Zuse Institute Berlin

Takustr. 7
14195 Berlin
Germany

JAAP PEDERSEN[1], NIELS LINDNER[2], DANIEL REHFELDT[3],
THORSTEN KOCH[4]

Integrated Wind Farm Design: Optimizing Turbine Placement and Cable Routing with Wake Effects

[1] 0000-0003-4047-0042
[2] 0000-0002-8337-4387
[3] 0000-0002-2877-074X
[4] 0000-0002-1967-0077

ZIB Report 25-01 (Januar 2025)

Zuse Institute Berlin
Takustr. 7
14195 Berlin
Germany

Telephone: +49 30 84185-0
Telefax: +49 30 84185-125

E-mail: bibliothek@zib.de
URL: http://www.zib.de

ZIB-Report (Print) ISSN 1438-0064
ZIB-Report (Internet) ISSN 2192-7782

Integrated Wind Farm Design: Optimizing Turbine Placement and Cable Routing with Wake Effects

Jaap Pedersen[i,a], Niels Lindner[iia,b], Daniel Rehfeldt[iii,a,c], and Thorsten Koch[iv,a,d]

[a]Zuse Institute Berlin, Berlin, Germany
[b]Freie Universität Berlin, Berlin, Germany
[c]IVU Traffic Technologies, Berlin, Germany
[d]Technische Universität Berlin, Berlin, Germany

Abstract

An accelerated deployment of renewable energy sources is crucial for a successful transformation of the current energy system, with wind energy playing a key role in this transition. This study addresses the integrated wind farm layout and cable routing problem, a challenging nonlinear optimization problem. We model this problem as an extended version of the quota Steiner tree problem (QSTP), optimizing turbine placement and network connectivity simultaneously to meet specified expansion targets. Our proposed approach accounts for the wake effect − a region of reduced wind speed induced by each installed turbine − and enforces minimum spacing between turbines. We introduce an exact solution framework in terms of the novel quota Steiner tree problem with interference (QSTPI). By leveraging an interference-based splitting strategy, we develop an advanced solver capable of tackling large-scale problem instances. The presented approach outperforms generic state-of-the-art mixed integer programming solvers on our dataset by up to two orders of magnitude. Further, we present a hop-constrained variant of the QSTPI to handle cable capacities in the context of radial topologies. Moreover, we demonstrate that our integrated method significantly reduces the costs in contrast to a sequential approach. Thus, we provide a planning tool that enhances existing planning methodologies for supporting a faster and cost-efficient expansion of wind energy.

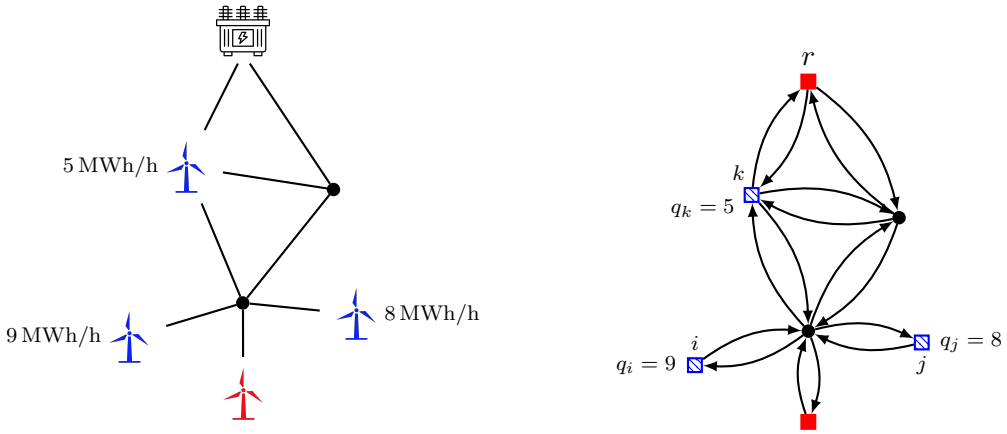
Keywords

OR in energy, combinatorial optimization, quota Steiner tree problem, wind farm design, wake effect

[i] 0000-0003-4047-0042, corresponding author, pedersen@zib.de
[ii] 0000-0002-8337-4387,
[iii] 0000-0002-2877-074X
[iv] 0000-0002-1967-0077

1 Introduction

The transformation of the existing energy system including an increasing deployment of renewable energy sources is vital to reduce global greenhouse gas emissions and to mitigate the impacts of climate change. Wind energy has proven to be an established source of electricity in this transformation (Veers et al., 2019). The global wind capacity has increased by more than 20% annually between 2000 and 2019 (Pryor et al., 2020), reaching 730 GW in 2020 (Our World in Data, 2021), with a further increase of 50% expected by the end of 2023 (Pryor et al., 2020). Additionally, a sharp decline of the already low cost of wind energy is expected by 2050 (Jansen et al., 2020; Wiser et al., 2016, 2021). This move towards renewable energy sources has created a more competitive market for companies investing in such technologies, making tools for designing and operating such systems optimally vital. With its impressive progress made in the last 20 years (Koch et al., 2022), mathematical programming has proven to be an efficient approach to tackle these challenging problems.



(a) IWFLCR instance with substation, three blue potential turbine locations with energy yield, one red fixed turbine, and two black additional nodes for cable routing (icons: flaticon.com). The aim is to connect the red turbine and some of the black turbines to the substation such that the investment costs are minimized, and a certain energy yield is met.

(b) QSTP instance with root r, three blue potential terminals i, j, k with their respective quota profits q_i, q_j, q_k, one red fixed terminal, and two black Steiner nodes. The aim is to find a minimum cost Steiner arborescence rooted in r and containing the red fixed terminal such that a certain quota is met.

Figure 1: Translating a wind farm to a combinatorial model.

In this paper, we investigate the optimal design of wind farms. We present a novel approach for solving the *integrated wind farm layout and cable routing optimization problem* (IWFLCR). Given a network of potential wind turbine locations and cable routes, the problem is to select turbine positions and a cable routing establishing connections to substations, i.e., the interfaces to the higher-level electricity grid. The goal is to minimize the total investment costs, that may depend on turbine location,

turbine type, cable type etc., while fulfilling a certain expansion target in terms of energy yield. In an abstract language, the IWFLCR can be modeled as a *Quota Steiner tree problem* (Pedersen et al., 2024) as visualized in Figure 1. This viewpoint however neglects important practical aspects, such as regions of slower wind speed and increased turbulence behind a turbine (wake effects), and bounds on the maximum number of turbines that can be connected by a single cable.

The contributions of our work are as follows. For our algorithmic approach, we focus on the combinatorial essence of such a network design problem, and introduce the *Quota Steiner tree problem with interference* (QSTPI). The QSTPI is an extension of the Steiner tree problem in graphs, and is enriched with important technical constraints such as maintaining a minimum distance between turbines and including wake effects. The minimum distances ensures that the blades to not touch and the impact of turbulence is limited (Fischetti, 2017; Fischetti & Fischetti, 2022). We therefore offer a discrete optimization model that can both profit from advanced computational techniques and as well as from integrating realistic aspects. We formulate QSTPI as mathematical program in several ways, both as binary quadratic integer program, and as integer linear program. Furthermore, we propose a method to split QSTPI instances in two, exploiting information on lower bounds on the total interference, and thereby accelerating the solution process. Our methods are incorporated into the SCIP-JACK software package (Rehfeldt, 2021), a state-of-the-art branch-and-cut-based solver specialized for Steiner tree problems, moreover advancing its shortest path primal heuristic to handle wake effects. We demonstrate on a realistic benchmark set that our splitting strategy solves more QSTPI instances, and about two orders of magnitude faster, than a black-box approach using the commercial solver GUROBI (Gurobi Optimization, LLC, 2024). Having developed an effective method to solve QSTPI instances, we also provide an extension, called *hop-constrained* QSTPI, that is able to deal with cable capacity limits. To this end, we apply our Steiner tree approach to a layered graph, and enhance our SCIP-JACK implementation with tailored cutting planes. In the end, we can solve hop-constrained QSTPI instances approximately seven times faster than GUROBI with a capacitated flow-based model. Finally, we underline that there is a significant price of sequentiality, i.e., there are noteworthy quantitative benefits that can be obtained by solving the wind farm layout and cable routing optimization simultaneously instead of sequentially. We further analyze the trade-off between using the hop-constrained QSTPI approach on one hand, and using unconstrained QSTPI with an a posteriori capacitated cable routing on the other.

This paper is structured as follows. We begin with a literature review in Section 2. The QSTPI and its integer programming formulations are presented in Section 3. We propose a solution strategy based on splitting the total interference experienced in the wind farm in Section 4. We integrate our proposed model into SCIP-JACK and

apply our new methodology in Section 5 on a large dataset. In Section 6, we extend our formulation to deal with cable capacity constraints. We highlight the benefits of using an integrated approach in contrast to the sequential one in Section 7. Finally, we discuss and conclude our method and results in Section 8.

2 Literature review

We will first review literature on wind farm planning in Section 2.1, and then proceed in Section 2.2 with our mathematical background, the Steiner tree problem.

2.1 Literature on wind farm planning

Designing wind farms involves a variety of decisions leading to many challenging optimization problems. In this paper, we focus on the integration of two main design tasks: the wind farm layout optimization (WFLO) and the wind farm cable routing (WFCR). There exists a vast amount of literature investigating both the WFLO and the WFCR. For an extensive overview we refer the reader to the surveys by Samorani (2013), Hou et al. (2019), and Fischetti and Pisinger (2019).

The WFLO maximizes the energy output of the wind farm. It does not only consider the number of turbines to be installed, but also takes the *wake effect* into account. A region of slower wind speed and increasing turbulence, a *wake*, is caused by each installed turbine, resulting in a reduced energy yield of turbines downwind. An investigation of the offshore wind farm areas along the US east coast by Pryor et al. (2021) shows that power reduction can be reduced by one-third due to wakes caused by upwind turbines and wind farms. Therefore, it is inevitable to consider the wake effect when modeling the WFLO. There exists a variety of studies modeling the WFLO as a mixed integer programming problem (MIP). A first MIP is presented in Donovan (2005) based on the independent set problem. Turner et al. (2014) approximate the nonlinear optimization problem of minimizing the wake effect by a quadratic integer and a mixed integer linear programming model (MILP). Based on the MILP proposed by R. Archer et al. (2011), Fischetti and Monaci (2016) combine the MIP approach with a proximity search heuristic to solve large-scale instances with the drawback of losing proven global optimality. Fleming et al. (2016) and P. Gebraad et al. (2017) couple the positioning and controlling of the turbines in the planning phase to further increase the annual energy production.

In the WFCR, the costs of connecting the chosen turbines to the power grid are minimized. In the case of offshore wind farms, González et al. (2014) point out that the costs of the electrical infrastructure account for 15% to 30% of the overall initial costs of the wind farm, making them an essential matter of expenses. During the operation,

power losses play a crucial role in terms of expected revenue. For example, Fischetti and Pisinger (2018) present a combination of MILP formulation and metaheuristics, named *matheuristics*, to minimize not only the initial costs, but also the reduced revenue due to power losses in the lifetime of a wind farm by considering cable capacities.

Due to the complexity of the WFLO and WFCR, the problems are usually solved sequentially. However, by design these problems are conflicting, i.e., WFLO aims to spread the turbines as far as possible to reduce wakes, whereas the WFCR tries to keep them as close as possible to reduce cable costs. Recent promising studies have been conducted to integrate these two problems balancing these conflicts (Cazzaro et al., 2023; Fischetti, 2021; Fischetti & Fischetti, 2022). Fischetti and Fischetti (2022) model the integrated layout and cable routing problem as a mixed-integer linear program in the context of offshore wind farms. The authors introduce a novel set of cutting planes and present combinatorial separation procedures integrated in a branch-and-cut solver. Although the problem is solved exactly, the approach is only performed on small instances with up to 40 potential turbine positions, whereas in practice, large wind farms might include up to 100 – 200 turbines[1]. Recently, Cazzaro et al. (2023) approach the integrated layout and cable optimization problem heuristically using an adapted variable neighborhood search. Even though the heuristic combining layout and cable routing provides better solutions than the previously proposed sequential approach, it cannot provide any conclusion in terms of quality with respect to a globally optimal solution.

2.2 Literature, algorithms and software for Steiner trees

The goal of this study is to provide a novel approach for solving the IWFLCR considering wake effects, minimum distance between turbines, and balancing investment cost and revenues. Conceptually, the IWFLCR is a network design problem with a number of additional constraints giving rise to a tree-shaped topology (Fischetti & Pisinger, 2018): We model the IWFLCR as an extension to the Steiner tree problem in graphs (STP), a classical NP-hard combinatorial problem (Karp, 1972), and one of the most studied problems in combinatorial optimization. Given an undirected graph with non-negative edge costs, the STP aims to find a tree that interconnects a given set of special points, referred to as *terminals*, at minimum cost. The STP and its variations arise in many real-world applications like, e.g., network design problems in telecommunication, electricity, or in district heating, as well as other fields such as biology; see, e.g., Leitner et al. (2014), Bolukbasi and Kocaman (2018), Ljubić et al. (2006), and Klimm et al. (2020), respectively. For a comprehensive overview of the STP and its variant the readers are referred to a recent surveys by Ljubić (2021), Re-

[1] https://en.wikipedia.org/wiki/List_of_offshore_wind_farms; Retrieved on 01/08/2025

hfeldt (2021), and Pedersen and Ljubić (2025), and the references therein. In the PhD thesis by Ridremont (2019), the STP is applied to find a robust cable network for the purpose of designing a wind farm. Recently, Pedersen et al. (2024) have investigated the quota-constrained variant of the STP (QSTP) in the context of large-scale onshore wind farm planning. However, in the context of Steiner trees, vital technical details, such as, among others, the wake effect or the minimum distance between turbines, are only implicitly modeled in the input data rather than in the optimization model. The authors introduce a novel transformation of the QSTP, which is integrated into the specialized Steiner tree solver SCIP-JACK, vastly outperforming state-of-the-art general out-of-the-box solvers. As an extension of the non-commercial general MIP solver SCIP (Bestuzheva et al., 2023), SCIP-JACK uses a branch-and-cut procedure to handle the exponential number of constraints induced by the Steiner cut-like constraints. These constraints will be recalled in Section 3.2. By additionally using various problem-specific reduction techniques and heuristics, SCIP-JACK has proven that it efficiently solves STP-related problems of magnitudes larger than what general out-of-the-box MIP solvers can even load into memory; see, e.g., Rehfeldt and Koch (2022, 2023) and Rehfeldt et al. (2019). For a comprehensive description of SCIP-JACK's operational principle the reader is referred to the PhD thesis by Rehfeldt (2021).

3 The Quota Steiner tree problem

We model the integrated wind farm layout and cable routing problem as a *Quota Steiner tree problem with interference* (QSTPI), extending and reformulating the classical quota Steiner tree problem (Johnson et al., 2000). In particular, we consider arbitrary tree-shaped network topologies. We introduce the QSTPI formally in Section 3.1, present a preliminary integer programming (IP) formulation in terms of directed cuts in Section 3.2, before applying the techniques by Pedersen et al., 2024 to obtain our final IP model in Section 3.3.

3.1 The Quota Steiner tree problem with interference

We adapt the formulation by Johnson et al. (2000) of the prize-collecting quota Steiner tree problem by adding vertex costs and integrating interference effects into the profit. Let $G = (V, E)$ be an undirected graph, whose vertex set V is partitioned into a set of *fixed terminals* T_f, a set of *potential terminals* T_p, and a set of additional *Steiner nodes*. We associate the potential terminals with *vertex costs* $b : T_p \to \mathbb{R}_{>0}$ and *quota profits* $q : T_p \to \mathbb{R}_{>0}$, referring to a *quota* $Q \in \mathbb{R}_{>0}$. Moreover, *edge costs* $c : E \to \mathbb{R}_{\geq 0}$ are given.

In terms of wind farm planning, the substations of the grid constitute T_f, while T_p

represents the available wind turbine positions. The Steiner nodes allow for a more flexible cable routing. The vertex costs b represent the investment costs for a turbine, e.g., foundation costs, which depend on the potential position, and the cost of the turbine itself, which may vary based on the chosen turbine type. The quota profits q the turbine's annual energy yield under a certain wind scenario. The built turbines should contribute an output of at least the quota Q. The set of edges E represents the cable connections we can choose from, and c comprises their costs.

In extension to the QSTP presented by Pedersen et al. (2024), each potential terminal, if chosen, may interfere with other potential terminals, reducing their assigned quota profit. As to wind turbine placement, each built turbine introduces a *wake*, an area of slower wind speed downstream, decreasing the power production of the following turbines, and, consequently, reducing its annually energy yield. We pin down this *wake effect* in a twofold way by *interference* and *minimum distance* notions: Let $I_{ij} \in \mathbb{R}_{\geq 0}$ be the interference experienced by the potential terminal $j \in T_p$ if the potential terminal $i \in T_p$ is chosen. Furthermore, let $d_{ij} \in \mathbb{R}_{\geq 0}$ denote the *distance* between two potential terminals $i, j \in T_p$, and let $D_{\min}$ be a *minimum distance*. While interference and minimum distance have been discussed before by Fischetti and Fischetti (2022) and Fischetti and Monaci (2016), we integrate these concepts into the QSTP from Pedersen et al. (2024).

Definition 3.1 (Quota Steiner tree problem with Interference)**.** *Given an instance* $(G, T_f, T_p, b, q, c, I, Q, d, D_{\min})$ *with* $G = (V, E)$ *as above, the* Quota Steiner tree problem with Interference (QSTPI) *is to find a tree* $S = (V', E') \subseteq G$ *that contains all fixed terminals* T_f*, minimizes the total cable and turbine costs*

$$C(S) \coloneqq \sum_{(i,j)\in E'} c_{ij} + \sum_{i \in T_p \cap V'} b_i, \tag{1}$$

fulfills the quota taking the induced interference into account:

$$Q(S) \coloneqq \sum_{i \in T_p \cap V'} \left(q_i - \sum_{j \in T_p \cap V' \setminus \{i\}} I_{ji} \right) \geq Q, \tag{2}$$

and guarantees that $d_{ij} \geq D_{\min}$ *for all* $i, j \in V' \cap T_p$ *with* $i \neq j$*.*

The QSTPI can also be seen as a single-objective specialization of a multi-objective problem, where cable and turbine costs are supposed to be minimized while simultaneously maximizing the output. In the QSTPI, a minimum output level is ensured by (2) and the number of turbines depends on the chosen quota, their individual quota profit, and the total induced interference, and, thus, is not known beforehand.

3.2 Directed cut formulation

To develop an integer programming (IP) formulation, we transform the QSTPI into a variant of the *Steiner arborescence problem* (SAP). The general SAP is presented, e.g., in Wong (1984). The original undirected graph G is transformed into a directed graph $D = (V, A)$ where $A \coloneqq \{(i,j),(j,i) : \{i,j\} \in E\}$. By shifting the costs of a vertex i onto the costs of its incoming arcs (see Ljubić et al., 2006), the arc costs are defined as

$$c_{ij} \coloneqq \begin{cases} c_e + b_j & \text{if } j \in T_p, \\ c_e & \text{otherwise,} \end{cases} \qquad \forall a = (i,j) \in A, \tag{3}$$

where c_e represents the cost of the corresponding edge $e = \{i,j\}$ in the original undirected graph G. For a subset of vertices $W \subseteq V$, we denote $\delta^+(W) \coloneqq \{(i,j) \in A : i \in W, j \in V \backslash W\}$ as the set of outgoing arcs and $\delta^-(W) \coloneqq \{(i,j) \in A : i \in V \backslash W, j \in W\}$ as the set of incoming arcs. For a single vertex i, we write $\delta^+(i) \coloneqq \delta^+(\{i\})$ and $\delta^-(i) \coloneqq \delta^-(\{i\})$. For any set K and any function $x : K \to \mathbb{R}$, we define $x(K) \coloneqq \sum_{i \in K} x_i$.

We introduce a binary variable x_{ij} for each $(i,j) \in A$ if the arc (i,j) is contained in the Steiner tree ($x_{ij} = 1$) or not ($x_{ij} = 0$). Furthermore, let y_k be a binary variable for each $k \in T_p$ indicating whether or not the potential terminal k is chosen. The rooted directed cut IP formulation of the QSTPI with an arbitrary root $r \in T_f$ is given as follows:

$$\begin{aligned}
\min \quad & c^T x && && (4)\\
\text{s.t.} \quad & x(\delta^-(W)) \geq 1 && \forall W \subset V, r \notin W, |W \cap T_f| \geq 1 && (5)\\
& x(\delta^-(W)) \geq y_i && \forall W \subset V, r \notin W, |W \cap T_p| \geq 1, i \in T_p && (6)\\
& \sum_{i \in T_p} \left(q_i - \sum_{j \in T_p} I_{ji} y_j \right) y_i \geq Q && && (7)\\
& y_i + y_j \leq 1 && \forall i,j \in T_p, i \neq j, d_{ij} < D_{\min} && (8)\\
& x_{ij}, y_k \in \{0,1\} && \forall (i,j) \in A, \forall k \in T_p && (9)
\end{aligned}$$

The Steiner cut constraint (5) requires that any subset of nodes of the graph $W \subset V$ that does not contain the root ($r \notin W$), but at least one fixed terminal, i.e., $|W \cap T_f| \geq 1$, has at least one incoming arc. In other words, for any cut between the root node and a fixed terminal, there has to exists a least one arc crossing this cut towards the side of the fixed terminal. This guarantees that there exists a path from the root r to each $t \in T_f$. Note, that if there exists multiple substations, i.e, $|T_f| > 1$, then there would exist a connection between the substations which is not desirable. However, by inserting an artificial root node r' connected to each substation $t \in T_f$ with an artificial

arc (r', t) with zero costs, one makes sure that there exists no directed path between any two substations. The artificial root node can also be seen as the higher level grid. (6) guarantees that there exists a path from r to the potential terminal $i \in T_p$ if it is chosen to contribute to the quota constraint (7) ($y_i = 1$). Finally, (8) ensures that no two potential terminals are chosen that are too close to each other.

3.3 Transformation

Having modeled the QSTPI as an IP by means of the directed cut formulation, we now shortly revisit the transformation of QSTP presented in Pedersen et al. (2024), which we extend by interference and minimal distance constraints. For each potential terminal $i \in T_p$, we add a new fixed terminal $i' \in T_f$. For each of these new fixed terminals i', we create an arc (r, i') with costs $c_{ri'} \coloneqq 0$ and an arc (i, i') with costs $c_{ii'} \coloneqq 0$, and add them to A; see Fig. 2. Since i' is a fixed terminal, the Steiner cut constraint (11) then requires that $x_{ii'} = 1$ or $x_{ri'} = 1$. We interpret the first case as turbine $i \in T_p$ being built, and the latter case as turbine i not being built. Following the argumentation in Pedersen et al. (2024), instead of counting the collected quota, we limit the amount of quota we can *waste*. Quota is wasted when a turbine i is not built, so that $x_{ri'} = 1$, see (12). The total interference I_{tot} is given by the interference between all chosen turbines, see (13). The minimum distance constraint (14) prevents building simultaneously two turbines that are too close. Note, that although the Steiner cut constraint (11) allows for solutions with $x(\delta^-(i')) = x_{ii'} + x_{ri'} = 2$, setting $x_{ii'} \coloneqq 0$ and $x_{ri'} \coloneqq 1$ results in a feasible solution with the same or less cost.

The transformed QSTP including interference and minimum distance constraint is given as follows:

QSTPI-TRANS :

$$\min \quad c^T x \tag{10}$$

$$\text{s.t.} \quad x(\delta^-(W)) \geq 1 \qquad \forall W \subset V, r \notin W, |W \cap T_f| \geq 1 \tag{11}$$

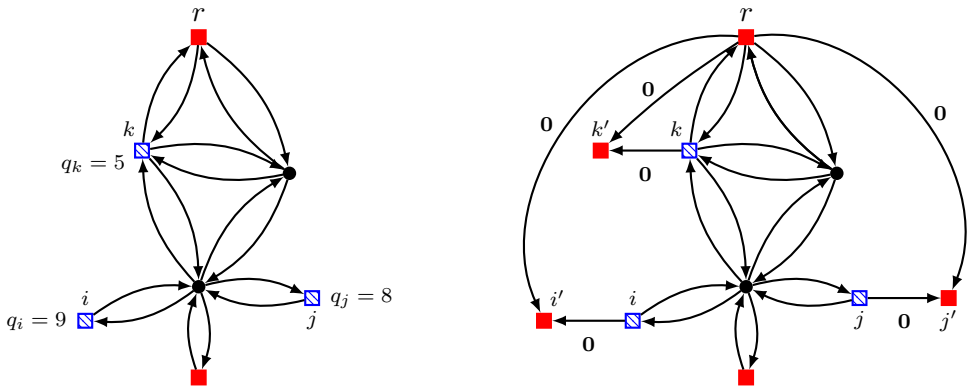
$$\sum_{i \in T_p} q_i x_{ri'} + I_{\text{tot}} \leq \sum_{i \in T_p} q_i - Q \tag{12}$$

$$I_{\text{tot}} \geq \sum_{i \in T_p} \sum_{j \neq i \in T_p} I_{ij} x_{ii'} x_{jj'} \tag{13}$$

$$x_{ii'} + x_{jj'} \leq 1 \qquad \forall i, j \in T_p, i \neq j, d_{ij} < D_{\min} \tag{14}$$

$$x_{i,j} \in \{0, 1\} \qquad \forall (i, j) \in A \tag{15}$$

The formulation QSTPI-TRANS is a quadratic-constrained binary optimization problem due to (13) and NP-hard (Pedersen et al., 2024). Due to the exponential number of constraints induced by the Steiner cut constraints (12), it is reasonable to



(a) Original instance. **(b)** Transformed instance.

Figure 2: Transformation of QSTPI. For each potential terminal $i \in T_p$ (blue squares) a fixed terminal (red square) i' is added, which is connected to the root node r and the original vertex by an arc of zero costs; Steiner nodes are shown as filled circles.

use a solver which is specialized in separating these constraints, e.g., SCIP-JACK (Rehfeldt & Koch, 2023). As an extension to the general framework SCIP (Bestuzheva et al., 2023), which can handle mixed integer nonlinear programs (MINLP), one could integrate the quadratic constraint (13) directly. However, preliminary experiments have shown that explicitly linearizing the bilinear products $x_{ii'}x_{jj'}$ in (13) using McCormick's linearization (McCormick, 1976) appears to be faster than let SCIP handle the nonlinearities. By introducing additionally variables m_{ij}, (13) can be replaced as follows:

$$I_{\text{tot}} \geq \sum_{i \in T_p} \sum_{j \neq i \in T_p} I_{ij} m_{ij} \tag{16}$$

$$m_{ij} \geq x_{ii'} + x_{jj'} - 1 \qquad \forall i < j \in T_p, \tag{17}$$

$$m_{ij} \geq 0 \qquad \forall i < j \in T_p. \tag{18}$$

As the total interference I_{tot} is bounded from above in (12), no upper bound has to be imposed on m.

3.4 The flow-based approach

Due to the exponential number of constraints (11), which cannot be handled by general MIP solvers out-of-the-box, the problem can alternatively be modeled by a standard flow-based MIP formulation, commonly used in such applications; see, e.g., Fischetti and Fischetti (2022). The flow-based MIP formulation of the non-transformed QSTPI

is given by

FLOW :

$$\min \quad c^T x \tag{19}$$

$$\text{s.t.} \quad \sum_{i \in T_p} q_i y_i - I_{\text{tot}} \geq Q \tag{20}$$

$$I_{\text{tot}} \geq \sum_{i \in T_p} \sum_{j \neq i \in T_p} I_{ji} y_j y_i \tag{21}$$

$$\sum_{a \in \delta^-(i)} f_a - \sum_{a \in \delta^+(i)} f_a = \begin{cases} 0 & \forall i \in V \setminus (T_f \cup T_p) \\ 1 & \forall i \in T_f \setminus \{r\} \\ y_i & \forall i \in T_p \end{cases} \tag{22}$$

$$y_i + y_j \leq 1 \qquad \forall i \neq j \in T_p, d_{ij} < D_{\min} \tag{23}$$

$$x_a \leq y_i \qquad \forall a \in \delta^-(i), \forall i \in T_p \tag{24}$$

$$f_a \leq M x_a \qquad \forall a \in A \tag{25}$$

$$y_i \in \{0, 1\} \qquad \forall i \in T_p \tag{26}$$

$$x_a \in \{0, 1\},\ f_a \in \mathbb{R}_{\geq 0} \qquad \forall a \in A \tag{27}$$

where x and y denote the decision variables if an arc and a vertex is chosen, respectively, and f describes the flow over the arcs. The costs are minimized in (19), note, that the costs c contain the turbine costs if chosen, see (3). Constraint (20) describes the quota constraint including the total inference I_{tot} caused by the chosen turbines in (21). The flow balance at each vertex depends on its type and is captured by constraint (22). For any Steiner node $i \in V \setminus (T_f \cup T_p)$ the ingoing and outgoing flow has to be equal. Any fixed terminal, i.e., the substations, which is not the root, has to be connected to the root node, so a "flow" f of one has to reach the terminal, i.e., the right-hand side is equal to one. Also, any chosen potential terminal (turbine), i.e., $y_i = 1$, has to be connected to the root node. The incoming arcs of a potential terminal can only be active if the potential terminal is chosen, as in (24). Equation (25) ensures that a flow over an arc is only possible if the arc is active. The big-M notation is used to limit the flow on an active arc, i.e., the upper bound on f_a. We choose $M = |T_f \cup T_p|$ as an upper limit, which would allow all fixed and potential terminals to be connected via a single string.

4 Split the problem in two

In this section, we introduce a splitting strategy based on lower bounds on the interference. We will use this method to accelerate the solution process for practical instances of the QSTPI later on.

4.1 Bounding local interference

Concerning wind farms, the layout problem promotes solutions with wide-spread wind turbine positions, whereas the cable routing problem aims to have them as close as possible to the substation. Furthermore, the linearization (17) allows for highly fractional solutions of the LP-relaxation of the QSTPI-TRANS ((10)–(12) and (15) – (18)), i.e., relaxation (15) to $x_{ij} \in [0,1] \forall (i,j) \in A$, with values $x_{ii'} = x_{jj'} = 0.5$ and $m_{ij} = 0$, which means that i and j contribute to the quota, but not to the interference value, thus causing highly fractional solutions. Fischetti and Fischetti (2022) present cuts for the flow-based formulation that increase the lower bound of interference caused by choosing to install a turbine, and that have also been useful for some hard quadratic problems, including the Quadratic Assignment Problem, see Fischetti et al. (2012). A particular and effective cut is given by

$$\sum_{j \neq i \in T_p} I_{ji} x_{jj'} x_{ii'} \geq \mathrm{LB}_i x_{ii'}, \quad \forall i \in T_p, \tag{28}$$

i.e., the total interference caused by turbine i if built is bounded by a value LB_i. The authors propose that if one aims to build $N_{\min}$ turbines, then LB_i can be chosen as the sum of the $N_{\min} - 1$ lowest values of I_{ij} for all i, j with $d_{ij} \geq D_{\min}$. One can extended this thought towards the total induced interference by taking the sum of the $N_{\min}$ lowest values of LB_i to determine a lower bound I_{LB}: Let $(\mathrm{LB}_{(1)}, \ldots, \mathrm{LB}_{(|T_p|)})$ be the ascending sort of all LB_i, i.e. $\mathrm{LB}_{(1)} \leq \cdots \leq \mathrm{LB}_{(|T_p|)}$. Then the lower bound of the total induced interference is given by:

$$I_{\mathrm{tot}} \geq I_{\mathrm{LB}} = \sum_{1}^{N_{\min}} \mathrm{LB}_{(i)}. \tag{29}$$

4.2 Splitting the total interference

Considering large areas with a high number of potential wind turbine location from which only a small subset is chosen to be built, the previous described lower bound easily is small compared to the actual induced interference. Motivated by this fact, we split QSTPI-TRANS into two subproblems as follows: We define a split value $I_{\mathrm{split}} \in \mathbb{R}_{>0}$ for the total interference, which will be determined in Section 4.3. In the first subproblem $P_{\geq}$, which is called the *upproblem*, this split value is imposed as

a lower bound on the total interference, i.e., $I_{\text{tot}} \geq I_{\text{split}}$. In the second subproblem $P_{\leq}$, the *downproblem*, the split value represents the upper bound on the total interference, i.e., $I_{\text{tot}} \leq I_{\text{split}}$. At first, this might seem counterintuitive, as we consider two possibly computationally challenging problems instead of one. However, as we show in Section 5.4, by carefully choosing I_{split}, this approach is very effective in practice. Furthermore, the following easy observation holds:

Lemma 4.1. *An optimal solution $S^*_{\geq}$ to the problem $P_{\geq}$ is also optimal to QSTPI if $P_{\leq}$ has no solution $\overline{S}_{\leq}$ with $c(\overline{S}_{\leq}) < c(S^*_{\geq})$.*

Proof. Given an optimal solution $S^*_{\geq}$ to the problem $P_{\geq}$ and let S^* be the optimal solution to QSTPI. If $c(S^*) < c(S^*_{\geq})$, then S^* is not feasible for $P_{\geq}$, but must be feasible for $P_{\leq}$. □

4.3 A valid lower bound for the total interference

There are multiple ways to choose the split value I_{split}, e.g., as a fixed parameter or based on some heuristic. Let us consider the following minimization problem:

$$\begin{aligned}
\text{MinI}: \quad \min \quad & I_{\text{tot}} && && (30)\\
\text{s.t.} \quad & \sum_{i \in T_p} q_i y_i - I_{\text{tot}} \geq Q && && (31)\\
& I_{\text{tot}} \geq \sum_{i \in T_p} \sum_{j > i \in T_p} (I_{ij} + I_{ji}) y_j y_i && && (32)\\
& y_i + y_j \leq 1 && \forall i, j \in T_p, i \neq j, d_{ij} < D_{\min} && (33)\\
& y_i \in \{0, 1\} && \forall i \in T_p && (34)\\
& I_{\text{tot}} \in \mathbb{R}_{\geq 0}, && && (35)
\end{aligned}$$

i.e., the problem is to choose a subset of potential terminals that minimizes the total interference (30), while fulfilling the desired quota (31), considering the total interference (32), and respecting the minimal distance constraints (33). This problem can be seen as a version of the WFLO. As the total interference is minimized in MinI, there will be no solution for QSTPI with a lower interference than in an optimal solution of MinI.

As splitting into up- and downproblem is pointless when $I_{\text{split}} = 0$, we will now give a necessary and sufficient criterion when MinI has an optimal solution of value zero. Let us consider the undirected graph $G_p = (T_p, E_p)$, where for each pair (i, j) of distinct vertices, there is an edge $(i, j) \in E_p$ if and only if $I_{ij} + I_{ji} > 0$ or $d_{ij} < D_{\min}$. A subset of vertices $S \subseteq T_p$ is called an *independent set* in G_p iff there is no edge in E_p that connects any two vertices in S. A *maximum independent set* (MIS) of graph G_p is an independent set that maximizes $q(S)$.

Theorem 4.2. *Let Q^*_{MIS} be the quota of an MIS in G_p and I^*_{tot} an optimal solution to* MinI*. Then*

$$Q \leq Q^*_{\mathrm{MIS}} \Leftrightarrow I^*_{\mathrm{tot}} = 0 \tag{36}$$

Proof. The total interference of MinI is bounded by zero from below. Let S be an MIS and set $y_i \coloneqq 1$ if and only if $i \in S$. By definition, S does not contain any pair (i, j) of vertices with $I_{ij} + I_{ji} > 0$ or $d_{ij} < D_{\min}$, so that (32) and (33) trivially hold. Moreover, (31) reads as $Q^*_{\mathrm{MIS}} - I_{\mathrm{tot}} \geq Q$. In particular, $I_{\mathrm{tot}} = 0$ is feasible and hence optimal for MinI if and only if $Q^*_{\mathrm{MIS}} \geq Q$. □

Rephrasing Theorem 4.2, a necessary condition for a non-trivial total interference ($I_{\mathrm{tot}} > 0$) is that $Q > Q^*_{\mathrm{MIS}}$. Combining Lemma 4.1 and Theorem 4.2, we obtain:

Corollary 4.3. *Let I_{LB} be a valid lower bound on the problem* MinI*. An optimal solution $S^*_{\geq}$ to the problem $P_{\geq}$ is also optimal to P if I_{LB} is chosen as split value I_{split}.*

That means, by finding a valid lower bound to MinI greater than zero, only the smaller upproblem $P_{\geq}$ has to be solved to solve the full QSTPI.

5 Computational Study

In the following, we describe the implementation and the setup of our computational study in Section 5.1 and Section 5.2, respetively. Furthermore, we present the data used in this study in Section 5.3. We conclude this section by demonstrating the strength of our approach compared to the state-of-the-art general MIP solver GUROBI in Section 5.4.

5.1 Implementation

Based on the promising results shown in Pedersen et al. (2024), we have integrated QSTPI-TRANS as in Section 3.3 into the Steiner tree software package SCIP-JACK. The exponentially-many constraints (11) are separated using the sophisticated maximum-flow algorithm implemented by Rehfeldt (2021) which is based on the classical max-flow/min-cut theorem. Additionally to the separation procedure, the solution process is supported by a *shortest-path heuristic* (SPH).

In general, heuristics are utilized to find good and feasible solutions in short time to provide upper bounds on the exact solution of the problem. Within the context of STP-related problems, the SPH is the best-known heuristic and was introduced by Takahashi and Matsuyama (1980). For this study, we adapt SCIP-JACK's SPH implementation by Rehfeldt (2021) and already customized for the case of the QSTP by

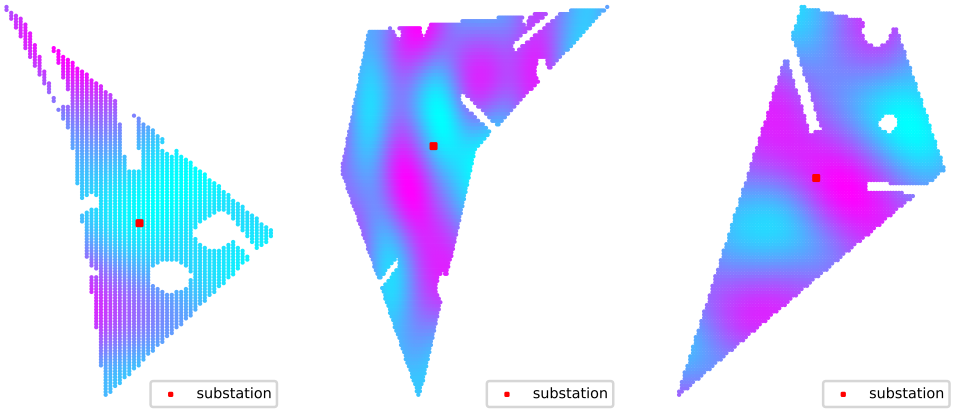
Pedersen et al. (2024): Starting at the root r, the closest fixed terminal or potential terminal is inserted to the *root component*. The interference between the newly inserted node and all nodes in the root component is calculated and added to the total interference. Then, the distance of all non-connected vertices to the root component is updated. This process is repeated until all fixed terminals are connected and the connected potential terminals minus the total interference fulfill the desired quota. Preliminary experiments have shown that, although the SPH finds good upper bounds at the root node, it rarely improves the primal bound during the branch-and-bound procedure. Therefore, we decided to run the SPH only at the root node, modifying the arc costs to $\overline{c}_a := (1.0 - \hat{x}_a)c_a$ for all $a \in A$ depending on the current LP-solution $\hat{x}$. Our solver will be benchmarked against a black-box MIP solver using the flow-based formulation presented in Section 3.4.

5.2 Computational setting

The QSTPI-TRANS is integrated into SCIP-JACK using SCIP 8.0.1 (Bestuzheva et al., 2023) with CPLEX 12.10 (IBM ILOG, 2022) as LP solver. The flow-based MIP formulation is implemented using the GUROBI PYTHON-interface in PYTHON 3.11.2, and is solved with GUROBI 11.01 (Gurobi Optimization, LLC, 2024). The following settings are run:

- `FLOW`: the basic flow-based model, presented in (19) – (27);
- `FLOW-Ilb`: as before, but introducing the lower bound I_{LB} described in Section 4 on the total induced interference;
- `QSTPI`: the QSTPI-TRANS formulation presented in (10) – (12) and (14) – (18);
- `QSTPI-Ilb`: as before, but introducing the lower bound I_{LB} described in Section 4 on the total induced interference;
- `QSTPI-S-H`α: as before, but a I_{split} is determined using the interference value from the initial shortest-path heuristic solution multiplied by $\alpha \in \{0.1, 0.5, 0.8\}$;
- `QSTPI-S-minI-`τ: as before, but $I_{\text{split}} := \max(I_{\text{LB}}, I^*_{\text{tot}})$, where I^*_{tot} corresponds to the best primal solution of the minimization problem MinI presented in (30) – (35) within a time limit of $\tau \in \{300, 600, 1800\}$ seconds using SCIP 8.0.1.

All computational experiments are executed single-threaded in a non-exclusive mode on a cluster with *Intel(R) Xeon(R) Gold 6342* CPUs running at 2.80 GHz, where four CPUs and 64 GB of RAM are reserved for each run. For a better comparison with our implementation in SCIP-JACK, which is only single-threaded, we restrict



(a) Instance A **(b)** Instance B **(c)** Instance C

Figure 3: Chosen representative wind farms by Cazzaro and Pisinger, 2022. Color of regions represents the costs of building a turbine from low to high costs.

the black-box solver GUROBI to one thread when discussing the results. However, further experiments have shown that under the given setup no performance improvement is gained when running GUROBI in 8-threaded mode. We set a time limit of six hours (21,600 s). In the cases of `QSTPI-S-H`α and `QSTPI-S-minI-`τ, the downproblem is only solved if the upproblem is solved to optimality. The downproblem is then run using the optimal objective value of the upproblem as an objective bound (see Lemma 4.1).

5.3 Data creation

As a benchmark set, we use three representative wind farm instances out of the dataset introduced by Cazzaro and Pisinger (2022), namely instances A, B, and C. Instance A and C are divided into two zones in the original data set of which we only consider the larger zone for this paper; see Fig. 3. For each wind farm, we create 20 small-sized, medium-sized, and large-sized instances each by randomly sampling 100, 200, and 500 potential positions, respectively.

For instance A we set the quota equivalent to the annual power production without interference of 10 and 20 wind turbines with 15 MW each based on the real wind measurements at Ten Noorden van de Waddeneiladen[2] for all sampled instances. For instance B the quota is set to 10, 20, and 40 wind turbines for the small-sized instances and 10, 20, 40, and 60 for the medium- and large-sized instances in the same way as before. For instance C the quota of the small- and medium-sized instances is set to 10, 20 and 40, and for the large-sized ones to 10, 20, 40, and 60 wind turbines in the same way as before. Due to unlucky sampling, too many turbines might be too

[2]reported by https://english.rvo.nl/topics/offshore-wind-energy/noorden-waddeneilanden Accessed on: 11/13/2025; Given in dataset from Cazzaro and Pisinger (2022)

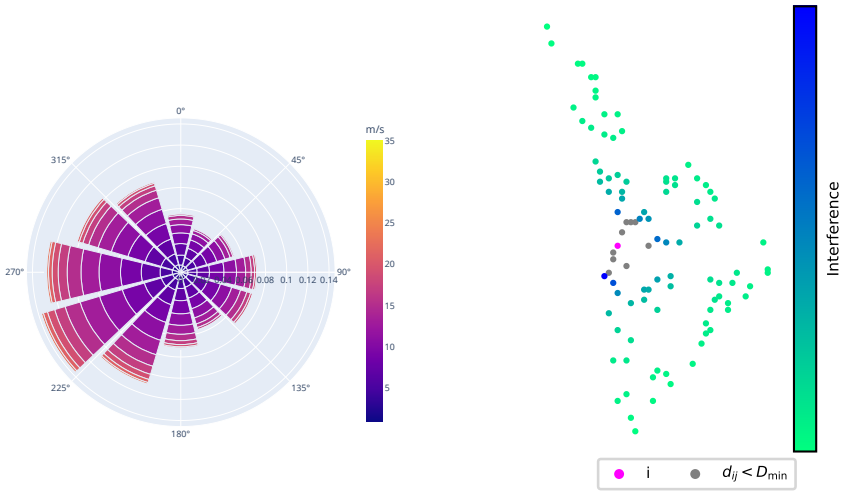
close to each other to satisfy the quota constraint. Removing these trivially infeasible instances, our complete benchmark set has 143 (instead of 160 instances) small-, 180 medium-, and 200 large-sized instances. As the focus lies on the number of potential turbine positions, no additional Steiner nodes are considered.

The interference is calculated by the sum of pairwise interferences using Jensen's model (Jensen, 1983), an analytical model for wind parks, as done in Fischetti and Monaci (2016), based on the same real wind measurements given in the dataset by Cazzaro and Pisinger (2022); see Fig. 4a. While more accurate interference models exist in comparison to Jensen's model (e.g., Bay et al. (2023), P. M. O. Gebraad et al. (2014), P. M. O. Gebraad and van Wingerden (2014), and King et al. (2021)), it still provides a useful approximation for early-stage wind farm layout planning (C. L. Archer et al., 2018; Fischetti & Monaci, 2016). As our focus is on network design methodology, and we assume that the interference is given as an input to our problem, we stick with this computationally simpler model. Fig. 4b shows the interference caused by a turbine under the considered wind scenario in an exemplary sample of instance A. No information regarding the position of the substation is given in the original dataset. Therefore, the substation is placed in the middle of each area, i.e., the coordinates of the substation are given by averaging the coordinates of all available positions. The dataset by Cazzaro and Pisinger (2022) is used, which contains the cost to build a turbine at a specific position, which includes the foundation costs, as well as data for 15 MW turbines with a rotor diameter of 240 meters. The minimum distance required between two turbines is usually three to five times the rotor diameter; Following Fischetti and Fischetti (2022), we set the minimum distance $D_{\min}$ to five times the rotor diameter, i.e., 1200 meters. The cable costs are chosen in accordance to Cazzaro et al. (2020). The authors provide cable costs for three types of cables: 430, 480, and 610 k€/km. Since we do not consider different cable types, a mean value of 504 k€/km is chosen.

5.4 Results

Fig. 5 shows the cumulative percentage of instances solved to optimality over time for the complete problem for all settings. In the legend of the figure, the first number in the brackets reports the number of instances completely solved to optimality. The second number gives the number of instances, for which the upproblem was solved to optimality. In all solved instances, the global optimal solution lies within the up-problem. Furthermore, the total interference induces by the optimally chosen turbines is far away of the introduced splitting value I_{split}, when considering all settings but `QSTPI-S-H0.8`.

Whereas the simple QSTP without the interference and minimum distance con-



(a) Wind rose **(b)** Interference caused by turbine i

Figure 4: Wind rose of Ten Noorden van de Waddeneilanden site (Panel 4a); Interference caused by turbine i in chosen wind scenario for a sampled instance of wind farm area A (Panel 4b); Interference calculated by the Jensen model (Jensen, 1983). Wind farm and wind data from Cazzaro and Pisinger, 2022.

straints has outperformed the flow-based formulation by up-to two orders of magnitude before (see Pedersen et al. (2024)), this advantage vanishes complete considering `QSTPI` and `FLOW`. `QSTPI` gives very poor performance only solving 61 instances within the time limit of six hours, which is not even half the number of instances solved by `FLOW`. The performance is drastically increased by introducing the lower bound I_{LB} described in Section 4. `FLOW-Ilb` is not only more than a magnitude faster than `QSTPI` and `FLOW`, but also solves 167 and 86 more instances, respectively. Applying the same lower bound value on our approach in `QSTPI-Ilb`, the performance is increased by another magnitude solving 239 of 323 instances to optimality. Remarkably, all these instances are solved within roughly one hour: In relation to that, `FLOW-Ilb` only solves around 50% of the instances in that time.

Considering `QSTPI-S-H`α, the best performance is achieved with $\alpha = 0.1$, being around a magnitude faster than $\alpha = 0.5$ and $\alpha = 0.8$. `QSTPI-S-H0.1` solves 240, `QSTPI-S-H0.5` solves 237, and `QSTPI-S-H0.8` solves 238 instances out of 323 instances to optimality. By using a higher split value the better is the performance in solving the number of instance for the upproblem, i.e., 240, 242, and 279 for α equal to 0.1, 0.5, and 0.8, respectively (see the second number in the brackets in the legend of Fig. 5). However, the downproblem becomes much harder to solve, and 0, 5, and 41 instances run into the time limit in the downproblem for α equal 0.1, 0.5, and 0.8, respectively.

For the settings `QSTPI-S-minI-`τ none of the starting problem MinI that determines the split value I_{split} is solved to optimality within the time limit of τ. Therefore,

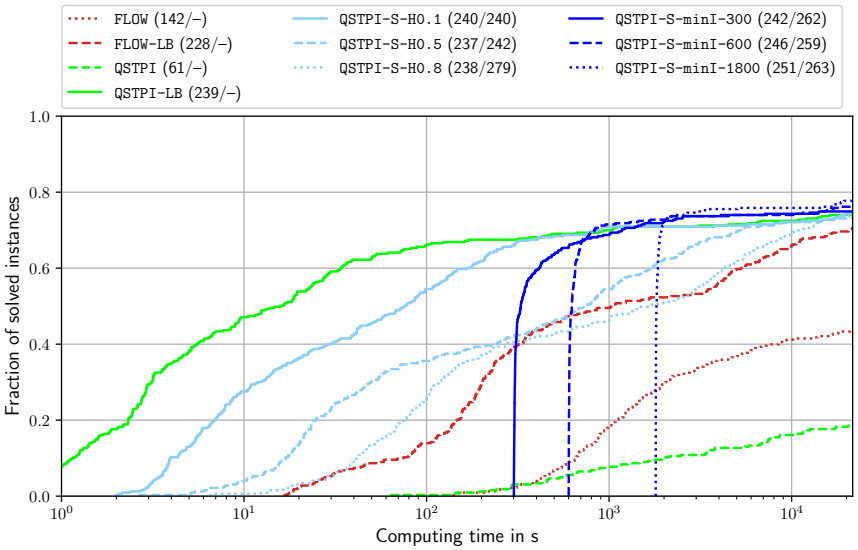
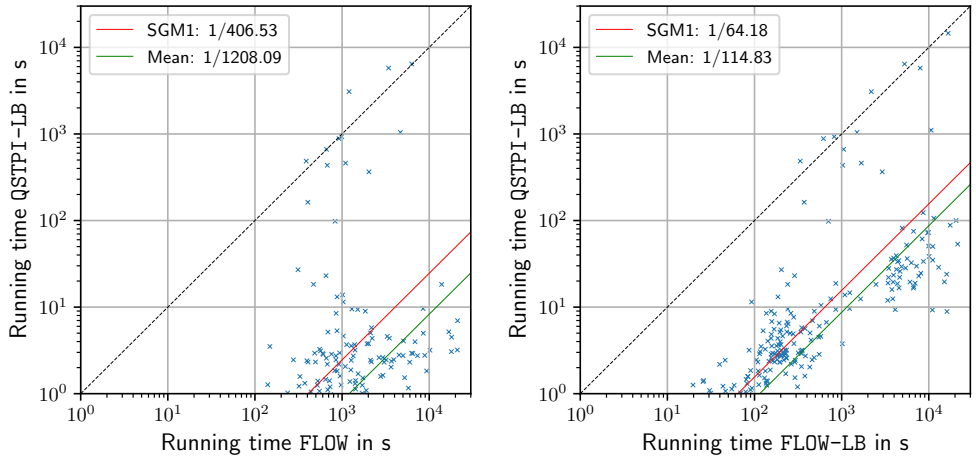


Figure 5: Cumulative percentage of instances solved to optimality over time for all settings for small- and medium-sized instances. The numbers in brackets state the number of instances which are solved to optimality completely and instances for which the upproblem is solved to optimality.

the minimum time to solve an instances using `QSTPI-S-minI-`τ is greater than τ. However, the overall performance is similar to `QSTPI-Ilb` with the advantage of solving three, seven, and 12 more instances for τ equals 300, 600, and 1800 second, respectively.

The performance profiles in Fig. 5 have already shown the advantages of our novel approach for the IWFLCR problem. In the following, the setting `QSTPI-Ilb` is first compared instance-wise to `FLOW` and then to `FLOW-Ilb` in Fig. 6a and Fig. 6b. For the pairwise comparison we only consider instances solved to optimality by both settings. In the legend, additional to the arithmetic mean of the speedup, the shifted geometric mean with a shift of 1 (SGM1) is reported. The *shifted geometric mean* (Achterberg, 2007) has become a standard measure in discrete optimization (Mittelmann, 2020) and is less biased towards harder instances. By using `QSTPI-Ilb` instead of `FLOW` an average speedup of 1208.9 and a SGM1 of 406.53 is achieved, where `QSTPI-Ilb` is faster on all but 5 instances, see Fig. 6a. `QSTPI-Ilb` is 114.83 times faster on average and 64.18 times faster considering the SGM1 compared to `FLOW-Ilb`; see 6b. Again, `QSTPI-Ilb` is faster on all but 5 instances.

The benefit of our novel approach becomes even more prominent when we are looking at the performance for the large-sized instances in Fig. 7. For these instances, only settings `FLOW-Ilb`, `QSTPI-Ilb`, `QSTPI-S-H0.5`, and `QSTPI-S-minI-1800` are run, as they are expected to perform best for the large-sized instances. The setting `FLOW-Ilb` only solves 22 instances out of 200 to optimality. The performance of the QSTPI related settings is relatively similar. However, the best setting is `QSTPI-S-minI-1800`



(a) `QSTPI-Ilb` vs. `FLOW`: 141 instances solved by both settings.

(b) `QSTPI-Ilb` vs. `FLOW-Ilb`: 227 instances solved by both settings.

Figure 6: Comparison of running times between `QSTPI-Ilb` and `FLOW` (Panel 6a) and of running times between `QSTPI-Ilb` and `FLOW-Ilb` (Panel 6b) of all instances solved by both settings, respectively. The dotted line shows the break even line; The *SGM1* line represents the shifted geometric mean of the speedup with a shift of one second; the *Mean* line represents the arithmetic mean of the speedup.

solving 86 out of 200 instances to optimality and seven more instances in the upproblem. The setting `QSTPI-S-H0.5` solves 77 instances and with that three more instances than `QSTPI-Ilb`.

6 A capacitated approach for the QSTPI

In the previous section, the advantages of formulating the IWLCR problem as Steiner-tree-based mixed integer program instead of a flow-based approach have become clear: We solve larger instances in shorter computational time. On the downside, our approach might produce solutions, in which a higher number of turbines are connected via a single cable with the substation, violating practically required cable capacity limits; see Fig. 8. These restrictions can be dealt with in an a-posteriori step, and we will do so in Section 7. However, we can also integrate cable capacity constraints into our previous model.

6.1 Hop-constrained QSTPI

Various topologies are possible for the cable routing in a wind farm, see, e.g., Fischetti and Pisinger (2018) and Pérez-Rúa and Cutululis (2019). A common requirement in industrial application is a radial string topology, in which turbines are connected in (balanced) strings outgoing of the substation, see, e.g., Cazzaro et al. (2022, 2023). Exploiting this fact, we model cable capacities by limiting the length of each string

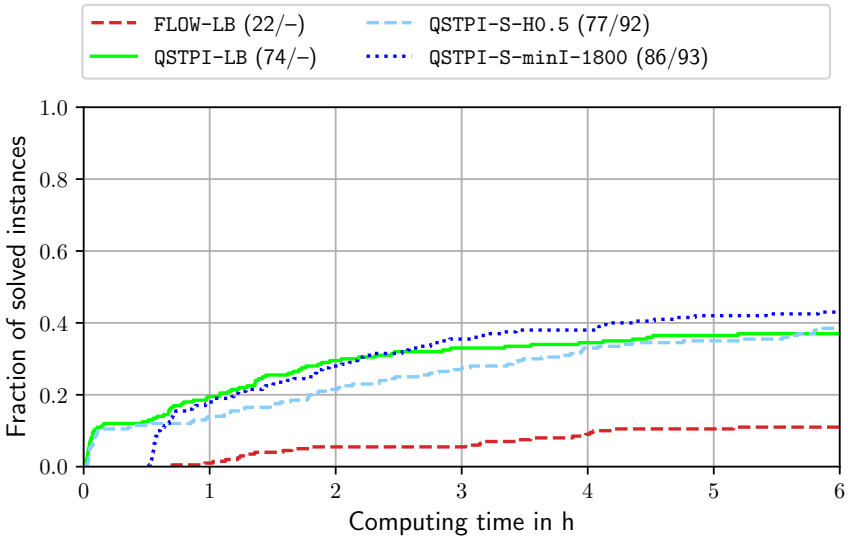


Figure 7: Cumulative percentage of instances solved to optimality over time for promising settings for large-sized instances. The numbers in brackets state the number of instances which are solved to optimality completely and instances for which the upproblem is solved to optimality.

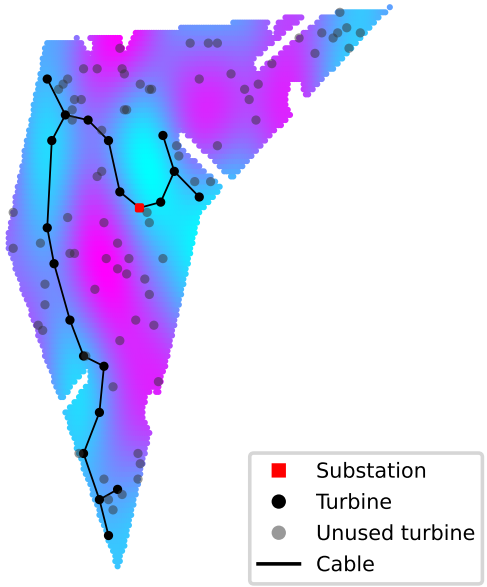


Figure 8: Exemplary unbalanced solution for the QSTPI connecting 16 turbines to the substation along a single. If, e.g., as in Cazzaro et al., 2023, a cable can only connect at most six 15 MW turbines, this is not a practically feasible solution.

instead of imposing capacity bounds on each arc as usually done in flow-based network planning approaches. This limit is referred to as *hop-limit*, i.e., the limit of arcs one is allowed to traverse (*hop*) to reach a chosen turbine. For more details on general hop-constrained Steiner tree problems, the reader is referred to Costa et al. (2009), Gouveia et al. (2011), and Sinnl and Ljubić (2016). We call our extension by hop limits the *hop-constrained QSTPI*.

6.1.1 Layered graph approach

To model the hop-constrained QSTPI, we follow the layered graph approach proposed by Sinnl and Ljubić (2016). We consider the transformed directed graph $G = (V, A)$ as described in Section 3.3. We distinguish between three pairwise disjoint types of V: The potential terminals $i \in T_p$ (turbines) with corresponding fixed terminals $i' \in T_f$, and the artificial root (substation) r. We assume that no further fixed terminals exist, as our aim is to build a radial layout, and we also disregard additional Steiner nodes. Given a *hop-limit* H, we create a layered graph $G_L = (V_L, A_L)$ with $H+1$ layers. The set of nodes is defined as $V_L \coloneqq V_0 \cup \ldots \cup V_H$, where

$$\begin{aligned} V_0 &\coloneqq \{r\} \cup T_f, \\ V_h &\coloneqq \{i_h \mid i \in T_p\}, \quad h \in \{1, \ldots, H\}, \end{aligned} \tag{37}$$

i.e., layer zero only contains the root node and the fixed terminals, while all other layers contain a copy of each original potential terminal. The node i_h refers to the node i in layer h. The set of arcs is defined as $A_L \coloneqq A_0 \cup \ldots \cup A_H$, where

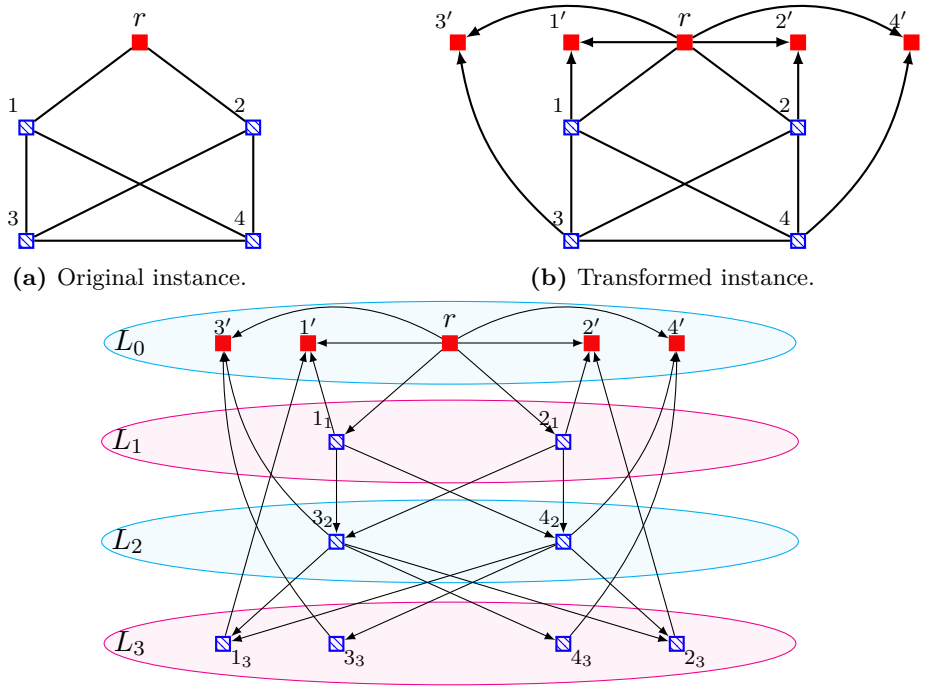
$$\begin{aligned} A_0 &\coloneqq \{(r, i') \mid (r, i') \in A, i' \in T_f\}, \\ A_1 &\coloneqq \{(i_h, i') \mid (i, i') \in A, i \in T_p, i' \in T_f, h \in \{1, \ldots, H\}\}, \\ A_h &\coloneqq \{(i_{h-1}, j_h) \mid (i, j) \in A, i, j \in T_p\}, \qquad h \in \{2, \ldots, H\}. \end{aligned} \tag{38}$$

Figure 9 shows the transformation from the original graph to the layered graph with a hop-limit of $H = 3$.

The rationale behind the layered graph construction is the following observation, that allows to translate between certain Steiner trees in the transformed graph G and hop-constrained radial-layout Steiner trees in G_L.

Lemma 6.1. *There is a one-to-one correspondence between the following sets:*

- *the arborescences in G_L that are rooted at r, contain T_f and at most one node i^h for each $i \in T_p$, and have the property that each $i^h \in V_1 \cup \cdots \cup V_{H-1}$ has at most one outgoing arc to layer $h+1$.*
- *the arborescences in G that are rooted at r, contain T_f, and have the property*



(a) Original instance.

(b) Transformed instance.

(c) Layered instance with hop-limit $H = 3$.

Figure 9: Transformation for the hop-constrained QSTPI. We add first the artificial root r, and for each potential terminal $i \in T_p$ (blue squares) a fixed terminal (red square) $i' \in T_f$ as before to obtain the transformed graph G. We then build the layered graph G_L with $H = 3$ layers. We omit nodes with no ingoing arcs, as they cannot be reached by a directed path from r.

that for all $i \in T_p$, the number of outgoing arcs is at most one, and that the length of the unique r-i-path is at most H.

Proof. If S_L is an arborescence in G_L with the above properties, then S_L is necessarily a union of directed r-i^h-paths for some $i^h \in V_h$ and directed r-i'-paths for some $i' \in T_p$. By construction of A_L, arcs to nodes in $V_1 \cup \cdots \cup V_H$ always go to the next layer, so that all nodes in $V_1 \cup \cdots \cup V_{H-1}$ must be reached from r using at most H arcs. As for all $i \in T_p$, S_L contains at most one of the i^h, so that, replacing i^h by i, we obtain an arborescence in S with the desired properties. For the converse, we only need to track the layer count. □

6.1.2 Model formulation

Before describing the model formulation in detail, we define an additional set of variables X for convenience. For each arc $(i, j) \in A$ with $i, j \in T_p$, let

$$X_{ij} \coloneqq \sum_{h=2}^{H} x_{i_{h-1} j_h}, \tag{39}$$

and for each arc $(i, i') \in A$ with $i \in T_p$ and $i' \in T_f$, let

$$X_{ii'} \coloneqq \sum_{h=1}^{H} x_{i_h i'}. \tag{40}$$

These variables link an arc of the original graph with the arcs in the layered graph: If arc $(i, j) \in A$ is chosen for the original graph ($X_{ij} = 1$), it must connect two consecutive layers. If the fixed terminal $i' \in T_f$ is not connected via the root node r ($x_{ri'} = 0$ and $X_{ii'} = 1$), it must be connected via any layer node i_h with $h \in \{1, \ldots, H\}$. The hop-constrained QSTPI is formulated as:

QSTPI-HOP:

$$\min \quad c^T x \tag{41}$$

$$\text{s.t.} \quad x(\delta^-(W)) \geq 1 \qquad \forall W \subset V_L, r \notin W, |W \cap T_f| \geq 1 \tag{42}$$

$$\sum_{i \in T_p} q_v x_{ri'} + I_{\text{tot}} \leq \sum_{i \in T_p} q_i - Q \tag{43}$$

$$I_{\text{tot}} \geq \sum_{i \in T_p} \sum_{j \neq i \in T_p} I_{ij} X_{ii'} X_{jj'} \tag{44}$$

$$X_{ii'} + X_{jj'} \leq 1 \qquad \forall i, j \in T_p, i \neq j, d_{ij} < D_{\min} \tag{45}$$

$$\sum_{(i,j) \in A, j \in T_p} X_{ij} \leq 1 \qquad i \in T_p \tag{46}$$

$$X_{ij} = \sum_{h=2}^{H} x_{i_{h-1} j_h} \qquad \forall (i,j) \in A, i, j \in T_p \tag{47}$$

$$X_{ii'} = \sum_{h=1}^{H} x_{i_h i'} \qquad \forall i \in T_p \tag{48}$$

$$x_{ij} \in \{0,1\} \qquad \forall (i,j) \in A_L \tag{49}$$

Constraint (42) – (45) are inherited from the QSTPI-TRANS (see Section 3.3), i.e., each terminal $i' \in T_f$ must be connected to the root node r in (42), the quota must be fulfilled in (43), the total interference is determined by the chosen turbines in (44), and only one of two turbines that are too close to each other can be chosen in (45). Constraint (46) ensures the radial topology by limiting the number of outgoing arcs of any i^h to one. Note that (46) technically still allows that the last node in a path of the radial layout belongs to a potential terminal that has been visited before, thus not contributing to the quota, but possibly increasing the interference and the total cost, so that this phenomenon will not occur in an optimal solution. Similarly, $X_{ii'} > 1$ is not explicitly excluded in our model, but we can set $X_{ii'} = 1$ without further consequences in a post-processing step. Constraints (47) and (48) link the arc variables of the layered graph to the arcs variables in the original graph. Note that (46) implies $X_{ij} \leq 1$.

6.1.3 Additionally valid inequalities for string topologies

Subtour elimination. The following generalized subtour elimination constraint of size two is valid (Sinnl & Ljubić, 2016):

$$X_{ij} + X_{ji} \leq X_{ii'} \quad \forall (i,j) \in A, i \in T_p. \tag{50}$$

Constraint (50) states that whenver two nodes $i, j \in T_p$ are connected, then the fixed terminal i' must be connected from some layer.

Using the fact that there is an optimal solution, where $i' \in T_f$ is either connected directly via the root ($x_{ri'} = 1$) or via a layer node ($X_{ii'} = 1$), we can use $x_{ri'} + X_{ii'} = 1$ to rewrite constraint (50) as

$$X_{ij} + X_{ji} \leq 1 - x_{ri'} \qquad \forall (i,j) \in A, i, j \in T_p, \tag{51}$$

which empirically yields much better performance.

Linking the layers. So far, our integer program has considerable symmetry with respect to the layers. We will now exploit that we are aiming at a radial topology. For every layer node i_h, let

$$y_{i_h} \coloneqq \sum_{(j_{h-1}, i_h) \in A_h} x_{j_{h-1} i_h}$$

describe whether we choose the potential terminal $i \in T_p$ in layer h. As we build an arborescence, each node cannot have more than one incoming arc, thus, y_v^h describes if a node v appears in layer h. Considering a node v^h in layer $h \in \{1, \ldots, H-1\}$, an arc only can leave the node if it appears in that layer. Thus, the following holds

$$\sum_{(i_h, j_{h+1}) \in A_h} x_{i_h j_{h+1}} \leq y_{i_h} \qquad \forall i_h \in V_h, h \in \{1, \ldots, H-1\}. \tag{52}$$

Additionally, in a radial layout, a layer $h \in \{2, \ldots, H\}$ cannot contain more nodes than its previous layer $h-1$, as, otherwise, the solution would contain a node in layer $h-1$ having more than one arc to layer h:

$$\sum_{i_h \in V_h} y_{i_h} \leq \sum_{i_{h-1} \in V_{h-1}} y_{i_{h-1}} \qquad \forall h \in \{2, \ldots, H\}. \tag{53}$$

Bounding the number of chosen turbines. Considering the quota constraint

$$\sum_{i \in T_p} q_i y_i - I_{\text{tot}} \geq Q, \tag{54}$$

cf. (20), we can impose a valid upper bound on the number of chosen turbines. Let $q^{\min} \coloneqq \left(q_1, \ldots, q_{|T_p|}\right)$ be sorted in non-decreasing order, $q_1 \leq q_2 \leq \cdots \leq q_{|T_p|}$, and let $q_i^{\min}$ be the i-th entry of $q^{\min}$. Let $Q^{\min}(k) \coloneqq \sum_{i=1}^{k} q_i^{\min}$ be the minimum amount of quota collected by k turbines. Given the interference matrix I, let $I_i^{\max}(k)$ denote the sum of the $k-1$ maximum I_{ij} values of row i. We can then compute the worst case induced total interference $I^{\max}(k) \coloneqq k \max_i I_i^{\max}(k)$. A valid upper bound on the

number of chosen turbines is given by the smallest k^{ub} for which

$$Q^{\min}(k^{\text{ub}}) - I^{\max}(k^{\text{ub}}) \geq Q \tag{55}$$

$$\text{and} \quad Q^{\min}(k^{\text{ub}} - 1) - I^{\max}(k^{\text{ub}} - 1) < Q. \tag{56}$$

This upper bound in mind, given H layers and the fact that no layer can contain more nodes than its predecessor, see (53), the following holds:

$$\sum_{(i_{H-1}, j_H) \in A_H} x_{i_{H-1} j_H} \leq \left\lfloor \frac{k^{\text{ub}}}{H} \right\rfloor, \tag{57}$$

$$\sum_{(i_H, i') \in A_0, i_H \in V_H} x_{i_H i'} \leq \left\lfloor \frac{k^{\text{ub}}}{H} \right\rfloor. \tag{58}$$

Inequality (57) limits the number of arcs connected to the last layer and (58) limits the nodes that connect to the terminals from the last layer. For instance, if $k^{\text{ub}} = 22$ and $H = 6$, then at most $\left\lfloor \frac{22}{6} \right\rfloor = \lfloor 3.667 \rfloor = 3$ turbines appear in the last layer, and, thus, only three arcs will enter the last layer.

6.2 Computational Study

We integrated the QSTPI-HOP model (41)–(49) including (51), (52), (53) and (57), (58) into our SCIP-JACK framework. During the transformation into the layered graph, the following *directed root cost test* decides if arcs are included in the layered graph (see, e.g., Gouveia et al. (2011)): If the arcs $(i, j), (r, j) \in A$ exist, $i \neq r$ and it holds that $c_{ij} \geq c_{rj}$, the arc (i, j) is removed during the transformation, as node j will always be connected to the root node. The shortest-path-based primal heuristic is extended to handle the layered graph, the hop-constraints, and the radial layout.

As before the bilinear products $X_{ii'} X_{jj'}$ in (44) are linearized, but, instead of adding them all in the beginning as done before by (16) – (18), they are separated on-the-fly during the the B&C algorithm. For the computational experiments in this section, we use *pseudocost branching with strong branching initialization* presented in Pedersen et al. (2025). Until depth three of the B&B tree the pseudcosts are initialized by performing strong branching on the ten fractional vertices which contribute the most interference in the current LP-solution. Afterwards, the pseudocosts are used. For more details, the reader is referred to Achterberg et al. (2005) and Pedersen et al. (2025). We call the resulting implementation `QSTPI-HOP`.

According to Cazzaro et al. (2023), each cable allows for a maximum of six 15 MW wind turbines, resulting in a hop-limit $H = 6$. In the flow-based formulation this cable capacity is simply introduced in Eq. (25). We refer to this capacitated flow-based formulation as `FLOW-CAPA`.

Setting	# solved	time [s]	B&B nodes
QSTPI-HOP	223	27.20	85.11
FLOW-CAPA	220	194.77	897.29
QSTPI-Ilb*	239	5.85	69.69
FLOW-Ilb*	228	384.08	2224.40

Table 1: Computational results: The second column shows the number of instances solved to optimality; the third column shows the mean time over all instances solved by both settings for the capacitated approach using the shifted geometric mean with a shift of one second; Last column shows the number of average B&B nodes over all instances solved by both settings using the shifted geometric mean with a shift of 100. *Row three and four summarize the results from the uncapacitated approach, see Section 5.

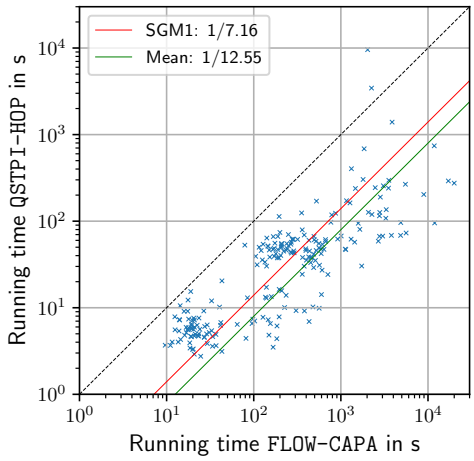


Figure 10: Comparison of running times between QSTPI-HOP and FLOW-CAPA of all instances solved by both settings. The dotted line shows the break even line; the *SGM1* line represents the shifted geometric mean of the speedup with a shift of one second; the *Mean* line represents the arithmetic mean of the speedup.

As a test set we consider the small- and medium-sized instances of Section 5.3. We use the same computational settings as before. We use the shifted geometric mean (Achterberg, 2007) with a shift of one second regarding time and a shift of 100 regarding the B&B nodes.

The results are summarized in Table 1. The computational time for each individual instance needed by the two settings is shown in Fig. 10. Our proposed approach outperforms the flow-based formulation in all categories: It solves three more instances, it shows an average speedup of 7.16 in terms of time, and it reduces the number of B&B nodes by a factor of 10.54. Furthermore, QSTPI-HOP is on all but two instances faster than FLOW-CAPA.

Comparing QSTPI-HOP and FLOW-CAPA to their uncapacitated versions, the flow-based approach benefits by introducing a capacity on the flow variables as the solution space is reduced. On the common solved instances, the computational time is reduced by around 50%, although eight instances are solved less in total. The opposite can

be observed for the QSTPI-HOP as we solve an enlarged graph: The computing time is increased by a factor of around five compared to the QSTPI-Ilb.

7 Price of sequentiality

The common approach of solving the wind farm layout and cable routing problem sequentially can lead to suboptimal solutions (Cazzaro, 2022; Cazzaro et al., 2023; Fischetti, 2017). As a final evaluation, we illustrate the price of that sequentiality. We compare:

- the sequential approach of first finding a feasible positioning minimizing the turbine costs, followed by solving the capacitated routing problem (SEQ),
- solving the capacitated routing problem on the positions found by the QSTPI (QSTPI-SEQ), and
- solving the hop-constrained QSTPI (QSTPI-HOP).

Concerning SEQ, finding a feasible positioning V^* minimizing the turbine costs is done by

$$\min \quad \sum_{i \in T_p} b_i y_i \tag{59}$$

$$\text{s.t.} \quad (31) - (35), \tag{60}$$

and is solved with GUROBI 11.01. Having found V^* by SEQ or QSTPI-SEQ, the capacitated routing problem is solved on the complete subgraph $G^* = (V^*, A^*, r)$ containing all chosen turbines V^* and the substation r and is formulated as

$$\min \quad c^T x \tag{61}$$

$$\text{s.t.} \quad \sum_{a \in \delta^-(i)} f_a - \sum_{a \in \delta^+(i)} f_a = 1 \qquad \forall i \in V^* \setminus \{r\} \tag{62}$$

$$f_a \leq H x_a \qquad \forall a \in A^* \tag{63}$$

$$x_a \in \{0,1\}, f_a \in \mathbb{R}_{\geq 0} \qquad \forall a \in A^* \tag{64}$$

and is solved with GUROBI 11.01.

The test set consists of the 214 small- and medium-sized instances solved to optimality by both QSTPI-HOP and QSTPI-Ilb. The relative cost change is calculated by

$$c_{\text{rel}}^{\text{red}} = \left(1 - \frac{c_{\texttt{SEQ}}}{c_{\texttt{QSTPI}}}\right) \cdot 100\%,$$

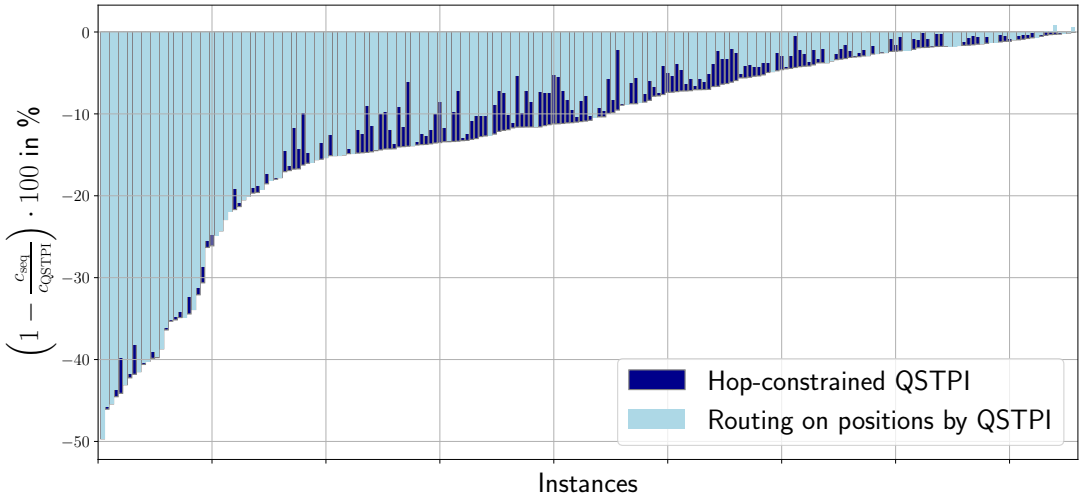


Figure 11: Cost reduction by using chosen turbines of `QSTPI-SEQ` (light) and additionally by using `QSTPI-HOP` (dark) compared to `SEQ`

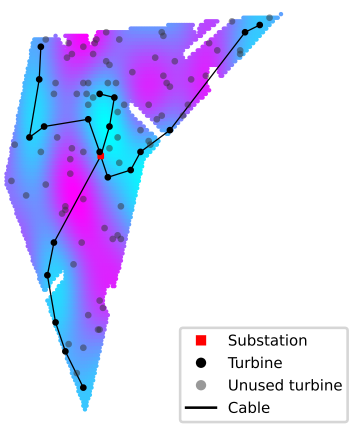
where $c_{\texttt{SEQ}}$ refers to the sequential approach of first minimizing turbine costs and the cable costs, and $c_{\texttt{QSTPI}}$ to the respective `QSTPI` approach.

Fig. 11 presents the relative cost change of our novel approaches compared to the sequential approach for each instance. Already with `QSTPI-SEQ`, the costs are reduced by 10.73% on average, while two instances have a worse objective compared to the sequential approach, but with costs increasing by less than one percent. Using `QSTPI-HOP` instead of the sequential approach the costs are reduced by 12.25% on average, always improving the solution. In 43 instances the `QSTPI-SEQ` gives the same solution as `QSTPI-HOP`. Fig. 12 shows the different solutions for an exemplary instance.

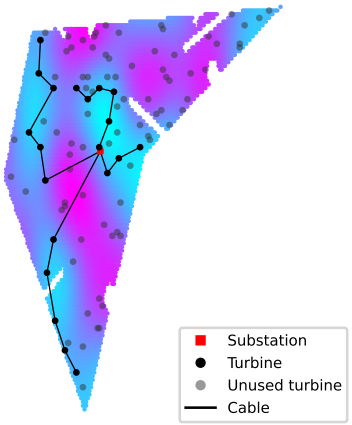
Already Cazzaro and Pisinger (2022) showed that an integrated approach improves the solution quality of their heuristic. We conclude that the price of sequentiality is significant and justifies invoking an integrated approach. Therein, the QSTPI can provide both a lower bound on the optimal solution of the capacitated IWLCR problem as well as serve as a heuristic for the problem.

8 Conclusion and outlook

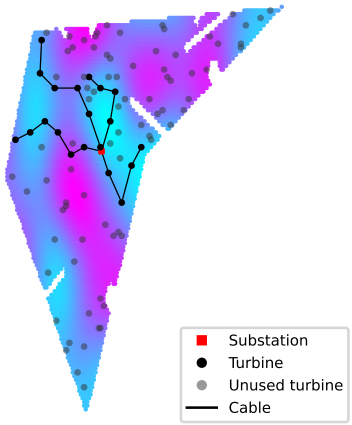
Current methods for solving the integrated wind farm layout and cable routing problem (IWFLCR) are either heuristic approaches or are limited to very small instances. This paper proposes a new approach for optimizing simultaneously the turbine locations and their connection to the grid's substation. The core is the QSTPI, an extension of the quota Steiner tree problem (QSTP). Besides the quota and connectivity constraints from the QSTP, the formulation includes the wake effect caused by a turbine reducing the profit of the surrounding turbines, and can also guarantee a minimum



(a) Solution for `SEQ`.



(b) Solution `QSTPI-SEQ`; cost reduction 5.64% compared to `SEQ`.



(c) Solution for `QSTPI-HOP`; cost reduction 9.94% compared to `SEQ`.

Figure 12: Three solutions for the same hop-constrained QSTPI instance showing the price of sequentiality.

distance between chosen turbines. Both are vital technical constraints for the WFLO. We have upgraded the state-of-the-art STP-solver SCIP-JACK so that it can handle the interference-constrained variant of the QSTP. Where the basic QSTP is of great advantage compared to standard flow-based formulation solvable by general MIP solvers, its interference-constrained variant bears no advantage at first. On the one hand this shows the immense difficulty of the quadratic constraints induced by the wake effect, and on the other hand the great improvements made by generic MIP solvers for these types of constraints.

Inspired by known cutting planes for the interference constraint, we therefore propose a solution strategy of splitting the problem based on the total interference. Although two possibly challenging subproblems must be solved, we demonstrate on a large testset that a) the solution strategy speedups the solution process and makes most of the instances solvable, and b) the advanced version of SCIP-JACK outperforms standard MIP solvers and is capable of finding optimal solutions for large-sized instances, where general state-of-the-art MIP solvers struggle. In particular, already using a small lower bound on the total interference results in a immense speed-up (see results for `QSTPI-Ilb`). By splitting the problem using valid lower bounds calculated by an auxiliary MIP the number of solvable instances increases even further (see results for `STPI-S-minI-1800`). Finally, we illustrate that to avoid excessive design costs for wind farm projects, an integrated view of the WFLO and WFCR is beneficial. This conclusion remains valid when cable capacity limits have to be respected.

Although our presented extensions to SCIP-JACK outperforms general MIP solvers in terms of computational time as well as in solving large-sized instances, there are several instances of the considered dataset that have not been solved to optimality. As stated in Section 5.1, the chosen primal heuristic often improves the primal solution only at the root node of the branch-and-bound tree, more efficient and suitable primal heuristics should be investigated.

Furthermore, improving the dual bound of the presented formulation is vital. However, the cuts proposed by Fischetti and Fischetti (2022) introduce very dense rows in the constraint matrix, making the LP-relaxation harder to solve, and preliminary experiments show that while improving the root node relaxation, the overall performance worsens.

In this study, our novel approaches is validated on instances which are based on generic offshore wind farm areas which are characterized by a relatively uniform terrain – although different foundation costs were considered –, wake expansion behavior, and using only a single turbine type. Nevertheless, we believe the proposed methodology is not strictly limited to the offshore case and could, for example, be applied for onshore wind farm planning in the future. By explicitly modeling heterogeneous terrains, varying turbine heights, complex wake calculations, and stricter layout constraints in

the input data, the workflow of the QSTPI and its hop-constrained variant provides an interesting approach.

We believe that the hop-constrained QSTPI approach can be further sharpened to integrate further practically relevant constraints, e.g., substation capacity limits that limit the number of connected cable strings, or generalized to other types of network topologies. Furthermore, it would be interesting to compare our optimization model with state-of-the-art simulation tools, e.g., FLORIS[3].

Acknowledgements

The work for this article has been conducted in the Research Campus MODAL funded by the German Federal Ministry of Education and Research (BMBF) (fund numbers 05M14ZAM, 05M20ZBM, 05M2025). We thank Davide Cazzaro for making his code available to compute the interference matrix for the used dataset.

CRediT statement

Conceptualization: J.P., T.K.; data curation: J.P.; formal analysis: J.P., N.L.; investigation: J.P.; methodology: J.P., N.L., D.R.; software: J.P., D.R.; validation: J.P.; visualization: J.P.; writing – original draft: J.P.; writing – review and editing: J.P., N.L., D.R.

References

Achterberg, T. (2007). *Constraint integer programming* [Doctoral dissertation, Technische Universität Berlin]. https://opus4.kobv.de/opus4-zib/frontdoor/index/index/docId/1018 (Accessed on 12/04/2024).

Achterberg, T., Koch, T., & Martin, A. (2005). Branching rules revisited. *Operations Research Letters*, *33*(1), 42–54. https://doi.org/10.1016/j.orl.2004.04.002

Archer, C. L., Vasel-Be-Hagh, A., Yan, C., Wu, S., Pan, Y., Brodie, J. F., & Maguire, A. E. (2018). Review and evaluation of wake loss models for wind energy applications. *Applied Energy*, *226*, 1187–1207. https://doi.org/https://doi.org/10.1016/j.apenergy.2018.05.085

Archer, R., Nates, G., Donovan, S., & Waterer, H. (2011). Wind turbine interference in a wind farm layout optimization mixed integer linear programming model. *Wind Engineering*, *35*(2), 165–175. https://doi.org/10.1260/0309-524X.35.2.165

Bay, C. J., Fleming, P., Doekemeijer, B., King, J., Churchfield, M., & Mudafort, R. (2023). Addressing deep array effects and impacts to wake steering with the cumulative-curl wake model. *Wind Energy Science*, *8*(3), 401–419. https://doi.org/10.5194/wes-8-401-2023

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., ... Witzig, J. (2023). Enabling Research through the SCIP Optimization Suite 8.0. *ACM Transactions on Mathematical Software*. https://doi.org/10.1145/3585516
Just Accepted.

[3]https://github.com/NREL/floris, accessed on: 12/02/2025

Bolukbasi, G., & Kocaman, A. S. (2018). A prize collecting Steiner tree approach to least cost evaluation of grid and off-grid electrification systems. *Energy*, *160*, 536–543. https://doi.org/10.1016/j.energy.2018.07.029

Cazzaro, D., & Pisinger, D. (2022). Variable neighborhood search for large offshore wind farm layout optimization. *Computers & Operations Research*, *138*. https://doi.org/10.1016/j.cor.2021.105588

Cazzaro, D. (2022). *Unified optimization for offshore wind farm design* [Doctoral dissertation]. Technical University of Denmark.

Cazzaro, D., Fischetti, M., & Fischetti, M. (2020). Heuristic algorithms for the wind farm cable routing problem. *Applied Energy*, *278*, 115617. https://doi.org/10.1016/j.apenergy.2020.115617

Cazzaro, D., Koza, D. F., & Pisinger, D. (2023). Combined layout and cable optimization of offshore wind farms. *European Journal of Operational Research*, *311*(1), 301–315. https://doi.org/10.1016/j.ejor.2023.04.046

Cazzaro, D., Trivella, A., Corman, F., & Pisinger, D. (2022). Multi-scale optimization of the design of offshore wind farms. *Applied Energy*, *314*, 118830. https://doi.org/10.1016/j.apenergy.2022.118830

Costa, A. M., Cordeau, J.-F., & Laporte, G. (2009). Models and branch-and-cut algorithms for the steiner tree problem with revenues, budget and hop constraints. *Networks*, *53*(2), 141–159. https://doi.org/10.1002/net.20274

Donovan, S. (2005). Wind farm optimization. *Proceedings of the 40th annual ORSNZ conference*, 196–205.

Fischetti, M. (2021). On the optimized design of next-generation wind farms. *European Journal of Operational Research*, *291*(3), 862–870. https://doi.org/10.1016/j.ejor.2020.10.048

Fischetti, M., & Pisinger, D. (2018). Optimizing wind farm cable routing considering power losses. *European Journal of Operational Research*, *270*(3), 917–930. https://doi.org/10.1016/j.ejor.2017.07.061

Fischetti, M. (2017). *Mathematical programming models and algorithms for oshore wind park design* [Doctoral dissertation]. DTU Management Engineering.

Fischetti, M., & Fischetti, M. (2022). Integrated Layout and Cable Routing in Wind Farm Optimal Design. *Management Science*. https://doi.org/10.1287/mnsc.2022.4470

Fischetti, M., & Monaci, M. (2016). Proximity search heuristics for wind farm optimal layout. *Journal of Heuristics*, *22*, 459–474.

Fischetti, M., & Pisinger, D. (2019). Mathematical optimization and algorithms for offshore wind farm design: An overview. *Business & Information Systems Engineering*, *61*, 469–485.

Fischetti, M., Monaci, M., & Salvagnin, D. (2012). Three ideas for the quadratic assignment problem. *Operations research*, *60*(4), 954–964. https://doi.org/10.1287/opre.1120.1073

Fleming, P. A., Ning, A., Gebraad, P. M., & Dykes, K. (2016). Wind plant system engineering through optimization of layout and yaw control. *Wind Energy*, *19*(2), 329–344. https://doi.org/10.1002/we.1836

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

Gebraad, P., Thomas, J. J., Ning, A., Fleming, P., & Dykes, K. (2017). Maximization of the annual energy production of wind power plants by optimization of layout and yaw-based wake control. *Wind Energy*, *20*(1), 97–107. https://doi.org/10.1002/we.1993

Gebraad, P. M. O., & van Wingerden, J. W. (2014). A control-oriented dynamic model for wakes in wind plants. *Journal of Physics: Conference Series*, *524*(1), 012186. https://doi.org/10.1088/1742-6596/524/1/012186

González, J. S., Payán, M. B., Santos, J. M. R., & González-Longatt, F. (2014). A review and recent developments in the optimal wind-turbine micro-siting problem. *Renewable and Sustainable Energy Reviews*, *30*, 133–144. https://doi.org/10.1016/j.rser.2013.09.027

Gouveia, L., Simonetti, L., & Uchoa, E. (2011). Modeling hop-constrained and diameter-constrained minimum spanning tree problems as steiner tree problems over layered graphs. *Math. Program.*, *128*, 123–148. https://doi.org/10.1007/s10107-009-0297-2

Gurobi Optimization, LLC. (2024). Gurobi Optimizer Reference Manual, Version 11.0.1. https://www.gurobi.com.

Hou, P., Zhu, J., Ma, K., Yang, G., Hu, W., & Chen, Z. (2019). A review of offshore wind farm layout optimization and electrical system design methods. *Journal of Modern Power Systems and Clean Energy*, *7*(5), 975–986. https://doi.org/10.1007/s40565-019-0550-5

IBM ILOG. (2022). V12.10: User's Manual for CPLEX. https://www.ibm.com/analytics/cplex-optimizer.

Jansen, M., Staffell, I., Kitzing, L., Quoilin, S., Wiggelinkhuizen, E., Bulder, B., Riepin, I., & Müsgens, F. (2020). Offshore wind competitiveness in mature markets without subsidy. *Nature Energy*, *5*(8), 614–622. https://doi.org/10.1038/s41560-020-0661-2

Jensen, N. (1983). *A note on wind generator interaction*. Risø National Laboratory.

Johnson, D. S., Minkoff, M., & Phillips, S. (2000). The prize collecting Steiner tree problem: Theory and practice. *Proceedings of the Eleventh Annual ACM-SIAM Symposium on Discrete Algorithms*, 760–769.

Karp, R. M. (1972). Reducibility among Combinatorial Problems. In R. E. Miller, J. W. Thatcher, & J. D. Bohlinger (Eds.), *Complexity of Computer Computations: Proceedings of a symposium on the Complexity of Computer Computations, held March 20–22, 1972, at the IBM Thomas J. Watson Research Center, Yorktown Heights, New York, and sponsored by the Office of Naval Research, Mathematics Program, IBM World Trade Corporation, and the IBM Research Mathematical Sciences Department* (pp. 85–103). Springer US. https://doi.org/10.1007/978-1-4684-2001-2_9

King, J., Fleming, P., King, R., Martínez-Tossas, L. A., Bay, C. J., Mudafort, R., & Simley, E. (2021). Control-oriented model for secondary effects of wake steering. *Wind Energy Science*, *6*(3), 701–714. https://doi.org/10.5194/wes-6-701-2021

Klimm, F., Toledo, E. M., Monfeuga, T., Zhang, F., Deane, C. M., & Reinert, G. (2020). Functional module detection through integration of single-cell RNA sequencing data with protein–protein interaction networks. *BMC Genomics*, *21*(1), 756. https://doi.org/10.1186/s12864-020-07144-2

Koch, T., Berthold, T., Pedersen, J., & Vanaret, C. (2022). Progress in mathematical programming solvers from 2001 to 2020. *EURO Journal on Computational Optimization*, *10*, 100031. https://doi.org/10.1016/j.ejco.2022.100031

Leitner, M., Ljubic, I., Luipersbeck, M., Prossegger, M., & Resch, M. (2014, January). *New Real-world Instances for the Steiner Tree Problem in Graphs* (tech. rep.). https://ivanaljubic.github.io/docs/stp/realworld-stp-report-short.pdf (Accessed on 11/04/2025).

Ljubić, I. (2021). Solving Steiner trees: Recent advances, challenges, and perspectives. *Networks*, *77*(2), 177–204. https://doi.org/10.1002/net.22005

Ljubić, I., Weiskircher, R., Pferschy, U., Klau, G. W., Mutzel, P., & Fischetti, M. (2006). An Algorithmic Framework for the Exact Solution of the Prize-Collecting Steiner Tree Problem. *Mathematical Programming*, *105*(2), 427–449. https://doi.org/10.1007/s10107-005-0660-x

McCormick, G. P. (1976). Computability of global solutions to factorable nonconvex programs: Part i—convex underestimating problems. *Mathematical programming*, *10*(1), 147–175.

Mittelmann, H. (2020). Benchmarks for optimization software. http://plato.asu.edu/bench.html.

Our World in Data. (2021). Installed wind energy capacity. https://ourworldindata.org/grapher/cumulative-installed-wind-energy-capacity-gigawatts?country=~OWID_WRL (Accessed on 08/21/2021).

Pedersen, J., Lindner, N., Rehfeldt, D., & Koch, T. (2025). *Comparing branching rules for the quota steiner tree problem with interference* (tech. rep. No. 25-16). ZIB. Takustr. 7, 14195 Berlin. https://opus4.kobv.de/opus4-zib/frontdoor/index/index/docId/10125 (Accessed on 11/14/2025).

Pedersen, J., & Ljubić, I. (2025). Prize-collecting steiner tree problem and its variants. In P. M. Pardalos & O. A. Prokopyev (Eds.), *Encyclopedia of optimization* (pp. 1–11). Springer International Publishing. https://doi.org/10.1007/978-3-030-54621-2_869-1

Pedersen, J., Weinand, J. M., Syranidou, C., & Rehfeldt, D. (2024). An efficient solver for large-scale onshore wind farm siting including cable routing. *European Journal of Operational Research*. https://doi.org/10.1016/j.ejor.2024.04.026

Pérez-Rúa, J.-A., & Cutululis, N. A. (2019). Electrical cable optimization in offshore wind farms—a review. *IEEE Access*, *7*, 85796–85811. https://doi.org/10.1109/ACCESS.2019.2925873

Pryor, S. C., Barthelmie, R. J., Bukovsky, M. S., Leung, L. R., & Sakaguchi, K. (2020). Climate change impacts on wind power generation. *Nature Reviews Earth & Environment*, *1*(12), 627–643. https://doi.org/10.1038/s43017-020-0101-7

Pryor, S. C., Barthelmie, R. J., & Shepherd, T. J. (2021). Wind power production from very large offshore wind farms. *Joule*, *5*(10), 2663–2686. https://doi.org/10.1016/j.joule.2021.09.002

Rehfeldt, D. (2021). *Faster algorithms for Steiner tree and related problems: From theory to practice* [Doctoral dissertation, Technische Universität Berlin]. https://opus4.kobv.de/opus4-zib/frontdoor/index/index/docId/8514 (Accessed on 02/24/2023).

Rehfeldt, D., & Koch, T. (2022). On the Exact Solution of Prize-Collecting Steiner Tree Problems. *INFORMS Journal on Computing*, *34*(2), 872–889. https://doi.org/10.1287/ijoc.2021.1087

Rehfeldt, D., & Koch, T. (2023). Implications, conflicts, and reductions for steiner trees. *Mathematical Programming*, *197*, 903–966. https://doi.org/10.1007/s10107-021-01757-5

Rehfeldt, D., Koch, T., & Maher, S. J. (2019). Reduction techniques for the prize collecting Steiner tree problem and the maximum-weight connected subgraph problem. *Networks*, *73*(2), 206–233. https://doi.org/10.1002/net.21857

Ridremont, T. (2019, April). *Design of robust networks. Application to the design of wind farm cabling networks.* [Doctoral dissertation, Polytechnique Montréal]. https://publications.polymtl.ca/3889/ (Accessed on 03/02/2023).

Samorani, M. (2013). The wind farm layout optimization problem. In P. M. Pardalos, S. Rebennack, M. V. F. Pereira, N. A. Iliadis, & V. Pappu (Eds.), *Handbook of wind power systems* (pp. 21–38). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-41080-2_2

Sinnl, M., & Ljubić, I. (2016). A node-based layered graph approach for the steiner tree problem with revenues, budget and hop-constraints. *Math. Program. Comp.*, *8*(4), 461–490. https://doi.org/10.1007/s12532-016-0102-1

Takahashi, H., & Matsuyama, A. (1980). An approximate solution for the Steiner problem in graphs. *Mathematica Japonica*, (6), 573–577.

Turner, S., Romero, D., Zhang, P., Amon, C., & Chan, T. (2014). A new mathematical programming approach to optimize wind farm layouts. *Renewable Energy*, *63*, 674–680. https://doi.org/10.1016/j.renene.2013.10.023

Veers, P., Dykes, K., Lantz, E., Barth, S., Bottasso, C. L., Carlson, O., Clifton, A., Green, J., Green, P., Holttinen, H., Laird, D., Lehtomäki, V., Lundquist, J. K., Manwell, J., Marquis, M., Meneveau, C., Moriarty, P., Munduate, X., Muskulus, M., … Wiser, R. (2019). Grand challenges in the science of wind energy. *Science (New York, N.Y.)*, *366*(6464). https://doi.org/10.1126/science.aau2027

Wiser, R., Jenni, K., Seel, J., Baker, E., Hand, M., Lantz, E., & Smith, A. (2016). Expert elicitation survey on future wind energy costs. *Nature Energy*, *1*(10). https://doi.org/10.1038/nenergy.2016.135

Wiser, R., Rand, J., Seel, J., Beiter, P., Baker, E., Lantz, E., & Gilman, P. (2021). Expert elicitation survey predicts 37% to 49% declines in wind energy costs by 2050. *Nature Energy*. https://doi.org/10.1038/s41560-021-00810-z

Wong, R. T. (1984). A dual ascent approach for steiner tree problems on a directed graph. *Mathematical Programming*, *28*(3), 271–287. https://doi.org/10.1007/BF02612335