

Original Research

Wind Farm Layout Optimization Subject to Cable Cost, Hub Height, and a Feasible 3D Gaussian Wake Model Implementation

Carsten Croonenbroeck ^{1,*}, David Hennecke ²

1. Rostock University, Environmental Science, Justus-von-Liebig-Weg 7, 18059 Rostock, Germany;
E-Mail: carsten.croonenbroeck@uni-rostock.de

2. Rostock University, Geodesy and Geoinformatics, Justus-von-Liebig-Weg 6, 18059 Rostock, Germany; E-Mail: david.hennecke@uni-rostock.de

* **Correspondence:** Carsten Croonenbroeck; E-Mail: carsten.croonenbroeck@uni-rostock.de

Academic Editor: Andrés Elías Feijóo Lorenzo

Collection: [Wind Energy](#)

Journal of Energy and Power Technology
2024, volume 6, issue 1
doi:10.21926/jept.2401008

Received: February 06, 2024

Accepted: March 07, 2024

Published: March 15, 2024

Abstract

We address the Wind Farm Layout Optimization (WFLO) problem and tackle the optimal placement of several turbines within a specific (wind farm) area by incorporating additional aspects of an economically driven target function. With this, we contribute three refinements for WFLO research: First, while many research contributions optimize the turbines' locations subject to maximum energy production or energy efficiency, we instead pursue a strategy of maximizing a profit objective. This enables us to incorporate inner-farm wiring costs (underground cable installation). For this, we explore the impact of using MSTs (Minimum Spanning Trees) and adding junction (so-called "Steiner") points to the terrain plane. Second, while most research focuses on finding optimal x and y coordinates (i.e., address two-dimensional turbine placement), we also optimize the turbines' hub heights z. Third, we also provide a software implementation of the Gaussian wake model. The latter finds entrance to the open-source WFLO research framework that comes as package **wflo** for statistical software R. We find that taking wiring cost into account may lead to very different turbine



© 2024 by the author. This is an open access article distributed under the conditions of the [Creative Commons by Attribution License](https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium or format, provided the original work is correctly cited.

placements, however, increasing overall profit significantly. Allowing the optimizer to vary the hub heights may have an ambiguous impact on the wind farm profit.

Keywords

Wind farm layout optimization (WFLO); Gaussian wake model; wiring cost; hub height optimization; wind energy; R package

1. Introduction

Finding the optimal spatial arrangement of turbines' locations inside a wind farm is known in the literature as "Wind Farm Layout Optimization" (WFLO). Since turbine arrangements affect the wind wake pattern in the farm under investigation, downwind turbines are requested to avoid the wake-upwind turbines emit, as wake reduces wind speed and therefore, leaves less kinetic energy for the downwind turbine, rendering it less effective in energy harvesting. WFLO is a mathematically challenging task. Only minor changes in the location of one turbine change the entire problem space, i.e., the wake pattern all other turbines are facing when they are to be placed. WFLO is said to be *NP*-hard (non-deterministic polynomial-time hard).

Founding contributions on WFLO by [1] and [2] made restrictive assumptions such as that turbines can only be placed on a relatively coarse grid of possible locations (discrete domain rather than a continuous domain), flat terrain, given hub heights, unique wind directions and speeds, and others. Subsequent research successively relaxed these assumptions. For example, [3] discuss a continuous problem domain, while [4] propose methods for complex terrain situations, and [5] optimize with respect to hub heights.

We address three aspects of the broader WFLO scope that have found minor investigation in the literature. First, we incorporate underground cable cost into the problem. By its very nature, WFLO seeks to optimize the locations of turbines within a find farm area (only) dependent on wind direction and wake interference. However, the underground wiring of each of the turbines is an important factor in the installation expense: The wire itself must be capable of carrying huge amounts of electricity and must, therefore, be amply dimensioned, which is costly. Also, trenching the wires is extensive as well, depending on the soil properties. Each meter of underground wiring, therefore, increases installation costs. However, the layout optimization procedure usually ignores the demand for asshortaspossible wire lengths.

Second, we discuss a Gaussian-Jensen wake model. In WFLO, the Jensen wake model is quite popular because it is simple to comprehend and quick to compute. However, it has several downsides: It is a two-dimensional model that shows a sharp distinction between wind regimes "inside" and "outside" the wake of an upwind turbine. Mathematically, this means that the model is not differentiable at its borders. Typically, a Gaussian wake model is proposed to overcome that downside. Gaussian wake has been widely investigated in the literature. Still, an "out-of-the-box" usable implementation of the model is missing in many research and application fields. We, therefore, provide a feasible implementation in **wflo**.

Third, we take into account possible variations in the hub heights of turbines. Most WFLO approaches optimize the turbines' locations only regarding their x and y coordinates. Variations in

terrain heights are typically neglected as a simplification, assuming flat terrain. Also, deliberately varying the turbines' tower heights to reduce downwind wake influence on other turbines has been discussed in the literature before but is still frequently ignored to reduce complexity. We, therefore, show an implementation of possible optimizations regarding the turbines' z dimension.

In all cases, we work with **wflo** and either use or extend its built-in functionality. **wflo** is a recently published open-source package for the statistical software R. Package **wflo** (see [6]) provides a quality data set and refined functions to deal with WFLO problems. It implements the widely used Jensen wake model, provides an interface to all-purpose optimization procedures, and employs an economically based (profit-oriented) target function, as according to [7], optimizing energy yield or farm efficiency alone, as it is done in many research contributions, is not always a meaningful goal. Note that multi-objective targets may also overcome this as proposed by, e.g., [8].

After all, **wflo** provides a quasi-standard of test conditions and thus works as a framework that ensures comparability among research contributions. Our work can be conducted using **wflo**, ensuring maximum comparability throughout. The remainder of this article is structured as follows: Section 2 provides additional details, motivation, and context for the three topics discussed here. In Section 3 we discuss our methods, while Section 4 presents the results. In Section 5 we draw a brief conclusion.

2. Contributions and Context

This section provides details on the paper's contributions. We discuss cable cost considerations in WFLO, the three-dimensional Gaussian-Jensen wake model implementation, and how we consider hub heights during WFLO.

2.1 Cable Length Optimization

Regardless of the configuration of V turbines within a wind farm, these turbines necessitate connection to the main grid via underground cables. As underground wiring is expensive, the usual approach is to connect the turbines point by point and then select an interconnection point from which to deliver the electricity to the grid. Therefore, the inner-farm wiring problem is finding the shortest connection from point to point, reducing possibly unnecessary cable routes to a minimum. In graph theory, the turbine locations (two-dimensional cartesian coordinates, i.e., pairs of x and y) are called "vertices," while the cable connections between them are "edges." Vertices and their edges altogether form a "tree." The desired solution to the wiring problem is expressed by a minimum sum of lengths of the E edges, while all vertices must be connected to the tree (at least once; multiple connections are allowed). For example, [9] or [10] deal with this problem. [11] provide an overview.

A "minimum spanning tree" (MST) is the first solution. In an MST, the number of edges will be $E = V - 1$. Given an MST, no other way to connect the vertices can be found at which the total length of edges is less than that in the MST solution. A farm planner will always develop an MST for the inner-farm wiring plan for a given turbine setup unless constraints like land property rights, irregularly processable ground, or similar dictate otherwise. Computing an MST from a given number of vertices can be carried out relatively quickly using efficient algorithms by Boruvka, developed in 1926 (see [12] for a translation from Czech to English), [13] or [14].

In this paper, we combine the profit-based WFLO target function used in **wflo** with an MST constraint: During each iteration of the WFLO optimization procedure, we compute the total length of the edges of the MST for the temporary solution the respective iteration handles, weight that total length with its economic cost and add the resulting cost to the installation cost already handled by the target function. This way, we integrate the shortest route constraint into the single-objective target function so that the optimizer can provide a solution that generates maximum profitability, not only taking annual electricity production (AEP), sale price, and thus, revenue into account but also turbine installation cost and, prominently, wiring cost. The resulting turbine setup can be expected not necessarily to provide the shortest possible total edge length but to minimize edge length as a constraint, subject to total profit in general.

A second step to optimize wiring cost can be reducing total edge length beyond an MST solution. This can be accomplished by adding additional vertices to the tree. These vertices do not have to represent actual wind turbines. It may be sufficient to consider the additional points as junction points for the cable. The vertices that represent actual turbines are called “terminals.” In contrast, the additional junction point vertices are “Steiner” points, so-called after Swiss mathematician Jakob Steiner (see [15] for an English recapitulation of the original work). Figure 1 provides an example. In the left panel, the problem is stated: Three points, denoted A, B, and C, represent the terminals (the actual turbines) in two-dimensional space. The middle panel shows an MST solution. Two edges connect the three points, and no other connection can be found that connects the points using a shorter route. As the edge from A to B is 50 and the edge from B to C is 40, the total length is 90. The right panel adds a Steiner point D (a junction point) to the terminals, resulting in four vertices. Now, the edge lengths are 40 (A to D), 29 (C to D), and 18 (B to D), resulting in a sum of $87 < 90$.

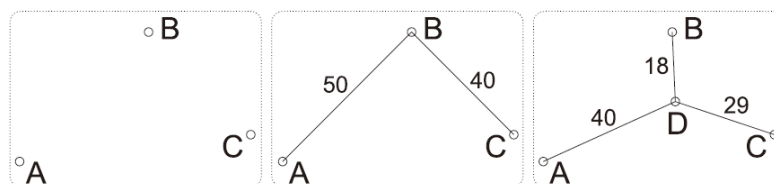


Figure 1 MST and Steiner solution to the inner-farm wiring problem. A, B, and C are “terminals” (e.g., actual turbine locations), while D is a Steiner point. Numbers denote edge lengths.

Minimum Steiner tree problems are *NP*-hard (unsolvable in non-deterministic polynomial time), as the optimal number of Steiner points and their optimal location must be determined. [16] describe the problem in detail, while [7] solve the problem for a given turbine setup using the open-source software package “GeoSteiner” (<http://www.geosteiner.com/>). Algorithms like the one implemented by GeoSteiner use approximations, and methods to find exact solutions are unknown. [17] provide an established approximation method, while [18] perform a more recent problem assessment.

As Steiner tree optimization is too complex for online WFLO, we provide a procedure that takes a given turbine setup (possibly a WFLO result) and based on the corresponding MST, iteratively adds Steiner points. The optimal locations of these points are subsequently approximated using a genetic algorithm (GA). The procedure keeps adding and optimizing Steiner points until no further

improvement in total edge length can be attained. The number of resulting Steiner points from that procedure is usually reasonably low compared to the number of terminals. This grants a relatively short computation time.

2.2 Gaussian Wake Model

The Jensen model is the most frequently used wake model in the WFLO literature. The reasons may be the model's simplicity, fast computability, and relatively high accuracy. However, the model has two major drawbacks: First, it is one-dimensional. This fact implies that within the wake cone, the model spans, at any given distance, the model's outcome - the wind speed - is constant, regardless of the angle between the point of interest and the wake-causing point, i.e., the turbine. In reality, however, the wake is more severe, close to the cone's core, and less severe towards its borders. This is connected to the second drawback, the fact that the model is non-continuous, i.e., the cone's borders imply a hard transition between "inside" and "outside" the wake, resulting in a "jump" in modeled wind speeds. Figure 2 shows both drawbacks in the usual top view of the wake model. Due to this step, the wind speeds are non-differentiable, and gradient-based optimizers can hardly be applied to the WFLO problem using Jensen's wake model.

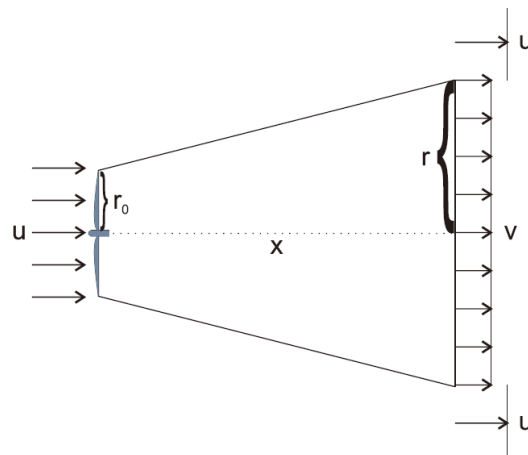


Figure 2 Typical Jensen wake model. Note that downwind speed v is proportional to distance x and that there is a strict "jump" between wind speeds "inside" and "outside" the wake cone.

Combining the model with a Gaussian smooth transition was first proposed by [19], resulting in the so-called Gaussian wake model. Several types of Gaussian wake models have been proposed. Most of them are based on a Jensen model, while [20] combine a Gaussian approach with the wake model by [21]. The Gaussian wake model addresses both inherent problems of the Jensen model: It models wind speed in a manner that allows for greater severity of the wake near the core while ensuring that the borders are smooth, continuous, and differentiable. Figure 3 shows the Gaussian Jensen model. [5] bring the model into the third dimension, allowing for free choice of x , y and z coordinates in positioning the downwind location for which the model is to return wind speed information. However, their model is computationally intense and, thus, likely not feasible in wind farm layout optimization runs, which require hundreds of millions of wake computations during their iterative nature of adjusting turbine locations. Here, we provide two Gaussian-Jensen wake

implementations that are already part of **wflo**. The first implementation, function `GenerateGauss()`, takes some time to pre-compute the three-dimensional space behind a turbine and stores the computed wind speeds in a tensor object. Users can choose the spatial resolution, which immediately impacts the pre-computation time. However, once pre-computations are done, a WFLO procedure can be speedy since wind speeds can be looked up in the tensor using the function `GaussWS()` instead of being computed. The second implementation, function `QuickGauss3D()`, computes the Gaussian-Jensen wind speed at one single location in three-dimensional space. This is still fast but slower than the look-up table approach. In exchange, this function does not require pre-computation.

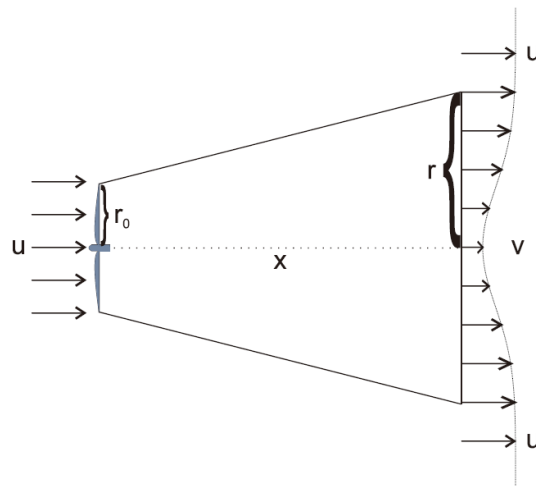


Figure 3 Gaussian Jensen wake model. Note that although there is a sharply determined border between wind speeds “inside” and “outside” the wake cone (lines at those borders), this is only to show the similarity to the Jensen model. However, the Gaussian model has no strict “jump” here.

2.3 Hub Heights Optimization

Virtually all real-world wind farms exhibit at least some variance in terrain elevation. Thus, although all turbines in the farm may use the same tower type, hub heights above overall ground may vary, resulting in differences in how turbines shed wind wakes on each other. Figure 4 sketches how hub height variation influences wake patterns in the farm.

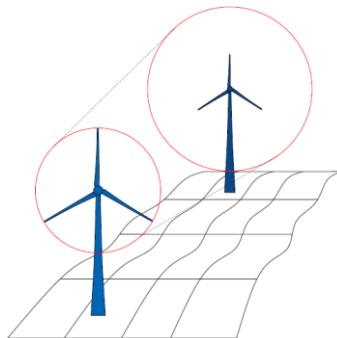


Figure 4 Upwind turbine (foreground) casting wake on downwind turbine (background) on a 3D topology.

Varying the hub heights of the turbines may seem an eligible way to avoid wake or reduce wake influence from full to only partial coverage.¹ Increasing hub height will result in higher and/or more constant wind speed (and thus, most likely, in higher energy yield) but also increases installation cost due to higher towers (material cost), additional cost for the installing tower crane, and others. Decreasing hub height has the reverse effect, but for pairs of turbines, it may be desirable to increase an upwind turbine's height and decrease the respective downwind turbine's height to have the downwind turbine avoid waking the upwind turbine shed into its general direction. In the literature, [22-25], among others, discuss this problem.

The topic results in the WFLO problem turning into a three-dimensional optimization problem, seeking optimal locations in terms of x and y coordinates for each turbine and z locations for the optimal hub heights. Generalizing the WFLO problem this way is relatively simple to do with **wflo**, mainly because the built-in function `PartialJensen()` is already prepared to compute partial wake coverage and different heights along the three-dimensional wake cone. Elevation data is available for the **wflo** wind farm terrain. Figure 5 presents the heights. The area possesses heights between 84 and 228 m.

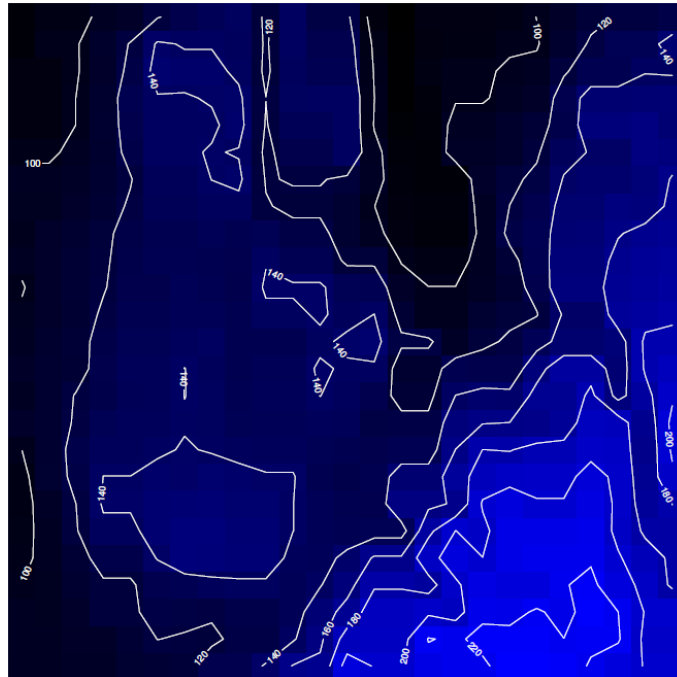


Figure 5 Elevation map in the wind farm area, contour lines show elevation in meters above sea level. Lighter shades of blue represent greater elevations, darker shades denote lower heights.

¹ The other way around may be worthwhile, i.e., avoid partial wake to obtain full wake. This may reduce energy output and asymmetric thrust on the downwind turbine, which is otherwise known to result in fatigue load. Our procedure can be applied to this type of problem as well.

3. Methodology

3.1 Cable Length Optimization

Our target function for optimization mimics the function Profit() from the **wflo** package. The function first checks for the validity of the turbine setup in terms of minimum distances between them and returns a large positive value if violated, which signals the optimizer that the current direction of optimization is suboptimal. The function then proceeds to minimize negative profit, which is equivalent to maximizing profit.² Profit is

$$\pi = \text{Revenue}(X, Y) - \text{Cost}(X, Y), \quad (1)$$

where X and Y are V -dimensional vectors containing the respective x and y coordinates of the V turbines, revenue depends on the sum of AEP at the given locations for each turbine, which is multiplied by an assumed sale price. **wflo** provides both internally. Also, for each of the turbine locations, the function computes whether that location is in the wake of any other turbines in the current setup. The function considers a Jensen wake model but also accounts for multiple wake and partial wake coverage. Given wake, the function computes a wind speed deduction factor. Wind turbine power output [W] can be expressed as the following

$$P = \frac{1}{2} C_p \rho A v^3, \quad (2)$$

where C_p is the power coefficient, ρ is air density [kg/m^3], A is rotor swept area [m^2] and v is wind speed [m/s]. As power output is proportional to the third power of wind speed ($P \propto v^3$), the target function multiplies AEP by the third power of the wind speed deduction factor.

Considering the cost, the function first utilizes the **wflo** internal Cost() function but then uses ComputeMST() from package **emstreeR** to compute the MST for the given setup. ComputeMST() uses the algorithm by Boruvka, which essentially operates by first subdividing the given tree into detached fractions and then iteratively reattaching them using the shortest possible routes. Given the MST, the total cable length is computed and multiplied by a cost factor [$\text{€}/\text{m}$]. As for that cost factor, we assume 300 $\text{€}/\text{m}$ for the cable and its installation cost (groundwork). This is a crude assumption based on several civil and underground engineering companies' proposals. Actual underground cable cost may depend on soil conditions, farm size, and market conditions. Using real-world cost data may refine the results.

Finally, revenue and cost are collected, and the negative difference between them is returned. We then use this function to optimize the locations of 20 turbines within the real-world area supplied by **wflo** and use function genoud() from package **rigenoud** by [26] to optimize. genoud() implements a GA, which is known to find satisfactory solutions, especially in the **wflo** literature. See [27] for an overview. Also, note that the workhorse functions of **wflo** are not limited to using GA for optimization. More recent optimization procedures, e.g., the ones presented by [8, 28-31] can also be plugged in.

² This is quite common since many optimizers are by default minimizers, so to maximize profit, it is equivalent to minimizing negative profit, which optimizers can handle more easily.

After optimization subject to MST, or for any given setup of V turbines in general, we first compute the respective MST. Based on that, we iteratively add Steiner points to the setup individually (outer loop). For each number of Steiner points, the inner loop optimizes the Steiner points' locations using `genoud()`. If, for any state of the outer loop, no improvement (i.e., no shorter route) can be found compared to the so-far best solution (i.e., the state of the former loop round), the function aborts and returns the best result found. Function `MakeSteiner()` is accessible in the supplemental material, together with a demonstration function call.

3.2 Gaussian Wake Model

The algorithm starts by computing downwind wake cone radius r [m] (see Figure 2). According to [32], for any given downwind distance x [m], the radius of the cone is based on the rotor radius r_0 [m] and the dimensionless factor α such that

$$r = r_0 + \alpha x, \quad (3)$$

where α depends on the hub height z_H [m] and the terrain roughness length z_0 [m]. The latter is usually assumed to be $z_0 = 0.1$, which is a fair estimate for typical on-shore sites. With

$$\alpha = \frac{0.5}{\log\left(\frac{z_H}{z_0}\right)}, \quad (4)$$

r can be computed immediately. In the three-dimensional setting discussed here, not only x is given (the distance downward the turbine), but also y and z . y denotes the perpendicular offset from the x axis (see Figure 2). Thus, it is assumed that $y = 0$ refers to a point on the $x - z$ plane and that y may take negative values. z is the height axis, starting at 0 on the ground and growing beyond the turbine's nacelle (hub height). The model processes y and z for any given x .

Typically, a single wind speed u_{Ref} [m/s], measured at a given reference height z_{Ref} , is given. This wind speed must be adjusted for the desired height above ground level. We do that using Hellmann's power law

$$u = u_{\text{Ref}} \cdot \left(\frac{z}{z_{\text{Ref}}}\right)^\beta, \quad (5)$$

where, according to empirical studies by [33], β usually takes values around 0.14. For bivariate Gaussian processing (computing wind speeds in the $y - z$ plane for a given x), the z -mean is assumed at the hub (z_H), also implying that the mean to y is 0:

$$\boldsymbol{\mu} = \begin{pmatrix} 0 \\ z_H \end{pmatrix}, \quad (6)$$

assuming Gaussian distribution with mean $\boldsymbol{\mu}$. As suggested by [5], assuming the squared rotor radius as the Gaussian standard deviation is reasonable. However, that implies one variance value at the hub and a different value at any given point down the x axis, as the Jensen model implies an increasing rotor swept area ("radius" r) down that axis. From that, we obtain the symmetric, positive definite variance-covariance matrix

$$\Sigma_1 = \begin{pmatrix} r^2 & 0 \\ 0 & r^2 \end{pmatrix} \quad (7)$$

at the point of interest, i.e. $\mathbf{X}_x = (y, z)^T$ given x , and

$$\Sigma_2 = \begin{pmatrix} r_0^2 & 0 \\ 0 & r_0^2 \end{pmatrix} \quad (8)$$

at the hub. Now, the Gaussian densities at these points are computed following

$$f(\mathbf{X}, \boldsymbol{\mu}, \Sigma) = \frac{1}{\sqrt{(2\pi)^2 |\Sigma|}} \exp\left(-\frac{1}{2}(\mathbf{X} - \boldsymbol{\mu})^T \Sigma^{-1}(\mathbf{X} - \boldsymbol{\mu})\right), \quad (9)$$

and using

$$\mathcal{D}_1 = f(\mathbf{X}_x, \boldsymbol{\mu}, \Sigma_1) \quad (10)$$

for the densities at the rotor as well as

$$\mathcal{D}_2 = f\left(\begin{pmatrix} 0 \\ z_H \end{pmatrix}, \boldsymbol{\mu}, \Sigma_2\right), \quad (11)$$

for the densities at distance x “behind” the rotor, we obtain the fraction of the densities as

$$\mathcal{D} = \frac{\mathcal{D}_1}{\mathcal{D}_2}. \quad (12)$$

The Jensen model computes the downwind speed v (see Figure 2) as a function of the undisturbed wind speed u as

$$v = u \left(1 - \frac{2}{3} \left(\frac{r_0}{ax + r_0}\right)^2\right). \quad (13)$$

Deriving concerning the *fraction* of these wind speeds yields $p = \frac{v}{u} \in (0,1)$, which determines the percentage amount of wake influence on the undisturbed wind speed as

$$p = 1 - \frac{2}{3} \left(\frac{r_0}{ax + r_0}\right)^2. \quad (14)$$

With that, we finally obtain

$$v_{x,y,z} = \mathcal{D} \cdot (1 - p) \cdot u. \quad (15)$$

3.3 Hub Heights Optimization

[34] report turbine installation costs for several different hub heights. We use their data and approximate a height-dependent cost function by

$$C(z_H) = 24,000 + 750 \cdot z_H, \quad (16)$$

where z_H again represents hub height and $C(\cdot)$ is the resulting cost in €. For example, heights of (100; 120; 140) m return (99,000; 114,000; 129,000) € installation cost, which is close to the reported values for these heights of (98,000; 116,000; 128,000) €.

However, not all turbine type/hub height combinations are feasible. As [35] points out, the clearance of the turbine blades above ground must be no less than 75 ft (23 m). Therefore, the rotor radius r_0 plus 23 m gives a lower threshold of hub height. As the default rotor radius in **wflo** is 45 m, a hub height must be at least $45 + 23 = 68$ m. Also, we consider a maximum hub height of 150 m since greater hub heights may begin to increase cost progressively instead of following our simple linear model.

The main challenge in using **wflo** for this type of three-dimensional optimization is that the entire **wflo** infrastructure is designed to accept a vector of interchanging x and y coordinates. To circumvent that, we define a global matrix object which contains three columns for x , y , and z . As the `Profit()` function conveniently allows us to replace the internal `Yield()` function with a user-defined one, we use this point to access the global object at each iteration of the optimizer, update the AEP information at each given turbine location by the height at that point using equation (5), and feed the adjusted yield back. We, therefore, require only a short wrapper function that wraps `Profit()` and accordingly updates the global data object. `Profit()` then calls `Yield()` and `Cost()` internally, where `Cost()` is a user-defined function that implements equation (16). The three functions (`Yield()`, `Cost()`, and the wrapper `HProfit()`) are available in the supplemental material, together with some code that sets up the domains for all variables, provides initial values, and calls the optimization run using `genoud()`, also exploiting the `genoud()` capability to utilize parallel computing by providing a standard R cluster object to it via the function call.

4. Results

We compare our optimization run subject to cable length to the built-in benchmark solution in **wflo**. That benchmark solution results from an extensive optimization using the built-in `Profit()` function. Details are provided by [36]. The benchmark encompasses 20 turbines located at high-yield (in terms of AEP) locations while minimizing wake influence. The wind farm area, located in the north-east of Germany, is a square of 5×5 km and contains pre-computed AEP data at a raster of 200×200 m resolution, resulting in 25×25 tiles. These tiles are depicted in Figure 6. Blue shades of the tiles show AEP values, where lighter blue represents higher values and darker shades depict lower values. Also, the figure contains AEP contour curves and an arrow field representing wind directions. Gold dots show the turbine locations. The depicted benchmark setup yields a total profit of 11,622,939 € per year.

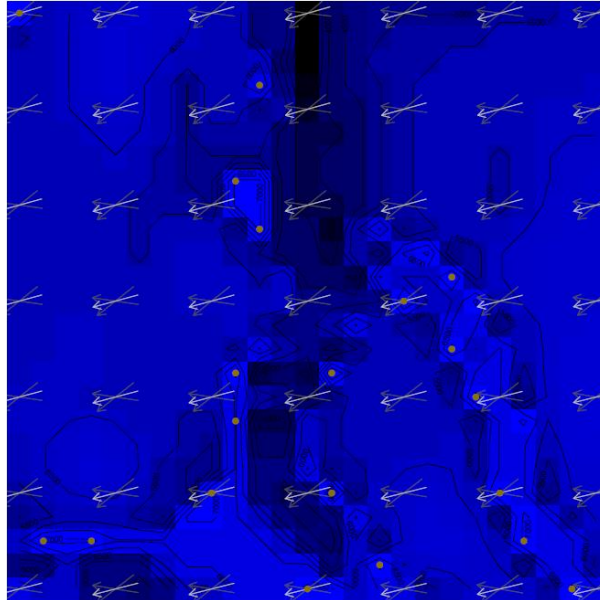


Figure 6 *wlfo* benchmark solution: Wind farm arrangement of 20 turbines, blue tiles, and contour curves represent AEP.

4.1 Cable Length Optimization

The benchmark setup's MST total route length is 15,220.4 m. Cable cost is not taken into account in the standard profit function but amounts to 4,566,119 € (228,306 € per year assuming a turbine's life span of 20 years and no interest), so the profit subject to cable cost remains 11,394,633 €. In contrast, the cable optimization result is depicted in Figure 7. Its pure AEP profit (i.e., not taking cable cost into account) is 11,603,227 €, i.e., 19,712 € below that of the benchmark solution. However, its route length is only 10,540.7 m (over 30% less than for the benchmark), resulting in a cable cost of 3,162,205 € (158,115 € yearly), so the profit taking cable cost into account is 11,445,112 € (50,478 € more than for the benchmark).

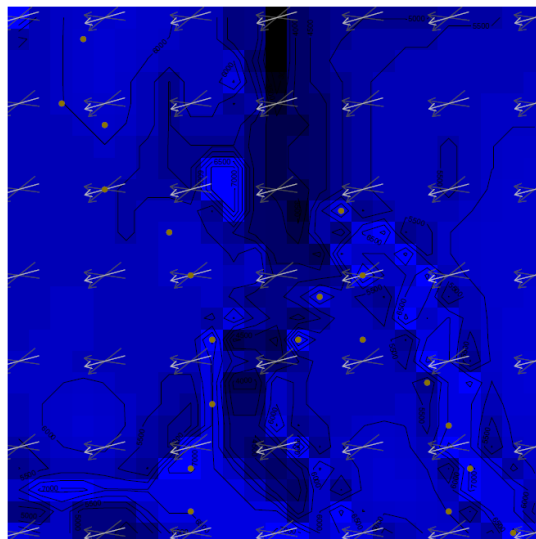


Figure 7 Wind farm arrangement of 20 turbines, optimized subject to cable length.

As an example for Steiner optimization, we again use the benchmark solution as a basis. Figure 8 shows the MST for that setup. We then run the iterative Steiner procedure and find an optimum for four added Steiner points. Figure 9 depicts the Steiner tree, where the Steiner points are drawn in blue. The total route length is reduced by 3% to 14,799.7 m, and the cable cost amounts to 4,439,905 € (221,995 € yearly), increasing total profit by 6,311 € compared to the benchmark.

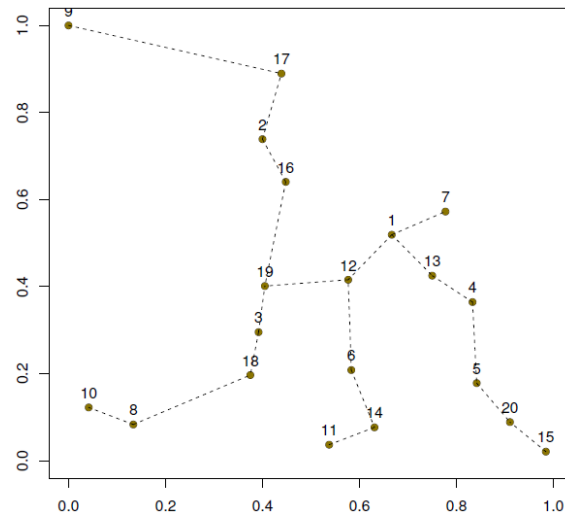


Figure 8 Minimum spanning tree for the built-in WFLO benchmark solution of **wifo**.

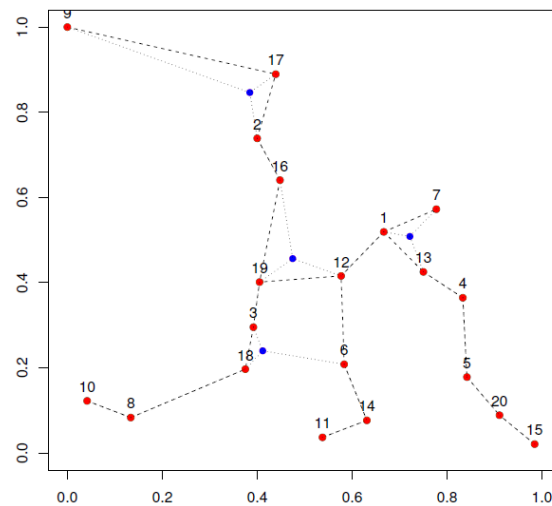


Figure 9 MST (red) and Steiner points (blue) for the built-in WFLO benchmark solution of **wifo**.

When applied to the cable cost considering the WFLO procedure result, we find an optimum of two added Steiner points, reducing route length from 10,540.7 m to 10,459.5 m. This results in a cable cost of 3,137,838 € (156,892 € yearly), a reduction of 1,223 €. Figure 10 shows the respective MST (red) and Steiner tree (blue).

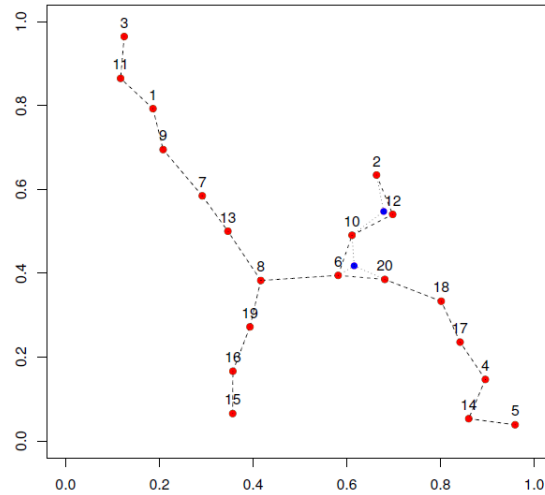


Figure 10 Steiner tree optimization after cable length WFLO optimization.

4.2 Gaussian Wake Model

For our example, we use an incoming wind speed of $u_{\text{Ref}} = 8$ m/s. We assume that this measured wind speed is given at a reference height of $z_{\text{Ref}} = 10$ m, the hub height is $z_H = 100$ m, and the terrain roughness length is $z_0 = 0.1$ m. Figure 11 shows the resulting Gaussian wind speed heat map and contour curves. The figure represents the “view” down the x axis, i.e., it depicts the wind speeds on the $y - z$ plane at $x = 0$. The maximum height depicted is $z = 180$ m. The lower row exhibits wind speeds close to zero, representing heights only a little more than z_0 .

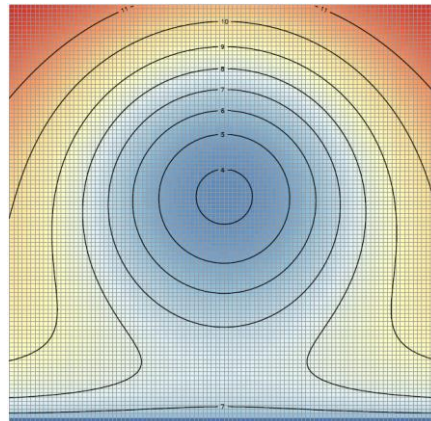


Figure 11 Gaussian wake model, view down the x -axis (turbine in the back).

Figure 12 shows a view from the side, i.e., down the y axis ($x - z$ plane), where the turbine is located on the left. Note that $y = 0$ means that the $x - z$ plane is perpendicular to the tower. Figure 13, finally, presents a view from the top towards the ground, i.e., down the z axis ($x - y$ plane). Again, the turbine is on the left. The selected height is $z_H = 100$ m.

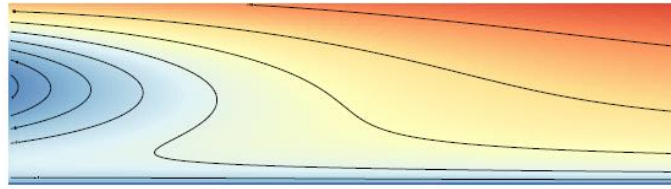


Figure 12 Gaussian wake model, view down the y-axis (from the side at the scene, turbine on the left).

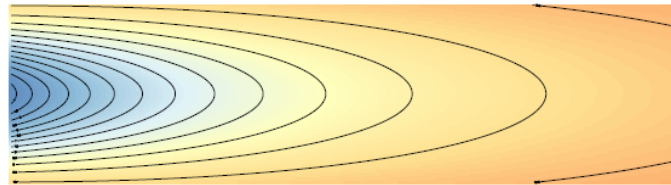


Figure 13 Gaussian wake model, view down the z-axis (from the top towards the bottom, turbine on the left).

It should be noted that while function `QuickGauss3D()` computes wind speed exactly at the specified location, `GaussWS()` looks up wind speed at the available location closest to the one specified. Therefore, both functions may return slightly different values. For example, `GaussWS (Gauss, 100, 1,100)` returns a value of 7.03462, while `QuickGauss3D (100, 1,100)` returns 7.035851. However, a tensor resolution of $500 \times 1,000 \times 1,000$ for `GenerateGauss()` (see section 2.2) will ensure that the deviation between both functions is never greater than 0.1 m/s.

4.3 Hub Heights Optimization

As height optimization adds another dimension to the already heavy load computations in WFLO, the optimizer consumes much computation time. For the results presented here, we have an Intel Core i7, 3.4 GHz, compute for 1,045 hours, 39 minutes, and 38 seconds (or almost 44 days).

Figure 14 presents the turbine arrangement in the usual top view, while Figure 15 shows the resulting heights. Interestingly, there seem to be two “regimes” of optimal heights, as nine turbines are set to 137 m, and six turbines attain heights of 126 m. To rule out a coincidence, four additional computation runs (again consuming between 40 and 43 days of computation time but computed simultaneously due to multiprocessing capabilities) were performed, resulting in virtually the same setup. The result, therefore, seems to be robust. A first conclusion might be that turbine heights are adjusted to allow turbines to evade the wake influences. However, at a second glance, that is not the case. As Figure 16 shows the resulting wake cones, almost no turbines are influenced by wake. Turbine 20 is (scarcely and only partially) struck by the wake turbines 1, 8, and 17 shed. Turbine 5 is in the wake of turbine 8 (full wake), see Figure 17, where P1 is turbine 8, P2 is turbine 5, the distance between A' and B' denotes the rotor diameter of turbine 5, CC marks the cone center of the wake, and A, B, C, D are the corner points of the wake cone. As these turbines are 4,475.3 meters apart, the Jensen model computes a wake diameter of 354.8 meters at the position of turbine 5. Turbine 8, the upwind turbine, is optimized to a hub height of 137.3 m, while turbine 5 (downwind) is set to 125.1 m. The total height difference, however, is also dependent on the terrain elevation. At the location of turbine 8, the elevation is 218 m; at turbine 5, it is 104 m. Therefore, the overall hub height of turbine 8 is $137 + 218 = 355$ m above sea level, while for turbine 5, it is 229 m, so the

difference is 126 m. From that, it can be seen that turbine 5 could hardly evade the wake of turbine 8 by adjusting its height. The wake cone shed by turbine 8 encompasses the rotor disc of turbine 5 and goes up beyond 500 meters, see Figure 18. Decreasing the hub height of turbine 5 to the lower bound (68 meters) could have at least positioned large fractions of the rotor disc outside the cone. However, lower wind speeds at lower heights would have been the downside (contrary to lower installation cost). However, as it is more than 4 km downwind, the Jensen penalty is only minor (less than one percent wind speed deduction); evading that weak wake would not be reasonable.

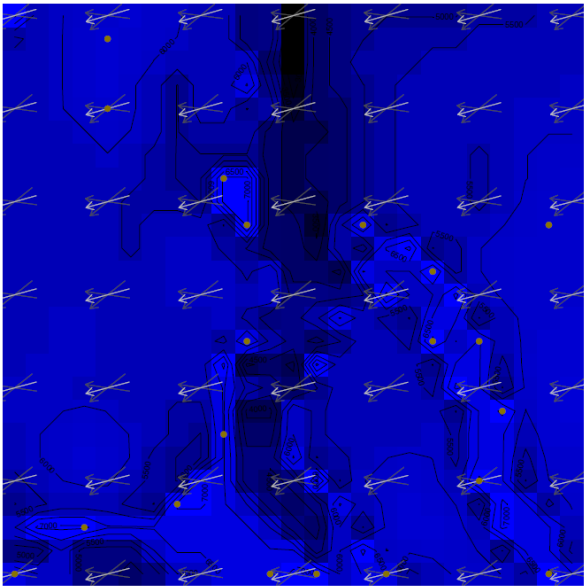


Figure 14 Optimal turbine arrangement, considering hub heights.

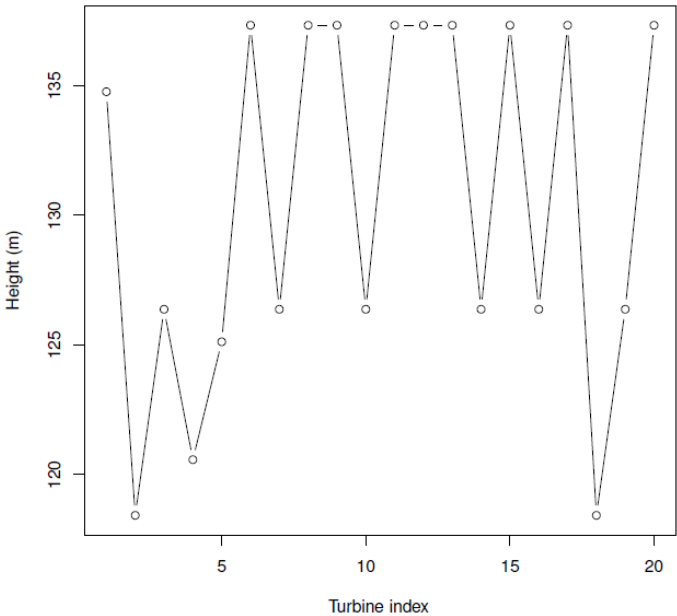


Figure 15 Optimal hub heights of turbines.

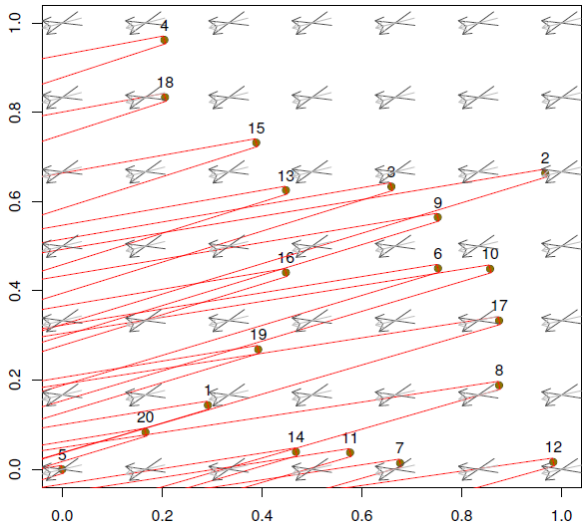


Figure 16 Wake penalizers for hub heights optimization result. Light arrows represent the wind direction field, while dark arrows depict wind directions plus/minus one standard deviation.

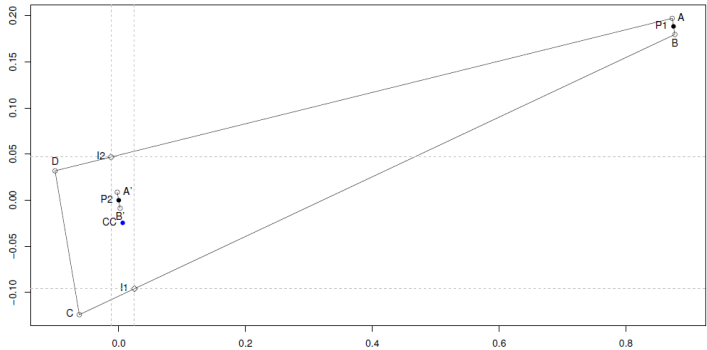


Figure 17 Wake situation between turbine 8 (upwind) and turbine 5 (downwind), top view.

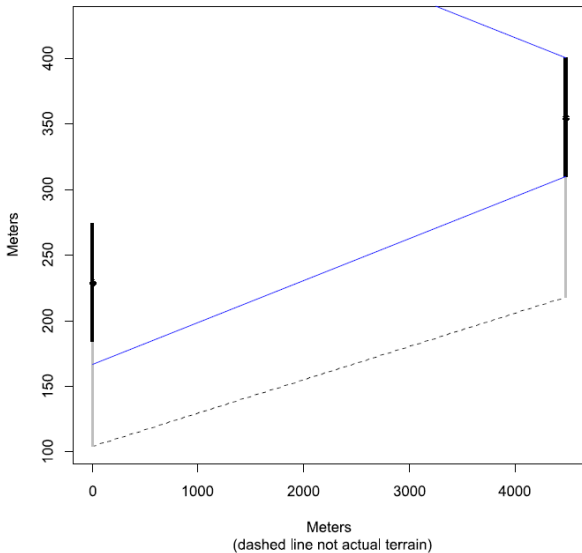


Figure 18 Wake situation between turbine 8 (upwind) and turbine 5 (downwind), side view.

Essentially, evading the wake by dodging below or above is not possible in many cases due to large wake cones, and also, it is not worth accepting the disadvantages. Still, considering hub heights for the WFLO procedure is important, as the robust results of different hub heights suggest. Heights around 126 m and 137 m may represent favorable trade-offs between installation cost and energy harvesting potential. Applying the hub heights as suggested by the result yields a yearly profit (i.e., after taking increased installation cost into account) of 11,616,543 € while ignoring the heights and using all 100 m hub heights returns only 11,491,234 €, i.e., 125,309 € less.

5. Conclusion

This article deals with three WFLO-related topics: cable cost, hub heights, and Gaussian-Jensen wake. Taking cable cost into account during the WFLO procedure leads to significantly different turbine setups than attained if the wake situation is essentially the only constraint in the optimization. The same holds when hub heights are considered. These results show that optimizing AEP or wind farm efficiency may only miss out on additional economic improvement of the solutions found if cable cost or hub heights are ignored.

Providing a Gaussian-Jensen wake model is not new to the literature. However, fostering its usage by implementing software to use the model may help researchers to use gradient-based optimizers in the future, which is virtually impossible using the non-differentiable Jensen model. Our implementation is fast and, while not as accurate as more realistic computational fluid dynamic models (CFD), will be sufficient to replace the Jensen model successively.

As a general insight, the article shows that the WFLO problem can (and should) be extended to incorporate more relevant aspects in the real world than just placing turbines on a two-dimensional plane. Varying wind regimes across the farm area, varying terrain heights, wiring costs, and more are subject to vivid research for a reason.

Our research presented here leaves room for improvement. For example, our assumption of underground cable cost of 300 €/m is crude and can be refined. Provided additional soil condition information, more sophisticated models could even be used to develop detailed cost-optimized cable routes. The hub heights optimizer could be generalized by enabling the procedure to use different types of turbines. Larger turbines are more expensive but can access better wind conditions. Higher towers can also carry larger turbines with greater rotor-swept areas due to longer rotor blades. This also demands larger and more expensive generators and transformers but is usually worth the additional cost.

Author Contributions

Carsten Croonenbroeck: Conceptualization, methodology, software, text writing. **David Hennecke:** Data curation, figures, methodology, text writing.

Competing Interests

The authors have declared that no competing interests exist.

References

1. Mosetti G, Poloni C, Diviacco B. Optimization of wind turbine positioning in large windfarms by means of a genetic algorithm. *J Wind Eng Ind Aerodyn.* 1994; 51: 105-116.
2. Grady S, Hussaini M, Abdullah MM. Placement of wind turbines using genetic algorithms. *Renew Energy.* 2005; 30: 259-270.
3. Wang F, Liu D, Zeng L. Study on computational grids in placement of wind turbines using genetic algorithm. *Proceedings of the 2009 World Non-Grid-Connected Wind Power and Energy Conference*; 2009 September 24-26; Nanjing, China. Piscataway, NJ: IEEE.
4. Hu W, Yang Q, Yuan Z, Yang F. Wind farm layout optimization in complex terrain based on CFD and IGA-PSO. *Energy.* 2024; 288: 129745.
5. Sun H, Yang H. Study on an innovative three-dimensional wind turbine wake model. *Appl Energy.* 2018; 226: 483-493.
6. Croonenbroeck C, Hennecke D. wflo: A new standard for wind farm layout optimization in R [Internet]. Vienna, Austria: The R Foundation; 2020. Available from: <https://cran.r-project.org/web/packages/wflo/vignettes/wflo.pdf>.
7. Wu Y, Zhang S, Wang R, Wang Y, Feng X. A design methodology for wind farm layout considering cable routing and economic benefit based on genetic algorithm and geosteiner. *Renew Energy.* 2020; 146: 687-698.
8. Yu X, Lu Y. Reinforcement learning-based multi-objective differential evolution for wind farm layout optimization. *Energy.* 2023; 284: 129300.
9. Herbert-Acero JF, Probst O, Réthoré PE, Larsen GC, Castillo-Villar KK. A review of methodological approaches for the design and optimization of wind farms. *Energies.* 2014; 7: 6930-7016.
10. Kirchner-Bossi N, Porté-Agel F. Realistic wind farm layout optimization through genetic algorithms using a gaussian wake model. *Energies.* 2018; 11: 3268.
11. Pérez-Rúa JA, Cutululis NA. Electrical cable optimization in offshore wind farms-a review. *IEEE Access.* 2019; 7: 85796-85811.
12. Nešetřil J, Milková E, Nešetřilová H. Otakar borůvka on minimum spanning tree problem translation of both the 1926 papers, comments, history. *Discrete Math.* 2001; 233: 3-36.
13. Kruskal JB. On the shortest spanning subtree of a graph and the traveling salesman problem. *Proc Am Math Soc.* 1956; 7: 48-50.
14. Prim RC. Shortest connection networks and some generalizations. *Bell Syst Tech J.* 1957; 36: 1389-1401.
15. Blåsjö V. Jakob steiner's systematische entwicklung: The culmination of classical geometry. *Math Intell.* 2009; 31: 21-29.
16. Lu CL, Tang CY, Lee RCT. The full Steiner tree problem. *Theor Comput Sci.* 2003; 306: 55-67.
17. Berman P, Karpinski M, Zelikovskiy A. 1.25-Approximation algorithm for steiner tree problem with distances 1 and 2. In: *Algorithms and data structures WADS 2009 lecture notes in computer science.* Berlin, Heidelberg: Springer; 2009. pp. 86-97.
18. Cohen N, Nutov Z. Approximating steiner trees and forests with minimum number of steiner points. *J Comput Syst Sci.* 2018; 98: 53-64.
19. Bastankhah M, Porté-Agel F. A new analytical model for wind-turbine wakes. *Renew Energy.* 2014; 70: 116-123.

20. Tao S, Kuenzel S, Xu Q, Chen Z. Optimal micro-siting of wind turbines in an offshore wind farm using frandsen-gaussian wake model. *IEEE Trans Power Syst.* 2019; 34: 4944-4954.
21. Frandsen S. On the wind speed reduction in the center of large clusters of wind turbines. *J Wind Eng Ind Aerodyn.* 1992; 39: 251-265.
22. Song Z, Zhang Z, Chen X. The decision model of 3-dimensional wind farm layout design. *Renew Energy.* 2016; 85: 248-258.
23. MirHassani SA, Yarahmadi A. Wind farm layout optimization under uncertainty. *Renew Energy.* 2017; 107: 288-297.
24. Abdulrahman M, Wood D. Investigating the power-COE trade-off for wind farm layout optimization considering commercial turbine selection and hub height variation. *Renew Energy.* 2017; 102: 267-278.
25. Syed AH, Javed A, Feroz RMA, Calhoun R. Partial repowering analysis of a wind farm by turbine hub height variation to mitigate neighboring wind farm wake interference using mesoscale simulations. *Appl Energy.* 2020; 268: 115050.
26. Mebane Jr WR, Sekhon JS. Genetic optimization using derivatives: The rgenoud package for R. *J Stat Softw.* 2011; 42: 1-26.
27. Shakoor R, Hassan MY, Raheem A, Wu YK. Wake effect modeling: A review of wind farm layout optimization using Jensen's model. *Renew Sustain Energy Rev.* 2016; 58: 1048-1059.
28. Moreno SR, Pierozan J, dos Santos Coelho L, Mariani VC. Multi-objective lightning search algorithm applied to wind farm layout optimization. *Energy.* 2021; 216: 119214.
29. Altan A, Karasu S, Zio E. A new hybrid model for wind speed forecasting combining long short-term memory neural network, decomposition methods and grey wolf optimizer. *Appl Soft Comput.* 2021; 100: 106996.
30. Sharma N, Sharma H, Sharma A, Bansal JC. A new effective solution for wind farm layout optimisation problem using self-sacrificing artificial bee colony algorithm. *Int J Renew Energy Technol.* 2022; 13: 396-442.
31. Kumar M, Sharma A. Progressive global best artificial bee colony algorithm for wind farm layout optimisation problem. *Int J Intell Eng Inform.* 2023; 11: 272-297.
32. Jensen NO. A note on wind generator interaction. Roskilde, Denmark: Risø National Laboratory; 1983.
33. Yeh TH, Wang L. A study on generator capacity for wind turbines under various tower heights and rated wind speeds using Weibull distribution. *IEEE Trans Energy Convers.* 2008; 23: 592-602.
34. Wallasch AK, Lüers S, Rehfeldt K, Ekkert M. Kostensituation der windenergie an land in deutschland [Internet]. Varel, Germany: Deutsche WindGuard GmbH; 2015. Available from: https://www.windguard.de/veroeffentlichungen.html?file=files/cto_layout/img/unternehmen/veroeffentlichungen/2015/KostensituationderWindenergieanLandinDeutschland-Update.pdf.
35. Oteri F. An overview of existing wind energy ordinances [Internet]. Golden, CO: National Renewable Energy Laboratory (NREL); 2008. Available from: <https://www.nrel.gov/docs/fy09osti/44439.pdf>.
36. Croonenbroeck C, Hennecke D. A comparison of optimizers in a unified standard for optimization on wind farm layout optimization. *Energy.* 2021; 216: 119244.