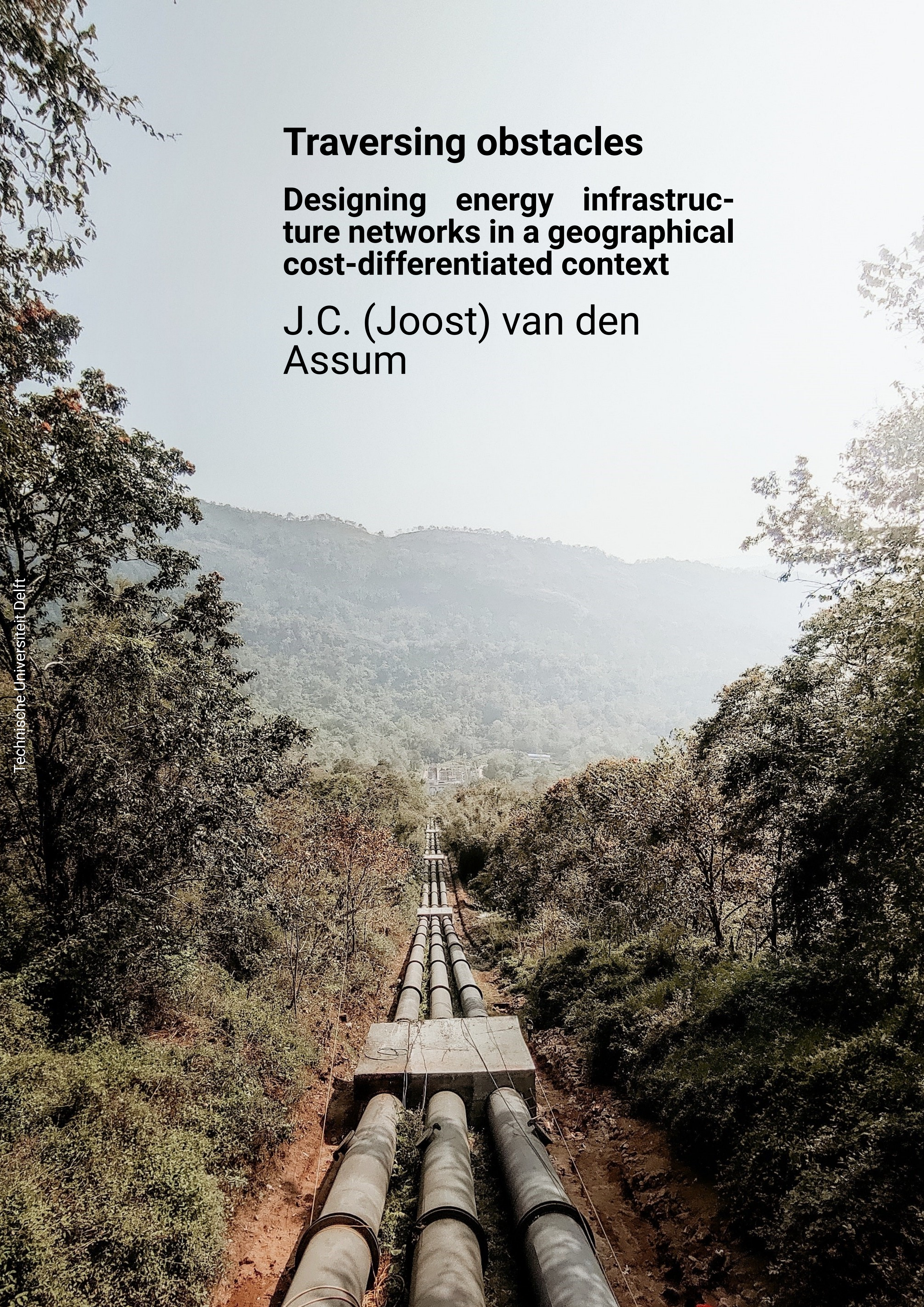
Traversing obstacles

Designing energy infrastructure networks in a geographical cost-differentiated context

J.C. (Joost) van den Assum

Technische Universiteit Delft



Traversing obstacles

Designing energy infrastructure networks in a geographical cost-differentiated context

by

J.C. (Joost) van den Assum

Master thesis submitted to Delft University of Technology
in partial fulfilment of the requirements for the degree of
MASTER OF SCIENCE
in Complex System Engineering And Management (CoSEM)
Faculty of Technology, Policy and Management
to be defended publicly on Wednesday August 24, 2022 at 10:30 PM.

Student number:	4460537	
Project duration:	February 7, 2022 – August 24, 2022	
Thesis committee:	Dr. A.F. (Aad) Correljé	TU Delft, chair
	Dr. ir. P.W. (Petra) Heijnen,	TU Delft, first supervisor
	Dr. A.F. (Aad) Correljé	TU Delft, second supervisor

An electronic version of this thesis is available at http://repository.tudelft.nl/.
The cover photo was shot by Ambady Kolazkikkaran [44].

Acknowledgement

Dear reader, the research report on your screen or in your hands contains my Master Thesis. A project that I have been working on for the past couple of months. It is the final work of my Master Complex System Engineering and Management and the conclusion of the six educational and bewildering years I got to spend at the Technical University of Delft. Finishing this last chapter of my studies would certainly not have been possible without the advice, help and support of my thesis committee to whom I would like to express my gratitude. First of all, I would like to thank Petra Heijnen, my first supervisor. Your constructive critiques during our weekly meetings and your amazingly quick responses to my many emails helped me immensely and ensured that I never got stuck on a topic for long. Furthermore, you were always able to keep me on track and made sure I saw the red line and bigger picture in my own work when I tended to pay too much attention to details. Your work has inspired me to choose this interesting topic and I am certain that you will inspire many more students in the years to come. Secondly, I would like to thank, Aad Correljé, my second supervisor. Your sharp critiques during the milestones of my thesis process helped me to sharpen my arguments and look at my work from a different perspective. Without your advice, my thesis would have been a lot less multidisciplinary. I would like to thank you specifically for your lectures that I got to follow over the years as well, which sparked my interest in energy infrastructures and contributed to the choice of the topic of this research report.

Furthermore, I would like to thank my friends and family for their support during the last months. There are a few people I would like to thank in particular. A large thank you to my parents who I can always fall back on and show me that it will all turn all right, at moments when I cannot do so myself. Next, I would like to thank Emma. Although your efforts of trying to discipline me in taking the early morning trains have not always paid off, the countless hours of mutual studying, coffee breaks, conversations and inside jokes have made the past months a lot more enjoyable. Furthermore, I would like to thank my friend David who did not let a day go by without calling me to discuss ideas, insights or arguments. I have never been destined to be a graphic designer, but thanks to your countless QGIS tutorials, I can now proudly present my figures. But most of all, I would like to thank you for the shared laughter which provided much-needed breaks during the last weeks of writing our theses. Moreover, my thanks goes out to my roommates, Tom, Tijmen, Thomas and Joris, with whom I could celebrate when I wrote a line of code that worked and by whom I could complain when it didn't. Lastly, I would like to thank my almost roommates, Sharif, Tom, Romeo and Nandor, at whose place I could always relax after a long day of studying. Your interest in my studies prompted me to describe my research in a way that was not TU Delft-specific, which greatly improved my own understanding of them.

Abstract

The decarbonization of economies around the world is crucial for reducing the impact of human-induced climate change. Many proposed means to achieve this decarbonization like the electrification of various sectors or the introduction of 'new' forms of energy such as hydrogen and carbon capture and storage require existing energy infrastructures to be expended or entirely new energy infrastructures to be created. Since energy networks are capital intensive, minimizing their construction costs is essential for their realization. Previous works studying the cost minimization of energy networks often neglect that their construction costs can be influenced by geographical areas such as mountain ranges, existing infrastructures or zoning rules. Using methods borrowed from graph theory and geometrical computing, a method has been developed that is able to find cost-optimal energy infrastructures in a spatial context of geographic regions with different costs for constructing power cables or pipelines through them. The method models these geographical areas as triangles on which borders a user-defined number of points are uniformly placed representing potential entry and exit points for pipelines or power cables.

Experimenting with 144 randomly generated routing problems has shown that, on average, increasing the number of these placed border points results in a reduction of the total network costs. Although, this effect flattens when more than 7 of these points are placed. The method is applied to two offshore electricity networks in the Dutch North sea. These cases demonstrate the method's strength in identifying trade-offs between energy networks' investment costs and their (negative) influence on their spatial surroundings. Additionally, the method's outcomes are well-suited to be used as a springboard for dialogue in energy infrastructure decision-making processes. Further research could focus on the extension of the model, by including existing connections, or the improvement of the model, by implementing a different method for modelling geographical areas that has a higher accuracy and is computationally faster.

Contents

List of Figures

List of Tables

List of Equations

1. Introduction

1.1 A growing demand for new energy infrastructures

Greenhouse gas emissions emitted by humans are raising the temperature of the atmosphere, ocean and land. Today, climate extremes such as heat waves, heavy precipitation, droughts and tropical cyclones are already more frequent due to this human-caused climate change. If greenhouse gas emissions are not significantly reduced in the coming decades, these weather extremes will become even more common and dire [39]. In order to reduce these negative effects of human-induced climate change, 175 countries have agreed to make an effort to keep the global average temperature below 2°C above pre-industrial levels with the aim of limiting the increase of the global average temperature to 1.5°C by decarbonizing their economies.

The electrification of economies around the world, combined with increased investments in renewable energy sources and the introduction of 'new' forms of energy such as hydrogen are means that can contribute to the reduction of greenhouse emissions while supporting the global economy with the energy that it needs [38]. Successfully satisfying these sustainability energy challenges requires existing infrastructures to be expended or entirely new energy infrastructures to be created [32]. Examples of energy networks that contribute to the reduction of CO_2 emissions are [33]:

- Biogas networks can be employed to reduce the consumption of natural gas. Farms in rural areas often produce a larger volume of biogas than they need to satisfy their own demand for energy. With a biogas network, the superfluous biogas of these farms could be transported to other nearby farms that are not self-sufficient in their energy demand. Another option is to connect the farms to existing infrastructures so they can supply local or national gas networks.

- Actors in industries that emit a lot of greenhouse gas emissions often advocate for carbon capture and storage (CCS). It is a method that captures CO_2 directly at the location where it is produced like power plants or steel mills and subsequently transports it as a commodity to storage facilities like subsurface salt caverns, instead of emitting the gas to the atmosphere. There are plenty of underground formations in which the entire global greenhouse gas emissions could be stored for years. However, the availability of storage capacity differs significantly between continents [53]. Hence, constructing a large-scale CO_2 network could arguably contribute to a significant reduction of greenhouse gas emissions.

- Hydrogen could play a major role in the energy mix of the future with its versatile applications in industry, transport, the built environment and the electricity sector [18]. A major barrier to the energy carrier's development is the lack of a large-scale transport infrastructure [5]. Realising such a large-scale network in Europe would allow for the cost-effective transportation of hydrogen from countries where it will be cheap to produce, like the Netherlands which will have access to a lot of offshore generated wind energy, to densely populated areas with a high demand for energy such as Northern Italy and Southern Germany [36].

- The North Sea is surrounded by countries like the UK, Germany, Denmark, Norway, Sweden, Belgium and the Netherlands that all have huge ambitions to develop offshore wind parks in the coming decades. The latter country even plans on increasing its installed offshore wind capacity from 2.5 GW to 61.5 GW in the next thirty years [66]. Designing an offshore electricity infrastructure that is able to transport the generated energy to shore will be crucial for the realization of these ambitions.

All of these energy infrastructures that transport different energy commodities share several characteristics that complicate their development. The energy networks all have a long lifetime that often spans over several decades, which makes altering them once they are developed difficult and the design of their layout and capacities all the more important. Moreover, these networks are developed in a complex and uncertain social-technical context, with unknown future participants whose demand of supply can fluctuate over the years. Lastly, these networks are all capital intensive. The design and development of infrastructures usually require billions of euros of capital investment [55]. Minimizing the development costs of energy infrastructures is therefore essential for their development.

1.2 The identified knowledge gap: lack of spatial cost-differentiation in energy infrastructure modelling

One method to minimize the construction costs for energy infrastructures is to base the layout design of energy infrastructures on the outcomes of energy network cost-optimization models. There are many academic works that use models to search for cost-optimal energy networks for the design of networks that transport different commodities like hydrogen, biogas and liquefied natural gas [5, 33, 34]. However, most of these works that studied the cost-optimization of energy infrastructures neglect demarcated spatial areas that complicate the routing of pipelines on power cables [36].

These spatially demarcated regions can be geographical obstacles like bodies of water or mountains, existing infrastructures or buildings or zoning rules (e.g. protected areas [4]). If cost-minimization models of energy infrastructures do make a distinction between different types of terrain, it is only between urban and rural areas [84], or onshore and offshore locations [43]. In other studies, different geographical limitations are considered, but are modelled in the form of exclusive boundaries or no-go areas that pose strict limitations on potential pipeline or power cable routes [34]. In actuality, these regions do allow for pipeline or power cables to be developed in these areas, but at greater costs. Including these spatial hindrances in energy network modelling would improve the alignment of these models with reality [32].

Surprisingly, despite the fact that it has never been studied to the knowledge of the author, academic literature frequently mentions the added value of including spatial regions that allow for cost-differentiation in energy modelling. Almaraz et al. [16] remark for example that geographic constraints could be introduced in modelling or geographic information systems (GIS) could be used to validate the feasibility of results of conducted infrastructure planning research. The International Energy Agency [37] goes even further and claims that transmission costs can only be realistically estimated if knowledge of geographical parameters is available.

1.3 Research questions

To answer the academic call described in the previous section, to incorporate geographical hindrances into energy infrastructure modelling, this research strives to expand the knowledge base of energy modelling by incorporating geographical cost-differentiation zones. The following main research question has been formulated with the aim to close this discovered academic gap:

> *How can geographical cost-differentiation be included in energy network cost-minimization models?*

To answer the main research question above, the following five subquestions have been formulated:

1. How can cost-minimal energy networks be modelled?

2. How can geographical cost-differentiation regions be modelled?

3. What input parameters influence a geographical cost-differentiation energy network cost-minimization model's performance?

4. How does including geographical cost-differentiation in energy network modelling influence the discovered networks?

5. How can a geographical cost-differentiation energy network cost-minimization model be used in energy infrastructure decision-making processes?

1.4 Research structure

Chapter 2 answers the first two subquestions by sketching a theoretical framework based on academic literature on energy network modelling and mathematical problems that deal with spatial cost-differentiations. Next, chapter 3 uses the knowledge presented in the theoretical framework to describe the methodology used to develop the geographical cost-differentiation energy network cost-minimization model. Chapter 4 answers the third subquestion by visually and statistically analyzing the relations between the model's input and output parameters. Hereafter, chapters 5 and 6 answer the fourth subquestion by presenting and discussing the outcomes of two case studies of offshore electricity networks on the North Sea of different scales. Subsequently, chapter 7 provides the answer to the fifth subquestion by discussing the role of network models in energy infrastructure decision-making processes. Next, chapter 8 provides the research's conclusion after which chapter 9 will discuss the developed model's results, assumptions and demarcation. Lastly, chapter 10 gives recommendations for policy makers and future research.

2. Theoretical framework

This chapter provides an overview of the existing academic literature on the modelling of network energy infrastructures and geographical cost-differentiation regions with the aim to select the methods that are most suitable for incorporating geographical cost-differentiations in network energy modelling. Section 2.1 elaborates on three different approaches for solving networked system design problems and argues why network theory is the best suited approach for designing networked energy infrastructures. Hereafter, section 2.2 dives deeper into various energy infrastructure design problems that can be solved with a network theory approach. Next, section 2.3 describes and compares different algorithms that solve the Minimum Weighted Region Shortest Path Problem. Subsequently, section 2.4 discusses two algorithms that provide solutions to the Steiner Minimum Tree problem. Lastly, section 2.5 combines network theory and the weighted region problem, to introduce the Minimum Cost Weighted Region Steiner Tree problem.

2.1 Energy infrastructure modelling

The design of networks has enjoyed great scientific interest in the last six decades. During this period a wide variety of network design methodologies has been developed [77, 44]. Heijnen et al. [33] list three main modern approaches for resolving networked system design problems, being agent-based modelling, Mixed Integer (Non-)Linear Programming and graph theory. All of these approaches can be used to investigate the effects of a wide variety of parameters such as uncertainty in demand patterns or network participants and allows for multiple design criteria like reliability, resilience and cost-minimization [34, 32, 33, 5]. However, there are some key differences between the three modelling methods.

The first approach for modelling network design problems is using different agent-based models like ant colony optimization (ACO) or particle swarm optimization (PSO) models to search for cost-optimal network layouts. ACO models try to mimic the observed behaviours of real ant colonies. The ants of these colonies relieve a chemical substance called pheromone on the ground while walking from and towards a food source. Other ants use their ability to sense different concentrations of pheromone in their surroundings and tend to follow the routes where the pheromone concentration is higher. As an emerging effect, the ants are able to identify and transport sources of food in an effective manner, which makes simulating ant behaviour in models suitable for various routing problems [12]. Heijnen et al. [32] developed an ACO routing model to find a cost-optimal topology that connects a single source to multiple sinks. In his master thesis, Nguyen [57] expended this model to allow for situations with multiple sources and multiple sinks.

Particle swarm optimization (PSO) models use biologically derived algorithms that share the underlying rationale that a member of a group can benefit from the experiences of other members of the same group [41]. Liu et al. [51] used a PSO model for finding a Steiner Minimum Tree for the development of oil- and gas transportation networks.

A second approach formulates the networked system design problem as a Mixed Integer (Non-)Linear Programming (MI(N)LP) problem. MILP and MINLP problems try to optimize an objective function under a set of linear or non-linear constraints, respectively. Using the MI(N)LP approach to model energy infrastructures requires that all the potential power cable or pipeline connections are already laid down in these constraints beforehand. This is not a necessity for the other two approaches, which have more freedom in that aspect. A large benefit of the MILP approach is that it is not constricted to minimizing or maximizing one objective at a time, but can simultaneously optimize multiple objectives [30]. Thapalia et al. [77] have used a stochastic demand MINLP model to search for a cost-minimum single commodity - such as oil or gas - network consisting of a set of edges that connect multiple sources with multiple sinks. Gorenstein Dedecca et al. [28] used a MILP model to design an offshore electricity grid that connects offshore wind parks to the mainland.

The third approach and most popular methodology for modelling networks in use today is the approach of graph theory. Graph theory is a computationally fast modelling approach that has been studied and applied on numerous occasions in the energy field such as in the design of large-scale hydrogen networks [7, 65] and local steam pipeline networks [35]. Because the approach has been widely utilized in the domains of system engineering and energy network modelling, it is simpler to build upon these earlier academic efforts [33]. Furthermore, graph theory allows for quick visualisation of found solutions which makes it easy to communicate the method's results with stakeholders. Additionally, these visualisations contribute to the comprehensibility of the method's outcomes for actors who are not familiar with mathematical models. These observations are in line with the findings of Zarghami et al. [87] who claim that graph theory is well suited to support system engineers and policy makers in the design of energy infrastructures. On top of that, graph theory is the only of the three methods that was specifically developed to analyse networks, while ACO, PSO and MILP models originate from different fields of research.

Unlike the MI(N)LP approach, graph theory and agent-based modelling do not require that all potential connections are predetermined, which makes them more flexible and applicable to a larger range of network modelling problems. In a direct comparison of graph theory with an ABM ACO routing model, Heijnen et al. [34] found that graph theory was the more rigid of the two methodologies. These advantages over the methods combined with the gains that can be achieved by using graph theory that were stated earlier, make graph theory the most suitable method for modelling the cost-minimization of energy networks.

In summary, this section has investigated the advantages and disadvantages of three different modelling approaches and has highlighted graph theory as the best approach for modelling energy networks. The following section will explain how graph theory can be used to search for cost-optimal energy infrastructures.

2.2 Graph theory

Graph theory finds its origin in the first half of the 18th century. Back then the problem of the Seven Bridges of Köningsberg, a city in Prussia, kept mathematicians busy. The issue at hand was whether one could walk through the city by crossing each of the seven bridges once and only once. When Leonhard Euler proved in 1736 that there was no such walk possible by depicting the two islands and two riverbanks as four nodes and the seven bridges as edges, graph theory was born [20].

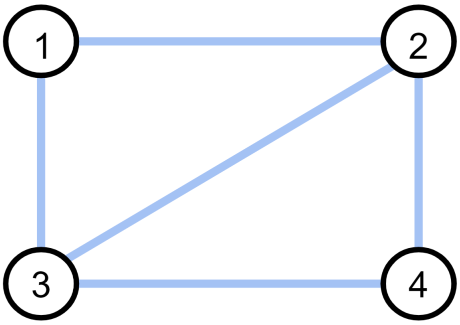


Figure 2.1: A graph consisting of four nodes and five edges

Graph theory is a field in mathematics that studies, not surprisingly, graphs. A graph consists of two sets; a set of nodes or vertices N: $\{n_1, n_2, ..., n_n\}$ and a set of edges E: $\{e_{12}, e_{23}, ..., e_{ij}\}$ that connect these nodes [79]. Figure 2.1 shows a simple graph with the nodes N: $\{n_1, n_2, n_3, n_4\}$ and edges E: $\{e_{12}, e_{13}, e_{23}, e_{24}, e_{34}\}$.

Geometric graph theory is a specialization of graph theory that studies geometric graphs. In geometric graphs, nodes are points in the geometric space, like a 2D-space, and edges are straight lines that connect these points [59]. In other words, geometric graphs are graphs in which the spatial position of its nodes is of essence.

An energy infrastructure network can be depicted as a geometric graph. One of the characteristics of energy infrastructures is that they transport a commodity such as oil, gas or hydrogen from sites of one or several producers to another location of one or several consumers [34]. Geometric graph theory can be used to model these locations with either a supply or demand of the commodity as source or sink nodes, respectively. Figure 2.2 shows an example of several sink and source nodes.

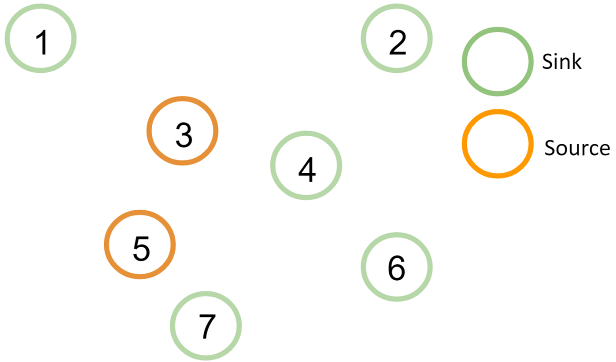


Figure 2.2: Example of several sink and source nodes to be connected

In order to be able to transport the energy commodity from the source or sink nodes, these nodes will need to be connected via edges that represent pipelines or power cables. The development of energy networks is a complex process due to the substantial capital investments required [55]. Since the construction costs of energy infrastructures are greatly dependent on the length of the edges [34], minimizing the total length of the network could significantly reduce the required investment costs.

A tree is a graph that does not contain cycles. This entails that there is only one path from each of the connected nodes to every other connected node. A graph that does not have any cycles and connects all the nodes is classified as a spanning tree [9]. The problem of connecting all the nodes in a weighted graph whose edges represent possible links and whose edge weights represent the edge's length or construction costs while minimizing the graph's total length or costs is a well-known problem in graph theory named the Minimum Spanning Tree (MST) problem [29]. The two most famous algorithms for solving the MST problem were proposed by Kruskal [46] and Prim [60] in 1956 and 1957, respectively. However, the first known algorithm to solve the MST problem stems from the Czech mathematician Otakar Borůvka as far back as 1926. Figure 2.3 shows a minimum spanning tree that connects all the nodes of figure 2.2 where the weights of the graph's edges are based on the edge lengths.

Basing the design of an energy infrastructure solely on the minimal construction costs does not come without trade-offs. Although minimum spanning trees score well on reducing the network's development costs, the networks are highly vulnerable to disruptions. In the example of the MST depicted in figure 2.3 a fallout of edge e_{34}, would have significant consequences for nodes $2, 4$ and 6, who would be completely cut off from their commodity supply.

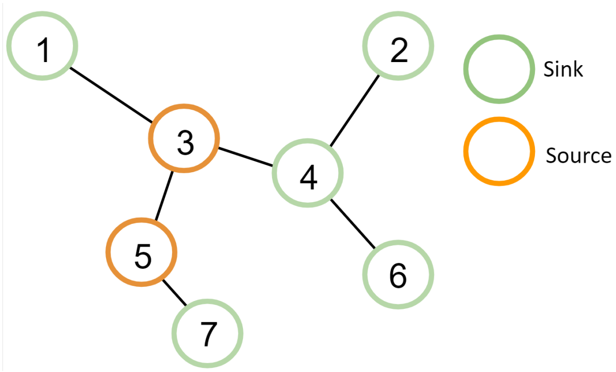


Figure 2.3: A Minimum Spanning Tree based on edge lengths

In reality, the costs of energy infrastructures are not solely based on the length of the pipelines or cables, their capacity plays an important role as well [32, 34, 55]. Melese et al. [55] presented a cost function to determine the costs of energy infrastructures that takes both a pipeline's length and its capacity into account. In their method the costs of constructing an edge e are defined as $C_e = l_e q_e^{\beta}$. Here l_e is the length of edge e and q_e is the capacity of edge e. The exponent β is the cost exponent for the edge's capacity that represents the economies of scale that occur when installing pipelines with a larger capacity for which holds that $0 \leq \beta \leq 1$ [33]. If $\beta = 0$, the edge's capacity has no influence on the edge's cost. When $\beta = 1$ building two pipelines with a capacity of 1 is just as expensive as building one pipeline of a capacity of 2. Generally, when modelling energy networks, β takes on a value of 0.6 indicating that economies of scale play a role when building pipelines with a higher capacity [35]. Equation 2.1 shows the function for determining the total investment costs $C(T)$ of a network T, which is found by taking the sum of the costs of all the network's edges.

$$C(T) = \sum_{\forall e \in E(T)} l_e q_e^{\beta} \tag{2.1}$$

In graph theory, the problem of connecting all the nodes in a weighted graph whose edges represent possible links and whose edge weights represent the construction costs based on the edge's length and the edge's capacity is named the Minimum-Cost Spanning Tree (MCST) problem. Section 3.4 will elaborate on several methods that can be used to solve this MCST problem.

In this section, graph theory was used to conceptualize energy networks as edges representing pipelines or power cables that connect sink or source nodes representing sites with either a demand or a supply for an energy commodity. The costs of these energy infrastructures can be determined via the capacity and the length of these edges.

Before the ultimate goal of incorporating geographical cost-differentiation in the graph theory framework can be reached, it is necessary to understand how geographical areas with varying prices of constructing energy infrastructures in them can be conceptualized. Therefore, the next section will elaborate on a mathematical problem, called the weighted region shortest path problem, that does just that, and will select a method to solve this problem which is suitable to be incorporated in the presented graph theory methodology.

2.3 The weighted region shortest path problem

One of the most studied problems in geometry, robotics and graph theory, is the shortest path problem. An optimal shortest path is the path from a source to a destination node where the (weighted) length of the path is minimized [52]. A variant of this shortest path problem between two points is the problem of determining the shortest path between two points in the plane in the presence of geographical polygonal obstacles. In this problem, the route of the path is limited by obstacles or 'no-go areas' that are considered to be impenetrable [32, 34].

A generalization of the shortest path problem with obstacles is the Weighted Region Shortest Path Problem (WRSPP) which was first described by Mitchell and Papadimitriou in 1991 [56]. In this problem, the polygonal obstacles are not inaccessible but are now polygonal regions which each have an associated weight that specifies the cost per unit distance for travelling in that region. Where in the shortest path problem with obstacles the regions in the plane have weights of either 1 or $+\infty$ depending on whether a region is surpassable or an obstacle, the regions in the weighted region problem can have any weight in the range from 0 to $+\infty$. Appendix A explains why assigning negative weights to weighted regions is problematic. Mitchell and Papadimitriou define the problem as: Given two points s and t and a plane P consisting of a set of faces f and face weights f_c find a path within P from s to t whose weighted Euclidean length is minimal among all paths from s to t.

Since 1991 multiple algorithms have been proposed to solve the weighted region shortest path problem, although most of the proposed algorithms are published by the same groups of authors. De Carufel et al. [15] have shown that an optimal solution to the problem cannot be computed exactly. Consequently, all examined algorithms are approximation algorithms that are likely to produce solutions with an absolute or relative error compared to the optimal solution. The algorithms found in the literature have been compared based on their accuracy (the size of their maximum absolute or relative error), computational complexity and difficulty to implement. An overview of the examined algorithms can be found in table 2.1.

Table 2.1: Examined Minimum Weighted Region Shortest Path Problem Algorithms

Year	**Author(s)**	**Type of Algorithm**	**Weighted Regions**
1991	Mitchell & Papadimitriou [56]	Continuous	Convex faces
1997	Mata & Mitchell [54]	Discrete	Convex faces
1997	Lanthier et al. [47]	Discrete	Triangulated faces
1998	Aleksandrov et al. [2]	Discrete	Triangulated faces
2000	Reif & Sun [63]	Discrete	Triangulated faces
2003	Aleksandrov et al. [3]	Discrete	Triangulated faces
2004	Roy et al. [69]	Discrete	Triangulated faces

Mitchell & Papadimitriou are the only authors that propose a continuous algorithm [56]. This entails that the solution space of their algorithm is way larger than that of the other algorithms, since it is not limited to a discrete amount of options like that of the other algorithms [40]. Consequently, the algorithm might be better able to approximate the optimal solution. However, this larger solution space is likely to steeply increase the algorithm's computational complexity.

The examined algorithms make different assumptions on how to model the weighted regions. The algorithms of Mitchell & Papadimitriou [56] and Mata & Mitchell [54] both require that the weighted polygons are convex. This is not a generalization of the problem, since simple polygons can be decomposed into convex sub-polygons [56], for example with the diagonal decomposing method proposed by Fernández et al. [25]. All the other algorithms assume that the weighted regions consist of triangulated faces. This generalization of the weighted regions does not have to be a problem either, since convex polygons can be triangulated into triangulated sub-polygons, for instance with Delaunay triangulation [17]. Section 3.1 elaborates on the differences between convex, concave and triangulated polygons.

The algorithm of Mitchell & Papadimitriou uses a continuous Dijkstra's method that exploits the fact that light obeys Snell's law of refraction and seeks the path of minimum time. Therefore, they argue, that the shortest path through a region with a uniform 'travelling weight' should also follow Snell's law. The computational complexity of their algorithm is O($n^8 M$), where n is the number of face vertices and M is a function of various input variables including the number of faces and a relative error tolerance ε [56].

Mata & Mitchell have presented an algorithm that discretizes the problem by constructing a pathnet graph which can be searched with Dijkstra's algorithm to obtain an approximate shortest path. The algorithm's time complexity is O($n^3 M$), where n is the number of face vertices and M is a function of multiple problem characteristics including the minimum and maximum weight of the faces and the relative error tolerance ε [54].

Both the algorithms of Lanthier et al. and Aleksandrov et al. (1998) discretize the weighted region problem by conceptualizing the regions as triangles with three vertex points after which an **m** number of boundary nodes are placed along the edges of the triangulated regions before Dijkstra's algorithm is used to search for the optimal path. Lanthier et al. [47] place the boundary points uniformly along the edges of the polygons, while Aleksandrov et al. (1998) [2] use a logarithmic distribution for the placement of the points. Both algorithms have a ε relative error. Additionally, the algorithm adopted by Lanthier et al. has an absolute error of LW_{max}-absolute error that results from the discretization where L is the length of the longest edge in the graph and W_{max} is the maximum face weight. The computational complexity of the algorithm of Lanthier et al. is $O(\frac{n^3}{\varepsilon} + n^3 logn)$ and that of Aleksandrov et al. (1998) is $O(nmlog(nm)$. For both algorithms, n denotes the number of face vertices. For the second algorithm, m denotes a logarithmic function with multiple input parameters including the length of the longest edge and the minimum angle between two edges.

Aleksandrov et al. (2003) [3] introduce another method of discretizing the weighted region problem by placing extra points in the interior of the weighted faces instead of on the faces' edges. Because this method reduces the number of added points, the size of the discretization is smaller and the computation complexity can be reduced to $O(\frac{n}{\sqrt{\varepsilon}} log\frac{n}{\varepsilon} log\frac{1}{\varepsilon})$ where n is the number of face vertices and ε is the relative error.

Roy et al. [69] propose an algorithm that places boundary points on the edges of the triangular faces as well. A first boundary point is placed in the exact middle of each edge. Additional boundary points are placed in the middle of the previous center boundary point and the triangle's vertex until a user-defined minimum range ε near the vertex is reached. The algorithm has a computational complexity of $O(nmlog(nm))$ where n is the number of triangular faces and m is the number of Steiner points on the longest edge. The approximation bound is $(1 + \frac{1}{sin(\theta)})$ where θ is the minimum angle incident at the vertices of all triangular faces.

Reif & Sun [63] do not propose a new form of discretization (like the uniform or logarithmic algorithms discretizations described above), but adopt a "wavefront-like" algorithm that can compute optimal paths on these discretizations more efficiently than the shortest path finding Dijkstra's method. Implementing this algorithm in the previously described algorithms of Aleksandrov et al. (1998) and Lanthier et al. could respectively match the computational complexity and relative error or reduce the time complexity from O($\frac{n^3}{\varepsilon} + n^3 logn$) to O($n^3 logn$) and improve the algorithm's accuracy: the relative error ϵ is maintained, but the absolute error LW has now been eliminated.

Table 2.2 evaluates the algorithms described above on their accuracy, computational complexity and difficulty to implement. The lighter the colour of the table's cell, the better the algorithm scores on the criterion. Due to the inexperience of the author with implementing complex algorithms, algorithms that are regarded as hard to implement, are deemed not feasible for the scope of this research.

Table 2.2: Evaluated Minimum Weighted Region Shortest Path Problem Algorithms

Study	**Accuracy**	**Computational complexity**	**Implementation difficulty**
Mitchell & Papadimitriou (1991)	ε-relative error	O(n^8M)	Hard
Mata & Mitchell (1997)	ε-relative error	O(n^3M)	Hard
Lanthier et al. (1997)	ε-relative error LW-absolute error	O($\frac{n^3}{\varepsilon} + n^3 logn$)	Easy
Aleksandrov et al. (1998)	ε-relative error	O($nmlog(nm)$)	Medium
Reif & Sun (2000)	ε-relative error	O($\frac{n^3}{\varepsilon} + n^3 logn$)	Hard
Aleksandrov et al. (2003)	ε-relative error	O($\frac{n}{\sqrt{\varepsilon}} log\frac{n}{\varepsilon} log\frac{1}{\varepsilon}$)	Hard
Roy et al. (2004)	$\frac{1}{sin(\theta)}$	O($nmlog(nm)$)	Medium

This leaves the algorithms of Aleksandrov et al. (1998) and Roy et al. and Lanthier et al. [2, 69, 47]. The computational complexity of the first algorithm and the accuracy of the second algorithm are dependent on the minimum angle among the triangular faces. The performance of both algorithms declines a great deal if this minimum angle is rather small. Therefore the algorithms' performance will decrease if even one of the triangulated weighted regions is skinny, entailing that the triangle's height is far greater than its width [81]. Since the algorithm will be used as part of a cost-optimization model for the design of networked energy infrastructures, the weighted regions in practice are likely to have all sorts of shapes; including concave polygons that will be triangulated into a number of triangles, at least one of which is likely to be a thin triangle. Because of the implications that the purpose of the model will bring, the algorithms of Aleksandrov et al. (1998) and Roy et al. are not suitable for this research. As a result, the last standing algorithm of Lanthier et al. will be implemented. This algorithm will be further elaborated on in section 3.2.

The last two sections have elaborated on modelling cost-optimal energy infrastructures and regions with different traversing costs independently. The upcoming section will describe how these two distinct methods can be combined to incorporate geographical cost-differentiations in energy network modelling.

2.4 Steiner trees in a graph

Section 2.2 introduced locations with a demand or a supply of an energy commodity as sink and source nodes, respectively. Hereafter, section 2.3 conceptualized weighted regions as triangles with vertex points and edges on which additional boundary points are placed. When combining these conceptualizations to be able to design cost-optimal energy infrastructures in a geographical context with weighted regions, the designer is basically searching for a cost-optimal tree that connects all the source and sink nodes and potentially uses the weighted obstacle nodes.

In graph theory, this problem is known as the Steiner Minimum Tree (SMT) problem. Different from the Minimum Spanning Tree problem (MST) introduced in section 2.2, the SMT does not require all the nodes of the graph to be spanned, but only some of the graph's nodes [83]. In a SMT, the nodes that are mandatory to be connected are known as terminals, whereas the optional nodes are called Steiner nodes [26]. When applied to energy infrastructures in a weighted obstacle context, the source and sink nodes form the mandatory terminals nodes and the vertices and boundary nodes of the triangulated faces the optional Steiner nodes.

Figure 2.4 shows an example of a simple Steiner Minimum Tree problem. The dashed lines illustrate the potential connections with their costs that the graph can exist out of. Just like the MST shown in figure 2.3, the goal of the SMT problem is to minimize the total costs of the graph's edges. Note that all the mandatory terminals, the sink and source nodes, are connected, while only some of the Steiner nodes are used in the graph. Two well-known heuristics for solving the SMT problem are the Minimum Path Heuristic

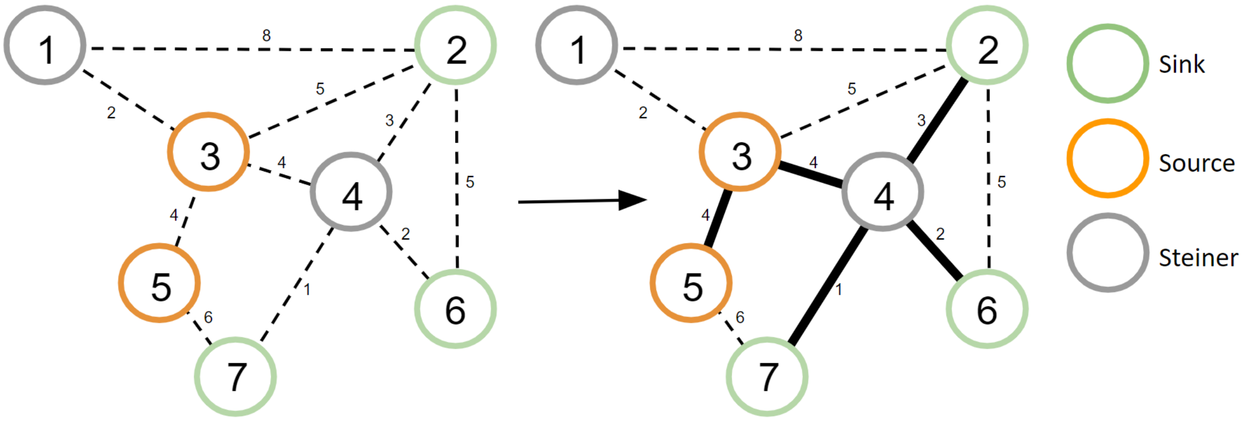


Figure 2.4: A Steiner Minimum Tree problem

(MPH) and the Distance Network Heuristic (DNH) [83], which will be further elaborated on in section 3.4.

This section has laid out the methodological groundwork for combining the modelling of cost-optimal energy infrastructures and weighted regions by conceptualizing them as source and sink nodes, and Steiner nodes respectively. The next section will expand on this conceptualization by presenting a formalization of the mathematical problem that allows for the incorporation of cost-differentiated obstacles in energy infrastructure modelling, called the Minimum Cost Weighted Region Steiner Tree problem.

2.5 The minimum cost weighted region steiner tree

Sections 2.1 and 2.2 have demonstrated that graph theory is a suitable tool for energy network cost minimization modelling due to its numerous well-known algorithms which enable the researcher to explore a wide variety of energy network design factors. In section 2.3 the algorithm of Lanthier et al. [47] that discretizes weighted regions by placing boundary nodes uniformly along the faces of weighted regions has been selected to model geographical regions in which the cost to traverse these regions can be differentiated. Combining the described graph theory methods, these discretized weighted regions and the Steiner tree in a graph problem described in section 2.4, allows the modeller to search for a cost-optimal multi-sink multi-source capacitated network which edges are able to cross regions with various traversing costs.

To the knowledge of the author of this report, there does not yet exist such a model or similar problem discussed in the academic literature. Therefore for the remainder of this report, this problem will be referred to as the Minimum Cost Weighted Region Steiner tree (MCWRST) problem, which will be defined as: Given a set of nodes N that are either a source with a supply or a sink with a demand d_i and a set of faces F which each have a face weight f_c, find a minimum-cost network topology that connects all sources and sinks with sufficient network capacity while guaranteeing that all demand of the sinks can be satisfied by the supply sources while optionally connecting the Steiner nodes formed by the vertex and boundary nodes of the triangulated faces. The methodology that underlies the model developed to solve this problem will be described in chapter 3.

3. Methodology

The previous chapter has formalized the Minimum Cost Weighted Region Steiner Tree (MCWRST). A modelling problem that searches for a cost-optimal network that spans all the supply and demand sites, or terminals, in a geographical context with regions with different costs of traversing through. This chapter describes the methodology that underlies the model developed in Python used to find this Minimum Cost Weighted Region Steiner Tree. Before the algorithms described in sections 3.4 and 3.5 can search for this MCWRST, a graph will need to be constructed that contains all the optional paths (and their costs) between the terminals which may or may not surpass a weighted region.

To achieve this graph, section 3.1 will prepare weighted regions of different shapes to be used as input in the model by transforming them into triangulated regions. Hereafter, section 3.2 will discretize all the potential paths through these weighted triangular regions by conceptualizing them as nodes that are connected with edges. To complete the graph, all the terminals are connected to each other and the weighted region based on a 'visibility' algorithm in section 3.3.

3.1 Preparing the weighted regions for model input via triangulation

Since the main contribution of this research is to model energy demand or supply sites that are to be connected through an energy infrastructure in a context of geographical areas with different costs for power cables or pipelines to 'pass through', being able to use the model on an as wide range of regions as possible, is highly desirable. As discussed in section 2.3 the method of Lanthier et al. [47] has been selected to model these regions. One of the inputs required by their algorithm is a set F of triangulated polygons f each with its own face weight f_c and a number of additional Steiner points m that are to be placed on the faces outskirts. To not exclusively limit the application of the model to weighted regions shaped like triangles; regions shaped in different, more complex forms will have to be prepared before they can be inserted into the model. Therefore this section will outline the model's first step, which uses geometrical techniques to transform weighted regions of various shapes and sizes into usable triangulated regions.

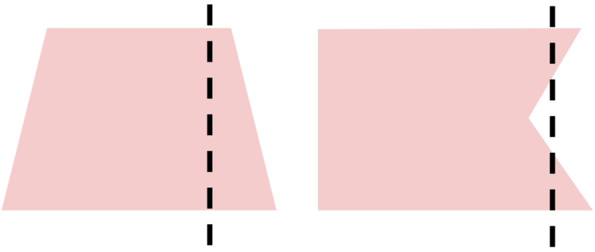


Figure 3.1: A Convex (l) and Concave (r) Polygon

To enlarge the pool of usable shapes regions can take on, regions that are concave polygons can be split into convex polygons [25], which in turn can be triangulated into triangle-shaped regions [17]. Polygons are figures in a 2D-plane that consist of a finite number of connected straight line segments that form a closed chain. Convex polygons are polygons of which all interior angles have a degree smaller than 180°. A polygon is concave if only one of its interior angles is larger than 180°. As a result, any line can be drawn through a convex polygon while meeting the polygon's line segments only twice, while this is not the case for concave polygons [10]. Figure 3.1 shows an example of a convex and a concave polygon. Note that the line drawn through the convex polygon crosses its boundaries only twice, while the line drawn through the concave polygon crosses the boundaries a total of four times.

Convex decomposition is the process of splitting up a single concave polygon into several convex polygons [19, 25]. Férnandez et al. [24] wrote a computationally quick algorithm for decomposing concave polygons into convex polygons. The algorithm allows for the decomposition of concave polygons with and without holes. However, the concave polygons must be simple, which entails that none of the polygon's straight line segments are allowed to intersect. Figure 3.2 shows an example of two simple polygons, with or without a hole and a non-simple polygon. Note that two of the straight line segments of the non-simple polygon, depicted on the right in the image, intersect each other in the middle.

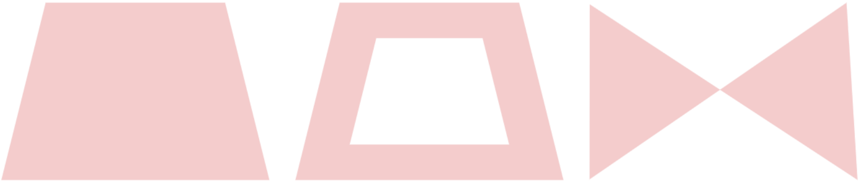


Figure 3.2: A Simple Polygon (l), A Simple Polygon with a Hole (m) and Non-simple Polygon (r)

The decompositions found by the algorithm proposed by Férnandez et al. [24] usually result in a number of convex pieces close to the optimal solution. This entails, that for complex concave polygons that, if optimally decomposed, consist of an X number of convex pieces, the algorithm usually finds a decomposition that also has an X number of convex pieces as the optimal solution or a total number of convex pieces that is not much higher than X.

The algorithm is based on decomposing with the use of diagonals. As a consequence, the endpoints of the partition edges must be vertices of the concave polygon. The algorithm operates via a "divide and conquer" tactic. It starts out with the whole concave polygon to be composed. Then given an initial vertex, a convex polygon of the partition is generated and is cut off from the initial polygon. This process is repeated with the remainder of the polygon until it is no longer concave.

A detailed step-for-step description of this procedure is given in appendix B.1. Figure 3.3 shows the application of the algorithm applied to the concave polygon that was introduced in figure 3.1. The result is a convex decomposition consisting of two convex polygons.

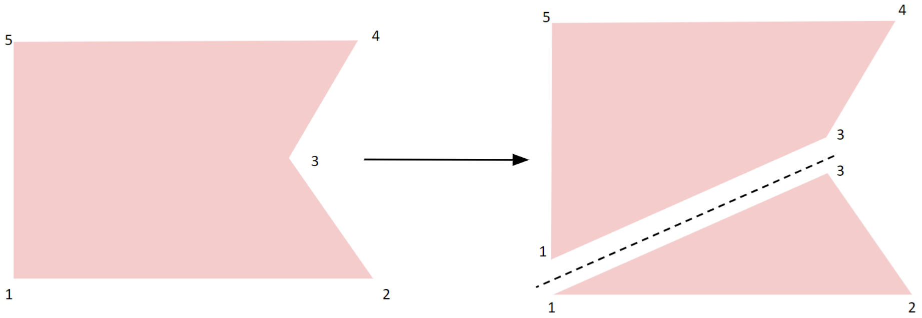


Figure 3.3: Convex decomposition of a concave polygon

The last step before being able to use the regions in the Weighted Region Shortest Path algorithm of Lanthier et al. [47] is to triangulate the now convex polygons. One of the most famous methods to achieve this is called the Delaunay Triangulation. A Delaunay triangulation is a set of triangles made from a set of vertices so that no vertex lies inside the circumcircle of any triangle in the set [17]. A circumcircle of a triangle is a circle passing through all the triangle's vertices. The algorithm used in the model to compute the Delaunay triangulation of the convex polygons is known as the Bowyer-Watson algorithm [62], which, together with a figure showing a circumcircle of a triangle, is illustrated more profoundly in appendix B.2. Figure 3.4 shows the Delaunay triangulation of the convex polygon that was the result of the convex decomposition in figure 3.3.

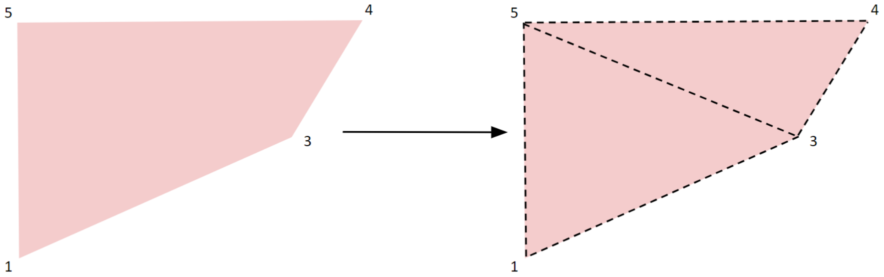


Figure 3.4: The Delaunay triangulation of a Convex Polygon

Concave polygons can undergo a Delaunay triangulation as well. However, these triangulations do not yield the desired result, since additional triangles that were not part of the original polygon are added during the triangulation process. Figure 3.5 demonstrates why concave polygons need to be decomposed into convex polygons before they can be triangulated by displaying the Delaunay triangulation of the concave polygon presented in figure 3.1. After the triangulation, the polygon itself is successfully cut up into three triangles $\Delta 123, \Delta 135$ and $\Delta 345$. Unfortunately, the Delaunay triangulation results in an additional unwanted triangle $\Delta 234$ that was not part of the original polygon.

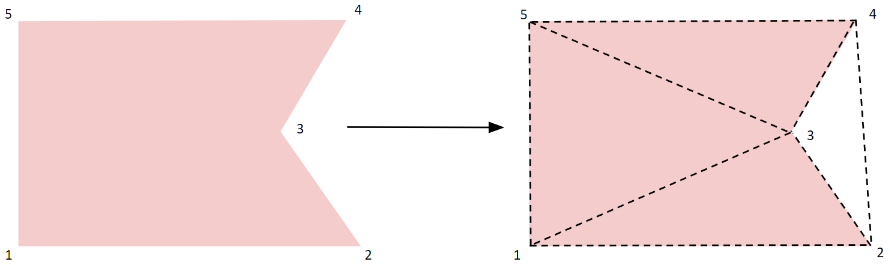


Figure 3.5: The Delaunay triangulation of a Concave Polygon

This section has described the methods to extend the possible shapes of regions that can be used as input in the Minimum Cost Weighted Region Steiner Tree problem. With the use of these decomposition and triangulation methods, the model can now use regions shaped as concave polygons, convex polygons and triangles (which are concave polygons as well) as input, instead of being constrained by only allowing triangle-shaped regions as input. The following section will elaborate on how these triangulated regions are modelled as weighted regions that contain different potential routes for pipelines or power cables to traverse these regions.

3.2 Creating subgraphs from the triangulated regions

After the input polygons have been triangulated, they can be modelled as weighted regions with the use of the method of Lanthier et al. [47] that was selected in section 2.3. The first step of their method consists of placing an m amount of Steiner points evenly distributed on the edges of the triangulated polygons. The larger the amount of Steiner nodes m that are placed on the edges of the weighted faces f, the more precise the shortest path through the regions can be obtained. The downside of increasing m is that the computational time of the algorithm will increase. In the second step the nodes of the same triangle are connected if a) the nodes lie on different edges of the same region f or b) the nodes lie adjacent on the same edge of region f. Figure 3.6 shows an example of these two steps for two triangular regions with an m of 1 and 2.

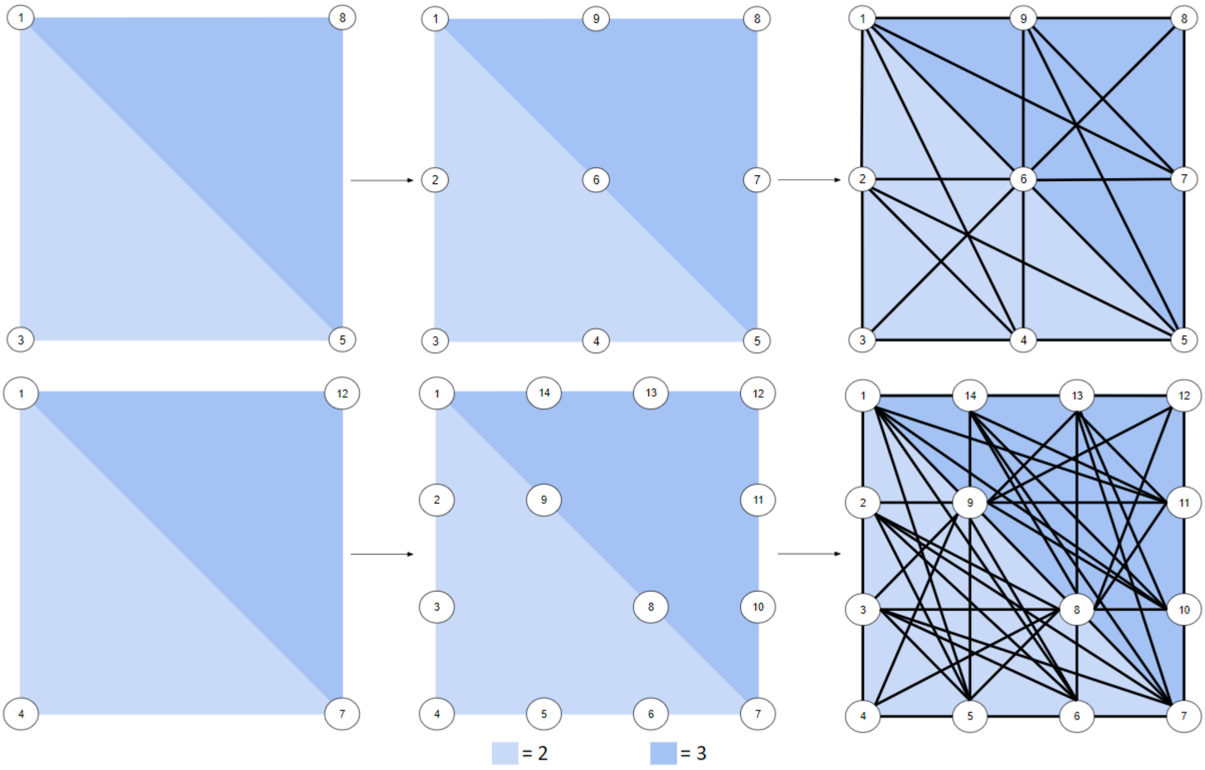


Figure 3.6: Adding $m = 1$ (top) and $m = 2$ (bottom) Steiner points to triangulated regions and connecting the nodes

Note that the amount of edges greatly increases if 2 Steiner points are placed on the faces of the triangles instead of 1. These extra edges are likely to decrease the total costs of the found network, because the pipelines or power cables have a lot more potential routes to traverse through the weighted regions. For example, there are far more potential routes to reach node 12 from node 4 in the bottom example than there are potential routes to reach node 8 from node 3 in the bottom example. As stated above, the downside of these extra Steiner points and edges is that the run time of the algorithms described in the rest of this chapter will increase.

If the edges lie on different sides of the same triangulated region f, their costs calculated as is shown in equation 3.1 in which l_e is their length and f_{c_e} the weight of the region that they are inside in. In this case, the region with the lighter colour blue has a face weight f_c of 2 and the region with the darker colour blue has a face weight f_c of 3. Since the edge e_{29} is located in the region with the lighter colour blue in the bottom example of figure 3.6, it has a face weight f_c of 2.

$$C_e = l_e f_{c_e} \tag{3.1}$$

If the edges are part of two or more weighted regions their costs are also calculated with the use of equation 3.1. For these cases, the face weight f_c is equal to the minimum weight of the weighted regions that the edge is a part of. In the top example of 3.6 edges e_{16} and e_{56} are both part of two triangles. Therefore, the edge costs of e_{16} and e_{15} will be calculated by using the weight of the cheapest, lighter color blue region as $C_e = l_e * 2$.

$$C_e = l_e \tag{3.2}$$

If the edges lie on the outskirt of only one weighted region, their weight is determined solely based on their length l_e by using the equation 3.2. However, there is one exception to this rule. The cost of edges that are on the brink of the model run's area are also determined by the formula of equation 3.1. The area is defined as the smallest rectangle that contains all the terminals that need to be connected, the vertices of the triangulated regions and the Steiner nodes placed on the regions' edges. The four points of the area can be found with the use of the minimum X- and Y-coordinates of all of these nodes as is shown in equation 3.3.

$$Area = [(x_{min}, y_{min}), (x_{max}, y_{min}), (x_{max}, y_{max}), (x_{max}, y_{min})] \tag{3.3}$$

The dashed line in figure 3.7 illustrates an example of such a smallest rectangular that contains all the terminals to be connected, the vertices of the triangulated regions and the Steiner nodes placed on the regions' edges. Because edges e_{34}, e_{45}, e_{19}, e_{98} all lie on the brink of this model run's area, their costs are based on equation 3.1. On the contrary, the edges e_{12}, e_{23}, e_{57}, e_{78} are not on the brink of the area, so their weight solely depends on their lengths as in equation 3.1.

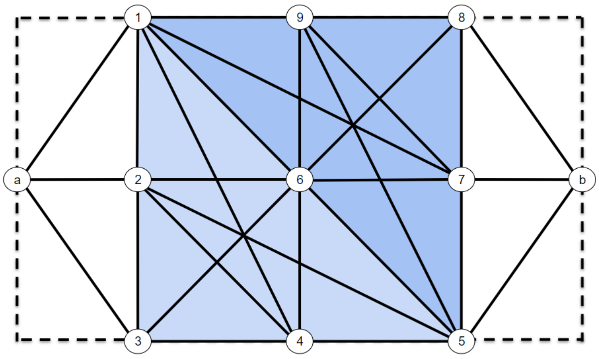


Figure 3.7: The area of an example with two terminals and two triangulated faces

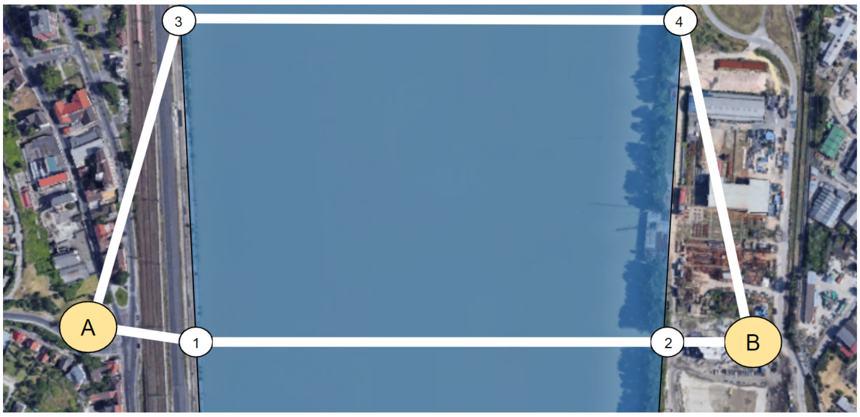


Figure 3.8: Two terminals on opposite river banks [27]

Figure 3.8 shows the rationalization behind this modelling choice. In the image two terminals A and B are depicted, each on their own side of a river. The white points represent Steiner points which make up two potential routes for connecting the terminals while crossing the blue weighted region which is representing a river. Note that the locations of these Steiner nodes are chosen arbitrarily for the sake of the argument and thus not based on the algorithm of Lanthier et al. described above.

If the costs of edges on the brink of the model's area would solely be based on their length l_e as in equation 3.2; depending on the costs of building a pipeline through the river region with face weight f_{c_e}, the path P consisting out of the edges $[e_{A1}, e_{12}, e_{2B}]$ might be more expensive than path P consisting of edges $[e_{A3}, e_{34}, e_{4B}]$, while the last path travels through longer distances, both over land and through the river. Therefore the costs of edge e_{34} are determined by equation 3.1 with both its length l_e together with the weight of the weighted region f_{c_e} it is on the brink of, instead of solely on the edge's length l_e. Because in reality, the river does not discontinue at either the end of the image or model run.

This section has made explicit how the triangulated obstacles are modelled as subgraphs with m Steiner points placed on the outskirts of the triangles that are connected with edges. In addition, the section has explained which edges are given a weight based dependent on solely their length and which edges are assigned a weight based on their length and on which region's face weight they are a part of. The following section will describe how the terminals, or sites with either a demand or supply for an energy commodity, are to be connected to these subgraphs, which will result in an underlying graph in which can be searched for the Minimum Cost Weighted Region Steiner Tree.

3.3 Connecting the terminals with the triangulated regions

After step 2 of the model that was described in the previous section, all the weighted regions have been transformed into regions' vertex nodes, nodes placed on the faces of the regions and edges that connect these nodes. This section outlines step 3 of the model that finalizes the graph by connecting the terminals to weighted regions and other terminals 'in view'. Since edges that pass through weighted regions introduce extra costs based on the region's weight f_{c_e}, while the costs of edges that do not traverse through weighted regions are solely based on their length l_e, it is necessary to make a distinction between these edges. As a result, an edge that connects a terminal to a weighted region or another terminal, may only either be located inside or outside a region. Therefore, it is necessary that terminals that are located outside a region are only connected with edges that do not cross a weighted region, while terminals that are located inside regions should only be connected to nodes that belong to that triangulated region or other terminal nodes situated in that weighted region.

These separate goals can both be achieved by constructing a visibility graph. De Berg et al. [14] define the visibility graph problem as follows: Given a set S of disjoint polygonal obstacles, we denote the visibility graph $G_{vis}(S)$ which nodes are the vertices of S for which there is an edge between vertices v and w, if they can 'see' each other. That is, if the segment $\bar{vw}$ does not intersect the interior of any obstacles in S.

A fast algorithm that finds the solution to the visibility graph problem and is not that difficult to implement is Lee's Visibility Graph Algorithm [64]. The algorithm runs through all the vertices v of graph G and looks for all the visible vertices of that node by using a scan line and keeping track of two lists of 'visible nodes' and 'visible edges' [13]. Both of these concepts will be further explained with the use of the example shown in figure 3.9. In the figure the terminal nodes are shown in yellow and the vertex and Steiner points of the weighted regions in red. The depicted example portrays how the algorithm searches for the nodes visible to node 1. The algorithm initializes with the scan line parallel to the x-axis beginning from point 1. The scan line is a half line, which entails that it is a line that extends indefinitely in a single direction [42]. By moving in a counter-clockwise circle, the scan line will visit all the other points one by one.

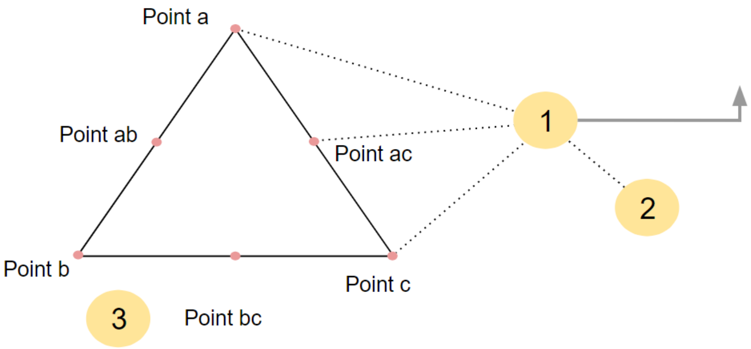


Figure 3.9: Application of Lee's visibility algorithm

Once the scan line hits point a, it adds the point to the 'visible nodes' list. The scan line also encounters two edges, e_{ab} and e_{ac} and checks if these are on the counter-clockwise side of the scan line [64]. Since both the edges are on the counter-clockwise side, they are added to the 'open edges' list. Once the scan line encounters point ab it checks if the scan line intersects with one of the 'open edges' to reach the point. Unfortunately for point ab, the open edge e_ac is intersected. Therefore, the point is not added to the 'visible nodes' list. In point ac the edge list is still not empty. However, the scan line does not need to cross an open edge to reach point ac. Hence, the point is added to the 'visible nodes' list. When the scan line visits point b, it does not add the point. Moreover, it also removes the edge e_{ab} from the 'open edges' list, since it will not cross the edge again while making its counter-clockwise circle, and adds edge e_{ac} to the 'open edges' list. This procedure continues until the scan line is back where it began, parallel to the x-axis. During the process, terminal 1 has seen vertex points a and c, Steiner point ac and terminal 2.

After the visible vertices have been identified for each node, the node is connected to these vertices with an edge. The costs of edges that connect nodes outside of weighted regions are equal to the edge length as in equation 3.2. Whereas the cost of edges that connect nodes inside of weighted regions are dependent on both the edge length and the weight of the region like in equation 3.1.

By connecting the terminals to each other and the weighted regions, the graph that contains all the terminal nodes and the triangulated region vertex and Steiner nodes has now been finalized. In the next section, the algorithms that use this graph to search for the Minimum Cost Weighted Region Steiner Tree will be discussed.

3.4 Determining the steiner minimum tree

This section will use the finalized graph constructed in the previous section to search for the Steiner Minimum Tree (SMT). To solve this SMT problem, two well-known heuristics called the Minimum Path Heuristic (MPH) and the Distance Network Heuristic (DNH) will be proposed to solve this SMT problem [83]. As described in section 2.4, the Steiner Minimum Tree is a tree that spans all the mandatory nodes called Terminals, while possibly making use of optional Steiner nodes. In the developed model, the sink and source nodes form the terminals, whereas the vertex and boundary nodes of the triangulated regions are the optional Steiner nodes.

The Minimum Path Heuristic proposed by Takahashi and Matsuyama [75], is a rather simple one. It starts by selecting an arbitrary terminal and finding the terminal closest to it. This terminal is connected via the shortest weighted path between the two terminals. Hereafter the closest terminal from all the (both Terminal and Steiner) nodes that are already part of the tree is connected. This process is repeated until all the terminals are connected.

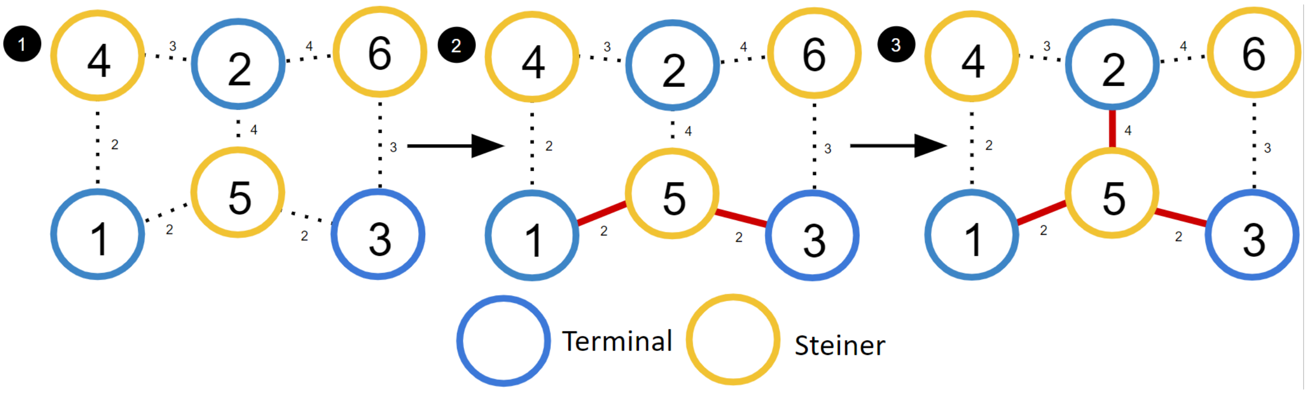


Figure 3.10: Minimum path heuristic

Figure 3.10 shows an example of the Minimum Path Heuristic. The search process starts from an arbitrarily chosen terminal, in this case node 1. Node 3 is the terminal nearest to node 1 so it is connected via the shortest path $[e_{15}, e_{53}]$. The final terminal to be added to the tree is node 2 via the shortest path $[e_{52}]$, making the sum of all the graph's weighted edges 8.

The Distance Network Heuristic as formulated by Kou et al. [45] is a bit more complex. The heuristic is initialised by constructing a graph G with both Terminal and Steiner nodes. It starts off by calculating the distances of the shortest paths between all the Terminal nodes to construct a distance graph G_1 that only contains the Terminal nodes that are connected via edges whose weights represent the length of their shortest paths and leaves out all the Steiner nodes. Subsequently, the Minimal Spanning Tree T_1 of this distance graph G_1 is found by using either Kruskal's [46] or Prim's [60] algorithm. Subsequently, a subgraph G_S of G is constructed in which all the edges of the MST T_1 are replaced by their corresponding shortest paths in G. Hereafter the MST T_S of G_S is found. Finally, a MST T_H is constructed from T_S by removing, if necessary, nodes until all the graph's leaves are Terminal nodes.

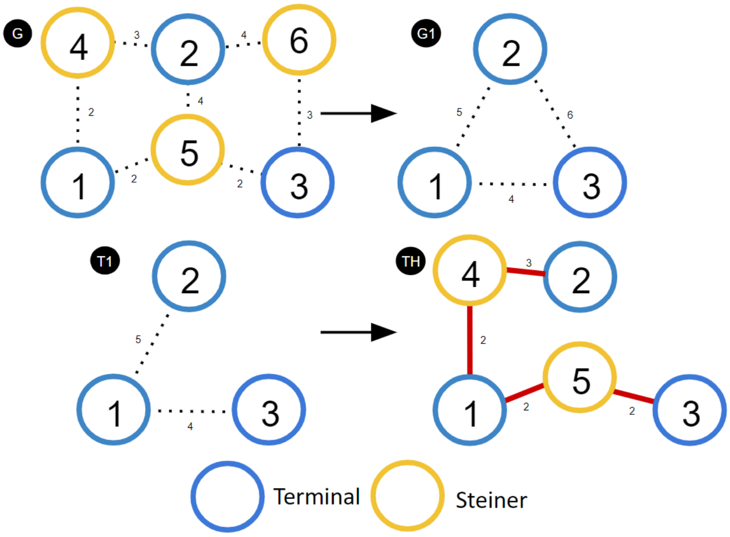


Figure 3.11: Distance network heuristic

Figure 3.11 shows an example of the DNH procedure by using the same initial graph as used in figure 3.10. The graphs G_1 and T_1 depict the distance graph of G and its MST, respectively. Graph T_H shows the final graph that is constructed by replacing the edges of graph T_1 with their corresponding paths of G. Note that T_H was immediately a MST without Steiner leaves. Thus, the last two steps of the DNH algorithm needed not to be performed. The total costs of the found network are 9, 1 more than the total costs of the network found by the MST algorithm.

Although the MPH algorithm outperforms the DNH algorithm in most cases, the DNH produces cheaper Steiner Minimum Trees on occasions [26]. Since both algorithms are computationally fast and not extremely difficult to implement, both the MPH and the DNH heuristics have been implemented in the model.

This section has described two heuristics for solving the Steiner Minimum Tree problem. When applied to energy infrastructure modelling in a geographical cost-differentiated content, a solution to this problem entails that a network has been found that minimizes the total traversing costs of the pipelines or power cables, that connects all the sites with either a source or a demand for an energy commodity, while possibly crossing regions that have a higher cost of traversing through. The following section will introduce a new cost factor in the form of assigning capacities to these pipelines or power cables and will introduce an algorithm that will try to minimize the total network costs.

3.5 Determining the minimum cost weighted region steiner tree

This section will build upon the Steiner Minimum Tree found in the previous section and introduce a new cost factor associated with the capacities which will be assigned to the pipelines and power cables of the energy network required to ensure that all the sink nodes are supplied with enough energy to satisfy their demands. Subsequently, different heuristics will be presented that are able to find improvements to the discovered Steiner Minimum Tree, from which one will be selected to search for the Minimum Cost Weighted Region Steiner Tree.

As shown in equation 2.1 in section 2.2, the costs of energy infrastructures' power cables or pipelines are depended on both their lengths (l_e), capacities (q_e) and a cost exponent (β) which represents the economies of scale that occur when installing pipeline or power cables with a larger capacity. In this research, another cost factor has been introduced in the form of the (potential) face weight f_{c_e} that a power cable or pipeline might traverse through. If a power cable or pipeline does not cross a weighted region f_{c_e} is equal to 1. Combining these different costs results in the cost function of equation 3.4.

$$C(T) = \sum_{\forall e \in E(T)} l_e f_{c_e} q_e^{\beta} \tag{3.4}$$

Before equation 3.4 can be used to calculate the total costs of the discovered Steiner Minimum Tree network, the network's edges will need to be assigned enough capacity to satisfy all the demand and supply constraints imposed by the source and sink nodes. Heijnen et al. [33] describe an algorithm for assigning capacity to a tree network that guarantees that the demand of all the sinks can be delivered by the supply of the source nodes. One additional requirement of this algorithm is that the total supply of all the source nodes is equal to the total demand of the sink nodes.

The algorithm initializes by assigning a capacity equal to the supply or demand of a leaf to the edge that connects that leaf to the remainder of the graph. In graph theory, leaves are nodes with a degree of one, which entails that they are only connected to one other node. Simultaneously, the demand or supply of the leaf is added or subtracted from the leaf's neighbouring node's demand or supply. Note that the supply of the source nodes is given as a negative number, while the demand of the sink nodes is given as a positive number. Hereafter, the leaf node and the edge connecting the leaf are removed. This process is repeated until all the edges have been assigned a capacity [31]. Figure 3.12 presents an example of the capacity assignment procedure using the found Minimum Spanning Tree of figure 2.3. The supply or demand of each node is depicted as a blue number. The edges that still need to be assigned capacity are presented as black dashed lines. Once edges have been assigned a capacity, they and their capacity are shown in red, where their thickness is an indicator of the edge's capacity.

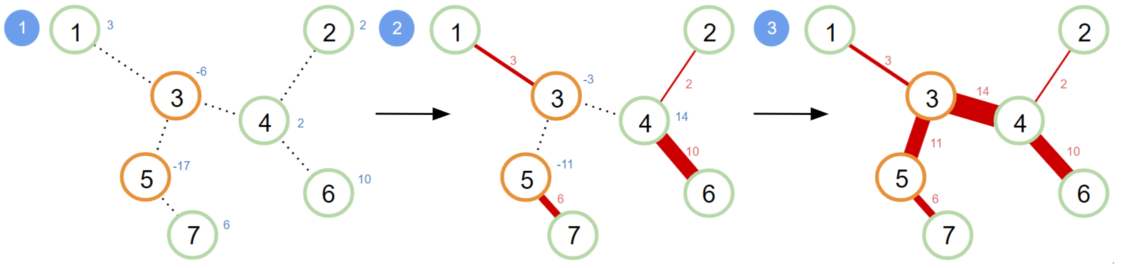


Figure 3.12: The capacity assignment procedure

If the weights of a graph's edges are only based on the length of the graph, the minimum length spanning tree is equal to the minimum cost spanning tree (MCST). However, as we have seen in equation 3.4, the costs of pipelines or power cables are not only influenced by their lengths, but by their capacity and potentially by the region that they traverse through as well. Therefore, assigning capacities to a MST, as we have done in figure 3.12, will not always result in the Minimum Cost Weighted Region Steiner Tree.

Searching for a multi-source, multi-sink network that minimizes the cost function of equation 3.4 is complicated because the weight of an edge e is dependent on the needed capacity of the edge q_e, which itself is again subject to the network's layout. This interdependency creates a huge non-linear optimization space, where switching only one edge in the network's topology could alter all the assigned capacities to the network's edges, thereby significantly changing the total costs of the network. Subsequently, local minima are likely to be found. Therefore, finding the Minimum Cost Weighted Region Steiner Tree requires iterating over all trees before attributing the edge capacities after which the network's costs can be determined [85].

A first heuristic for finding a minimum cost capacitated spanning tree by starting from a MST was proposed by Rothfarb et al. [68]. Their procedure, called the Delta Change heuristic, searches for potentially cheaper networks by connecting two nodes in the tree that were not yet connected via an edge, thereby creating a cycle. Hereafter, one of the pre-existing edges of the cycle is removed, before reassigning the network's capacities and calculating its costs. The algorithm restarts when a cheaper solution is found and ends when this is no longer the case. In their Local Search algorithm, Brimberg et al. [10] altered the Delta Change heuristic by removing, one by one, all the pre-existing edges of the cycle and saving the cheapest solution, if any.

Heijnen et al. [33] proposed a different algorithm for finding a MCST based on edge turns. Every turn an edge is removed, after which the other nodes of the two disconnected parts of the remaining tree are connected one by one to one of the initial nodes from the part of the tree that is disconnected. This reduces the number of possible recombinations compared to the Delta Change or the Local Search heuristics since the recombination must occur at one of the nodes of the removed edge.

Yeates et al. [85] add an additional algorithmic step in the process of searching for a MCST. There High Valency Shuffle algorithm does not start from a MST, but continues with one of the potentially, locally optimal minimum-cost capacitated spanning trees found by the Delta Change, Local Change or Edge Change heuristics. High valency nodes are nodes that are adjacent to at least three other edges. All the edges of these high valency nodes are transferred to the closest node (in Euclidean distance) after which the capacity is reassigned and the total costs are calculated. This process is repeated until no cheaper solution is found.

A comparison of these heuristics and algorithms shows that the edge turn heuristic performs comparably with the delta change or local search heuristic, but at quicker calculation times. Combining the edge turn heuristic with the High Valency Shuffle algorithm can lower the costs of the found networks by 2.6%, but the computational time expands by more than 80 times [85]. Since a quick computational time is necessary to quickly compare different network parameters, the Edge Turn heuristic will be used to search for the MCST.

Figure 3.13 shows an example of one step of the Edge Turn heuristic by using the capacitated MST of figure 3.12 as a starting point. In this step, firstly, edge e_{46} is removed from the graph and replaced with edge e_{67}. Secondly, the capacities are reassigned to the graph's edges. Although dependent on the value of β the total costs of the network with the new edge e_{67} are probably lower than that of the capacitated MST, since the number of edges with a large capacity is significantly reduced, while the length of only one edge has only increased somewhat.

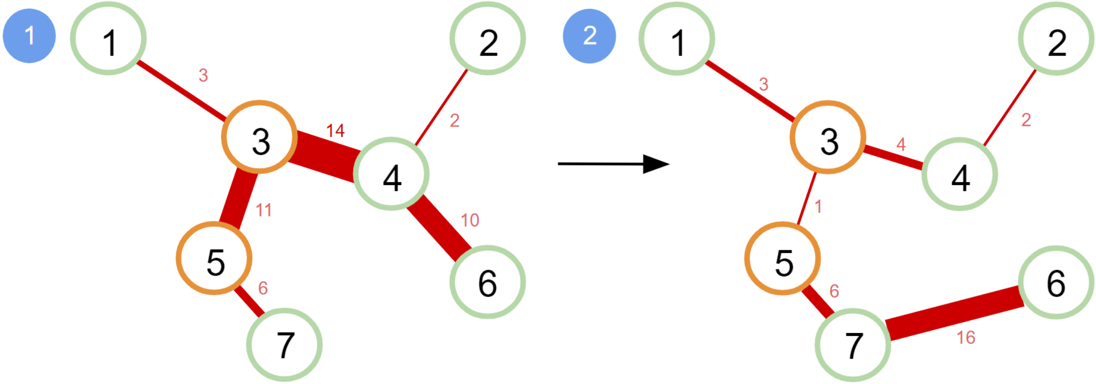


Figure 3.13: Edge Turn heuristic to search for the MCST

Since the Edge Turn heuristic is used to search for Minimum Cost Weighted Trees instead of Minimum Cost Weighted Steiner Trees, it had to be slightly altered before it could be implemented in the model. In each turn, an edge is still removed and the other nodes of the two disconnected parts of the remaining tree are still connected one by one of the initial nodes from the part of the tree that is disconnected. What is different, is that these nodes can not be directly connected via one edge in a Minimum Cost Weighted Steiner Tree Problem, but are constrained by paths of the underlying graph. Therefore, each turn the other nodes of the two disconnected parts of the remaining tree are connected one by one to one of the initial nodes from the part of the tree that is disconnected via the shortest path leading to that initial node. A shortest path is the path from a source to a destination node where the (weighted) length of the path is minimized [52].

This section has outlined the final step of the model in which a (potentially local minimum) cost-optimal Minimum Cost Weighted Region Steiner Tree has been found. All the network's edges now have sufficient capacity to satisfy the demand of all the sites with a demand for an energy commodity. Furthermore, the total costs of these networks are now dependent on the power cables' or pipelines' lengths, capacities and the face weight of regions that they potentially cross. Lastly, the developed model has applied a heuristic to search for MCWRSTs that have lower total costs than the Steiner Minimum Trees. The next chapter will validate the model presented in this chapter and investigate the relations between certain input and output parameters.

4. Model verification and validation

This chapter contains the verification and exploration of the model presented in the previous chapter. The model has three distinct outputs, (1) the discovered network's topology with its (2) associated total network costs and the (3) time that the model takes to run an experiment. Section 4.1 will verify the model by comparing the observed behaviour in network topology and associated total costs with the anticipated results by differentiating a weighted region's face weight. Hereafter, section 4.2 will generate 1140 random experiments. The data extracted from these random experiments is analysed in sections 4.3 and 4.4 to investigate the impact of the number of placed Steiner points on the relative network costs and the input parameters on the model's run time, respectively.

4.1 Model verification

A model must be tested and verified by its developer in order to guarantee that it is correct in its predictions of the performance of the real-world system that it attempts to represent or to forecast the difference between two different model configurations. Verification entails the process of assuring that a model is correct and matches agreed-upon specifications and assumptions. When the process of verifying a model is done well, it leads to improving a model's credibility by decision-makers [11].

To verify the model that was presented in chapter 3, the example graph G of figure 4.1 will be used. In this example the two red terminals a and b need to be connected while potentially crossing the weighted region that is depicted in blue by making use of the connections with the green Steiner nodes. The locations of the graph's nodes can be found in appendix C.

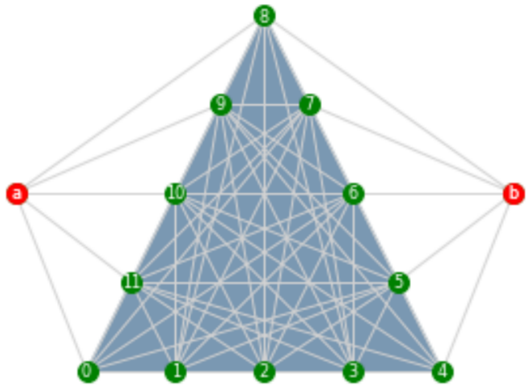
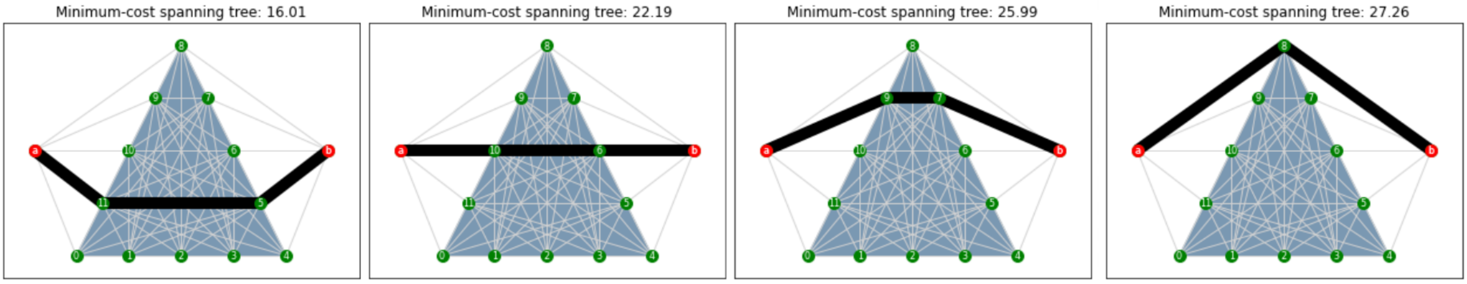


Figure 4.1: Graph G used for the verification process

Because one of the goals of this verification is to observe behaviour in different network topologies, the model will be verified by using different values of the blue weighted region's face weight f_c as variations in other input parameters such as the demand or supply of the source and sink node, will not variate the discovered network topologies.

If the face weight f_c of the weighted region rises, the Minimum Cost Weighted Region Steiner tree should make less and less use of the edges that intersect the weighted region. If the face weight f_c drops, it becomes cheaper to utilize the edges intersecting the weighted region.



Minimum-cost spanning tree: 16.01

Minimum-cost spanning tree: 22.19

Minimum-cost spanning tree: 25.99

Minimum-cost spanning tree: 27.26

Figure 4.2: Model runs with face weights $f_{c_e} = [0.25, 1, 1.5, 2]$

Figure 4.2 shows the optimal found MCWRST with different inputs of the face weight f_c. As anticipated, the discovered optimal network avoids the weighted region more if it becomes costlier to cross the region. Appendix C presents the data underlying the examples with various face weights depicted in figure 4.2 that demonstrate that the model achieves the desired results.

In this section, the effect of differentiations in a weighted region's face weight f_c on the discovered network's layout and total costs have been verified. If a region becomes costlier to traverse through, the model successfully finds networks that reduce the length of pipelines or power cables that traverse the region or even avoid the region altogether. Nevertheless, the face weight f_c is not the only input parameter that has an impact on the model's outcomes. Therefore the next section will generate 1140 random experiments that will be used to analyse the impact of all the input parameters on the model's outcomes.

4.2 Generating 1140 random experiments

The previous section has successfully demonstrated that the model performs as it should in one artificial example. Nevertheless, the results of a single example cannot be used to draw conclusions about the model's behaviour in general. Furthermore, real-life cases are likely to be more complex than the example presented in the previous section, with more nodes and triangulated obstacles for example. To get a better insight into how the model behaves in different circumstances and explore the model's behaviour on a different scale, this section describes how data has been randomly generated with the use of a design of experiments. In the following sections, this data will be used to analyse if and to what extent the input parameters influence the model's outcomes.

Table 4.1 presents the design of experiments that indicates which values the input parameters can take on. For example, the amount of triangles in each scenarios is either one, two or three and there are scenarios in which these nodes can be allocated a demand between 1 and 25 and scenarios in which the nodes' demand can fluctuate between 1 and 50.

Table 4.1: Design of random experiments

Input parameter	Min	Max	Δ	#Experiments
#Terminals	5	15	5	3
#Triangles	1	3	1	3
Min face weight f_c	0.5	1.5	1	2
Max face weight f_c	2	4	2	2
Max demand	25	50	25	2
Max supply	25	50	25	2
#Steiner nodes m	1	10	1	10

The upper six input parameters of the table, namely, the number of nodes, the number of triangles, the minimum and maximum face weight f_c and the maximum demand and supply the sink or sources could take on were used to create random examples. In every example, the nodes were randomly given a non-zero value in the range of the negative maximum supply and the maximum demand, making them either a source or a sink node. Each example contains one of the possible combinations that the input parameters can take. Making the total examples that were conducted 114 ($= 3^2 * 2^4$) examples. The triangles and the nodes are randomly placed in an area of 100 by 100 while making sure that the triangles do not intersect each other. These examples were run 10 times, with the number of Steiner nodes that were placed on the triangles' faces increasing from 1 to 10, resulting in a total of 1140 experiments.

Figure 4.3 shows an example randomly generated by the model that consists of two triangles and ten nodes. Since there is only one Steiner point placed on each face of the triangles, one can deduce that this is the scenario's first run and that 9 more runs of the same scenario will be run; every time with a higher number Steiner nodes.

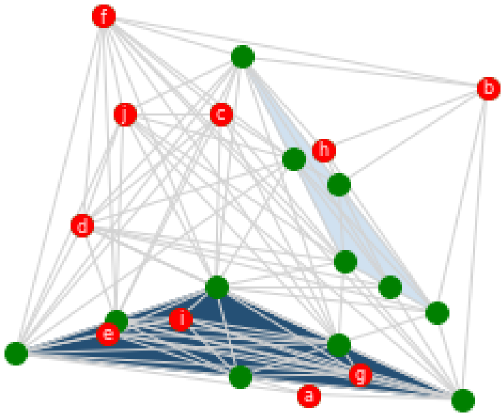


Figure 4.3: Randomly generated example with two triangles (in blue), ten nodes (in red) and one placed Steiner point (in green)

4.3 Impact of the number of placed steiner points on the relative network costs

This section will use the data extracted from the randomly generated examples described in the previous section to visually inspect the impact of the number of Steiner points on the relative network costs and the time the model takes to run one experiment. These two outputs are the most important to evaluate, because the relative network costs represent the accuracy of the found network and the run time determines the model's usability. Naturally, there is a trade-off between these two outputs. Because adding more Steiner points increases the number of potential routes to traverse through the weighted regions, adding more Steiner points will most likely lead to a solution closer to the optimal solution, thereby positively contributing to the model's accuracy. However, more placed Steiner points will increase the number of edges the algorithms described in the previous chapter will have to check, which will expand the model's run time and decreases the model's usability.

To be able to compare the network costs of the 10 different experiments conducted per randomly generated example, the total costs of each run were relativised by using equation 4.1. To find the relative costs of an experiment the difference between the cost of the model run with m amount of Steiner nodes placed on each edge and the minimum costs of the 10 runs of the same example was divided by the scenario's run with the minimum costs.

$$C_{m_{rel}} = \frac{C_m - C_{min}}{C_{min}}, \; m = \{1, 2, ..., 10\} \tag{4.1}$$

Figure 4.4 shows the observed change in relative network costs at various numbers of Steiner points placed on the faces of the triangles. Three matters catch the attention. First and second, the average relative costs decline as more Steiner nodes are placed on the triangulated faces; although the rate of decline of average relative costs becomes less and less as the number of placed Steiner nodes surges. Third, the spread of observed relative costs decreases as the amount of Steiner nodes increases.

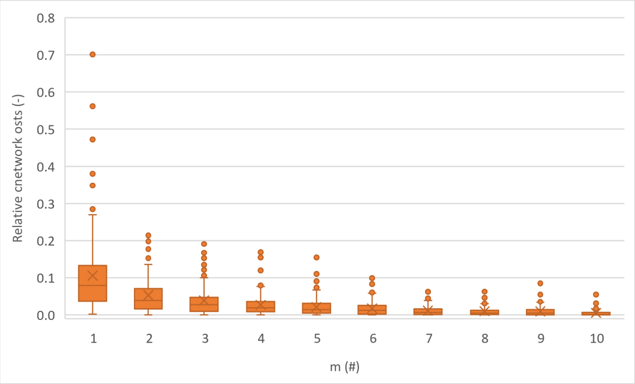


Figure 4.4: Boxplot of the relative network costs at different amounts of placed steiner points m

The first observation can be explained by the fact that placing more Steiner nodes will increase the number of potential paths that an edge can take to cross a weighted region. With many potential paths, it becomes more likely that there is a path whose costs are close to the cost-optimal path that can be taken to traverse a region. Because the Steiner nodes are placed uniformly on the regions' faces, the distance between Steiner decreases with each extra placed Steiner node. This decreased distance between Steiner nodes explains the second observation that the decline of average relative costs decreases with an increase in placed Steiner nodes.

The third observation regarding the decrease in the spread of observed relative costs can be justified by the fact that placing a few Steiner nodes can result in a 'lucky' outcome that is very close to the cost-optimal outcome as well as an 'unfortunate' underlying graphs with potential paths that are relatively far away from the optimal paths.

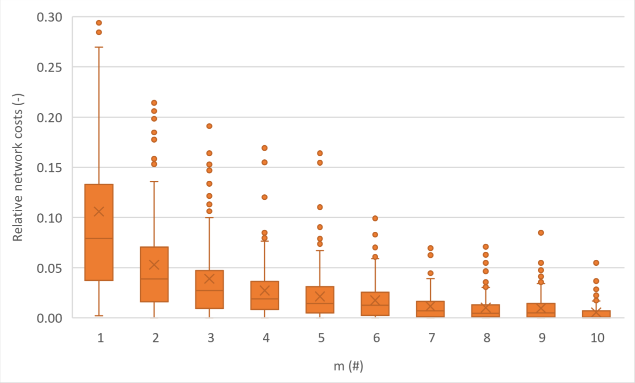


Figure 4.5: Boxplot of the relative network costs at different amounts of placed steiner points m without outliers

Figure 4.5 presents the same data as figure 4.4, but its vertical axis showing the relative network costs is shortened; thereby not showing the outlier data points. This makes the decline in spread, average relative costs and decline in average relative costs stand out even more.

In figure 4.6 two line diagrams are presented that show the average relative costs and the computational time needed to run the experiment in blue and orange, respectively. Not surprisingly, the model takes longer to run as the amount of placed Steiner nodes increases. However, the increase looks to be exponential; a hypothesis that will be further explored in section 4.4.

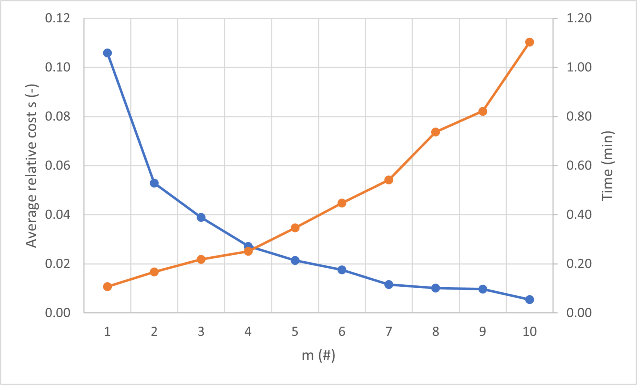


Figure 4.6: Line diagram of the effect of m on the average relative costs and the computational time

In the evaluation of their algorithm, Lanthier et al. [47] remark that six Steiner points per triangulated region suffice to obtain results close to the optimal approximations. After an inspection of 4.6, this number does not seem to be in line with the perceived behaviour of the model described in this report. The average relative costs of the found solutions seem to descend rather steeply until seven Steiner nodes are placed on each face. A potential explanation for this aberration is that the algorithm of Lanthier et al. solves the weighted region path problem, while the developed model searches for a Minimum Cost Weighted Region Steiner Tree, which has a lot more different input parameters than the former problem. These additional input parameters might require a bit more precision to obtain a MCWRST that is close to the optimal approximation.

4.4 Impact of input parameters on the model's run time

The data generated in section 4.2 with the use of running randomly generated examples, can be used for more than to visually demonstrate the influence of the amount of placed Steiner points m on the relative network costs and the computational time. Another application of the data is to search for statistical relationships between the input parameters and the model's computational time, or in other words, the time the model needs to run one experiment, as well. Therefore this section will investigate if there are statistical relations between the minimum and maximum face weight f_c, the maximum demand and supply and the number of nodes, triangles and Steiner points on the one hand and the model's run time on the other with the use of two distinct statistical tests.

The first test to determine statistical relationships is known as the independent Student t-test. This test determines whether there is a significant difference between the means of two unrelated groups. The test's hypothesis assumes that the mean of these two groups is equal [49]. Applied to the data gathered in this research, the test assumes that the model's run time of two different sets of experiments is equal. Since the test requires a categorization into two distinct groups, the minimum and maximum face weight and maximum demand and supply qualify for this test because as can be seen in table 4.1 these input factors have been varied for two values. The results of the conducted Student t-tests that investigated the correlation between these inputs and the computational time are shown in table 4.2.

Table 4.2: Independent Student t-test results

			95% Confidence Interval			
Parameter	Groups	ΔMean	Lower	Higher	2-tailed Significance	Conclusion
Max Supply	[25,50]	0.063	-0.066	0.193	0.337	Insignificant
Max Demand	[25,50]	-0.240	-0.369	-0.111	0.000	Significant
ΔSD	[0,25]	0.001	-0.129	0.130	0.989	Insignificant
Min Face Weight	[0.5,1.5]	-0.023	-0.152	0.107	0.731	Insignificant
Max Face Weight	[2,4]	0.248	0.119	0.377	0.000	Significant

The significance level, also known as alpha, is a value that tells us how likely it is that the hypothesis is true. Normally, in statistical hypothesis tests, a test's hypothesis can be rejected if the significance level is lower than 0.05 [49]. From the results presented in table 4.2, the hypotheses that the values of the Max Demand and Max Face Weight do not influence the computational time can thus be disregarded, because their significance level is much lower than 0.05. On the contrary, the hypotheses that the max supply and the minimum face weight do not influence the model's run time, cannot be rejected. The ΔMean column shows, the direction and quantity of the discovered (significant) statistical links. For example, the average run time of experiments with a maximum demand of 50 had a run time which was 0.240 minutes or 14.4 seconds lower than that of experiments with a maximum demand of 25.

What strikes the attention is that an increase in both the maximum demand and the maximum face weight has a significant effect on the model's run time, while an increase in the maximum supply or minimum face weight does not. To investigate if this difference in influence on the model's run time was related to the absolute difference between the maximum supply and maximum demand that the nodes could be assigned, an extra Student t-test has been performed which results are shown in row ΔSD of table 4.2. This does not seem to be the case however, as the significance level of the test is much higher than 0.05. With no further statistical background, the author does not have an explanation for this remarkable difference. Later on in this section, an explanation for the difference in the influence of variations in the minimum and maximum face weight on the model's run time will be investigated.

The other three input parameters of the number of nodes, triangles and Steiner points have been varied 3 or 10 times and can therefore not be categorized into two groups. As a consequence, the independent t-test cannot be applied to these parameters. The inputs are technically of a ratio scale. However, the range in which they are randomly generated is rather small. Hence, instead of a Pearson correlation test used for values with an interval or a ratio scale, a Spearman rank correlation test, which is normally used on parameters of an ordinal scale is used to test their influence on the run time. Although the latter test has last proving power, in this case, it is the safer option to reduce the chance that the hypothesis is not unjustifiably rejected. The Spearman rank correlation test indicates the strength and direction (either negative or positive) of a relationship between two variables [6]. Table 4.3 presents the results of these conducted Spearman correlation tests.

Table 4.3: Spearman rank correlation test results

Parameter	Input	Spearman coëfficient	2-tailed Significance	Conclusion
#Terminals	[5,10,15]	0.684	0.000	Significant
#Triangles	[1,2,3]	0.433	0.000	Significant
#Steiner points	[1,2,. . . ,10]	0.407	0.000	Significant
ΔFace Weight	[0.5,1.5,2.5,3.5]	0.037	0.158	Insignificant

From the table above, it can be deduced that there are positive relations between the number of nodes, triangles and Steiner points and the model's run time. This effect is the strongest for an increase in the number of nodes followed by the number of triangles and lastly the number of Steiner points placed on the edges of the triangles. These effects can be quite easily explained. An increase in the number of terminals greatly increases the number that for example the edge turn heuristic described in 3.5 needs to be repeated. Whereas an increase in the number of triangles or Steiner nodes greatly increases the number of potential paths there are to connect two nodes, which will increase the time for one repetition of for example the edge turn procedure.

To investigate if the difference in significance in the influence between changes in the minimum and maximum face weight is dependent on the difference between the minimum and the maximum face weight an additional Spearman rank correlation test was performed. As can be seen in the bottom row of table 3.5, there is no significant relation in the difference between the minimum and maximum face weights and the model's run time. An alternative explanation for the significant influence of an increase in the maximum face weight on the model's run time is that a higher face weight might increases the importance of finding the 'cheapest' path through a weighted obstacle, which might lead to more repetitions of the edge turn heuristic, for example, until a local minimum is found.

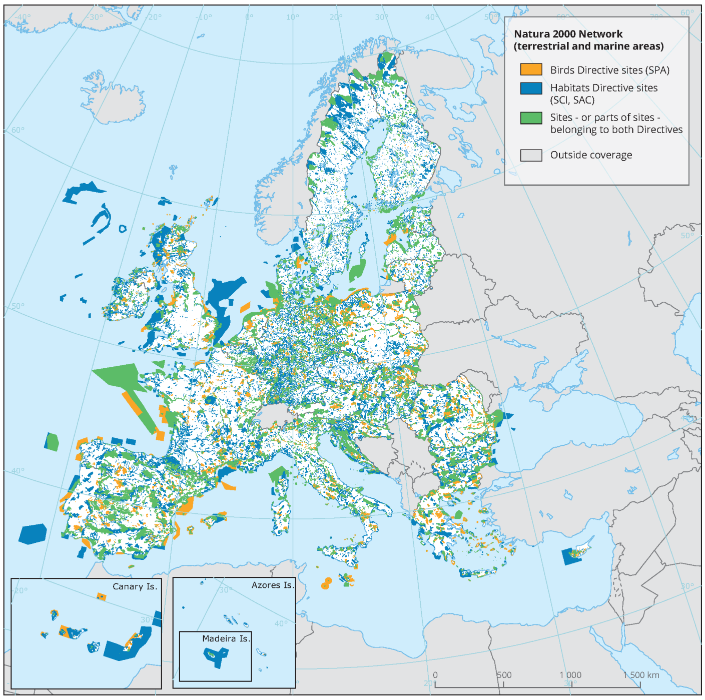
This section has demonstrated that there are statistical relationships between the input parameters of the number of terminal, triangles and Steiner points, the maximum face weight and the maximum demand that can be assigned to nodes and the model's run time. The latter statistical link is negative, while all the others constitute positive relationships. These conclusions have some complications for the model's usability. Especially when the model is used to design large networks with a large number of sink and source nodes, networks in a context of a lot of weighted obstacles, or the design of a very precise cost-optimal network by increasing the number of Steiner points, the model might take an extremely long time to run, which deteriorates the model's usability.

5. Case studies on offshore electricity networks on the North Sea

This chapter outlines the cases of two wind park projects on the North Sea of different scopes which will be used to showcase some application possibilities of the developed model that was verified in the previous chapter. Section 5.1 gives some background information on Natura 2000 areas and describes how these areas can be used as inputs for the weighted regions in the model. Sections 5.2 and 5.3 discuss the routing problems of connecting a wind park with the mainland through the Waddenzee and the design of the power cable infrastructure in the entire Dutch North for the year 2050, respectively.

5.1 Natura 2000 areas as weighted regions

The extensive network of Natura 2000 areas provides a ‘safe haven’ for treasured and threatened species in Europe. The areas, that are protected under European Union (EU) law, stretch over 18% of the EU’s land area and 8% of its marine territory across all of its 27 member states [21]. The main objective of the Natura 2000 project is to create a coherent European ecological network that is able to ensure the survival of the most valuable and endangered animals and plants residing in the European Union [82]. The rationale behind the network is to combat habitat fragmentation, which is the primary cause of the extinction of plants and animals [61]. Figure 5.1 presents an overview of all the Natura 2000 areas, showing both the protected terrestrial and marine regions. Note that this map was created in 2018. Subsequently, the map still displays the Natura 2000 areas of the United Kingdom, even though the country has stopped being an official member of the European Union since the end of January 2020 and thereby ceased being a part of the Natura 2000 network.


Natura 2000 Network
(terrestrial and marine areas)
Birds Directive sites (SPA)
Habitats Directive sites (SCI, SAC)
Sites - or parts of sites - belonging to both Directives
Outside coverage
Canary Is.
Azores Is.
Madeira Is.
0
500
1 000
1 500 km

Reference data: ©ESRI

Figure 5.1: Overview of the EU’s Natura 2000 areas [23]

The EU member states specifically appoint certain areas as Natura 2000 areas which are then protected by the Birds Directive (Directive 79/409/EEG) and the Habitats Directive (Directive 92/43/EEG). These directives provide three different types of legal protection for Natura areas. With regards to its Natura 2000 areas a member state is obliged to [82]:

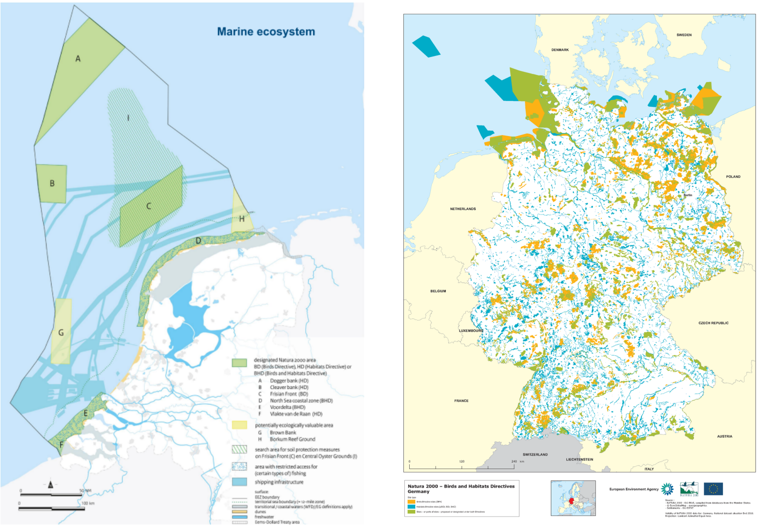
1. Take necessary conversation measures to preserve the natural habitats and species for whom the area is assigned.

2. Ensure that existing use of the assigned Natura 2000 area; through recreation, agriculture, fishing or military use, for example, shall not lead to a significant deterioration of the quality of the habitat or disturbance of the species for whom the area is assigned.

3. Assess each new plan or project that potentially has significant implications for the site's conservation objectives. The competent national authority shall only agree to the plan or project after having ascertained that the project will not adversely affect the integrity of the site concerned.

The development of new energy infrastructures only concerns the third form of protection of the habitats. Therefore, the other two forms of protection will not be discussed any further. A consequence of the third form of protection for the development of energy infrastructures in or around Natura 2000 areas is that, without prior authorization, it is prohibited to realise an infrastructure project that could potentially harm the quality of a natural habitat or have a disturbing effect on the species for which the area is assigned [70].

If an ecological assessment of an infrastructure project shows that it will harm the quality of a natural habitat or interfere with the species that inhabit it, a project can only be permitted by the competent national authority if, and solely if, the requirements of the so-called 'AIC-exception' have been fulfilled. This 'AIC-exception' entails that a permit for such a project can only be given if there are no Alternative solutions and there are Imperative reasons of overriding public interest, including those of economic or social nature. Moreover, the member state should take Compensatory measures, before the construction of the project can start, to ensure that the coherence of the Natura 2000 network is not altered, even temporarily [82].


Marine ecosystem
A
B
C
D
E
F
G
H
I
designated Natura 2000 area
BD (Birds Directive), HD (Habitats Directive) or
BHD (Birds and Habitats Directive)
A Dogger bank (HD)
B Cleaver bank (HD)
C Frisian Front (BD)
D North Sea coastal zone (BHD)
E Voordelta (BHD)
F Vlakte van de Raan (HD)
potentially ecologically valuable area
G Brown Bank
H Borkum Reef Ground
search area for soil protection measures
on Frisian Front (C) en Central Oyster Grounds (I)
area with restricted access for
(certain types of) fishing
shipping infrastructure
surface:
EEZ boundary
territorial sea boundary (= 12-mile zone)
transitional / coastal waters (WFD)/EG definitions apply)
dunes
freshwater
Eems-Dollard Treaty area
DENMARK
SWEDEN
POLAND
NETHERLANDS
BELGIUM
LUXEMBOURG
CZECH REPUBLIC
FRANCE
SWITZERLAND
LIECHTENSTEIN
AUSTRIA
ITALY
Natura 2000 – Birds and Habitats Directives
Germany
European Environment Agency

(a) The Dutch North Sea [58]

(b) The German Seas [22]

Figure 5.2: The Dutch and German Natura 2000 areas of the North and Baltic Seas

The dark green shapes in figure 5.2a represent the six Natura 2000 areas located in the Dutch North Sea. Likewise, the blue, orange and green polygons in figure 5.2b represent the German Natura 2000 areas in the North and Baltic seas. Since these areas could obstruct potential cable routes or require additional Compensatory measures if power cables do cross Natura 2000 areas, these areas should be taken into account while developing energy infrastructures. Especially during the routing planning phase of energy infrastructure layouts the effects of Natura 2000 areas on potential pipeline or power cable routes should be assessed. Modelling these habitats as weighted regions as described in sections 2.3 and 3.2 could assist routing planners in comparing the network cost reductions obtained if Natura 2000 areas are crossed instead of avoided. Subsequently, these benefits could be compared to the costs of potentially required compensatory measures. This allows the routing planner to identify the tipping point at which the price of required compensatory measures, avoiding the areas becomes more economically appealing.

5.2 Power cable route from wind park Doordewind to the main shore

The Dutch government aims to expand its installed wind capacity from the currently installed capacity of 6.6 GW (on both land and sea) to an installed capacity of 21 GW by 2030 [50]. To realize this goal, the government plans on developing several offshore wind farms on the North Sea in the next decade. One of these planned wind farms, called ‘Doordewind’, will have an installed capacity of 4 GW and will be located north of the Wadden islands [67]. During the routing planning phase of the power cable connecting ‘Doordewind’ to the mainland, two onshore landing points, Eemshaven and Vierverlaten, were considered, which are visualized in figure 5.3. In the image, ‘Doordewind’ is depicted as the blue triangle labelled with the number 5. The other blue polygon in the image labelled ’5 mb’ is a potential site for the development of offshore wind energy after 2030 which will be disregarded in this application example of the model.

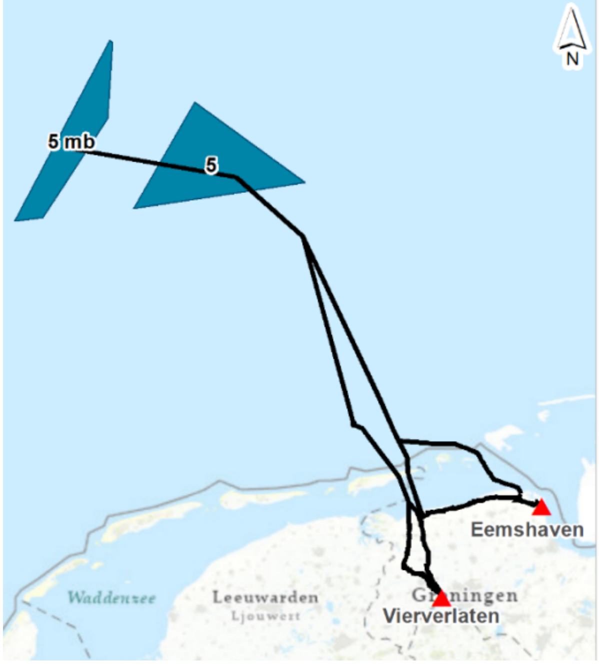


Figure 5.3: Considered routes connecting offshore wind farm Doordewind [86]

Figure 5.4 shows the last part of the considered routes that connect Doordewind with the mainland in more detail. A report of Royal HaskoningDHV commissioned by the Ministry of Economic Affairs and Climate Policy [70] reviewed the potential routes on their impact on ecology, the feasibility of required technology, space occupancy and estimated costs. The research concluded that route 11, heading to landing point Vierverlaten and route 6, connecting the Doordewind to the Eemshaven were the most promising.

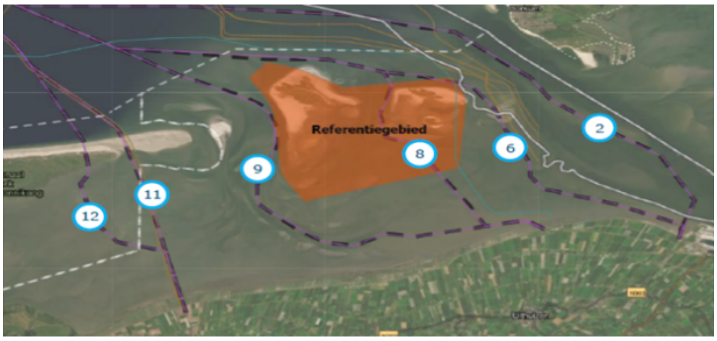


Figure 5.4: Considered routes connecting wind farm Doordewind in more detail [86]

The power cable route to Vierverlaten has the disadvantage that it crosses the Wadden Sea and the Wadden Island Schiermonnikoog, both part of the Natura 2000 area shown as the North Sea coastal zone D in figure 5.2a. A choice for this route will entail the digging of long underground power cables through this valuable and protected landscape [86], which could have a negative impact on this Natura 2000 habitat. The other potential route, to the Eemshaven landing point however, could cross two of the German Natura areas shown in figure 5.2b.

Figure 5.5 shows the conceptualization of the routing problem which consists of finding the most cost-effective route from the Doordewind wind park to one of the two landing sites which can be used to compare the economical benefits of the shorter route 11 with the more Waddenzee friendly route 6. The weighted region that represents the North Sea coastal Natura 2000 area is depicted as the black dashed line polygon. The two potential landing points are depicted as red circles and the offshore wind farm Doordewind is represented by the green circle. In section 6.1 the results of this investigated routing problem will be presented.

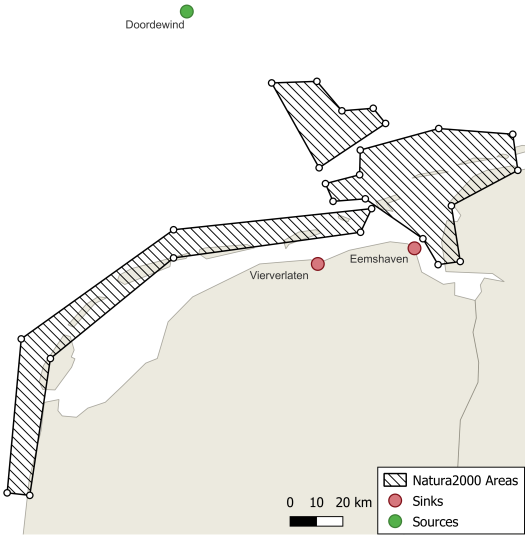


Figure 5.5: Conceptualization of Doordewind power cable routing problem

This section has discussed the first case study of the routing problem of connecting 1 wind farm in the Dutch part of the North Sea with 2 potential landing points while potentially crossing 3 Natura 2000 areas. Because this routing problem only contains one wind farm that needs to be connected, the case study's outcome will likely be a single power cable from one of the two landing points to the wind farm, which can hardly be called an energy network. To show the full potential of the developed model, the following section will elaborate on the second case study which entails the design of an entire power cable infrastructure spanning 15 Dutch offshore wind farms with 4 potential landing sites.

5.3 Dutch North Sea wind park design of 2050

The sustainability goals of the Dutch government for the year 2050 are even more ambitious than those of 2030. Their aim is to have completely switched to renewable energy sources in 2050. Offshore wind energy will play an important role in the energy mix of the future, with predicted scenarios of having between 38 and 72 GW installed offshore wind energy capacity in the North Sea [66]. To realize this ambition the Dutch government has already assigned 15 areas of the Dutch part of the North Sea in which market players could develop wind parks in the next three decades. Figure 5.6 shows these zones in orange and yellow. If all of these zones are fully developed, the total installed offshore wind capacity could reach 61.5 GW [1].

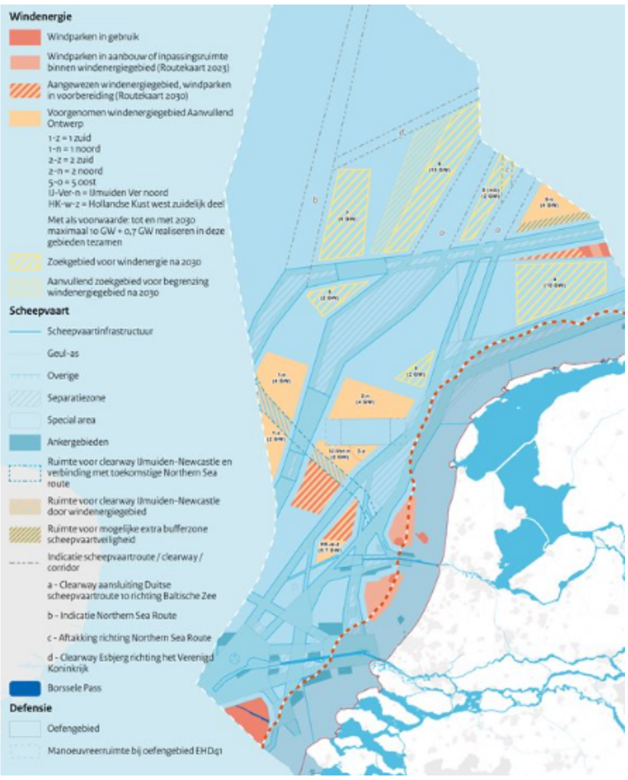

Windenergie
Windparken in gebruik
Windparken in aanbouw of inpassingsruimte binnen windenergiegebied (Routekaart 2023)
Aangewezen windenergiegebied, windparken in voorbereiding (Routekaart 2030)
Voorgenomen windenergiegebied Aanvullend Ontwerp
1-z = 1 zuid
1-n = 1 noord
2-z = 2 zuid
2-n = 2 noord
5-o = 5 oost
IJ-Ver-n = IJmuiden Ver noord
HK-w-z = Hollandse Kust west zuidelijk deel
Met als voorwaarde: tot en met 2030 maximaal 10 GW + 0,7 GW realiseren in deze gebieden tezamen
Zoekgebied voor windenergie na 2030
Aanvullend zoekgebied voor begrenzing windenergiegebied na 2030
Scheepvaart
Scheepvaartinfrastructuur
Geul-as
Overige
Separatiezone
Special area
Ankergebieden
Ruimte voor clearway IJmuiden-Newcastle en verbinding met toekomstige Northern Sea route
Ruimte voor clearway IJmuiden-Newcastle door windenergiegebied
Ruimte voor mogelijke extra bufferzone scheepvaartveiligheid
Indicatie scheepvaartroute / clearway / corridor
a - Clearway aansluiting Duitse scheepvaartroute 10 richting Baltische Zee
b - Indicatie Northern Sea Route
c - Aftakking richting Northern Sea Route
d - Clearway Esbjerg richting het Verenigd Koninkrijk
Borssele Pass
Defensie
Oefengebied
Manoeuvreerruimte bij oefengebied EHD41

Figure 5.6: Designated search areas for future wind farm development [66]

Of these 15 wind park locations, only the Borrsele wind park, off the coast of the southern province Zeeland, is completely operational. The wind farms Hollandse Kust South and North and the wind farm north of the Wadden islands already generate electricity, but their capacity will be greatly expanded in the coming years. The current installed capacity of offshore wind farms is 2.5 GW, which is only a fraction of the potential 61.5 GW that could be installed in 2050 [1].

As can be seen in figure 5.7, power generated on the North Sea is currently transported to four landing points, located at Borrsele, Maasvlakte, Beverwijk and Eemshaven that connect the offshore wind parks to the high voltage net [66]. In this conceptualization of the offshore wind power cable infrastructure system of 2050 design problem, it is assumed that in the coming decades, these four landing points will remain in use and no additional landing points will be constructed.

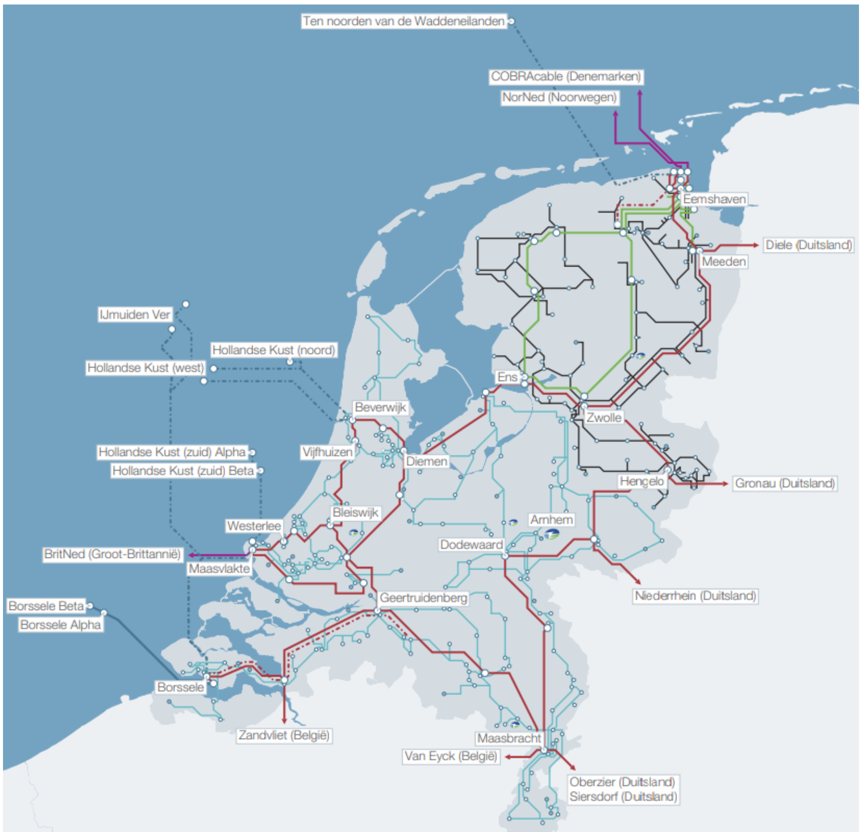


Figure 5.7: The Dutch electricity grid [76]

Comparing the sites assigned for future wind farm development and the power landing points with the Natura 2000 areas on the North Sea depicted in sections 5.6, 5.7 and 5.1 respectively, six Natura 2000 regions stand out. The Dutch Vlakte van de Raan, the Voordelta, the Frisian Front and the North Sea coastal zone and two protected regions in the German North Sea are all Natura 2000 areas that block potential pipeline routes that directly connect wind farm sites with straight cables with the landing points. Therefore these areas should be included in the process of designing a power cable infrastructure that transports the produced power of future wind parks.

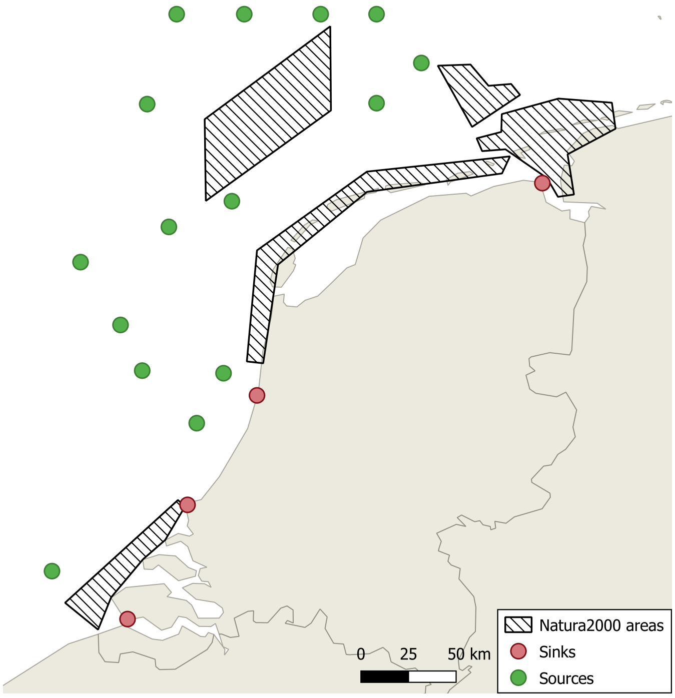


Figure 5.8: Conceptualization of the North Sea's 15 wind parks routing problem

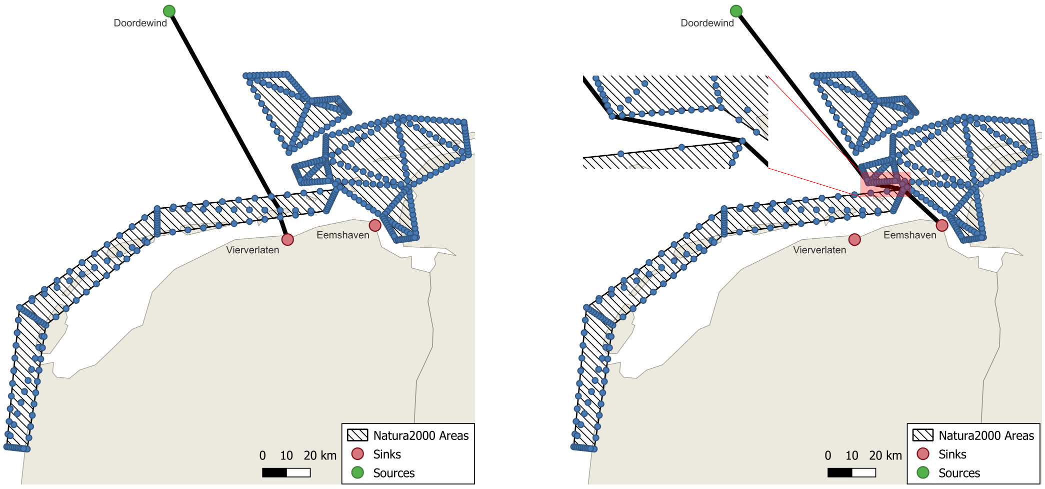
Figure 5.8 shows the conceptualization of the routing problem that aims to connect all the fifteen wind parks with the four onshore landing sites. The three dashed-lined polygons represent the four Natura 2000 areas that could alter the most economically viable power cable layouts. Because the two Natura 2000 areas, Voordelta and Vlakte van de Raan, are adjacent (see figure 5.2a), the most southern polygon represents both of these areas. Considering that only 4.1% of the potential capacity of 2050 is currently installed, the existing offshore power cables are not considered in the design of the offshore power cable infrastructure of 2050.

6. Cost-optimal electricity infrastructures on the North Sea

This chapter discusses the outcomes of the two case studies that were introduced in the previous chapter. Section 6.1 displays two potential routes for connecting the Doordewind wind farm with the Dutch coastline and shows how various face weights f_c influence the relative costs of these routes. Next, section 6.2 discusses three different discovered electricity network topologies that connect 15 Dutch wind farms with 4 landing points on the Dutch coastline.

6.1 Connecting offshore wind park Doordewind to the main shore

This section will apply the model to the real-life case of the routing problem that concerns connecting the offshore wind park Doordewind to one of two landing points on the northern part of the Dutch coastline as described in section 5.2. The coordinates of the wind farm, the two landing points and the weighted region polygon representing the North Sea coastal Natura 2000 area as depicted in figure 5.5 were extracted from QGIS by using the UTM zone 31N coordinates reference system. The UTM system measures coordinates between certain degrees of longitude in meters [80], which makes modelling the distance between two coordinates more straightforward. Just as in the development plans, the Doordewind wind park has a capacity of 4 GW. As is common practice in the modelling of energy infrastructures [34, 35, 55], the cost capacity exponent β is set to 0.6; indicating that there are economies of scale advantages to building higher capacity pipelines. The number of Steiner points placed on the edges of the triangulated faces m is set to 10, because the exploratory results of section 4.3 have indicated that the benefits of setting a higher m are minimal.



(a) $f_c = 2.99$

(b) $f_c = 3.00$

Figure 6.1: The tipping point of the Doordewind routing problem

By using the input data described above, different power cable routes could be compared on their costs for different values of the face weight f_c for the Dutch and German maritime Natura 2000 weighted regions. The cost-optimal route for low values of the face weight f_c is the route with the shortest length from Doordewind to Vierverlaten. When the face weight f_c increases, travelling through the Natura 2000 area becomes more expensive, and the route with the longer distance to Eemshaven becomes more economically attractive. Figure 6.1 shows that the tipping point of the cost-optimal route lies around a face weight f_c value of 2.99. Increasing this value with only 0.01 to 3.00 alters the most cost-optimal route from Vierverlaten to Eemshaven. Note that at a weighted region weight of 2.99 a Natura 2000 area is still traversed, while in the example of a face weight of 3.00 all the weighted regions are avoided.

Table 6.1 presents the chosen landing points and the related relative costs of connecting Doordewind at different values of face weight f_c. Experiments 3 and 4 illustrate the tipping point of this routing problem at which an increase of the face weight f_c of only 0.01 results in a different power cable layout. Note that the capacity of the pipelines is never divided between the two landing points, but in all experiments only one landing point is connected to Doordewind with a power cable of 4 GW that can handle the full capacity of the wind park. This observed model behaviour is easily explainable with the cost capacity exponent β which was set at 0.6. Because of the economies of scale that the parameter represents, it will always be more cost-effective to construct one power cable with a capacity of 4 GW than, for example, two pipelines of the same length with a capacity of 2 GW. The one exception is if β is assigned a value of 1 at which there are no economies of scale at play and the costs of constructing two power cables of 2 GW is equal to the costs of constructing one power cable of 4 GW.

Table 6.1: Relative costs of connecting wind park Doordewind at different face weights

Experiment	**Face weight f_c**	**Power to VV (GW)**	**Power to EH (GW)**	**Relative costs**
1	1.00	4	0	0.0%
2	2.00	4	0	+8.32%
3	2.99	4	0	+15.29%
4	3.00	0	4	+15.30%
5	4.00	0	4	+15.30%

The last column of the table shows the relative costs of the different routes. Since this Doordewind power farm case is used solely as an example to demonstrate a single source application of the model, the input parameters do not represent the actual material or digging costs associated with power cables of different lengths and capacities. Therefore, the total costs of the different routes can only be compared relatively and not absolutely.

Not surprisingly, the more expensive it is to traverse the weighted region, the higher the relative costs become. After the tipping point of a face weight of 2.99 – 3.00, the relative costs do not increase when the face weight is further increased. If figure 6.1b is closely inspected, one might notice that the power cable layout does not intersect any of the weighted regions. Therefore the costs of the power cable route to Eemshaven will not be affected by a further increase in the face weights of the North sea's weighted regions.

Even though the model does not allow for an absolute comparison of the costs of the different routes, the model's outcomes are not that different from that of the extensive report conducted on this routing problem by Royal HaskoningDHV. Their report concluded that there are too limited distinctions between the potential routes, cost-wise, because the distances of the routes are all quite similar [70]. The report even goes so far as to not make any distinction whatsoever in the evaluation of the routes on costs, because the required investment costs of the different routes are in the same order of magnitude. A maximum difference of 15.3% (since a face weight less than 1 does not seem realistic for a protected weighted region like that of a Natura 2000 area) seems also in the same order of magnitude, thus the model is quite accurate in its predictions.

6.2 Exploration of the North Sea's electricity infrastructure of 2050

In the previous section, the problem of connecting the wind park Doordewind was described. Although this application example demonstrated how a change in a weighted region face weight f_c influenced the chosen power cable route, it did not show the full potential of the model. Instead of solving the Minimum Cost Weighted Region Steiner Tree problem as discussed in 2.5, it compared the costs of two minimum cost weighted region shortest paths. To do show the full potential of the model, this section will explore the case that was introduced in section 5.3 different power cable infrastructure layout designs that connect all the planned wind park sites for 2050 in the Dutch part of the North Sea.

The coordinates of the wind farm and landing points were extracted from QGIS with the UTM 31 North zone. The capacities of the wind farms were taken from 4C Offshore, a company that provides an online Geographical Information System that contains information about all contemporary and future wind farm sites [1]. Appendix D contains data on all the coordinates and capacities of the wind farms utilized in this real-world example. The total potential capacity of the fifteen wind park combined is 61.5 GW. Because today, only 2.5 GW of offshore wind energy capacity, or the small fraction of 4.1% of the potential offshore wind energy capacity of 2050, is installed, the assumption is made that there is no existing power cable infrastructure yet in place. Just like in the scenario described in the previous section, the cost capacity exponent β and the number of Steiner points placed on the edges of triangulated faces m have been given values of 0.6 and 10, respectively.

Figures 6.2 to 6.4 on the next page show three discovered networks that connect all the fifteen wind park sites with the four landing points on shore for different face weights f_c of the Natura 2000 areas being $1, 1.5$ and 2, which have been chosen, since they demonstrate all the potential network topologies with their associated total network costs that have been observed during experimentation with this case study. The thickness of the black lines that connect the nodes, indicates the capacity of the power cables. All three found networks are forests instead of trees. Forests are graphs that consist of disjoint trees. This was to be expected, since if the discovered cost-optimal networks had been trees, they would only have made use of one landing point, which would have increased the total length of the power cables significantly.

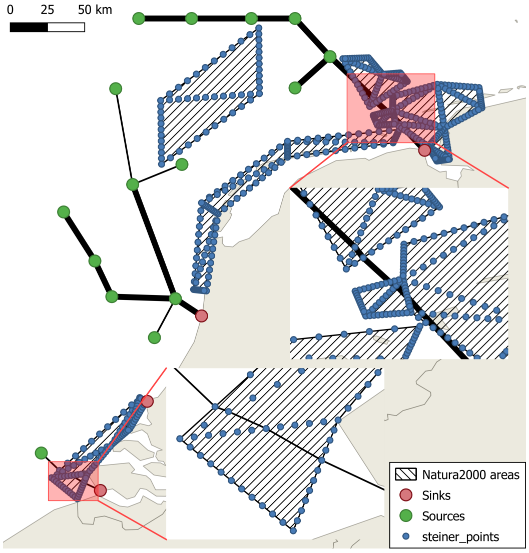


Figure 6.2: North Sea's future wind park sites with $f_c = 1.0$

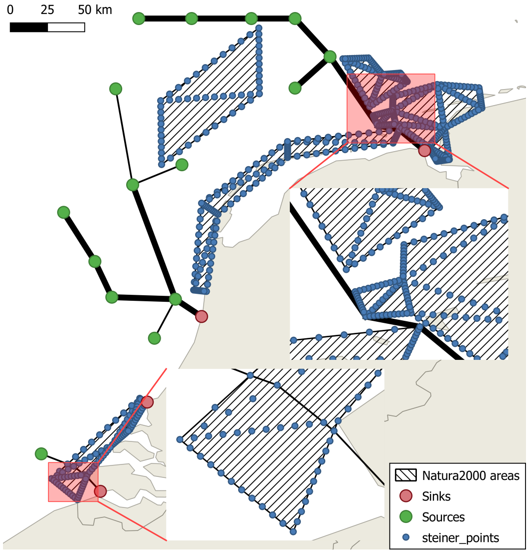


Figure 6.3: North Sea's future wind park sites with $f_c = 1.5$

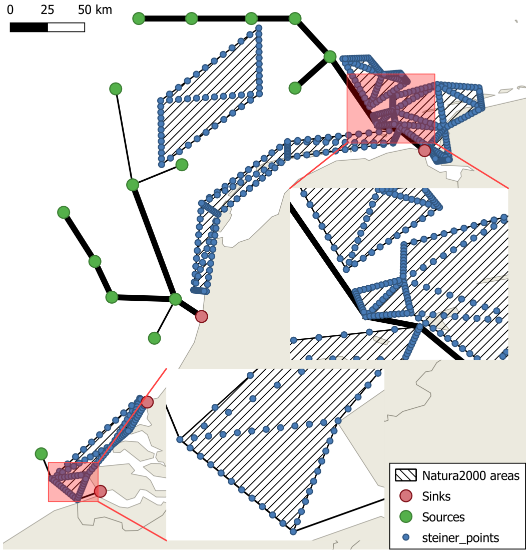


Figure 6.4: North Sea's future wind park sites with $f_c = 2.0$

Two Natura 2000 zones, the Frisian Front and the North Sea coastal zone, are never crossed or surpassed in all of the three scenarios. Figure 6.2 shows that if the weighted regions are assigned a face weight equal to that of the exterior of weighted regions, both the Natura 2000 areas before the coast of the southern province of Zeeland and the two included maritime German Natura 2000 areas are crossed. When the face weights of the weighted regions are increased to 1.5 as shown in figure 6.3, the northern power cable starts avoiding the German Natura 2000 areas and the southern power cable chooses a shorter route through the weighted region before the Zeeland delta. At a face weight f_c of 2, the Natura 2000 region for the coast of Zeeland is also evaded.

Table 6.2 presents the relative costs of five conducted experiments with different face weights f_c that illustrate the tipping points in the discovered optimal networks. When the face weight surpasses the 2, the relative costs of the found networks do not increase anymore. This is due to the fact that all the weighted regions are already avoided by the power cables in the $f_c = 2$ experiment. Moreover, the difference between the relative costs over all experiments is rather small. This can be explained by the fact that most of the wind sites can be connected to a landing point or each other without ever having to cross a weighted region. Therefore an increase in the face weight of weighted regions will barely affect the network's costs.

Table 6.2: Relative costs of the network connecting the Dutch North sea's wind park sites

Experiment	**Face Weight f_c**	**Power LP1**	**Power LP2**	**Power LP3**	**Power LP4**	**Relative costs**
1	1.0	35300	24628	0	1502	+0.0%
2	1.5	35300	24628	0	1502	+1.42%
3	2.0	35300	24628	0	1502	+1.46%
4	2.5	35300	24628	0	1502	+1.46%
5	3.0	35300	24628	0	1502	+1.46%

A remarkable observation that can be made when looking at table 6.2 is that the landing points are assigned the same capacity in each scenario. In other words, the same wind sites are connected to the same landing point for each of the five experiments. Another interesting outcome of the experiments is, that there is never a power cable connected to LP3, which represents the Maasvlakte landing point. This is surprising, since in reality, four wind parks were connected to the Maasvlakte as can be seen in figure 5.7. Apparently, more factors than just the required network investment expenditures were taken into consideration in designing the North Sea's power cable infrastructure layout. Therefore, the next section will investigate which other factors play a role in decision-making processes regarding the design of energy infrastructures and how the developed model can be of use in this decision-making process.

7. Using energy network modelling in decision-making processes

The previous chapter examined how the developed model can be used to identify the energy infrastructures with the lowest investment costs in a context of regions with varying costs for constructing power cables. However in practice, several other factors than merely the minimization of initial investment costs are considered when comparing various energy network designs. Therefore, this section will discuss how the insights gained from the model could be used in a decision-making process regarding the design of energy infrastructures. First, section 7.1 will elaborate on the benefits and limitations of using graph theory models in decision-making processes. Second, section 7.2 will discuss how the model described in chapter 3 could have been used in the decision-making process of the first real-life case study of connecting wind farm Doordewind with the Dutch coastline.

7.1 Advantages and disadvantages of using graph theory models in decision-making processes

This section will discuss two advantages and four limitations of using the results of graph theory models in real-life decision-making processes. A first advantage of graph theory is that it is able to optimize several different parameters. This makes it a rather suitable method to consult during decision-making processes when there is a strong relative importance of one parameter over another [74]. In these situations, the graph theory model's outcome which has optimized this most important parameter could be used as a starting point, initial design or food for thought in the decision-making process. Later on in the process, this initial design could be modified or altered to better fulfill the wishes of the actors involved or better satisfy the constraints set by other, less relevant, parameters.

A second advantage of using graph theory models in decision-making processes is that both the assumptions and outcomes of graph theory models are easy to interpret. Rudin [71] warns of the rise of the application of black box models in high-stakes decision making, like in that of critical (energy) infrastructures, in society. She argues that adopting black box models which employ prediction techniques that are not even comprehensible to their developers, causes transparency and accountability problems, especially in the public sector. If possible, these black-box models should be replaced by interpretable models. For example, instead of using a black box preventive criminal behaviour model that uses unclear and non-transparent prediction strategies, crime fighters could use conventional graph theory algorithms combined with a social graph model which maps the social connections between criminals known to the police to predict which small-time criminals are likely to become the 'Escobars' of the future.

Unfortunately, there are also downsides to using graph theory. Zarghami et al. [87] mention five limitations of using graph theory networks in decision-making processes concerning infrastructures:

1. A graph is merely a model of a network that simplifies the network as a set of nodes that are connected with edges. This abstractification does not quite capture the complexity of real-world networks which impedes a realistic depiction of infrastructures.

2. Using graph theory to identify the most 'important' actors, locations or connections in a network is highly dependent on the chosen characteristic to measure 'importance'.

3. Each infrastructure is unique and is developed in a different technological, political and social context. Using graph theory as a 'one-size-fits-all' approach therefore might lack effectiveness in some cases, because the conventional graph theory tools are not able to include these contextual differences.

4. Graph theory methods are static in the sense that they can only provide snapshots of the development of a network at discrete times. In reality, the topological characteristics are likely to variate over time as a response to a change in internal and external factors.

5. Because graph theory focuses on the connections between demand or supply sites rather than on the sites themselves, exclusively using graph theory tools yields insufficient data on the changes that some of these sites are not able to deliver their supply or have a lower demand than anticipated [8].

This section has mentioned some benefits and limitations of using graph theory models in decision-making processes in general. The following section will elaborate on how the model developed in this research report could be used in the decision-making process regarding connecting a Dutch North Sea wind farm to the Dutch coastline.

7.2 Using the developed model in the Doordewind wind farm decision-making process

As discussed in chapter 5, the Dutch government is planning to develop several offshore wind farms in the Dutch North Sea in the coming decade. One of them is called Doordewind, a 4 GW wind park located north of the Netherlands which is planned to be realized in 2030. A technical report that studied the 'best' way to connect the wind farm with a power cable that will transport the electricity generated by the wind farm to the Dutch shore, came up with six potential power cable routes which are depicted in figure 7.1.

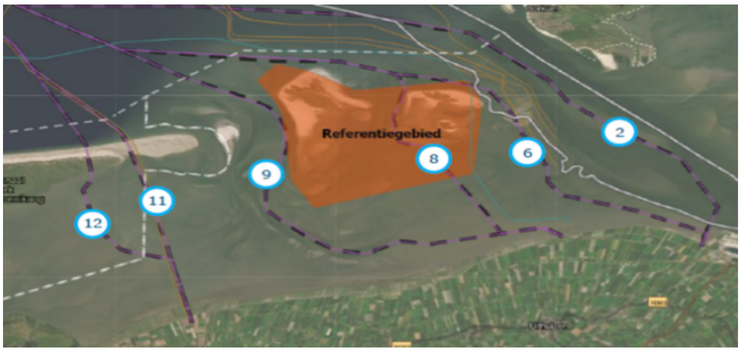


Figure 7.1: The six investigated routes to connect wind park Doordewind

Naturally, one of the evaluation criteria was the initial investment costs of the power cables. However, the different routes were also evaluated on five other criteria being their (1) life-time costs, (2) the realisation time which includes the design, permit and construction phase, (3) effects on nature, (4) feasibility of required construction methods and (5) the effects of topology on the location and stability of underwater gullies and salt marshes [70].

The model described in this research report is probably not very helpful in evaluating the realisation time, life-time costs and feasibility of required construction methods of the different routes. Nevertheless, by conceptualizing the protected nature areas underwater gullies and salt marshes as weighted regions, the model could be of use to quantify the trade-offs between the initial investment costs on the one hand and the effects on nature

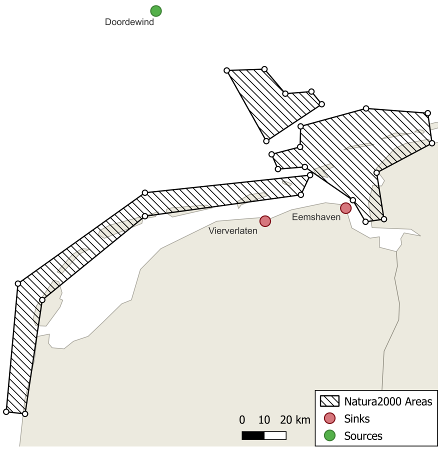
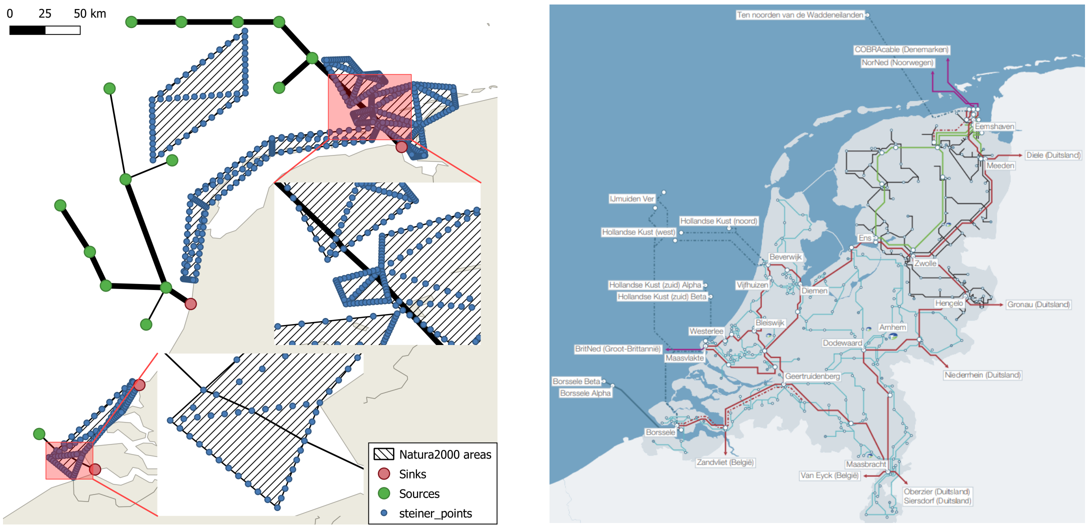


Figure 7.2: Conceptualization of Doordewind power cable routing problem

and the effects of the routes' topologies on the location and stability of underwater gullies and salt marshes on the other. Figure 7.2 illustrates how this research studied the first of these trade-offs by conceptualizing the protected Natura 2000 areas as weighted regions.

The very specific evaluation criterion of the effects of the power cable routes on the stability of underwater gullies and salt marshes is a clear example of the third limitation of using graph theory models in infrastructure modelling mentioned in the previous section that every infrastructure is unique and developed in its individual context. Luckily, because the developed model allows for the modelling of concave, convex and triangulated polygons as weighted regions, the model might still be of service for a broad spectrum of specific spatial requirements.

Next to the evaluation criteria mentioned in the technical report, the responsible minister of Economic Affairs and Climate laid down two additional objective criteria to evaluate the routes. The timing of finalizing the power cable should match the demand expectations for sustainable energy. Additionally, the landing point should be able to absorb the electricity generated by the wind parks and transform it so it can be transported via the national high-voltage grid [86]. Both of these additional criteria are possible explanations for the discrepancy shown in figure 7.3 between the cost-optimal network designed by the model and the real-life Dutch electricity grid.



(a) Cost-optimal design by the model

(b) Real-life Dutch electricity grid

Figure 7.3: The Dutch and German Natura 2000 areas of the North and Baltic Seas

The model only took the initial investment costs into account, which resulted in an offshore electricity infrastructure that did not assign a single power cable to the Maasvlakte landing point (the second-most bottom landing point in figure 7.3a. In reality, the Maasvlakte landing point is close to the Rotterdam harbour which is the home of one of the largest industrial complexes in the Netherlands, as well as, South-Holland, the most densely populated province of the country, which both have a large demand for electricity.

Ultimately, the minister of Economic Affairs and Climate decided to disregard all the power cable routes that ended up at the Vierverlaten landing point, because there was no political support for the construction of the power cable in the region [86]. Graph theory models are likely to be ill-equipped to make these kinds of political considerations.

8. Conclusion

The pressing issue of human-induced climate change demands the decarbonization of economies around the world in the coming decades. Satisfying these sustainability challenges requires the development of new energy infrastructures and the expansion of existing energy networks for which minimizing their capital investment costs is crucial. However, the models that estimate the required investment costs of energy networks, currently lack the possibility to include spatial regions in which the costs of constructing pipelines or power cables can be differentiated. Therefore this research has tried to answer the main research question: *How can geographical cost-differentiations be included in network energy infrastructure cost-minimization models?* To this end, a new problem called the Minimum Cost Weighted Region Steiner Problem has been formulated. To solve this problem, a novel method was proposed that combined algorithms borrowed from the fields of graph theory and computational geometry.

The developed method finds a cost-optimal topology for new energy infrastructures of multiple sites with either a demand or a supply of an energy commodity in a context of weighted regions. Weighted regions are geographically limited areas in which the costs of constructing energy infrastructures can differentiate. The model exploits the geographical locations of the demand or supply sites and weighted regions, the demand or supply of the sites, the relative costs of constructing energy networks in these weighted regions and a user-defined number of points placed on the borders of the weighted regions that represent potential paths that can be taken to cross these regions to construct a network layout of pipelines or power cables with specific lengths, capacities and routes to traverse the weighted regions that ensures that the demands of all sites are satisfied.

Through a series of randomly generated experiments, the influence of input parameters on the model's performance has been investigated. Although the literature study has shown that the optimal solution to Weighted Region problems cannot be computed exactly, adding more points on weighted regions that represent potential paths to traverse these regions results, on average, in discovered networks that appear to get closer and closer to this optimal solution. Nevertheless, the gains of adding more border points on weighted regions seem to flatten after more than 7 points are placed on the edges of the weighted regions, while simultaneously the model's run time starts to increase steeply.

The statistical tests performed on the data gathered from the randomly generated experiments indicate that there are statistical relationships between the input parameters of the max demand of sink nodes, the maximum weight assigned to weighted regions, and the number of terminal nodes, triangles and placed border points on the model's computational time. This entails that if the values of these input parameters are increased, the model will, on average, require a longer time to run.

The performed case studies have shown that the total costs and network topologies are highly dependent on the weights assigned to weighted regions. Only a small rise in the weight assigned to a region can alter the taken route of a specific pipeline or power cable greatly, often leading to networks that choose to avoid weighted regions altogether. Nevertheless, a change in the weight assigned to regions did not impact the overall construction costs of the discovered network topologies in a remarkable manner.

Since the desirability of energy infrastructures is not solely dependent on a minimization of the investment costs required to construct them, the outcomes of the model are not ready-to-use designs that can be implemented immediately. However, the networks provided by the model are suitable as a starting point in decision-making processes and can clearly quantify the trade-offs between the required investments costs on the one hand and the difficulties that arise when constructing energy infrastructures in areas that are protected by nature conservation laws, are politically or culturally sensitive or owned by other actors.

In conclusion, this study has proposed a new method that allows for cost variation of geographical demarcated areas in energy network modelling. To quote George Box; “all models are wrong”, but hopefully this new approach can contribute to making energy network modelling a bit less so.

9. Discussion

This chapter contains a discussion of the conducted research described in this report. This section addresses five limitations of the developed model, the validity of the model's underlying assumptions and the validity of the results obtained from the developed model.

9.1 Limitations of the model

The first major limitation of the proposed methodology is that it searches for networks while only optimizing one criterion: minimizing the construction costs of the energy network's pipelines or power cables. In reality, there are more costs associated with the development of an energy network like the construction of gas compressors [86] or electricity transformers and more criteria to optimize like a network's reliability, robustness and resilience. As discussed in chapter 7, the cost-optimal offshore electricity grid discovered by the model differs quite a lot from the electricity grid already in place. This discrepancy can partly be assigned to this limitation, since the assignment of power cables' capacities to different landing points is not solely dependent on cost-minimization as in the model, but dependent on factors like onshore electricity demand and required construction methods as well.

A second limitation of the model is that it is only capable to design entirely new energy infrastructures. These 'greenfield' roll-outs of new networked infrastructures are rare in the developed world. Most networked infrastructures in use today have evolved over several decades, or even centuries, in an organic way. Over the years, small local networks have grown into national or international networks [35]. Due to this limitation, in the second case study that explored the North Sea's offshore electricity infrastructure, an assumption has been made that all of the wind farms had still to be connected, while in reality three smaller wind farms were already connected to the Dutch coast. Because the power cable infrastructure that was already in place was neglected, the investment costs estimated by the model will be higher than is realistic.

A third limitation is that the model is not able to assign negative weights to weighted regions, a problem that is elaborated on in appendix A. As a result, the model cannot allow for regions in which the constructor receives money to build an energy network. This limitation might have less practical implications than the previous two, since it is unlikely that a rational actor would pay for the disturbance caused by a pipeline or power cable on his or her land.

Another limitation of the model is that it requires that the total demand of the sink nodes is equal to the total supply of the source nodes. This limitation might be incompatible with the design of modern electricity networks that experience an increase of intermittency on the demand site due to the rise of sustainable electricity generation. To ensure that electricity networks can always meet the electricity demands of households and the industry, the total installed capacity of both sustainable and fossil electricity suppliers, must be higher than the total demand for electricity.

Lastly, the model only accepts weighted regions in the form of convex, concave or triangle-formed polygons. In nature and practice however, obstacles with different costs of constructing energy networks in them are not always made up out of straight lines. This makes the model less suitable for very detailed designs in which the precise route through an obstacle is required.

9.2 Validity of the case studies assumptions

One of the assumptions that was made while designing the second case study that shows the application of the model, was that the wind farms that will be developed on the Dutch North Sea will be connected solely to the Dutch shoreline and subsequently to the Dutch electricity grid. However in reality, the countries neighbouring the North Sea plan on connecting their North Sea energy systems in order to better deal with the intermittency inherent to the generation of wind energy. During production spikes, the energy can be allocated to other countries, while during a shortage of wind, energy can be bought from abroad. These plans are not just ambitions for the far future. The North Seas Energy Cooperation (NSEC) which was founded in 2016 by Ireland, the United Kingdom, Sweden, Norway, Denmark, Germany, the Netherlands, Belgium and Luxembourg, is already working on concrete cross-border network projects that have the potential to use economies of scale to reduce the costs for all countries. Two examples in which the Netherlands are involved are a WindConnecter that will connect the Dutch wind park IJmuiden Ver not only to the Netherlands, but to the United Kingdom as well and an energy hub that will allocate electricity generated by Danish, German and Dutch wind farms to these countries where it is needed the most [66]. Since there are already concrete plans to interconnect wind farms from different countries in the next decade, the assumption that the Dutch power cable infrastructure of 2050 will only be connected to the Dutch shoreline does not seem that realistic.

9.3 Validity of the case studies results

The case study used to show different applications of the model concerned connecting the wind park Doordewind with a planned capacity of 4 GW to two potential landing points in the northern part of the Netherlands. Since the method described in this research report assumes some level of economies of scale when constructing pipelines or power cables with a larger capacity, the results of all scenarios contained a single power cable of 4 GW that connected the wind park through different routes with one of the two landing points. In reality, the routing plans for connecting the wind farm have been finalized and the choice was made to connect the site with two power cables, each with a capacity of 2 GW [70, 86]. This is an interesting discrepancy between the outcome of the model and the observed decision-making process, for which the next chapter will make some suggestions fur further research.

Figures 6.2 to 6.4 show the model's designs of the Dutch North Sea's power cable infrastructure that connects all the current and potential offshore wind sites with the mainland. Interestingly, in all of the designed scenarios, the wind farms are connected to either the landing points of Eemshaven, Beverwijk or Borrsele, but never to the Maasvlakte landing point. Figure 5.7 presents the current onshore and offshore electricity grid, on which can be seen that the wind farms Dutch Coast South and even the IJmuiden Far wind site, which is much closer to the Beverwijk landing point, are connected to the Maasvlakte landing point. Both of these discrepancies between the model's outcomes and either the observed decision-making process or the existing power cable layout were not expected and therefore interesting. As a consequence, the following chapter will make some recommendations for further research that might overcome these discrepancies.

10. Recommendations

The last chapter of this report will give several recommendations to both policy makers who are part of the decision-making processes that realize energy infrastructures and researchers in the field of energy infrastructure modelling who would like to use this model for different research or expand on the knowledge gained in this research.

10.1 Policy recommendations

The main contribution of the model developed in this research is that it is able to generate cost-optimal energy networks in a context of spatial regions with different costs of constructing energy networks in them. Policy makers could use this model's contribution to their advantage in three ways.

First of all, the model is well suited to identify, quantify and visualize the trade-offs between the construction costs of energy networks on the one hand and their influence on their spatial context on the other. During different stages of the design process, discussions about the energy infrastructures' impact on nature, the possibility of obtaining land from other actors or the traversing of regions with a high cultural or societal relevance can now be assisted with numbers in a quick and easy manner. An example of an energy infrastructure project that might have benefited from the developed model is the Keystone Oil Pipeline which was planned to transport oil from the Canadian province of Alberta to refineries in Texas and Illinois in the United States. The oil pipeline became highly controversial because it crossed protected nature reserves as well as areas sacred to the indigenous Sioux population. Eventually, the permit for the last phase of the pipeline was revoked on the first day of office of president Joe Biden [78]. If the model would have been used to quantify the trade-offs between the proposed sensitive route and a longer, more delicate, route, in combination with a cost-benefit analysis that included the costs of potential harm to nature and political and cultural risk, a different route for the pipeline might have been chosen which could have resulted in a successful completion of the project.

A second application of the model for policy makers arises in projects including several geographical regions with varying costs of constructing energy networks in them. Finding the cost-optimal network becomes a complex task in these circumstances. In these situations, the model is able to represent reality better than other models known to the author.

Thirdly, the cost-optimal outcomes of the model can be used as a starting point in the decision-making processes of energy infrastructures. As mentioned in section 7.2, the design of energy networks is dependent on many factors of which the criterion of minimizing the initial investment costs is often an important one. Initializing the decision-making process with a cost-optimal network might assist policy makers in tweaking this first design to perform better on other evaluation criteria, while keeping the investment costs relatively low.

10.2 Other applications of the model

As discussed in the previous discussion chapter, it is anticipated that the collaboration between countries bordering the North Sea in developing their offshore energy infrastructures will enhance in the coming decades. Therefore, the developed model could provide a blueprint of a cost-optimal design of an offshore electricity infrastructure that connects all the to be constructed wind farms on the North Sea that potentially cross Natura 2000 areas which could be used to start discussions between the countries on the division of development costs between them and the utilization of the limited space the North Sea provides.

In addition, the model could also be used to research the two proposed cases of connecting wind farms on the North Sea in more depth. The (Dutch) North Sea is not simply split between protected Natura 2000 zones and areas suitable for the development of offshore wind parks. The crowded marine region is, among other things, also used for fishing, sand mining, shipping, military purposes and (digital) communication infrastructures [67]. Expanding the model's inputs by including these marine areas with other purposes as weighted regions, might ensue in results that are better aligned with reality.

10.3 Further research

A first direction for potential research can be found in improving the algorithm of Lanthier et al. [47] that was selected in section 2.3 to solve the weighted region shortest path problem. In a later paper, Lanthier et al. [48] proposed an interval scheme instead of the fixed scheme to place Steiner nodes on the edges of triangulated regions that can reduce the number of Steiner points placed on some edges considerably, thereby reducing the computational time of the model. Other researchers familiar with energy network modelling could also try to implement the other examined algorithms that solve the Minimum Weighted Region Shortest Path Problem also mentioned in section 2.3 in the described methodology to compare their performance and potentially point out the algorithm best suited to solve the described Minimum Cost Weighted Region Steiner Tree problem.

Another avenue for further research is to expand the developed model via existing connections. Most developed countries already have energy infrastructures in place that need to be expanded [32] or adapted [36] to provide energy to a greater amount of their populations or reach their sustainability goals and have therefore no need to design energy infrastructures from scratch. Integrating these existing connections as an optional input parameter, for example by using the method proposed by Heijnen [31], increases the application possibilities of the model to not only be limited to the design of 'green-field' infrastructures.

11. Bibliography

[1] 4COffshore. Offshore wind, 2022.

[2] L. Aleksandrov, M. Lanthier, A. Maheshwari, and J. R. Sack. An - Approximation algorithm for weighted shortest paths on polyhedral surfaces. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 1432, pages 11–22, 1998.

[3] L. Aleksandrov, A. Maheshwari, and J. R. Sack. An improved approximation algorithm for computing geometric shortest paths. In *International Symposium on Fundamentals of Computation Theory*, pages 246–257, Berlin, 2003. Springer.

[4] L. Aleksandrov, A. Maheshwari, and J. R. Sack. Determining approximate shortest paths on weighted polyhedral surfaces. *Journal of the ACM*, 52(1):25–53, 2005.

[5] J. André, S. Auray, J. Brac, D. De Wolf, G. Maisonnier, M. M. Ould-Sidi, and A. Simonnet. Design and dimensioning of hydrogen transmission pipeline networks. *European Journal of Operational Research*, 229:239–251, 2013.

[6] Barcelona Field Studies Centre. Spearman's Rank Correlation Coefficient, 2022.

[7] S. Baufumé, F. Grüger, T. Grube, D. Krieg, J. Linssen, M. Weber, J. F. Hake, and D. Stolten. GIS-based scenario calculations for a nationwide German hydrogen pipeline infrastructure. *International Journal of Hydrogen Energy*, 38(10):3813–3829, 2013.

[8] R. E. Bloomfield, P. Popov, K. Salako, V. Stankovic, and D. Wright. Preliminary interdependency analysis: An approach to support critical-infrastructure risk-assessment. *Reliability Engineering and System Safety*, 167(July 2015):198–217, 2017.

[9] M. Bóna. *A Walk Through Combinatorics; An Introduction to Enumeration and Graph Theory.* World Scientific Publishing Co. PTe. Ltd., London, fourth edi edition, 2017.

[10] J. Brimberg, P. Hansen, K. W. Lih, N. Mladenović, and M. Breton. An oil pipeline design problem. *Operations Research*, 51(2), 2003.

[11] J. Carson. Model Verification and Validation. In *Proceedings of the winter simulation conference*, pages 52–58. IEEE, 2002.

[12] B. Chandra Mohan and R. Baskaran. A survey: Ant Colony Optimization based recent research and implementation on several engineering domain. *Expert Systems with Applications*, 39(4):4618–4627, 2012.

[13] D. Coleman. Lee's O(n 2 log n) Visibility Graph Algorithm Implementation and Analysis. 2012.

[14] M. De Berg, O. Cheong, M. Van Kreveld, and M. Overmars. *Computational Geometry: Algorithms and Applications.* Springer, Berlin Heidelberg, 3rd edition, 2007.

[15] J. L. De Carufel, C. Grimm, A. Maheshwari, M. Owen, and M. Smid. A note on the unsolvability of the weighted region shortest path problem. *Computational Geometry: Theory and Applications*, 47(7):724–727, 2014.

[16] S. De-León Almaraz, C. Azzaro-Pantel, L. Montastruc, L. Pibouleau, and O. B. Senties. Assessment of mono and multi-objective optimization to design a hydrogen supply chain. *International Journal of Hydrogen Energy*, 38(33):14121–14145, 2013.

[17] B. Delaunay. Sur la sphere vide. *Izv. Akad. Nauk SSSR, Otdelenie Matematicheskii i Estestvennyka Nauk*, 7:793–800, 1934.

[18] M. E. Demir and I. Dincer. Cost assessment and evaluation of various hydrogen delivery scenarios. *International Journal of Hydrogen Energy*, 43(22):10420–10430, 2018.

[19] E. Elburrz. WTF is Convex Decomposition, 2019.

[20] L. Euler. Solutio problematis ad geometriam situs pertinentis. *Commentarii academiae scientiarum Petropolitanae*, pages 128–140, 1734.

[21] European Commission. Natura 2000, 2008.

[22] European Environment Agency. Germany, 2020.

[23] European Environment Agency. Natura 2000 Network (terrestrial and marine areas), 2020.

[24] J. Fernández, L. Cánovas, and B. Pelegrín. Algorithms for the decomposition of a polygon into convex polygons. *European Journal of Operational Research*, 121:330–342, 2000.

[25] J. Fernández, B. Tóth, L. Cánovas, and B. Pelegrín. A practical algorithm for decomposing polygonal domains into convex polygons by diagonals. *Top*, 16:367–387, 2008.

[26] M. Gendreau, J. F. Larochelle, and B. Sansò. A tabu search heuristic for the Steiner Tree Problem. *Networks: An International Journal*, 34(2):162–172, 1999.

[27] Google. Google Earth, 2022.

[28] J. Gorenstein Dedecca, S. Lumbreras, A. Ramos, R. A. Hakvoort, and P. M. Herder. Expansion planning of the North Sea offshore grid: Simulation of integrated governance constraints. *Energy Economics*, 72:376–392, 2018.

[29] R. L. Graham and P. Hell. On the History of the Minimum Spanning Tree Problem. *Annals of the History of Computing*, 7(1):43–57, 1985.

[30] M. Guo and N. Shah. Bringing Non-energy Systems into a Bioenergy Value Chain Optimization Framework. *Computer Aided Chemical Engineering*, 37(June):2351–2356, 2015.

[31] P. Heijnen. sinks and multi-sources, 2020.

[32] P. Heijnen, E. Chappin, and I. Nikolic. Infrastructure network design with a multi-model approach: Comparing geometric graph theory with an agent-based implementation of an ant colony optimization. *Jasss*, 17(4):1–14, 2014.

[33] P. W. Heijnen, E. J. Chappin, and P. M. Herder. A method for designing minimum-cost multisource multisink network layouts. *Systems Engineering*, 23:14–35, 2020.

[34] P. W. Heijnen, A. Ligtvoet, R. M. Stikkelman, and P. M. Herder. Maximising the Worth of Nascent Networks. *Networks and Spatial Economics*, 14(1):27–46, 2014.

[35] P. W. Heijnen, R. M. Stikkelman, A. Ligtvoet, and P. M. Herder. Using Gilbert networks to reveal uncertainty in the planning of multi-user infrastructures. In *2011 International Conference on Networking, Sensing and Control*, pages 371–376. IEEE, 2011.

[36] R. Huisman. *Towards a robust European hydrogen network.* Master thesis, TU Delft, 2021.

[37] IEA. Technology Roadmap: Hydrogen and Fuel Cells. Technical report, 2015.

[38] IEA. World Energy Outlook 2021. Technical report, 2021.

[39] IPCC. Climate Change 2021: The Physical Science Basis. Contribution of Working Group I to the Sixth Assessment Report of the Intergovernmental Panel on Climate Change. Technical report, 2021.

[40] K. Joshi. *Foundations of Discrete Mathematics.* New Age International (P) Limited, Publishers, New Delhi, 1989.

[41] J. Kennedy and R. Eberhart. Particle Swarm Optimization. In *Proceedings of ICNN'95-international conference on neural networks*, pages 1942–1948. IEEE, 1995.

[42] J. Kitzinger. *The Visibility Graph Among Polygonal Obstacles : a Comparison of Algorithms.* Doctoral dissertation, University of New Mexico, 2003.

[43] M. M. Knoope, W. Guijt, A. Ramírez, and A. P. Faaij. Improved cost models for optimizing CO2 pipeline configuration for point-to-point pipelines and simple networks. *International Journal of Greenhouse Gas Control*, 22:25–46, 2014.

[44] A. Kolazhikkaran. Bergen zonder bomen pijpleiding, 2022.

[45] L. Kou, G. Markowsky, and L. Berman. A Fast Algorithm for Steiner Trees. *Acta Informatica*, 15:141–145, 1981.

[46] J. Kruskal. On the Shortest Spanning Subtree of a Graph and the Traveling Salesman Problem. *American Mathematical Society*, 7(1):48–50, 1956.

[47] M. Lanthier, A. Maheshwari, and J. R. Sack. Approximating shortest paths on weighted polyhedral surfaces. *Computational Geometry*, pages 274–283, 1997.

[48] M. Lanthier, A. Maheshwari, and J. R. Sack. Approximating Shortest Paths on Weighted Polyhedral Surfaces. *Algorithmica*, 30:527–562, 2001.

[49] Leard Statistics. Independent t-test for two samples, 2022.

[50] J. Linders, R. Segers, M. Van Middelkoop, S. Brummelkamp, K. Keller, G. Muller, and S. Houweling. Hernieuwbare Energie in Nederland 2020, Windenergie, 2021.

[51] Q. Liu, L. Mao, and F. Li. An intelligent optimization method for oil-gas gathering and transportation pipeline network layout. In *Proceedings of the 28th Chinese Control and Decision Conference, CCDC 2016*, pages 4621–4626. IEEE, 2016.

[52] A. Madkour, W. G. Aref, F. U. Rehman, M. A. Rahman, and S. Basalamah. A Survey of Shortest-Path Algorithms. 2017.

[53] R. Malischek and S. McCulloch. CCS: Identifying the best underground locations for storing CO2 can take 10 years, so let's start now, 2021.

[54] C. S. Mata and J. S. B. Mitchell. A New Algorithm for Computing Shortest Paths in Weighted Planar Subdivisions. *Computational Geometry*, pages 264–273, 1997.

[55] Y. G. Melese, P. W. Heijnen, and R. M. Stikkelman. Designing networked energy infrastructures with architectural flexibility. *Procedia Computer Science*, 28:179–186, 2014.

[56] J. S. Mitchell and C. H. Papadimitriou. The Weighted Region Problem: Finding Shortest Paths Through a Weighted Planar Subdivision. *Journal of the ACM (JACM)*, 38(1):18–73, 1991.

[57] V. Q. Nguyen. *Non-deterministic networked infrastructure design of multiple sources and multiple sinks.* PhD thesis, Delft University of Technology, 2015.

[58] Noordzeeloket. Protected sites, 2022.

[59] J. Pach. The beginnings of geometric graph theory. In *Erdős Centennial*, pages 465–484. Springer, Berlin, 2013.

[60] R. C. Prim. Shortest Connection Networks And Some Generalizations. *Bell System Technical Journal*, 36(6):1389–1401, 1957.

[61] D. Quammen. *The Song of the Dodo: island biogeography in an age of extinctions.* Simon Schuster, New York, 1997.

[62] S. Rebay. Efficient Unstructured Mesh Generation by Means of Delaunay Triangulation and Bowyer-Watson Algorithm. *Journal of Computational Physics*, 106:125–138, 1993.

[63] J. Reif and Z. Sun. An efficient approximation algorithm for weighted region shortest path problem. In *Proceedings of the 4th Workshop on Algorithmic Foundations of Robotics*, pages 191–203, 2000.

[64] C. Reksten-Monsen. Distance Tables Part 2: Lee's Visibility Graph Algorithm, 2016.

[65] M. Reuß, T. Grube, M. Robinius, and D. Stolten. A hydrogen supply chain with spatial resolution: Comparative analysis of infrastructure technologies in Germany. *Applied Energy*, 247:438–453, 2019.

[66] Rijksoverheid. Wind op zee na 2030, 2022.

[67] Rijksoverheid. Wind op zee rond 2030, 2022.

[68] B. Rothfarb, H. Frank, D. Kleitman, D. Rosenbaum, and K. Steiglitz. Optimal Design of Offshore Natural-Gas Pipeline Systems. *Operations Research*, 18(6):992–1020, 1970.

[69] S. Roy, S. Das, and S. Nandy. A practical algorithm for approximating shortest weighted path between a pair of points on polyhedral surface. In *International Conference on Computational Science and Its Applications*, pages 42–52, Berl, 2004. Springer.

[70] Royal HaskoningDHV. Rapportage Onderzoek Innovatie Doorkruising Waddengebied. Technical report, 2021.

[71] C. Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5):206–215, 2019.

[72] SClco. Delaunay Triangulation, 2020.

[73] R. Sedgewick and K. Wayne. *Algorithms*. Addison Wesley, Boston, fourth edi edition, 2011.

[74] R. Singh, R. Kataria, and S. Singhal. Decision-Making in Real-Life Industrial Environment through Graph Theory Approach. In *Computer Architecture in Industrial, Biomechanical and Biomedical Engineering*, pages 119–131. 2019.

[75] H. Takahashi and H. Matsuyama. An approximate solution for the Steiner problem in graphs. *Math, Japan*, 24:573–577, 1980.

[76] Tennet. Netkaarten, 2022.

[77] B. K. Thapalia, T. G. Crainic, M. Kaut, and S. W. Wallace. Single-commodity network design with stochastic demand and multiple sources and sinks. *INFOR: Information Systems and Operational Research*, 49(3):193–211, 2011.

[78] Trouw. Trump-regering in beroep tegen stillegging van omstreden oliepijpleiding, 2019.

[79] R. Trudeau. *Introduction to graph theory*. Courier Corporation, 2013.

[80] United States Geological Survey. How are UTM coordinates measured on USGS topographic maps?, 2022.

[81] S. V. Vasan. *Basics of Photonics and Optics*. Srini Vasan, Albuquerque, 2004.

[82] J. Verschuuren. Over kokkels , hamsters en woelmuizen: de Vogel- en Habitat- richtlijn in Nederland. *Nederlands tijdschrift voor Europees recht*, 4:85–94, 2001.

[83] P. Winter and J. MacGregor Smith. Path-Distance Heuristics for the Steiner Problem in Undirected Networks. *Algorithmica*, 7:309–327, 1992.

[84] C. Yang and J. Ogden. Determining the lowest-cost hydrogen delivery mode. *International Journal of Hydrogen Energy*, 32(2):268–286, 2007.

[85] C. Yeates, C. Schmidt-Hattenberger, W. Weinzierl, and D. Bruhn. Comparison of heuristic methods for achieving minimum-cost capacitated networks with a new metaheuristic based on node valency. pages 1–21, 2020.

[86] D. Yeşilgöz-Zegerius. Kamerbrief over verkenning aanlanding wind op zee 2030. Technical report, Ministry of Economic Affairs and Climate Policy, 2021.

[87] S. A. Zarghami, I. Gunawan, and F. Schultmann. The Application of Graph Theory to Public Infrastructure Asset Management. In *World Congress on Resilience, Reliability and Asset Management*, number November, pages 160–163, 2019.

A. The problem with assigning negative weights to weighted regions

This appendix demonstrates why assigning negative weights to weighted regions leads to problems when searching for shortest paths. Figure A.1 shows an example of an underlying graph in which the minimum Steiner tree algorithms described in section 3.4 could look for a minimum length spanning tree. A minimum length spanning tree in this example would connect source node A with sink nodes B, C and D while possible connecting the Steiner nodes of the obstacle depicted with the grey nodes. Note that the weights assigned to the edges are for an illustration of the problem and do not exactly match their relative length. Moreover, some of the edge weights of the triangulated obstacle are not shown in the figure to keep the illustration clear.

Both the Minimum Path Heuristic and the Distance Network Heuristic described in section 3.4 will have no difficulty in connecting the nodes A, C and D by picking either e_{AC} and e_{AD}, e_{AC} and e_{CD} or e_{CD} and e_{AD}. The problem arises when the shortest path needs to be found from node A to node B, this is when the used shortest path algorithm (Dijkstra's) encounters negative cycles. In an undirected graph, a negative cycle is a cycle whose edges all have a negative weight. The concept of a shortest path becomes meaningless if it encounters a negative cycle on one of the possible routes that connect two nodes [73].

One of the negative cycles that can be encountered between nodes A and B is the cycle [6,2,4] whose edges all have a weight of -1. As a result, every time the path chooses this route, the length of the path decreases by 3. This cycle will constantly be 'walked through' by Dijkstra's algorithm, thereby never ending and never giving a result.

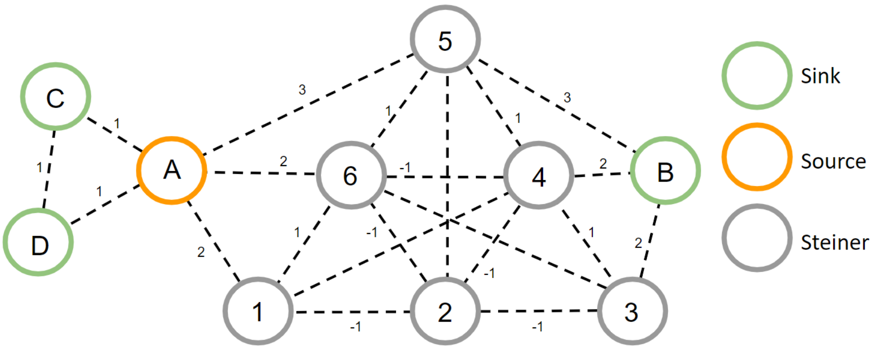


Figure A.1: Steiner Minimum Tree problem with a weighted region with a negative weight

Please note that finding the shortest path by using a detour, namely by looking for the longest path by first multiplying all the edge weights with -1 is not an option. This creates new negative cycles, like [A,5,6] for example, which will still 'break' the used algorithm.

B. Algorithms used in the model

This appendix dives deeper into some of the algorithms used in the methodology described in chapter 3 and were only briefly discussed in the research's main text.

B.1 Decomposition algorithm

This section describes the algorithm proposed by Fernández et al. [24] that can be used to decompose convex polygons. Convex decomposition is the process of splitting up a single concave polygon into several convex polygons [19, 25].

This procedure to generate a convex polygon out of a concave polygon works as follows. The algorithm runs through the vertices $v_1, v_2, ..., v_n$ of concave polygon P in clockwise order. The vertices of the next future convex polygon to be cut off are stored in a list L, initially with only one vertex. The adjacent vertices of P to this initial vertex are added to this list (in clockwise order) as long as it is possible to generate a convex polygon from the vertices (thus as long as there will be no interior angles larger than 180° when connecting the first vertex with the last vertex of the list). When this is no longer possible, the convex polygon to be cut off is obtained by adding a diagonal edge, joining the last and first vertices v of L [24]. This process is repeated with the remainder of the polygon until it is no longer concave.

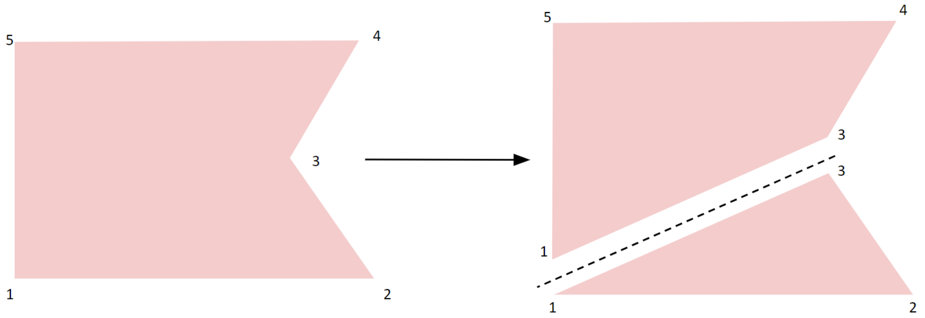


Figure B.1: Convex decomposition of a concave polygon

Figure B.1 shows the convex decomposition of the concave polygon that was introduced in figure 3.1. The decomposition algorithm starts with vertex n_1 which it added to the list L before continuing clockwise to the other vertices. The algorithm stops adding nodes to the list at vertex v_4, since connecting $v1$ to $v4$ would create a polygon with an interior angle larger than 180°, namely $\angle 234$. The convex polygon with the vertices from list $L = [v_1, v_2, v_3]$ is cut off from the initial polygon. Since the remainder of the polygon is now convex as well (polygon P with vertices $[v_1, v_3, v_4, v_5]$ has no interior angles larger than 180°), the algorithm is brought to an end.

B.2 Bowyer-Watson algorithm

The Bowyer-Watson algorithm is used to compute Delaunay Triangulations. A Delaunay Triangulation is a set of triangles made from a set of vertices so that no vertex lies inside the circumcircle of any other triangle in the set [17]. Figure B.2 shows a circumcircle of a triangle that, by definition, passes through all the triangle's vertices.

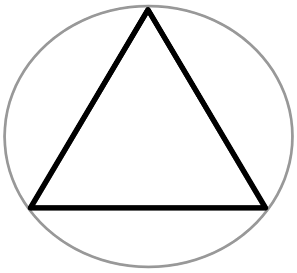


Figure B.2: A triangle with its circumcircle

The Bowyer-Watson algorithm initializes by drawing a 'super-triangle' that contains all the points that one wants to triangulate. One by one, consider each point inside the larger 'super-triangle'. For each of these points, save all the triangles whose circumcircles contain the point, which we will call 'bad triangles'. If any, remove all the edges that are shared by at least two 'bad triangles'. For every edge belonging to a 'bad triangle' that was not removed, form a triangle by connecting the point to the two vertices adjacent to the edge. These steps are repeated until there are no more 'bad triangles' after which the vertices of the 'super triangle' and all edges adjacent to them are removed [62].

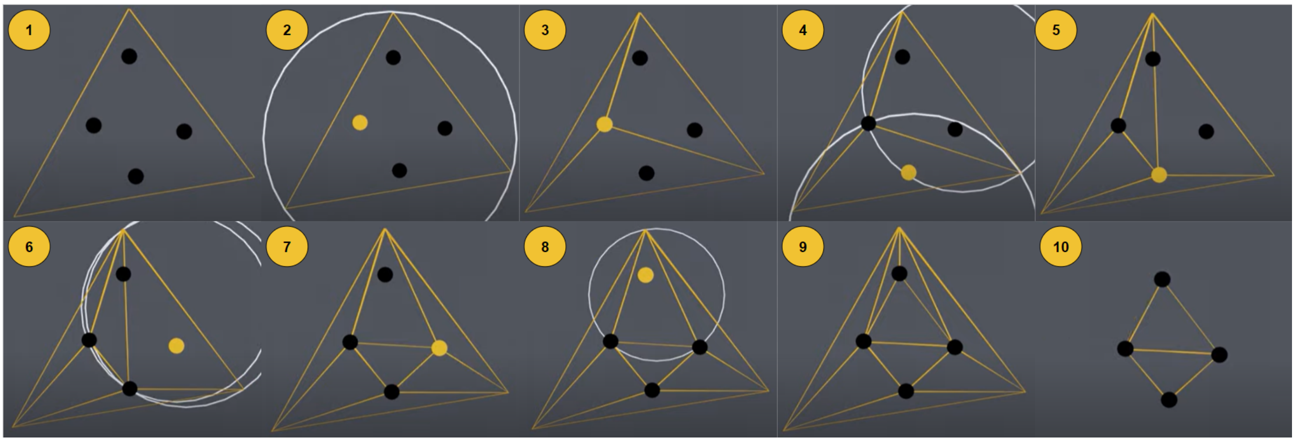


Figure B.3: A Delaunay Triangulation by using the Bowyer-Watson algorithm [72]

C. Verification data

The graph used in the verification process described in section 4.1 used the example shown in figure C.1 to assure that the model performs as it is supposed to. This appendix presents the calculations that underlie the assumptions of the model and thereby proves that the model's outcomes are accurate.

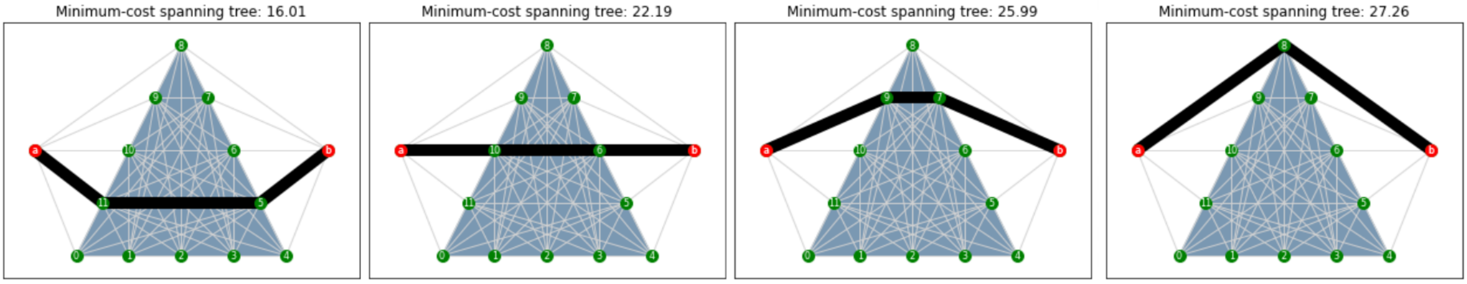

Minimum-cost spanning tree: 16.01
Minimum-cost spanning tree: 22.19
Minimum-cost spanning tree: 25.99
Minimum-cost spanning tree: 27.26

Figure C.1: Model runs with region weights $f_c = [0.25, 1, 1.5, 2]$

The costs of the spanning trees, represented by the black lines in the figure above, are calculated with the formula shown in equation 3.4. To calculate the total costs we therefore need the length of the edges l_e, which can be calculated by using the nodes' coordinates presented in table C.1, the capacity of the edges q_e, the capacity cost exponent β, which is 0.6 and the weight of the region f_{c_e} which is varied in the different model runs as is shown in the caption of figure C.1.

Table C.1: Nodes' coordinates of verification example

Node	a	b	0	1	2	3	4	5	6	7	8	9	10	11
X	-0.2	1.2	0	0.25	0.5	0.75	1	0.875	0.75	0.625	0.5	0.375	0.25	0.125
Y	0.5	0.5	0	0	0	0	0	0.25	0.5	0.75	1	0.75	5	0.25

Assuming that the region depicted in blue has a positive face, there are four routes from terminal a to terminal b that are in the running for becoming the cost-optimal route. These routes are $[a, 8, b]$, $[a, 9, 7, b]$, $[a, 10, 6, b]$ and $[a, 11, 5, b]$. Note that route $[a, 0, 1, 2, 3, 4, b]$ will never be less costly than route $[a, 11, 5, b]$, since the combined length of edges e_{a0} and e_{4b} is larger than the combined length of edges e_{a11} and e_{5b}, all the edges have the same capacity and do not cross a weighted region. Therefore it does not matter if the path from Steiner point 0 to 4 is free of costs, the other paths will be cheaper in all scenarios wherein the region has a positive weight.

Table C.2 shows the total costs of the different paths from a to b under variations of the region's weight f_c. The lighter the colour of the cell, the cheaper the path is compared to the other potential paths. The paths that traverse the weighted region f_c less, or not at all, become more attractive as the region weight increases.

Table C.2: The costs of paths under different region weights

Region weight Paths	0.25	1	1.5	2
$[a, 8, b]$	27.26	27.26	27.26	27.26
$[a, 9, 7, b]$	20.96	23.93	25.99	27.89
$[a, 10, 6, b$	16.25	22.19	26.15	30.11
$[a, 11, 5, b]$	16.01	24.88	30.83	36.77

.

D. Offshore wind park data

This appendix provides the data of the offshore wind parks that were used to model the offshore wind park cases described in chapter 4 and which results were presented in chapter 5. Table D.1 shows the UTM coordinates, (potential) capacity and (planned) construction year of the 15 wind farm sites in the North Sea and the 4 landing points on the Dutch shoreline.

Table D.1: Offshore windpark data

Code	**Name**	**Capacity (MW)**	**UTM Coordinates (31N)**	**Construction year**
NW1	Borssele wind farm	1,502	(500000, 5714697)	2021
NW2	Dutch Coast (South)	1,649	(576031, 5802408)	2023
NW3	Dutch Coast (North)	879	(590118, 5828643)	2024
NW4	Dutch Coast (West)	2,100	(547400, 5830000)	2026
NW5	North of the Frisian Islands	1,300	(694032, 5991602)	2026
NW6	Ijmuiden Far	6,000	(536000, 5854000)	2028
NW7	Nederwiek	6,000	(515000, 5887000)	2029
NW8	Doordewind	4,000	(670500, 6017250)	2030
NW9	Lagelander	4,000	(561400, 5905500)	>2030
NW10	Search Area 3	2,000	(550000, 5970000)	>2030
NW11	Search Area 4	10,000	(680000, 5970500)	>2030
NW12	Search Area 5	2,000	(641150, 6023000)	>2030
NW13	Search Area 6	10,000	(611000, 6027500)	>2030
NW14	Search Area 7	8,000	(580500, 6020000)	>2030
NW15	Search Area 8	2,000	(594500, 5919000)	>2030
NA1	Eemshaven	-	(757600, 5928500)	-
NA2	Beverwijk	-	(607700, 5817000)	-
NA3	Maasvlakte	-	(571200, 5759500)	-
NA4	Borssele landing point	-	(539700, 5699500)	-

.