Diese Arbeit wurde vorgelegt am
Lehrstuhl für Mathematik (MathCCES)

Optimale Kabelführung in Offshore-Windparks durch ganzzahlige lineare Optimierung

Optimal Cable Layout for Offshore Wind Farms by Integer Linear Optimization

Bachelorarbeit
Mathematik

August 2016

Vorgelegt von Presented by	Paula Cremerius Junkerstr. 23 52064 Aachen Matrikelnummer: 329302 paula.cremerius@rwth-aachen.de
Erstprüfer First examiner	Prof. Dr. Martin Frank Lehrstuhl für Mathematik (MathCCES) RWTH Aachen University
Zweitprüfer Second examiner	Prof. Dr. Erika Ábrahám Lehr- und Forschungsgebiet: Theorie der hybriden Systeme RWTH Aachen University
Koreferent Co-supervisor	Dipl.-Math. Dipl.-Inform. Pascal Richter Lehrstuhl für Mathematik (MathCCES) RWTH Aachen University

Eidesstattliche Versicherung

____________________________ ____________________________

Name, Vorname Matrikelnummer (freiwillige Angabe)

Ich versichere hiermit an Eides Statt, dass ich die vorliegende Arbeit/Bachelorarbeit/ Masterarbeit* mit dem Titel

____

____

____

selbständig und ohne unzulässige fremde Hilfe erbracht habe. Ich habe keine anderen als die angegebenen Quellen und Hilfsmittel benutzt. Für den Fall, dass die Arbeit zusätzlich auf einem Datenträger eingereicht wird, erkläre ich, dass die schriftliche und die elektronische Form vollständig übereinstimmen. Die Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

____________________________ ____________________________

Ort, Datum Unterschrift

*Nichtzutreffendes bitte streichen

Belehrung:

§ 156 StGB: Falsche Versicherung an Eides Statt

Wer vor einer zur Abnahme einer Versicherung an Eides Statt zuständigen Behörde eine solche Versicherung falsch abgibt oder unter Berufung auf eine solche Versicherung falsch aussagt, wird mit Freiheitsstrafe bis zu drei Jahren oder mit Geldstrafe bestraft.

§ 161 StGB: Fahrlässiger Falscheid; fahrlässige falsche Versicherung an Eides Statt

(1) Wenn eine der in den §§ 154 bis 156 bezeichneten Handlungen aus Fahrlässigkeit begangen worden ist, so tritt Freiheitsstrafe bis zu einem Jahr oder Geldstrafe ein.

(2) Straflosigkeit tritt ein, wenn der Täter die falsche Angabe rechtzeitig berichtigt. Die Vorschriften des § 158 Abs. 2 und 3 gelten entsprechend.

Die vorstehende Belehrung habe ich zur Kenntnis genommen:

____________________________ ____________________________

Ort, Datum Unterschrift

Contents

List of Figures

List of Tables

1. Introduction

Renewable energy sources have gained importance over the last years. In the result of this year's climate conference, the Paris Agreement, 170 countries proclaimed their intent to significantly reduce greenhouse gas emissions: in the second half of the 21^{st} century, there is to be a balance of emission and absorption of greenhouse gases [18]. In order to reach this goal, major parts of the conventional energy production need to be replaced by power plants working with renewable energies. A very promising alternative is wind energy. Large wind turbines based on so called monopiles are used to convert the kinetic energy of the wind into electric energy. Most common are onshore turbines, built within an accumulation of turbines called wind farm or standing alone as a single turbine. Usually they can be connected easily to a nearby power supply line. Thanks to intensive research, the producing costs of onshore wind have decreased a lot. As can be seen in figure 1, the costs of onshore wind energy on favourable sites are already lower than those in new hard coal or gas-steam power plants (combined cycle).

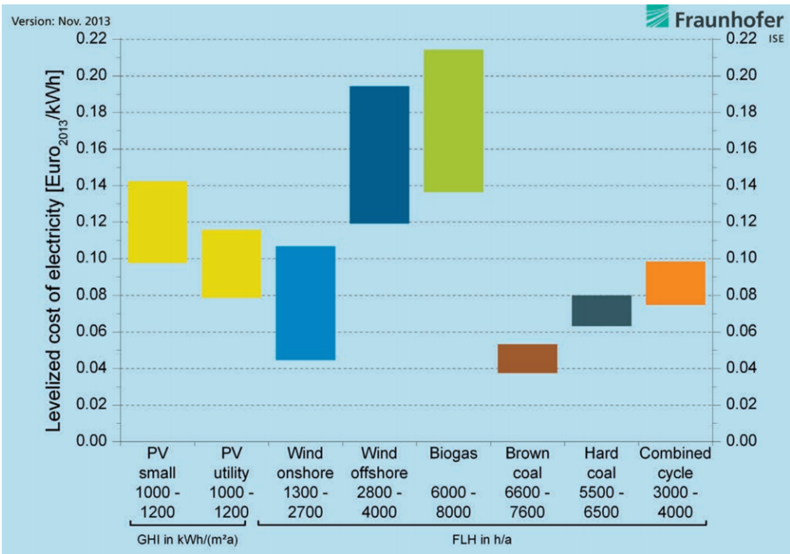


Figure 6: LCOE of renewable energy technologies and conventional power plants at locations in Germany in 2013. The value under the technology refers in the case of PV to solar irradiation (GHI) in kWh/(m²a); in the case of other technologies it reflects the number of FLH of the power plant per year. Specific investments are taken into account with a minimum and maximum value for each technology. Additional assumptions are presented in Table 3-7.

Figure 1: Levelized cost of energy in Germany cf. [10]

Investment in offshore wind energy, however, is a quite new development. In Germany it was only in 2009 that the first offshore wind farm was taken into operation [1]. The investment in offshore wind mainly started as residents were complaining more and more about wind turbines in their neighbourhood. The growing protests have actually become the limiting factor for a potential further expansion of onshore wind energy in Germany. Offshore wind farms, on the other hand, can be built so far in the sea that they cannot be seen from the shore. In addition offshore sites have the advantage of providing stronger and more constant wind. Disadvantages are the high installation costs and the costs of the connection to a potentially faraway power supply line. Nevertheless, it is expected that the price of offshore wind energy will decrease with further research on the topic.

A problem that has not yet been treated to full extent, is that of the cabling. Since subsea cables are very expensive, savings in the cable layout can significantly influence the overall costs. In order to keep these as low as possible, offshore turbines are never standing alone but are always installed in wind farms consisting of 80 to 120 turbines on average. Depending on the number of turbines, they have one or sometimes two transformer stations usually called substations, pictured on graphic 2 in green as the 'farm collection point'. The turbines are connected to the substation through a network of cables called the infield power collection. A high capacity submarine cable, the shore connection, is transferring the collected energy from the substation to the shore.

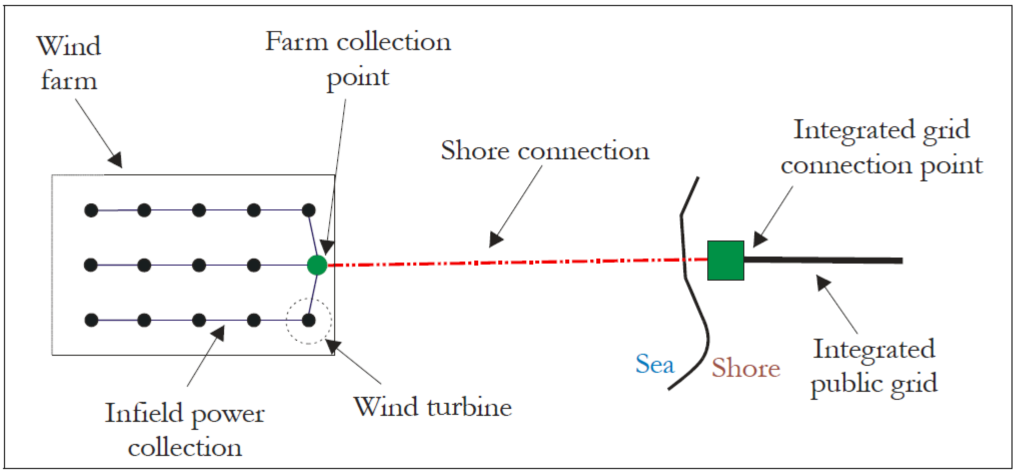


Figure 2: Wind farm layout cf. [17]

The aim of this work is to improve the infield power collection. What we will not deal with is the shore connection, since its design and cost are usually rather determined by external circumstances.
The cost of the shore connection cable could obviously be reduced by positioning the wind farm closer to the shore. This, however, would corrupt the natural coastline view, which would upset residents and have negative effects on tourism. Whether or not a wind farm can be built on a certain site is usually the decision of the respective governments. In Germany wind farms are usually not allowed within a 12 sea mile zone around the coast. In addition the positioning can be restricted by nature reserves.

Regarding the layouts of the cabling between the turbines and the substation, however, there is still a lot of room for improvement. Layouts for current wind farms are still developed manually. In [12] the problem of optimizing the costs of the infield power collection was already investigated more thoroughly. It amounts to the mathematical problem of finding a capacitated minimum spanning tree in a complete graph. This, however, is an NP-hard problem meaning that there is no known algorithm that can solve the problem in polynomial time [15]. In [12] the problem was approached using heuristics. By using an algorithm that has a runtime of only a few seconds, a layout was found that would have saved around three million euros compared to the actual layout. In the work it was assumed that the heuristic used, a version of the Esau Willams algorithm, is providing results that differ up to five percent from the optimal solution. Although the quality of the result of the heuristics was unclear, considering the huge amount of savings already made, further examination seemed worthwhile.

Therefore in this work the aim is to see whether the optimal layout can be found using optimization solvers. In addition, the quality of the solution provided by the heuristics from [12] can be assessed knowing the optimal solution. It can also be decided whether using the heuristic version might be acceptable for cases where short run time is desired.
Contrary to the task in [12], runtime should not be a limiting factor here, the aim is to find the best solution. Thereby we design the cost model as realistic as possible not using major simplifications.

There are several mathematical programs providing such optimization solvers. For this work we started using the solvers integrated into Matlab, then using Cplex and finally changing to Gurobi due to its good performance on large problems. In order to use a solver, it is necessary to first define a mathematical model describing the problem. There are different categories of models that can be solved with optimization solvers; for nearly all of them solvers are offered by the Matlab optimization tool.

	Objective Type			
Constraint Type	Linear	Quadratic	Smooth nonlinear	Nonsmooth
None	`linprog`	`quadprog`	`fminsearch`	`fminsreach`
Bound	`linprog`	`quadprog`	`fminbnd, fmincon`	`fminbnd`
Linear	`linprog`	`quadprog`	`fmincon, fseminf`	`ga, patternsearch`
General Smooth	`fmincon`	`fmincon`	`fmincon, fseminf`	`ga, patternsearch`
Discrete	`intlinprog`	`ga`	`ga`	`ga`

Table 1: Decision table for Matlab optimization solvers cf. [13]

At first we conducted some unpromising tests using a genetic algorithm (`ga`), a solver that can handle nonsmooth constraints as well as integer (discrete) constraints. It is obvious that to model the problem discrete constraints are needed. However, it became clear, that if possible solvers based on linear constraints, like `intlinprog`, were by far the better option. Therefore it would be the main difficulty to find a completely linear description of all the features that we want to include in the model.

2. Approach: Using Integer Linear Optimization

We want to solve our problem using an integer linear optimization solver. Therefore, firstly, an overview of basic optimization theory is given to show how exactly a problem needs to be formulated 'linearly' to be solved by an integer linear optimization solver. Then, after summing up the problem we want to solve, the key idea to describing it linearly is presented.

2.1. General formulation of an Integer Linear Optimization Problem

In an optimization problem we are dealing with an objective function or cost function $c^T x$ that shall be minimized or maximized. c is the vector containing information on the 'cost' of possible solutions, while x is the solution vector. The vector c is of the same length as x, the cost function is their scalar product. Additionally, there can be constraints on the solution x that in the case of a linear optimization have to be formulated linearly. That means that they can be written as $A_{eq}x = b_{eq}$ and $Ax \leq b$ where the A_{eq}, b_{eq}, A and b can be chosen freely with dimensions matching each other and the solution vector x. Separately an upper and lower bound for the solution vector can be defined by vectors lb and ub. Obviously they must be of the same dimension as x. If furthermore some or all of the entries of x are constraint to be integer, the problem we deal with is called an integer linear optimization problem. If we assume that we want to minimize the cost function, as in our case we do, it can be presented in the following way:

$$\text{minimize } c^T x \quad \text{while} \quad \left\{ \begin{array}{l} A_{eq}x = b_{eq} \\ Ax \leq b \\ lb \leq x \leq ub \\ \text{some or all } x_j \in \mathbb{Z} \end{array} \right\}.$$

With $x \in \mathbb{R}^m$ for a $m \in \mathbb{N}, A_{eq} \in \mathbb{R}^{n_1 \times m}, b_{eq} \in \mathbb{R}^{n_1}, A \in \mathbb{R}^{n_2 \times m}, b \in \mathbb{R}^{n_2},$ with $n_1, n_2 \in \mathbb{N}, lb, ub \in \mathbb{R}^m$ all as explained above. When modeling an optimization problem one defines $c, A_{eq}, b_{eq}, A, b, lb, ub$ and the entries of x that shall be integers. The solution of the problem is the vector x which minimizes the cost function while the constraints are fulfilled.

Shorthand for integer linear optimization, which is also called integer linear programming, we will occasionally write ILP. The abbreviation LP is referring to linear programming or linear optimization.

2.2. Problem definition

Our model is based on the one used in [12] but some additional features are added. The aim is to minimize the costs of the infield power collection of wind farms while respecting the constraints naturally given by a wind farm. Since we use our model to compare layout costs with each other during the optimization process, we do not consider those costs that occur in every layout such as those for the turbines, the substation or the shore connection cable. What we consider are:

- necessary cable meters to connect the turbines based on the topography of the seabed,
- cable prices according to the needed cable capacities,
- monetary loss due to ohmic and dielectric losses,
- costs to connect the cables to the turbines.

The constraints that need to be respected are:

- every turbine must be connected to the substation,
- every cable needs to withstand the energy of the number of turbines connected to it,
- cables are allowed to branch at the turbines,
- no cable crossings.

Additionally we implement some useful features for wind farm operators:

- optional restriction on incoming cables to a turbine / to the substation,
- choice of best cables (i.e. 3 out of 12),
- scaled prices for the cables.

2.3. Classification as a graph theory problem

To relate to existing literature on the topic, the classification of our problem as a graph theory problem is presented in a short excursion. First, some definitions as they can be found in [2, 5], are needed.
A **graph** $G = (V, E)$ defined mathematically is a number of nodes V and a number of edges $E \subset V \times V$. A **directed graph** means that the edges have an orientation so the edge (i, j) would not be equal to the edge (j, i).
A **tree** is a graph that does not contain loops and where all nodes are connected to the graph. Directed trees are also called rooted trees, containing a **root node** and edges orientated towards or away from the root. The nodes at the opposite end of the root, which are apart from the root the only ones that are connected to only one other node, are called **leaves**.
A **spanning tree** S of G is a subgraph of some graph G that includes all nodes ($S = (V, E'), E' \subset E$) and is a tree itself. If there are no superfluous edges, we call it a **minimum spanning tree**.
Finally there is the concept of a capacitated tree, meaning that in every subtree of the root node the sum of nodes is limited by a certain capacity k.

Finding the optimal cabling layout amounts to the problem of finding a specially restricted **capacitated minimum spanning tree**. The turbines are interpreted as nodes and the cable connections as the edges connecting the nodes. It contains a root node, namely the substation. We define the orientation to be towards the root node, in the direction of the current flow. Its capacity is limited by the biggest available cable. k is the number of turbines this cable can connect.
The different cabling layouts can be seen as minimum spanning trees, we are looking for the one with minimum costs according to the defined cost model while the constraints listed in the section above are fulfilled.
In graph theory graphs are often represented by an adjacency matrix. These quadratic matrices have as many rows and columns as the number of nodes in the graph. The entry (i, j) is one if node i and node j are connected, otherwise the entry is zero. We will be using this concept in the following.

2.4. Main idea for our linear approach

To use the concept of ILP the solution vector must contain all the information about the cable configuration. Let $n - 1$ be the number of turbines in our wind farm, so adding one substation we have n stations to connect in total. Here station 1 is going to be the substation and stations 2 to n will be the turbines.
To represent where cables are placed, imagine an adjacency matrix $A \in \{0, 1\}^{n \times n}$. $(A)_{i,j}$ is 1 when there is a cable from station i to station j and 0 otherwise.
To be able to integrate this information into the solution vector x, we vectorize the matrix A writing its columns into a large column vector. When explaining solution variables, it will often be easier to imagine them as a matrix, nevertheless, in the so-

lution vector they will always be in vectorized form.
However, with x just being the vectorized adjacency matrix, it is not possible to express linearly that all turbines must be connected to the substation. Neither does it depict that at the different positions different cable capacities are needed depending on how many turbines are already connected to the route.
We find that the hop-indexed formulation used in [4] is solving both problems. Expanding the concept of an adjacency matrix, an extra index is added whose value increases on cable routes towards the substation. We let x contain the variables

$$x_{ij}{}^{h} = \begin{cases} 1 & \text{a cable runs from } i \text{ to } j,\, i \text{ is the } h^{th} \text{ turbine on the route} \\ 0 & \text{otherwise} \end{cases}$$

$$\text{for } i = 1...n, j = 1...n, h = 1...k.$$

While the first two indices i and j still define the position of a cable, h contains the information on how many turbines are connected to the route, thus determining the necessary cable capacity. Furthermore this allows us to formulate the constraints on the layout linearly.

Extensions on the solution variables to consider all features of our model such as the linear implementation of the constraints are explained in detail in chapter 4.

2.5. Overview and realization

The optimization will be realized in six steps. We distinguish between the aspects that belong to the cost model, which will be represented in the cost vector c, and the constraints, which will be expressed in A_{eq}, b_{eq}, A and b. In the brackets on the right we mark from which step onwards the feature will be included in the implementation. Steps one to three derive from the model in [12], steps four to six have been added additionally.

Cost model

- Cable meters to connect the turbines based on the topography } Step 1
- Cable prices according to the needed cable capacities } Step 1
- Monetary loss due to ohmic and dielectric losses } Step 3
- Costs to connect the cables to the turbines } Step 4
- Scaled prices for the cables } Step 6

Optimizer: Constraints

Find a layout with minimal cost while considering the following:

• Every turbine must be connected to the substation • Every cable needs to withstand the energy of the number of turbines connected to it • Cables are allowed to branch	} Step 1
• No cable crossings	} Step 2
• Optional restriction on incoming cables to a turbine / to the substation	} Step 4
• Choice of best cables (i.e. 3 out of 12)	} Step 5

Using data from the wind farms Horns Rev 1 and Sandbank we will create examples to explain and visualize the results of every step.

The linear formulation of the costs and the data we use to calculate the costs in our examples will be treated in chapter 3. But first it will be explained how the optimization solvers which we will be using, manage to find the optimal solution to an ILP. The problem, as it is equivalent to finding a capacited minimum spanning tree, is an NP-hard one. Thus the acceptable runtimes that we will encounter for most of the examples are all the more impressive.

2.6. Integer linear optimization solvers

There is a large number of optimization software offering solvers for integer linear problems. In the course of this work solvers by Matlab, Cplex and Gurobi were used. Due to its good performance we mainly worked with the `intlinprogGurobi` solver, thus in the following we will explain its basic functionality as described in [8]. As most ILP solvers, those from Cplex and Matlab are also using many of the methods that will be mentioned.

2.6.1. Branch-and-Bound algorithm

The main part of the solver is the Branch-and-Bound algorithm. It starts by removing the integrality constraints and solving the so-called LP-relaxation of the problem. Since matrix derivatives can be used for this, the optimum in an LP-problem is usually found very quickly.

We then have a look at the integrality constraints. If all happen to be satisfied although they were not presumed, the LP-solution is also a solution of the ILP-problem. Otherwise we pick one of the integer constraint variables x that is fractional in the

LP-solution, so $n < x < n+1$ for an $n \in \mathbb{N}$. We then define new ILPs P_1 and P_2 that have the additional constraints (1) $x \geq n+1$ and (2) $x \leq n$ respectively. The ILP P_1 and P_1 are so called 'branches' of the original ILP that we denote with P_0. Since in the solution of P_0 the variable x must be integer, the solution of P_0 is the minimum of the solutions of P_1 and P_2.

By recursively calculating the solutions of P_1 and P_2, a 'search tree', as in figure 3, is generated where every node is a new ILP.

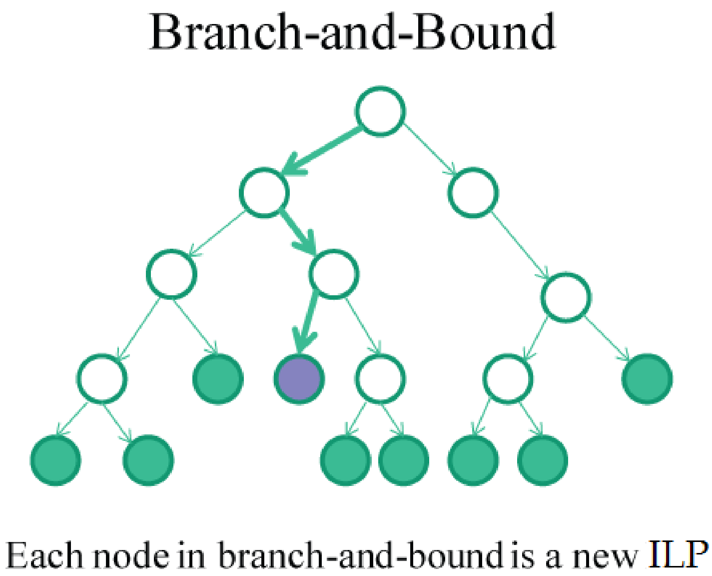


Figure 3: Branch-and-Bound search tree cf. [8]

The solution of the original ILP problem is found when all the leaf problems are **fathomed**, meaning that we do not branch further on those nodes. During the process nodes can be fathomed for several reasons.

The first is that in its LP solution all the integrality constraints are fulfilled. Then we have found a feasible solution for the original ILP, although not necessarily a minimal one. At every time in the process the best integer solution is denoted as the **incumbent**. So if a new-found feasible solution has a lower objective function value than the incumbent, it is defined to be the new incumbent.
Another case is if the LP relaxation of a node is higher than the current incumbent, then the even more restricted ILP problem cannot have a better solution and the node can be fathomed.
Finally it can happen that by imposing the additional constraint the LP relaxation of a node becomes unfeasible. Then there is obviously also no integer valid solution and the node can be omitted.

2.6.2. Optimality of the result

To demonstrate optimality `intlinprogGurobi` continuously determines the lower and upper bound to the solution. The difference is called duality gap. When the duality gap is zero, optimality is demonstrated. If we assume that our problem is a minimization problem, at any time the current incumbent is an upper bound. Since it is a feasible solution we will not have to accept a solution with a higher objective value.
In addition the minimum of the objective functions of all the current leaf nodes is a lower bound, the so-called **best bound.** Since those values are based on the LP relaxations any feasible solution would be more or equally restricted, thus having a bigger or equal objective value.

2.6.3. Presolve, Cutting Planes, Heuristics, Parallelism

An important part of modern ILP solvers are the Presolve algorithms. Before starting Branch-and-Bound, those algorithms eliminate unnecessary constraints or variables. Furthermore when dealing with integer constraints, solutions can be excluded in advance due to infeasibility. This leads to a more compact formulation of the problem that can be solved more quickly.

So called **Cutting Planes** are another strategy that has been increasingly used in the last few years, accounting for most of the progress made in ILP. They can be a tool to exclude infeasible fractional solutions during the optimization process without creating more sub-problems as in the Branch-and-Bound algorithm.

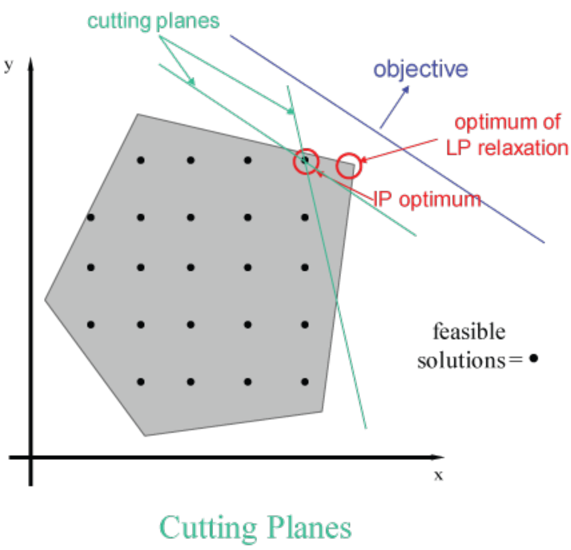


Figure 4: Cutting planes cf. [8]

Furthermore Gurobi Intlinprog uses a range of general heuristics trying to find good feasible solutions. By increasing the current incumbent those solutions allow to cut off branches with a higher LP-relaxation. Especially when working on big problems we have found heuristics to be very beneficial to the optimization process.

What is more the algorithm saves a lot of time by running parallel. The root relaxation offers only very restricted possibilities for parallelism. But when the algorithm spends a lot of time branching, as it is often the case in big problems, `intlinprog-Gurobi` uses cores very effectively. As seen in figure 5 at every step in the process the leaves can be handled parallelly.

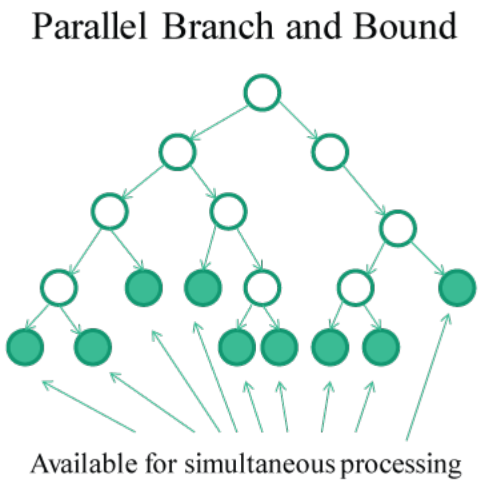


Figure 5: Parallel processing of the Branch-and-Bound search tree cf. [8]

2.7. Using the optimization solvers

All of the solvers mentioned can be used within the Matlab environment. The integrated integer linear optimization solver in Matlab,`intlinprog` is called by

```
x = intlinprog(c,intcon,A,b,Aeq,beq,lb,ub).
```

Hereby the input arguments correspond with the notation explained in the general formulation of ILPs

$$\text{minimize } c^T x \quad \text{while} \quad \left\{ \begin{array}{l} A_{eq}x = b_{eq} \\ Ax \leq b \\ lb \leq x \leq ub \\ \text{some or all } x_j \in \mathbb{Z}. \end{array} \right\}.$$

Hereby in `intcon` the integer-constraint entries of `x` are listed. The corresponding expression to call the solvers by Cplex and Gurobi are:

```
x = cplexbilp(c,A,b,Aeq,beq)
```

and

```
x = intlinprogGurobi(c,intcon,A,b,Aeq,beq,lb,ub).
```

Unlike the solvers from Matlab and Gurobi, `cplexbilp` does not use integer constraints nor upper or lower bounds since it assumes all variables to be binary. In chapter 5 we will show a comparison on solver runtimes for Matlab, Cplex and Gurobi on different examples.

3. Cost Model

This chapter will provide the details on the cost data used and will explain how it was possible to arrange it linearly. Since we do not include all of the costs at once but stepwise as shown in 2.5, in this chapter we cannot describe how to construct a general cost vector. It will be included in the stepwise explanation of the implementation in chapter 4 where in every step we will give the cost vector based on the the complexity of the current model.
Therefore, the notations for cost data which will be introduced in the following, will be used again. Generally we should keep in mind that the cost vector must always be of the same length as the solution vector. Since in our case the solution vector will be completely binary, for every entry of the solution vector the corresponding entry of the cost vector must contain the costs that arise if the entry is one.

3.1. Distances of the turbines based on topography

Earlier works have investigated cable choices in offshore wind farms [17]. Although overhead transmission would have been a possibility to avoid using subsea cables, this alternative was discarded since the necessary overhead line towers would be even more expensive. However, the high cost of acquisition and installation of subsea cables are a major cost factor.
It is therefore important to depict the distances covered by the cables built-in in a layout as realistic as possible.
The cables are usually buried in the seabed about one meter below the ground. Assuming that the needed cable length would simply be the beeline between two turbine positions would not represent the reality, as we usually encounter irregularities of the sea ground.
By using a grid as in [12], we can simulate the actual path of the cable in the seabed. We derived the depth of the seabed for each grid point in the given properties from Google Maps. This way we are able to consider subsea hills or canyons. We can also mark areas where cables should not be placed, such as sites with war wrecks.
Our examples are based on two wind farms: Horns Rev 1 and Sandbank. Both are located in the North Sea.

Horns Rev 1, which will just be called Horns Rev in the following, is situated at the Danish west coast, 20 sea miles west into the ocean. It is built on a 20 km^2 area and was first taken into operation in 2002. With an installed power of 160 MW in 80 turbines it was one of the first big offshore wind farms (cf. [11]) For the topography of Horns Rev a grid with a grid width of 50 m is used. In figure 6 we see the topography of the property where Horns Rev was built.

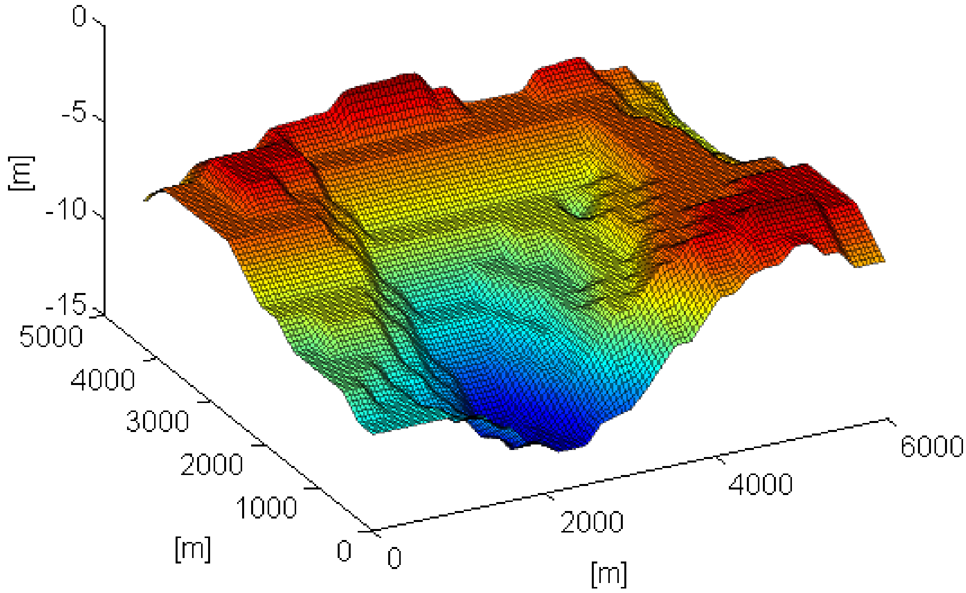


Figure 6: Topography of the Horns Rev wind farm

The other wind farm that will be investigated is Sandbank, a wind farm run by Vattenfall and Stadtwerke München.
It is situated in the German North Sea, 90 km west of Sylt. Currently still under construction, it will presumably go into operation in 2017. In the first step, 73 turbines were placed on an area covering 59 km^2 [20]. Several expansions are planned for the future. As can be noted the used area is much bigger than for Horns Rev, which is due to the steady growth of turbine dimensions in the past few years. To ensure productive operation, the distances between the turbines needed to be adapted to the rotor sizes. To keep the number of grid points in our depth grid in an acceptable dimension we use a grid width of 100 m in the Sandbank examples. Figure 7 shows the seabed of the Sandbank property.

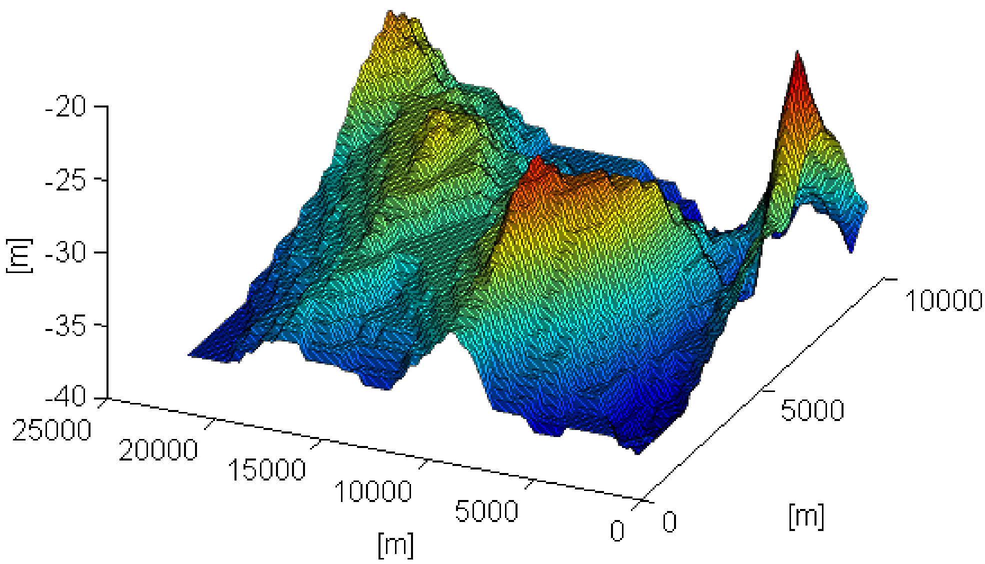


Figure 7: Topography of the Sandbank wind farm

In order to calculate the distances every turbine position is identified with the closest grid point. A maximal error of $\frac{1}{\sqrt{2}} \times$ *grid width* can be made here. Using the Dijkstra-Algorithm, the shortest distance and the paths between every pair of identified grid points are calculated. Hereby we are only allowing connections from grid points to their eight surrounding grid points. Quite significant errors might happen using this approach, which is why for realistic application a more exact grid is recommended. If the number of grid points is higher, the calculation of distances and paths can take very long, but it is only necessary once for every property.
To minimize the error at both ends of the path, we allow a direct connection of the turbine to the second grid point on the path if it is shorter than the calculated path.

3.2. Cable costs and capacities

As we have already calculated the distances between the turbines in meters, the next step is to provide the per meter cost of acquisition and installation of the cables.

For our examples we use a number of different subsea cables. The data on the cables was provided by Vattenfall or is derived from [14] using the submarine power cables of Type (F)2XS2Y>c<RAA in different sizes. All of them have a nominal voltage of 33 kV allowing a maximum voltage of 36 kV. For each cable we calculate the number of turbines, the current of which can be carried by the cable. We will be referring to this

number as the **turbine capacity** of a cable, contrary to the cable capacity which will be used in the calculation of the power losses. Table 2 gives an overview of the cable data that will be used in the examples in this work. For all cable types installation costs of 550 € /m can be assumed [16].

Cable type	Price	Current rating	Cable capacity C	AC resistance R_{AC}	$\tan(\delta)$
1	131 €/m	384 A	0,22 μF/km	0,16 Ω/km	0.0004
2	147 €/m	415 A	0,21 μF/km	0,13 Ω/km	0.0004
3	166 €/m	430 A	0,23 μF/km	0,13 Ω/km	0.0004
4	173 €/m	490 A	0,26 μF/km	0,10 Ω/km	0.0004
5	198 €/m	543 A	0,27 μF/km	0,08 Ω/km	0.0004
6	234 €/m	600 A	0,30 μF/km	0,06 Ω/km	0.0004
7	270 €/m	659 A	0,33 μF/km	0,05 Ω/km	0.0004
8	400 €/m	721 A	0,37 μF/km	0,04 Ω/km	0.0004
9	347 €/m	740 A	0,34 μF/km	0,04 Ω/km	0.0004

Table 2: Cable data cf. [14] and [16]

To be able to refer to the certain cables in the course of this work, we name them cable type one to cable type nine. The **current rating** gives the maximal amount of current a cable can support and will be needed in the following. Cable capacity, AC resistance and $\tan(\delta)$ will be used in the calculation of the power losses.

In order to calculate how much current is produced by a single turbine, it is necessary to look at the system in which they are used. Both of the wind farms we are looking at run on a nominal voltage of 33 kV.

In Horns Rev V80 turbines from Vestas (Vestas Wind Systems A/S, Randers, Denmark) are used. They have a diameter of 80 m and a 70 m hub height producing a power output of 2 MW and allow variable wind speeds up to 25 m/s. [9].
The infield power collection runs on three-phase electric power, thus the maximal current produced by a turbine, I_{turbine}, is calculated using the power of a turbine P_{turbine}, the voltage of the system U and the power factor $\cos(\phi)$. It gives the ratio between the real power to the apparent power in the system. In wind farms it can be assumed to be 0.925 [16], thus we calculate:

$$I_{\text{turbine}} = \frac{P_{\text{turbine}}}{\sqrt{3}U\cos(\phi)} = \frac{2 \cdot 10^6}{33 \cdot 10^3 \cdot \sqrt{3} \cdot 0.925} \approx 38[A].$$

Now we calculate the turbine capacity of a cable by dividing the current rating by I_{turbine} and then round down to the next integer, the results are shown in 3.

Cable type	Current rating	Turbine capacity
1	384 A	10
2	415 A	10
3	430 A	11
4	490 A	12
5	543 A	14
6	600 A	15
7	659 A	17
8	721 A	19
9	740 A	19

Table 3: Horns Rev: Turbine capacities of our cable selection

We denote with k the number of turbines that can be connected with our biggest cable. That means there can be at most k turbines at one cable route leading towards the substation.
The solution vector is based on the variables ${x_{ij}}^h$ which for every used cable also define the current of how many cables it needs to transport. For the corresponding cost vector entries we will thus need the costs of the respectively necessary cable. However, as can be seen in table 3, not for every number of turbines there is a cable. In order to still receive a linear description the cost vector entry must in this case contain the cost of the cable that has to be used instead.
The cable that is actually used to connect h turbines ($1 \leq h < k$) we will call the cable of **category** h. For Horns Rev the following table shows our cables choice.

We choose the cable of category	1	2	3	4	5	6	7	8	9	
to be of cable type	1	1	1	1	1	1	1	1	1	
We choose the cable of category	10	11	12	13	14	15	16	17	18	19
to be of cable type	1	3	4	5	5	6	7	7	9	9

Table 4: Horns Rev: Cable categories

As the number of cables that can be connected with particular cables varies in other wind farms, there will be a different cable choice for Sandbank.

In the actual layout of Horns Rev there are two cable types used, one connecting up to eight cables, the other one carrying the current from up to 16 cables. Since we do not have the actual cable data at hand, we chose cable 1 and cable 7 to realistically reconstruct the layout. The data on turbine positions is freely accessible [7]. The result can be seen in figure 8, the turbines are represented by the black circles and the substation is the blue square in the upper right corner. The coloring in the background displays the topography of the seabed. On the top, the costs for the layout are given in euro. In the course of this chapter we will give more details about the calculation of those costs.

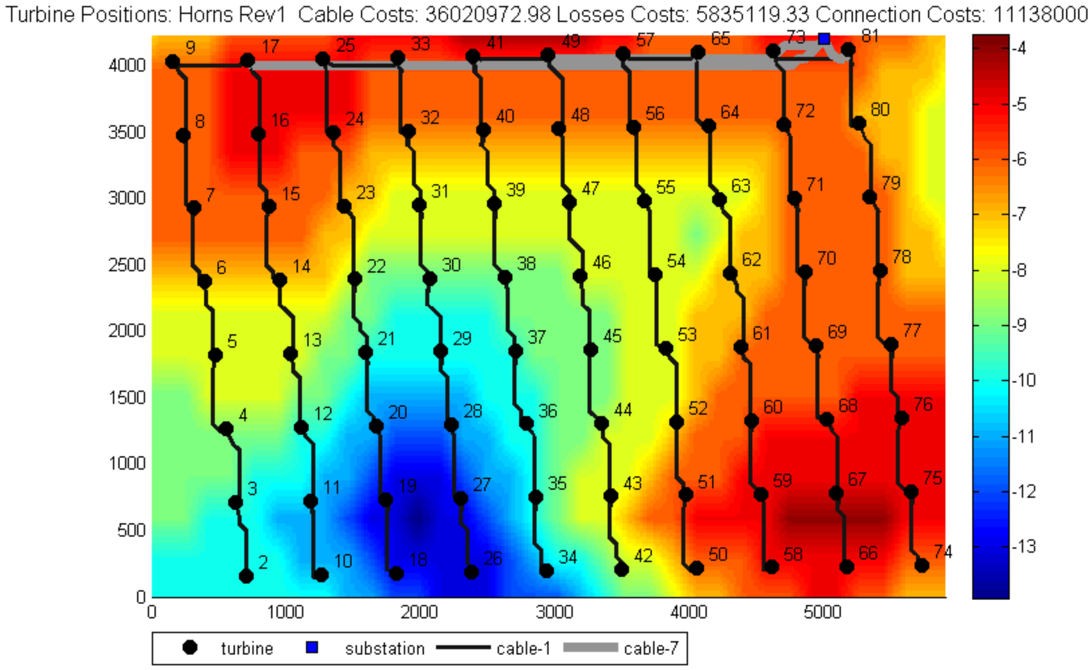


Figure 8: Actual layout of Horns Rev wind farm cf. [7], costs in €

The turbines used in Sandbank are of the type Siemens SWT-4.0-130. They have a rotor diameter of 130 m and a maximal hub height of 70 m. They produce a nominal power output of 4 MW and allow variable wind speeds from 3 up to 25 m/s. [19]
Again we calculate the maximal current produced by one turbine.

$$I_{\text{turbine}} = \frac{P_{\text{turbine}}}{\sqrt{3}U cos(\phi)} = \frac{4 \cdot 10^6}{33 \cdot 10^3 \cdot \sqrt{3} \cdot 0.925} \approx 76[A].$$

The cable types used in Sandbank itself are the cables types number 2 and 9. Considering the current rating we calculated that cables 2 and 9 can support up to 5 and 9 turbines respectively. Additionally to the eight lines, each connecting nine turbines to the substation there are four extra cables that are colored purple in the figure 9. They can transport energy of the connected line in case of cable damage. However, this is only possible if turbines are not working at their full power since the cables do not have superfluous turbine capacity.
Another advantage of the extra cables is that we do not need the extra diesel generator for turbines at the end of a line since they can be supplied with energy from the neighbouring line.

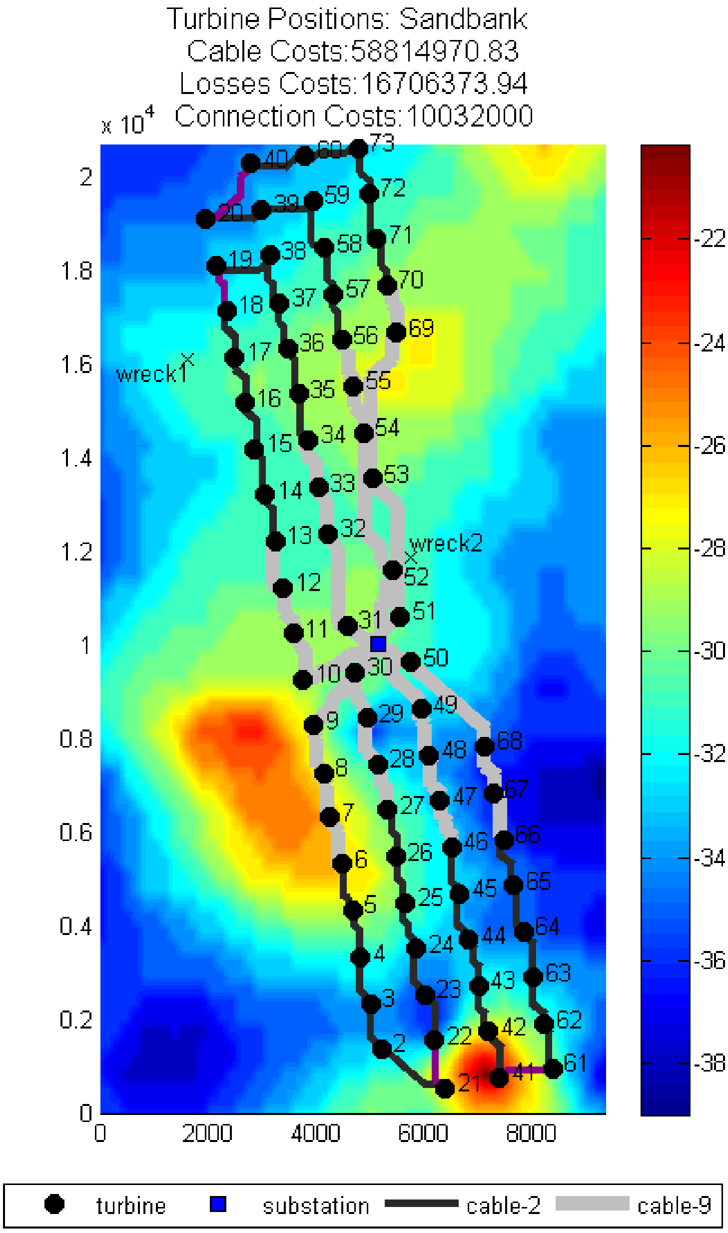


Figure 9: Actual layout of Sandbank, data cf. [16]

For the rest of our cables we calculated the cable capacities presented in table 5 for Sandbank.

Cable type	Current rating	Turbine capacity
1	384 A	5
2	415 A	5
3	430 A	5
4	490 A	6
5	543 A	7
6	600 A	7
7	659 A	8
8	721 A	9
9	740 A	9

Table 5: Sandbank: Turbine capacities of our cable selection

Accordingly the cables that are used in Sandbank are selected as follows:

We choose the cable of category	1	2	3	4	5	6	7	8	9
to be of cable type	2	2	2	2	2	4	5	7	9

Table 6: Sandbank: Cable categories

3.3. Ohmic and dielectric losses

There are different types of power losses occurring in the cables. As in [12] this work concentrates on those that amount to the major part of the losses, the ohmic and dielectric losses. As different layouts produce different amounts of losses, depending for example on the degree of branching, we want to include them in our cost model. To be able to compare layout prices, the monetary loss due to power losses in a layout during the expected operating lifetime of the wind farm is added to the layout costs.

The main cause of power loss is the ohmic loss. It depends on the electric current going through a cable and the resistance of the cable. Measured in W/m for a system with alternating current it is calculated by

$$P_{\Omega} = R_{AC} I^2$$

with R_{AC} being the AC resistance in Ω /m and I the electric current in A.

Note that in an optimal calculation we would need the actual current produced by a turbine at any given point in its operational life. What was used instead in our calculation is the average current flow. We calculate it by using the so called capacity factor, giving the percentage of the power actually produced in relation to the installed power. For Horns Rev, we use a capacity factor of 41 % as was measured in Horns Rev in the course of the last year [3]. Since Sandbank is not yet operating, the capacity factor can only be predicted. Based on values of recently built Danish wind farms shown in [3] we used a capacity factor of 50%.

In our calculation we need the ohmic loss values when using the different cable categories while they carry different amounts of current. We prepare a $k \times k$ matrix L, where $L(i,j)$ contains the ohmic losses for our chosen cable of turbine capacity i carrying the current of j turbines. Here k is the category of the biggest cable. As this simplification has almost no effect on the total amount of the losses [12], we are not considering that the arriving current is already reduced by previous losses.
The matrix must be constructed taking into account the cable choice in the wind farm we are treating. As an example we show in table 7 the matrix L for Sandbank. Entries in the upper triangular do not exist since the respective cables cannot connect the given numbers of turbines. As can be seen, the first five rows are equal, since cable 3 is the cable that we use to connect up to five turbines.

	Connected to n turbines								
Cable	n=1	n=2	n=3	n=4	n=5	n=6	n=7	n=8	n=9
3	9.04								
3	9.04	36.14							
3	9.04	36.14	81.32						
3	9.04	36.14	81.32	144.56					
3	9.04	36.14	81.32	144.56	225.88				
4	6.95	27.80	62.55	111.20	173.75	250.20			
5	5.56	22.24	50.04	88.96	139.00	200.16	272.44		
7	3.48	13.90	31.28	55.60	86.88	125.10	170.28	222.40	
9	2.78	11.12	25.02	44.48	69.50	100.08	136.22	177.92	225.18

Table 7: Ohmic losses matrix L for Sandbank, in €/m

The second type of power loss does not depend on the amount of current flowing but on the voltage U, the cable capacity C, the insulation loss factor $\tan(\delta)$ and the main frequency f used in the system. It is calculated by

$$W_d = 2\pi f C U^2 \tan(\delta).$$

The following table 8 shows the dielectric losses vector used for the cost calculation in the Sandbank examples.

Cable type	
3	0.5697
3	0.5697
3	0.5697
3	0.5697
3	0.5697
4	0.6732
5	0.6991
7	0.8545
9	0.8804

Table 8: Dielectric loss vector $\tilde{l}$ for Sandbank, in €/m

3.4. Connection cost

We found that many existing offshore wind farms do not allow cable branching in the infield power connection. Instead, all the turbines are connected in lines to the substation. As we see in figure 10, however, there are some examples of cable branching in offshore wind parks such as in the Irish wind farm Walney.

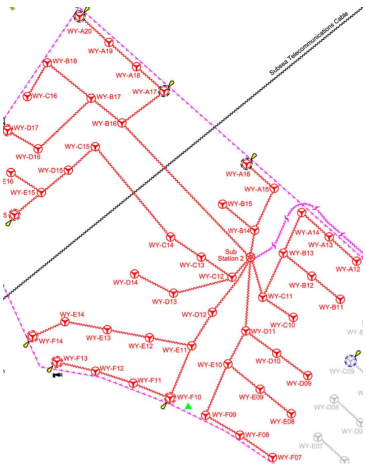


Figure 10: Grid of the offshore wind farm Walney 2, cf. [21]

Being asked for the reasons not to choose branching layouts Vattenfall argued with significantly higher costs. Those would arise when connecting more than the standard two cables (one incoming, one outgoing) to a turbine. However, the calculations in [12] suggest that by allowing branching the costs can be reduced immensely.

In this work branching is considered, but we include the cost related to the connection of the incoming and outgoing cables to the turbine. This includes the additional costs that may occur when connecting more than two cables to a turbine. According to [16], what needs to be considered per each incoming cable is: a switch gear, a cable protection system (CPS), a Hang-Off and a termination kit of T-connectors. All together those costs add up to approximately 66.266 €. This data makes it look like only linear costs are involved (for h connected cables $h\cdot$ 66.266 €). However, we want to leave the possibility of nonlinear costs for connecting non-standard numbers of cables to a turbine. Thus the input data can be chosen freely and not necessary linearly.

This also takes into consideration the additional cost of turbines that are located at the end of a cable route. All turbines must be connected to a source of electricity to be able to maintain emergency functions in the case of a turbine failure. Security lightening needs to be maintained and the rotor should be able to turn out of the wind to avoid further damage. For most of the turbines this energy can be provided by other turbines further down the cable route. Turbines on the end of a cable route, however, can not rely on such a source, a problem that can be solved by using an external diesel generator. The price of such a generator amounts to approximately 5000 € [16].

When k is the turbine capacity of the biggest available cable, the highest number of cables that could possibly be connected to a turbine is k. There is always only one outgoing cable, so if we have the incoming current from $k-1$ turbines plus the energy of the turbine itself, this is the maximum capacity that we have an outgoing cable for. At most the incoming current could be divided onto $k-1$ cables, so then, including the outgoing cable, the turbine would be connected to k cables.
Thus a connection cost input vector of length k is used where entry h contains the cost to connect h cables to a turbine. The costs for a generator can be added to the first entry of the connection cost input vector since generators have to be installed at those turbines connected to only one cable (the outgoing one).

The situation is different for the substation, from there on it is always the full amount of energy that is transported to the shore by the shore connection cable. Thus there could be an arbitrary number of incoming cables to the substation, at maximum as many as the number of cables connecting all turbines directly to the substation. It is also possible that connection costs vary when the cable goes to the substation. Therefore in our model a separate input vector for the substation connection costs is used. If there are n stations to connect, it has $n-1$ entries and contains at position h the cost to connect h cables to the substation. In our examples, however, the connection costs used are the same for turbines and the substation, as we were not given differing data. Whenever we connect a cable to a station we assume costs of 66.266 € per cable.

3.5. Scaled prices for the cables

As scaled prices are often used in commerce, we thought that adding them to our cost model would help to depict the reality better. Theoretically an arbitrary number of scales can be used in the optimization. However, we found that adding this step would make run times for the solver rather unacceptable. Speaking with Vattenfall they told us that the standard prices they use are only valid for a cable length of about 10 km upwards [16]. Therefore it might be more realistic to only use two steps, and establish that the usual prices can only be granted when cable length exceeds the fixed purchasing volume of 10 km. If less is to be purchased, we define the price to be 125 % of the standard price. However, we leave the decision to the user and allow for each cable type and each scale a limit to be set. Accordingly the cable cost vector needs to be extended to another dimension for the scales so that prices for each scale can be set.

4. Stepwise Linear Inplementation of the Costs and Constraints

In this section it is shown how the components of the model overview in 2.5 are implemented linearly within six steps.
In step one we consider the capacity restrictions and the cable costs based on the realistic turbine distances. As constraints, those necessary for a consistent layout are included; every turbine must be connected to the substation and all cable needs to withstand the energy of the number of turbines connected to them.
For step two some changes to the program structure are made so that during the program run we add constraints to avoid cable crossings if needed.
In the next step, step three, ohmic and dielectric losses are included to the cost function, basically as it was done in [12]. After that our model is on the same level as the model used in [12] and we will be able to compare the result of both approaches, which will be done in chapter 5.
Additionally in step four, the connection costs are considered in our cost model. We also allow optional restriction on the number of incoming cables to a turbine and to the substation.
In step five the feature of cable choice is implemented.
Finally, in step six, we will add the concept of scaled prices, so cable types are cheaper when we buy a certain amount of them.

The constraints we use will be explained and the results of every step can be seen in examples of the wind farms Horns Rev or Sandbank, using the cost data as explained in 3.

4.1. Step 1: Consistent layout

In the first step all turbines shall be connected to the substation. To avoid damage and allow current flow it is also necessary that every cable withstands the energy of the turbines connected to it. As discussed in the connection cost section we include branching in our layouts by default.
Let $n-1$ be the number of turbines in our wind farm, so adding one substation we have n stations to connect in total. Station 1 is the substation and stations 2 to n are the turbines. With k we denote the category of the biggest cable, meaning it can connect at least up to k turbines.

We use the basic solution variables $${x_{ij}}^h = \begin{cases} 1 & \text{a cable runs from } i \text{ to } j,\ i \text{ is the } h^{th} \text{ turbine on the ro} \\ 0 & \text{otherwise} \end{cases}$$

$$i = 1...n, j = 1...n, h = 1...k.$$

as explained in chapter 2.
Thus the solution vector x will be the vectorized matrix $X = ({x_{ij}}^h)_{i=1...n,j=1...n,h=1...k}$.
In the following we will keep using the notation x for the collection of solution variables

${x_{ij}}^h, i = 1...n, j = 1...n, h = 1...k$ and apply it for new solution variables respectively.

Input data

- Cable cost vector $p \in \mathbb{R}^k$: the vector where $p(h)$ contains the cost of a cable of category h as described in chapter 3.
- Distances matrix $D \in \mathbb{R}^{n \times n}$: the matrix where $D(i, j)$ contains the realistic distances between turbines i and j as calculated by the Dijkstra algorithm.

Cost function

If i is the h^{th} turbine on the route, the cable from i to another turbine j needs to transport the current of h turbines. At this stage we use the cable of category h (and not higher) in this case. This means that the necessary turbine capacity is equal to the cable category we chose.
Thus the cost vector c is created in the following way:

$$c = ({c_{ij}}^h)_{i=1...n, j=1...n, h=1...k} \quad \text{where} \quad {c_{ij}}^h = D(i, j)p(h).$$

The entry ${c_{ij}}^h$ contains the cost to connect turbine i with turbine j by a cable of category h. In the cost function

$$c^T x = \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{h=1}^{k} {x_{ij}}^h {c_{ij}}^h \in \mathbb{R}$$

those entries are multiplied with the binary solution vector so that in total we get the cost of the cables actually used.
The solver aims to minimize this function while fulfilling the constrains described in the next section.

4.1.1. Constraints for x

Although in [4] cable branching is not allowed, we can adapt some of the constraints from [4] and use them for our model. Let n be the number of stations and k the category of the biggest cable. In the first step the following constraints are used:

(1) According to the definition all the solution variables are binary.

$${x_{ij}}^h \in \{0, 1\}$$

for all $i = 1..n, j = 1...n, h = 1...k$.

(2) Every turbine has exactly one outgoing cable (in the direction towards the substation).

$$\sum_{j=1}^{n} \sum_{h=1}^{k} {x_{ij}}^h = 1$$

for all $i = 2...n$.

(3) For every turbine, in the expression $\sum_{r=1}^{n} \sum_{h=1}^{k} {x_{jr}}^h$ there is only one ${x_{jr}}^h \neq 0$ so $\sum_{r=1}^{n} \sum_{h=1}^{k} h{x_{jr}}^h$ yields the category of the outgoing cable of turbine j. It is equal to the sum of the capacities of the incoming cables $\sum_{i=1}^{n} \sum_{h=1}^{k} h{x_{ij}}^h$ plus one, since also turbine j is producing energy.

$$1 + \sum_{i=1}^{n} \sum_{h=1}^{k} h{x_{ij}}^h - \sum_{r=1}^{n} \sum_{h=1}^{k} h{x_{jr}}^h = 0$$

for all $j = 2...n$.
Together with constraints (1) and (2), this induces the tree structure. By excluding the substation from the constraints, it yields a layout where all turbines are connected to the substation, making it the root node.

We also use a number of additional constraints that are implied by the constraints (1) to (3) when minimizing the cost function.

(4) Neither of the turbines nor the substation can be connected to itself.

$${x_{ii}}^h = 0$$

for all $i = 1..n, h = 1...k$.

(5) The cable of highest category can only be used to the substation.

$${x_{ij}}^k = 0$$

for all $j = 2...n$.

(6) Between two turbines there is at most one cable.

$$\sum_{h=1}^{k} {x_{ij}}^h + {x_{ji}}^h \leq 1$$

and

$$\sum_{h=1}^{k} {x_{ij}}^h \leq 1$$

for all $i = 1..n, j = 1...n$.

4.1.2. Example: how to use `intlinprog`

As explained earlier the input data needs to be written in the following form

```
x = intlinprogGurobi(C,intcon,A,b,Aeq,beq,lb,ub).
```

Since all of the used solution variables shall be binary `intcon`, `lb` and `ub` are defined as follows

```
lb=zeros(dimx,1);
ub=ones(dimx,1);
intcon=(1:dimx);
```

where `dimx` is the dimension of the solution vector `x`.

As an example we will show how to write constraint (2) into `Aeq` and `beq`.
First we initialize the constraint matrices with zeros. Every constraint is written into one row i of `A` or `Aeq` together with the i^{th} entry of `b` or `beq` respectively. Thus the dimension of `Aeq` is the number of equality constraints times the dimension of `x` relating to $A_{eq}x = b_{eq}$.

```
Aeq=zeros(anzEqconstraints,dimx);
beq=zeros(anzEqconstraints,1);
counter=1;
```

So we can implement the constraint

(2)

$$\sum_{j=1}^{n}\sum_{h=1}^{k} {x_{ij}}^h = 1$$

for all $i = 2...n$ as:

```
for j=2:n
   for i=1:n
       for h=1:k
           Aeq(counter,(n·n·(h-1)+n·(j-1)+i))=1;
       end
   end
   beq(counter)=1;
   counter=counter+1;
end
```

In the same way constraint (3) and all of the future constraints are written into `Aeq` and `beq` or `A` and `b`. `c` will just be the cost vector as defined above.

4.1.3. First result

Using the described constraints and turbine positions from Horns Rev the optimization solver yields the cable layout displayed in figure 11.

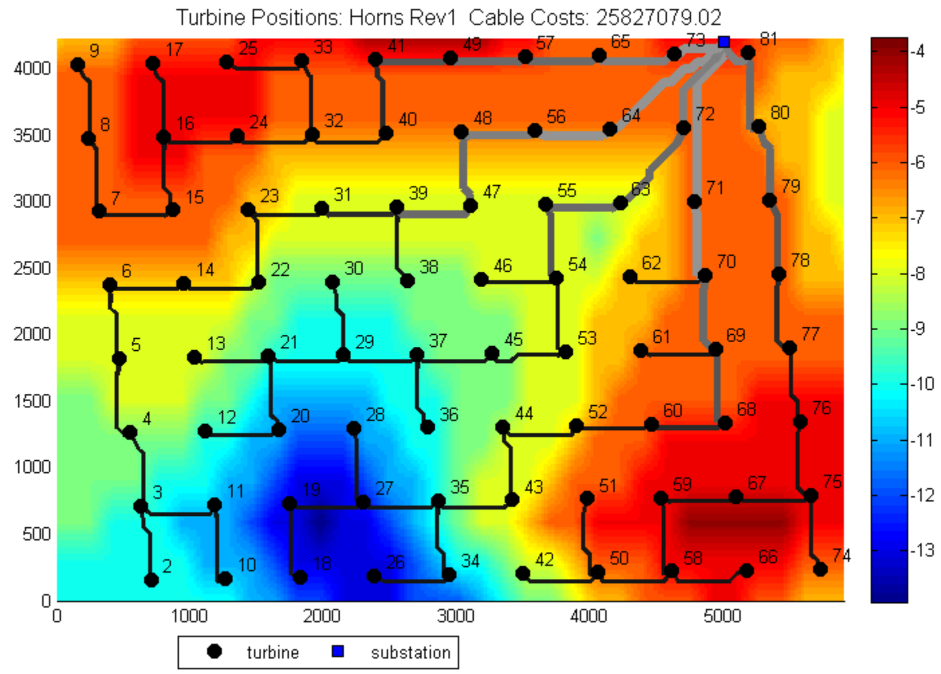


Figure 11: Optimal layout of Horns Rev wind farm considering the constraints from Step 1, cable costs in €

The different colors of the cables symbolize the different cable types used. As we can see the capacity of the cables increases towards the substation. Since cable one (black in the picture) can connect up to 10 turbines it is used the most. Overall cables of type 1,3,4,5,6 and 7 are used.

Applying the solver to the sandbank example we receive the following result:

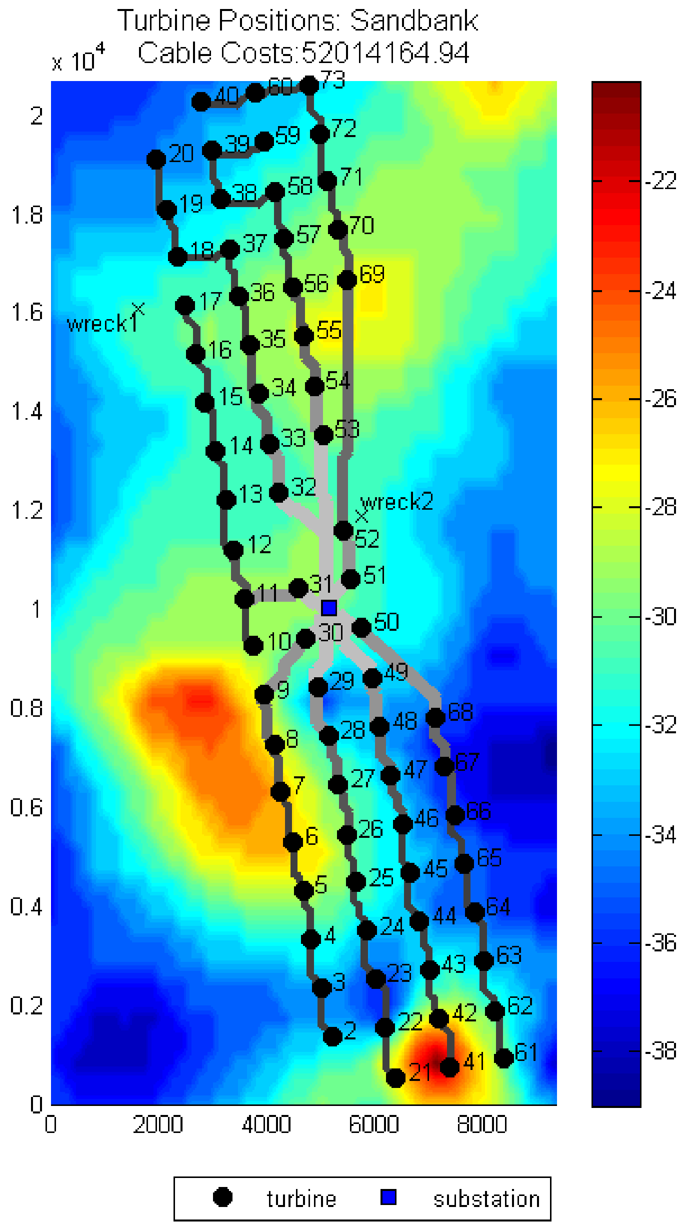


Figure 12: Optimal layout of Sandbank considering the constraints from Step 1, cable costs in €

Within this layout we find cables of types 2,4,5,7 and 9.

4.2. Step 2: Eliminate crossings

As investigated in [12], the wind farm operators avoid crossings, because one cable would have to be buried exceptionally deep below the other causing extremely high costs. Therefore our resulting layout should be free of crossings crossings.

We denote a cable between turbine i and j with $\{i, j\}$. If two cables $\{i, j\}$ and $\{u, w\}$ would cross when they are both in the layout, the following constraint can be added to only allow one of them in the layout

$$\sum_{h=1}^{k} {x_{ij}}^h + {x_{ji}}^h + {x_{uw}}^h + {x_{wu}}^h \leq 1.$$

We want to use the program on wind farms that in our example have around 80 turbines, thus the number of necessary constraints is too high if we want to add constraints for all possibly crossed cable pairs. In Horns Rev there are $81 \cdot 80 = 6480$ possible cable placings, to pairwise exclude those cable pairs that would cross, we would need around four million constraints.
A solution also used in [4] is to only add constraints to cable pairs that cross in a previous solution. Thus after we have started the solver the first time, we look for crossings in the solution layout. If we find none we have finished, if we do, we save crossed cables $\{\{i, j\}, \{u, w\}\}$ in a matrix χ.
Then the constraint

(7)
$$\sum_{h=1}^{k} {x_{ij}}^h + {x_{ji}}^h + {x_{uw}}^h + {x_{wu}}^h \leq 1$$

for all $\{\{i, j\}, \{u, w\}\} \in \chi$ is added. Next we run the solver again with the changed constraint matrices. Again we start looking for crossings adding the associated constraints and continue the procedure until we get a layout without crossings. Even thought this could yield to a potentially infinite process we found that in reality it was functioning well.

As we have seen in figure 11 the layout in Step 1 did not contain crossings even though constraint (7) had not yet been used. Therefore we especially constructed an example containing crossings, figure 13, to be able to show the effects of the changed program in figure 14.

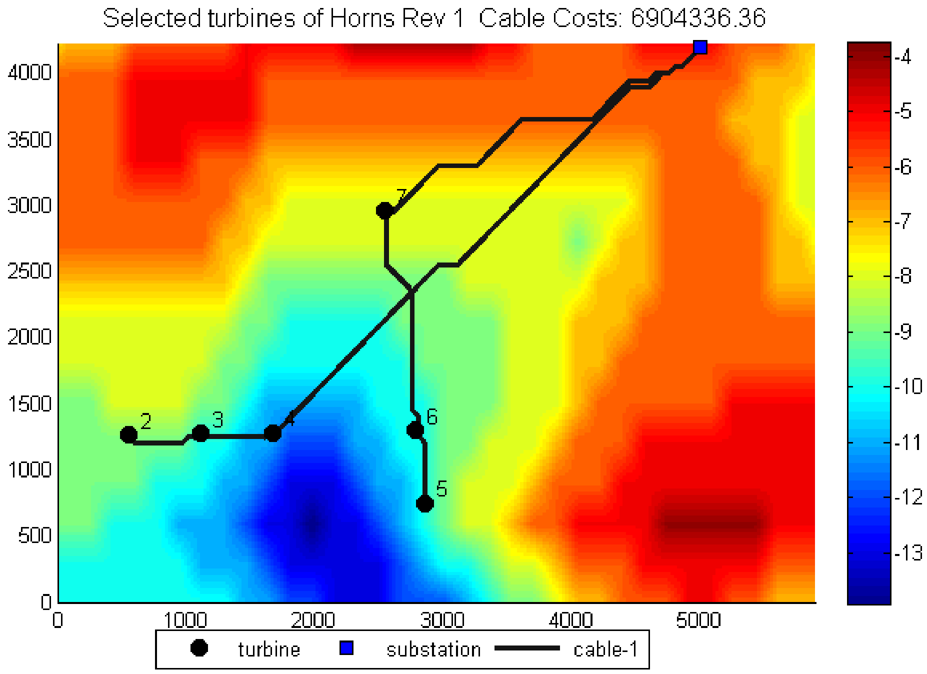


Figure 13: Optimal layout with crossings, cable costs in €

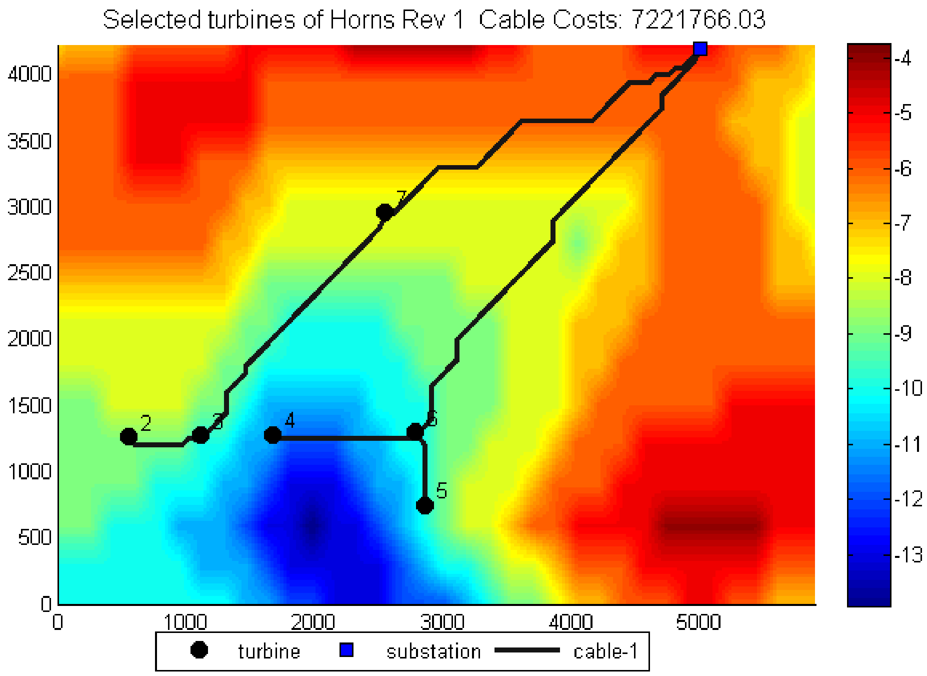


Figure 14: Optimal layout without crossings, cable costs in €

As in [12] the search of crossings is realized by creating straight lines between connected turbines. If the lines run parallel they do not cross, but if they have different slopes, we calculate the point of intersection and check whether it lies between the turbine positions. However, since the cables do not run straight there may be some exceptions when using the recommended cable course, two cables would actually cross. We recommend improving the cable crossing detection in further work. Anyway, since the straight connections of the turbines do not cross anymore after using step 2, we can alter the cable course concerned manually, in the worst case to the straight connection.

4.3. Step 3: Add losses

In step 3 the power losses calculated in 3 will be integrated in the cost vector. The solution vector does not need to be extended since the losses depend on the cable used and the current flow, information that are contained in the solution variable x.
Input data

Apart from the input data that we already used before, the cable cost vector p and the distances matrix D, we now need:

- ohmic losses $L \in \mathbb{R}^{k \times k}$: a matrix as given in figure 7 but for the respective wind farm. The entry $L(h, g)$ contains the monetary loss in €$/m$ for our cable of category h carrying the current of g turbines.
- dielectric losses $\tilde{l} \in \mathbb{R}^k$: a vector as given in figure 8, adapted to the respective wind farm, where $\tilde{l}(h)$ contains the monetary loss in euro $/m$ for our cable of category h.

Cost function

If between i and j the current of h turbines needs to be transported, we still definitely use our cable of category h (and not higher). Thus in this step we will only need the diagonal entries of the ohmic losses matrix L.
The resulting cost vector

$$c = ({c_{ij}}^h)_{i=1\ldots n, j=1\ldots n, h=1\ldots k}$$

is given by

$${c_{ij}}^h = D(ij)(p(h) + L(h,h) + \tilde{l}(h)).$$

The cost function $c^T x$ still has the form

$$\sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{h=1}^{k} {x_{ij}}^h {c_{ij}}^h$$

just with changed values for c_{ij}.

On the next page we can see the changes in the optimal Horns Rev layout when power losses are considered. Additionally to the previously used cables 1,3,4,5,6,7, in the new version shown on figure 16 cable 9 (the one in light grey) is used. We can also see that branching increased in some parts of the layout. This makes sense since larger cables and branched layouts result in lower power losses.

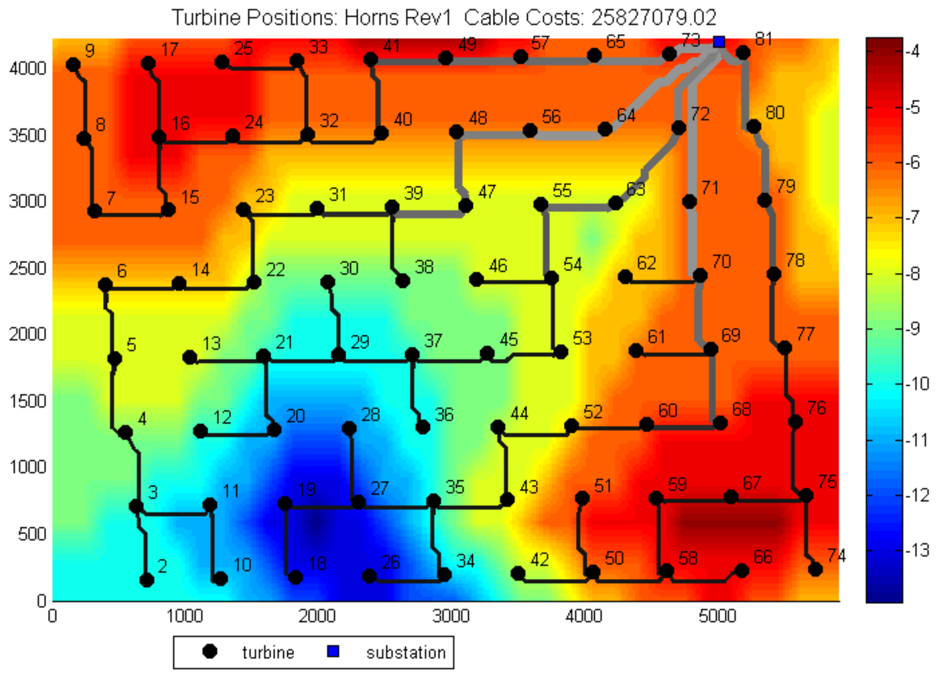


Figure 15: Optimized layout of Horns Rev not considering power losses, costs in €

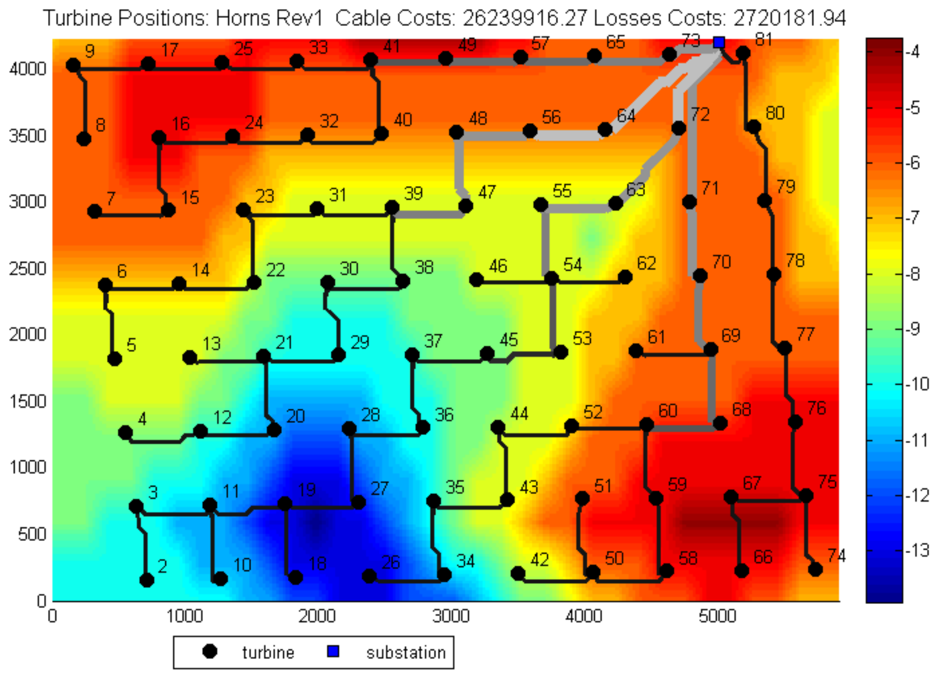


Figure 16: Optimized layout of Horns Rev considering power losses, costs in €

4.4. Step 4: Connection costs

In step four the costs related to the connection of the incoming and outgoing cables of a turbine are added to the cost vector. To be able to include them linearly in our layout we add solution variables ${v_i}^l$:

$${v_i}^l = \begin{cases} 1 & \text{if turbine } i \text{ is connected with } l \text{ cables} \\ 0 & \text{otherwise} \end{cases}$$

$$\text{for } i = 1, l = 1...n-1 \text{ and } i = 2...n, l = 1...k$$

From Step 1 we keep the variables x, thus the new solution vector z contains both of the matrices in vectorized form

$$z = [x; v].$$

Furthermore, we add an optional restriction on incoming cables to a turbine and to the substation to allow searches for optimal layouts that are less or not at all branched. Although by adding the connection costs we have created a tool to completely compare the economic effects of layouts with branching to layouts without branching, this might be the will of wind farm operators.

Input data

We extend the previously used input data (cable cost vector p, distances matrix D, ohmic losses matrix L and dielectric losses vector $\tilde{l}$), adding

- connection cost to the turbines: $r \in \mathbb{R}^k$ the vector where $r(l)$ gives the cost to connect l cables to a turbine, we use the data assumed in chapter 3.
- connection cost to the substation: $rs \in \mathbb{R}^{n-1}$ the vector where $rs(l)$ gives the cost to connect l cables to the substation, again we use the data assumed in chapter 3.
- $maxR$: the maximal number of cables that the user wants to be connected to one turbine, its default value is k.
- $maxRS$: the maximal number of cables that the user wants to be connected to the substation, its default value is $n - 1$.

The costs are calculated as before, simply adding two terms for the connection costs. As for the solution variables the collection of cost vector entries ${c_{ij}}^h$ is named c_1 and respectively for the other components.

$$c = [c_1; c']$$

with

$${c_{i,j}}^h = D_{ij}(p(h) + L(h, h) + \tilde{l}(h))$$

$$c_1'^l = rs(l), c_i'^l = r(l).$$

This yields the changed cost constraint:

$$\min_z c^T z = \min_z \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{h=1}^{k} x_{ij}{}^h c_{ij}{}^h + \sum_{l=1}^{n-1} c_1'^l v_1{}^l + \sum_{i=2}^{n} \sum_{l=1}^{k} c_i'^l v_i{}^l.$$

4.4.1. Constraints for v

Additionally to the constraints (1) to (7), we use constraints on the new solution variables $v_i{}^l$:

(1') As an extension of constraint (1) and according to our definition, we define

$$x_{ij}{}^h, v_i{}^l \in \{0, 1\}$$

for all $i = 1..n, j = 1...n, h = 1...k, l = 1...k$ and $l = 1...n - 1$ respectively.

(8) For a turbine i, $v_i{}^l$ is one only for one l so $\sum_{l=1}^{k} l v_i{}^l$ is the number of cables connected to turbine i. To determine the values of v from the entries of x, we define this sum to be equal to the number of the incoming cables $\sum_{j=1}^{n} \sum_{h=1}^{k} x_{ji}^h$ plus the one outgoing cable.

$$\sum_{l=1}^{k} l v_i{}^l = \sum_{j=1}^{n} \sum_{h=1}^{k} x_{ji}{}^h + 1$$

for all $i = 2..n$ and

$$\sum_{j=1}^{n} \sum_{h=1}^{k} x_{j1}{}^h = \sum_{l=1}^{n-1} l v_1{}^l$$

for the connection to the substation.

(9) To assure that the optional restrictions on the number of connections are met, the variables in question are simply defined to be zero.

$$v_i{}^l = 0$$

for all $i = 1, l > maxRS$ and $i = 2...n, l > maxR$.
To improve the optimization speed we add:

(10) Following from the definition v_i^l can only be one for one l for every station i

$$\sum_{l=1}^{k} v_1{}^l = 1 \quad \text{and} \quad \sum_{l=1}^{k} v_i{}^l = 1$$

for all $i = 2...n$. It is implied by the necessary constraints when minimizing the cost function.

On the following page the changes in the layout when considering connection cost in the optimization can be observed. Using the example of Horns Rev we see that branching is being reduced compared to the solution of step 3 (figure 17).
The effect must be from the extra cost for a diesel generator of turbines at the end of the route. It should not be caused by the general connection costs since they occur at both ends of the cables and the overall number of connected cables did not change.

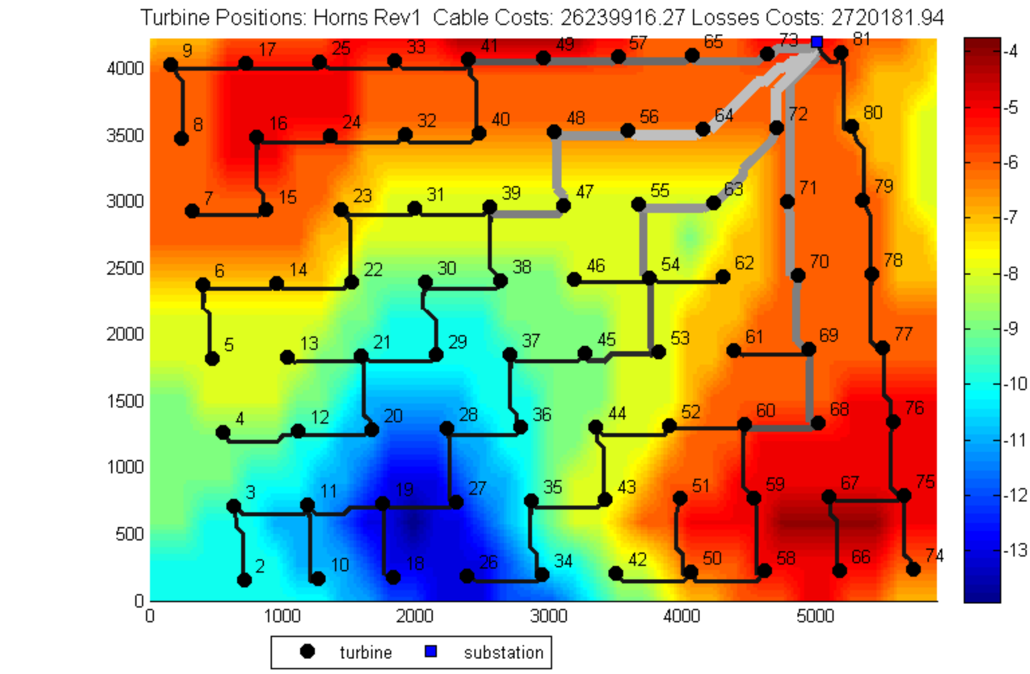


Figure 17: Optimized layout of Horns Rev not considering connection costs, costs in €

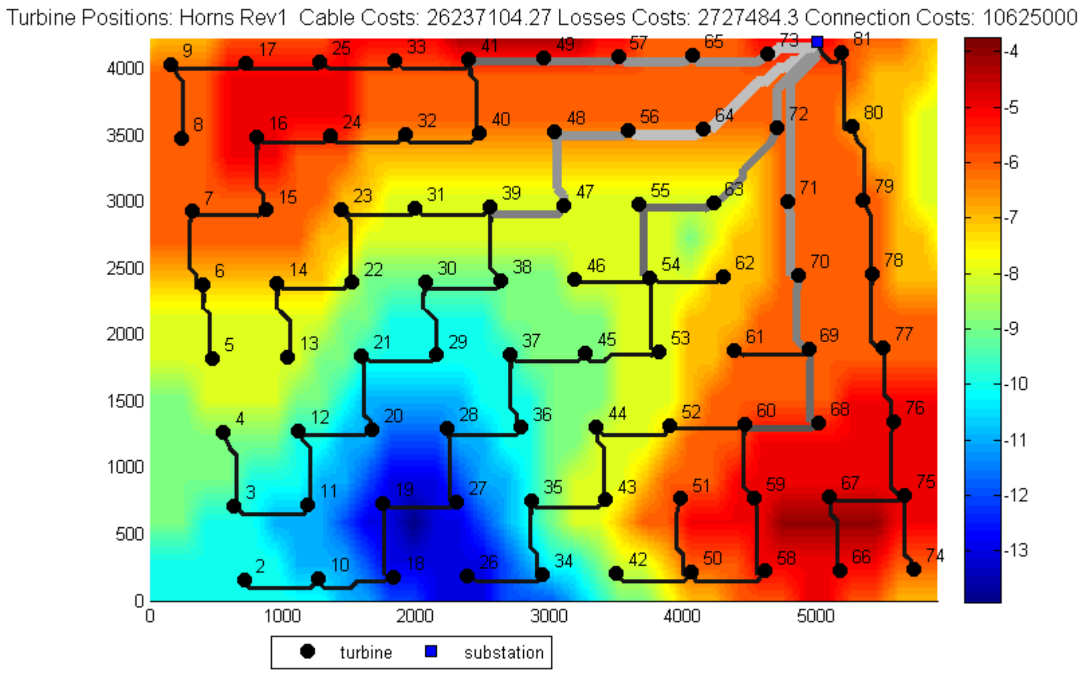


Figure 18: Optimized layout of Horns Rev considering connection costs, costs in €

4.5. Step 5: Best cable choice

In this step, an automatic choice of the best cables is implemented. The user chooses the number of cables which shall be used in the layout and the program will determine which cables to use best. The feature is closely linked to the scaled costs, most likely using scaled costs would cause the solver to find an optimal layout that has less different cable types. To implement both features we need to add solution variables to our model. The best cable choice can be realize using fewer solution variables, thus runtimes are more likely to stay in an acceptable scale.

From now on we are separating the chosen cables from the required cables. This means that where a cable of turbine capacity h is needed, we do not necessarily use our cable of category h, but possibly a bigger cable in order to reduce the total number of cables.
However the the solution variables ${x_{ij}}^h$ will still be used, giving the necessary turbine capacity, so the current flow in the respective positions. To describe which cables were chosen, we use the variables ${\tilde{x}_{ij}}^h$. To depict which cable categories are used in the layout variables u_h are introduced. Due to the separation of required cable capacities and chosen cable categories also the calculation of the losses needs to be adapted. The dielectric losses only depend on the chosen cables, thus they can be calculated by

$$\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{h=1}^{k} {\tilde{x}_{ij}}^h \cdot D(ij) \cdot \tilde{l}(h).$$

The ohmic losses however depend on the current flow, that is the necessary turbine capacity, and on the cable capacity, the characteristic value of the respectively used cable. To be able to express those costs linearly, new solution variables ${y_{ij}}^{h,g}$ are used. With them we will calculate the ohmic losses by

$$\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{h=1}^{k}\sum_{g=1}^{k} {y_{ij}}^{h,g} \cdot D(ij) \cdot L(h,g).$$

We now use the solution vector $z = [x; \tilde{x}; v; u; y]$ with

$$
{x_{ij}}^h = \begin{cases} 1 & \text{a cable of turbine capacity } h \text{ is required from } i \text{ to } j \\ 0 & \text{otherwise} \end{cases}
$$

for $i = 1...n, j = 1...n, h = 1...k,$

$$\tilde{x}_{ij}{}^h = \begin{cases} 1 & \text{the cable of category } h \text{ is used from } i \text{ to } j \\ 0 & \text{otherwise} \end{cases}$$

for $i = 1...n, j = 1...n, h = 1...k$,

$$v_i{}^l = \begin{cases} 1 & \text{if turbine } i \text{ is connected with } l \text{ cables} \\ 0 & \text{otherwise} \end{cases}$$

for $i = 1, l = 1...n-1$ and $i = 2...n, l = 1...k$,

$$u_h = \begin{cases} 1 & \text{if we use the cable of category } h \text{ in the layout} \\ 0 & \text{otherwise} \end{cases}$$

for $h = 1...k$

and

$$y_{ij}{}^{h,g} = \begin{cases} 1 & \text{while a cable of turbine capacity } h \text{ is needed we use the cable of category } g \text{ from } i \text{ to } j \\ 0 & \text{otherwise} \end{cases}$$

for $i = 1...n, j = 1...n, h = 1...k, g = 1..k$.

Note that in the new description of the case that $x_{ij}{}^h = 1$, 'a cable of turbine capacity h is required from i to j' is equal to the old one 'a cable runs from i to j, i is the h^{th} turbine on route'.

Input data

The only value we add to the previous input data is $maxCables$, the maximum number of cable categories that shall be used, it is chosen by the user.

The cable cost in our new model obviously depends on the selected cables, thus there are no more costs depending on the required cable capacities $x_{ij}{}^h$. Also there are no costs based on the solution variables u_h. In the cost vector we add an empty vector in the beginning and in the end and change the calculation of the losses and the cable cost.

$$c = [e_1; \tilde{c}; c'; e_2; \mathfrak{c}]$$

where

$$\tilde{c}_{ij}^h = D(ij) \cdot (p(h) + \tilde{l}(h)),$$

$$c_1'^{\,l} = rs(l), c_i'^{\,l} = r(l),$$
$$\mathfrak{c}_{ij}{}^{hg} = D(ij) \cdot L(h,g)$$

and

$$e_1 \in \mathbb{R}^{n \cdot n \cdot k}, e_1(s) = 0, e_2 \in \mathbb{R}^k, e_2(h) = 0.$$

The new cost function $c^T x$ has the form

$$\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{h=1}^{k} \tilde{x}_{ij}{}^h \, \tilde{c}_{ij}{}^h + \sum_{l=1}^{n-1} c_1'^{\,l} v_1{}^l + \sum_{i=2}^{n}\sum_{l=1}^{k} c_l' v_i{}^l + \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{h=1}^{k}\sum_{g=1}^{k} y_{ij}{}^{hg} \mathfrak{c}_{ij}{}^{hg}.$$

4.5.1. Constraints for x̃,u,y

(1") As before all of the used solution variables have to be binary.

$$x_{ij}{}^h, v_i{}^l, \; \tilde{x}_{ij}{}^h, u_h, y_{ij}{}^{hg} \in \{0,1\}$$

for all $i = 1..n, j = 1...n, h = 1...k, g = 1...k, l = 1...k$ and $l = 1...n-1$ respectively.

(11) The used cables have be of a category higher than the required category.

$$\sum_{h=1}^{k} x_{ij}{}^h h \leq \sum_{h=1}^{k} \tilde{x}_{ij}{}^h h$$

for all $i = 1..n, j = 1..n$.

(12) The sum of the used cable categories must be smaller than the defined maximal cable number.

$$\sum_{h=1}^{k} u_h \leq maxCables.$$

Note that this does not cause a to strong restriction on used cable types. If $maxCables = 3$ and the use of three cable types would be optimal, this constraint causes that there is only one used cable category that uses any of those cable types. For example the cable of type 3 is used in the categories one to five in the Sandbank layout. If without constraint (12) cables of the categories 2 and 5 were used in the optimal layout of Sandbank, by using (12) only category 5 in stead of 2 and 5. Since in both categories the same cable is used, this change does not affect the cable cost. Also the ohmic losses stay unchanged since their calculation in step 5 is no longer based on the chosen cable category.

(13) The values of y are defined by the values of x and $\tilde{x}$. If at cable position (i, j) x_{ij}^h and $\tilde{x}_{ij}^h$ are one for fixed h and g, we force ${y_{ij}}^{hg}$ to be one.

$$\tilde{x}_{ij}{}^{h} + x_{ij}^{h} - 1 \leq y_{ij}{}^{hg}$$

for all $i = 1..n, j = 1...n, h = 1...k, g = 1...k$.

(14) At every position ${y_{ij}}^{hg}$ can only be one for one combination hg.

$$\sum_{h=1}^{k} \sum_{g=1}^{k} y_{ij}{}^{hg} \leq 1$$

for all $i = 1..n, j = 1...n$.

For speed we add the following constraints:

(15) if the cable category h is not used in the layout, the corresponding entries in $\tilde{x}$ must be zero.

$$\tilde{x}_{ij}{}^{h} \leq u_h$$

for all $i = 1..n, j = 1..n, h = 1...k$.

(16) Since there is exactly one outgoing cable from every turbine, this means that we have to select exactly one outgoing cable.

$$\sum_{j=1}^{n} \sum_{h=1}^{k} \tilde{x}_{ij}{}^{h} = 1$$

for all $i = 2...n$.

(17) In every position there can be at maximum one type of cable.

$$\sum_{h=1}^{k} \tilde{x}_{ij}{}^{h} \leq 1$$

for all $i = 1...n, j = 1...n$.

(18) Between every two turbines there is at maximum one cable.

$$\sum_{h=1}^{k} \tilde{x}_{ij}{}^{h} + \tilde{x}_{ji}{}^{h} \leq 1$$

for all $i = 1..n, j = 1...n$.

4.6. Step 6: Scaled prices

To be able to calculate the scaled cost linearly $\tilde{x} = (\ \tilde{x}_{i,j}{}^{h})_{i,j,h}$ has to be expanded to another dimension for the scales. For every cable type h the scaled price is calculated depending on the purchasing volume of cable h. The new variable

$$(\ \tilde{x}_{i,j}{}^{ht})_{i,j,h,t}$$

will contain the same data as $\tilde{x}$ but written into the entries with the fitting scale t. Let s be the number of scales we are using. Thus the final solution vector has the form

$$z = [x; \tilde{x}; v; u; s; y]$$

with

$$x_{ij}{}^{h} \quad = \begin{cases} 1 & \text{a cable of turbine capacity } h \text{ is required from } i \text{ to } j \\ 0 & \text{otherwise} \end{cases}$$

for $i = 1...n, j = 1...n, h = 1...k,$

$$\tilde{x}_{i,j}{}^{h,t} = \begin{cases} 1 & \text{a cable of category } h \text{ is used from } i \text{ to } j\text{, it has cableprice of scale } t \\ 0 & \text{otherwise} \end{cases}$$

for $i = 1...n, j = 1...n, h = 1...k, t = 1...s,$

$$v_i{}^{l} \quad = \begin{cases} 1 & \text{if turbine } i \text{ is connected with } l \text{ cables} \\ 0 & \text{otherwise} \end{cases}$$

for $i = 1, l = 1...n-1$ and $i = 2...n, l = 1...k,$

$$u_h \quad = \begin{cases} 1 & \text{if we use the cable of category } h \text{ in the layout} \\ 0 & \text{otherwise} \end{cases}$$

for $h = 1...k$

and

$$y_{ij}{}^{h,g} \quad = \begin{cases} 1 & \text{while a cable of turbine capacity } h \text{ is needed we use the cable of category } g \text{ from } i \text{ to } j \\ 0 & \text{otherwise} \end{cases}$$

for $i = 1...n, j = 1...n, h = 1...k, g = 1..k.$

Input data

Since this is the final step in the model, here the complete list of the input data is given:

- distances matrix: $D \in \mathbb{R}^{n \times n}$ where $D(i, j)$ contains the realistic distances between turbines i and j.
- ohmic losses: $L \in \mathbb{R}^{k \times k}$, the matrix where $L(h, g)$ contains the monetary loss in €$/m$ for a cable of category h
- dielectric losses: $\tilde{l} \in \mathbb{R}^k$, the vector where $\tilde{l}(h)$ contains the monetary loss in €$/m$ for a cable of category h.
- cable connection cost to the turbines: the vector $r \in \mathbb{R}^k$ where $r(l)$ gives the cost to connect l cables to a turbine.
- cable connection cost to the substation: the vector $rs \in \mathbb{R}^{n-1}$ where $rs(l)$ gives the cost to connect l cables to the substation.
- $maxR$: the maximal number of cables that the user wants to be connected to one turbine, its default value is k.
- $maxRS$: the maximal number of cables that the user wants to be connected to the substation, its default value is $n - 1$.
- $maxCables$: the maximum number of cables that shall be used, it is chosen by the user.
- cable cost matrix $P \in \mathbb{R}^{k \times t}$: the cable cost vector p is changed into a matrix which contains in the entry $P(h, t)$ the cost for a cable of category h if we need to pay the price of scale t.
- scales matrix $S \in \mathbb{R}^{k \times t}$: the matrix where $S(h, t)$ gives the lower bound for the cable meters that need to be bought to pay a price of scale t for a cable of category h.

In the cost vector

$$c = [e_1; \tilde{c}; c'; e_2; \mathfrak{c}]$$

the calculation of $\tilde{c}$ is adapted to the scaled model and the empty vector e_2 is extended, as the values of s do not have an influence on the cost:

$$\tilde{c}_{ij}^{ht} = D(ij) \cdot (P(ht) + \tilde{l}(h)),$$
$${c'_1}^{l} = rs(l), {c'_i}^{l} = r(l),$$
$${\mathfrak{c}_{ij}}^{hg} = D(ij) \cdot L(h, g)$$

and

$$e_1 \in \mathbb{R}^{n \cdot n \cdot k}, e_1(s) = 0, e_2 \in \mathbb{R}^{k+s}, e_2(h) = 0.$$

So the final cost function $c^T z$ has the form

$$\sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{h=1}^{k} \sum_{t=1}^{s} \ {\tilde{x}_{ij}}^{ht} \tilde{c}_{ij}^{ht} + \sum_{l=1}^{n-1} {c'_1}^{l} {v_1}^{l} + \sum_{i=2}^{n} \sum_{l=1}^{k} c'_l {v_i}^{l} + \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{h=1}^{k} \sum_{g=1}^{k} {y_{ij}}^{hg} {\mathfrak{c}_{ij}}^{hg}.$$

4.6.1. Complete enumeration of all the constraints

n is the number of stations to connect, k the category of the biggest cable and s the number of scales.

(1*)

$$x_{ij}{}^h, v_i^l, \; \tilde{x}_{ij}{}^{ht}, u_h, y_{ij}{}^{hg} \in \{0,1\}$$

for all $i = 1..n, j = 1...n, h = 1...k, g = 1...k, t = 1...s, l = 1...k$ and $l = 1...n-1$ respectively.

(2)

$$\sum_{j=1}^{n}\sum_{h=1}^{k} x_{ij}{}^h = 1$$

for all $i = 2...n$.

(3)

$$1 + \sum_{i=1}^{n}\sum_{h=1}^{k} h x_{ij}{}^h - \sum_{r=1}^{n}\sum_{h=1}^{k} h x_{jr}{}^h = 0$$

for all $j = 2...n$.

(4)

$$x_{ii}{}^h = 0$$

for all $i = 1..n, h = 1...k$.

(5)

$$x_{ij}{}^k = 0$$

for all $j = 2...n$.

(6)

$$\sum_{h=1}^{k} x_{ij}{}^h + x_{ji}{}^h \leq 1$$

and

$$\sum_{h=1}^{k} x_{ij}^h \leq 1$$

for all $i = 1..n, j = 1...n$.

(7)

$$\sum_{h=1}^{k} x_{ij}{}^h + x_{ji}{}^h + x_{uw}{}^h + x_{wu}{}^h \leq 1$$

for all $\{\{i,j\},\{u,w\}\} \in \chi$.

(8)
$$\sum_{l=1}^{k} l v_i{}^l = \sum_{j=1}^{n} \sum_{h=1}^{k} x_{ji}{}^h + 1$$

for all $i = 2..n$

and
$$\sum_{j=1}^{n} \sum_{h=1}^{k} x_{j1}{}^h = \sum_{l=1}^{n-1} l v_1{}^l$$

(9)
$$v_i{}^h = 0$$

for all $i = 1, h > maxRS$ and $i = 2...n, h > maxR$.

(10)
$$\sum_{h=1}^{k} v_1{}^h = 1 \quad \text{and} \quad \sum_{h=1}^{k} v_i{}^h = 1$$

for all $i = 2...n$.

Constraints (11′) to (18′) (without (12)) are the adaption of the constraints (11) to (18) to the extended $\tilde{x}$.

(11')
$$\sum_{h=1}^{k} x_{ij}{}^h h \leq \sum_{h=1}^{k} \sum_{t=1}^{s} \ \tilde{x}_{i,j}{}^{h,t} h$$

for all $i = 1..n, j = 1..n$.

(12)
$$\sum_{h=1}^{k} u_h \leq maxCables.$$

(13')
$$\sum_{t=1}^{s} \ \tilde{x}_{ji}{}^{ht} + x_{ji}{}^h - 1 \leq y_{ij}{}^{hg}$$

for all $i = 1..n, j = 1...n, h = 1...k, g = 1...k$.

(14')

$$\sum_{h=1}^{k}\sum_{g=1}^{k} {y_{ij}}^{hg} \leq 1$$

for all $i = 1..n, j = 1...n$.

(15')

$$\tilde{x}_{i,j}{}^{h,t} \leq u_h$$

for all $i = 1..n, j = 1..n, h = 1...k, t = 1...s$.

(16')

$$\sum_{j=1}^{n}\sum_{h=1}^{k} \tilde{x}_{ij}{}^{ht} = 1$$

for all $i = 2...n, t = 1...s$.

(17')

$$\sum_{h=1}^{k} \tilde{x}_{ij}{}^{ht} \leq 1$$

for all $i = 1...n, j = 1...n, t = 1...s$.

(18')

$$\sum_{h=1}^{k} \tilde{x}_{ij}{}^{ht} + \tilde{x}_{ji}{}^{ht} \leq 1$$

for all $i = 1..n, j = 1...n, t = 1...s$.

(19) If we do not use a cable of a certain cable category in the layout, so if $u_h = 0$ for a h, by constraint (19) we force all the ${s_h}^t$ variables to be zero meaning we leave the cable price for this category undefined.

$${s_h}^t \leq u_h$$

for all $h = 1...k, t = 1...s$.

(20) Here the definition of ${s_h}^t$ is implemented, for one cable there can not be more than one cable price. However as explained before there might be no price defined if we do not use the cable category.

$$\sum_{t=1}^{s} {s_h}^t \leq 1$$

for all $h = 1 ... s$.

(21) The last constraint is the core constraint to realize the scales concept. On the left side, for every cable category h only one ${s_h}^t$ is one. $S(ht)$ gives the lower bound of the amount of cable we need to buy of category h pay the price of scale t. So the sum $\sum_{t=1}^{s} {s_h}^t S(ht)$ represents the lower bound of the amount of cable we need to buy of category h to pay the cable price of the step defined in the ${s_h}^t$ variable.
On the right side of the equation $\sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{t=1}^{s} \ {\tilde{x}_{i,j}}^{h,t} D(ij)$ gives the amount of cable meters of cable category h used in the layout. Obviously it should be bigger than the lower bound given by the left side.

$$\sum_{t=1}^{s} {s_h}^t S(ht) \leq \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{t=1}^{s} \ {\tilde{x}_{i,j}}^{h,t} D(ij)$$

for all $h = 1 ... k$.

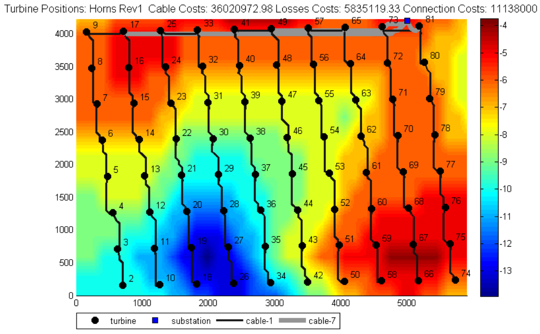
5. Comparison of the Results

The program that was developed in the Steps 1 to 6 can be applied to a lot of different configurations in any offshore wind farm. In this section we want to present some more results and compare them to each other.
We compare the quality of the solutions to those produced by the heuristics in [12] and present the amount of savings our program allows in comparison to the actual layouts of Horns Rev and Sandbank.
In chapter 3 we mentioned that different kinds of solvers were tried in the course of this work. Presenting runtimes for a selection of examples, it becomes clear why the bigger examples in this work were produced using Gurobi .
Finally in 5.4 we present some tests, examining whether using a start value can speed up the optimization.

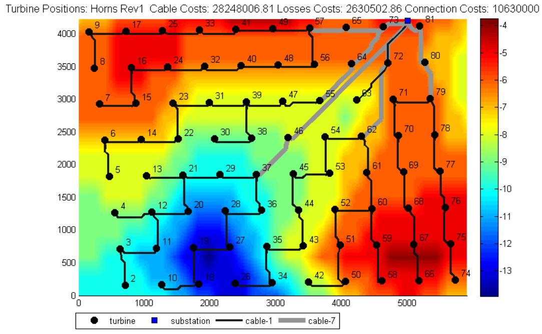
5.1. Horns Rev

In the first group of figures we compare the different results to the actual layout of Horns Rev wind farm. Therefore in our examples we only allow the cable types number 1 and 7 that were also used in the reconstruction of the actual layout (figure 19a). In figure 19b we see the solution proposed by the heuristics from [12]. Figure 19c shows the optimal solution produced by the ILP solver when using the same model as in [12]: Apart from the general consistency, crossings were avoided and power losses were considered in the optimization process. In the last figure 19d we find the solution from step 4 of our program, which also considers the connection costs. With the choice of cables being very restricted in this example, the solutions of step 3 and 4 happen to be equal. Nevertheless in order to choose the optimal layout, the connection costs should be included.

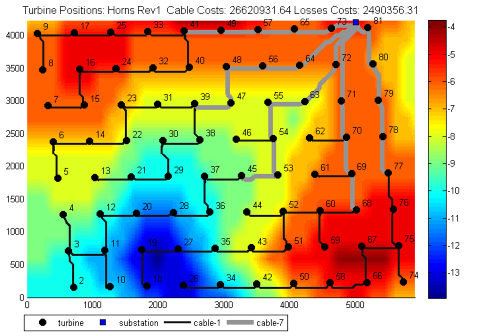
The following table 9 lists the costs for the different versions, showing that using the solution from step 3 or 4, over 13 million euros could have been saved, which amounts to a saving of 25% of the original cable cost!
Compared to the heuristic solution the ILP solution saves additional 1.7 million euros, which is still an improvement of another 4.25%. In the lower part of the table we also list the examples that did consider all the cables from our cable selection. As can be seen, the layouts here are even cheaper but in comparison the additional savings are not that substantial.



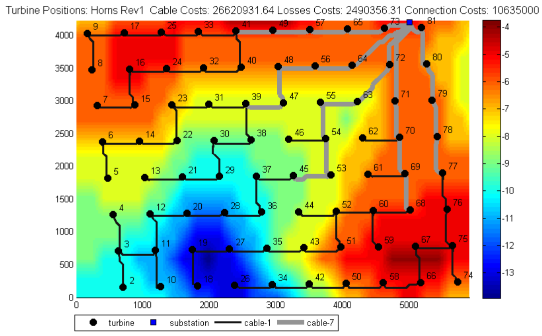
(a) Actual layout



(b) Layout using heuristics from [12]



(c) Optimal layout Step 3



(d) Optimal layout Step 4

Figure 19: Comparison of Horns Rev cabling layouts, costs in €

	Cable cost	Losses cost	Connect. cost	Overall cost	Savings
(a) Actual layout	36.021	5.835	11.138	52.994	
Using only the cables of the actual layout					
(b) Heuristics layout	28.248	2.631	10.630	41.509	21.67 %
(c) Layout Step 3	26.621	2.490	10.635	39.746	25.00 %
(d) Layout Step 4	26.621	2.490	10.635	39.746	25.00 %
Using all the cables in our selection					
(b*) Heuristics layout	27.725	2.531	10.645	40.901	22.82 %
(c*) Layout Step 3	26.240	2.720	10.640	39.600	25.27 %
(d*) Layout Step 4	26.237	2.728	10.625	39.590	25.29 %

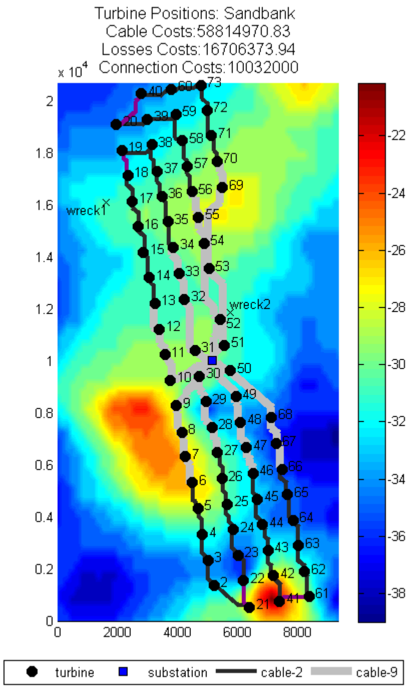
Table 9: Prices of Horns Rev cable layouts in million euros

5.2. Sandbank

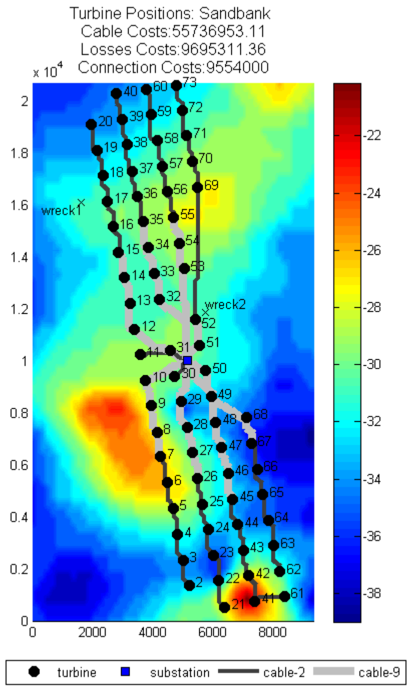
The same comparison was done for the Sandbank example. We use the cables of type 2 and 9. As the property is bigger the overall cable costs are higher. As the following table 10 shows the percental savings here were not as high as for Horns Rev. Due to the longish wind farm property there are less reasonable alternatives to connecting the turbines in a linear layout as it was also done in the actual layout. However the wind farm is going to be extended in the near future. When connecting the rather irregular additions, a separate optimization for the new cabling could be helpful.
However, we find that even now our version of the layout as in (c) or (d) saves 10.6 million euros.
The corresponding figures to the examples can be found on the next page.

	Cable cost	Losses cost	Connect. cost	Overall cost	Savings
(a) Actual layout	58.8150	16.7064	10.0320	85.5534	
Using only the cables of the actual layout					
(b) Heuristics layout	55.737	9.695	9.554	74.986	12.35 %
(c) Layout Step 3	55.363	9.160	9.554	74.077	13.41 %
(d) Layout Step 4	55.306	9.219	9.549	74.074	13.42 %
Using all the cables in our selection					
(b*) Heuristics layout	54.411	11.544	9.564	75.519	11.73 %
(c*) Layout Step 3	52.218	11.602	9.554	73.374	14.24 %
(d*) Layout Step 4	52.218	11.602	9.554	73.374	14.24 %

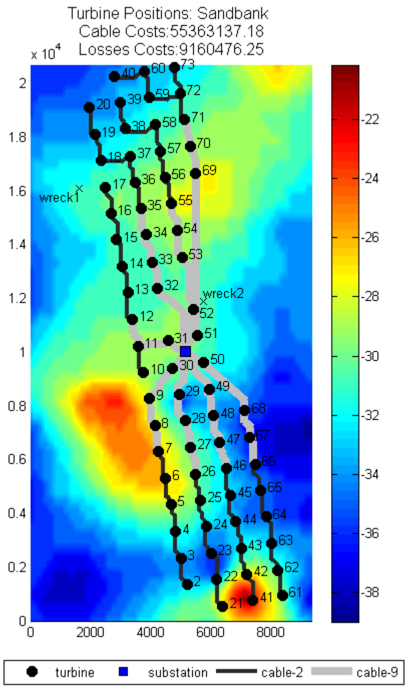
Table 10: Prices of Sandbank cable layouts in million euros



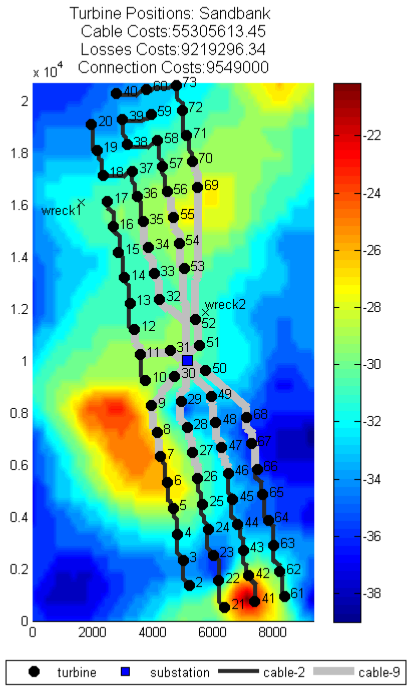
(a) Actual layout



(b) Layout using heuristics from [12]



(c) Optimal layout Step 3



(d) Optimal layout Step 4

Figure 20: Comparison of Sandbank cabling layouts, costs in €

5.3. Runtimes

To show the different runtimes of the ILP solvers from Gurobi, Cplex and Matlab, we were running a number of examples on Sandbank using the first step of our program. Since the solvers are not performing equally good, we start with small examples, not using the full number of turbines. They were run on an Intel(R)i7-3612QM processor with 4 cores/ 8 virtual cores and 8 GB of primary memory.
As can be seen in table 11 runtimes usually increase as the size of the problem increases. However, every ILP is different and as the problem of solving them is NP-hard, runtimes cannot be predicted. It is not typical, but possible that a bigger problem is solved more quickly than a smaller one, as the examples of Cplex with 15 and 20 turbines show. In every example runtime depends on the specific structure of the search tree and the heuristics found.
As we see in the table Gurobi was by far the best solver in our examples and we found it working quite well even on larger versions of the problem. The Matlab solver however started to have problems solving the model when we were using only 40 turbines, a problem that took the other solvers 1.36 and 15.45 seconds. Contrary to both of the other solvers Matlab does not run parallel, which seems to an have extreme impact even on relatively small problems.

Number of turbines	Gurobi	Cplex	Matlab
10	0.04 s	3.42 s	0.42 s
15	0.19 s	2.39 s	4.66 s
20	0.16 s	0.94 s	61.56 s
30	0.89 s	2.34 s	10.2 min
40	1.36 s	15.45 s	>24 h
50	4.98 s	44.08 s	>24 h
60	25.15 s	12.2 min	>24 h
73	76.99 s	14.2 min	>24 h

Table 11: Sandbank step 1, runtime for different solvers with different numbers of turbines

Using only the Gurobi Intlinprog solver, we were running most of our bigger examples on a better computer, offering 30 cores and a primary memory of 1 TB. Thereby we achieved the following runtimes. For unfinished runs we give some data on the duality gap, the difference between the current upper and lower bound of the objective function. As can also be seen in figure 21 it is usually developing hyperbolicly. The examples (*) have not been run. As can be seen in the table the Horns Rev examples need much more computation time as those of Sandbank, thus in step 5 and six we assume their runtimes to exceed the acceptable scale.

	Horns Rev	Sandbank
Step 1	445 s	30 s
Step 2	451 s	30 s
Step 3	1255 s	20 s
Step 4	1095 s	79 s
Step 5	*	
Step 6	*	1 h 8 min: 79.7% 3 d 9 h: 1.05% 11 d: 0.46%

Table 12: Runtimes for the different steps of the problem

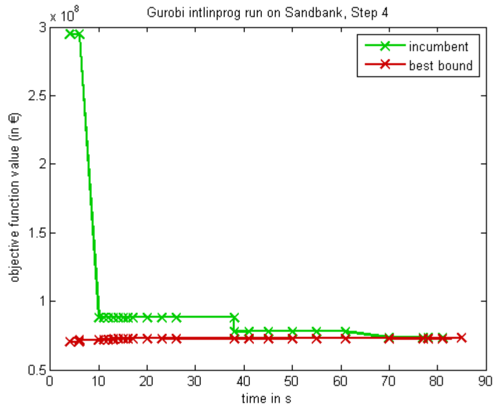
5.4. Using a start value

The ILP solvers we used also allow us to use start values. Since the heuristic from [12] generates a quick solution that is often close to the optimal solution, it can be used as a start value. If the solver would be given a good feasible solution, this could speed up the optimization process. As there would be a low incumbent from the beginning on, unpromising branches can already be cut off.
Since the heuristic solution is based on the model we use in step three, it usually only makes sense to use it as a start value up to that level. Still we were extending the heuristic solution by counting the number of cables connected to every turbine to use it as an input for step 4. To try out start values for higher steps, creating an adjusted heuristic would be recommended. Unfortunately this extended the scope of this work.
In table 13 runtimes for Gurobi with and without start value for step 1, 3 and 4 are compared. In the column titled start value the first number gives the time to produce the heuristic solution, the second value the time Gurobi took for the ILP using the heuristic solution as start value. However we found that using a start value did not always speed up the optimization. Although in some cases it was helpful, it cannot be used as a reliable tool.

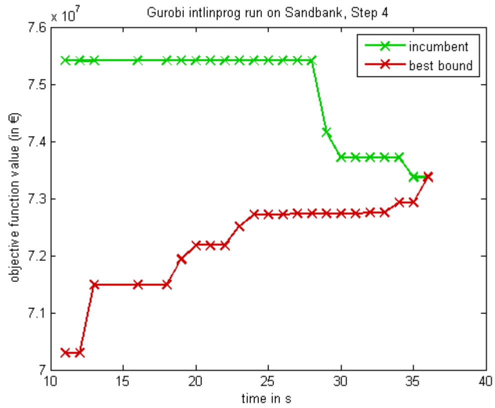
Step	Number of turbines	with start value	without start value
1	10	0.07 +0.06	0.04
1	20	0.27 +0.15	0.16
1	30	0.79 +0.77	0.89
1	40	2 + 1.43	1.36
1	50	3.65 +6.05	4.98
1	60	6.15 +6.4	25.15
1	73	17.2 + 58.9	76.99
3	73	17.2 +62.85	71.28
4	73	17.2 + 36.06	85.14

Table 13: Runtimes for Sandbank with and without start value

Surprisingly the heuristic solution was speeding up the optimization in step four significantly. As an example we will display the development of the duality gap during this run in figure 21. It shows the currently found best solution, the incumbent as well as the best bound, which gives the current lower bound to the possible optimal solution. As we can see the duality gap, the difference of the two values, is decreasing until it is zero. Figure 21a depicts the run without a start value, in figure 21b the heuristic was used as start value. Consider the different scale of the y-axis. When the incumbent does not decrease in the last steps, as can be noted in 21b, the optimal solution has been found before, but some further steps were necessary to prove the optimality of the result.



(a) Without start value



(b) With start value

Figure 21: Developement of the duality gap during the run of Gurobi Intlinprog

6. Outlook and Conclusion

Although there are a lot of features considered in our wind farm model, in the course of this work some aspects came up that exceeded the framework of this thesis. They will be explained in the following so that future work on the topic can start where we finished.

- **Exact cable crossings detection**: As explained earlier the current detection of cable crossings does not actually match with the realistic cable courses but is bases on straight connections between the turbines. To avoid having to change the recommended cabling of the affected cable manually an adjusted cable crossing detection is recommended. We suggest it uses the grid structure we used to save sea depths and actual routes between the turbines. Marking the grid points of the routes used in the solution, the squares where cables cross can be identified.

- **Improved calculation of ohmic losses**: In the calculation of the ohmic losses we were using the average current to approximate the ohmic loss values over the course of the operating lifetime of the wind farm. If long-term data on the produced current (at least over the course of a year to even out seasonal effects) is available, in the calculation the root mean square of the current should be used instead of the average current. Since in the losses calculation the value is squared, the resulting losses are equal to the average of the losses at the used times. Thus the more data available the more exact is this method.

- **Additional substations**: If needed for a wind farm the constraints used in step 1 to 6 can be easily adapted to a model with two (or more) substations. This can be achieved by fixing the stations indexed by 1 and 2 as the substations and change the indexing in the constraints accordingly.

- **Wind park expansions**: If a wind farm is extended obviously our program can be used to separately connect the new turbines to the substation. However, it can be used to find an even better solution. The complete layout can be optimized adding constraints that fix the already existing cable connections. This way potentially existing unused cable capacities could be put to use. The solution generated will most likely not be as good as the optimization on an unconnected configuration, but it would be the best one possible while leaving the existing cables unchanged. If changing cables is an option, this should be integrated into the program with consideration to the arising costs.

- **Expand heuristic**: The current heuristic from [12] is only on the level of step 3, meaning that it cannot be used to produce good start values for the steps four to six. A good heuristic could speed up the usually very long optimization processes for those steps.

Overall the approach using ILP was very successful. The complete wind farm model could be described linearly. We considered the cost of the cables, the losses and the connection to the turbines and allowed scaled costs. The additional constraints (no cable crossings, best cable choice and optional restrictions on the amount of branching) were implemented linearly as well as the constraints describing the general structure of a wind farm cabling. This way the model allowed us to find optimal layouts respecting the given conditions as well as parameters chosen by the user. With the Intlinprog Gurobi solver a tool was found that can efficiently handle integer linear problems at least if their size stays in an acceptable dimension. Thus a problem that is NP-hard could be solved in reasonable time.

In addition we can confirm that the results of the heuristic approach used in [12] do usually not differ more than 5 % from the optimal solution. Thus they can be used if a quick estimate on the optimal cost is needed, such as in algorithms that determine the placement of the turbines. Since many different placements are to be considered, in this application speed is very relevant. After the turbine positions are fixed, the approach using ILP solvers can be applied, determining the optimal infield power collection cabling.

We highly recommend that wind farm operators will start using this approach when looking for future cabling layouts. As presented in chapter 5 the savings achieved compared to the manually created layout are extremely high.

A. Appendix: Wind farm data

A.1. Turbine positions of Horns Rev

	x	y
OSS	5000	4200
2	703.881	157.563
3	626.070142857	711.759714286
4	548.259285714	1265.95642857
5	470.448428571	1820.15314286
6	392.637571429	2374.34985714
7	314.826714286	2928.54657143
8	237.015857143	3482.74328571
9	159.205	4036.94
10	1262.01311111	166.829333333
11	1184.20225397	721.026047619
12	1106.39139683	1275.2227619
13	1028.58053968	1829.41947619
14	950.76968254	2383.61619048
15	872.958825397	2937.81290476
16	795.147968254	3492.00961905
17	717.337111111	4046.20633333
18	1820.14522222	176.095666667
19	1742.33436508	730.292380952
20	1664.52350794	1284.48909524
21	1586.71265079	1838.68580952
22	1508.90179365	2392.88252381
23	1431.09093651	2947.0792381
24	1353.28007937	3501.27595238
25	1275.46922222	4055.47266667
26	2378.27733333	185.362
27	2300.46647619	739.558714286
28	2222.65561905	1293.75542857
29	2144.8447619	1847.95214286
30	2067.03390476	2402.14885714
31	1989.22304762	2956.34557143
32	1911.41219048	3510.54228571
33	1833.60133333	4064.739
34	2936.40944444	194.628333333
35	2858.5985873	748.825047619
36	2780.78773016	1303.0217619
37	2702.97687302	1857.21847619
38	2625.16601587	2411.41519048
39	2547.35515873	2965.61190476
40	2469.54430159	3519.80861905
41	2391.73344444	4074.00533333

	x	y
42	3494.54155556	203.894666667
43	3416.73069841	758.091380952
44	3338.91984127	1312.28809524
45	3261.10898413	1866.48480952
46	3183.29812698	2420.68152381
47	3105.48726984	2974.8782381
48	3027.6764127	3529.07495238
49	2949.86555556	4083.27166667
50	4052.67366667	213.161
51	3974.86280952	767.357714286
52	3897.05195238	1321.55442857
53	3819.24109524	1875.75114286
54	3741.4302381	2429.94785714
55	3663.61938095	2984.14457143
56	3585.80852381	3538.34128571
57	3507.99766667	4092.538
58	4610.80577778	222.427333333
59	4532.99492063	776.624047619
60	4455.18406349	1330.8207619
61	4377.37320635	1885.01747619
62	4299.56234921	2439.21419048
63	4221.75149206	2993.41090476
64	4143.94063492	3547.60761905
65	4066.12977778	4101.80433333
66	5168.93788889	231.693666667
67	5091.12703175	785.890380952
68	5013.3161746	1340.08709524
69	4935.50531746	1894.28380952
70	4857.69446032	2448.48052381
71	4779.88360317	3002.6772381
72	4702.07274603	3556.87395238
73	4624.26188889	4111.07066667
74	5727.07	240.96
75	5649.25914286	795.156714286
76	5571.44828571	1349.35342857
77	5493.63742857	1903.55014286
78	5415.82657143	2457.74685714
79	5338.01571429	3011.94357143
80	5260.20485714	3566.14028571
81	5182.394	4120.337

A.2. Turbine positions of Sandbank

	x	y
OSS	5141.07	10030.63
2	5184.23	1362.58
3	4996.68	2342.76
4	4809.19	3321.83
5	4654.62	4331.09
6	4467.36	5311.29
7	4249.92	6322.51
8	4124.90	7268.53
9	3939.53	8279.90
10	3752.63	9259.03
11	3565.82	10238.17
12	3379.14	11218.44
13	3225.31	12227.77
14	3039.43	13206.92
15	2853.40	14187.22
16	2666.72	15166.41
17	2480.50	16145.61
18	2327.16	17156.09
19	2140.15	18104.17
20	1955.20	19114.55
21	6362.86	538.19
22	6176.59	1549.44
23	5988.80	2528.46
24	5832.97	3506.52
25	5645.40	4486.67
26	5457.90	5465.72
27	5303.30	6474.95
28	5116.02	7455.14
29	4928.81	8434.21
30	4710.51	9414.26
31	4556.31	10424.65
32	4214.39	12381.89
33	4027.70	13362.13
34	3842.67	14372.41
35	3656.18	15352.67
36	3469.76	16331.831
37	3315.19	17310.01
38	3129.97	18321.45
39	2975.58	19299.64
40	2789.54	20278.84

	x	y
41	7356.27	756.46
42	7168.21	1736.54
43	6980.86	2715.50
44	6792.96	3694.48
45	6637.02	4672.51
46	6482.14	5682.81
47	6262.68	6662.80
48	6075.21	7642.93
49	5919.63	8621
50	5733.90	9631.17
51	5546.72	10611.34
52	5391.40	11589.43
53	5017.45	13547.59
54	4862.44	14526.82
55	4677.30	15537.05
56	4458.88	16517.15
57	4304.15	17496.40
58	4149.47	18474.55
59	3932.37	19486.95
60	3777.87	20465.11
61	8348.65	943.81
62	8191.29	1890.62
63	8004.02	2900.70
64	7816.58	3880.74
65	7628.57	4859.69
66	7440.66	5838.65
67	7252.88	6818.73
68	7097.94	7827.90
69	5478.13	16671.69
70	5292.98	17681.91
71	5106.37	18662.11
72	4951.57	19640.23
73	4765.11	20619.34

References

[1] AlphaVentus. Der erste deutsche Offshore-Windpark. https://www.alpha-ventus.de/ueberblick.

[2] Anita Amberg, Wolfgang Domschke, and Stefan Voß. Capacitated minimum spanning trees: Algorithms using intelligent search. 1996.

[3] Andrew. Capacity factors at Danish offshore wind farms. http://energynumbers.info/capacity-factors-at-danish-offshore-wind-farms, 2016.

[4] Joanna Bauer and Jens Lysgaard. The offshore wind farm array cable layout problem: a planar open vehicle routing problem. *Journal of the Operational Research Society*, 66(3):360–368, 2014.

[5] Chad Casarotto. Graph Theory and Cayley's Formula. Master's thesis, University of Chicago, 2006.

[6] Christophe Duhamel, Luís Gouveia, Pedro Moura, and Maurício Souza. Minimum cost degree constrained spanning trees with node-degree costs. Technical report, Centro de Investigação Operational, 2007.

[7] Danish energy agency. Data on operating and decommissioned wind turbines (as at end May 2016). http://www.ens.dk/sites/ens.dk/files/info/tal-kort/statistik-noegletal/oversigt-energisektoren/stamdataregister-vindmoeller/anlaegprodtilnettet.xls, May 2016.

[8] Gurobi. Mixed-Integer Programming (MIP) - A Primer on the Basics. http://www.gurobi.com/resources/getting-started/mip-basics.

[9] Kurt S Hansen, Rebecca J Barthelmie, Leo E Jensen, and Anders Sommer. The impact of turbulence intensity and atmospheric stability on power deficits due to wind turbine wakes at Horns Rev wind farm. *Wind Energy*, 15(1):183–196, 2012.

[10] Christoph Kost. Levelized Cost of Electricity- Renewable Energy Technologies. study, Frauenhofer, November 2013.

[11] Lorc. Horns Rev 1 wind farm. http://www.lorc.dk/offshore-wind-farms-map/horns-rev-1.

[12] Julia Lust. Shortest cable routing in offshore wind farms, 2016.

[13] MathWorks. Choosing a Solver. http://de.mathworks.com/help/optim/ug/choosing-a-solver.html?requestedDomain=www.mathworks.com.

[14] Nexans. Submarine Power Cables. http://www.nexans.no/Germany/2013/SubmPowCables_FINAL_10jun13_engl.pdf, 2013.

[15] Christos H Papadimitriou. The complexity of the capacitated tree problem. *Networks*, 8(3):217–230, 1978.

[16] Yannik Paris. private communication, 2016.

[17] Josef Schachner. *Power connections for offshore wind farms*. na, 2004.

[18] *Paris Agreement*, December 2015. United Nations.

[19] Vattenfall. Technik und Komponenten. https://corporate.vattenfall.de/uber-uns/geschaftsfelder/erzeugung/bauprojekte/sandbank/technik-komponenten, .

[20] Vattenfall. Das Windpark-Gebiet Sandbank. https://corporate.vattenfall.de/uber-uns/geschaftsfelder/erzeugung/bauprojekte/sandbank, .

[21] Walney. Walney 2 offshore wind farm, 2011. URL http://www.kis-orca.eu/media/32125/Walney_2_LResx.pdf.