https://doi.org/10.5194/wes-2024-120
Preprint. Discussion started: 8 October 2024


eawe european academy of wind energy
Open Access WIND ENERGY SCIENCE DISCUSSIONS

# Integer programming for optimal yaw control of wind farms

Felix Bestehorn[1,2], Florian Bürgel[1], Christian Kirches[1], Sebastian Stiller[1], and Andreas M. Tillmann[1]
[1]TU Braunschweig, Institute for Mathematical Optimization, Universitätsplatz 2, 38106 Braunschweig, Germany
[2]now with Carl Zeiss SMT GmbH
**Correspondence:** Florian Bürgel (f.buergel@tu-braunschweig.de)

**Abstract.** It is well-known that wakes caused by the wind turbines within a wind farm negatively impact the power generation and mechanical load of downstream turbines. This is already partially considered in the farm layout. Nevertheless, the strong interactions between individual turbines provide further opportunities to mitigate adverse effects during operation, e.g., by repeatedly adjusting axial induction or yaw angles to current wind conditions. We propose a mathematical approach in form of integer programming for globally optimal yaw control (under some mild assumptions). While we prove the wind farm yaw problem to be strongly $\mathcal{NP}$-hard in general, we demonstrate through numerical experiments that our method is efficient in practice and enables optimal yaw control under real-world requirements on control update periods. In particular, the approach remains efficient if turbines are deactivated and scales reasonably well with increasing farm width.

**Keywords.** Wind farm, optimization, yaw-based control, integer programming.

## 1 Introduction

Wind turbines are considered one of the most important electric power plants of the future energy grid, since they can generate electricity cheaply and with low green house gas emissions, see, e.g., Kost et al. (2018). Naturally, they are placed in wind farms at locations with desirable year-round wind conditions (onshore or offshore), to ensure a high energy yield in general. To further increase efficiency, during operation, one exerts a *control* on a turbine, i.e., periodically adjusts parts of it (e.g., nacelle orientation towards the wind, generator torque or blade pitch angles) to positions that are most beneficial with respect to the current wind speed and direction. The conventional control consists of a greedy strategy in which a turbine maximizes its own power output under certain durability considerations, see, e.g., Hau (2013). In a farm, such greedy control can lead to suboptimal total power output of the farm as a whole (as well as to increased maintenance requirements for the turbines) because the turbines influence one another due to their spatial proximity: Each turbine causes a wake, which is characterized by decreased wind speed and increased turbulence, and naturally impacts downstream turbines regarding both power output and mechanical load (wear and tear). The local control of a (single) turbine affects the length and the spatial distribution of its wake. This opens the possibility for a *global* control that incorporates turbine interactions within the entire wind farm. In general, such a wind farm flow control is state of the art, see, e.g., Meyers et al. (2022) for an overview. In this paper we will focus on the optimization of the nacelle orientation towards the wind, i.e., the so-called yaw-based control, to maximize the total power of the wind farm, which is the most important objective, see, e.g., van Wingerden et al. (2020). The effect of yaw angles on wakes is shown in Fig. 1.

https://doi.org/10.5194/wes-2024-120
Preprint. Discussion started: 8 October 2024

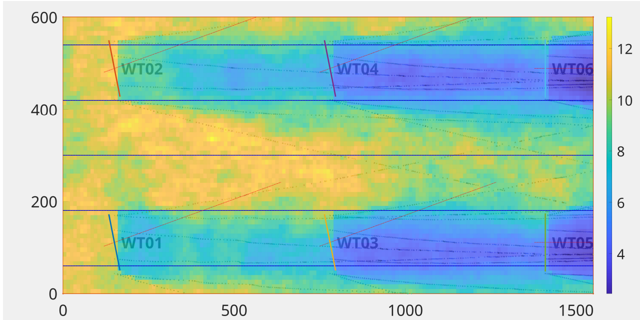




**Figure 1.** Simulation of the local wind speed (in $\mathrm{m\,s^{-1}}$) with 6 turbines (of type NREL 5-MW) arranged in a $2 \times 3$ grid layout; axes give on-ground distances (in $\mathrm{m}$). The wind blows from the left side with a speed of $11\,\mathrm{m\,s^{-1}}$ with a turbulence intensity of $6\%$. It is decreased by the turbines and, additionally, the wakes are deflected by the turbines in the first and second column because of their yaw angle of $15°$. (The downstream-most turbines have a yaw angle of $0°$.) This figure was produced using the software WinFaST, see Sect. 3.1 for a description.

In the following, we discuss the wake models, farm layouts and operations to motivate the wind farm yaw problem and paraphrase our approach along with contributions and limitations; an outline of the remaining paper concludes this introduction.

### 1.1 Related work

Wake models in the literature are often parameterized to NREL 5-MW turbines, see Jonkman et al. (2009) for details, in usual atmospheric conditions for onshore wind farms, see Sect. 3.1 for details. The following brief survey refers to this turbine type; for a detailed overview of the complex topic of wind farm flow control, we refer to, e.g., Meyers et al. (2022). High-fidelity simulations of wind farms in 3D like the *Simulator fOr Wind Farm Application* (SOWFA), see Churchfield et al. (2012a, b); Fleming et al. (2013), are time-consuming and, hence, impractical for use in control. An overview of the most important control-oriented models in 2D is given in (Annoni et al., 2018, Sect. 2.1): first, the Jensen model, see Jensen (1983); Katic et al. (1987) (also for a further developed model); second, the multi-zone model FLORIS, see Gebraad et al. (2014), which has the dynamic extension FLORIDyn, see Gebraad and van Wingerden (2014); and third, the Gaussian model, see, e.g., Bastankhah and Porté-Agel (2016). These models are continuously being developed further. In the present work, we use a simulation software which utilizes a wake model based on FLORIDyn, cf. Sect. 3. If the focus were only on power, i.e., disregarding loads, a static model would suffice. In that case, the state-of-the-art Gauss-Curl-Hybrid model, which incorporates secondary effects of wake steering, see King et al. (2021), is recommended, or the cumulative-curl wake model if there are more than a few wake interactions, see Bay et al. (2023). Both are implemented in the rapidly evolving software FLORIS (not to be confused with the original model), incorporating several steady-state control-oriented wake models, see NREL (2023).

These control-oriented wake models have limitations: For a comparison between FLORIDyn and SOWFA we refer to Gebraad and van Wingerden (2014); in particular, Fig. 5 there illustrates locally strong speed fluctuations in the wind field. In general, wind turbulence depends on a number of meteorological and topographical factors, see (Hau, 2013, Sect. 13.4). However, the power output essentially depends on long-term fluctuations whereas loads are caused by short-term fluctuations,

https://doi.org/10.5194/wes-2024-120
Preprint. Discussion started: 8 October 2024


eawe european academy of wind energy
Open Access WIND ENERGY SCIENCE DISCUSSIONS

see Hau (2013). Furthermore, in the case of specific atmospheric conditions, real-world wakes of offshore wind farms can have a length of $45\,\mathrm{km}$, see Platis et al. (2018), which cannot be predicted by control-oriented models. In any case, the mathematical approach we will propose utilizes a superordinate model in which the underlying wake model is, in principle, interchangeable.

A profitable wind farm needs a suitable location and careful planning of turbine numbers and placement. *Wind farm layout optimization* relies on yearly wind frequency data (wind direction and speed). It has been known for decades that it is useful to avoid wake-induced yield reductions of downstream turbines by setting them far enough apart, see, e.g., Katic et al. (1987). While first attempts to account for such effects merely simulated the annual average output of a specific farm layout, see Katic et al. (1987), in recent years, the layout problem was optimized globally by mixed-integer and constraint programming, see, e.g., Zhang et al. (2014); Fischetti (2017, 2021). In addition to the annual energy output, noise propagation is a concern in case of onshore farms that can also be considered, see Zhang et al. (2014). For offshore farms, aspects of cable routing and jacket foundations are worth taking into account, see Fischetti (2017); Fischetti and Pisinger (2019).

For a given wind farm (i.e., a fixed layout), it is conventional to run greedy control for each individual turbine, see, e.g., Hau (2013). As mentioned earlier, adopting a global control for the whole farm instead of local control of separate turbines can potentially improve the overall power output and relieve physical strain on the turbines. In general, there are two common global control concepts, cf., e.g., Gebraad et al. (2015); Annoni et al. (2016); Meyers et al. (2022): axial induction-based control (of generator torque and/or the collective blade pitch angle) and yaw-based control (of the turbine yaw angles), which is also known as wake steering control, see, e.g., Howland et al. (2019). Both control concepts effectively reduce power generation of upstream turbines by adjusting torque/pitch or yaw, respectively, which in turn leads to increased wind speeds (relative to those under greedy control) in their respective wakes and, consequently, higher power yield of the affected downstream turbines. The main aim of these concepts is to achieve a net gain, and even small improvements are deemed promising, see, e.g., the wake steering study by Howland et al. (2022) with average power increases of $0.3$ to $2.7\%$ for a commercial wind farm.

In general, it depends on the allocation of the turbines, their characteristics, and the wind conditions whether a control different from the greedy one can indeed yield the desired gains. For example, it may happen that wind speeds are so high that all turbines operate at maximal capacity anyway; nevertheless, some control could then still be meaningful to reduce mechanical loads. Furthermore, there are cases in which axial induction control shows no positive effect on total power output while yaw control yields significant improvements, see the high-fidelity computational fluid dynamics simulations in Gebraad et al. (2015). Thus, we will focus on yaw control in this paper, where changing the yaw angle of a turbine deflects its wake.

There is already a lot of research on yaw-based control. We follow the literature distribution in Stanley et al. (2022), which divides it into two parts to tackle the optimization problem, i.e., using continuous yaw angles between lower and upper bounds, see Gebraad et al. (2014); Fleming et al. (2016); Gebraad et al. (2017), and using discretized yaw angles, see Dar et al. (2017); Dou et al. (2020). In Gebraad et al. (2014) a slow game-theoretic approach is used, which does not necessarily deliver a global optimum as desired. This is also not delivered in Fleming et al. (2016); Gebraad et al. (2017) as their optimization method is based on sequential quadratic programming (SQP). However, a combination of yaw control and farm layout optimization has been considered in both references, which is an interesting application but out of scope of the present paper. In Dar et al.

https://doi.org/10.5194/wes-2024-120
Preprint. Discussion started: 8 October 2024


eawe european academy of wind energy
Open Access WIND ENERGY SCIENCE DISCUSSIONS

(2017), the authors modified the Jensen model (cf. Jensen (1983)) to include the effect of yaw angle adjustments and developed a dynamic programming formulation (DPF) to pass the wind speed downstream from turbine to turbine, which results in a very efficient method for turbines in a single row. However, the nonlinearity of the equation to compute the wind speed for a turbine located in several turbine wakes prevents transferring this concept to a 2D farm layout. Nevertheless, Dar et al. (2017) showed that optimizing each row separately with DPF, i.e., ignoring effects in adjacent rows, yields an accuracy of $1\%$ compared to the global optimum (obtained by full enumeration). Our goal is to determine the global optimum—under some mild assumptions, see Sect. 1.2.1 for details—without resorting to full enumeration; for this, our superordinate model takes dependencies of adjacent rows into account and can integrate any kind of wake effect simulation. In Dou et al. (2020), the covariance matrix adaptation evolution strategy is employed, which is a heuristic algorithm for black-box functions. In contrast, our focus is to exploit the structure of the optimization problem. In Stanley et al. (2022), the structure of the problem is used in the new so-called Boolean approach. This considers which wind turbines have downstream ones in their wake, starts at the upstream-most turbine, and fixes a yaw angle to either $0°$ or $20°$ if it increases the power of the farm; the simulations used the software FLORIS (NREL (2023)) with the Gauss-Curl-Hybrid wake model (cf. King et al. (2021)). This greedy approach is efficient, but generally does not result in a globally optimal solution as it fixes angles sequentially and only allows two yaw angles (although $20°$ was selected thoughtfully). In contrast, our approach uses a finer discretization and optimizes yaw angle settings simultaneously for the whole farm. (From the view of blade load, which we do not take into account, a slightly positive yaw offset is best, whereas the exact location depends on level of wind shear, see the field study Ennis et al. (2018).)

Another general idea is to divide the farm into partitions, see Siniscalchi-Minna et al. (2020). This idea goes back to Spudić and Baotić (2013), which tackles distributed systems. We will also consider subsets of turbines but in contrast we allow that a turbine is in several subsets, i.e., the subsets are coupled to exploit their structural connection. In fact, we will consider the wind farm as a network. This general point of view has already been used in Annoni et al. (2019) for a different application, namely to share information among nearby turbines to improve wind direction estimation; the underlying method, which allows simultaneous clustering and optimization on graphs, was developed in Hallac et al. (2015).

### 1.2 Contributions and limitations

We provide a method of globally optimal yaw control (under some mild assumptions) that also includes the possibility to deactivate wind turbines, e.g., for maintenance reasons. To that end, we propose a novel superordinate model which exploits the coupled nature of wind turbines in a wind farm and can be based on any wake model.

We refer to the determination of a globally optimal combination of yaw angles for a given wind farm layout and given wind conditions (i.e., subject to arbitrary but fixed wind speed and direction) as *wind farm yaw problem* (WFYP); see Sect. 2 for details and a mathematical problem definition. In this context, global optimality refers to an objective function, which takes the total power output of the farm into account—our main goal—and can include other quantities representing mechanical loads; in fact, we provide to include the important tower load and the pitch angle changes (causing some wear) as so-called tower activity and pitch activity, see Sect. 3 for a definition; we do not consider the blade load as the used simulation software, described in Sect. 3.1, does not provide a suitable output quantity.

https://doi.org/10.5194/wes-2024-120
Preprint. Discussion started: 8 October 2024

WIND ENERGY SCIENCE DISCUSSIONS

In the following, we paraphrase our problem solving approach after emphasizing what we exploit for this. The complex nonlinearities of wake flow dynamics and turbulence are typically only available through simulation, which makes a direct integration into an optimization model almost impossible. A naive approach to solve the WFYP would amount to simulating the behavior for *every possible combination* of the yaw angles, i.e., full enumeration, but is already impractical for small farms. The crucial observation is that the wake interactions of turbines adhere to certain patterns with respect to the farm layout that occur repeatedly, and with overlaps, across the entire farm. We exploit these redundancies to greatly reduce the number of required combinations: Our superordinate model constructs the farm on the basis of these patterns of depending turbines and ensures the consistency of selected yaw angles in regions of overlapping patterns; we formulate a corresponding integer program to receive the desired yaw angles as solution.

Our numerical experiments are intended as *proof of concept* as we use error-free simulation data. They will show that state-of-the-art solver software for this problem class—e.g., Gurobi, see Gurobi Optimization, LLC (2022), or SCIP, see Bestuzheva et al. (2021)—can solve these WFYP problems within reasonable time, demonstrating the practicality of our approach.

The use of our superordinate model enables deactivating any turbines and a large scaling of the wind farm size orthogonal to the wind direction, whereas a scaling in wind direction significantly increases the computational effort due to a stronger growth of relevant patterns—while beyond the scope of this paper, our model could, in principle, be extended to mitigate the effort required for scaling in wind direction. Moreover, the weighting flexibility of the objective function terms (power output and mechanical loads) allows additional objectives to be considered, e.g., putting selected turbines in a low-load operating mode.

#### 1.2.1 Assumptions

We consider the setting in which all turbines in the wind farm are of the same type and arranged on an underlying irregular grid. These assumptions are not particularly restrictive in practice since, on the one hand, turbines within one farm are typically of the same type—although layout optimization can result in different turbine heights, see, e.g., Stanley et al. (2017)—and on the other hand, we can, in principle, choose the grid resolution as fine as needed to allow representing any layout (by leaving some grid points unused); e.g., the results of layout optimization in Thomas et al. (2015); Gebraad et al. (2017) are not arranged on a simple grid. However, an irregular grid with, e.g., three and five rotor diameters distance between the turbines, respectively, is quite common, see, e.g., Gebraad et al. (2014); Gebraad and van Wingerden (2014); Boersma et al. (2018a, b).

Moreover, our model of the WFYP problem relies on two operational assumptions, which can also be realized with arbitrary fine resolution and should therefore not be restrictive in applications: The admissible yaw angles are bounded—to prevent overly strong mechanical loads, cf., e.g., Boersma et al. (2019)—as well as discretized, and we impose a threshold below which the influence of wakes on downstream turbines is deemed negligible.

In particular, Fleming et al. (2016) limit the yaw offset to $[-45°, +45°]$ and $[-25°, +25°]$, Boersma et al. (2019) to $[-25°, +25°]$, Stanley et al. (2022) to $[0°, 30°]$, and our industry partner suggests $[-15°, +15°]$ to protect the turbines. In our computational experiments, e.g., we choose yaw angles from $[-15°, +15°]$ at $5°$ increments and set the downstream-most turbines to $0°$ (to reduce the number of options, see Sect. 2.1.2)—we always state yaw angles relative to the (fixed) wind direction, i.e., as yaw offsets with respect to the mathematically positive sense of rotation—and disregard wake influence if

https://doi.org/10.5194/wes-2024-120
Preprint. Discussion started: 8 October 2024


the wake-induced wind speed reduction (relative to the given speed) at the downstream turbine is at most $5\%$. We will also refer to this exemplary setup for illustrative purposes when we formally define the WFYP and our solution approach in Sect. 2. Nevertheless, our method admits arbitrary other settings, e.g., discretized yaw angles for the downstream-most turbines.

Our choice is not unrealistic: Quick et al. (2020) describe the problem of uncertainty of incident wind conditions for metrological reasons and for real-world causes; in fact, the inflow of a wind farm can consist of several wind directions, speeds, turbulence intensities and shears (e.g., caused by a mountain). Stanley et al. (2022) deduce that it is unrealistic to solve the wind farm yaw problem with continuous or finely discretized yaw angles and choose their Boolean optimization approach only deciding whether a turbine is yawed or not, i.e., set $0°$ or $20°$ (which is a result of power simulations of turbines with a yaw angle discretization of $5°$); they also compared their approach with a common continuous yaw optimization (based on gradients) and mostly achieve the same power improvement. Against this background, our choice of $5°$ increments is reasonable.

We will fix the turbulence intensity (see Sect. 3.1 for a definition) throughout this paper; a deeper investigation is out of scope. For general effects we refer to the study in Talavera and Shu (2017): first, there is a correlation between the increase of turbulence intensity and faster wake recovery (as wind speed recovers faster for turbulent shear flow in comparison to laminar shear flow) and second, turbulent inflow increases the power output of a wind turbine (because of suppressed flow separation).

The correlation between important weather characteristics like temperature, relative humidity, wind speed and wind gusts are investigated in (Vladislavleva et al., 2013, Fig. 2): as expected, wind speed and gusts have a strong positive correlation with the power output while the pressure has a slightly negative correlation. Finally, the spectrum of possible influences is wide. We focus on the most influential factors in this paper, i.e., wind speed and direction, and fix the others like turbulence intensity and air density for simplicity.

#### 1.2.2 Complexity theory point of view and computation time

In fact, while the above-described homogenization of turbines as well as layout structure, and the yaw angle discretization, may seem to simplify the problem, this is not the case from the viewpoint of computational complexity theory: As we will prove in Appendix A, the WFYP is generally $\mathcal{NP}$-hard, which means that an efficient solution algorithm—i.e., one with runtime polynomially bounded by the input size—is highly unlikely to exist, cf. Garey and Johnson (1979). This computational intractability result, along with discretization-related aspects, motivates and justifies tackling the WFYP by integer programming techniques; see, e.g., Schrijver (1986) for a thorough introduction to integer programming.

Thus, the only remaining potentially limiting aspect is the computation time, which naturally increases with growing problem size and complexity. Nevertheless, in general, we can exploit several mitigating facts: First, it is useful to avoid continuous small yaw movements in order to not unduly increase mechanical loads, and second, the yawing rate must be slow (approximately $0.5\,°\,\mathrm{s}^{-1}$) to avoid gyroscopic moments, see (Hau, 2013, Sects. 6.3.1 and 11.3). As a consequence, yaw angles are usually only adjusted at a fairly low rate, e.g., every $15\,\mathrm{min}$, which provides time for solving the WFYP instance. As our integer programming approach is capable of determining globally optimal yaw angle combinations for entire wind farms of considerable size at most under one minute (after required precomputations), e.g., 27 wind turbines in $6\,\mathrm{s}$, it is suitable for real-world application.

https://doi.org/10.5194/wes-2024-120
Preprint. Discussion started: 8 October 2024

eawe european academy of wind energy
Open Access WIND ENERGY SCIENCE DISCUSSIONS

### 1.3 Paper outline

The paper is structured as follows: In Sect. 2, we formulate the wind farm yaw problem (WFYP) mathematically, we motivate and develop our covering approach and the integer program (IP for short). The details on precomputations, i.e., on the simulation and the resulting performance indicators are described in Sect. 3. We explain and discuss the results of our computational experiments in Sect. 4 and finally, conclude the paper in Sect. 5. The theoretical complexity of the WFYP is examined in Appendix A. We abbreviate farm for wind farm and turbine or WT for wind turbine.

## 2 The wind farm yaw problem

Recall that our wind farm yaw problem (WFYP) aims to find a set of yaw angles—choosing one of a set of admissible angles for each turbine—that maximizes the total power output of the farm, optionally along with other quantities, under a given wind scenario, i.e., fixed wind direction and speed. We will formalize this optimization task in the following Sect. 2.1; to that end, we will introduce some notation and an illustrative example that will accompany us throughout the present section. In Sect. 2.2, we develop our covering approach and derive the corresponding integer program in Sect. 2.3.

### 2.1 Notation and basic WFYP formulation

We consider a farm with $n_{\mathrm{WT}}$ turbines, each identified by an index from the set $T := \{1, 2, \ldots, n_{\mathrm{WT}}\}$. Later, in Example 2.2, we will use our ultimate assumption that turbines are located on an irregular grid with certain spacing in both dimensions. We assign to each turbine $i \in T$ a set $\Gamma_i$ of admissible yaw angles, see Sect. 1.2.1, with respective size $n_{\Gamma,i} := |\Gamma_i| < \infty$. For every turbine $i$, we associate an index set $N_{\Gamma,i} := \{1, \ldots, n_{\Gamma,i}\}$ with its admissible yaw angles. This general notation allows for turbines of different types, but even when working with identical ones, for which the yaw angle sets usually coincide, difference may arise, e.g., if maintenance reasons limit the options for specific turbines.

#### 2.1.1 WFYP formulation as integer program with black-box objective

Recalling that the turbines can influence each other, the overall power output of the farm and load-related other quantities depend on the global yaw configuration, i.e., the collection of the set yaw angles of all individual turbines, as well as the considered wind conditions (in particular, direction and speed). Since the precise relation of these aspects has no known analytical form, the objective function of the WFYP must generally be considered a *black-box* whose values for a specific combination of input parameters can be determined, or estimated, by running a simulation of the corresponding farm scenario.

To specify a basic mathematical formulation of the WFYP, we introduce binary decision variables $x_{i,j}$ for all $i \in T$, $j \in N_{\Gamma,i}$; if turbine $i$ is set to the yaw angle (from $\Gamma_i$) indexed by $j$, then $x_{i,j} = 1$, and otherwise $x_{i,j} = 0$. As any turbine can only operate with one yaw angle at a time, these decision variables must adhere to $\sum_{j \in \Gamma_i} x_{i,j} = 1$ for all $i \in T$. The black-box objective can then be described by a function $f_\omega \colon \{0,1\}^{n_{\Gamma,1}+\cdots+n_{\Gamma,n_{\mathrm{WT}}}} \to \mathbb{R}^{n_{\mathrm{WT}}}$, where we omit the dependency on the (here, fixed) wind direction and speed as well as farm layout for notational convenience and where the vector $\omega$ consists of two weighting factors,

https://doi.org/10.5194/wes-2024-120
Preprint. Discussion started: 8 October 2024


eawe european academy of wind energy
Open Access WIND ENERGY SCIENCE DISCUSSIONS

which we will discuss later. This function is comprised of the objective contribution of every turbine, which is impacted by its own yaw angle as well as the yaw configuration of the remaining farm (in fact, not all other turbines influence any given one, but for now, we do not utilize this). In particular, the function $f_\omega(x) = (f_{\omega,1,j_1}(x), \ldots, f_{\omega,n_{\mathrm{WT}},j_{n_{\mathrm{WT}}}}(x))^\top$, where $f_{\omega,i,j}(x)$ denotes the objective contribution of turbine $i$ when set to yaw angle index $j$ from its admissible set $\Gamma_i$ (as per $x_{i,j} = 1$). Here, the yaw configuration of all turbines (in particular those that influence $i$) is fixed as prescribed by the decision variables $x$. To find the optimal yaw configuration of all turbines, i.e., a solution $x \in \{0,1\}^{n_{\Gamma,1}+\cdots+n_{\Gamma,n_{\mathrm{WT}}}}$, our objective function sums up contributions of individual turbines:

$$f_\omega^\Sigma(x) := \sum_{i=1}^{n_{\mathrm{WT}}} \sum_{j=1}^{n_{\Gamma,i}} f_{\omega,i,j}(x)\, x_{i,j} \quad \text{with black-box simulation function} \quad f_{\omega,i,j}(x) := P_{i,j}(x) - \omega^{(\mathrm{T})}\, a_{i,j}^{(\mathrm{T})}(x) - \omega^{(\mathrm{P})}\, a_{i,j}^{(\mathrm{P})}(x),$$

where $P_{i,j}$, $a_{i,j}^{(\mathrm{T})}$ and $a_{i,j}^{(\mathrm{P})}$ are the average power output, the tower activity, and the pitch activity—all over a certain (fixed) period of time—of turbine $i \in T$ with yaw angle index $j \in \Gamma_i$ (and the remaining yaw angles corresponding to the selections encoded in $x$). To evaluate the black-box for a specific $x$, one needs to resort to *simulation* to obtain the power and mechanical load values for each turbine in the considered farm (under the given wind scenario)—the details, including definitions of the notions of tower and pitch activity, will be described in Sect. 3. The (nonnegative) weighting factors $\omega^{(\mathrm{T})}$ and $\omega^{(\mathrm{P})}$ are set a priori and determine the relative importance of the respective quantities in the optimization objective; in particular, both weights can be set to zero to take only the power into account; in addition, we could choose individual weights for each turbine.

Thus, we formulate the wind farm yaw problem (WFYP) as an integer program (IP) with black-box objective:

$$\max_x \quad \sum_{i=1}^{n_{\mathrm{WT}}} \sum_{j=1}^{n_{\Gamma,i}} f_{\omega,i,j}(x)\, x_{i,j} \tag{1}$$

$$\text{s.t.} \quad \sum_{j=1}^{n_{\Gamma,i}} x_{i,j} = 1 \qquad \text{for } i = 1, \ldots, n_{\mathrm{WT}} \tag{2}$$

$$x_{i,j} \in \{0,1\} \qquad \text{for } i = 1, \ldots, n_{\mathrm{WT}} \quad \text{and} \quad j = 1, \ldots, n_{\Gamma,i}. \tag{3}$$

### 2.1.2 The curse of dimensionality

Due to the black-box nature of the objective function, the above formulation cannot simply be handled by off-the-shelf IP solvers. Indeed, we call it the "basic" formulation because it essentially requires computing *all* $f_{\omega,i,j}(x)$ to obtain a standard (non-black-box) IP, and hence corresponds to the naive brute-force full enumeration. Clearly, this approach is only viable for very small WFYP instances—i.e., few turbines with a small set of admissible yaw angles—but is expensive for larger instances due to the exponential growth of yaw angle combinations; see also the Example 2.2 given below. Moreover, each simulation run incurs a certain run time that itself increases with the farm size. Thus, the WFYP suffers from the typical "curse of dimensionality" often encountered in combinatorial problems. In fact, our following result establishes that an efficient (polynomial-time) solution method for the WFYP very likely does not exist; the proof is deferred to Appendix A.

**Proposition 2.1** (Theorem A.3 and Corollary A.4 from Appendix A)**.** *The wind farm yaw problem is strongly $\mathcal{NP}$-hard and cannot be approximated within any factor $\alpha \leq 1$ in polynomial time (unless $\mathcal{P} = \mathcal{NP}$).*

https://doi.org/10.5194/wes-2024-120
Preprint. Discussion started: 8 October 2024

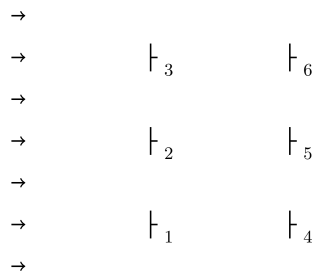




**Figure 2.** Illustration of the $3 \times 2$ farm layout from Example 2.2; the wind direction is indicated by small arrows. The drawing is true to scale for NREL 5-MW turbines, which have a rotor diameter of $D = 126\,\mathrm{m}$, and for a turbine spacing of $3D \times 5D$.

The following example illustrates the WFYP and its "combinatorial explosion", and motivates the circumvention of some complexity issues by the covering approach we will introduce next, in Sect. 2.2. This example, along with some others, will also be used for our computational experiments in Sect. 4.

**Example 2.2** (A $3 \times 2$ farm)**.** We consider $n_{\mathrm{WT}} = 6$ turbines, arranged in a $3 \times 2$ farm, see Fig. 2; we assume the wind blows from the left side and we identify the turbines with the index set $T = \{1,\ldots,6\}$ as depicted. The turbines may be homogeneous of type NREL 5-MW turbines with a rating value of $5\,\mathrm{MW}$ and a rotor diameter of $D = 126\,\mathrm{m}$, cf. (Jonkman et al., 2009, Table 1-1). We set the turbine spacing to $3D \times 5D$, i.e., turbines are on an irregular grid with three and five rotor diameters distance between turbines in the width ("column") and depth ("row") direction, respectively. We choose the notation analogous to matrices; in the literature, our example would usually be referred to as $2 \times 3$ with $5D \times 3D$. However, the $3D \times 5D$ choice is common, see, e.g., example farms in Gebraad et al. (2014); Gebraad and van Wingerden (2014); Boersma et al. (2018a, b) (also, Katic et al. (1987) mentions $5D$ as row value but no column value). We restrict the permissible yaw angles to $\gamma \in [-15°, +15°]$ at $5°$ increments (cf. Sect. 1.2.1), i.e., $\Gamma_i = \Gamma := \{-15°, -10°, \ldots, +15°\}$ and $n_{\Gamma,i} = n_\Gamma := 7$ for every $i \in T$.

**Remark 2.3** (Total number of possible yaw configurations of a $3 \times 2$ farm)**.** *Example 2.2 yields a total number of* ${n_\Gamma}^{n_{\mathrm{WT}}} = 7^6 = 117\,649$ *possible yaw configurations. Consequently, this number of farm simulations would be required to solve the WFYP with the basic approach for one wind scenario. Therefore, we need a different approach to reduce the number of simulations. A coarser yaw angle discretization is not an option as it would sacrifice the level of exercisable control. Indeed, our approach achieves this by reducing the number of yaw angle combinations to consider and by reusing simulation results where possible.*

A turbine has the highest power output with a yaw offset of $0°$ (i.e., it runs greedily), see, e.g., Dar et al. (2017). So, an initial approach that most likely preservers the most power output but reduces the number of options is to let the downstream-most ones run greedily. (We verified by an experiment that the wake of a yawed turbine has no influence on this approach at distance of $3D$.) The number of possible yaw configurations in Example 2.2 would then reduce to $7^3 = 343$; however, such an approach does not scale—a $3 \times 3$ farm again results in $7^6$ configurations. This emphasizes the need for an altogether different approach.

In the following subsection, we propose a method that exploits certain structural redundancies with respect to wind wake interactions to significantly alleviate the number of required simulation runs while retaining the global optimality of the WFYP.

https://doi.org/10.5194/wes-2024-120
Preprint. Discussion started: 8 October 2024


eawe european academy of wind energy
Open Access WIND ENERGY SCIENCE DISCUSSIONS

### 2.2 Section covering approach for WFYP solution

The assumed homogeneous farm layout and composition—i.e., using identical turbines, placed on nodes of an irregular grid—gives rise to certain structural similarities, or reoccurring patterns, that can be exploited to equivalently reformulate the WFYP in a way that reduces the number of black-box evaluations (i.e., simulation runs).

#### 2.2.1 Downstream and upstream sections

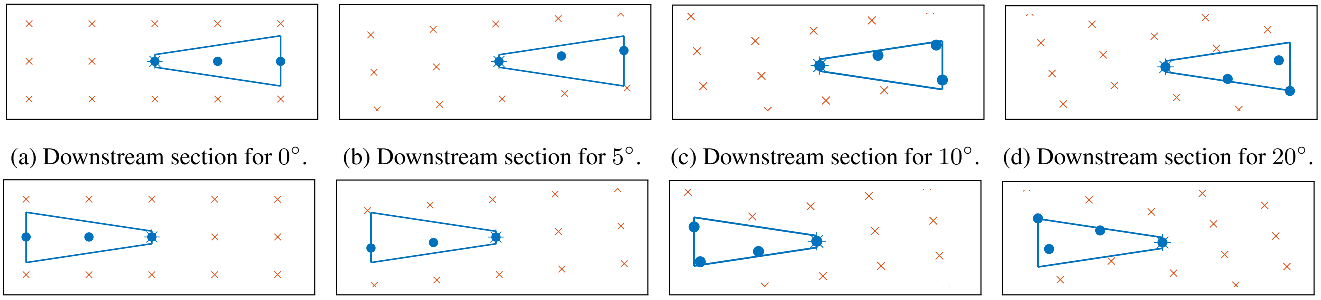
To that end, we take a closer look at the turbines that are *influenced* by a specific turbine, i.e., affected by the downstream wind wake of the latter; we call the set of influenced turbines together with the specific one a *downstream section*. Similarly, we refer to the set of turbines that *influence* a specific one as an *upstream section* (including the specific one). Both depend on the wind conditions (in particular, the wind direction) and the yaw angle(s) of the influencing turbine(s). They include all turbines that *could* be influenced by (as downstream section), or *could* influence (as upstream section), the specific turbine in focus, i.e., sections are based on the admissible yaw angle *ranges*. Moreover, we remind that one assumption is to disregard wake influence if the wake-induced wind speed reduction (relative to the free stream) at the downstream turbine is at most a specific threshold—throughout this paper, $5\%$, see Sect. 1.2.1. The concrete chosen area is a trapezoid (for visualization we sometimes use triangles) based on this threshold: For this, we simply use the wind speeds at the so-called observation points (from the simulation) in the wake, see Sect. 3.1, and set the absolute value of the slope of the trapezoid to at least $0.15$, which corresponds to an angle of approximately $8.5°$, see Fig. 3. (For a rough comparison, Dar et al. (2017) neglect the deficit in velocity for angles beyond $20°$ from the center of the turbine based on Jensen (1983).) Finally, the depth of the trapezoid constructed as described above is usually truncated (or, rarely, extended) to match the depth of the farm—in fact, we use the *effective depth of the wind farm*, which we define in Sect. 2.2.3; it coincides with the depth of the wind farm if the wind direction is $0°$ and can be smaller otherwise.

To illustrate down- and upstream sections, reconsider Example 2.2 and its farm in Fig. 2. Under these fixed wind conditions, WTs $4$, $5$ and $6$ are influenced by WT $2$, depending on the yaw angle $\gamma_2 \in [-15°, +15°]$, i.e., the downstream section at WT $2$ is given by $\{2, 4, 5, 6\}$. Similarly, the upstream section at, e.g., WT $5$ is $\{1, 2, 3, 5\}$. The chosen sections are useful to explain the concept; in fact, with the selected experimental setup for our results in Sect. 4 the corresponding down- and upstream sections both would only include WTs 2 and 5, see Figs. 3(a)/(e) for a corresponding farm with three turbines in depth; and Fig. 3 for an overview with wind directions of $0°$, $5°$, $10°$ and $20°$. In fact, we only consider the downstream section to determine the upstream one, whose pattern is the point reflection of the downstream one on the upstream-most turbine (marked by asterisk).

#### 2.2.2 Section configurations

Depending on the positions of subsets of turbines within the farm (keeping all other aspects fixed), upstream sections can take on different patterns, which can be identified based solely on the grid layout of the farm, see, e.g., Fig. 4. In particular, we can omit (or deactivate) turbines within any upstream section, thereby obtaining what we call *section configurations* as structural subsets of the complete section configuration, i.e., the upstream section itself. Crucially, if all turbines are of the same type,

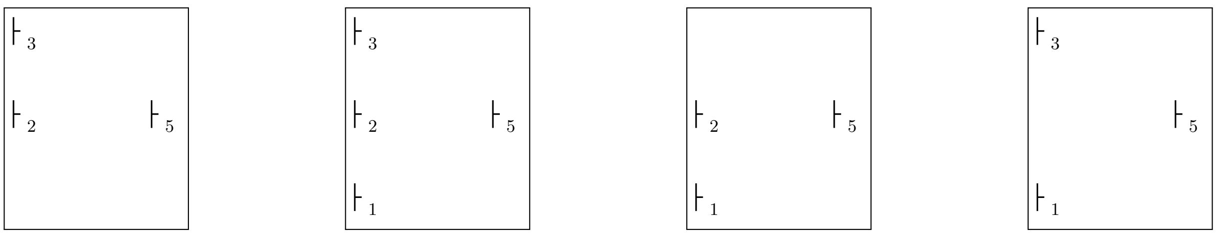
https://doi.org/10.5194/wes-2024-120
Preprint. Discussion started: 8 October 2024

(a) Downstream section for 0°. (b) Downstream section for 5°. (c) Downstream section for 10°. (d) Downstream section for 20°.

(e) Upstream section for 0°. (f) Upstream section for 5°. (g) Upstream section for 10°. (h) Upstream section for 20°.

**Figure 3.** Down- and upstream sections (blue trapezoids) with included turbines (marked as points; asterisks are upstream-most (downstream-most) turbine in downstream (upstream) section) in a grid of turbines (red crosses). The grid represents a farm with infinite expansion.



(a) Section configuration with inactive WT 1. (b) Complete section configuration (i.e., upstream section). (c) Section configuration with inactive WT 3. (d) Section configuration with inactive WT 2.

**Figure 4.** Example of some section configurations of the upstream section with $n_{\mathrm{WT},u} = 4$ turbines. We kept the numbering from Fig. 2, but the depicted patterns may and do occur in other parts of the farm as well.

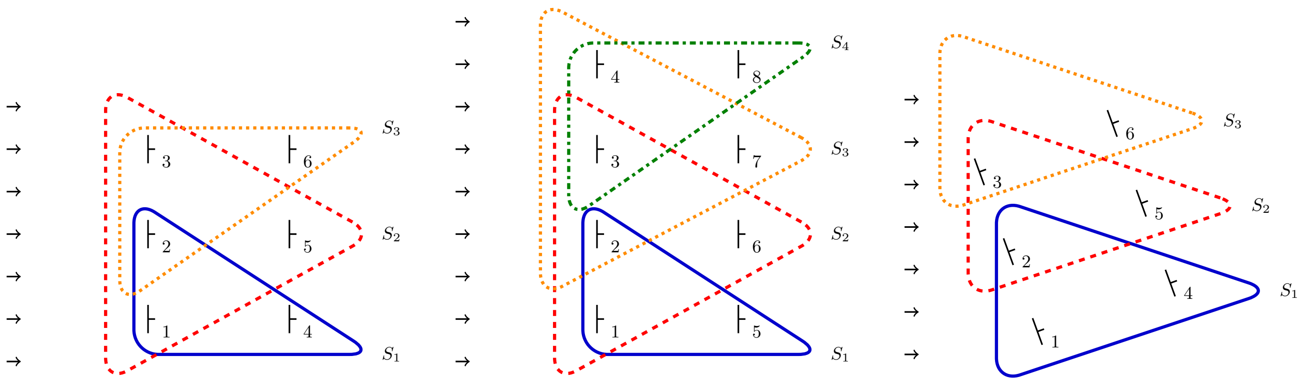
only a single upstream section is needed as a "template" from which to extract the appropriate "patterns" with which the farm can be represented—i.e., we can *cover* the entire farm using (overlapping) section configurations—and simulations can focus on the area of upstream section (reusing simulation results of the section configurations) rather than the whole farm directly. After the following example, we will formalize and explain this *covering approach*.

We can cover the $3 \times 2$ farm from Example 2.2 (cf. Fig. 2) by those section configurations shown in Fig. 4: we anchor the configuration from (b) at WT 5 and the ones from (a) and (c) at WTs 4 and 6 (as the respective downstream-most turbine instead of 5), respectively, since the corresponding parts of the farm exhibit the same structural pattern, see also Fig. 5(a). A change of the farm layout would require other section configurations; e.g., without WT 2 we would need the configuration depicted in Fig. 4(d) for WT 5, and the one with just two turbines directly behind each other (not depicted) for WTs 4 and 6. In total, for the upstream section, we have 16 possible section configurations, including the complete one and the empty one. In general, if an upstream section encompasses $n_{\mathrm{WT},u}$ turbines, there is a total of ${n_{\mathrm{WT},u}}^2$ possible section configurations.

As in the example, we only need a small number of the possible section configurations to cover the farm during normal operation. However, we take into account all possible section configurations: it increases the precomputation time but preserves flexibility, i.e., we are prepared for deactivated turbines and can enlarge the farm orthogonal to the wind direction.

https://doi.org/10.5194/wes-2024-120
Preprint. Discussion started: 8 October 2024


eawe european academy of wind energy
Open Access
WIND ENERGY SCIENCE
DISCUSSIONS




(a) In a $3 \times 2$ farm: $S_1$ (anchored at WT 4) uses the SC from Fig. 4(a) to cover WTs $\{1,2,4\}$, $S_2$ (at WT 5) employs the SC from Fig. 4(b) for $\{1,2,3,5\}$, and $S_3$ (at WT 6) uses the SC in Fig. 4(c) for $\{2,3,6\}$.

(b) In a $4 \times 2$ farm: $S_1$ (anchored at WT 5) uses the SC from Fig. 4(a) to cover WTs $\{1,2,5\}$, both $S_2$ and $S_3$ (at WTs 6 and 7, resp.) use the SC from Fig. 4(b) for $\{1,2,3,6\}$ and $\{2,3,4,7\}$, resp., and $S_4$ (at WT 8) for $\{3,4,8\}$ by the SC from Fig. 4(c).

(c) In a $3 \times 2$ farm with a wind direction of $20°$: $S_1$ to $S_3$ are sufficient to cover the farm although their depth coincides with that of the upstream section, which is smaller than that of the farm. (As we rotated the complete farm, the figure shows a yaw offset of $20°$.)

**Figure 5.** Covering sections $S_1, \ldots, S_3$ in a $3 \times 2$ farm and $S_1, \ldots, S_4$ in a $4 \times 2$ farm, respectively, where $S_1$ is outlined by the solid blue, $S_2$ by the dashed red, $S_3$ by the dotted orange, and $S_4$ by the dash-dotted green line. Subfigure captions specify which section configuration (SC for short) is used by the covering section at a turbine and which turbines are covered by that.

### 2.2.3 Covering sections to reduce the computational burden

To formalize the notion of section configurations that are suitable for covering the farm, it suffices to focus on the downstream-most turbines, whose number we denote by $n_{\rm s}$, and determine so-called covering sections anchored at them.

**Definition 2.4** (Covering sections)**.** A *covering section* is a set $S_k \subseteq T$ of turbines in a farm that influence each other with respect to (wake) disturbances. We denote the *set of covering sections* in a farm by $S := \{S_1, \ldots, S_{n_{\rm s}}\}$. Furthermore, we denote the set of those covering sections that contain a specific turbine $i$ by $S(i) := \{S_k \in S : i \in S_k\}$.

To cover the farm, we must assign one covering section to each downstream-most turbine, as illustrated in Fig. 5, but as different covering sections can have the same pattern, a section configuration can be used several times, e.g., in Fig. 5(b). The core advantage of this covering approach is the significantly reduced number of simulations required to find the best WFYP solution (in comparison to full enumeration) as we only need to precompute the yaw configurations within every (distinct) section configuration and, accordingly, obtain the simulation results for all covering sections.

We already mentioned that wind directions deviating from $0°$ require to define the *effective depth* of the wind farm. For illustration, we use a $3 \times 2$ farm with a wind direction of $0°$ and $20°$, see Figs. 5(a)/(c). In the case of $0°$, the farm and the corresponding upstream section both have a depth of $5D$. For other directions, e.g. as in Fig. 5(c), this depth changes. The *depth of the wind farm* is the distance in $x$-direction between the upstream- and downstream-most turbines *inside the wind farm*, i.e.,

https://doi.org/10.5194/wes-2024-120
Preprint. Discussion started: 8 October 2024


in our example between WTs 3 and 4. The *depth of the upstream section* is analogously defined *inside the upstream section*—in our example, WTs 2 and 4. As this depth is sufficient to finally cover the farm we also call it *effective depth of the wind farm*. Usually, our use of terms "upstream-" and "downstream-most" turbines refers to these covering sections, e.g., in our example, WTs 4 to 6 are the downstream-most ones and serve as anchors for the covering sections. If an anchor turbine is missing, say WT 4, we relocate the covering section, i.e., in our example, we attach $S_1$ at WT 1 (where $S_1$ uses the section configuration with only one active turbine). Then we can assume without loss of generality that the downstream-most turbine inside a section configuration is always active, thus circumventing the half of all combinations in which the downstream-most turbine could be inactive. (Alternatively to this relocation, one could have anchored the covering section at a deactivated "virtual" WT 4.)

### 2.2.4 The required number of simulations

Before we turn to the WFYP model based on the covering approach, we take a closer look at the required number of simulations to solve it. For simplicity, we assume the same set of admissible yaw angles, say, $\Gamma$ with $n_\Gamma := |\Gamma|$ for the (identical) turbines. In the basic approach for the whole farm with $n_{\mathrm{WT}}$ turbines, we saw below Remark 2.3 that the total number of distinct yaw configurations, which coincides with the required number of simulations, amounts to ${n_\Gamma}^{n_{\mathrm{WT}}-n_{\mathrm{s}}}$ if the $n_{\mathrm{s}}$ downstream-most turbines run greedily. Analogously, again running the downstream-most turbine greedily, an upstream section with $n_{\mathrm{WT},u}$ turbines admits ${n_\Gamma}^{n_{\mathrm{WT},u}-1}$ possible yaw configurations and a covering section $S_k$ with $n_{\mathrm{WT},k}$ turbines admits $n_{\Delta,k} := {n_\Gamma}^{n_{\mathrm{WT},k}-1}$.

To solve a single WFYP instance, we perform precomputations, i.e., simulation runs for all yaw configurations on all possible section configurations; recall that we include all possible ones (not only those that occur as covering sections) to preserve flexibility, see Sect. 2.2.2, and thus, we derive the worst-case number of simulations. In addition, we remind that farm layout and wind conditions (in particular, direction and speed) are fixed for a single WFYP instance. Consequently, while for each scenario, the number of simulations is much lower than in the basic approach, preparing our approach for application in a variety of wind scenarios for a given farm will still result in a large precomputation time to run all required simulations. However, we propose to store these precomputed simulation results in a database so that data corresponding to any currently encountered scenario can be retrieved efficiently to solve the corresponding WFYP instance in order to update the yaw control.

In the following, we derive the number of combinations (section configurations and corresponding yaw configurations) and the required number of simulation runs. As there are different ways to avoid further redundant computations in specific situations, these numbers are upper bounds and might be further reduced; we will mention some examples of this aspect.

An upstream section (i.e., complete section configuration) with $n_{\mathrm{WT},u}$ turbines has ${n_{\mathrm{WT},u}}^2-1$ non-empty section configurations. All other section configurations have fewer active turbines than $n_{\mathrm{WT},u}$, and, consequently, admit fewer possible yaw configurations (than the complete one), i.e, the number of required simulation runs is smaller than $({n_{\mathrm{WT},u}}^2-1)\,{n_\Gamma}^{n_{\mathrm{WT},u}-1}$. We derive the exact worst-case number of simulations needed, i.e., the total count of all yaw configurations for all possible section configurations (for $n_{\mathrm{WT},u}$ turbines) that are non-empty and have an active downstream-most turbine running greedily (cf. Sect. 2.2.3). The remaining ones can then either be inactive or active with one of $n_\Gamma$ yaw angles. Thus, to select $n \in \{0,1,\ldots,n_{\mathrm{WT},u}-1\}$ active turbines among these, there are $\binom{n_{\mathrm{WT},u}-1}{n}$ distinct possibilities and, for any selection of $n$

active turbines, ${n_\Gamma}^n$ possible yaw configurations. Thus, the total number of simulations amounts to

$$n_{\mathrm{sim}} := \sum_{n=0}^{n_{\mathrm{WT},u}-1} {n_\Gamma}^n \binom{n_{\mathrm{WT},u}-1}{n}. \tag{4}$$

In case of our Example 2.2 ($3 \times 2$ farm), see Fig. 5(a), with $n_{\mathrm{WT},u} = 4$ turbines in the upstream section, the formula yields $7^0\binom{3}{0} + 7^1\binom{3}{1} + 7^2\binom{3}{2} + 7^3\binom{3}{3} = 1 + 21 + 147 + 343 = 512$ combinations (i.e., number of simulations); this also applies to the $4 \times 2$ farm, see Fig. 5(b), and all enlargements orthogonal to wind direction as they build on the same upstream section. In comparison, if all turbines are active (and the downstream-most ones still run greedily), the basic approach (full enumeration) leads to ${n_\Gamma}^3 = 7^3 = 343$ required simulations (for $3 \times 2$) and to $7^4 = 2401$ (for $4 \times 2$), which seems to be a better choice for the $3 \times 2$ farm. However, the covering approach already includes the possibility to deactivate any turbines, see Sect. 2.2.3. If this were included in the basic approach, we would end up with $(n_\Gamma + 1)^3 \cdot 2^3 = 8^3 \cdot 2^3 = 4096$ combinations (for $3 \times 2$) and $8^4 \cdot 2^4 = 65\,536$ (for $4 \times 2$). Thus, we expect that the covering approach provides a significantly higher efficiency than the basic approach for most real-world farm layouts and wind directions.

Recall that we need these precomputations for each wind condition, in particular, we focus on direction and speed, see Sect. 1.2.1. Usually, these are also discretized in a wind rose, see, e.g., the figures in Zhang et al. (2014); Fleming et al. (2016), which is a circular histogram with the distribution of directions and speeds (e.g., in $5°$ and $1\,\mathrm{m\,s^{-1}}$ steps from 3 to $20\,\mathrm{m\,s^{-1}}$; lower/higher speeds are summarized separately, see Gebraad et al. (2017)). Analogously to the yaw angle discretization, a finer discretization is possible but questionable because of the uncertainty of incident wind conditions as discussed in Sect. 1.2.1.

### 2.3 Formulation of the covering approach as an integer program

It remains to formalize how we can use the covering sections to solve the WFYP globally optimal. So, recall the idea to represent the farm as a set of overlapping covering sections (cf. Def. 2.4) rather than of single turbines. Instead of deciding directly on the yaw angle of each turbine, decision variables assign a specific yaw configuration to each covering section. For the consistency of the farm covering, we require that each turbine in intersecting parts of different covering sections consistently has the same yaw angle in these. This, together with the requirement that each covering section is assigned exactly one yaw configuration, mirrors the constraint of the basic approach that each turbine can only be set at one yaw angle, cf. Eqs. (1) to (3).

#### 2.3.1 Contributions of wind turbines located at overlaps of covering sections

Recall that in the basic WFYP approach, the objective function has coefficients (from simulations) for each turbine and yaw angle configuration. Now, we have contributions related to assigning yaw configurations (with respect to the underlying section configuration) to covering sections. To avoid multiple counting of the individual contributions of turbines located at overlaps of covering sections, which are available from the simulation results (see vector-valued function $f_\omega$ in Sect. 2.1.1), we consider the covering sections in order of their indices $(S_1, S_2, \ldots, S_{n_s})$ and construct the objective by only adding contributions of turbines in a current covering section $S_k$ if they were not already contained in the previous covering sections $S_1, \ldots, S_{k-1}$. Let $T_k := S_k \setminus (\cup_{m=1}^{k-1} S_m)$ denote the set of new turbines in covering section $S_k$; e.g., in the example from Fig. 5(a), it holds

https://doi.org/10.5194/wes-2024-120
Preprint. Discussion started: 8 October 2024


that $T_1 = \{1,2,4\}$, $T_2 = \{3,5\}$, and $T_3 = \{6\}$. Then, we can express the WFYP objective value of a given yaw configuration assignment (one $\ell_k$ for each respective covering section $S_k$) with our previously-used black-box function as

$$\sum_{k=1}^{n_s} \sum_{i \in T_k} f_{\omega,i,j(\ell_k)}(x(\ell_k)), \tag{5}$$

where $x(\ell_k)$ stands for the individual-turbine yaw angle settings across covering section $S_k$, which now depend on the yaw configuration $\ell_k$ given for each section $S_k$, and $j(\ell_k)$ is the corresponding yaw angle index of turbine $i$.

### 2.3.2 Compatibility of yaw configurations in covering sections

The following paragraphs discuss the consistency of the farm covering. In doing so, we will describe the details of the covering approach outlined above and lead up to the integer program (7) to (10). Again, for simplicity, we assume the same set of admissible yaw angles, say, $\Gamma$ with $n_\Gamma := |\Gamma|$ for the (identical) turbines. The appropriate covering sections (and required underlying configuration sections) are defined before building the integer programming model, recall Sect. 2.2.3.

To specify the IP model, we need additional notation. For a covering section $S_k \subseteq T$ ($k = 1,\dots,n_s$) with $n_{\mathrm{WT},k}$ turbines, we identify the yaw configurations inside $S_k$ by indices $\ell_k = 1,2,\dots,n_{\Delta,k}$ with $n_{\Delta,k} := {n_\Gamma}^{n_{\mathrm{WT},k}-1}$ (as defined in Sect. 2.2.4). Let $\gamma_i(\ell_k)$ denote the yaw angle assigned to turbine $i \in S_k$ under yaw configuration $\ell_k$. For consistency of the global yaw configuration as a composition of sectional yaw configurations, the yaw configurations of overlapping covering sections must match on the yaw angles of turbines located in the respective intersection. To that end, if the yaw configuration $\ell_k$ was selected for $S_k$, then, for any $S_{\hat{k}}$ with $S_{\hat{k}} \cap S_k \neq \emptyset$ for $\hat{k} \neq k$, only a subset of yaw configurations is compatible with this selection, namely those $\ell_{\hat{k}} \in \{1,2,\dots,n_{\Delta,\hat{k}}\}$ for which the yaw angles $\gamma_i(\ell_{\hat{k}}) = \gamma_i(\ell_k)$ for all WTs $i \in S_{\hat{k}} \cap S_k$. In fact, it suffices to enforce these conditions explicitly for directly adjacent pairs of covering sections, if they are numbered in ascending sequence in accordance with the downstream-most turbines (say, $1,\dots,n_s$ from left to right from behind the farm looking against the wind direction). Then, establishing consistency of the respective overlaps of $S_k$ and $S_{k+1}$ by resorting to valid yaw configurations for $S_{k+1}$ (defined relative to $S_k$ and each $\ell_k$), for $k = 1,\dots,n_s - 1$, is indeed enough to guarantee global consistency, as by construction, for any WT $i \in S_{\hat{k}} \cap S_k$ with $\hat{k} \geq k+2$, necessarily also WT $i \in S_{k+1}$. In addition to $L_k := \{1,2,\dots,n_{\Delta,k}\}$, the index set of all possible yaw configurations for $S_k$, we therefore also need the set of *valid* (or *compatible*) *yaw configurations* for $S_{k+1}$ relative to $S_k$ with $\ell_k \in L_k$, which we denote as $\tilde{L}_{k+1,\ell_k} := \{\ell_{k+1} \in L_{k+1} : \gamma_i(\ell_{k+1}) = \gamma_i(\ell_k) \text{ for all } i \in S_{k+1} \cap S_k\}$. For $S_1$, all possible yaw configurations in $L_1$ are already valid, i.e., $\tilde{L}_1 = L_1$, as $S_1$ has no “preceding” covering section. Tables with these dependencies, i.e., the set of valid yaw configurations $\tilde{L}_{k+1,\ell_k}$ for the current covering section (numbered as $k+1$) in dependence of the previous one (numbered as $k$) and its chosen yaw configuration $\ell_k$, can be computed straightforwardly.

To illustrate the notions, we use Example 2.2 ($3 \times 2$ farm) again. For each covering section, marked in Fig. 5(a), we need to uniquely identify every possible yaw configuration with an index, e.g., by sorting them lexicographically with respect to the yaw angles (in increasing order of the turbine indices); Table 1 shows an example for covering section $S_1$, assuming for simplicity $\{-15°, 0°, +15°\}$ as admissible yaw angles—downstream-most turbines (4, 5, and 6) are fixed to 0°, cf. Sect. 1.2.1.

Assuming the same lexicographic indexing to the yaw configurations for each section configuration (and thus for the covering sections), we can determine the sets of valid yaw configurations; Table 2 shows the sets $\tilde{L}_{k+1,\ell_k}$ for our simplified example.

https://doi.org/10.5194/wes-2024-120
Preprint. Discussion started: 8 October 2024


**Table 1.** Lexicographic indexing for the $n_{\Delta,1} = 3^2 = 9$ yaw configurations for the underlying section configuration to covering section $S_1 = \{1,2,4\}$ in the $3 \times 2$ farm Example 2.2, cf. Fig. 5(a), with simplified $\Gamma = \{-15°, 0°, +15°\}$ (and WT 4 fixed to 0°).

| $\ell_1$ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|
| WT 1 | $-15°$ | $-15°$ | $-15°$ | $0°$ | $0°$ | $0°$ | $+15°$ | $+15°$ | $+15°$ |
| WT 2 | $-15°$ | $0°$ | $+15°$ | $-15°$ | $0°$ | $+15°$ | $-15°$ | $0°$ | $+15°$ |
| WT 4 | $0°$ | $0°$ | $0°$ | $0°$ | $0°$ | $0°$ | $0°$ | $0°$ | $0°$ |

**Table 2.** Valid yaw configurations $\tilde{L}_{k+1,\ell_k}$ of covering section $S_{k+1}$ (depending on yaw configuration $\ell_k$ of the previous one $S_k$) compared to all possible yaw configurations $L_{k+1}$ of $S_{k+1}$ for the $3 \times 2$ farm from Example 2.2, Fig. 5(a), assuming admissible yaw angle offsets $\{-15°, 0°, +15°\}$ (simplified for the example) for every turbine, fixed angle 0° for downstream-most turbines and yaw configurations being indexed in lexicographical order. For $S_1$, all possible yaw configurations $L_1 = \{1, 2, \ldots, 9\}$ are also valid by design, i.e., $\tilde{L}_1 = L_1$.

| $k$ | 1 | 1 | 1 | … | 1 | 2 | 2 | 2 | … | 2 |
|---|---|---|---|---|---|---|---|---|---|---|
| $\ell_k$ | 1 | 2 | 3 | … | 9 | 1, 4, 7 | 2, 5, 8 | 3, 6, 9 | … | 21, 24, 27 |
| $L_{k+1}$ | $\{1,\ldots,27\}$ | $\{1,\ldots,27\}$ | $\{1,\ldots,27\}$ | … | $\{1,\ldots,27\}$ | $\{1,\ldots,9\}$ | $\{1,\ldots,9\}$ | $\{1,\ldots,9\}$ | … | $\{1,\ldots,9\}$ |
| $\tilde{L}_{k+1,\ell_k}$ | $\{1,2,3\}$ | $\{4,5,6\}$ | $\{7,8,9\}$ | … | $\{25,26,27\}$ | $\{1\}$ | $\{2\}$ | $\{3\}$ | … | $\{9\}$ |

For instance, if yaw configuration $\ell_1 = 3$ was used for $S_1$, then only those yaw configurations for $S_2$ in which turbines 1 and 2 also have yaw angles $-15°$ and $+15°$, respectively, are valid for $S_2$; with the used indexing, this amounts to $\tilde{L}_{2,3} = \{7, 8, 9\}$. For $\ell_2 = 7$, only yaw configuration $\ell_3 = 1$ is valid; indeed, $\tilde{L}_{3,7} = \tilde{L}_{3,1} = \tilde{L}_{3,4} = \{1\}$, as these yaw configurations for $S_2$ set both turbines 2 and 3 to $-15°$, for which the only compatible (and hence, valid) yaw configuration for $S_3$ is precisely $\ell_3 = 1$.

### 2.3.3 WFYP formulation as regular integer program

Now, we introduce binary decision variables $y_{k,\ell_k}$ that encode whether covering section $S_k$ is assigned yaw configuration $\ell_k$ ($y_{k,\ell_k} = 1$) or not ($y_{k,\ell_k} = 0$). Using these variables, the black-box objective function, cf. Eq. (1), can be replaced by a fully linear one once we have precomputed the simulation results for the section configurations. Indeed, the simulation results allow us to specify cost coefficients $c_{k,\ell_k}$ for every pair of a covering section $S_k$ and any one of its associated yaw configurations $\ell_k$; in order to avoid counting the objective contributions of turbines within intersecting parts of different covering sections multiple times, we again use a summation that only considers contributions of new turbines in a covering section, cf. Eq. (5):

$$c_{k,\ell_k} := \sum_{i \in T_k} f_{\omega,i,j(\ell_k)}(x(\ell_k)). \tag{6}$$

To achieve a globally optimal yaw configuration for the whole farm, we now have to optimize over all compatible combinations, see Sect. 2.3.2, of covering section and yaw configuration assignments (each of which has one associated coefficient $c_{k,\ell_k}$ and

https://doi.org/10.5194/wes-2024-120
Preprint. Discussion started: 8 October 2024


one decision variable $y_{k,\ell_k}$). This yields the following integer linear program to solve the WFYP:

$$\max_{y} \quad \sum_{k=1}^{n_\mathrm{s}} \sum_{\ell_k=1}^{n_{\Delta,k}} c_{k,\ell_k}\, y_{k,\ell_k} \tag{7}$$

$$\text{s.t.} \quad \sum_{\ell_k=1}^{n_{\Delta,k}} y_{k,\ell_k} = 1 \qquad \text{for } k = 1,\dots,n_\mathrm{s}, \tag{8}$$

$$0 \leq \sum_{\ell_{k+1} \in \tilde{L}_{k+1,\ell_k}} y_{k+1,\ell_{k+1}} - y_{k,\ell_k} \leq 1 \qquad \text{for } k = 1,\dots,n_\mathrm{s}-1 \quad \text{and} \quad \ell_k = 1,\dots,n_{\Delta,k}, \tag{9}$$

$$y_{k,\ell_k} \in \{0,1\} \qquad \text{for } k = 1,\dots,n_\mathrm{s} \quad \text{and} \quad \ell_k = 1,\dots,n_{\Delta,k}. \tag{10}$$

Constraints (8) ensure that exactly one yaw configuration is selected for each covering section, analogously to Eq. (2). Constraints (9) ensure compatibility as they enforce the selected yaw configuration for a covering section $S_{k+1}$ to be *valid* with respect to the yaw configuration chosen for its preceding one $S_k$, as described in Sect. 2.3.2: If $y_{k,\ell_k} = 1$, i.e., $S_k$ uses yaw configuration $\ell_k$, then for $S_{k+1}$, a yaw configuration from $\tilde{L}_{k+1,\ell_k}$ must be selected, i.e., one of the associated binary variables—and hence, their sum—must be one. If $y_{k,\ell_k} = 0$, the constraint imposes no restriction[1] with respect to $\tilde{L}_{k+1,\ell_k}$.

Finally, we emphasize that both the black-box IP (1) to (3) and the regular IP (7) to (10) are different formulations of *the same problem*, i.e., the WFYP; as such, they are equivalent. Nevertheless, the covering approach exploits the problem structure in a way that can significantly reduce the required number of simulations and enables the utilization of modern IP solvers to perform efficient implicit enumeration by branch and bound, thereby avoiding total enumeration.

## 3 Simulation

We obtain the simulation function output from simulation software; in principle, our approach allows to employ *any* suitable simulation. Moreover, recall that our WFYP integer programs (Eqs. (1) to (3) or (7) to (10)) need simulation function values for different yaw angle configurations. As we control the yaw angles, we only denote the corresponding decision variables $x$ as simulation function arguments; the other inputs (farm layout, wind direction and speed) will be made clear in our experiments. In the following, we introduce other conditions on which the simulation also depends, along with the corresponding fixed values we used. We describe the simulation software and our parameter setup in Sect. 3.1. Then, in Sect. 3.2, we specify relevant simulation outputs and define performance indicators based on them, which are combined to form the simulation function.

[1] The upper bound in Eq. (9) is redundant: we investigated the effects for $6 \times 3$ to $9 \times 3$ farms as in series 2, see Table 5; the redundant conditions slightly increase or decrease (in two out of four cases each) the solving time of the IPs if Gurobi is used as solver, e.g., for $9 \times 3$, $11.25\,\mathrm{s}$ instead of $14.69\,\mathrm{s}$, but significantly decrease it for this case with SCIP ($9071.55\,\mathrm{s}$ instead of $20\,812.22\,\mathrm{s}$); in the other cases SCIP is slower with the redundant conditions (e.g., for $8 \times 3$, $1760.04\,\mathrm{s}$ instead of $838.63\,\mathrm{s}$).

https://doi.org/10.5194/wes-2024-120
Preprint. Discussion started: 8 October 2024


eawe european academy of wind energy
Open Access WIND ENERGY SCIENCE DISCUSSIONS

### 3.1 Simulation software and parameter setup

For the wind farm simulations, we used the software package WinFaST[2]. This simulation framework requires a fixed farm layout, yet setting individual parameters for axial induction and yaw angles is generally possible. As our focus lies on optimal yaw angles, we leave the greedy control with respect to axial induction to the local controller. The dynamic wake model of WinFaST is based on FLORIDyn, see Gebraad and van Wingerden (2014). As FLORIDyn, it uses so-called observation points to compute local wake characteristics and wake interaction is based on Katic et al. (1987). The turbine controller in WinFaST is inspired by Jonkman et al. (2009), which is widely used for NREL 5-MW turbines, extended by the options (not used by us) to reduce the power and damp tower oscillations, each with respect to its own respective turbine. Moreover, WinFaST uses a modified version (to include yaw control and effects) of the dynamic wind turbine model by Ritter et al. (2016, 2018). The wind field in WinFaST is simulated by Veers method, see Veers (1988).

We denote the average wind speed value of the (horizontal) ambient wind field by $U_{\mathrm{ave}}$. The turbulence intensity (TI) is defined as $I = \sigma / U_{\mathrm{ave}}$, where $\sigma$ is the associated standard deviation; it depends on the average wind speed, the roughness of the surface, the atmospheric stability, and the topography, see, e.g., (Hau, 2013, Sect. 13.4). The software WinFaST uses the same parametric model parameters for turbine and wake as in (Gebraad and van Wingerden, 2014, Table 1), that were adjusted for $8\,\mathrm{m\,s^{-1}}$ with a TI of 6%, with the exception of the air density, which is set to $1.225\,\mathrm{kg\,m^{-3}}$ as in (Jonkman et al., 2009, Appendix B.1). In our exemplary experiments, we fix the TI to $6\%$.

### 3.2 Performance indicators and simulation function

It takes a while for the wake of the upstream-most turbine(s) to reach the downstream-most one. Thus, we need to choose a sufficiently long *simulation time interval* $[t_{\mathrm{s}_1}, t_{\mathrm{s}_2}]$, depending on the wind speed and farm layout; here, we used $9\,\mathrm{min}$. Moreover, for data analysis and as yaw angles are adjusted at a fairly low rate, we are only interested in the simulation part in which the wake already influences the downstream-most turbine. Also, the wind field is equipped with turbulence and the turbines produce some, so we analyze data over an *observation time interval* $[t_{\mathrm{o}_1}, t_{\mathrm{o}_2}]$. In our case, this includes the middle $5\,\mathrm{min}$ of our simulation time interval; we use it to compute the performance indicators and to define our simulation function.

The performance indicators (cf. Eq. (11) later) consist of the following three outputs of WinFaST: The *power* generated by each turbine is given in the unit W as function $p_x : [t_{\mathrm{s}_1}, t_{\mathrm{s}_2}] \to \mathbb{R}^{n_{\mathrm{WT}}}_{\geq 0}$. To compute loads we use the *velocity of the nacelle* in unit $\mathrm{m\,s^{-1}}$ in wind direction, $v_x : [t_{\mathrm{s}_1}, t_{\mathrm{s}_2}] \to \mathbb{R}^{n_{\mathrm{WT}}}$, and the *blade pitch angle* in the unit degree, $\beta_x : [t_{\mathrm{s}_1}, t_{\mathrm{s}_2}] \to \mathbb{R}^{n_{\mathrm{WT}}}$.

Now, we define the three performance indicators for each turbine $i$ as averages over the observation time interval $[t_{\mathrm{o}_1}, t_{\mathrm{o}_2}]$, namely the power (output) $P_i$ (in MW), the *tower activity* $a_i^{(\mathrm{T})}$, and the *pitch activity* $a_i^{(\mathrm{P})}$. The tower load is high when the nacelle is oscillating; therefore, we use the absolute value of the nacelle velocity $v$ to estimate the tower load by the so-called tower activity. The pitch rate should be kept within limits because of the load of the pitch actuators; therefore, similarly, we use the absolute value of the velocity of the blade pitch angle $\beta$ to estimate the load of the pitch actuators by the so-called pitch

[2]The MATLAB package WinFaST (Wind Farm Simulation Tool), written by Bastian Ritter and Thorsten Schlicht, is proprietary software for company-internal use at our industry partner IAV GmbH, who provided it to us for experimentation within the joint MOReNet project.

https://doi.org/10.5194/wes-2024-120
Preprint. Discussion started: 8 October 2024



activity. The performance indicators are defined as follows:

$$P_i(x) := \frac{1}{t_{o_2} - t_{o_1}} \int_{t_{o_1}}^{t_{o_2}} 10^{-6}\,(p_x(t))_i\,\mathrm{d}t, \quad a_i^{(\mathrm{T})}(x) := \frac{1}{t_{o_2} - t_{o_1}} \int_{t_{o_1}}^{t_{o_2}} |(v_x(t))_i|\,\mathrm{d}t, \quad a_i^{(\mathrm{P})}(x) := \frac{1}{t_{o_2} - t_{o_1}} \int_{t_{o_1}}^{t_{o_2}} \left|\frac{\mathrm{d}}{\mathrm{d}t}(\beta_x(t))_i\right|\mathrm{d}t.$$

The respective units of tower and pitch activity are $\mathrm{m\,s^{-1}}$ and $^\circ\,\mathrm{s}^{-1}$ but have no physical meaning.

Finally, we define the weighted sum of these three performance indicators as the simulation function depending on the control input, i.e., the decision variables $x$. Recall that the dependence on yaw configurations also includes that of a turbine $i \in T$ on its own yaw angle, which can be expressed using the yaw angle index $j \in n_{\Gamma,i}$. Therefore, following the notation introduced in Sect. 2.1, we write $P_{i,j}(x)$, $a_{i,j}^{(\mathrm{T})}(x)$ and $a_{i,j}^{(\mathrm{P})}(x)$. With weights $\omega = (\omega^{(\mathrm{T})}, \omega^{(\mathrm{P})}) \in \mathbb{R}^2_{\geq 0}$ for the activity terms the entries of the simulation function, which yields the black-box function to maximize, are:

$$f_{\omega,i,j}(x) := P_{i,j}(x) - \omega^{(\mathrm{T})}\, a_{i,j}^{(\mathrm{T})}(x) - \omega^{(\mathrm{P})}\, a_{i,j}^{(\mathrm{P})}(x). \tag{11}$$

It represents our two main objectives when controlling the farm: maximizing the total power output and minimizing the turbines' mechanical load; the weights balance these typically conflicting objectives. For clarity of notation, we omitted individually weighting the turbines. Moreover, for simplicity, we focus on the power output, i.e., we set $\omega^{(\mathrm{T})} = \omega^{(\mathrm{P})} = 0$ in Sect. 4. In practice, one "simulation run" consists of evaluating the (vector-valued) simulation function $f_\omega : \{0,1\}^{n_{\Gamma,1}+\cdots+n_{\Gamma,n_{\mathrm{WT}}}} \to \mathbb{R}^{n_{\mathrm{WT}}}$ (with entries of the form $f_{\omega,i,j}(x)$) once for the associated section and assignment of decision variables $x$.

## 4 Computational results

This section presents four series of computational experiments in a farm for our covering approach: In Sect. 4.1, we consider optimizing yaw angles for different wind directions (series 1). In Sect. 4.2, we demonstrate that reusing some precomputed simulation results of series 1 enables our covering approach to enlarge farms orthogonal to the wind direction and to handle cases with deactivated turbines (series 2). In this context, we also investigate the influence of the farm size on the solving time to assess how the theoretical complexity of the WFYP is reflected in practice. In series 3, we consider different wind speeds, see Sect. 4.3, and finally, in series 4 we experiment with a wider range of yaw angles and a finer discretization, see Sect. 4.4.

Before we discuss the results, we briefly summarize the most important parts of the overall experimental setup. As admissible yaw angles we choose $\gamma \in \Gamma = \{-15^\circ, -10^\circ, \ldots, 15^\circ\}$ for NREL 5-MW turbines with a rotor diameter of $D = 126\,\mathrm{m}$, see (Jonkman et al., 2009, Table 1-1), in different farm layouts from $6 \times 3$ to $9 \times 3$ with a turbine spacing of $3D \times 5D$. In all figures, we display wind direction as blowing from left to right; the actual wind direction is then reflected as a "rotation" of the relative layout placement of a farm by a corresponding amount of $0^\circ$, $5^\circ$, $10^\circ$ or $20^\circ$, respectively. As average wind speeds we consider 6, 11, and $12\,\mathrm{m\,s^{-1}}$. Deviations from this setup are made clear where they occur. Finally, we frequently compare the optimized yaw offsets against the baseline, i.e., yaw offsets of $0^\circ$.

All computations were carried out on a Linux workstation with an Intel(R) Core(TM) i7-6700 CPU with 3.40 GHz (4 cores, 8 threads) and 32 GB memory. The precomputations (simulation runs) were done using MATLAB R2023a, utilizing

https://doi.org/10.5194/wes-2024-120
Preprint. Discussion started: 8 October 2024




**Table 3.** Data and results for a $6 \times 3$ farm, wind speed of $11\,\mathrm{m\,s^{-1}}$ and wind directions of $0°$, $5°$, $10°$ and $20°$ (series 1). Detailed results are in Table 4 (cases 1 to 3) and Fig. 6 (case 4). In all cases, the IP optimality gap is $0.00\%$, i.e., all instances were solved to global optimality.

| | wind farm | | | IP solver | | | | total power output | | |
|---|---|---|---|---|---|---|---|---|---|---|
| case | wind direction (in °) | # covering sections | precomputation time (in h) | # variables | # constraints | solving time (in s) SCIP | Gurobi | baseline (in MW) | optimized (in MW) | improvement rel. (in %) |
| 1 | 0 | 6 | 0.5 | 294 | 496 | 0.04 | 0.12 | 43.09 | 46.01 | 6.78 |
| 2 | 5 | 6 | 0.5 | 294 | 496 | 0.04 | 0.01 | 43.84 | 51.41 | 17.27 |
| 3 | 10 | 6 | 2.9 | 1764 | 3436 | 0.82 | 0.27 | 52.33 | 57.40 | 9.69 |
| 4 | 20 | 8 | 2.9 | 1430 | 2866 | 86.85 | 0.35 | 62.05 | 62.98 | 1.50 |

**Table 4.** Detailed results of series 1, cases 1 to 3 (see Fig. 6 for case 4) as shown in Table 3. For both yaw controls (baseline or optimized) the power output of each individual turbine is sorted by rows, i.e., WTs 1 to 6 in row 1, WTs 7 to 12 in row 2 and WTs 13 to 18 in row 3, as they often coincide in these examples; exceptions are indicated in brackets, e.g., in case 3, WT 1 outputs $4.15\,\mathrm{MW}$. Further, optimal yaw angles are given sorted by rows (without row 3, as all yaw angles are fixed to $0°$ there).

| | wind farm | baseline turbines' power output | | | optimized turbines' power output | | | opt. yaw angles | |
|---|---|---|---|---|---|---|---|---|---|
| case | wind direction (in °) | row 1 (in MW) | row 2 (in MW) | row 3 (in MW) | row 1 (in MW) | row 2 (in MW) | row 3 (in MW) | row 1 (in °) | row 2 (in °) |
| 1 | 0 | 4.87 | 1.79 | 0.52 | 4.79 | 1.97 | 0.91 | 15.0 | 15.0 |
| 2 | 5 | 4.37 | 1.83 | 1.11 | 4.14 | 2.45 | 1.97 | 15.0 | 15.0 |
| 3 | 10 | (1: 4.15) 4.13 | (12: 2.74) 2.60 | (18: 2.32) 1.90 | (1: 3.92) 3.88 | (12: 3.20) 3.02 | (18: 3.09) 2.53 | 15.0 | 15.0 |

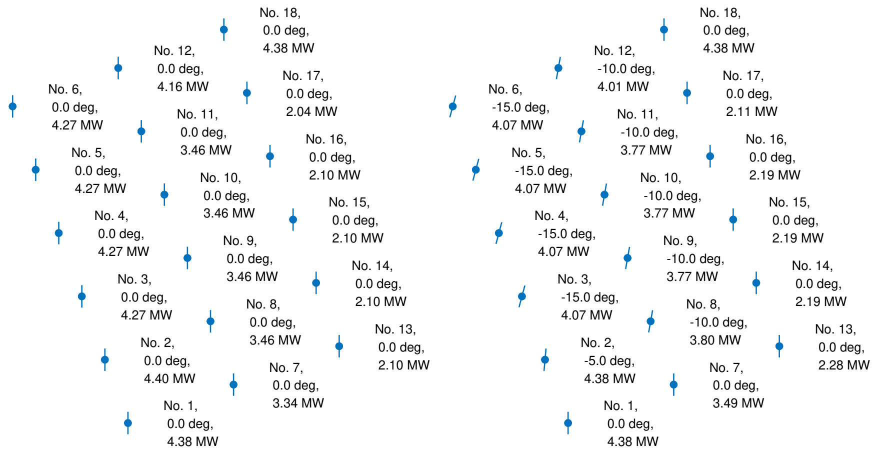
parallelization. The IPs resulting from our covering approach were solved with state-of-the-art IP solvers, namely the open-sourced academic solver SCIP 8.0.3, utilizing the LP solver SoPlex 6.0.3, which only supports single-thread, see Bestuzheva et al. (2021), as well as the proprietary Gurobi 10.0.0, which can employ all threads, see Gurobi Optimization, LLC (2022).

### 4.1 Wind farm yaw angle optimization under different wind directions

In series 1, we consider a $6 \times 3$ farm with a wind speed of $11\,\mathrm{m\,s^{-1}}$ and wind directions of $0°$, $5°$, $10°$ and $20°$. The respective results are presented in Tables 3 and 4 (cases 1 to 3) and Fig. 6 (case 4). In all cases, the improvement of the total power output is between $2\%$ and $17\%$. In most of these cases, the optimal yaw angles exhaust the given limits of $\pm 15°$. In case 4 (i.e., wind direction of $20°$), the distances between the turbines in the downstream direction are already comparatively high and consequently, the wake influence comparatively low; therefore, the improvement is $2\%$ here, but larger in the other cases.

The main part of the overall run time is the precomputation time, e.g., in series 1 between half an hour and $3\,\mathrm{h}$. The IPs in our covering approach then are all solved in well below $1\,\mathrm{s}$ by Gurobi, and still within at most $87\,\mathrm{s}$ by SCIP. We remind that the precomputations were designed with flexibility in mind, see Sect. 2.2.2, i.e., they can be reused for many cases, see our series 2 for a demonstration. Hence, for the actual optimization process used for control updates, only the solving times of the IPs are

https://doi.org/10.5194/wes-2024-120
Preprint. Discussion started: 8 October 2024


(a) $20^\circ$, baseline: $62.05\,\mathrm{MW}$. (b) $20^\circ$, optimized: $62.98\,\mathrm{MW}$.

**Figure 6.** Detailed results for a $6 \times 3$ farm, wind speed of $11\,\mathrm{m\,s^{-1}}$ and wind direction $20^\circ$ (series 1, case 4) as shown in Table 3. Subfigure captions specify the wind direction, yaw control (baseline or optimized), and the resulting total power output of the farm.

relevant, which turned out to be so small that we can speak of real-time capable optimization (recall that yaw angles are only updated every $15\,\mathrm{min}$ or so in practice, see Sect. 1). Finally, we see that the suggested database of precomputated simulation results, see Sect. 2.2.4, is time-consuming to build but enables significant gains through optimization.

The main influence on the precomputation time is the turbines' number in the upstream section; the impact of the allowed number of yaw angles is smaller, cf. Sect. 4.4. The specific upstream sections in series $1$ are depicted in Fig. 3; those of cases $1$ and $2$ (i.e., wind directions $0^\circ$ and $5^\circ$) include three turbines and yield precomputation times of about half an hour (cf. Table 3), whereas those of cases $3$ and $4$ (i.e., $10^\circ$ and $20^\circ$) include four turbines and take about $3\,\mathrm{h}$. Four turbines (with seven possible yaw angles) result in $512$ yaw configurations (i.e., simulations), see the example in Sect. 2.2 and Eq. (4).

As assumptions and discretizations were employed to arrive at our IP model for the WFYP via the covering approach (CA), we *validate* it by comparing the computed power outputs with those obtained from a direct farm simulation with the baseline and optimized yaw configurations. To that end, we examine case $4$, but with a TI of $0\%$, see Fig. 6(a) for the layout: In the baseline setup, the maximal deviation of any single turbine is $0.02\,\mathrm{MW}$, and $0.05\,\mathrm{MW}$ in the optimized one. For the farm's total power in the baseline, our CA results in $63.52\,\mathrm{MW}$ (instead of $63.44\,\mathrm{MW}$) and in $64.54\,\mathrm{MW}$ (instead of $64.37\,\mathrm{MW}$) for the optimal setting; it indicates that the CA may slightly overestimate the achievable improvement over the baseline setup ($1.61\%$ instead of $1.45\%$). The deviations are limited (in baseline and/or optimization) to WTs $\{7, 12, 14, 15, \ldots, 18\}$ and are due to the size (or shape) of the upstream section, which influences the accuracy of the discretized, section-based WFYP model and therefore is a compromise in terms of accuracy and run time. For example, the covering section (based on upstream section Fig. 3(h)) anchored at WT 7 includes WT 2, but WT 1 is marginally outside, i.e., its influence on WT 7 is not considered; other deviations can be explained analogously. An increase in size of the upstream section would ameliorate the small model inaccuracy, but also significantly increase the precomputation time. Therefore, we accept small inaccuracies in the power output.

https://doi.org/10.5194/wes-2024-120
Preprint. Discussion started: 8 October 2024


eawe european academy of wind energy
Open Access WIND ENERGY SCIENCE DISCUSSIONS

As the improvements over the baseline are significantly larger than the deviations from full simulation, the optimal solutions of our covering approach will likely still be optimal for a model using increased upstream sections, though suboptimality is technically possible. The IP solver run times themselves indicate that handling larger sections should not pose an issue.

### 4.2 Flexibility and scalability of the covering approach to solve the WFYP

In series 2, we examine the influence of the farm size on the solving time of the WFYP, illustrating the practical scalability of our covering approach vis-à-vis the theoretical hardness of the problem (cf. Prop. 2.1). Furthermore, we also conduct an experiment with a mix of active and inactive turbines to show our method's practical flexibility. These experiments also demonstrate the reusability of precomputed simulation results for section configurations, see Sect. 2.2. In particular, the experimental design for series 2 (i.e., wind direction of $20°$ and speed of $11\,\mathrm{m\,s^{-1}}$) is the same as in case 4 of series 1 (case 1.4, for short), see Table 3, i.e., we reuse case 1.4 precomputations (originally for a $6\times3$ farm) for $7\times3$, $8\times3$ and so on.

For the first experiments, we only change the size of the farm (i.e., 7 to $9\times3$) and monitor the workload of the IP solvers, see Table 5. The optimization consistently improves the farm's total power output by roughly $2\%$. The impact of the farm size on the IP solving time emphasizes the complexity of the WFYP (see Prop. 2.1): the solver SCIP quickly takes an impractically long time—it could be stopped after $5\,\mathrm{min}$ resulting in a duality gap of, e.g., $1.06\%$ for $9\times3$; Gurobi is faster by orders of magnitude remaining well below one minute in all cases considered here, despite an (apparently exponential) increase in run time with the farm size. Thus, in light of the low yaw sampling rate of, e.g., $15\,\mathrm{min}$ (see Sect. 1), our covering approach enables real-time WFYP optimization even for farms with more turbines than explored here.

As mentioned before, our covering approach also supports the reuse of precomputed data if some turbines become inactive (or do not exist at all), i.e., it allows a flexible reaction to a shutdown for maintenance or an unexpected reason. Case 2.5 illustrates the influence of deactivated turbines on the IP solver behavior and the achievable power gains. Again, we reuse case 1.4, cf. Table 3, but now in the setting depicted in Fig. 7, i.e., WTs $\{2,5,6,9,12\}$ are inactive in comparison to Fig. 6(a)/(b). The "thinning out" of the farm reduces the solving time of the IP from $86.85\,\mathrm{s}$ to $0.02\,\mathrm{s}$ with SCIP, and from $0.35\,\mathrm{s}$ to below $0.01\,\mathrm{s}$ using Gurobi. The absolute (relative) improvement of the total power output over the baseline is $0.92\,\mathrm{MW}$ $(1.77\%)$. Thus, the WFYP optimization still yields about $2\%$ gain in generated power, while a lower farm density in terms of active turbines leads to notably shorter IP solving times.

### 4.3 Impact of different wind speeds

In series 3, we again consider a $6\times3$ farm and a wind direction of $20°$, and evaluate WFYP solutions for different wind speeds. The baseline and optimization results of 6, 11 and $12\,\mathrm{m\,s^{-1}}$ are reported in Table 6 ($11\,\mathrm{m\,s^{-1}}$ repeats case 1.4) and Fig. 8. In all cases, the optimization increases the farm's total power output by about $2\%$. The maximum of the precomputation time is about $3\,\mathrm{h}$ and of the IP solving time about $87\,\mathrm{s}$ with SCIP, or under $1\,\mathrm{s}$ with Gurobi.

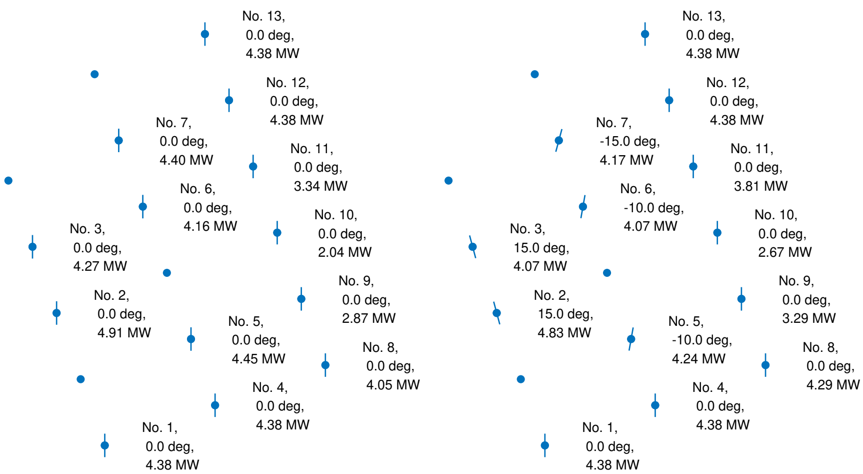
Additional experiments with $14\,\mathrm{m\,s^{-1}}$ and $15\,\mathrm{m\,s^{-1}}$ show the limit of interesting speeds from power output optimization perspective for the present example: for the $15\,\mathrm{m\,s^{-1}}$ ($14\,\mathrm{m\,s^{-1}}$) case, all turbines output (almost) their maximum power already in the baseline yaw setting—nevertheless, optimization could still be meaningful if mechanical loads were actually included in

https://doi.org/10.5194/wes-2024-120
Preprint. Discussion started: 8 October 2024




**Table 5.** Data and results to illustrate scalability and complexity of the WFYP (series 2). All cases have a wind direction of 20° and a speed of $11\,\mathrm{m\,s^{-1}}$ in common. In all cases, the IP optimality gap is 0.00%. (Case 2.1 repeats case 1.4.)

| | wind farm | | IP solver | | | | total power output | | |
|---|---|---|---|---|---|---|---|---|---|
| case | layout | # covering sections | # variables | # constraints | solving time (in s) SCIP | solving time (in s) Gurobi | baseline (in MW) | optimized (in MW) | improvement rel. (in %) |
| 1 | $6\times3$ | 8 | 1430 | 2866 | 86.85 | 0.35 | 62.05 | 62.98 | 1.50 |
| 2 | $7\times3$ | 9 | 1773 | 3553 | 364.94 | 1.26 | 71.89 | 73.01 | 1.56 |
| 3 | $8\times3$ | 10 | 2216 | 4240 | 1760.04 | 3.62 | 81.73 | 83.04 | 1.60 |
| 4 | $9\times3$ | 11 | 2459 | 4927 | 9071.55 | 11.25 | 91.56 | 93.07 | 1.77 |




(a) Baseline: $52.03\,\mathrm{MW}$.   (b) Optimized: $52.95\,\mathrm{MW}$.

**Figure 7.** Detailed results for a $6\times3$ farm in which some turbines are inactive (case 2.5 with wind speed of $11\,\mathrm{m\,s^{-1}}$ and direction of 20°). Subfigure captions specify yaw control (baseline or optimized) and the resulting total power output of the farm.

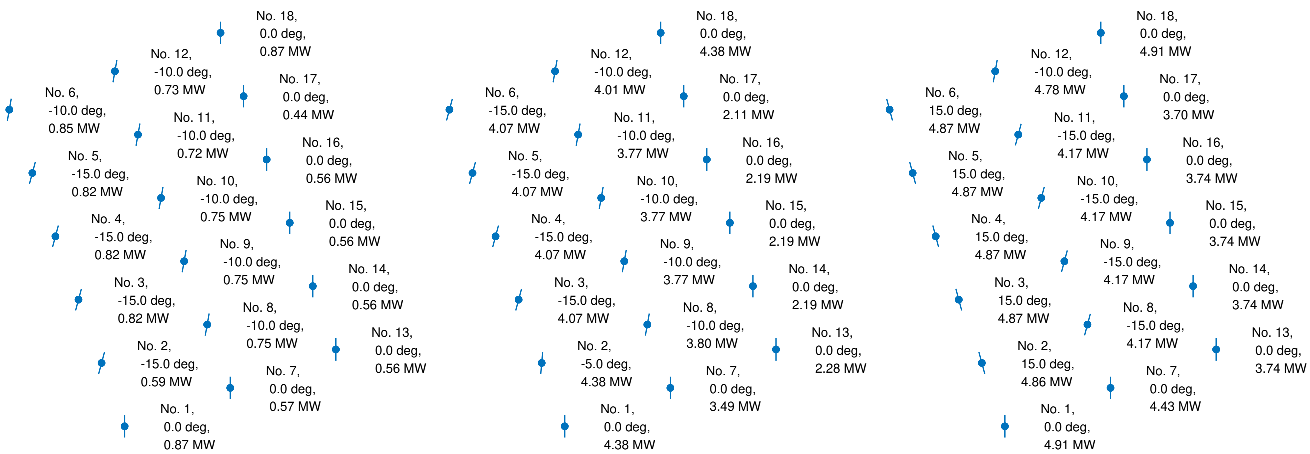
the optimization objective. Identifying the corresponding maximum-output wind speed threshold (w.r.t. the baseline) for each wind direction can avoid unnecessary precomputations associated with wind speeds beyond the levels at which optimization can have an effect. Moving on, in an analysis together with additional experiments, i.e., for wind speeds $\{6,8,10,11,\ldots,14\}\,\mathrm{m\,s^{-1}}$, we observe that turbines in a row (apart from those at the farm borders) appear to typically have identical optimal yaw angles (in the same experiment), i.e., the optimal yaw angles for WTs 2 to 5 are the same (except for WT 2 at 8 and $11\,\mathrm{m\,s^{-1}}$), as are those for WTs 8 to 11 (except for WT 11 at $8\,\mathrm{m\,s^{-1}}$). This is likely due to the grid layout. In addition, the optimal yaw angles in the first and second row always differ from 0, except for WTs 1 and 7. So, the result would be the same if we had computed the complete section configuration (i.e., upstream section) without the possibility of 0 in the first and second row; probably, we only need 0 to detect the greedy wind speed threshold. This may provide opportunities to reduce the running times of both precomputations and the resulting IP solution process.

https://doi.org/10.5194/wes-2024-120
Preprint. Discussion started: 8 October 2024


eawe european academy of wind energy
Open Access WIND ENERGY SCIENCE DISCUSSIONS

**Table 6.** Data and results to illustrate the impact of wind speeds (series 3). All cases have a $6 \times 3$ farm layout and a wind direction of $20°$ in common. In all cases, the IP optimality gap is $0.00\%$. Detailed results are in Fig. 8. (Case 3.2 repeats case 1.4.)

| | wind farm | | IP solver | | total power output | | |
|---|---|---|---|---|---|---|---|
| case | wind speed $U_{\mathrm{ave}}$ (in $\mathrm{m\,s^{-1}}$) | precomputa-tion time (in h) | solving time (in s) SCIP | Gurobi | baseline (in MW) | optimized (in MW) | improvement rel. (in %) |
| 1 | 6 | 2.4 | 27.61 | 0.42 | 12.31 | 12.57 | 2.11 |
| 2 | 11 | 2.9 | 86.85 | 0.35 | 62.05 | 62.98 | 1.50 |
| 3 | 12 | 3.1 | 8.06 | 0.19 | 76.99 | 78.73 | 2.26 |




(a) $6\,\mathrm{m\,s^{-1}}$, optimized: $12.57\,\mathrm{MW}$. (b) $11\,\mathrm{m\,s^{-1}}$, optimized: $62.98\,\mathrm{MW}$. (c) $12\,\mathrm{m\,s^{-1}}$, optimized: $78.73\,\mathrm{MW}$.

**Figure 8.** Detailed optimization results for series 3 as shown in Table 6. Subfigure captions specify the wind speed and the farm's total power output. Baseline turbines' power output (in MW) is sorted by rows: case 1, row 1 (WTs 1: 0.87; 2: 0.63; 3–6: 0.87), row 2 (7: 0.50; 8–11: 0.68; 12: 0.76), row 3 (13–16: 0.52; 17: 0.41; 18: 0.87)—case 2, row 1 (1: 4.38; 2: 4.40; 3–6: 4.27), row 2 (7: 3.34; 8–11: 3.46; 12: 4.16), row 3 (13–16: 2.10; 17: 2.04; 18: 4.38)—case 3, row 1 (1: 4.91; 2: 4.93; 3–6: 4.94), row 2 (7: 4.55; 8–11: 4.52; 12: 4.84), row 3 (13–16: 2.95; 17: 3.19; 18: 4.91).

### 4.4 Modifying the yaw angle range or discretization

In series 4, we deviate from the yaw angle discretization ($[-15°, 15°]$ in $5°$ steps, i.e., seven yaw angles) used so far: first (case 4.1), we expand the range and use $[-40°, 40°]$, again in $5°$ steps (giving 17 yaw angles), and second (case 4.2), we reconsider $[-15°, 15°]$ but with a finer discretization of $2.5°$ (yielding 13 yaw angles). To this end, we consider a $6 \times 3$ farm with a wind speed of $11\,\mathrm{m\,s^{-1}}$ and a direction of $5°$ in case 4.1 and $20°$ in case 4.2, i.e., we compare to cases 1.2 and 1.4, respectively.

The extension of allowed yaw angles in case 4.1 increases the precomputation time from $0.5\,\mathrm{h}$ (cf. Table 3, case 2) to $1.7\,\mathrm{h}$, while the number of turbines in the upstream section is unaffected, i.e., three. The IP consists of 1734 variables and 2896

https://doi.org/10.5194/wes-2024-120
Preprint. Discussion started: 8 October 2024


constraints; SCIP needs $0.82\,\mathrm{s}$ and Gurobi $0.80\,\mathrm{s}$ to solve it. As optimal result, all turbines in the first row are set to $30°$ and each output $3.37\,\mathrm{MW}$, which is significantly lower than in case $1.2$ ($4.14\,\mathrm{MW}$) due to their higher yaw offset. This enables a similar power output of the second row despite the larger yaw offsets of $35°$ there, i.e., $2.40\,\mathrm{MW}$ (in comparison to $2.45\,\mathrm{MW}$ at case $1.2$). Finally, this enables a significantly increased power output of the downstream-most turbines (with $0°$), i.e., $3.49\,\mathrm{MW}$ (instead of $1.97\,\mathrm{MW}$). The total farm output is $55.57\,\mathrm{MW}$ (instead of $51.41\,\mathrm{MW}$ in case $1.2$), i.e., compared to the baseline, the total power output is increased by $26.76\%$ (instead of $17.27\%$). The non-use of the new limits of $\pm 40°$ shows that at the extreme yaw angle settings, the power loss at a turbine would exceed the gain at turbines downstream.

Case $1.4$ is particularly interesting for finer discretization (case $4.2$) as not all turbines attain extremal yaw angles $\pm 15°$. However, all optimal yaw angles in case $4.2$ in fact remain the same as in case $1.4$, though the finer discretization increases the precomputation time from $2.9\,\mathrm{h}$ (cf. Table 3, case $4$) to $14.6\,\mathrm{h}$ (of course, still with four turbines in the upstream section). The IP in case $4.2$ consists of $8972$ variables and $17950$ constraints, for which SCIP needs $58.28\,\mathrm{h}$ to solve while Gurobi takes only $34.13\,\mathrm{s}$. In any case, it is unnecessary to solve the WFYP with arbitrarily finely discretized yaw angles because of the uncertainty of incident wind conditions, see Stanley et al. (2022) (referenced in Sect. 1.2.1).

## 5 Concluding remarks

We formulated the wind farm yaw problem mathematically, established its computational complexity and developed a covering approach to solve it with integer programming. Building on a number of simulation results that can be precomputed at any time before the need for yaw control arises, the method is efficient in practice in spite of the problem's strong $\mathcal{NP}$-hardness and inapproximability. In particular, we fully expect even very long precomputation times (e.g., months) to be acceptable as the simulations can easily be run for various wind scenarios while the farm is not yet operational. Given the envisioned database, our covering approach efficiently delivers optimal yaw control using a state-of-the-art IP solver like Gurobi. The solution is even globally optimal under some mild assumptions as discussed in Sect. 1.2.1. In addition, it enables tackling even farms with many turbines. We demonstrated the performance of our approach with several proof-of-concept examples that illustrate its effectiveness, flexibility and scalability, particularly through the reuse of precomputations if we enlarge the farm orthogonally to the wind direction or deactivate turbines. On the other hand, the enlargement in wind direction increases the upstream section and therefore the number of turbines inside, which mainly increases the precomputation time. As our covering approach is a superordinate model, the simulation is interchangeable. The utilized dynamic models (for wakes and turbines) possibly made the precomputation more expensive than necessary when only focusing on the power output. Presuming that the influence of a substitution on the resulting optimal yaw angles is small, the precomputation time would decrease considerably. Finally, it might be helpful to solve the WFYP for the associated farms by our covering approach for a variety of wind directions and speeds to recognize structures whose exploitation reduces the computational effort in precomputation or simplifies the WFYP itself. Solving the WFYP for the associated farms by covering approach in all considered wind scenarios should provide knowledge to reduce the computational effort.

https://doi.org/10.5194/wes-2024-120
Preprint. Discussion started: 8 October 2024


*Code and data availability.* For simulation, we used the MATLAB software package WinFaST. This company-internal software is not publicly available, but based on known methods, as described in Sect. 3.1. As our own optimization framework is presently entwined with WinFaST and hence not a stand-alone program, we have not made it publicly available at this time. Nevertheless, in Sect. 2, we provide a detailed description of the problem, the novel covering approach, integration/utilization of simulation results and the WFYP formulation as an integer program. As data to supplement the article, we provide the integer programs (`lp`-files) and corresponding solver log files for each case of our series of experiments. The data is available at Zenodo: https://doi.org/10.5281/zenodo.13837557.

## Appendix A: Complexity of the wind farm yaw problem

This section addresses the computational complexity of the wind farm yaw problem from a theoretical viewpoint; we assume a basic knowledge of mathematical complexity theory and refer to Garey and Johnson (1979) for a detailed introduction. Using the basic black-box IP formulation (1) to (3) of the wind farm yaw problem (WFYP; cf. Sect. 2.1.1), we show that the WFYP is strongly $\mathcal{NP}$-hard (Theorem A.3) and even hard to approximate (Corollary A.4). These two results together yield Proposition 2.1 as stated in Sect. 2.1.2.

We use the well-known strongly $\mathcal{NP}$-complete Hamiltonian Circuit (HC) problem, see, e.g., (Garey and Johnson, 1979, problem GT37), for our proof.

**Definition A.1** (Hamiltonian Circuit Problem (HCP))**.** Let an undirected graph $G=(V,E)$ on $n$ vertices, $V:=\{v_1,\ldots,v_n\}$, be given. Does $G$ contain a Hamiltonian circuit, i.e., a subset of edges $\mathcal{H}=\big\{\{v_{\pi(1)},v_{\pi(2)}\},\{v_{\pi(2)},v_{\pi(3)}\},\ldots,\{v_{\pi(n-1)},v_{\pi(n)}\},\{v_{\pi(n)},v_{\pi(1)}\}\big\}\subset E$ for a permutation $\langle v_{\pi(1)},v_{\pi(2)},\ldots,v_{\pi(n)}\rangle$ of $V$?

For clarity, we also explicitly state the decision version of the WFYP problem.

**Definition A.2** (WFYP, decision version (WFYP-DEC))**.** Let the index set $T$ of wind turbines in a wind farm, its layout $\mathcal{L}$, the index set $\Gamma_i:=\{1,\ldots,n_{\Gamma,i}\}$ of admissible yaw angles for each turbine $i\in T$, the WFYP objective function $f_\omega^\Sigma$ as defined for (the (black-box) IP formulation of) the WFYP, and a number $F\in\mathbb{R}$ be given. Does there exist a feasible yaw configuration for the given farm (i.e., exactly one yaw angle per turbine) such that, for the associated binary yaw angle assignment vector $x$, $f_\omega^\Sigma(x)\geq F$?

Recall that the assignment vector $x\in\{0,1\}^{n_{\Gamma 1}+\cdots+n_{\Gamma n_{\mathrm{WT}}}}$, with the entry in position $\sum_{\ell=1}^{i-1} n_{\Gamma\ell}+j$ being denoted by $x_{i,j}$ and having value $1$ if and only if the $i$-th turbine is set to the $j$-th yaw angle from among the respective admissible set $\Gamma_i$. In the following, we will also use some additional notation: We denote the so-called triangular numbers by $\Delta_n:=\binom{n+1}{2}=\frac{n^2+n}{2}$ and abbreviate the set of the first $N$ triangular numbers as $\Delta(N):=\{\Delta_1,\Delta_2,\ldots,\Delta_N\}$. Also, for a natural number $n\in\mathbb{N}=\mathbb{Z}_{>0}$, we denote $[n]:=\{1,2,\ldots,n\}$.

We are now prepared to prove the $\mathcal{NP}$-hardness and inapproximability of the WFYP(-DEC).

**Theorem A.3.** *The wind farm yaw problem is strongly $\mathcal{NP}$-hard.*

*Proof.* We show hardness for the decision version WFYP-DEC, which directly implies hardness for the optimization version (WFYP, as defined in Sect. 2), cf. Garey and Johnson (1979). To that end, we reduce from the strongly $\mathcal{NP}$-complete HCP.

https://doi.org/10.5194/wes-2024-120
Preprint. Discussion started: 8 October 2024


eawe european academy of wind energy
Open Access WIND ENERGY SCIENCE DISCUSSIONS

Let $G = (V, E)$ be an arbitrary HCP instance, and denote $n = |V|$. We can assume w.l.o.g. that $n \geq 2$, every vertex has degree at least 2, and that $G$ consists of a single connected component (otherwise, the answer to the HCP is trivially "no"). We construct an instance $(T, \mathcal{L}, \{\Gamma_i\}_{i\in T}, f^{\Sigma}_{\omega}, F)$ of the WFYP-DEC as follows:

We set the number of turbines to $n_{\mathrm{WT}} := \Delta_n + 1$ and identify the turbines by their index, i.e., $T := [\Delta_n + 1]$. The turbines are arranged in a triangle-like layout $\mathcal{L}$ (on a regular grid)[3] defined by the following "row" sets:

$$R_1 := \{\Delta_1\} = \{1\},$$
$$R_2 := \{\Delta_2, 2\},$$
$$R_3 := \{\Delta_3, 5, 4\},$$
$$\vdots$$
$$R_n := \{\Delta_n, \Delta_n - 1, \ldots, \Delta_{n-1} + 1\},$$
$$R_{n+1} := \{\Delta_n + 1\}.$$

Furthermore, each turbine $i \in T$ is given the same set of admissible yaw angles $\Gamma_i := \Gamma := [n]$, so $n_{\Gamma,i} = |\Gamma_i| = n$ for all $i \in T$. Finally, we set $F := n$, and define the terms $f_{\omega,i,j}(x)$ of the simulation function $f^{\Sigma}_{\omega}(x) := \sum_{i=1}^{n_{\mathrm{WT}}} \sum_{j=1}^{n_{\Gamma,i}} f_{\omega,i,j}(x)\, x_{i,j} = \sum_{i=1}^{\Delta_n+1} \sum_{j=1}^{n} f_{\omega,i,j}(x)\, x_{i,j}$ as

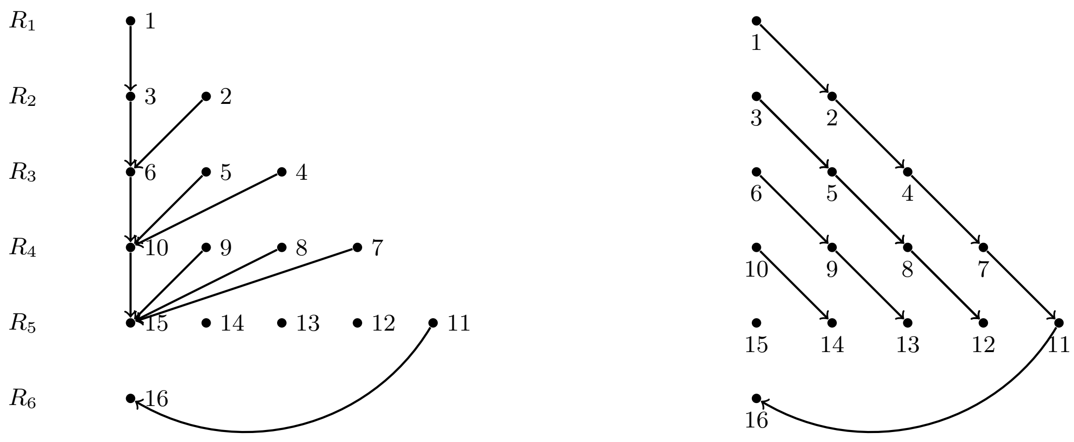
$$f_{\omega,i,j}(x) := \begin{cases} 0 & \text{if } i = 1 (= \Delta_1) \text{ and } j \in [n], \\ 1 & \text{if, for some } k \in [n],\ i = \Delta_k > 1,\ x_{\Delta_{k-1},\ell} = 1 \text{ for some } \ell \text{ such that } \{v_\ell, v_j\} \in E, \\ & \quad \text{and } j \in [n] \setminus \{q \in [n] : \sum_{s \in R_{k-1}} x_{s,q} > 0\} \\ 0 & \text{if, for some } k \in [n],\ i \in R_k \setminus \big(\Delta(n) \cup \{\Delta_n + 1\}\big) \text{ and } x_{i-k+1,j} = 1, \\ 1 & \text{if } i = \Delta_n + 1,\ x_{\Delta_{n-1}+1,j} = 1, \text{ and } \{v_j, v_\ell\} \in E \text{ for } j \text{ such that } x_{\Delta_n,\ell} = 1, \\ 1 - n & \text{otherwise.} \end{cases} \tag{A1}$$

For a (feasible) overall yaw configuration of the farm as determined by $x$, the components $f_{\omega,i,j}(x)$ of $f^{\Sigma}_{\omega}(x)$ specify the objective contribution (or profit, for short) incurred by turbine $i$ using the yaw angle (indexed by) $j$. Specifically, the first case in Eq. (A1) defines a zero profit for any arbitrary yaw angle assignment to the first turbine. The second case then yields a profit of 1 if a turbine that corresponds to a triangular number $\Delta_k > 1$ has been assigned (according to the input yaw configuration $x$) a yaw angle $j$ that was not chosen for any turbine in the previous row (set $R_{k-1}$) and is such that for the yaw angle $\ell$ chosen for turbine $\Delta_{k-1}$, the edge $\{v_\ell, v_j\}$ exists in $G$. (For example, supposing turbines $\Delta_1$ and $\Delta_3$ are assigned yaw angles $\ell \in [n]$ and $j$, respectively, then the setting for turbine $\Delta_3$ yields a profit of 1 only if $\{v_\ell, v_j\} \in E$ and the $j$-th angle was not selected for any turbine in the previous row, which in this case translates to $j \neq \ell$.) The third case means that using yaw angle $j$ for

[3]In fact, the precise layout does not matter, since all implications regarding the resulting wake influences are "hidden" in the black-box function (i.e., in the practical application, handled within the simulation framework); the same holds for the exogenously given (arbitrary but fixed) wind speed and direction.

https://doi.org/10.5194/wes-2024-120
Preprint. Discussion started: 8 October 2024

eawe european academy of wind energy
Open Access WIND ENERGY SCIENCE DISCUSSIONS




(a) (b)

**Figure A1.** Visualization of black-box function dependencies in the reduction from HCP to WFYP-DEC, exemplified for graph with $n = 5$ nodes. Arcs in (a) represent the dependencies in the constructed farm (i.e., which turbines bear influence on which others) for the first two cases in Eq. (A1), while arcs in (b) represent those for the remaining cases in Eq. (A1). Note that actual function values depend on input $x$.

any turbine $i \in R_k \setminus (\Delta(n) \cup \{\Delta_n + 1\}) = \{\Delta_{k-1}+1, \dots, \Delta_k - 1\}$ (with respect to some $k \in [n]$) incurs zero profit if this yaw angle $j$ was used at turbine $i - k + 1$ (which belongs to row set $R_{k-1}$, since $\Delta_{k-2} + 1 \le i - k + 1 \le \Delta_{k-1}$). The penultimate case yields a unit profit in the special case that the turbine is $i = \Delta_n + 1$ and the assigned yaw angle $j$ is also used by turbine $\Delta_{n-1} + 1$, provided that the edge $\{v_j, v_\ell\} \in E$ for $\ell$ being the yaw angle assigned to turbine $\Delta_n$. Finally, the last case sets the function value to $1 - n$ for all other configurations. Fig. A1 illustrates the dependency structure of the function.

This completes the construction of a WFYP-DEC instance $(T, \mathcal{L}, \{\Gamma_i\}_{i \in T}, f^{\Sigma}_{\omega}, F)$ from the input HCP instance $G$. Note that the reduction (dimensions, all arithmetic operations and constructed numbers) clearly requires only polynomial time and space with respect to the size of the input; in particular, the objective function can be evaluated in $\mathcal{O}(n^3)$, since $|T| = n_{\mathrm{WT}} = \Delta_n + 1 \le n^2$ and $n_{\Gamma_i} = n$ for all $i \in T$. (In fact, it can easily be seen that our reduction allows the "strongly" part of $\mathcal{NP}$-hardness to carry over from the HCP, cf. Garey and Johnson (1979).)

It remains to show that the given graph contains a Hamiltonian circuit $\mathcal{R}$ if and only if the constructed WFYP-DEC instance $(T, \mathcal{L}, \{\Gamma_i\}_{i \in T}, f^{\Sigma}_{\omega}, F)$ admits a solution $x$ with objective value $f^{\Sigma}_{\omega}(x) \ge F = n$.

To that end, first assume that $\hat{x} \in \{0,1\}^{(\Delta_n+1)n}$ is a feasible solution for WFYP-DEC (so every turbine is assigned exactly one yaw angle) with objective value $f^{\Sigma}_{\omega}(\hat{x}) \ge F$. Since by construction, only $n$ turbine yaw settings can possibly incur a profit of 1 each (and all others at most zero), $f^{\Sigma}_{\omega}(\hat{x}) = F = n$ does in fact hold, which also implies that the last case in the function definition (A1) never occurs (since otherwise, $f^{\Sigma}_{\omega}(\hat{x}) \le 1 - n + n = 1 < n$ would hold—a contradiction). In particular, tracing the functional dependencies with regard to which yaw angle assignments incur which costs for subsequent turbines (in the "cascading" row sets), we can conclude that no yaw angle is chosen twice among the turbines $1, \Delta_2, \Delta_3, \dots, \Delta_n$. Moreover, due to the first two cases in Eq. (A1) (and since $\hat{x}$ represents a feasible yaw configuration), and because every turbine has

https://doi.org/10.5194/wes-2024-120
Preprint. Discussion started: 8 October 2024


the same set of $n$ admissible yaw angles, it holds that each yaw angle is chosen exactly once for this set $\{1, \Delta_2, \Delta_3, \ldots, \Delta_n\}$ of turbines. Furthermore, note that by definition, any yaw setting $j$ for turbines $i = \Delta_{k-1} + 1$, $k = 2, \ldots, n$, incurs a profit of either $1 - n$ or 0, but that since $f^{\Sigma}_{\omega}(\hat{x}) = n$, the respective settings prescribed by $\hat{x}$ in fact all yield zero profit. Thus, these costs necessarily arise from the third case in Eq. (A1), which means that turbines $\Delta_1 + 1, \Delta_2 + 1, \ldots, \Delta_{n-1} + 1$ all have the same yaw angle as turbine 1. Consequently, by the fourth case in the definition, the yaw configurations chosen for turbines $\Delta_1 = 1$ and $\Delta_n + 1$ are also identical.

We can now construct a Hamiltonian circuit in $G$ from this WFYP solution $\hat{x}$: Starting at vertex $v_{p_1} \in V$, where $p_1 \in [n]$ is the yaw angle chosen for turbine $\Delta_1 = 1$, we visit the other $n - 1$ nodes in the order prescribed by the yaw angles selected for the turbines $\Delta_1, \ldots, \Delta_n$, and finally moving from the last node back to $v_{p_1}$. Indeed, this traversal produces the tour $\hat{\mathcal{R}} = \{\{v_{p_1}, v_{p_2}\}, \{v_{p_2}, v_{p_3}\}, \ldots, \{v_{p_{n-1}}, v_{p_n}\}, \{v_{p_n}, v_{p_1}\}\}$, which shows that a "yes" answer for the constructed WFYP-DEC instance yields a "yes" answer for the original HCP instance.

For the converse direction, let a circuit $\hat{\mathcal{R}} = \{\{v_{\pi(1)}, v_{\pi(2)}\}, \ldots, \{v_{\pi(n-1)}, v_{\pi(n)}\}, \{v_{\pi(n)}, v_{\pi(1)}\}\}$ be a "yes" certificate for the given HCP instance $G$. Then, we can derive a solution $\hat{x}$ with cost $f^{\Sigma}_{\omega}(\hat{x}) = n = F$ for the constructed WFYP-DEC instance from $\hat{\mathcal{R}}$ as follows: For turbines 1, $\Delta_2$, $\Delta_3$, ..., $\Delta_n$, $\Delta_n + 1$, we respectively select the yaw angles corresponding to the indices of the vertices in the order prescribed by the tour $\hat{\mathcal{R}}$, starting (and ending) at $v_{p_1} = v_{\pi(q)}$ for some $q \in [n]$, i.e., we set $\hat{x}_{1,\pi(q+1)} = \hat{x}_{\Delta_2,\pi(q+2)} = \cdots = \hat{x}_{\Delta_n,\pi(q+n)} = \hat{x}_{\Delta_n+1,\pi(q)} = 1$ (yielding total profit $0 + n \cdot 1 = n$) while the entries corresponding to these turbines and the respective remaining yaw angles are all set to zero. For the remaining turbines, we pick yaw angles that incur no negative profits, i.e., for any $k = 2, \ldots, n$, turbine $i \in R_k \setminus (\Delta(n) \cup \{\Delta_n + 1\})$ is assigned the same yaw angle as turbine $i - k + 1$, respectively, all with profit 0. Since this way, every turbine is assigned exactly one yaw angle, $\hat{x}$ indeed describes a feasible yaw configuration and by construction, its objective function value corresponds to $f^{\Sigma}_{\omega}(\hat{x}) = n = F$. This shows that the constructed WFYP-DEC instance also has a "yes" answer, which completes the proof. □

We remark that the above construction could easily be adapted so that only non-negative terms can occur in the objective[4], as would be the case in our application when focusing solely on power generation. Note also that, due to the generality of the black-box function $f^{\Sigma}_{\omega}$ in WFYP-DEC it is unclear whether one could always find a *rational* certificate of an arbitrary "yes" instance, so containment in the complexity class $\mathcal{NP}$ (and thus, $\mathcal{NP}$-completeness) remains open. However, more importantly, we can slightly modify the proof of Theorem A.3 to obtain the following inapproximability result.

**Corollary A.4.** *There is no polynomial-time $\alpha$-approximation algorithm for WFYP, for any $\alpha \leq 1$, unless $\mathcal{P} = \mathcal{NP}$.*

*Proof.* Revisiting the construction from the proof of Theorem A.3 we modify the function values in Eq. (A1) to 0, $\delta$, $\delta$, $\delta$ and $-\varepsilon$ for the five cases, respectively, which then establishes the existence of a Hamiltonian circuit in $G$ if and only if there is a feasible WFYP solution with value $\frac{1}{2}(n^2 + n - 2)\delta$. Let $\varepsilon := \frac{1}{2}(n^2 + n - 4)$ and $\delta := 4(\Delta_n + 1)\varepsilon = n^4 - 3n^2 + 6n - 8$. Then, if the original HCP instance was a "yes"-instance, any non-optimal solution of the constructed WFYP instance has solution

[4] To that end, we can replace the profit for the third and fifth case in Eq. (A1) by $\varepsilon > 0$ and 0, respectively. Then, the construction instance has a solution of value $n + \varepsilon(n^2 - n - 2)/2$ if and only if $G$ is Hamiltonian.

value at most $\frac{1}{2}(n^2+n-4)\delta-\varepsilon$.[5] Now suppose there exists a polynomial-time $\alpha$-approximation algorithm for some arbitrary $1/(n^2+n-1)<\alpha\leq 1$. Since any non-optimal solution (for a WFYP instance constructed from a HCP "yes"-instance) has value at most $\frac{1}{2}(n^2+n-4)\delta-\varepsilon$, but the $\alpha$-approximation algorithm outputs a solution with value at least $\alpha\cdot(n^2+n-2)\delta/2$, it can only be the case that the solution computed by the algorithm is non-optimal if

$$\alpha\left(\frac{n^2+n-2}{2}\right)\delta\leq\left(\frac{n^2+n-4}{2}\right)\delta-\varepsilon \quad\Leftrightarrow\quad \alpha\leq\frac{n^2+n-4}{n^2+n-2}-\frac{2\varepsilon}{n^2+n-2}=1-\frac{2}{n^2+n-2}-\left(1-\frac{2}{n^2+n-2}\right)=0,$$

which contradicts the prerequisite $\alpha>1/(n^2+n-2)>0$. Thus, the $\alpha$-approximation algorithm does, in fact, always yield an (optimal) solution of value $\alpha(n^2+n-2)\delta/2$ if and only if the input HCP instance was a "yes"-instance. It could therefore be used to decide the existence of a Hamiltonian circuit in polynomial time, contradicting $\mathcal{NP}$-hardness of the HCP. To see that this implies no polynomial-time $\alpha$-approximation can exists (provided $\mathcal{P}\neq\mathcal{NP}$) for *any* $0<\alpha\leq 1$, it suffices to observe that $1/(n^2+n-2)\to 0$ as $n\to\infty$. □

*Author contributions.* CK and SS deliberated the overarching research goals, wrote the corresponding project proposal, were responsible for the funding acquisition, and had the idea to tackle the wind farm yaw problem by discretization. FBü formulated the integer program of the covering approach, wrote the source code, and designed as well as performed the experiments. FBü and AT deliberated the experimental design and result analysis. FBe and AT construed and wrote the $\mathcal{NP}$-hardness and inapproximability proofs. FBe, FBü and AT wrote and revised the manuscript.

*Competing interests.* The authors declare that they have no conflict of interest.

*Acknowledgements.* FBü's research was funded by the Bundesministerium für Bildung und Forschung (BMBF, German Federal Ministry of Education and Research) under grant number 05M18MBA-MOReNet. FBe's research was funded by Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) through Priority Programme 1962. The authors would like to thank Robert Scholz (University of Heidelberg, Interdisciplinary Center for Scientific Computing), Ole Falkenberg (IAV GmbH; now with DNV Maritime Software GmbH), Bastian Ritter (Industrial Science GmbH; now with Wölfel Engineering GmbH + Co. KG.) and Axel Schild (IAV GmbH) for fruitful discussions about wind turbines.

[5] The same upper bound holds for the optimal value in case of HCP "no"-instances. Moreover, it is easy to see that $f_\omega^\Sigma(x)>0$ for any non-trivial feasible $x$, i.e., any $x$ that yields profit $\delta$ for at least one index pair $(i,j)$, which is trivially always achievable, so no reasonable algorithm would ever yield $f_\omega^\Sigma(x)\leq 0$.

https://doi.org/10.5194/wes-2024-120
Preprint. Discussion started: 8 October 2024


## References


Annoni, J., Gebraad, P. M. O., Scholbrock, A. K., Fleming, P. A., and van Wingerden, J.-W.: Analysis of axial-induction-based wind plant control using an engineering and a high-order wind plant model, Wind Energy, 19, 1135–1150, https://doi.org/10.1002/we.1891, 2016.

Annoni, J., Fleming, P., Scholbrock, A., Roadman, J., Dana, S., Adcock, C., Porté-Agel, F., Raach, S., Haizmann, F., and Schlipf, D.: Analysis of control-oriented wake modeling tools using lidar field results, Wind Energy Science, 3, 819–831, https://doi.org/10.5194/wes-3-819-2018, 2018.

Annoni, J., Bay, C., Johnson, K., Dall'Anese, E., Quon, E., Kemper, T., and Fleming, P.: Wind direction estimation using SCADA data with consensus-based optimization, Wind Energy Science, 4, 355–368, https://doi.org/10.5194/wes-4-355-2019, 2019.

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

Bay, C. J., Fleming, P., Doekemeijer, B., King, J., Churchfield, M., and Mudafort, R.: Addressing deep array effects and impacts to wake steering with the cumulative-curl wake model, Wind Energy Science, 8, 401–419, https://doi.org/10.5194/wes-8-401-2023, 2023.

Bestuzheva, K., Besançon, M., Chen, W.-K., Chmiela, A., Donkiewicz, T., van Doornmalen, J., Eifler, L., Gaul, O., Gamrath, G., Gleixner, A., Gottwald, L., Graczyk, C., Halbig, K., Hoen, A., Hojny, C., van der Hulst, R., Koch, T., Lübbecke, M., Maher, S. J., Matter, F., Mühmer, E., Müller, B., Pfetsch, M. E., Rehfeldt, D., Schlein, S., Schlösser, F., Serrano, F., Shinano, Y., Sofranac, B., Turner, M., Vigerske, S., Wegscheider, F., Wellner, P., Weninger, D., and Witzig, J.: The SCIP Optimization Suite 8.0, ZIB-Report 21-41, Zuse Institute Berlin, http://nbn-resolving.de/urn:nbn:de:0297-zib-85309, 2021.

Boersma, S., Doekemeijer, B., Vali, M., Meyers, J., and van Wingerden, J.-W.: A control-oriented dynamic wind farm model: WFSim, Wind Energy Science, 3, 75–95, https://doi.org/10.5194/wes-3-75-2018, 2018a.

Boersma, S., Keviczky, T., and van Wingerden, J.-W.: Stochastic Model Predictive Control: uncertainty impact on wind farm power tracking, https://www.researchgate.net/publication/329365115_Stochastic_Model_Predictive_Control_uncertainty_impact_on_wind_farm_power_tracking. Accessed: January 2019, 2018b.

Boersma, S., Doekemeijer, B. M., Siniscalchi-Minna, S., and van Wingerden, J.-W.: A constrained wind farm controller providing secondary frequency regulation: An LES study, Renewable Energy, 134, 639–652, https://doi.org/10.1016/j.renene.2018.11.031, 2019.

Churchfield, M., Lee, S., Moriarty, P. J., Martinez, L. A., Leonardi, S., Vijayakumar, G., and Brasseur, J. G.: A Large-Eddy Simulation of Wind-Plant Aerodynamics, in: 50th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition, Nashville, Tennessee, https://doi.org/10.2514/6.2012-537, 2012a.

Churchfield, M. J., Lee, S., Michalakes, J., and Moriarty, P. J.: A numerical study of the effects of atmospheric and wake turbulence on wind turbine dynamics, Journal of Turbulence, 13, 1–23, https://doi.org/10.1080/14685248.2012.668191, 2012b.

Dar, Z., Kar, K., Sahni, O., and Chow, J. H.: Windfarm Power Optimization Using Yaw Angle Control, IEEE Transactions on Sustainable Energy, 8, 104–116, https://doi.org/10.1109/TSTE.2016.2585883, 2017.

Dou, B., Qu, T., Lei, L., and Zeng, P.: Optimization of wind turbine yaw angles in a wind farm using a three-dimensional yawed wake model, Energy, 209, 118 415, https://doi.org/https://doi.org/10.1016/j.energy.2020.118415, 2020.

Ennis, B. L., White, J. R., and Paquette, J. A.: Wind turbine blade load characterization under yaw offset at the SWiFT facility, Journal of Physics: Conference Series, 1037, 052 001, https://doi.org/10.1088/1742-6596/1037/5/052001, 2018.

Fischetti, M.: Mathematical Programming Models and Algorithms for Offshore Wind Park Design, Ph.D. thesis, Technical University of Denmark, Copenhagen, https://orbit.dtu.dk/en/publications/9c92d369-d44d-4476-a4f0-85dbac7f362f, 2017.

https://doi.org/10.5194/wes-2024-120
Preprint. Discussion started: 8 October 2024


Fischetti, M.: On the optimized design of next-generation wind farms, European Journal of Operational Research, 291, 862–870, https://doi.org/10.1016/j.ejor.2020.10.048, 2021.

Fischetti, M. and Pisinger, D.: Mathematical Optimization and Algorithms for Offshore Wind Farm Design: An Overview, Business & Information Systems Engineering, 61, 469–485, https://doi.org/10.1007/s12599-018-0538-0, 2019.

Fleming, P. A., Gebraad, P. M. O., van Wingerden, J.-W., Lee, S., Churchfield, M., Scholbrock, A. K., Michalakes, J., Johnson, K., and Moriarty, P. J.: SOWFA Super-Controller: A High-Fidelity Tool for Evaluating Wind Plant Control Approaches, Tech. rep., National Renewable Energy Laboratory (NREL), https://www.osti.gov/biblio/1068611, proceedings of the EWEA 2013, 2013.

Fleming, P. A., Ning, A., Gebraad, P. M. O., and Dykes, K.: Wind plant system engineering through optimization of layout and yaw control, Wind Energy, 19, 329–344, https://doi.org/10.1002/we.1836, 2016.

Garey, M. R. and Johnson, D. S.: Computers and Intractability; A Guide to the Theory of NP-Completeness, W. H. Freeman & Co., USA, 1979.

Gebraad, P. M. O. and van Wingerden, J.-W.: A Control-Oriented Dynamic Model for Wakes in Wind Plants, Journal of Physics: Conference Series, 524, 012 186, https://doi.org/10.1088/1742-6596/524/1/012186, 2014.

Gebraad, P. M. O., Teeuwisse, F. W., van Wingerden, J.-W., Fleming, P. A., Ruben, S. D., Marden, J. R., and Pao, L. Y.: A data-driven model for wind plant power optimization by yaw control, in: 2014 American Control Conference, pp. 3128–3134, https://doi.org/10.1109/ACC.2014.6859118, 2014.

Gebraad, P. M. O., Fleming, P. A., and van Wingerden, J.-W.: Comparison of actuation methods for wake control in wind plants, in: 2015 American Control Conference (ACC), pp. 1695–1701, https://doi.org/10.1109/ACC.2015.7170977, 2015.

Gebraad, P. M. O., Thomas, J. J., Ning, A., Fleming, P. A., and Dykes, K.: Maximization of the annual energy production of wind power plants by optimization of layout and yaw-based wake control, Wind Energy, 20, 97–107, https://doi.org/10.1002/we.1993, 2017.

Gurobi Optimization, LLC: Gurobi—Gurobi Optimizer Reference Manual, https://www.gurobi.com, accessed: December 2022, 2022.

Hallac, D., Leskovec, J., and Boyd, S.: Network Lasso: Clustering and Optimization in Large Graphs, in: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '15, pp. 387–396, Association for Computing Machinery, New York, NY, USA, https://doi.org/10.1145/2783258.2783313, 2015.

Hau, E.: Wind Turbines : Fundamentals, Technologies, Application, Economics, Springer, Berlin, 3rd ed. 2013 edn., https://doi.org/10.1007/978-3-642-27151-9, 2013.

Howland, M. F., Lele, S. K., and Dabiri, J. O.: Wind farm power optimization through wake steering, Proceedings of the National Academy of Sciences, 116, 14 495–14 500, https://doi.org/10.1073/pnas.1903680116, 2019.

Howland, M. F., Quesada, J. B., Martínez, J. J. P., Larrañaga, F. P., Yadav, N., Chawla, J. S., Sivaram, V., and Dabiri, J. O.: Collective wind farm operation based on a predictive model increases utility-scale energy production, Nature Energy, 7, 818–827, https://doi.org/10.1002/essoar.10510347.1, 2022.

Jensen, N. O.: A Note on Wind Generator Interaction, Tech. Rep. 2411, Risø National Laboratory, 1983.

Jonkman, J., Butterfield, S., Musial, W., and Scott, G.: Definition of a 5-MW Reference Wind Turbine for Offshore System Development, Tech. rep., National Renewable Energy Laboratory, https://doi.org/10.2172/947422, 2009.

Katic, I., Højstrup, J., and Jensen, N. O.: A Simple Model for Cluster Efficiency, in: EWEC'86. Proceedings. Vol. 1, edited by Palz, W. and Sesto, E., pp. 407–410, Raguzzi, A., 1987.

King, J., Fleming, P., King, R., Martínez-Tossas, L. A., Bay, C. J., Mudafort, R., and Simley, E.: Control-oriented model for secondary effects of wake steering, Wind Energy Science, 6, 701–714, https://doi.org/10.5194/wes-6-701-2021, 2021.

Kost, C., Shammugam, S., Jülch, V., Nguyen, H.-T., and Schlegl, T.: Levelized Cost of Electricity : Renewable Energy Technologies, Tech. rep., Fraunhofer Institute for Solar Energy Systems ISE, Freiburg, https://www.ise.fraunhofer.de/content/dam/ise/en/documents/publications/studies/EN2018_Fraunhofer-ISE_LCOE_Renewable_Energy_Technologies.pdf, 2018.

Meyers, J., Bottasso, C., Dykes, K., Fleming, P., Gebraad, P. M. O., Giebel, G., Göçmen, T., and van Wingerden, J.-W.: Wind farm flow control: prospects and challenges, Wind Energy Science, 7, 2271–2306, https://doi.org/10.5194/wes-7-2271-2022, 2022.

NREL: FLORIS. Version 3.4, https://github.com/NREL/floris, accessed: June 2023, 2023.

Platis, A., Siedersleben, S. K., Bange, J., Lampert, A., Bärfuss, K., Hankers, R., Cañadillas, B., Foreman, R., Schulz-Stellenfleth, J., Djath, B., Neumann, T., and Emeis, S.: First *in situ* evidence of wakes in the far field behind offshore wind farms, Scientific Reports, 8, Art. Nr.: 2163, https://doi.org/10.1038/s41598-018-20389-y, 2018.

Quick, J., King, J., King, R. N., Hamlington, P. E., and Dykes, K.: Wake steering optimization under uncertainty, Wind Energy Science, 5, 413–426, https://doi.org/10.5194/wes-5-413-2020, 2020.

Ritter, B., Schild, A., Feldt, M., and Konigorski, U.: The design of nonlinear observers for wind turbine dynamic state and parameter estimation, Journal of Physics: Conference Series, 753, 052 029, https://doi.org/10.1088/1742-6596/753/5/052029, 2016.

Ritter, B., Mora, E., Schlicht, T., Schild, A., and Konigorski, U.: Adaptive Sigma-Point Kalman Filtering for Wind Turbine State and Process Noise Estimation, Journal of Physics: Conference Series, 1037, 032 003, https://doi.org/10.1088/1742-6596/1037/3/032003, 2018.

Schrijver, A.: Theory of Linear and Integer Programming, John Wiley & Sons, Chichester, UK, 1986.

Siniscalchi-Minna, S., Bianchi, F. D., Ocampo-Martinez, C., Domínguez-García, J. L., and De Schutter, B.: A non-centralized predictive control strategy for wind farm active power control: A wake-based partitioning approach, Renewable Energy, 150, 656–669, https://doi.org/10.1016/j.renene.2019.12.139, 2020.

Spudić, V. and Baotić, M.: Fast coordinated model predictive control of large-scale distributed systems with single coupling constraint, in: 2013 European Control Conference (ECC), pp. 2783–2788, https://doi.org/10.23919/ECC.2013.6669391, 2013.

Stanley, A. P., Thomas, J., Ning, A., Annoni, J., Dykes, K., and Fleming, P. A.: Gradient-Based Optimization of Wind Farms with Different Turbine Heights, in: 35th Wind Energy Symposium, American Institute of Aeronautics and Astronautics, https://doi.org/10.2514/6.2017-1619, 2017.

Stanley, A. P. J., Bay, C., Mudafort, R., and Fleming, P.: Fast yaw optimization for wind plant wake steering using Boolean yaw angles, Wind Energy Science, 7, 741–757, https://doi.org/10.5194/wes-7-741-2022, 2022.

Talavera, M. and Shu, F.: Experimental study of turbulence intensity influence on wind turbine performance and wake recovery in a low-speed wind tunnel, Renewable Energy, 109, 363–371, https://doi.org/10.1016/j.renene.2017.03.034, 2017.

Thomas, J., Tingey, E., and Ning, A.: Comparison of two wake models for use in gradient-based wind farm layout optimization, in: 2015 IEEE Conference on Technologies for Sustainability (SusTech), pp. 203–208, IEEE, https://doi.org/10.1109/sustech.2015.7314347, 2015.

van Wingerden, J.-W., Fleming, P. A., Göçmen, T., Eguinoa, I., Doekemeijer, B. M., Dykes, K., Lawson, M. J., Simley, E., King, J., Astrain, D., Iribas, M., Bottasso, C. L., Meyers, J., Raach, S., Kölle, K., and Giebel, G.: Expert Elicitation on Wind Farm Control, Journal of Physics: Conference Series, 1618, 022 025, https://doi.org/10.1088/1742-6596/1618/2/022025, 2020.

Veers, P. S.: Three-dimensional wind simulation, Tech. rep., Sandia National Laboratories, 1988.

Vladislavleva, E., Friedrich, T., Neumann, F., and Wagner, M.: Predicting the energy output of wind farms based on weather data: Important variables and their correlation, Renewable Energy, 50, 236–243, https://doi.org/10.1016/j.renene.2012.06.036, 2013.

Zhang, P. Y., Romero, D. A., Beck, J. C., and Amon, C. H.: Solving wind farm layout optimization with mixed integer programs and constraint programs, EURO Journal on Computational Optimization, 2, 195–219, https://doi.org/10.1007/s13675-014-0024-5, 2014.