammarly, Claude Code, and GitHub co-pilot. Model training was performed on the Sophia cluster (Technical University of Denmark, 2019).

Financial support. This work is partially funded by the Joint Assessment of Models for Wind Energy project (Energy Technology Development and Demonstration Programme, project number 134242-521856).

Review statement. This paper was edited by Xiaolei Yang and reviewed by two anonymous referees.

References

Ainslie, J.: Calculating the flowfield in the wake of wind turbines, J. Wind Eng. Ind. Aerod., 27, 213–224, https://doi.org/10.1016/0167-6105(88)90037-2, 1988.

Anagnostopoulos, S. and Piggott, M.: Offshore wind farm wake modelling using deep feed forward neural networks for active yaw control and layout optimisation, J. Phys. Conf. Ser., 2151, 012011, https://doi.org/10.1088/1742-6596/2151/1/012011, 2022.

Andersen, S. J. and Murcia Leon, J. P.: Predictive and stochastic reduced-order modeling of wind turbine wake dynamics, Wind Energ. Sci., 7, 2117–2133, https://doi.org/10.5194/wes-7-2117-2022, 2022.

Andersen, S. J., Sørensen, J. N., Ivanell, S., and Mikkelsen, R. F.: Comparison of engineering wake models with CFD simulations, J. Phys. Conf. Ser., 524, 012161, https://doi.org/10.1088/1742-6596/524/1/012161, 2014.

Bak, C., Zahle, F., Bitsche, R., Kim, T., Yde, A., Henriksen, L. C., Hansen, M. H., Blasques, J. P. A. A., Gaunaa, M., and Natarajan, A.: The DTU 10-MW reference wind turbine, in: Danish Wind Power Research, https://orbit.dtu.dk/en/publications/the-dtu-10-mw-reference-wind-turbine/ (last access: 3 August 2026), 2013.

Baker, N. F., Stanley, A. P., Thomas, J. J., Ning, A., and Dykes, K.: Best practices for wake model and optimization algorithm selection in wind farm layout optimization, in: AIAA Scitech 2019 Forum, American Institute of Aeronautics and Astronautics, https://doi.org/10.2514/6.2019-0540, 2019.

Bastankhah, M. and Porté-Agel, F.: A new analytical model for wind-turbine wakes, Renew. Energ., 70, 116–123, https://doi.org/10.1016/j.renene.2014.01.002, 2014.

Bastankhah, M. and Porté-Agel, F.: Experimental and theoretical study of wind turbine wakes in yawed conditions, J. Fluid Mech., 806, 506–541, https://doi.org/10.1017/jfm.2016.595, 2016.

Bleeg, J., Purcell, M., Ruisi, R., and Traiger, E.: Wind farm blockage and the consequences of neglecting its impact on energy production, Energies, 11, 1609, https://doi.org/10.3390/en11061609, 2018.

Blondel, F. and Cathelain, M.: An alternative form of the super-Gaussian wind turbine wake model, Wind Energ. Sci., 5, 1225–1236, https://doi.org/10.5194/wes-5-1225-2020, 2020.

Bodini, N., Moriarty, P., Thedin, R., Doubrawa, P., Archer, C., Blaylock, M., Bottasso, C., Carmo, B., Cheung, L., Dubreuil, C., Floors, R., Herges, T., Houck, D., Kanjari, A., Kaul, C. M., Kelley, C., LI, R., Lundquist, J. K., Major, D., Nguyen, A. K., Optis, M., Parada, L. R. C., Peña, A., Quick, J., Ricarte, D., Radünz, W. C., Rai, R. K., Garcia Santiago, O., Schulte, J., Seim, K. S., van der Laan, M. P., Vimalakanthan, K., and Wise, A.: The AWAKEN wind farm benchmark, Part 2: Modeling results, Wind Energ. Sci. Discuss. [preprint], https://doi.org/10.5194/wes-2026-34, in review, 2026.

Bortolotti, P., Bay, C., Barter, G., Gaertner, E., Dykes, K., McWilliam, M., Friis-Moller, M., Pedersen, M. M., and Zahle, F.: System Modeling Frameworks for Wind Turbines and Plants: Review and Requirements Specifications, Tech. rep., National Renewable Energy Laboratory (NREL), https://doi.org/10.2172/1868328, 2022.

Crespo, A. and Hernández, J.: Turbulence characteristics in wind-turbine wakes, J. Wind Eng. Ind. Aerod., 61, 71–85, 1996.

Criado Risco, J., van der Laan, M. P., Pedersen, M. M., Meyer Forsting, A., and Réthoré, P.-E.: A RANS-based surrogate model for simulating wind turbine interaction, J. Phys. Conf. Ser., 2505, 012016, https://doi.org/10.1088/1742-6596/2505/1/012016, 2023.

Dar, A. S., Berg, J., Troldborg, N., and Patton, E. G.: On the self-similarity of wind turbine wakes in a complex terrain using large eddy simulation, Wind Energ. Sci., 4, 633–644, https://doi.org/10.5194/wes-4-633-2019, 2019.

Dong, G., Qin, J., Wu, C., Xu, C., and Yang, X.: Reinforcement learning-enhanced genetic algorithm for wind farm layout optimization, Renew. Energ., 259, 125093, https://doi.org/10.1016/j.renene.2025.125093, 2026.

DTU Wind and Energy Systems: PyWakeEllipSys v5.3, https://topfarm.pages.windenergy.dtu.dk/cuttingedge/pywake/pywake_ellipsys (last access: 27 May 2025), 2023.

Forsting, A. R. M., Diaz, G. P. N., Segalini, A., Andersen, S. J., and Ivanell, S.: On the accuracy of predicting wind-farm blockage, Renew. Energ., 214, 114–129, https://doi.org/10.1016/j.renene.2023.05.129, 2023.

Frandsen, S.: Turbulence and turbulence-generated structural loading in wind turbine clusters – Risoe-R No. 1188(EN), Tech. rep., Forskningscenter Risoe, 2005.

Gafoor, A., Boya, S. K., Jinka, R., Gupta, A., Tyagi, A., Sarkar, S., and Subramani, D. N.: A physics-informed neural network for turbulent wake simulations behind wind turbines, Phys. Fluids, 37, https://doi.org/10.1063/5.0245113, 2025.

Gunn, K., Stock-Williams, C., Burke, M., Willden, R., Vogel, C., Hunter, W., Stallard, T., Robinson, N., and Schmidt, S. R.: Limitations to the validity of single wake superposition in wind farm yield assessment, J. Phys. Conf. Ser., 749, 012003, https://doi.org/10.1088/1742-6596/749/1/012003, 2016.

Herbert-Acero, J., Probst, O., Réthoré, P.-E., Larsen, G., and Castillo-Villar, K.: A review of methodological approaches for the design and optimization of wind farms, Energies, 7, 6930–7016, https://doi.org/10.3390/en7116930, 2014.

IEC: IEC 61400-1 ed. 3 Wind turbines-Part 1: Design requirements, International Standard, https://standards.iteh.ai/catalog/standards/sist/c065aa89-0f3c-4c31-8817-6bdec19e47eb/iec- (last access: 3 August 2026), 2010.

Jensen, N. O.: A note on wind generator interaction – RISØ-M-2411, Tech. rep., Risø National Laboratories, 1983.

Kainz, S., Quick, J., de Alencar, M. S., Moreno, S. S. P., Dykes, K., Bay, C., Zaaijer, M. B., and Bortolotti, P.: The IEA Wind 740-10-MW Reference Offshore Wind Plants, Tech. rep., IEA Task 55, https://github.com/IEAWindTask37/IEA-Wind-740-10-ROWP (last access: 3 August 2026), 2024.

Klemmer, K. S. and Howland, M. F.: Wake turbulence modeling in stratified atmospheric flows using a novel *k*-*l* model, J. Renew. Sustain. Ener., 17, https://doi.org/10.1063/5.0249278, 2025.

Kraft, D.: A software package for sequential quadratic programming. DFVLR-FB 88-28, Tech. rep., Institut für Dynamik der Flugsysteme, Deutsche Forschungs- und Versuchsanstalt für Luft- und Raumfahrt (DFVLR), 1988.

Lanzilao, L. and Meyers, J.: A new wake-merging method for wind-farm power prediction in the presence of heterogeneous background velocity fields, Wind Energy, 25, 237–259, https://doi.org/10.1002/we.2669, 2022.

Li, B., Ge, M., Li, X., and Liu, Y.: A physics-guided machine learning framework for real-time dynamic wake prediction of wind turbines, Phys. Fluids, 36, https://doi.org/10.1063/5.0194764, 2024a.

Li, H., Yang, Q., and Li, T.: Wind turbine wake prediction modelling based on transformer-mixed conditional generative adversarial network, Energy, 291, 130403, https://doi.org/10.1016/j.energy.2024.130403, 2024b.

Liu, S., Li, Q., Lu, B., and He, J.: Prediction of offshore wind turbine wake and output power using large eddy simulation and convolutional neural network, Energ. Convers. Manage., 324, 119326, https://doi.org/10.1016/j.enconman.2024.119326, 2025.

Liu, X., Li, Z., and Yang, X.: PhyWakeNet: a dynamic wake model accounting for aerodynamic force oscillations, Wind Energ. Sci., 11, 771–793, https://doi.org/10.5194/wes-11-771-2026, 2026.

LoCascio, M. J., Bay, C. J., Martínez Tossas, L. A., Bastankhah, M., and Gorlé, C.: FLOWERS AEP: an analytical model for wind farm layout optimization, Wind Energy, 27, 1563–1580, https://doi.org/10.1002/we.2954, 2024.

Madsen, H. A., Larsen, T. J., Pirrung, G. R., Li, A., and Zahle, F.: Implementation of the blade element momentum model on a polar grid and its aeroelastic load impact, Wind Energ. Sci., 5, 1–27, https://doi.org/10.5194/wes-5-1-2020, 2020.

Martínez-Tossas, L. A., Annoni, J., Fleming, P. A., and Churchfield, M. J.: The aerodynamics of the curled wake: a simplified model in view of flow control, Wind Energ. Sci., 4, 127–138, https://doi.org/10.5194/wes-4-127-2019, 2019.

Michelsen, J. A.: Basis3D – a platform for development of multiblock PDE solvers, Tech. rep., AFM 92-05, DTU, 1992.

Céspedes Moreno, J. F., Murcia León, J. P., and Andersen, S. J.: Convergence and efficiency of global bases using proper orthogonal decomposition for capturing wind turbine wake aerodynamics, Wind Energ. Sci., 10, 597–611, https://doi.org/10.5194/wes-10-597-2025, 2025.

Mosetti, G., Poloni, C., and Diviacco, B.: Optimization of wind turbine positioning in large windfarms by means of a genetic algorithm, J. Wind Eng. Ind. Aerod., https://doi.org/10.1016/0167-6105(94)90080-9, 1994.

Niayifar, A. and Porté-Agel, F.: Analytical modeling of wind farms: a new approach for power prediction, Energies, 9, https://doi.org/10.3390/en9090741, 2016.

Nogueira, F.: Bayesian Optimization: Open source constrained global optimization tool for Python, https://github.com/bayesian-optimization/BayesianOptimization (last access: 3 August 2026), 2014.

Nygaard, N. G., Steen, S. T., Poulsen, L., and Pedersen, J. G.: Modelling cluster wakes and wind farm blockage, J. Phys. Conf. Ser., 1618, https://doi.org/10.1088/1742-6596/1618/6/062072, 2020.

Nygaard, N. G., Pedersen, J. G., Hansen, S. D., and Krastins, P.: A Turbulence Optimized Park model, Tech. rep., Ørsted, https://github.com/OrstedRD/TurbOPark/blob/main/TurbOPark%20description.pdf (last access: 3 August 2026), 2022.

Ott, S., Berg, J., and Nielsen, M.: Linearised CFD Models for Wakes. Danmarks Tekniske Universitet, Risø Nationallaboratoriet for Baeredygtig Energi, Tech. rep., Risø National Laboratory, https://orbit.dtu.dk/en/publications/linearised-cfd-models-for-wakes/ (last access: 3 August 2026), 2011.

Panofsky, H. A. and Dutton, J. A.: Atmospheric turbulence. Models and methods for engineering applications, New York, Wiley, ISBN: 9780471057147 (ISBN 10: 0471057142), 1984.

Pedersen, M. M., Forsting, A. M., van der Laan, P., Riva, R., Romàn, L. A. A., Risco, J. C., Friis-Møller, M., Quick, J., Schøler, J. P., Rodrigues, R. V., Olsen, B. T., and Réthoré, P.-E.: PyWake 2.5.0: An open-source wind farm simulation tool, https://gitlab.windenergy.dtu.dk/TOPFARM/PyWake (last access: 3 August 2026), 2023.

Pedersen, M. M., Friis-Møller, M., Réthoré, P.-E., Simutis, E., Riva, R., Quick, J., Dimitrov, N. K., Rinker, J., and Dykes, K.: DTUWindEnergy/TopFarm2: Release of v2.6.1, Zenodo [code], https://doi.org/10.5281/zenodo.17540961, 2025.

Pish, F., Göçmen, T., and Laan, M. P. V. D.: Generalization of single wake surrogates for multiple and farm-farm wake analysis, J. Phys. Conf. Ser., 2767, 092058, https://doi.org/10.1088/1742-6596/2767/9/092058, 2024.

Pookpunt, S. and Ongsakul, W.: Optimal placement of wind turbines within wind farm using binary particle swarm optimization with time-varying acceleration coefficients, Renew. Energ., 55, 266–276, https://doi.org/10.1016/j.renene.2012.12.005, 2013.

Poole, D. J.: Characterization of multimodality in wind farm layout optimization, Energy Science and Engineering, https://doi.org/10.1002/ese3.70377, 2025.

Porté-Agel, F., Bastankhah, M., and Shamsoddin, S.: Wind-turbine and wind-farm flows: a review, Bound.-Lay. Meteorol., 174, 1–59, https://doi.org/10.1007/s10546-019-00473-0, 2020.

Quick, J., Rethore, P.-E., Mølgaard Pedersen, M., Rodrigues, R. V., and Friis-Møller, M.: Stochastic gradient descent for wind farm optimization, Wind Energ. Sci., 8, 1235–1250, https://doi.org/10.5194/wes-8-1235-2023, 2023.

Schneemann, J., Theuer, F., Rott, A., Dörenkämper, M., and Kühn, M.: Offshore wind farm global blockage measured with scanning lidar, Wind Energ. Sci., 6, 521–538, https://doi.org/10.5194/wes-6-521-2021, 2021.

Schøler, J. P., Riva, R., Andersen, S. J., Leon, J. P. M., van der Laan, M. P., Risco, J. C., and Réthoré, P.-E.: RANS-AD based ANN surrogate model for wind turbine wake deficits, J. Phys. Conf. Ser., 2505, 012022, https://doi.org/10.1088/1742-6596/2505/1/012022, 2023.

Schøler, J. P., Rosi, N., Quick, J., Riva, R., Andersen, S. J., Leon, J. P. M., Laan, M. P. V. D., and Réthoré, P.-E.: RANS wake surrogate: impact of physics information in neural networks, J. Phys. Conf. Ser., 2767, 092033, https://doi.org/10.1088/1742-6596/2767/9/092033, 2024.

Sørensen, N. N.: General purpose flow solver applied to flow over hills, PhD thesis, DTU, https://orbit.dtu.dk/en/publications/general-purpose-flow-solver-applied-to-flow-over-hills/ (last access: 3 August 2026), 1994.

Sørensen, J. N., Nilsson, K., Ivanell, S., Asmuth, H., and Mikkelsen, R. F.: Analytical body forces in numerical actuator disc model of wind turbines, Renew. Energ., 147, 2259–2271, https://doi.org/10.1016/j.renene.2019.09.134, 2020.

Stanley, A. P. J. and Ning, A.: Massive simplification of the wind farm layout optimization problem, Wind Energ. Sci., 4, 663–676, https://doi.org/10.5194/wes-4-663-2019, 2019.

Sun, H. and Yang, H.: Wind farm layout and hub height optimization with a novel wake model, Appl. Energ., 348, 121554, https://doi.org/10.1016/j.apenergy.2023.121554, 2023.

Technical University of Denmark: Sophia HPC Cluster, https://doi.org/10.57940/FAFC-6M81, 2019.

Thomas, J. J. and Ning, A.: A method for reducing multimodality in the wind farm layout optimization problem, J. Phys. Conf. Ser., 1037, 042012, https://doi.org/10.1088/1742-6596/1037/4/042012, 2018.

Thomas, J. J., Gebraad, P. M., and Ning, A.: Improving the FLORIS wind plant model for compatibility with gradient-based optimization, Wind Engineering, 41, 313–329, https://doi.org/10.1177/0309524X17722000, 2017.

Ti, Z., Deng, X. W., and Zhang, M.: Artificial Neural Networks based wake model for power prediction of wind farm, Renew. Energ., 172, 618–631, https://doi.org/10.1016/j.renene.2021.03.030, 2021.

Troldborg, N. and Meyer Forsting, A. R.: A simple model of the wind turbine induction zone derived from numerical simulations, Wind Energy, 20, 2011–2020, https://doi.org/10.1002/we.2137, 2017.

Troldborg, N., Sørensen, N., Réthoré, P.-E., and van der Laan, M.: A consistent method for finite volume discretization of body forces on collocated grids applied to flow through an actuator disk, Comput. Fluids, 119, 197–203, https://doi.org/10.1016/j.compfluid.2015.06.028, 2015.

van der Laan, M., Andersen, S., Kelly, M., and Baungaard, M.: Fluid scaling laws of idealized wind farm simulations, J. Phys. Conf. Ser., 1618, 062018, https://doi.org/10.1088/1742-6596/1618/6/062018, 2020.

van der Laan, M. P., Sørensen, N. N., Réthoré, P. E., Mann, J., Kelly, M. C., and Troldborg, N.: The k-ε-f_p model applied to double wind turbine wakes using different actuator disk force methods, Wind Energy, 18, 2223–2240, https://doi.org/10.1002/we.1816, 2015a.

van der Laan, M. P., Sørensen, N. N., Réthoré, P. E., Mann, J., Kelly, M. C., Troldborg, N., Schepers, J. G., and Machefaux, E.: An improved k-ε model applied to a wind turbine wake in atmospheric turbulence, Wind Energy, 18, 889–907, https://doi.org/10.1002/we.1736, 2015b.

van der Laan, M. P., Andersen, S. J., and Réthoré, P.-E.: Brief communication: Wind-speed-independent actuator disk control for faster annual energy production calculations of wind farms using computational fluid dynamics, Wind Energ. Sci., 4, 645–651, https://doi.org/10.5194/wes-4-645-2019, 2019.

van der Laan, M. P., Andersen, S. J., Réthoré, P.-E., Baungaard, M., Sørensen, J. N., and Troldborg, N.: Faster wind farm AEP calculations with CFD using a generalized wind turbine model, J. Phys. Conf. Ser., 2265, 022030, https://doi.org/10.1088/1742-6596/2265/2/022030, 2022.

van der Laan, M. P., Meyer Forsting, A., and Réthoré, P.-E.: A Computational Fluid Dynamics surrogate model for wind turbine interaction including atmospheric stability, Wind Energ. Sci. Discuss. [preprint], https://doi.org/10.5194/wes-2025-287, in review, 2026.

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S. J., Brett, M., Wilson, J., Millman, K. J., Mayorov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., Carey, C. J., Polat, İ., Feng, Y., Moore, E. W., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E. A., Harris, C. R., Archibald, A. M., Ribeiro, A. H., Pedregosa, F., van Mulbregt, P., and SciPy 1.0 Contributors: SciPy 1.0: fundamental algorithms for scientific computing in Python, Nat. Methods, 17, 261–272, https://doi.org/10.1038/s41592-019-0686-2, 2020.

Wächter, A. and Biegler, L. T.: On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming, Math. Program., 106, 25–57, https://doi.org/10.1007/s10107-004-0559-y, 2006.

Yang, K. and Deng, X.: Layout optimization for renovation of operational offshore wind farm based on machine learning wake model, J. Wind Eng. Ind. Aerod., 232, 105280, https://doi.org/10.1016/j.jweia.2022.105280, 2023.

Yang, S., Deng, X., Ti, Z., Yan, B., and Yang, Q.: Cooperative yaw control of wind farm using a double-layer machine learning framework, Renew. Energ., 193, 519–537, https://doi.org/10.1016/j.renene.2022.04.104, 2022.

Zehtabiyan-Rezaie, N., Iosifidis, A., and Abkar, M.: Data-driven fluid mechanics of wind farms: a review, J. Renew. Sustain. Ener., 14, https://doi.org/10.1063/5.0091980, 2022.

Zhang, J. and Zhao, X.: Spatiotemporal wind field prediction based on physics-informed deep learning and LIDAR measurements, Appl. Energ., 288, 116641, https://doi.org/10.1016/j.apenergy.2021.116641, 2021.

Zong, H. and Porté-Agel, F.: A momentum-conserving wake superposition method for wind farm power prediction, J. Fluid Mech., 889, A8, https://doi.org/10.1017/jfm.2020.77, 2020.