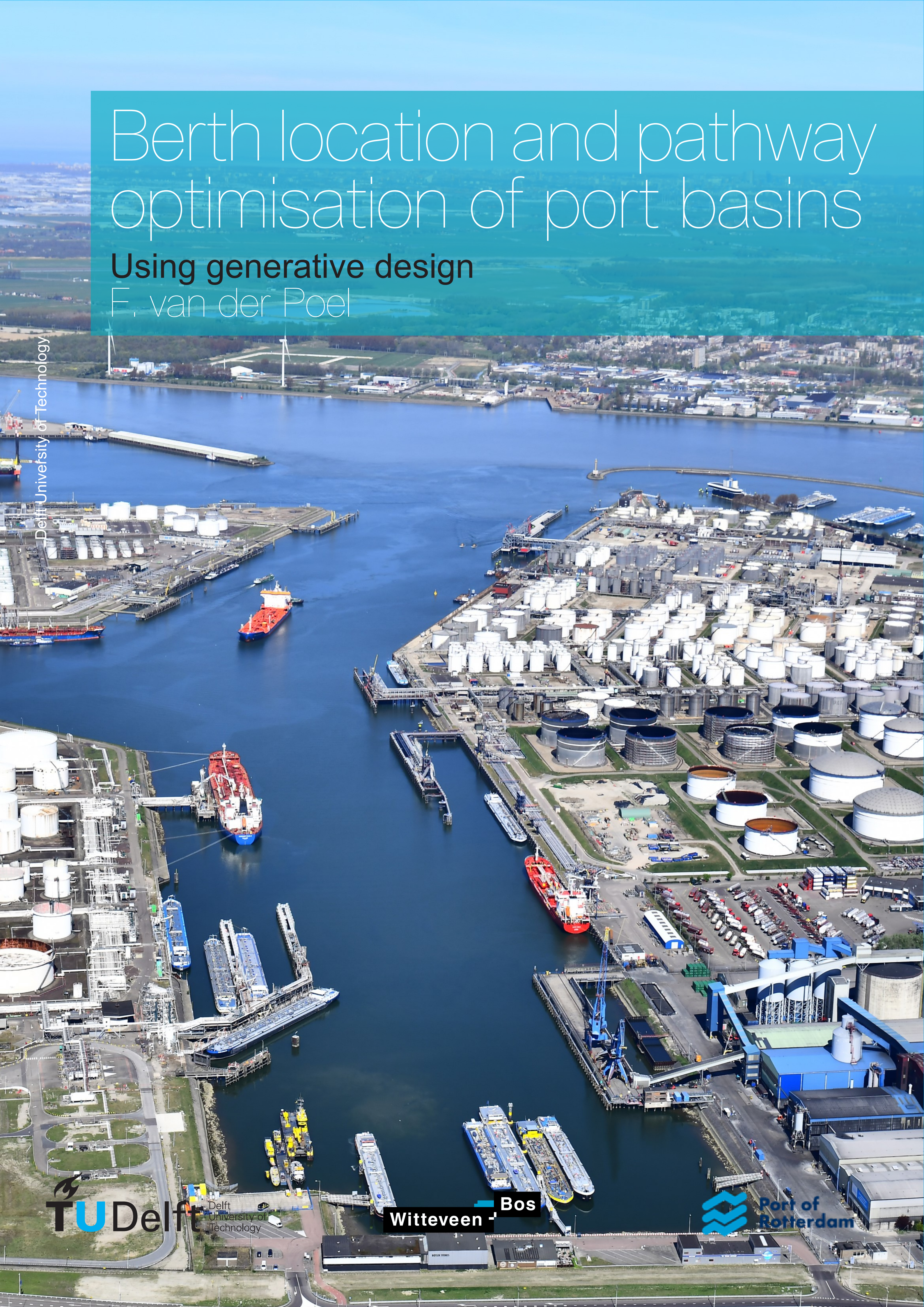
Berth location and pathway optimisation of port basins

Using generative design

F. van der Poel



Delft University of Technology

Berth location and pathway optimisation of port basins

Using generative design

by

F. van der Poel

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Friday September 22, 2023 at 14:00 PM.

Student number:	4607678	
Thesis committee:	Prof. Dr. Ir. M. van Koningsveld,	Delft University of Technology, chairman
	Dr. Ir. A. Roubos,	Delft University of Technology, assessor
	Dr. Ir. P. Taneja,	Delft University of Technology, assessor
	Ing. M. Omar,	Port of Rotterdam Authority, mentor
	Ing. E. Slump,	Witteveen+Bos, mentor
	Dr. P. Nourian,	Twente University, mentor

TU Delft

Preface

Picking the easy route might be the comfortable way to go, but the long-term joy and satisfaction is in the tough road. Most teachers and fellow students suggest picking a thesis subject that is fun and sparks your interest. The result is me picking this thesis topic; applying generative design on a small part of the port planning discipline; berth layouts of port basins. Not knowing where this research would go, I started investigating and learning about interesting topics, which I would never imagined learning about during my MSc Civil Engineering; complex mathematical topics as mathematical programming, digital geometry, graph theory, and mathematical morphology. And looking back at the research, these complex topics made the ride fun and interesting.

As for the people involved, I want to thank them. First of all, I want to thank my professor Mark van Koningsveld for our animated discussions, his critical view, and his willingness to help with my thesis. From the Port of Rotterdam, I want to thank Alfred Roubos for his time, support, and ability to help put problems occurring during my thesis into perspective. Also from the Port of Rotterdam, I want to thank Mo Omar for his daily supervision, enthusiasm, and availability in case help was needed. From Witteveen+Bos I want to thank Elmo Slump, who was also my daily supervisor during my Witteveen+bos internship. His enthusiasm, port planning experience, and interest in innovations were of great value to this project. I also want to thank the companies of Port of Rotterdam Authority and Witteveen+Bos for providing me with the resources to help me produce this thesis.

From my committee, I also want to thank Poonam Taneja for helping me in terms of academic writing. I also want to thank Pirouz Nourian for his time and great knowledge of generative design. He gave invaluable advice regarding setting up an environment that could be used for generative design.

Finally, I want to thank my friends and family for supporting me and being part of my student days and life in general.

Fabian van der Poel, 22 September 2023, Delft

Abstract

In the early project phase, decisions are made when considering a port basin berth layout that affects the outcome and the success of the project. In order to make a decision regarding the berth layout, port planners create a set of variants for comparison and assessment in order to pick the basin berth layout that suits the project goals the best. The selected basin berth layout is the preferred variant that will be used to eventually be realised. However, since the decision in the early project phase has a great influence on the entire project, port planners want to improve the decision-making for selecting the best berth layout. This research aims to improve the decision-making process for selecting the berth layout of preference by developing a design generation method for berth layouts and its respective pathway. This method might give insights into optimal berth layout configurations and its pathway in order for decision-makers to assess their preferred variant resulting from the set of variants developed by experts and the generated layout. After the method is developed, the method is tested on conceptual benchmark cases and a real case in order to assess whether such method can add value, or improve, design solutions of port basin layouts during the early project phase. Although limited studies and methods exist that have resemblance or commonalities with berth layout optimisation, the idea is in its infancy.

The concept studied is generative design; a systematic approach for the exploration of layouts, shapes, and networks (Nourian, 2023). The systematic aspect of generative design in this research is performed by means of a method that is suitable for a computational algorithm. In order to generate optimal berth layouts, mathematical programming is used. The literature study regarding basin layouts assessed that constraints regarding berth distancing and location restrictions for berths are important. Hence, the developed optimisation method has to take these types of constraints into account. The objective considered in this research is dredging volume, since locating a berth, and forming a pathway to the berth, may require dredging, which is aimed to be minimized. The decision to be made are the location of the berths and the depths for the basin bottom that allows for vessels to traverse through the port basin.

Methods for berth layout optimization have been developed to generate berth layouts and its respective pathway through the basin in this report. The optimization is separated into two steps: firstly, berth location optimization using a pre-defined central basin pathway, and secondly, optimization of the pathway belonging to the respective berth layout. The separation of the berth location and pathway optimisation reduces computational time. The heuristic berth location optimization optimizes the locations of the berths to minimize the central pathway dredging, while satisfying constraints prescribing a minimal distance between the berths. After the berth locations are optimized, the pathways can either be optimized by means of a heuristic pathway approach, or by means of an exact mathematical programming approach, both of which are developed in this research. The heuristic pathway optimization approach models required dredging by translating the dredging for a berth into a graph with weighted edges and then sequentially defining the route that requires the least dredging per berth, eventually forming the complete basin pathway. The exact pathway approach uses a mathematical programming setup. The methods are applied to conceptual benchmark cases a real-life case. The benchmarks differ in basin dimensions, basin bottom profile, and fleet characteristics. The real-life case concerned the Scheurhaven, which is a small port basin for mainly tugboats. The layout of the Scheurhaven had to be adapted to fit more tugboats in the basin. The Scheurhaven case was optimised with the heuristic approach, due to the potentially large computational time of the exact approach. As the berth locations were limited and the method lacked features to make the design comparable, such as structures and berth geometry, only heuristic pathway optimisation was done.

The heuristic pathway optimization then optimizes the pathway, which improves the dredging objective compared to the assumed central pathway used for the berth location optimisation, by requiring less dredging. However, due to the nature of the sequential pathway finding and the lack of consider-ance of slope stability when forming a pathway as a result of pathway dredging, the resulting pathway is less credible as a pathway for minimal dredging. The exact pathway approach does take both problems into account and also improve the objective value of the central pathway in all benchmark cases compared to the central pathway. The drawback of the exact approach is the computational time it takes to converge to a solution. It can take up to 3 hours for objective values to stabilize. However, the exact pathway approaches the real optimal pathway and, hence, can give insights regarding decision-making for basin pathways. The results for the Scheurhaven case were that the pathway has to remain as close to the berths as possible and to combine the pathways going from each berth to the exit. However, the case results are to be taken with a grain of salt, because it lacked comparable basin layout features used in the real case.

In conclusion, the heuristic berth optimisation and the exact pathway optimisation can provide insight into berth layout optimality and a respective optimal pathway. However, the result of the method is not a variant that can be used for consideration in picking the best variant out of a bunch to select and realise. The generated optimal berth layout and pathway can be used for decision-makers to verify that expert-developed variants roughly comply with the generated layout. In order to improve the decision support for port planners, it's suggested to further extend the exact pathway optimisation to a berth location optimisation and to study implementation of berth geometry and structures. This research didn't manage to implement these features due to the complex nature of digital geometry and mathematical programming. Structures in basin layout design are geometric elements, like berths, and due to the selection to model the environment in discrete space, the problem required knowledge of digital geometry, which is a stretch from the common analytical mathematics in civil engineering.

Contents

1 Introduction

1.1. Motivation and societal relevance

Port planning is a discipline that translates a port authority's strategic goals to the physical environment. A port master plan is a broad plan which describes all relevant factors for realizing a port area as well as keeping the port area operational. In short, a Port Master Plan outlines the objectives of the port and how they are expected to be achieved (Taneja et al., 2010). A port master plan ranges from strategic and project planning to port design and operation and maintenance (van Koningsveld et al., 2021). The outcome of the strategic master plan is a go/no go decision to realize the strategic goals. After the go/no go decision, the project master plan, which can include multiple projects, is made. The project master plan generally includes (van Koningsveld et al., 2021):

- the strategic goal: what exactly one aims to achieve with the project;
- the relevant data (site conditions, cargo and traffic forecasts, vessel mix, etc.);
- specific functional requirements (service level, embedding in the network, etc.);
- a basic design, making it possible to investigate the project's feasibility;
- financial and economic feasibility studies, including a risk analysis;
- environmental aspects, not only in the form of an Environmental Impact Assessment (EIA), but also considering the possibilities of 'Building with Nature';
- social aspects, often in the form of a Social Impact Analysis (SIA), or a Societal Cost Benefit Analysis (SCBA);
- safety and security;
- management structure;
- type of contracting (construct-only, design and construct, design-finance-construct, design-construct-maintain, etc.).

After the project master plan is made, design activities for the port are done to eventually fulfill the master plan. Typical steps in the development of a port, or elements within an existing port, involve pre-feasibility studies, aiming to explore the initial idea and to develop a port strategy, and feasibility studies, which concludes whether a project is feasible and steps toward implementation can be started (van Koningsveld et al., 2021). The focus of the research is on the feasibility studies of spatial port layout design. A flow chart of the feasibility study is depicted in Figure 1.1. The spatial layout can be grouped into 4 categories of spatial needs (PIANC, 2019):

1. Marine areas and depths: navigation channel and maneuvering basins
2. Shelter: breakwaters and coastal protection
3. Land areas and access operational areas: (a) cargo handling, storage, and processing and (b) support activities and access (road, rail networks, offices, workshops, etc.)
4. Environmental mitigation and compensation areas

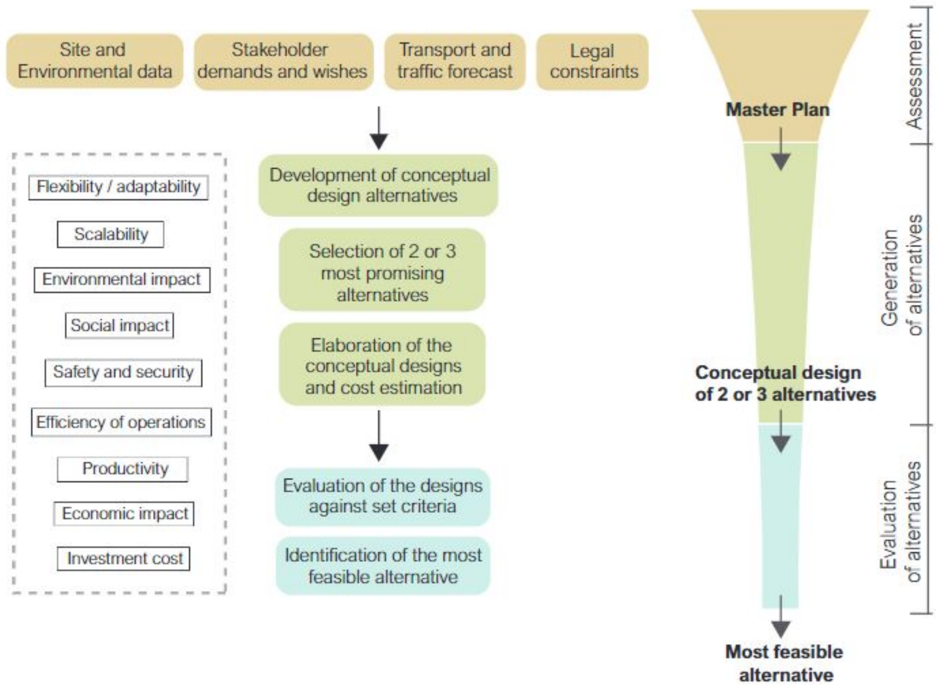


Figure 1.1: Flow chart of a port feasibility study (van Koningsveld et al., 2021)

From the mentioned categories, the Port of Rotterdam Authority is interested in the Marine area. Specifically, the interest is in the berth layouts in the port basin, leaving out other main components of marine areas such as outer channel approach, anchorage areas, shelter, harbour entrance, turning basin, construction facilities, etc. (PIANC, 2019). Since the berth layout is considered in the feasibility study, which can be referred to as the early project phase, and that influence in the early project phase for these infrastructures is vast, design decisions in this phase impact further project implications, as depicted in Figure 1.2.

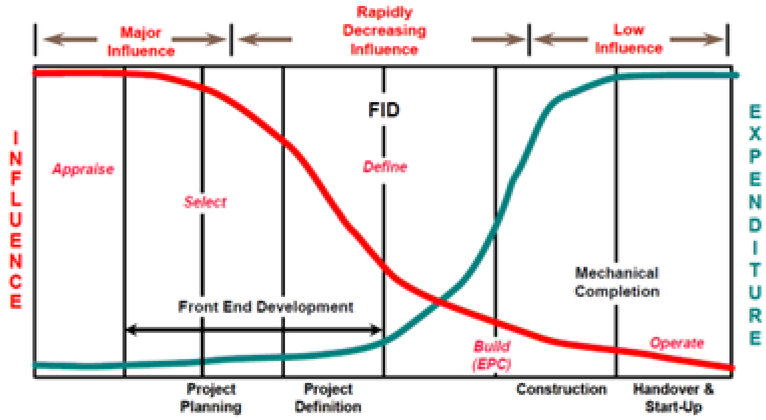


Figure 1.2: Graph of influence over the project lifetime (Bakker, 2008)

Making the right decision for the project requires developing and comparing variants. However, in reality, the variants studied are limited to a small number. This is also the case for the Scheurhaven project, which is a case study for this research. After doing an inventory of the design process with a port planner of the Port of Rotterdam, it was found that the Scheurhaven project had 5 developed variants for the layout over a period of multiple months of which substantial time was spent developing variants (see Appendix C). After evaluation, the preferred variant is picked, known as the 'voorkeursvariant'. The aim is to research the possibility for systematically generated berth layout designs to find a variant that provides support in decision-making for selecting the preferred berth layout variant.

1.2. Literature review

Although generative design to optimise spatial layouts will be researched, numerous examples exist of optimisation in a port engineering context. One of the known problems is the berth allocation problem (BAP). One of the first formulations of the BAP came from Imai et al. (1997) and describes the problem as the trade-off between efficient utilization of public container terminals and dissatisfaction of vessels with regard to waiting time due to their berthing order. The main dissatisfaction comes from waiting time and berth time for vessels inside the port. The assumptions made for the problem formulation already reduces the scope to allocation to existing berth locations. This means that the spatial optimisation is done upfront, or assumed constant, and the problem concerns the scheduling of the vessel at the quay walls. Current research takes spatial constraints little into account and focuses on solving the problem using other solving techniques (Correcher et al., 2019, Cheong and Tan, 2008).

Apart from the scheduling of berths, research has been done on the spatial shapes and layouts of port elements. Examples of spatial optimisation is the series of breakwater shape optimisation (Woerlee, 2019, Teeling, 2020, Schaly, 2021) and container terminal design optimisation (Alvita, 2020). The breakwater layout optimisation was done by means of genetic algorithms, which optimized the shape of the breakwater (length and angle), along with the constraint dimensions of the sheltered port area. The study only concerns the shape of the breakwater and does not take into account the topological variation of the breakwater. The terminal optimisation does allow for topological optimisation, but the shape of the terminal areas is limited to predefined geometric shapes.

Because there is limited research in spatial port layout optimisation, this research requires information from other sources regarding spatial optimisation. Architectural layout optimisation has more research available. An example of generative design would be the design of a hospital layout (Cubukcuoglu et al., 2022). Hospitals are complex buildings with requirements for the location of the rooms. For example, an intensive care unit must be close to operating theatre room due to the use of common equipment (Cubukcuoglu et al., 2022). The research translated functional requirements in constraints and objectives to be suitable for mixed integer programming (MIP).

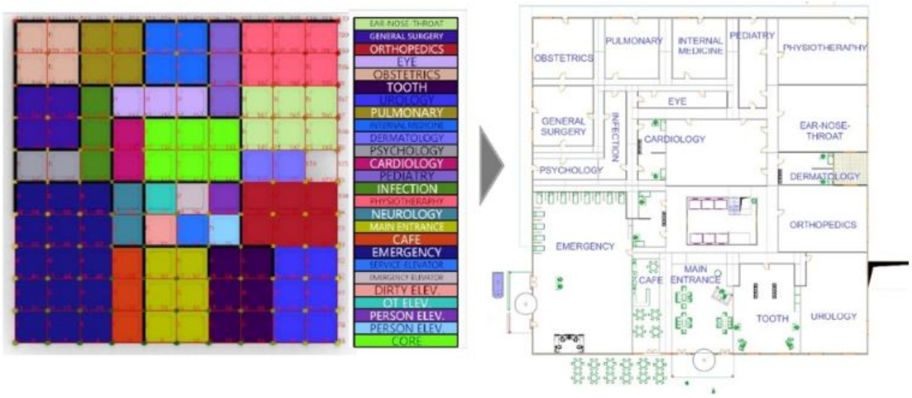


Figure 1.3: Example of an optimized ground floor layout produced by (Cubukcuoglu et al., 2022)

The building design problems resemble a studied layout optimisation problem: facility layout problems (FLP). The placement of facilities in the plant area is often referred to as a facility layout problem (Drira et al., 2007). This type of problem is mainly about work centers, warehouses, shops, etc. The problem does have spatial complexity, since it is about layouts of physical objects. However, the complexity in space is quite limited compared to berths in port basins. Berths in port basin not only affect the layout in space, but it also incurs a change in the space itself: a berth requires a dredged path from berth to entrance/exit. Since this report does not assume any geometry for the pathway or basin itself, the FLP seems to be lacking in spatial complexity, but it does provide a starting point. A more strategic optimisation, that concerns logistics in networks, is the distribution network optimisation problem (Chan & Chung, 2004). However, this problem disregards geometry and focuses on scheduling a supply chain.

The architectural layout optimisation does allow for inspiration for the port environment, since it shows layout optimisation techniques. However, a port context is different from architectural contexts in the sense that ports are designed for vessels that determine the port basin dimensions, while architectural designs mostly concerns design for humans. A difference is in the interaction with the environment; vessels operate in a waterborne environment, interacting with a basin bottom, waves, currents, etc., while the interaction of humans with the building not having these interactions.

Generative design concerns generation of topology (layouts), geometry (shapes) and networks. Topological optimisation is, for example, done in structural engineering. The basis for the topological optimisation is structural mechanics and material mechanics. An example is the optimisation of the steel profile of a cantilever beam (Sigmund, 2001). The objective is to minimize the material costs while maintaining structural integrity (Figure 1.4). More complex cantilevers have also been studied and optimized (Wang et al., 2002).

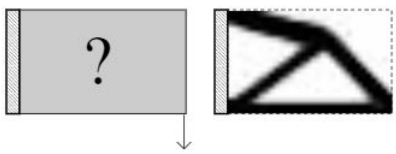


Figure 1.4: Example of an optimized cantilever topology produced by (Sigmund, 2001). The left displays the design domain and the right displays the optimized beam

A cantilever beam is a simple structure. In building engineering, it is common to find structures composed of more elements. Hofmeyer et al. (2017) presents a topology optimisation for structures with multiple elements, which interact with each other and transfer force through the structure. These structures are so-called hierarchical structures. The optimisation of these structures concerns mechanical properties of materials and structural mechanics theory.

Although there are examples that have commonalities and resemblance in spatial port layout design or generative design, research with regard to port basin berth optimisation seems little. The previously mentioned literature concerned the generation of the shape of the port infrastructure. The optimal berth layout in the basin is, to my knowledge, researched little. The commonalities with the port optimisation studies is the spatial optimisation of port infrastructure and the BAP optimisation concerns berth scheduling, but neither concerns the optimisation of berth layouts, which is aimed to be contributed to in this research.

For basin layout design, literature comes in a variety. The ports and waterways section of the TU Delft civil engineering and geosciences faculty published a free access book. The book provides an integral context of ports and waterway design and provides general port layout design rules (van Koningsveld et al., 2021). Other port design guidelines are more thorough in describing the technical design of such port basins (Thoresen, 2014, Tsinker, 1997), while others also provide a guide in site selection (PIANC, 2019). These guidelines function as general guidelines for every port basin. The PIANC group also provides guidelines for design guides for elements in the port basin, such as approach channels (PIANC, 2014), breakwaters (PIANC, 2016), fender systems, (PIANC, 2004), and more.

The general guidelines provide a basis, but most ports require special guidelines to comply with its unique situation. Thus, port authorities create their own guidelines. The Port of Rotterdam Authority have their own guideline 'Ontwerprichtlijnen havens en vaarwegen' (Port of Rotterdam, 2022). Guidelines are also created on a national basis. An alternative guideline is provided by the Spanish ministry of transport (Cortezón & de la Peña, 2007). Since research is on the instructions of the Port of Rotterdam, the 'Ontwerprichtlijnen' is the main document for basin layout design. The relevant elements from The guideline are discussed in Chapter 2.

1.3. Research aim and questions

The aim is to research the possibility for systematically generated berth layout designs to find a variant that provides support in decision-making for selecting the preferred berth layout variant. In order to do so, a method will be developed to generate berth layouts. Using this method, port basin layouts can be (near) optimised with its respective (near) optimal basin pathway in the early project phase with predetermined basin dimensions. After the method development and application, the added value of the concept of generative design on berth layout optimisation is discussed. To support the research aim, the following research question should be answered:

"How can generative design be used to optimise (or improve) design solutions of port basin layouts during the early project phase?"

The central research question is accompanied by the following sub-questions :

1. What is the definition of generative design and what are methods to generate design solutions of port basin layouts?
2. What objective and constraints will be considered to evaluate layout design variants?
3. How can berth layout designs be translated into a generative algorithm?
4. What is an applicable method to solve a berth layout optimisation problem?
5. What is the added value of algorithmic generated design solutions compared to traditional design solutions of port basin layouts in the early phase?

1.4. Methodology

The research sub-questions are answered using the approaches and methodologies mentioned respectively.

What is generative design and what are methods to generate design solutions of port basin layouts?
The first question aims to give insight into a definition of generative design and to assess the fundamentals of generative design. This question can be answered by providing theory used to apply generative design. The theories are then combined to allow for design generation and optimisation.

What objective and constraints will be considered to evaluate layout design variants?
This sub-question translates berth layout problem formulations into optimisation formulations. Based on the literature study on port planning and generative design, the decision variables, constraints, and objectives can be formulated. The formulations are then used for the design algorithm for the next research question.

How can berth layout designs be translated into a generative algorithm?
For the generative design method, mathematical programming will be used to optimise the basin layout. Answering the previous sub-question allows for an inventory of the objectives, constraints, and decision variables. However, these are not yet ready to be optimized. Thus, the translation has to be made from a qualitative to a quantitative description. The answer to this question should contain mathematical formulations and/or computational functions.

What is an applicable method to solve a berth layout optimisation problem?
To address this research question methods to optimise generated designs have to be developed. The methods, by preference, could be taken from existing optimisation problems or the problems have to be adapted. The developed methods are tested on fictive benchmarks for comparison.

What is the added value of algorithmic generated design solutions compared to traditional design solutions of port basin layouts in the early phase?
The question makes a comparison. Thus, a study has to be done to set the status quo of the port planning process. The method to do so is to collect information on the process time of designing the variants for a specific case and the produced designs by the algorithm. The information comes partly from interviewing people, acquiring documentation on the design process, as well as the accompanying planning.

1.5. Research scope

Port planning is a very broad discipline, according to the description provided in Section 1.1. In the short timeframe of a master thesis, optimisation of all port planning aspects would be too much. The Port of Rotterdam Authority is interested in an application on a current project on the Scheurhaven, a port basin near the North Sea entrance of the port. The research scope is narrowed down further to the berth layout optimisation in the basin of which the basin dimensions are constant. The berth layouts are designed for a given fleet composition with constant vessel dimensions. The method should allow for a fleet with, diverse design depths.

The berth layouts are optimised for initial dredging costs to facilitate the berths. This concerns the dredging for the berth and the pathway to the berth. The result is that structures in the basins, such as the construction of new quay walls, jetties, basin bottom protection, breakwaters, etc., are not considered. In Chapter 8 suggestions are made to include structures. Moreover, basin sedimentation processes, structure maintenance, changes in fleet characteristics and composition, changes in basin bottom soil properties, and other effects over time are not considered.

The basin of interest for this research is a sheltered basin. Hence, the influences of metocean conditions are not considered and so are the consequences of the metocean conditions; vessel movement as a result of waves, accessibility of berths, downtime induced by metocean conditions, pathway adjustments to take the effect of metocean conditions on vessel movement into account, etc.

1.6. Report outline

Starting off Chapter 2 of the report sets the reader up to understand the concepts used and gives background on the relevant aspects of port basin planning as well as generative design. using the background knowledge, the generative design approaches for basin layouts and pathways a re developed. chapter 3 presents the exact problem to solve the optimisation problem and the computational cheaper approach to optimizing the basin layout and in Chapter 4 the approach is validated and is analysed on sensitivity on basin parameters and its impact on dredging and pathway finding. Chapter 5 applies the method on the Scheurhaven and presents results on the case. The research is concluded by discussing the methods and results in Chapter 6, answering the main question in Chapter 7, and recommendations are made in Chapter 8. Figure 1.5 presents a flow chart of the report structure.

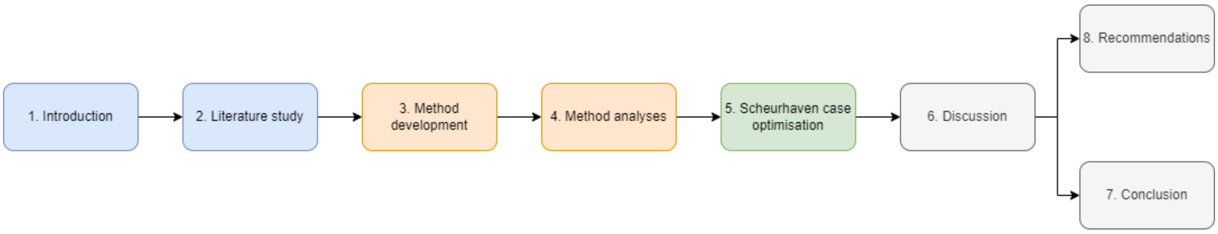


Figure 1.5: Flowchart of report structure

2

Literature study

2.1. Port basin layout design

This section provides the basis for basin layout design. This chapter provides an inventory of guidelines relevant to port basin design for tug boats and sea-going vessels from Port of Rotterdam (2022). The inventory is done to extract design constraints for the optimisation. The conclusion regarding the constraints are in the final subsection of this section. In addition to the guideline, a section is devoted to basin bottom stability, to provide information on basin bottom stability after defining a dredging pathway. The following sections address the types of design guidelines in order to assess the types of constraints required for the optimisation.

2.1.1. Design depth for vessels

A driver of port basin design is the berth designed for vessels to lay berth. The design depth is a combination of the nautical guaranteed depth (NGD) and a margin for dredging. The Port of Rotterdam gives the following equation for the NGD for tidal unbound vessels:

$$NGD_{NAP} = T_{max} + (FWA\% * T_{max}) + UKC_{Gross} + ALAT + HMT$$

Where:

NGD_{NAP} = Nautical guaranteed depth
T_{max} = The normative depth of a vessel measured in salt water (1.025 kg/m3)
$FWA\%$ = Fresh Water Allowance, additional depth compensating for sink
UKC_{Gross} = The Gross UKC used for calculating the NGD
$ALAT$ = The reduction plane is the reference plane for depths with respect to the sea level. (ALAT)
HMT = Hydro meteo allowance

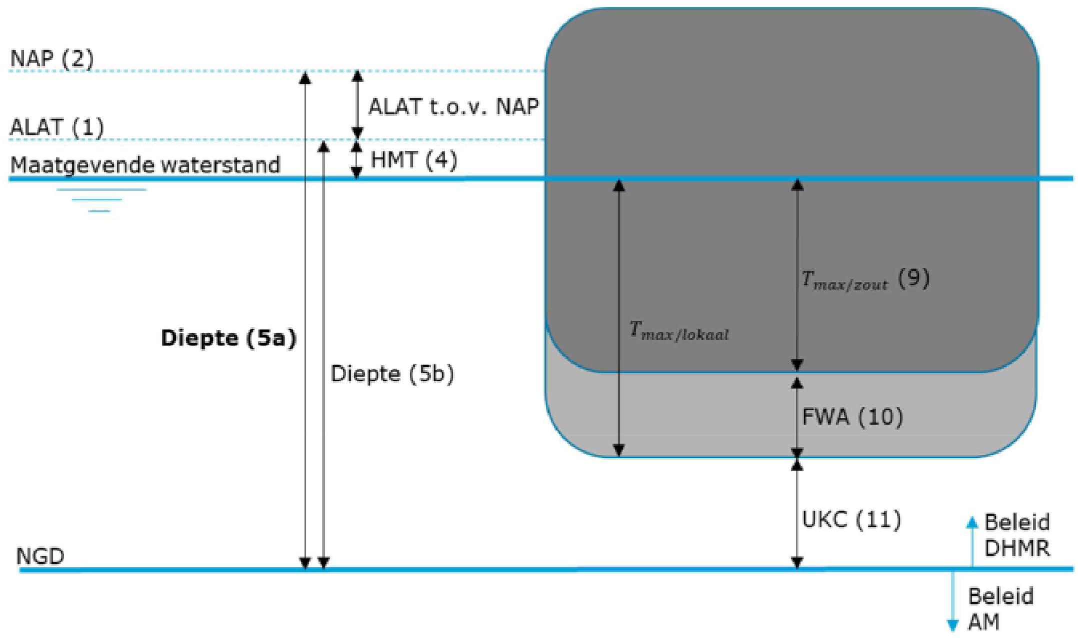


Figure 2.1: Vertical profile for tide unbound vessels

Most of the parameters are dependent on the location of the berth, because the Port of Rotterdam has changing conditions for different areas in terms of salt levels, vessel types, hydrometeo allowances, etc. The policies per region of each parameter are in Appendix B and Appendix A. The data are of relevance for the Scheurhaven case in Chapter 5.

The NGD is not yet the design depth in the basin. The nautical guaranteed depth is the depth that is guaranteed at a certain location and due to sedimentation in the port, the depth for the basin bottom design has to be larger than the NGD. The dredging depth is set in the guidelines provided by the asset management department of the port. The design depth for the Scheurhaven is one meter deeper than NGD (Port of Rotterdam, 2022).

2.1.2. Basin layout design

The following paragraphs present the spatial design guidelines for sea-going vessels. Although the case presented in Chapter 5 only concerns tugboats in the design, other seagoing guidelines are also inventoried to assess which constraints are required for a more general layout design. At the end of this subsection, all the figures are taken from the guidelines provided by the Port of Rotterdam Authority, referred to as 'The guideline' in some instances in this report (Port of Rotterdam, 2022).

Fairway width

The horizontal design, as it is called in The guideline, concerns the design in the 2D plane of the port surface area. Even though for this research the basin dimensions are not affected, it is of interest to keep the guideline into account to inventory the type of constraints for later research. The guideline makes a distinction between two types of basins: long and short basins. Short basins are basins suited for one design vessel length, while long basins allow for multiple berths to be in line with each other. The design requirements are depicted in Figure 2.2 and Figure 2.3.

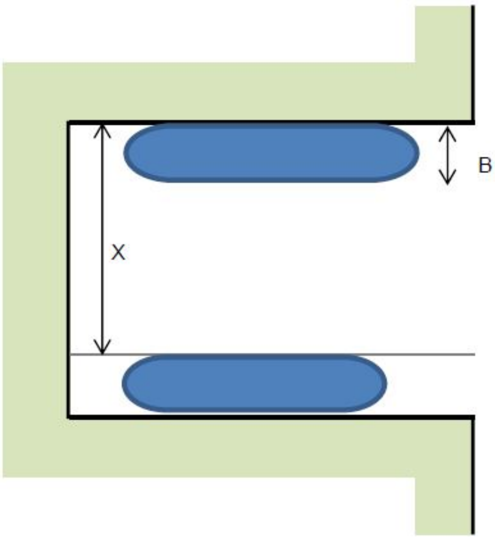


Figure 2.2: Required width for short basins (Port of Rotterdam, 2022)

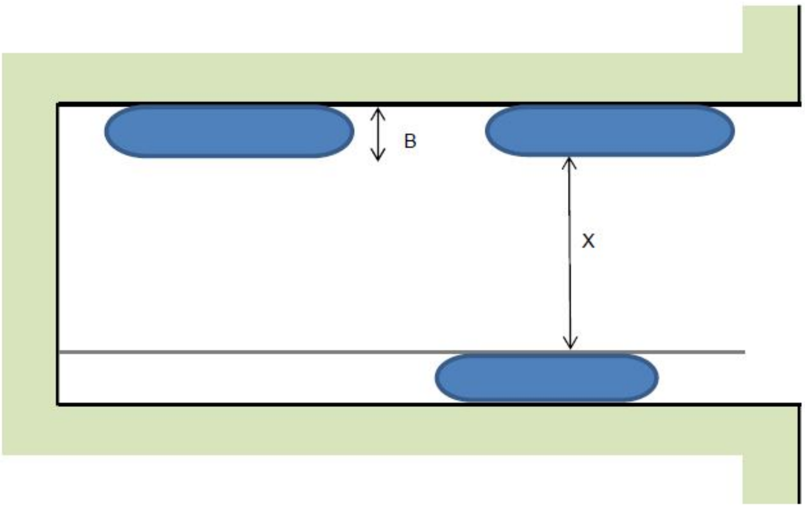


Figure 2.3: Required width for long basins (Port of Rotterdam, 2022)

The parameter B, beam of the vessel as depicted in Figure 2.3 is the normative design width for the largest berth and the parameter X is a location-specific parameter, which is found in Table 2.1

Region	Width X [m]
Europoort, Maasvlakte	2B + 70 m
Botlek, City	2B + 60 m
Dordrecht	3,3B
General (without tug boats)	3B

Table 2.1: Parameter X per region (Port of Rotterdam, 2022)

By entering the B into the functions for X, the basin width is obtained.

Berth guidelines
With the width of the basin set, the berth guidelines are applied. These guidelines provide relative

distance between berths and the required distance from basin infrastructure to allow for tug boats to move the vessels and to keep a safe margin for movement. The first guideline for berths prescribes distance between berths. The guideline provides a minimal distance requirement in Figure 2.4

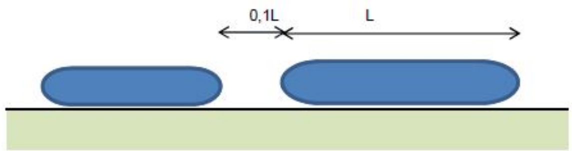


Figure 2.4: Required relative distance between berths (Port of Rotterdam, 2022)

With the parameter L being the Length overall (LOA) of the vessel. The relative distance is only given in the length direction of the vessel. The value 0,1L displays the gap between the berths. However, this value is not always 0,1L; it depends on the length of the vessel whether 0,1L is satisfactory. Table 2.2 provides the policy for the gap for several vessels.

Length [m]	Distance to obstacles and vessels [m]
L < 120 [m]	max(0,1L ; 10 [m])
120 ≤L < 350 [m]	min(max(15 [m] ; 0,1L) ; 35 [m])
L > 350 [m]	≥50 [m]

Table 2.2: Parameter X per region (Port of Rotterdam, 2022)

Other obstacles in the port basin can be embankments, vessels laying berth perpendicular to each other, and room for tugboats. All these obstacles have a related distance R, as depicted in Figure 2.5, Figure 2.6 and Figure 2.7.

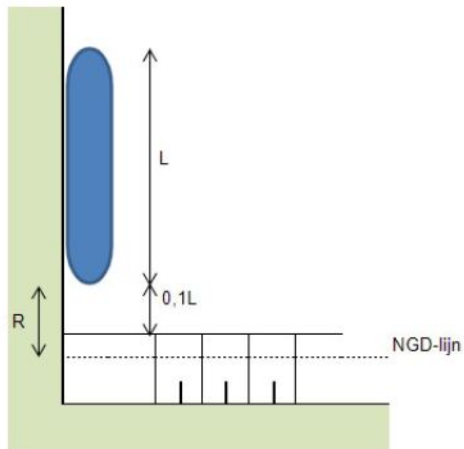


Figure 2.5: Required distance relative to embankment toe and NGD depth(Port of Rotterdam, 2022)

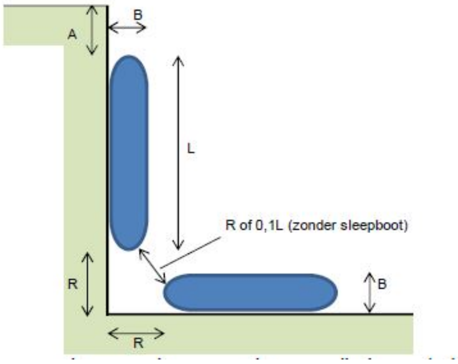


Figure 2.6: Required relative distance for perpendicular berths (Port of Rotterdam, 2022)

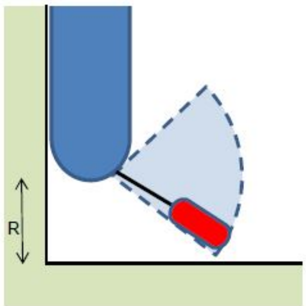


Figure 2.7: Margin for tugboat operation (Port of Rotterdam, 2022)

There is a dedicated guideline for tugboat berths. Instead of margins as functions of the vessel dimensions, the margins around the vessel are 5 meters overall, as depicted in Figure 2.8

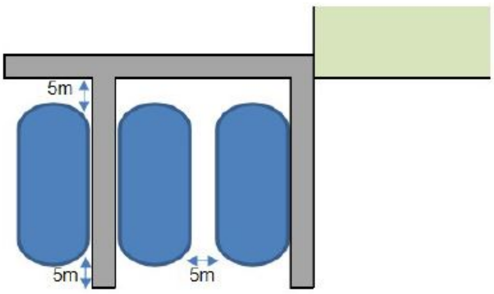


Figure 2.8: Berth requirements for tugboats (Port of Rotterdam, 2022)

The combination of all the above-mentioned guidelines provide a basis for port basin layout design.

Safety contours
In the port of Rotterdam, plenty of cargo is handled which requires caution. Vessels with dangerous goods have a safety radius surrounding them where vessels are not allowed to lay berth. These guidelines are set by the 'Binnenvaart politie regelement (BPR)' (Rijksoverheid, 2022) and European Agreement concerning the international Carriage of Dangerous Goods by Inland Waterways (Ministerie van Infrastructuur en Waterstaat, 2021). These guidelines are taken into account when designing.

On the contrary, these constraints are imposed by permanent structures, such as wind turbines. For wind turbines, Figure 2.9 depicts the constraint for berths. It's not allowed to reside for longer than 8 hours in the safety contour of the wind turbine, with people on board. Since workers may reside for longer times on tugboats and releasing cargo can take multiple days, it is safe to assume that berths are not allowed in the safety contour of a wind turbine.

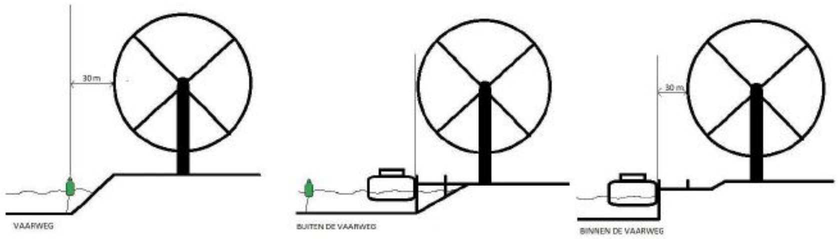


Figure 2.9: Safety contour for wind turbines (Port of Rotterdam, 2022)

In formula form, the safety contour is:

$$SC_{wt} = \frac{R_{wt}}{2} + 30$$

Where

SC_{wt} = Safety contour radius of a wind turbine [m]
R_{wt} = Wind turbine radius [m]

For the optimisation framework, the safety contours impose a constraint on possible berth locations inside the basin within range of the wind turbines.

2.1.3. Port basin bottom stability

Facilitating the layouts may require dredging of the port basin. This section will not go into the manners of dredging a port basin, but it provides the foundation for calculating stable slopes in a port basin. The stability slope of a soil is related to its friction angle (Verruijt, 2001). The friction angle for several soils is depicted in Figure 2.10.

Typical values of angle of repose [30].

Material (condition)	Angle of repose
Ashes	40°
Asphalt (crushed)	30–45°
Bark (wood refuse)	45°
Chalk	45°
Clay (dry lump)	25–40°
Clover seed	28°
Coconut (shredded)	45°
Coffee bean (fresh)	35–45°
Earth	30–45°
Flour (corn)	30–40°
Flour (wheat)	45°
Granite	35–40°
Gravel (crushed stone)	45°
Gravel (natural w/ sand)	25–30°
Malt	30–45°
Sand (dry)	34°
Sand (water filled)	15–30°
Sand (wet)	45°
Snow	38°
Wheat	27°

Figure 2.10: Angles of repose for several soil types (Al-Hashemi & Al-Amoudi, 2018)

The stability slope of soil is not equal to the friction angle, but it provides a reference for assuming stability slopes. The stability slope is also dependent on other factors, for example, the vessels inside the port basin spill material, which falls onto the basin bottom and affects soil properties. The result is soil that is mixed up with exhausted materials (Yazdi & Teshnizi, 2021). However, these effects are not taken into account further in this report.

2.1.4. Takeaways from literature for optimisation methods

The design guidelines in most cases prescribed required distancing between basin elements. Hence, the optimisation should give a formulation for distance constraints and measuring distances in the discrete space, which is presented in the next section. Another constraint is the pathway depth inside a basin to facilitate the vessels moving to the berth. Finally, safety contours impose constraints on the berth locations. Hence, the berth location optimisation should contain location constraints.

2.2. Generative design and theoretical background

Generative design is a systematic approach for exploration of layouts, shapes and networks (Nourian, 2023). In this section the concepts of generative design is introduced. First the environment is formulated as the basis. After, theories are presented that are used for the methods in Chapter 3. This includes different mathematical starting points and disciplines. Finally, the optimisation framework is presented, along methods to solve such problems. After reading this chapter, the concepts and operations done in Chapter 3 should be familiar.

2.2.1. Discrete space

In this research, the method to formulate the environment discrete is chosen. The benefit of such space formulation is the ability to create layouts and networks. In a continuous space, such analysis is hard due to the limitation in description of the space. Discrete space allows for shapes to be irregular (manifolds), while continuous space is to be described using analytical functions. Appendix E provides an insight in using continuous space and its implications of constraint formulation. The downside of this formulation is the difficulty of defining geometric shapes. The discrete elements in the environment have to be grouped to form a geometric shape. Geometric shapes in discrete space, specifically grids, is studied in the field of digital geometry. Digital geometry is the study of geometrical properties of subsets of digital images (Rosenfeld & Klette, 2002). If you chose to discretise the environment into a grid, it can be interpreted as an image. A handbook for digital geometry is provided by (Klette, 2004). This book functions as a staring point for digital geometry methods and concepts are used in the following sections.

The environment is modelled as a regular or irregular mesh. In the context of this thesis a regular mesh is formulated, in the form of a grid. A grid is a uniformly spaced mesh. Although a irregular mesh might be better, a grid is chosen due to its ease in understanding discrete space.

The grid in itself does not hold any information; it's just a division of space. In order to store information in the grid, the cells in the grid require an identity. The method to distribute identities is by application of space filling curves (Purss, 2017). These curves travel through the grid and give integer values a identities. The space filling curve applied is the morton order or morton code (Figure 2.11). The benefit of this formulation is that it preserves locality in the data structure (Purss, 2017). Although this feature is not taken advantage of in the algorithm, it can be useful for later applications with more advanced search algorithms. In order to preserve perfect "z's" in the grid, the following expression has to be satisfied:

$$n = 2^p$$
$$p \in \mathbb{Z}^+$$

In which n is the amount of rows and columns in the grid and Z all positive integers.

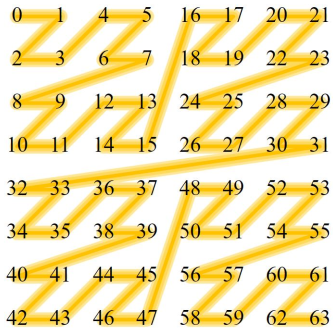


Figure 2.11: Morton code in a 8x8 grid. original on (Wikipedia, 2023)

2.2.2. Connectivity and Graph theory

With all the cells in the grids identified, all types of information can be attached to the cells: values, colors and other labels. Another important piece of information that has to be added to the grid is the relation between the cells in terms of location. The question is then: in what way are the cells connected? The way the cells are connected is by defining a neighbourhood. Two types of neighbourhoods exist:

1. Von Neumann or 4-connected neighbourhood, or;
2. Moore or 8-connected neighbourhood

The neighbourhood types are depicted in Figure 2.12. In the context of port planning, it is the choice of preference to go for the Moore neighbourhood, since diagonal traversal result in realistic modeling.

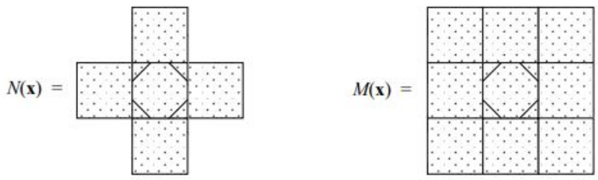


Figure 2.12: Neumann neighbourhood (left) and Moore neighbourhood (right)

The connectivity of each cell is stored in an adjacency matrix. This matrix stores '1' in the row and column index if the cells are connected and '0' if the cells do not have a connection. Doing this for whole grid, while satisfying the expression for the grid's dimensions, results in a matrix of n^2xn^2. Since each row stores at maximum eight ones and the dimension of the adjacency matrix grow exponentially as the grid increases in resolution, i.e. more grid cells in the grid, the resulting matrix is stored as a sparse matrix, a matrix which stores rows and columns indices with the respective value instead of the complete matrix.

The adjacency matrix created from the grid described in the previous section allows for a transformation of the adjacency matrix into a graph. Each connection between the cells are edges and the grid cell are schematised as vertices. The result is a graph, denoted as;

$$G(V,E)$$

In which V are the set of vertices in the grid and E are the edges in the graph. The visualisation of such graph is depicted in Figure 2.13. The graph can be adjusted by adding weights to the edges to allow for shortest path finding algorithms, such as Dijkstra's shortest path algorithm (Dijkstra, 1959) or A* (Doran & Kendall, 1966) amongst general graph theory (Diestel, 2017)

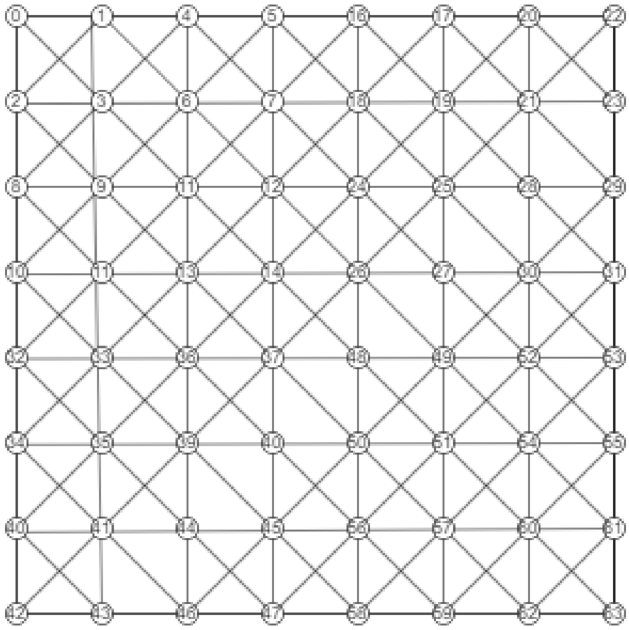


Figure 2.13: 8x8 grid graph with Moore neighbourhood and morton order vertices

2.2.3. Set theory

The vertices in the graph are in essence a set of index values. The set of vertices is denoted as:

$$V = \{1, 2, 3, ..., n^2\}$$

In which the integers stores in the set are all indices in the grid. In set theory there are a few basic operations used in this report:

- union, denoted as '∪'
- intersection, denoted as '∩'
- set difference, denoted as '\' or '-'

The operations are depicted in Figure 2.14.

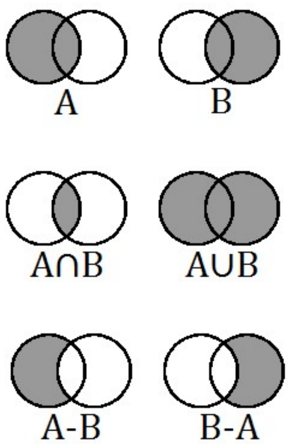


Figure 2.14: Basic set operations in set theory (Erdelsky, n.d.)

2.2.4. Mathematical morphology

Set theory forms the basis form many mathematical disciplines. One such discipline is mathematical morphology. Mathematical morphology provides an approach to processing digital images based on shapes (Haralick et al., 1987). In order for the grid to be ready for mathematical morphology, the grid has to be transformed into an image. This seems complicated, but a matrix with values between 0 and 255 are grayscale images. This characteristic allows for useful operations on the grid. Some basic operations in mathematical morphology are:

- Erosion, denoted as '$\ominus$'
- Dilation, denoted as '$\oplus$'

The visualisation of these operations are depicted in Figure 2.15

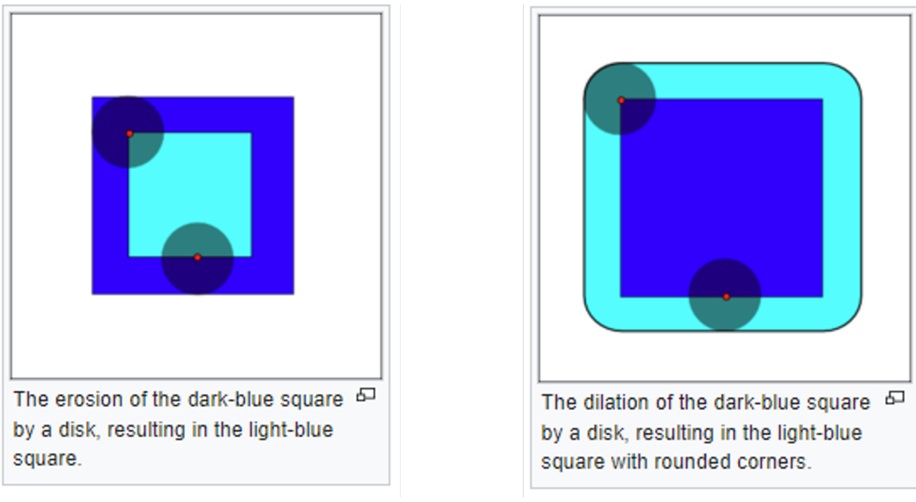


Figure 2.15: Basic set operations mathematical morphology

2.2.5. Mathematical topology

Going back a level from mathematical morphology to set theory, another discipline which is touched upon is mathematical topology. A topology is by definition the set of sets. This means that all sets in a space, together with the null space, form a topology. Translating this concept to a port layout design, a berth layout is basically a spatial set of berths, quay walls, water, land etc. Figure 2.16 in combination with the following expression describe the port basin as a topology:

$$T = \{berth\ 1, berth\ 2, berth\ 3, quay\ wall, water, land, \emptyset\}$$

Figure 2.16: Example of a topology in a basin. Each element in the topology has its own characteristics

2.2.6. Mathematical programming

In the previous subsections the theory is introduced to describe the environment in abstract concepts. This final section introduces the basic concept of mathematical optimisation. The typical mathematical optimisation formulation is (Snyman & Wilke, 2010):

$$minimize \quad f(\mathbf{x}) = [x_1, x_2, ..., x_n]^T \in \mathbb{R}^n \tag{2.1}$$
$$s.t. \quad g_j(\mathbf{x}) \leq 0, \quad j = 1, 2, .., m \tag{2.2}$$
$$h_j(\mathbf{x}) = 0, \quad j = 1, 2, ..., r \tag{2.3}$$

Where $f(\mathbf{x})$, $g_j(\mathbf{x})$, $h_j(\mathbf{x})$ are scalar functions of the real column vector $\mathbf{x}$. The abbreviation s.t. means 'subject to', which describes the inequality and equality constraints given by the respective functions $g_j(\mathbf{x})$ and $h_j(\mathbf{x})$. The function $f(\mathbf{x})$ describes the objective function for the design variables x_i of vector $\mathbf{x}$.

A special case is when all functions described are linear continuous functions. Such problem is called a linear programming problem. If at least one constraint or objective is an integer function, the problems described as a mixed integer problem or mixed integer linear problem.

Solving these problems and finding the global optimal solution can be done in various ways. Figure 2.17 provides an overview of optimisation algorithms. Note that the figure does not state (meta-)heuristic algorithms directly. Meta-heuristic algorithms are, for example, simulated annealing, tabu search and evolutionary computation. Deterministic algorithms do not have any randomness in its approach and tries to find the exact optimal solution, while stochastic methods approach the optimal solution by chance. Engineering problems are often not suitable for deterministic or enumerative methods, thus it would be to rely on the stochastic methods (Brameier & Banzhaf, 2007). Some meta-heuristic algorithms are nature-inspired algorithms such as evolution, bee colonies, ant colonies, among many others (Sun et al., 2022). The benefit of using meta-heuristic algorithms is that meta-heuristic optimisation algorithms seek quality feasible solutions to optimisation problems in circumstances where the complexities of the problem or the limited time available for solution do not allow exact solution. (Rardin & Uzsoy, 2001)

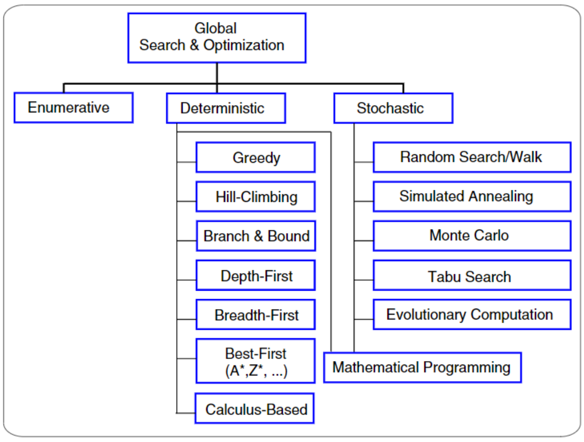


Figure 2.17: Global optimisation Approaches (Brameier & Banzhaf, 2007)

The problem definition along with the algorithms provide a framework in developing a method to optimize problems. The theory provided in the previous sections provide a framework to deal with design in a spatial context. Combining the two provides a method to optimize spatial problems.

2.2.7. Heuristic optimisation

As discussed in the previous section, heuristic algorithms tend to perform well in engineering problems, as well as in investing and moral behaviour (Gigerenzer, 2008). Heuristic algorithms are based on the heuristic processes in nature. It concerns the process of people doing something and learning from by means of feedback. Process is simulated by these algorithms, in an effort to approach the optimal solution.

These heuristic methods are not uncontroversial. According to (Sörensen, 2015), there has been a 'true tsunami' of so-called meta-heuristics. In the context of this research, metaheuristic algorithms are heuristic algorithms to solve heuristic problems. However, in this report, the narrative is about the same algorithms. The paper argues 'that this line of research is threatening to lead the area of meta-heuristics away from scientific rigor.' The scientific rigor of these algorithms to be in line with "that a heuristic did not necessarily have to be completely problem-dependent, but that general optimisation techniques could be developed that were applicable to a broad class of different problems". The reason that this section in the report is to show that these algorithms are interesting and promising, but some conservatism with these algorithms is appropriate, especially with unproven algorithms.

However, the research does not nullify every algorithm, but the background of these algorithms have to be checked. A python library that provides proven heuristic algorithms is pygmo (Hagberg et al., 2008). This library uses algorithms that are well known and verified, such as the improved harmony search (IHS) algorithm (Mahdavi et al., 2007).

3

Method development for berth layout and pathway optimisation

3.1. Port basin layout development theory

As stated in the introduction, the focus of the research is on the berth layout of a port basin and in specific developing layout variants. However, before the development, a process was walked through that assessed the physical, ecological, and socio-economic factors of the design and eventually translates it into a basin layout. Figure 3.1 depicts a schematic of the port layout development.

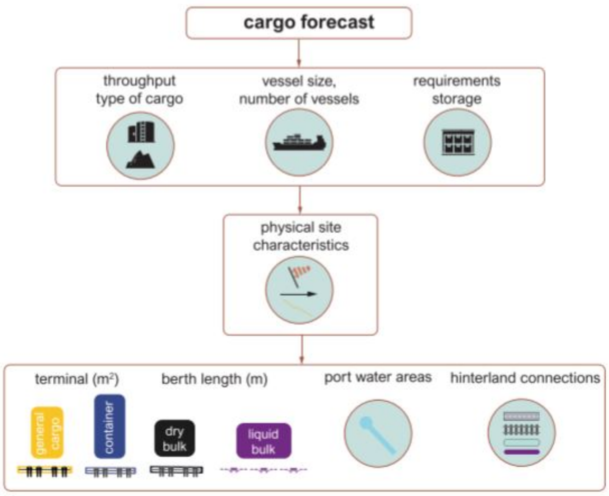


Figure 3.1: Schematic of port development (van Koningsveld et al., 2021)

The layout of interest is the berth layout in the port water areas. For this design exercise, many layout design factors are not taken into account, for example: pilot boarding areas, anchorage areas, outer and inner access channels, and turning basins. Along with the physical elements, operational considerations are also left out, such as cargo transfer, (de)berthing facilities, vessel turning assistance, etc. The next section provides an elaborate look at port basin layout design.

3.2. Port basin layout puzzle

The port basin layout is a system, where the optimal configuration depends on element interaction. The optimal berth location depends on different factors, but the focus is on the dredging required in the port basin. The parameters that affect the performance of a layout, e.g. the required dredging, depend on the interaction between berths, pathways of each berth, and the ability to insert structures in the layout. All the parameters in the port basin influence each other when dredging a pathway. A fundamental reason for this interaction is the effect of dredging on the stability of the bottom profile and the associated dredging required for this basin bottom profile. The effect is described in Chapter 2. This effect is illustrated in Figure 3.2

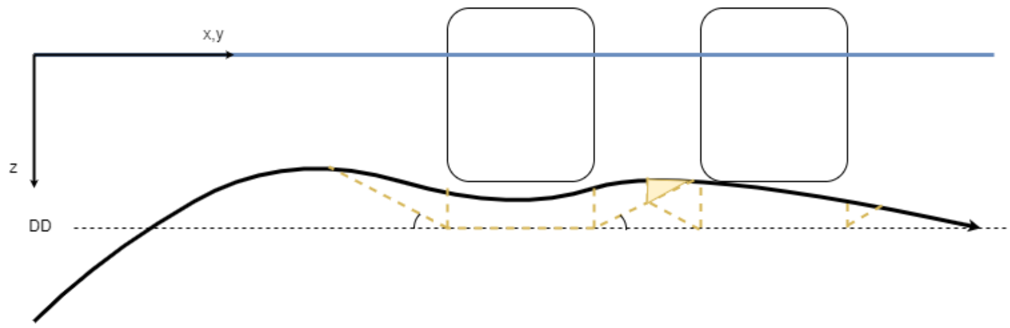


Figure 3.2: Dredging savings when berths are near each other. yellow dashed lines indicate the stability slope and the yellow triangle shows the saved dredging volume. The design depth is indicated with DD

The basin pathway is affected by the berth locations, but also by the structures inside the port basin. For example, a quay wall, if structural integrity is not affected, can function as a 'dredging shield'. Because the quay wall is rigid and solid, it can prevent the costs of dredging due to the formation of the stability slope. Another structure, which does not provide cost benefits, is an embankment. The embankment is a slope near the basin outline, which is not allowed to be affected. Hence, it will force the pathway away from it as the influence of the stability slope may not affect the embankment slope. The same goes for berth placement near an embankment.

The above-mentioned factors have to be taken into account for the optimisation. In order to truly optimize, all these factors have to be put in one optimisation problem formulation to find the (global) optimum. If the optimisation is cut into parts, the optimisations are done sequentially, which can lead to sub-optimal results.

To conclude the section, figure 3.3 displays the dependencies in the system that form the basin layout puzzle.

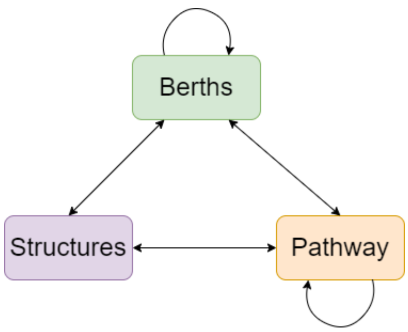


Figure 3.3: Pathway puzzle dependencies: Arrows indicate the influence between port basin elements. The arced arrows indicate the influence between between identical elements.

3.3. Heuristic berth layout optimisation

This section describes a heuristic approach to optimizing berth layout problem. The heuristic approach should find near-optimal solutions in a relatively short amount of time compared to the exact solution, which is discussed later. In order to do so, the problem is split up into two parts along with some assumptions which make the optimisation easier to solve. The first section of this chapter presents a berth location optimisation approach. The berth location optimisation is based on a common central path in the basin. After the locations are optimized, the pathway through the basin is optimized. In other words: the assumption of a central pathway is not taken into account. These optimisation steps are depicted in figure 3.4

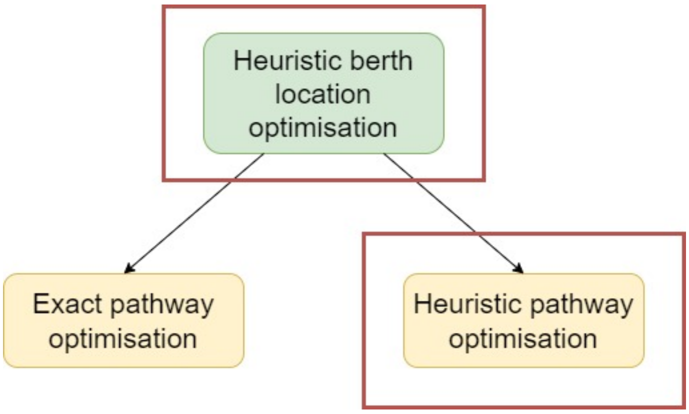


Figure 3.4: The optimisation workflow to optimise port basins using the proposed methods. This section presents the heuristic approaches

First, the design method is presented that discusses how to get to a formulation that can be optimised. After, the optimisation problem is presented and an algorithm to solve it is presented. To support these approaches a fictive basin is introduced to display the capabilities of this approach and the techniques used. The basins are not part of the Scheurhaven case nor the benchmarks.

3.3.1. Design concepts and assumptions

Optimizing the complete problem at once can lead to excessive computation in the optimisation process. If every berth layout gets optimized by means of location and pathway, it means that low-quality berth location configurations also get a pathway. To avoid this explicit enumeration, first, the locations are optimized. This assumes that a large part of layout performance on dredging is based on the location; a strong correlation exists between location in the port basin and the optimal dredging volumes; locating a berth further down the basin requires a longer pathway to be dredged.

In order to facilitate his assumption and to be able to optimise berth locations without finding the exact optimal pathway for each berth configuration, a central pathway is assumed to which the berths connect for its pathway to the entrance of the port basin, even if it's not the optimal path. The central path extends to the furthest berth inside the basin, such that each berth has a path to the entrance, but there is no unnecessary pathway dredging.

Another assumption is that the berths are represented in a single cell, neglecting the geometry of the berths. This assumption does compromise the representativeness of real port basin design, but this assumption allows for a starting point for optimisation.

The final assumption is that the paths of the berths connect to a single node in the middle on the entrance nodes, such that the central pathway connects to the central entrance node.

3.3.2. Example case introduction

For this section, a case is introduced to display the algorithms and its potential. The example basin is an F-shaped basin that is encircled by only quay walls. The values for the land grid cells is 1 and the values of the water cells is 0 in a matrix of 64 x 64 cells. The water area has a bottom profile that is constant over the whole basin with a depth of -4 relative to the water level. The basin is depicted in figure 3.5

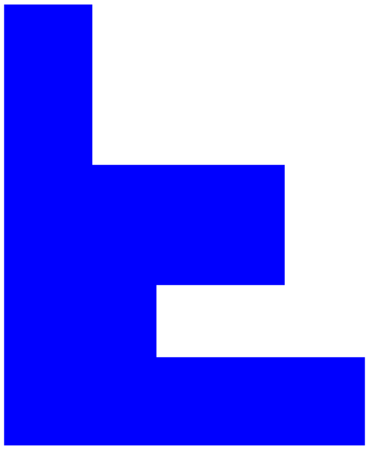


Figure 3.5: Example basin

The created environment has to be translated into a graph for the optimisation. The graph is made by checking the Moore neighbours for each cell. The result is an adjacency matrix of dimensions 4096 x 4096 with each column and row indexed representing a grid cell. This adjacency matrix can easily be translated into a graph by means of the networkx library in python, using the 'from_scipy_sparse' function. The graph is denoted as G(V, E) with V being all the vertices and E being all the edges. Note that the weight of the edges in the graph represents the relative distance between cells.

The depths of each vertex (grid cell center) are stored in a vector of length 65536 with each position coinciding with indices in the graph. This vector is called the depth vector, denoted as:

$$\mathbf{d_V} = \{d_{V,0}, d_{V,1}, d_{V,2}, ..., d_{V,4093}, d_{V,4094}, d_{V,4095}\}$$

The scope of the case is on the waterborne side of port planning and no adjustments are allowed to the land vertices. For the sake of the dredging algorithm and computational speed, the land nodes have to be expelled from the graph. The set of vertices consists of land vertices, V_L, and water nodes, V_W. The land nodes are the nodes with depth 0 or larger and the water vertices have smaller depths. The depths are with respect to the water level with the negative values being below water level and positive values being above water level. Finally, the nodes are either water vertices or land vertices, and no overlap in identities exists. With these characteristics, the following identities hold:

$$V_L = \{i \ : \ d_{V,i} \geq 0\}$$

$$V_W = \{i \ : \ d_{V,i} < 0\}$$

$$V_L \cup V_W = V$$

$$V_L \cap V_W = \emptyset$$

The edges have to be filtered accordingly, leading to the following notation:

$$E_W = \{(i,j) \ ; \ \forall_i \forall_j \in V_W\}$$

The considered graph is:

$$G(V_W, E_W)$$

The berths are only allowed to be located next to a quay wall, which allows only for a limited set of vertices as possible berth locations (PBL). The PBL set is found by applying linear algebra and by the assumption that berth locations are in the water and have a land connection. In order to check the land connection, an indexed vector is constructed which stores the water and land values in a vector of length 4096. The vector is referred to as the land/water vector:

$$\mathbf{v}_{lw} = \{v_{lw,0}, v_{lw,1}, ..., v_{lw,4094}, v_{lw,4095}\}$$

With:

$$v_{lw,i} = \begin{cases} 1 & \forall_i \in V_L, \\ 0 & \forall_i \in V_W \end{cases}$$

To check the connectivity with land vertices for every vertex, the dot product is applied to the adjacency matrix and the land/water vector. The result is a vector, $\mathbf{c}_L$ which stores the amount of connections with a land vertex:

$$A\mathbf{v}_{lw} = \mathbf{c}_L$$

The values in the connectivity vector range from 0, no connections to a land vertex, to 8, all connections are with land vertices. The vertices that are suitable for vessel locations is the vector indices where the connectivity vector is larger than 0 and the vertices are water vertices. The following set of indices contains a land connection:

$$V_{cL} = \{i \ : \ c_{L,i} > 0\}$$

With:

V_{cL} = The set of indices with a land connection
$c_{L,i}$ = The number of land connections for vertex i

The vertices that suit the criteria are the following set:

$$V_{PBL} = V_{cL} \cap V_W$$

This set is used as the basis for locating berths and for the optimisation. The location for the example basin is depicted in figure 3.6

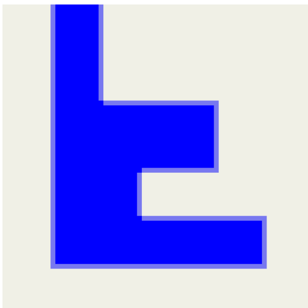


Figure 3.6: The possible berth locations, V_{PBL}, are displayed in purple.

3.3.3. Central basin pathway algorithm

The assumption of a central pathway has to be put into practice. Defining the central pathway seems easy: it has to be related to the center of gravity of the port basin shape and a direct line to the entrance point of the basin. This does hold up for convex shapes such as rectangles, circles, hexagons, etc., but not for irregular shapes (see figure 3.7). Port basins do have elements of geometric shapes, but the shape of the whole basin is not a geometric shape, such as a rectangle, circle, etc. Therefore it is more correct to assume that port basins are of manifold shape.

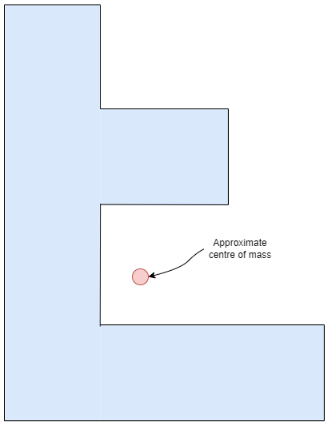


Figure 3.7: Centre of mass for a non-geometric shape

The central pathway has to be found in more complex theory. This is where mathematical morphology is of good use. In mathematical morphology algorithms exist that can find a so-called morphological skeleton. This skeleton is found by thinning (eroding the layers of a shape) a shape until it is a line. One such algorithm is provided by the scipy library in python, which uses a skeletonization algorithm provided by (Zhang & Suen, 1984). The pixels that are in the skeleton are related to the indices in the graph. Hence, a large part of the pathway vertices is defined. The morphological skeleton in itself is not yet a path to the entrance, thus a connection has to be made. This is done by using the multi-source Dijkstra function from networkx in python. This function finds the shortest path from the skeleton vertices (sources) to the entrance point (target). The result is depicted in figure 3.8 as the skeleton is now connected to the entrance, thus forming a central path.

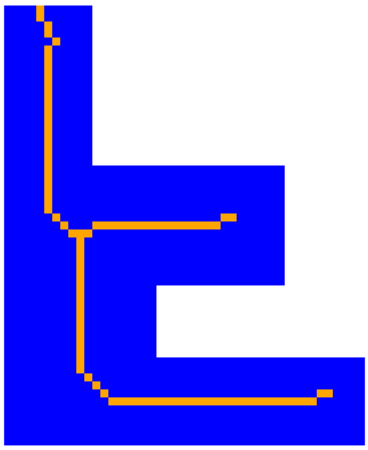


Figure 3.8: Central pathway using the skeletonization algorithm indicated in yellow

With the central pathway defined, the berth pathway has to be defined. The berths used are a subset of the possible vessel locations. Each berth has it separate path to the central basin. Since the berths are just one vertex (or pixel), the multi-source Dijkstra function can be used again: the target is the berth instead of the entrance node from the previous step. This has to be done for each berth. The result is displayed in figure 3.9.

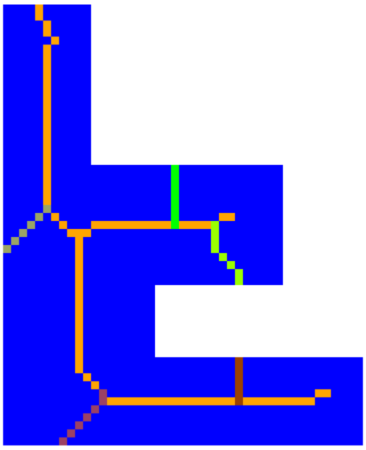


Figure 3.9: Pathway through a port basin for a set of random berths

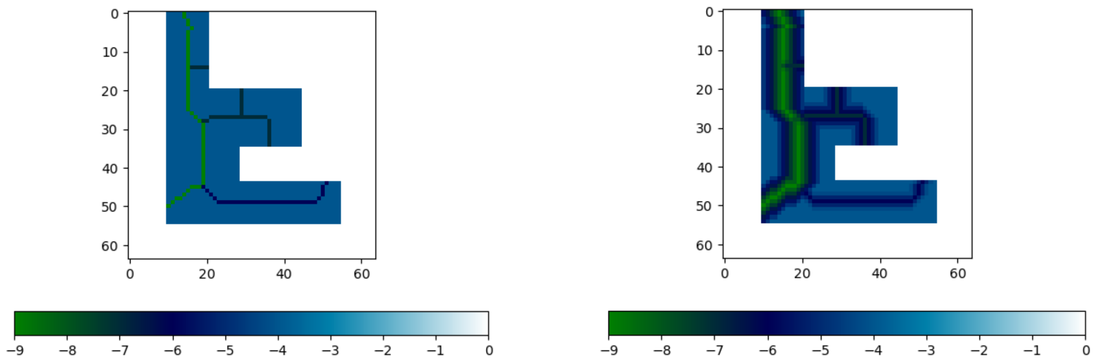
The final part is the actual dredging required for this path to be realized. The dredging function can be found in appendix D. The function comes down to checking each of the adjacent vertices if they are in equilibrium or not and if not, it has to be 'dredged' to equilibrium. The dredging function is referred to as the function:

$$\mathbf{f}_{dredge}(\mathbf{J}_{berths}, \mathbf{d}_{berths}) = \mathbf{d}_{eq}$$
$$\mathbf{J}_{berths} = \{J_1, J_2, ..., J_{nb}\} \in V_{PBL}$$
$$\mathbf{d}_{berths} = \{d_1, d_2, ..., d_{nb}\}$$

With:

$\mathbf{f}_{dredge}$ = The dredging function which creates equilibrium after
$\mathbf{J}_{berths}$ = The vertices vector of the berth locations
$\mathbf{d}_{berths}$ = The respective depths vector of the berths
$\mathbf{d}_{eq}$ = The resulting vector which stores the depths of the vertices
nb = Number of berths

The first step of the dredging is to set the required depths for the central pathway in such way that for every berth the path is deep enough for its vessel to follow the path. This is done ordering the berth depth from shallow to deep and excavate the path from shallow to deep. This result of this process is depicted in figure 3.10a. After the required depths have been set throughout the graph, the equilibrium is formed using the dredging algorithm. The result is a basin with the required depths along the path with a stable basin bottom, depicted in figure 3.10b.



(a) The pathway is dredged with the respective required depth. The locations of the berths are illustrative and not optimised.

(b) The equilibrium bottom profile is created to facilitate the required pathway depths

Figure 3.10: Generation of an equilibrium basin bottom profile to facilitate the berths in the basin using a central path

Using the new generated equilibrium profile, the dredged volume can be calculated. The required volume dredged comes down to subtracting the new equilibrium profile from the initial bottom profile, summing all the differences and multiplying with the surface of the grid cell. The equilibrium depth is vector:

$$d_{eq} = \{d_{eq,i} \ : \ \forall_i \in V_W\}$$

The dredged volume is:

$$[\mathbf{d}_V - \mathbf{f}_{dredge}(\mathbf{J}_{berths}, \mathbf{d}_{berths})]^T \cdot \mathbf{1} \quad \forall_i \in V_W$$

With Δ being the surface value of the vertex. In the example case, this value is set to 1.

3.3.4. Berth location optimisation

The process described in the previous section is now suitable for optimisation. Chapter 2 presented a general optimisation problem and this structure will be used for this optimisation as well. Taking into account all the requirements and assumptions for the fictive case, the optimisation problem is:

$$\min_{\mathbf{J}_{berth}} \quad [\mathbf{d}_V - \mathbf{f}_{dredge}(\mathbf{J}_{berths}, \mathbf{d}_{berths})]^T \cdot \mathbf{1} \qquad \forall_i \in V_W \tag{3.1}$$

$$s.t. \quad min(dist(I_i, I_j)) \geq max(d_{req,v_i}, d_{req,v_j}) \qquad \forall_i \forall_j \in \mathbf{J}_{berths},\ i \neq j \tag{3.2}$$

With:

$$\mathbf{J}_{berth} \in_R V_{PBL}$$

Where:

$min(dist(J_i, J_j))$ = The shortest path between berth i and j
d_{req,J_i} = The minimal required distance from berth i to other berths

Note that the actual optimisation of berth locations is not done in this section, but in Chapter 4. This chapter provides the optimisation formulation to be used and tested later. The optimisation problem is put into an heuristic algorithm, which picks random samples from the set of possible berth locations. Hence, the set of berths is a random sample of V_{PBL}.

The meta-heuristic optimisation algorithm is provided by the python library pygmo (Hagberg et al., 2008), which offers two algorithms for constrained optimisation:

- Improved harmony search (IHS)
- Extended ant colony optimisation (GACO)

By doing a short comparison, which will be discussed more in depth in Chapter 4, the IHS algorithm seemed to be the most promising for this problem. The result of this optimisation is a set of vertices which represent the optimal berth locations, which is depicted in figure 3.11. The produced solution is not optimized to the fullest, but is provides a good output for the next optimisation

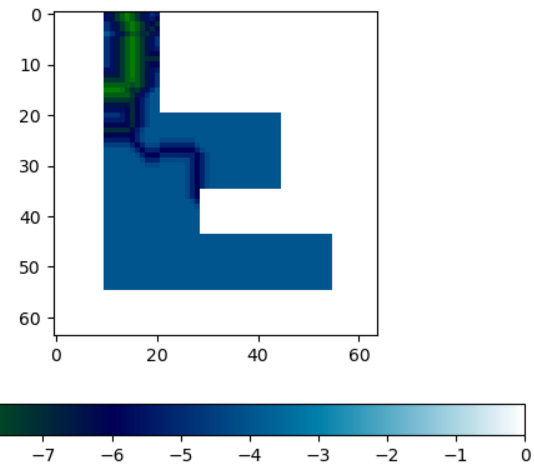


Figure 3.11: Optimal berth locations after a short run. This is not the exact optimum.

3.4. Heuristic pathway optimisation

In this section we let go of the assumption that the pathway has to be through the centre of the basin shape, i.e. equal to the morphological skeleton of the basin. At the moment of writing, an approach to solve this problem does not exist yet, thus a method has to be developed. In the following sections, like the previous section, the design concepts as well as the assumptions are discussed. After, the algorithm is described to create an improved, near optimal pathway through the basin, given the berth locations from the previous section. Finally, the optimisation is presented along with the result

3.4.1. Design concepts and assumptions

The central pathway functions as a benchmark for the pathway optimisation. Finding the optimal solution for multiple berths is a difficult problem; the grid changes as one berth is placed for the next berth. The optimal solution can only be found in a formulation defined in Section 3.4, as the optimisation for all pathways from the berths have to happen in the same optimisation. However, this, as stated before, increases computational time significantly.

To keep the optimisation cheap, a workaround is defined. The workaround is to formulate the dredging costs in the edges of the graph. This way the Dijkstra algorithm is, to some extend, applicable for determining the cheapest path in terms of required dredging volumes. The algorithm works according to the following steps:

1. Update weight of edges using the vertices
2. Find shortest path for berth 1
3. Dredge pathway in the graph
4. Update weights in the graph with new equilibrium profile
5. Repeat step 2 till 4 for N berths
6. Repeat for all berth orders

The resulting graph should be a path that improves the central pathway approach, but claiming that it is the optimal pathway would be an overstatement. However, it does provide a pathway that takes advantage of traversing through existing pathways.

The optimisation will be done different compared to previous sections in this report. Since the berth locations are determined from the previous optimisation and the proposed algorithm is finished after N iterations, there is no need for heuristic optimisation. The optimisation will now be a simple script without any mathematical optimisation formulation.

3.4.2. Optimal basin pathway algorithm and results

As described in the previous paragraph, the dredging volumes have to be stored as weight in the edges in order for shortest path algorithms to work. According to the first step, the weights of the edges have to be updated for the dredging. The following factors have to be taken into account:

- Dredging is with respect to the current depth
- If an edge is used and the path from the berth the vertex passed has to be dredged

Taking into account the first factor, the graph vertices are updated with regard to the required depth of the first berth. The result is a graph with each vertex representing the difference between the design depth for the berth and the initial depth. Since negative dredging (soil accretion) does not make sense in this context, all negative values are set to 0. Next, the edges are updated using this new graph using the following formula:

$$c_{ij} = \frac{v_i + v_j}{2} \quad \forall_{ij} \in E$$

In figure 3.12 an example is given in a short 4 x 4 grid. In this grid a berth with depth -8 will be located.

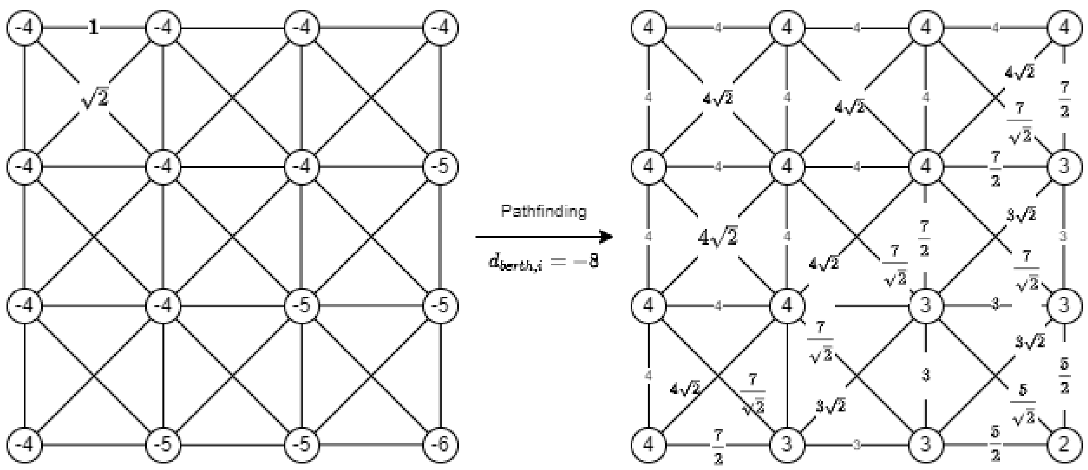


Figure 3.12: A graph with its original depths stored in the vertices and the length of the edges stored in the edge. The right figure displays the pathway finding graph if a berth with depth -8 will be placed. The values on the edges on the right indicate value $c_{ij} * d_{ij}$

This will, to some extend, take into account dredging when traversing through the graph. The graph then is ready for finding the optimal path. The Dijkstra algorithm, provided by the networkx library in python, will find the path with least dredging costs and the result is a set of vertices which will represent the path in the basin. The pathway will be dredged, as stated in step 3 of the algorithm, in the original graph, not in the updated graph, in which the vertex depth value represented the difference between the berth depth and initial depth.

The next step is to repeat the previous steps, but with the new berth and its respective depth. Note that the graph has to updated with respect to the new profile, which facilitated berth 1. After the repetition for every berth, the result is a basin with an optimal path for the set of berths in the basin:

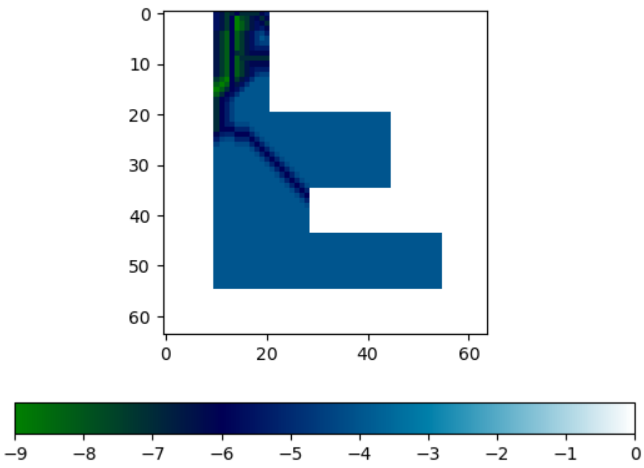


Figure 3.13: Optimal pathway through the port basin using the described optimal pathway algorithm

3.5. Exact pathway optimization

This section introduces the exact optimisation formulation, given a set of assumptions to make optimisation possible. First the assumptions are given for the optimisation, after the complete exact optimisation problem is presented. The problem description is supported by an explanation of the objective, constraints as well as the decision variables.

This section only give the optimisation problem formulation for the basin pathway. The basin pathway is the pathway from all berths to the entrance of a port basin. It takes manual berth location input or heuristically optimised berth locations, as depicted in figure 3.14.

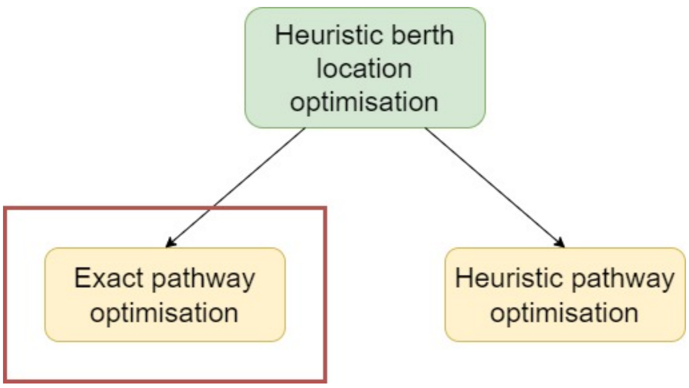


Figure 3.14: This section gives the exact pathway optimisation as part of the workflow in the figure. The complete basin layout optimisation consists of berth optimisation and sequential pathway optimisation.

3.5.1. Basin pathway optimisation problem formulation

Before getting into the problem formulation, some assumptions are made in bullet points.

Assumptions

- The main assumption is that berths are represented as single-cell entities, implying that the berth does not have geometry
- The vessels inside the basin are indifferent of pathway directions and length and that vessels are allowed to make any turn, even sharp corners.
- The basin bottom is of uniform material, making the stability slope constant over the whole basin.

.

Using the assumptions mentioned and given a graph $G(V_W, E_W)$ which represents the water area of a basin with a given basin bottom profile, mathematical programming model is:

$$\min_{v,x,\lambda} \sum_{i=1}^{V_W} v_i - \delta_{0,i} \tag{3.3}$$

$$s.t. \sum_{j=1}^{V_W} x_{ij} - \sum_{j=1}^{V_W} x_{ji} = \begin{cases} 1 & \forall_i \in S, \\ -n_b & \forall_i \in T, \\ 0 & \forall_i \in V \setminus S, T \end{cases} \tag{3.4}$$

$$v_i \geq \delta_i \qquad \forall_i \in S \tag{3.5}$$

$$x_{ij} \geq 1 - \mu(1 - \lambda_{ij}) \qquad \forall_{(i,j)} \in E_W \tag{3.6}$$

$$0 \geq x_{ij} - \mu\lambda_{ij} \qquad \forall_{(i,j)} \in E_W \tag{3.7}$$

$$v_j \geq v_i - \mu(1 - \lambda_{ij}) \qquad \forall_i \forall_j \in V_W \setminus T,\ i \neq j \tag{3.8}$$

$$v_j \geq v_i - \Delta d_{ij} \qquad \forall_{(i,j)} \in E_W \tag{3.9}$$

$$v_i \geq \delta_{0,i} \qquad \forall_i \in V_W \tag{3.10}$$

$$v_i \in [0, d_{max}]$$
$$x_{ij} \in \{0, 1, 2, .., n_b\}$$
$$\lambda_{ij} \in \{0, 1\}$$

Sets

V_W = The vertices that represent water in the grid
E_W = The edges that connect the water vertices V_W
S = Source node, i.e. the berth locations, $S \subseteq V_W$
T = Termination node, i.e. basin entrance/exit, $T \subseteq V_W$

Parameters

nb = Number of berths in the port basin, $nb \in \mathbb{Z}^+$
δ_i = The required depth of berth i, $i \in V_W$
μ = A large constant number, $\mu >> nb$
Δ = Unit depth difference allowed for stability slope ϕ, $\phi \in \mathbb{R}$
d_{ij} = Distance from vertex i to vertex j, $d_{ij} \in E_W$
$\delta_{0,i}$ = The original depth of vertex i, $i \in V_W$

Variables

v_i = Depth of vertex i, $i \in V_W$
x_{ij} = Flow out of vertex i to vertex j, $(i, j) \in E_W$
λ_{ij} = Slack variable for edge activation if part of the optimal pathway, $(i, j) \in E_W$

The complete problem is in essence a modified elementary shortest path problem formulation for integer programming (Taccari, 2016) and it has some aspects from a network flow distribution optimisation problem. Instead of optimizing for the distances over the edges, the optimisation is over the vertices, which represent the depth of each cell. In short, the optimisation consists of the following parts:

1. The objective (3.3)
2. The flow constraint for path finding (3.4)
3. Berth depth requirement (3.5)
4. Big M pathway edge activation constraints (3.6-3.8)
5. Soil stability constraint (3.9)
6. Minimum depth constraint (3.10)

Each of the parts is discussed to provide the idea behind the constraint.

Objective The objective in this context is to minimize the sum of 'depths' of all the vertices v_i minus the original depth δ_i. This makes the optimisation algorithm 'push the values down' of the depths facilitating the formation of an equilibrium profile. This objective formulation in combination with constraint 3.9 generate a stable equilibrium profile

Flow constraints for path finding The basic idea is stability of 'flow' between the activated edges. For the source it states that there should be at least one outgoing activated edge x_{ij} more than incoming activated edges x_{ji}. The constraint 'activates' the path finding, since the path cannot end in the source edge. For the sink T, which is a dummy node connected to all vertices at the entrance of the basin, the amount of activated edges going out should be at least one less than the activated edges incoming. The other edges, so the vertices which are not the source or sink, require to have balance in flow, i.e. the incoming activated edges should be equal to the outgoing activated edges.

Berth depth requirement The berths (sources) should satisfy a minimum depth δ_i determined by the vessel dimensions. The depth is not set as an equilibrium constraint, because that would be detrimental to the algorithm performance. The constraint is set such that larger depths are allowed, but the algorithm, in practice, always tend to the depth given due to the objective.

Big M pathway edge activation constraints This set of constraints ensures that, if the edge is part of the shortest path, the vertices crossed should at least require the depth determined by the source it traveled from. essentially, this constraint is an if-statement: if the edge is in the path from the berth, it should be at least as deep as the berth it departed from.

Integer programming does not allow for if statements, thus the big M method is applied to imitate such statement. μ is large number, which allows for v_j be any value if λ_{ij} is 0 (v_j is not in the path) and it has to be equal or larger than v_i if λ_{ij} is 1 (v_j is part of the shortest path for the berth).

Soil stability constraint The vertices which are connected, should satisfy the soil stability constraint. In the equation the ϕ depicts the soil stability angle (Al-Hashemi & Al-Amoudi, 2018) and d_{ij} is the distance between vertex i and j. This constraint only consists of the low limit, since it has to counteract the minimization objective.

Minimum depth constraint The vertice v_i are forced to 0 by the optimisation objective. However, because the basin bottom has an initial depth, it is not allowed for the vertices to be shallower than the initial depth $\delta_{0,i}$. A depth shallower than the initial depth means sediment supplementation, which is not allowed in this context.

3.5.2. Exact solution approach

The problem formulated at the start of the section can be solved with a MIP-solver of choice, e.g. commercial solvers: gurobi (Gurobi, 2023), cplex (IBM, 2023), xpress (FICO, 2023) or free solvers such as glpk (GNU, 2023). Note that the performance can vary per solver, but the commercial solver tend to perform better overall (Mittelman, 2011), especially on MIP problems.

However, solving the exact solution can take a long time as the amount of vertices and edges get larger and more constraints are added. In a port planning context, it's safe to assume that the constraints can get numerous, which would make the problem computationally expensive to solve. Some problems can take weeks to solve using the exact method.

4

Method verification

The case presented in the previous chapter was an example to show what the methods do in the discrete port environment. The performance is not yet tested. The testing of the method requires benchmark cases. In this chapter, 5 benchmark cases are presented. Each of the cases has a unique problem, which will test the algorithms.

4.1. Method validation

Before the validation of the algorithms, remarks regarding brute force optimisation are done. For benchmarking often the results are compared to the brute force solution to display the benefit of alternative approaches and to measure performance. However, these types of problems grow exponentially in number of combinations, which is shown in the first section.

After the remarks, the environment is presented with the water area, basin bottom profile, and the possible berth locations. Then, the berth optimisations are optimized and a qualitative reflection is done on the locations. After that, the pathways are optimized using the heuristic, as well as the exact approach. To make the approaches comparable, the optimisations are limited in time. The optimisation approaches are also compared to the objective value of the central pathway used for the berth optimisation. Finally, the results are discussed for each of the benchmarks.

An important notion is that the objective value is dependent on the dredging formulation for the exact optimisation and the dredging function for the heuristic approaches. The exact optimisation uses the Moore neighbourhoods, while the dredging function uses the Neumann neighbourhoods. The result is that the dredging function, hence the heuristic approach, underestimates the dredging value. The reason is the stability requirement between diagonal vertices that is not adapted for.

4.1.1. Remarks on brute force optimisation

This chapter does not include a brute force calculation for comparison. The downside is that the absolute global optimum is not found and, hence, cannot be compared to the exact and heuristic approach. The sole reason for this decision is the immense computational complexity of such computation. No exact formula exists for the computational complexity of paths in a graph, but a simple approximation can be done.

Given the graph G(V,E) in which V is the vertices and E is the edges. Assume a port basin with B number of berths and the goal is to find the optimal path for all the berths through the basin to some entrance T. To find the optimal pathway by brute force, all berth configurations and for each configuration all the pathways have to be evaluated. Let's assume that only a select set of vertices is suitable for the berths, V_{PBL}. The set V_{PBL} consists of N elements. Berth 1 has N possible locations and the next berth,

berth 2, has N-1 possible locations and berth 3 has N-2 possible locations, and so on. This sequence can be continued for B berths. The amount of berth configurations is:

$$N_{configs,berth} = \binom{N}{B}$$

where $N_{configs,berth}$ is the number of berth configurations.

For every configuration, all pathways from berth to entrance have to be considered. The approximation for this varies per graph, hence no formula exists to estimate all the possible paths in a graph. Algorithms exist to calculate the number of paths by means of a depth-first-search algorithm. However, in this section, a simple approximation is done. Assume that for every path, also paths which do not connect the berth to the entrance, have to be evaluated. A path is in essence the activation of edges in a graph; an edges is either activated, 1, or not, 0. This has to be done for every edge in the graph. Let's assume that the graph G(V,E) has k edges. The number of possible combinations of (de)activated edges is:

$$K_{configs,edges} = 2^k$$

where $K_{configs,edges}$ is the number of configurations of (de)activated edges.

The simple estimation for the total amount of evaluations is:

$$N_{evaluations} = N_{configs,berth} * K_{configs,edges}$$

where $N_{evaluations}$ is the total number of evaluations done.

To give a sense of magnitude, a fictional grid graph with 42 nodes (6 x 7 cells) is considered, which is very small. This basin requires 3 berths in it. Not all cells are suitable for berths, thus the amount of possible berth locations is reduced to 17, which represents the 3 outer sides of the grid graph. The number of berth configurations is:

$$N_{configs,berth} = \binom{17}{3} = 4080$$

The grid graph uses Moore neighbourhoods for connectivity, resulting in 262 edges in total. The total amount of combinations for (de)activated edges is:

$$K_{configs,edges} = 2^{262} = 7,4 * 10^{78}$$

The total amount of evaluations is:

$$N_{evaluations} = 4080 * 7,4 * 10^{78} = 3,0 * 10^{82}$$

This number is so large, even if you manage to evaluate a configuration in 1 nanosecond (10^{-9} seconds) it will take you longer than the elapsed time of planet Earth's existence, which is about $1,45 * 10^{17}$ seconds.

There are faster ways to brute force the optimal solution, but with this small calculation, the aim is to display the size these combinatorial optimisation problems can amass. Hence, brute forcing a solution does not seem feasible in the time frame of interest. Thus, we will only consider the heuristic and exact optimisation scores in the validation

4.1.2. Benchmark 1: Rectangle basin with uniform basin bottom and uniform fleet

In order to validate the methods, a new test case environment is modeled. In contrary to the previous case, the set-up and methods will not be discussed, because the methods applied are the same. This section simply concerns the layout and the optimisation outcomes.

The case consists of a basin of 10 units in width and 60 units in length, depicted in figure 4.1. The basin should facilitate 6 berths with a design depth of -7 meters. All of the berths have a minimal required distance with respect to each other of 10 units, as described in Table 4.1

Berth no.	Required depth [units]	Minimal distance to other berths [units]
1	-7	10
2	-7	10
3	-7	10
4	-7	10
5	-7	10
6	-7	10

Table 4.1: Fleet composition for benchmark case 1

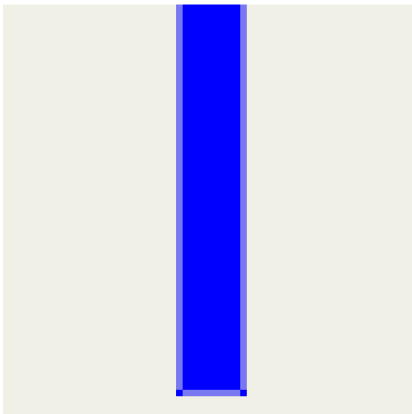
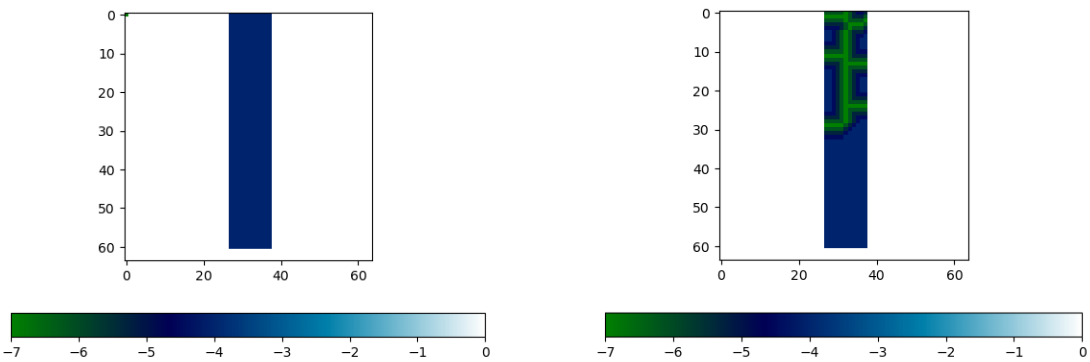


Figure 4.1: Rectangle basin with the possible berth locations in purple.

The basin and the berths are used as input for the heuristic berth optimisation formulation. The optimisation consist of a population of 20000 genomes and is evolved for 10000 generations, resulting in $2 * 10^8$ evaluations. The algorithm used is the improved harmony search (IHS) algorithm provided by pygmo, using the standard settings. The location optimisation resulted in the following berth locations and pathway:

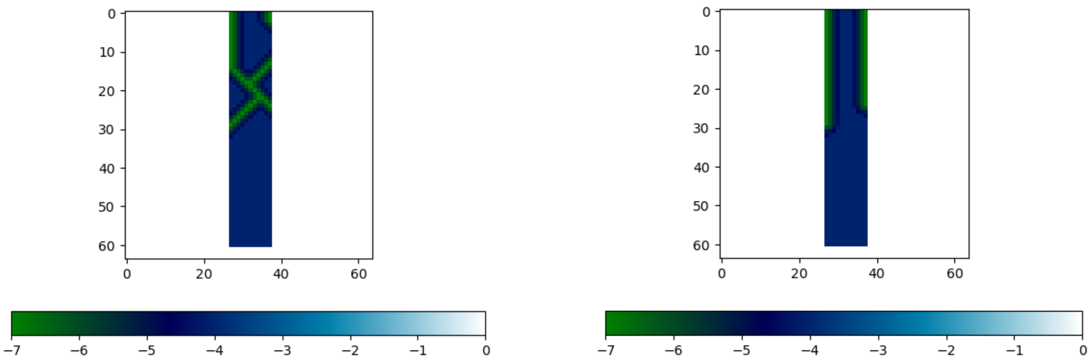


(a) Rectangle basin bottom

(b) Optimal berth locations

Figure 4.2: Basin bottom and optimised berth locations

The optimal pathway produced by the exact and heuristic approach are depicted in the figure below:



(a) Optimal heuristic pathway through rectangle basin with uniform fleet and non-uniform basin bottom

(b) Optimal exact pathway through rectangle basin with uniform fleet and non-uniform basin bottom

Figure 4.3: Optimal pathways through rectangle basin

For comparison the objective values are compared with each other. However, an important difference between the heuristic and exact optimisation, is the fact that the heuristic approach uses Neumann neighbourhood for dredging equilibrium, while the exact approach uses Moore neighbourhoods. The results is that the heuristic approach underestimates dredging and the exact approach gives a more correct measure for dredging. Hence, the comparison is not fair, but is gives an indication. The best comparison would be to qualitatively reason the outcomes, but the objective values are provided as indication of performance in table 4.2.

The optimal pathway provided by the exact optimisation seems to be the most logical, since it traverses as much along the quay wall to limit dredging and its path is as straight as possible. The heuristic pathway has a larger total pathway length and it does not maximizes its dredging savings from the quay wall. Moreover, the objective value of the exact optimisation is not far of the heuristic objective value, even though the heuristic objective value is an underestimation.

Pathway	Objective value	Computational time
Central pathway (underestimate)	529	-
Heuristic (underestimate)	366	3 minutes
Exact	400	3 hours

Table 4.2: Benchmark 1 objective values

4.1.3. Benchmark 2: Rectangle basin with uniform basin bottom and berth location restrictions

The second case is a rectangular port basin where the berths are only allowed in a small portion of the port basin. Along with the berth restriction, the basin bottom is non-uniform and is deepest on the right basin side. The basin along with the berth locations is depicted in figure 4.4. The fleet is the same as in benchmark 1, described in Table 4.1.

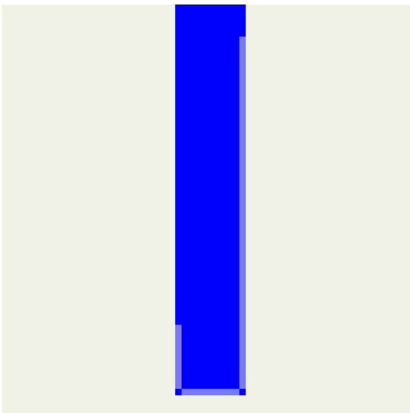
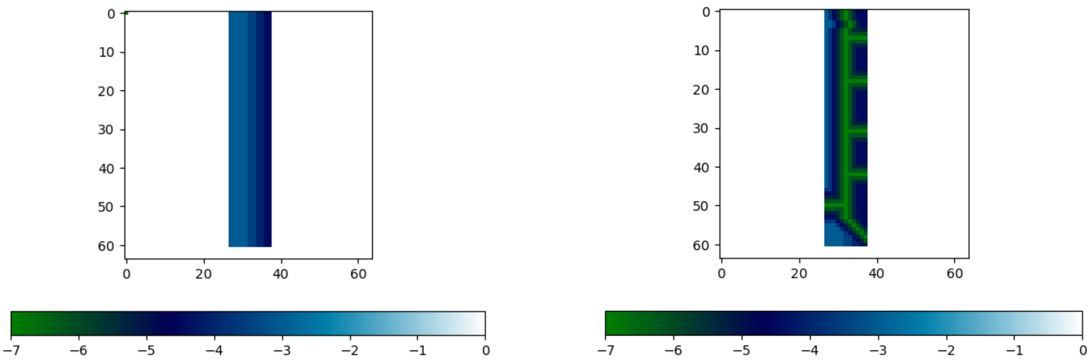


Figure 4.4: The rectangle basin as a benchmark. Purple cells indicate possible berth locations

The optimal berth locations in the basin after 20 million evaluations are depicted in the figure below:

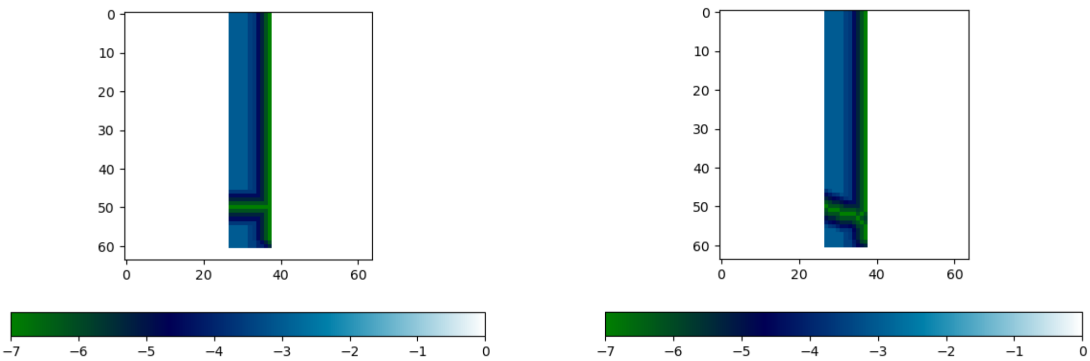


(a) rectangle-shaped basin bottom

(b) Optimised berth locations for uniform fleet with berth restrictions

Figure 4.5: Y-shaped basin bottom and optimised berth locations

The optimal pathway produced by the exact and heuristic approach are depicted in the figure below:



(a) Heuristic pathway for uniform fleet and non-uniform basin bottom.

(b) Exact pathway for uniform fleet and non-uniform basin bottom.

Figure 4.6: Optimal pathways through basin

Both the optimal paths travel through the right side of the basin and the left berth travels to the right side. The difference in pathways between the heuristic and the exact path is the pathway shape; the heuristic pathway goes in a straight line, while the exact pathway sees benefit in traversing along an arced line. The arced pathway even goes further inside the basin, however, this could be due to the pre-solved solution from the MIP solver. Mirroring the arced line towards the basin entrance should provide the same objective value, as the left berth does not gain or lose an advantage due to the quay wall. The objective values are stored in table 4.3. The exact pathway scores the best, even when the other pathway measures are underestimated.

Pathway	Objective value	Computational time
Central pathway (underestimate)	1046	-
Heuristic (underestimate)	513	3 minutes
Exact	510	3 hours

Table 4.3: Benchmark 2 objective values

4.1.4. Benchmark 3: Y-shaped basin with uniform bottom and uniform fleet

The Y-shaped basin differs from the rectangle basin in shape but also in possible berth locations. The benchmark case does not allow for berths to be located in the channel that connects the two basins. Thus, the berths are forced to be located in the port basin ends.

The case consists of a Y-shaped basin with a uniform depth across the basin, depicted in figure 4.7 and 4.8a. The basin should facilitate 5 berths with all a design depth of -7 meter. The possible berth locations in this case are restricted; the main fairway that connects the two side basin does not allow for berth allocation. All of the berths have minimal required distance with respect to each other of 10 units, as described in Table 4.4

Berth no.	Required depth [units]	Minimal distance to other berths [units]
1	-7	10
2	-7	10
3	-7	10
4	-7	10
5	-7	10

Table 4.4: Fleet composition for benchmark case 3

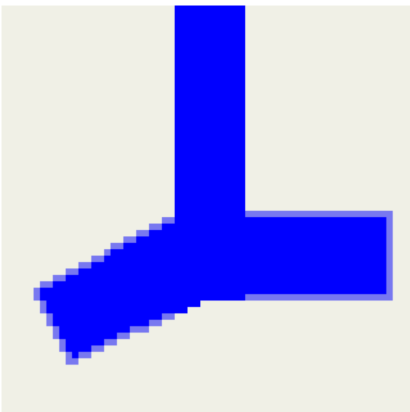
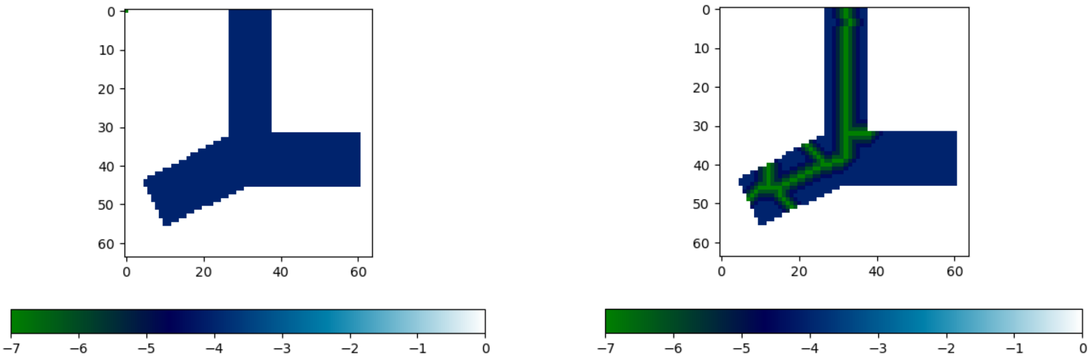


Figure 4.7: The Y-shaped basin as benchmark. Purple cells indicate possible berth locations

The basin and the berths are used as input for the heuristic berth optimisation formulation. The optimisation consist of a population of 20000 genomes and is evolved for 10000 generations, resulting in $2 * 10^8$ evaluations. The algorithm used is the improved harmony search (IHS) algorithm provided by pygmo, using the standard settings. The location optimisation resulted in the following berth locations and pathway:

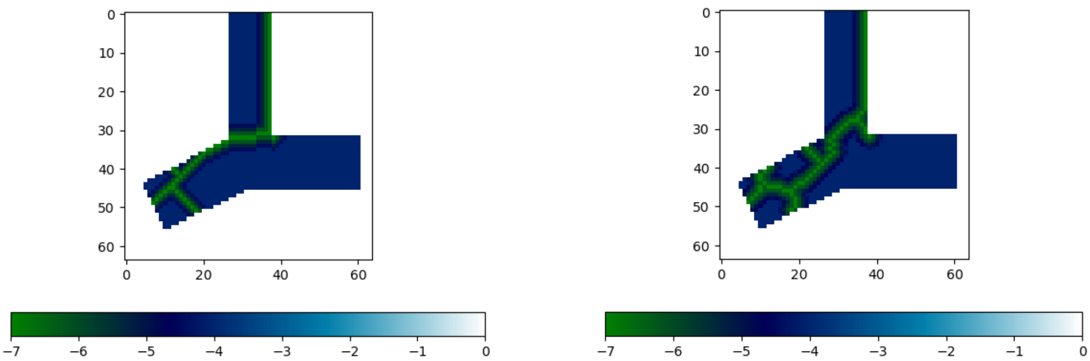


(a) Y-shaped basin bottom

(b) Optimised berth locations for the uniform fleet.

Figure 4.8: Y-shaped basin bottom and optimised berth locations

The optimal pathway produced by the exact and heuristic approach are depicted in the figure below:



(a) Heuristic pathway for uniform fleet and uniform basin bottom.

(b) Exact pathway for uniform fleet and uniform basin bottom.

Figure 4.9: Optimal pathways through Y-shaped basin with uniform fleet and uniform basin bottom

The pathways generated are significantly different from each other. The heuristic pathway prefers to travel as much along the quay walls as possible, even if it means that one berth's path has to cross the entire basin. The exact approach, however, does not see the advantage of moving along the quay walls as much as possible, but prefers a central path through the left basin that is angled towards the path of the most right berth.

This case displays interesting behaviour in the exact approach. It is hard to say whether the exact pathway is wrong, because it does perform well in the other benchmarks and it does show optimal behaviour in its pathways. The following factors could play a role in this result:

- The optimisation failed to converge during the time-limited run. The reason could be the lack of sensitivity in changing the pathway to objective value improvement.
- The optimisation is stuck in a local optimum. The optmisation uses a 'warm start' as in heuristic algorithms pre-solved the solution for the MIP solver to solve exactly
- The difference in berth design depth and initial depth allows for limited quay wall benefits to pathway dredging. It considers a combined pathway more valuable than pathways along a quay wall.

In terms of objective value, the heuristic approach provides the best result, but the result is an underestimation compared to the exact approach's formulation for dredging. The results are in table 4.5

Pathway	Objective value	Computational time
Central pathway (underestimate)	858	-
Heuristic (underestimate)	591	3 minutes
Exact	775	3 hours

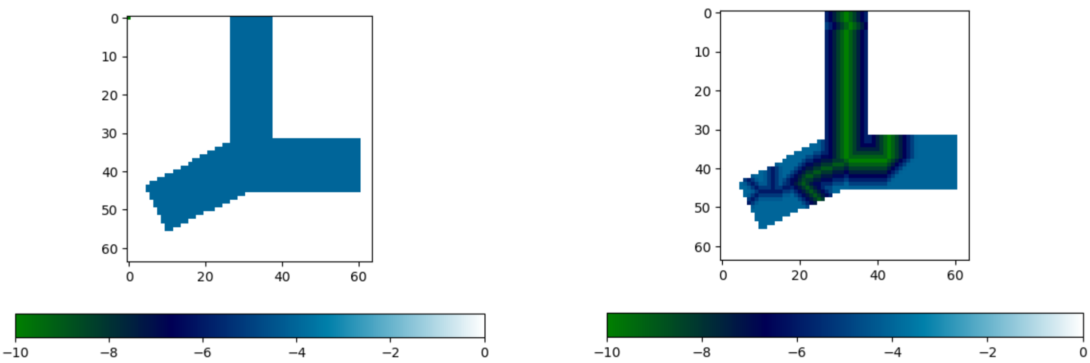
Table 4.5: Benchmark 3 objective values

4.1.5. Benchmark 4: Y-shaped basin with uniform bottom and divers fleet

The port basin environment is the same as benchmark 3, shown in figure 4.7. The difference in this case is the fleet composition. The fleet is composed of 5 vessels with different depths and distance constraints, as described in Table 4.6. The basin bottom and optimal berth locations are depicted in the figure below:

Berth no.	Required depth [units]	Minimal distance to other berths [units]
1	-10	20
2	-9	10
3	-6	5
4	-6	5
5	-6	5

Table 4.6: Fleet composition for benchmark case 4

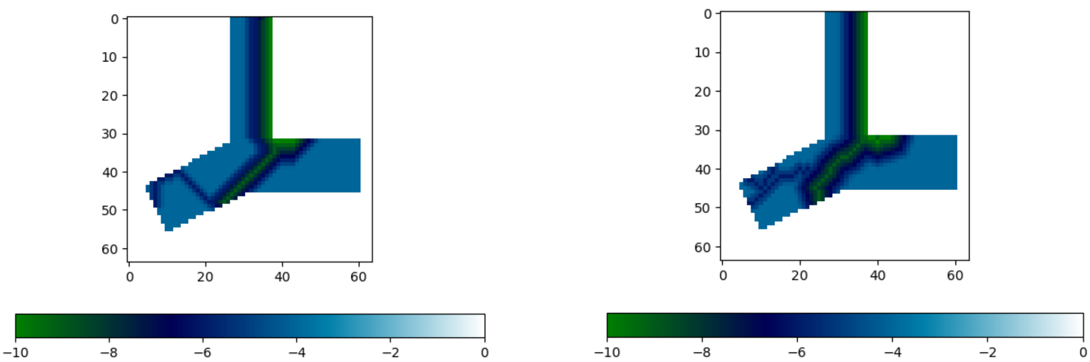


(a) Y-shaped basin bottom

(b) Optimised berth locations for the non uniform fleet.

Figure 4.10: Y-shaped basin bottom and optimised berth locations

The optimised pathways through the basin are shown in figure 4.11



(a) Heuristic pathway for non-uniform fleet and uniform basin bottom.

(b) Exact pathway for non-uniform fleet and uniform basin bottom.

Figure 4.11: Optimal pathways through Y-shaped basin with non-uniform fleet and uniform basin bottom

Looking at the results produced, the exact pathway provides a more logical optimal route. However, the heuristic pathway maximizes the benefit provided by the quay walls, even if it means that the pathway in itself will be longer. The exact pathway prefers a pathway that traverses through the middle of the basin. Thus, the exact pathway does not see the benefit of traversing around the basin perimeter compared to traversing using the shortest route.

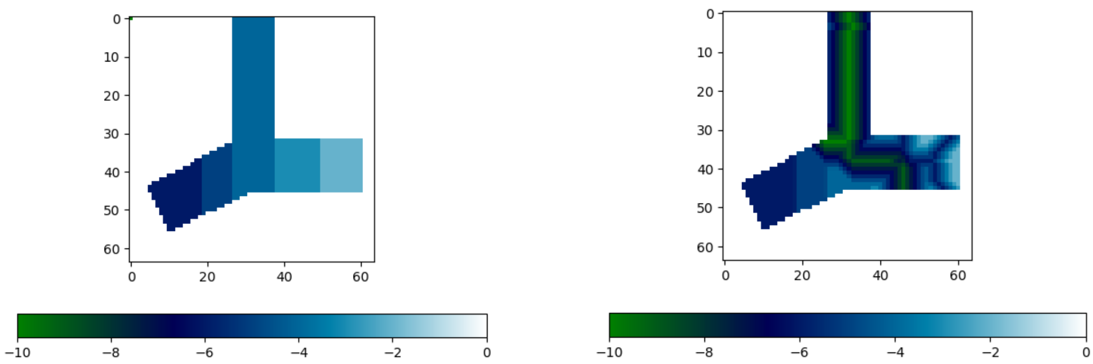
Another observation of the exact optimisation is the 'wiggle' in the pathway. This phenomenon can be explained by the lack of convergence of the solution. The optimisation was done in 3 hours, which seems to be too little for the pathway optimisation.

Pathway	Objective value	Computational time
Central pathway (underestimate)	2665	-
Heuristic (underestimate)	1451	3 minutes
Exact	1726	3 hours

Table 4.7: Benchmark 4 objective values

4.1.6. Benchmark 5: Y-shaped basin with non-uniform bottom and divers fleet

The port basin environment is the same as benchmark 3, shown in figure 4.7. The difference in this case is the fleet composition and the basin bottom. The fleet is composed of 5 vessels with different depths and distance constraints, like case 4, as described in Table 4.6. The basin bottom is deeper in the left basin and shallower in the right basin. The basin bottom and optimal berth locations are depicted in the figure below:

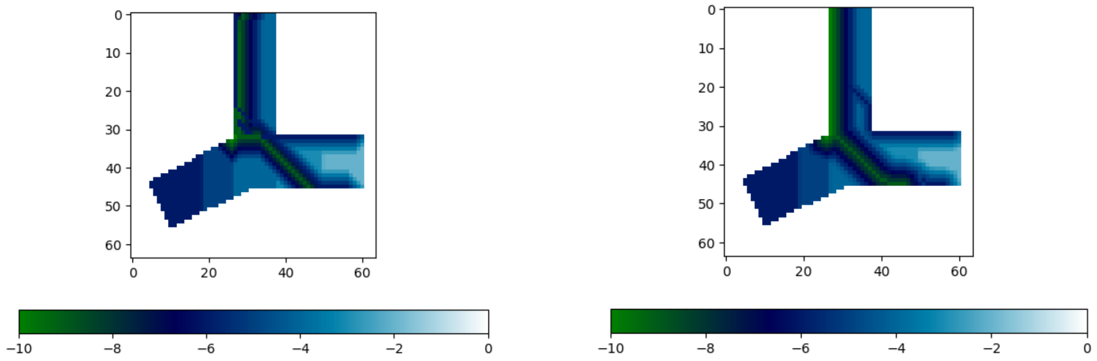


(a) Y-shaped basin with non-uniform bottom.

(b) Optimised berth locations non-uniform fleet.

Figure 4.12: Y-shaped basin with a non-inform fleet and a non-uniform bottom

The optimised pathways through the basin are shown in figure 4.13



(a) Heuristic optimal pathway through the basin with a non-uniform basin bottom with non-uniform fleet.

(b) Exact optimal pathway through the basin with a non-uniform basin bottom with non-uniform fleet.

Figure 4.13: Optimal pathways through Y-shaped basin with non-uniform fleet and non-uniform basin bottom

The heuristic pathway shows the most remarkable behaviour, as the depth to the basin entrance is not deep enough for the deepest design depth of -10 meters. This is due to a bug in the algorithm, hence the result is not valid. The exact pathway does provide a valid solution. The pathway also maximizes the benefit of the quay wall. The berth with -9 depth sees the benefit of moving along the quay wall and then creating the shortest route to the berth of -10 meters, compared to picking the shortest path to the basin entrance. The lack of convergence does play a role again; the berths located at the top of the right basin have a route that crosses the middle basin, instead of latching onto the pathway of the -9 depth berth.

Pathway	Objective value	Computational time
Central pathway (underestimate)	2637	-
Heuristic (underestimate)	1600	3 minutes
Exact	1904	3 hours

Table 4.8: Benchmark 5 objective values

4.2. Sensitivity analysis

This section provides a sensitivity analysis of the exact optimisation approach. The goal of this analysis is to see how the pathway behaves for different basin geometries, vessel depths, and stability slope ϕ. The topic of interest is the evolution of the pathway and the respective objective value. This is studied by moving berth locations further from the entrance of the basin and changing parameters to study the formation of a central pathway. The study is inspired by the previous section, where benchmarks 3 and 4 showed interesting pathway behaviour. Each of the cases is analysed separately and the end conclusions are presented.

The reason only the exact pathway is studied is due to the nature of the heuristic pathway. The heuristic pathway approach can never formulate a central pathway. The heuristic approach in essence answers the question: is it more beneficial to latch onto an existing pathway, or to create a separate pathway to the basin entrance? Since berths are located near the side of the basin, it will never consider a combined central pathway with other berths.

As stated before, the locations of the berths are moved through the basin to study the formation of a central pathway. The basin used for this analysis is simple; it is the same basin as presented in benchmark 1 (rectangular basin with uniform bottom). However, in this study, we only consider 2 berths which are located on the opposite side of the basin.

4.2.1. Berth location sensitivity

The berths start at a distance of 5 units to the basin entrance. Every iteration, the berths are both moving 5 units further from the basin next to the quay wall. The stability slope ϕ and berth depth δ_i are constant and are 40°and -7 units respectively. These parameters and the berth locations form the standard for the other sensitivity analyses. Figure 4.14 displays the evolution of the pathway and figure 4.14 plots the objective value per berth location. The location is measured in unit distance from the basin entrance. Table 4.9 presents the objective values and figure 4.15 plots the values.

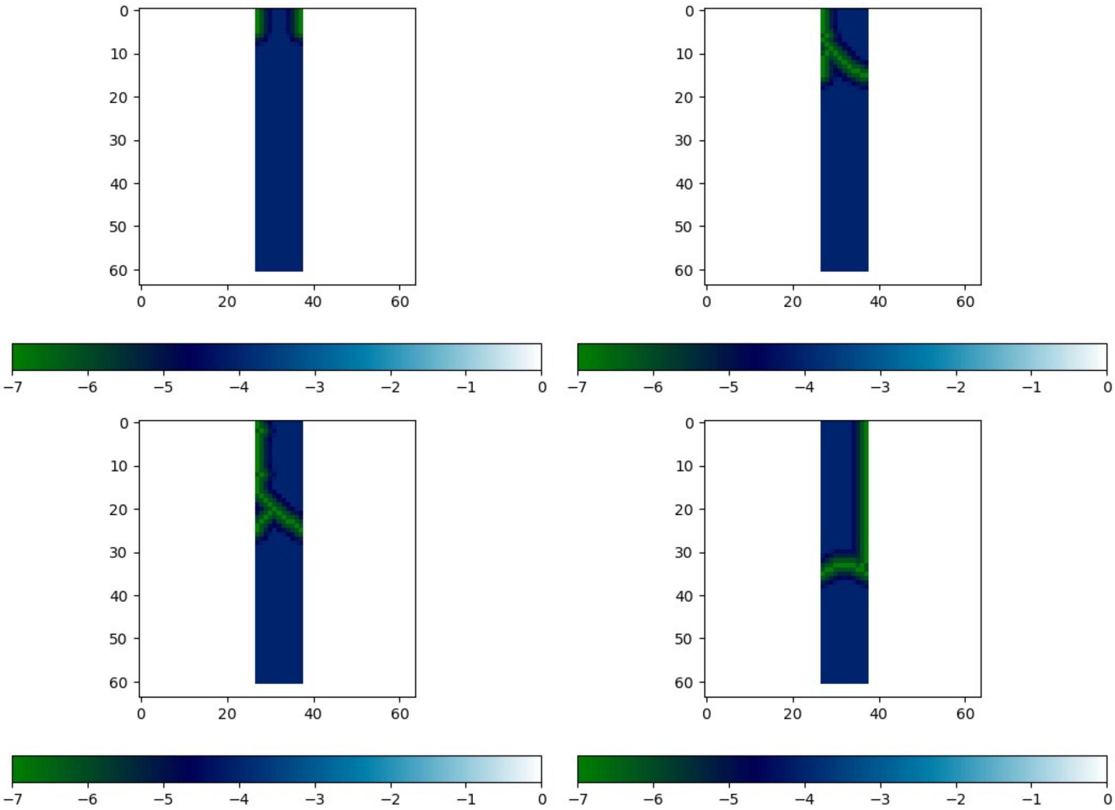


Figure 4.14: Optimal pathways for 4 different locations

Location [m]	Obj.
5	100
15	226
25	303
15	354

Table 4.9: Objective value of exact pathway per berth location.

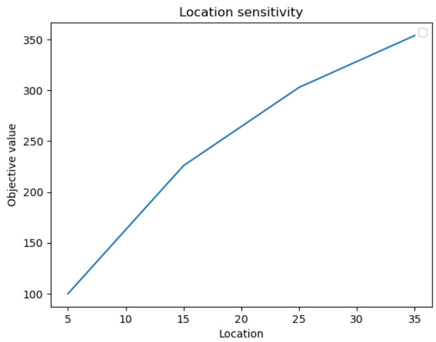


Figure 4.15: Graph plotting the objective values over the different locations

Judging by the data and plots provided, the algorithm quickly sees benefit of combining pathways to the berths. However, it decided to not formulate a central pathway, but it preferred to traverse to one side and then head out of the basin. This is logical, since the quay wall provides dredging protection. Thus, the pathway is sensitive using the current parameters to the berth locations.

4.2.2. Basin width sensitivity

The parameters set for the basin are the same as in the previous analysis. However, this time the basin width changed. The widths are set to 10, 15, 20, 25 units. The results produced by the exact optimisation algorithms are depicted in figures 4.16 - 4.19. The objective values are stored in table 4.10 and plotted in figure 4.20.

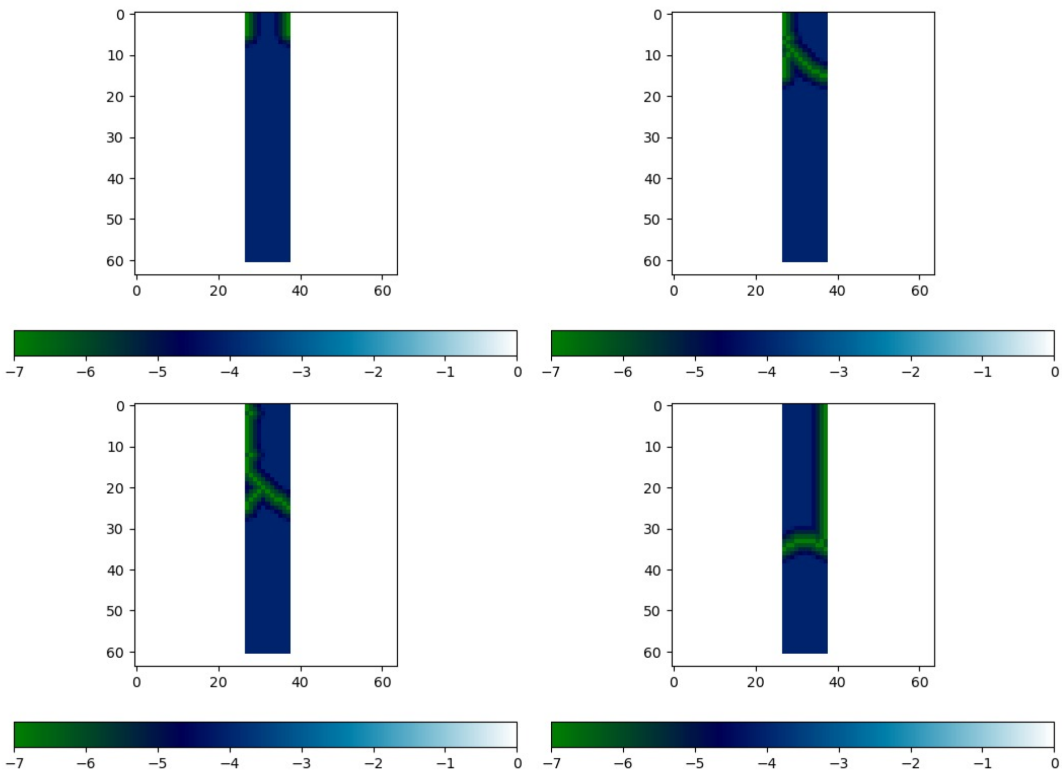


Figure 4.16: Optimal pathways for 4 different locations and basin width 10

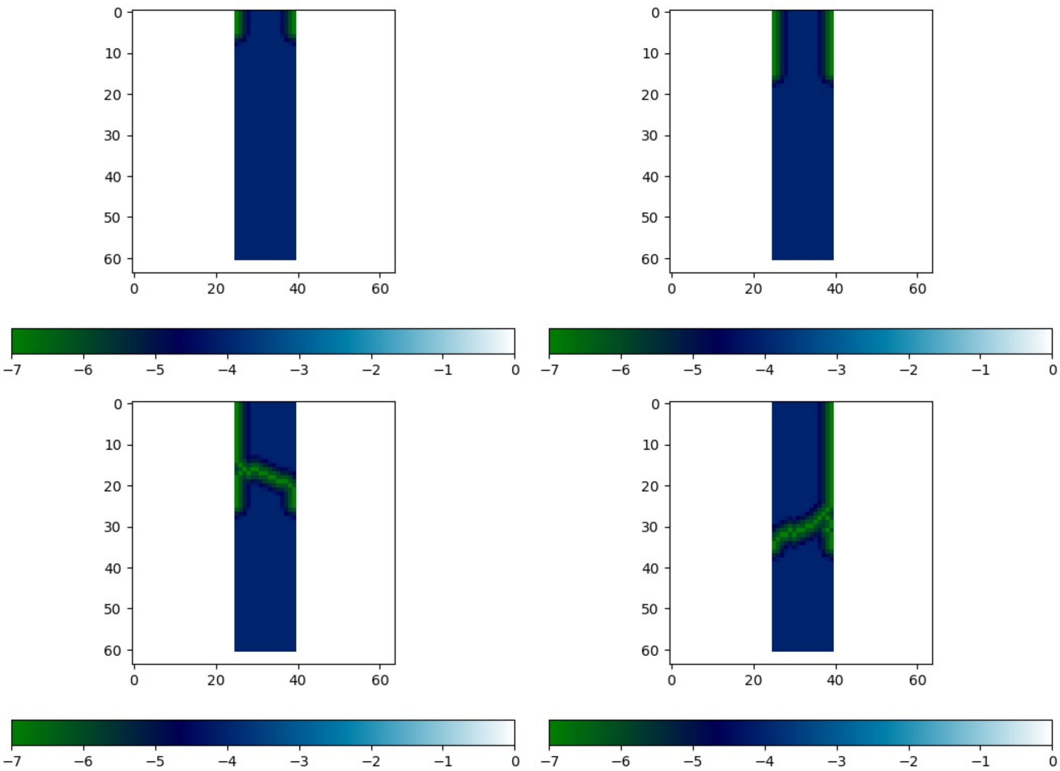


Figure 4.17: Optimal pathways for 4 different locations and basin width 15

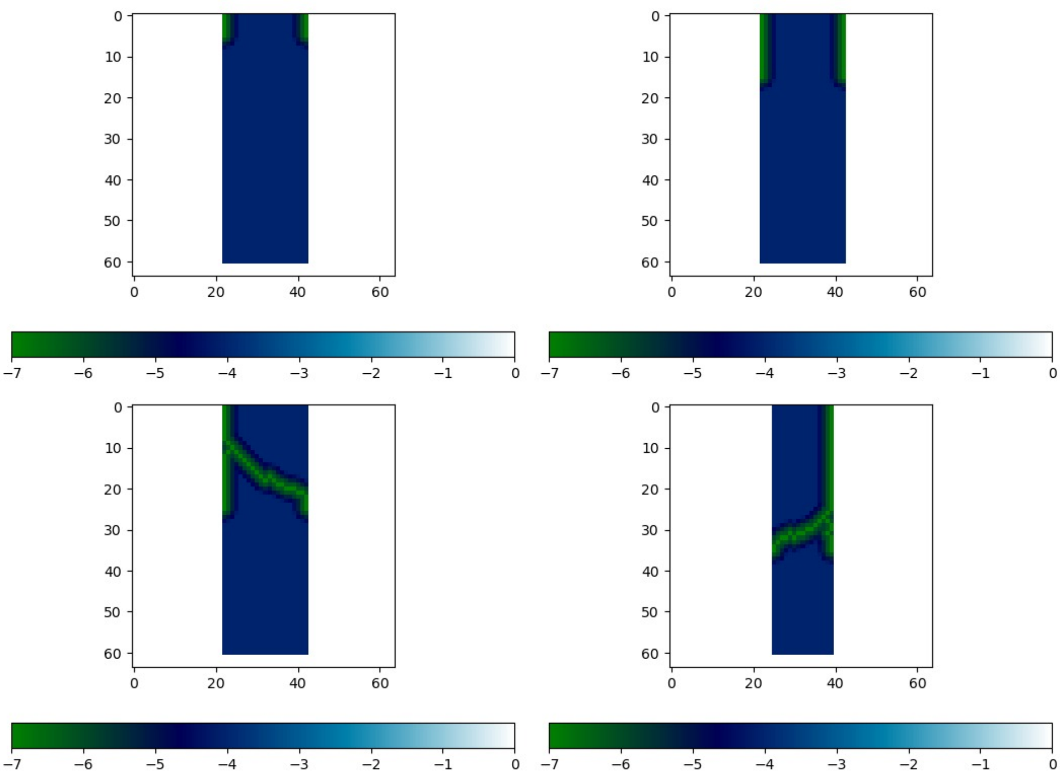


Figure 4.18: Optimal pathways for 4 different locations and basin width 20

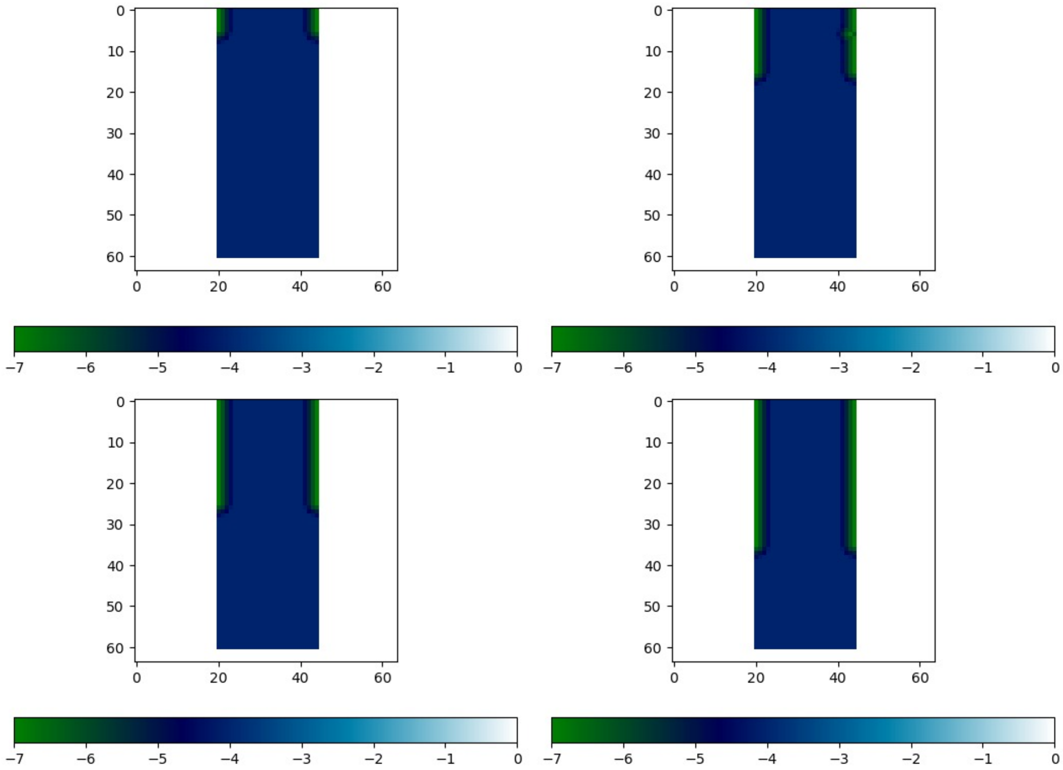


Figure 4.19: Optimal pathways for 4 different locations and basin width 25

Location [m]	Obj. W = 10	Obj. W = 15	Obj. W = 20	Obj. W = 25
5	100	100	100	100
15	226	240	240	244
25	303	363	442	379
35	354	419	502	518

Table 4.10: Objective value of exact pathway per berth location with different widths

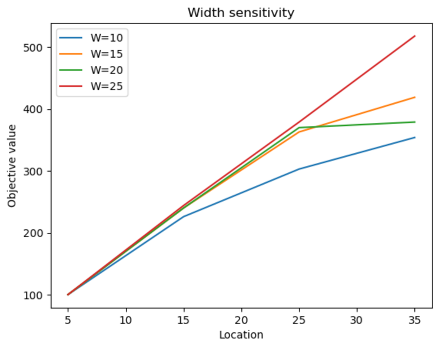


Figure 4.20: Graph plotting the objective values over the different locations and different widths

The optimisations show that the pathway is sensitive to the width of the basin. When the basin gets wider compared to the length of the basin, the optimal pathway is no longer a combined path. For width 25 units the basin pathway only prefers a combined path for berth far up the basin. Even though the pathways are sensitive to the basin width, the objective value shows it is not influenced as strongly.

4.2.3. Berth depth requirement sensitivity

The parameters set for the basin are the same as the berth location analysis. However, this time the berth design depths change. The depths are set to -6, -7, -8, and -9 units. The results produced by the exact optimisation algorithms are depicted in figure 4.21 - 4.24. The objective values are stored in table 4.11 and plotted in figure 4.25

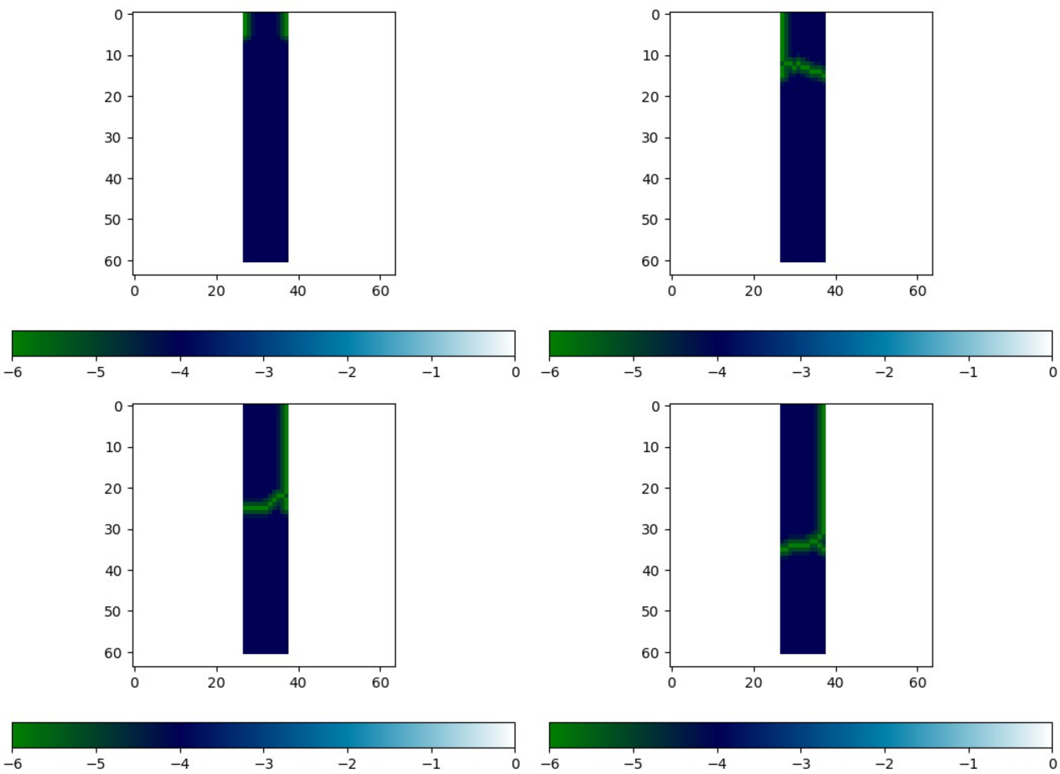


Figure 4.21: Optimal pathways for 4 different locations and berth depth -6

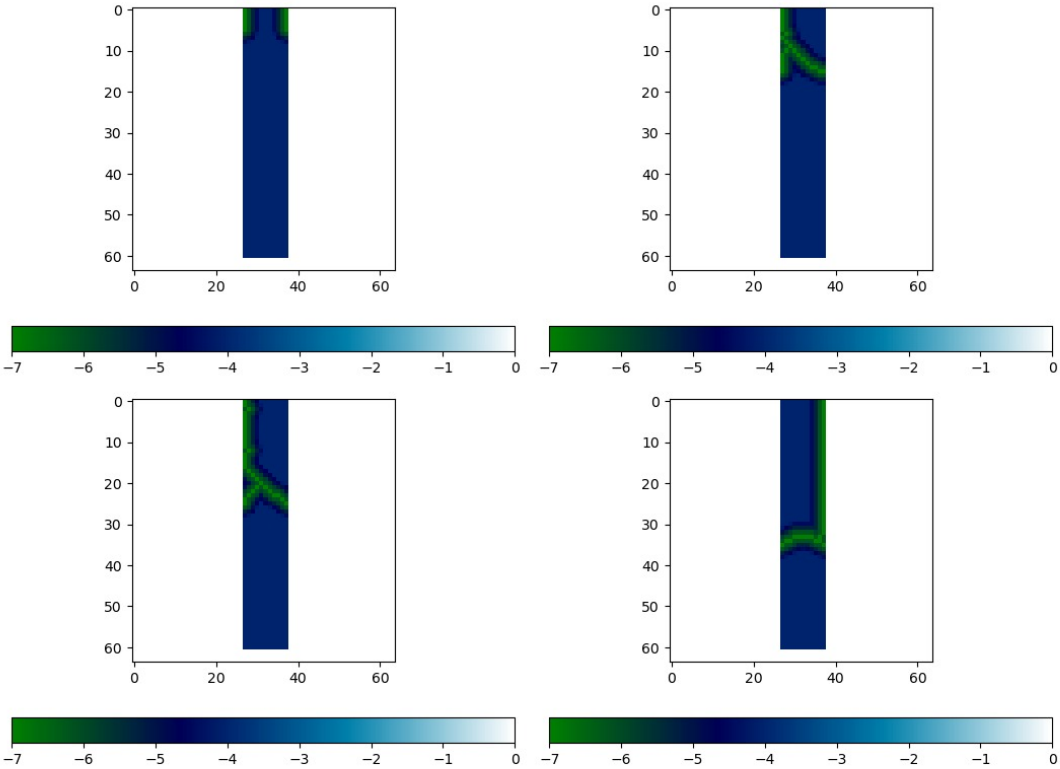


Figure 4.22: Optimal pathways for 4 different locations and berth depth -7

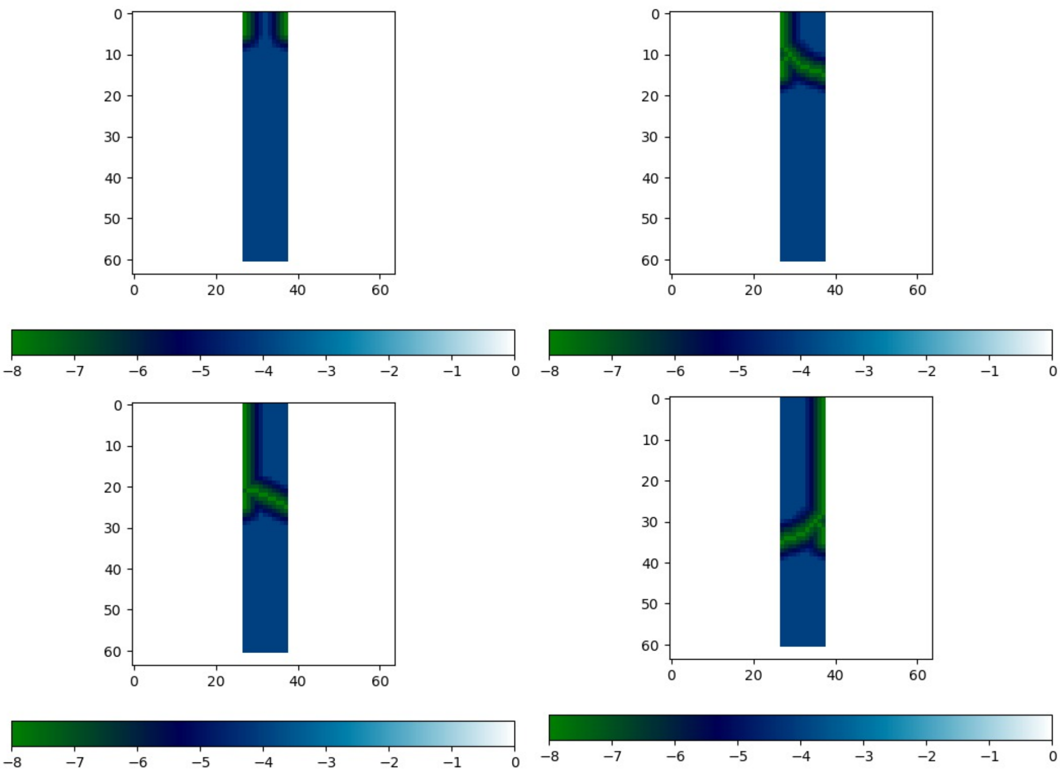


Figure 4.23: Optimal pathways for 4 different locations and berth depth -8

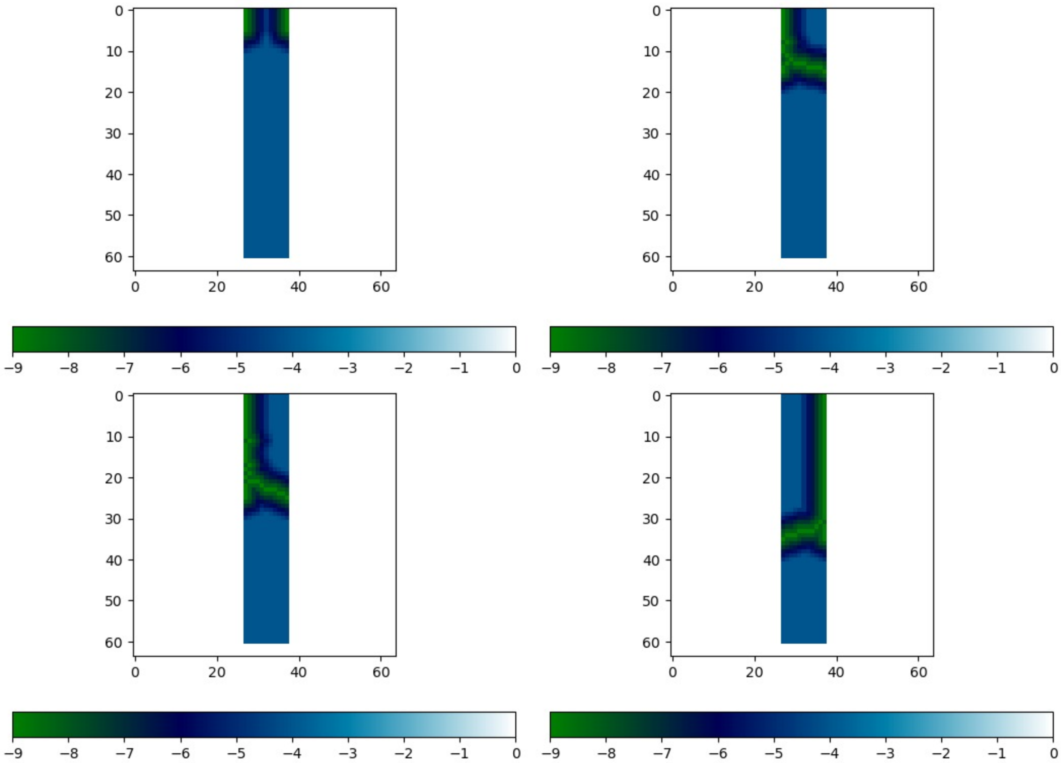


Figure 4.24: Optimal pathways for 4 different locations and berth depth -9

Location [m]	Obj. δ= -6	Obj. δ= -7	Obj. δ= -8	Obj. δ= -9
5	46	100	181	285
15	105	226	364	537
25	138	303	478	730
35	172	354	596	879

Table 4.11: Objective value of exact pathway per berth location.

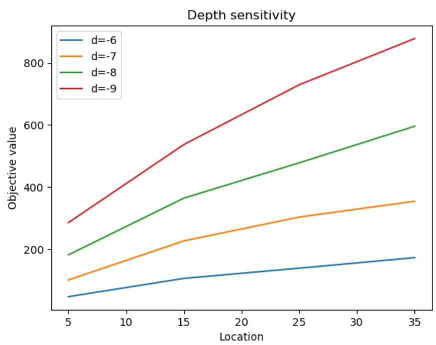


Figure 4.25: Graph plotting the objective values over the different locations and different depths

By comparing the columns, it can be concluded that the objective is very sensitive to the design depths of the berth. A depth increase of 1 meter comes close to doubling the objective value for small depths and gets smaller, but still significant, for larger depths.

4.2.4. Stability slope sensitivity

The parameters set for the basin are the same as the berth location analysis. However, this time the stability slope changes. The stability slopes are set to 30, 35, 40 and 45 units. The results produced by the exact optimisation algorithms are depicted in figure 4.26 - 4.29. The objective values are stored in table 4.12 and plotted in figure 4.30.

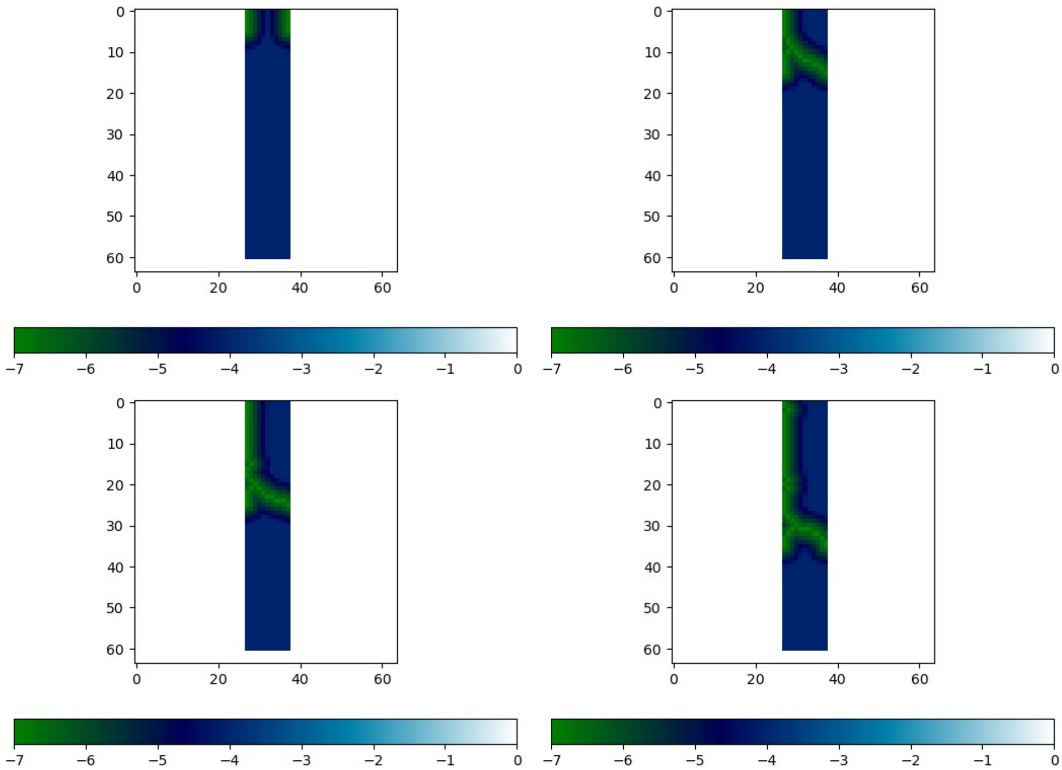


Figure 4.26: Optimal pathways for 4 different locations and ϕ is 30 degrees

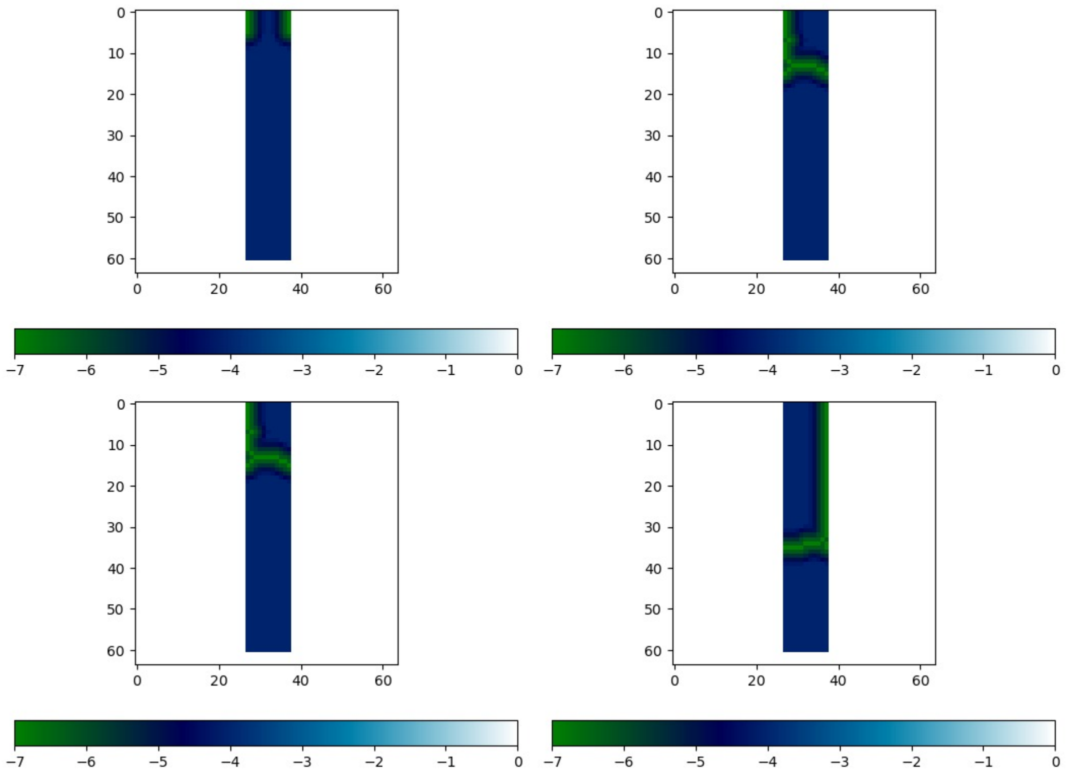


Figure 4.27: Optimal pathways for 4 different locations and ϕ is 35 degrees

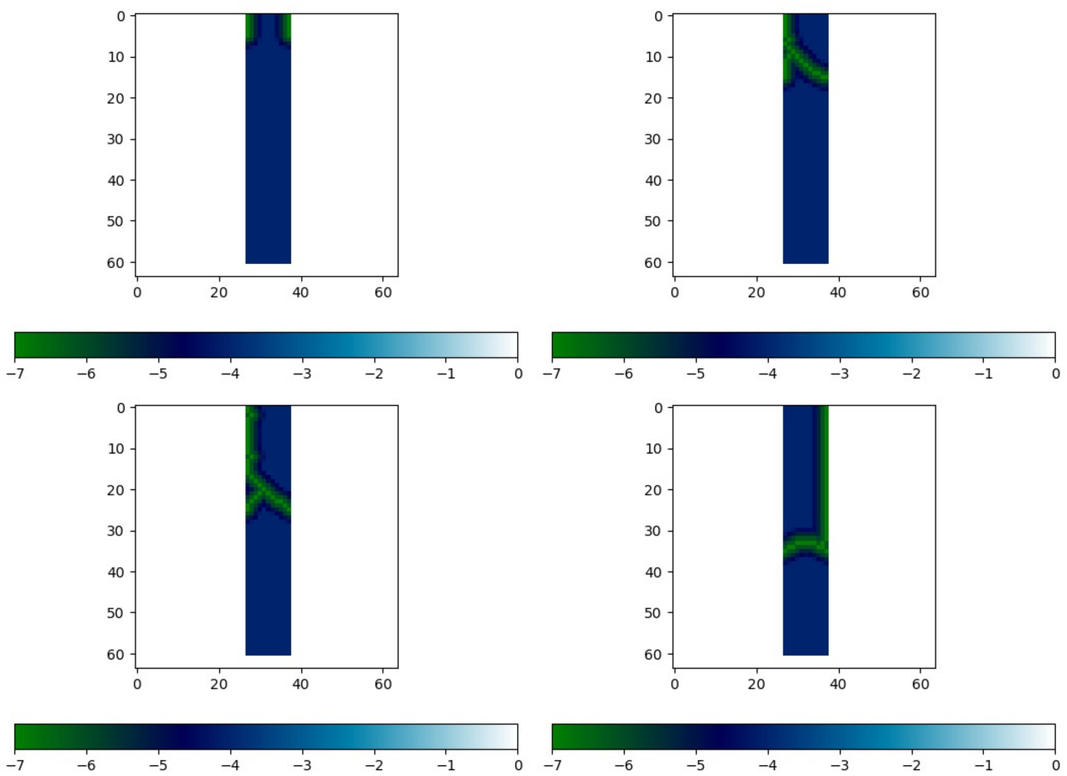


Figure 4.28: Optimal pathways for 4 different locations and ϕ is 40 degrees

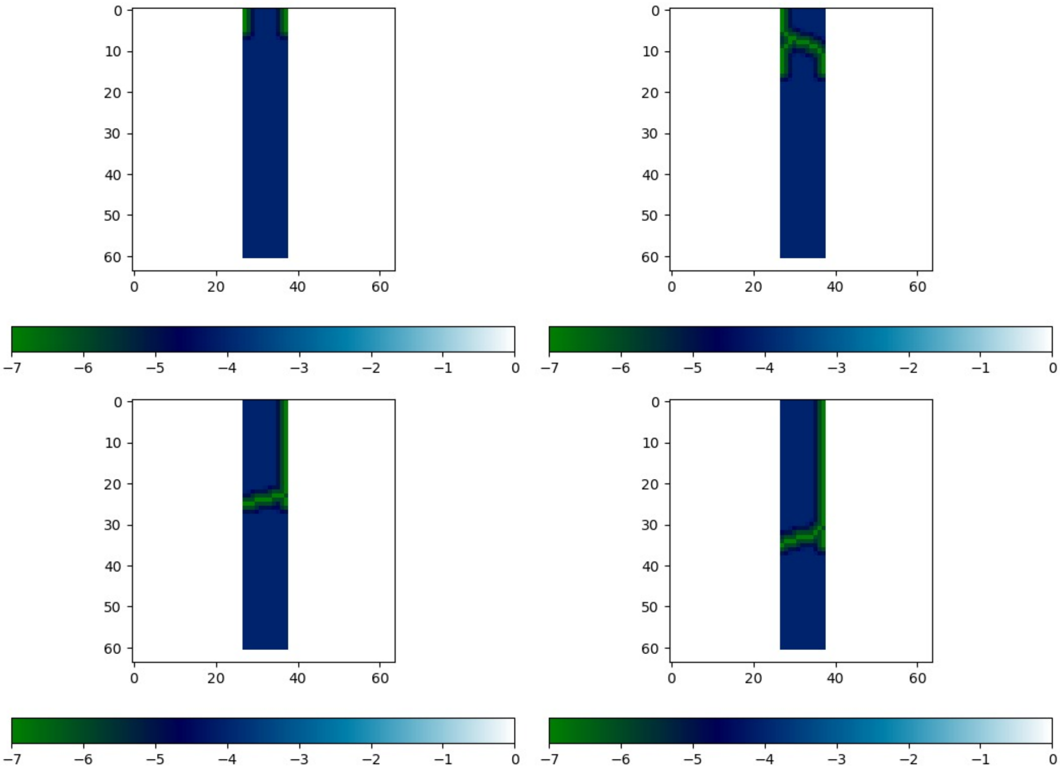


Figure 4.29: Optimal pathways for 4 different locations and ϕ is 45 degrees

Location	Obj. φ= 30	Obj. φ= 35	Obj. φ= 40	Obj. φ= 45
5	148	121	100	83
15	297	250	226	213
25	390	325	303	243
35	502	404	354	305

Table 4.12: Objective value of exact pathway per berth location with different stability slopes φ

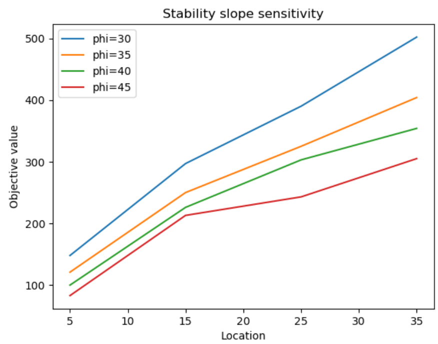


Figure 4.30: Graph plotting the objective values over the different locations and different stability slopes

The stability slope exercises influence on the objective value. A larger φimplies a smaller objective value. This makes sense, since an increase in stability slope allows for basin bottoms to be steeper and, hence, be cheaper to dredge.

The pathway is also slightly influenced by the stability slope. Figure 4.26 shows pathways that are more 'fluid', in the sense that there are little sharp corners. The reason for this is the dredging influence region that attracts the pathway gradually. The pathway benefits from the dredged region by traversing to it, while moving to the quay wall and minimizing the total pathway length.

4.2.5. Sensitivity analysis outcomes

To conclude the sensitivity analyses, the most important outcomes are summed up in the list below:

- The location of the berths increase the objective value. This increase is not completely linear due to the pathways merging.
- The objective is most sensitive to the depths of the berths. For every meter depth extra, the objective value increases significantly. This increase does reduce between the deeper berths.
- The stability slope of the soil allows for pathways that are slightly arced when joining another pathway. The reason for this is that the pathway tries to both find a short route while benefiting as much from the dredging influence region of the other pathway.
- The pathway is most sensitive to the width of the basin. As the basin gets wider, the optimal pathway is no longer a combined pathway.

5

Scheurhaven case optimisation

5.1. The Scheurhaven case

The Scheurhaven is a small port basin located in the port of Rotterdam, as depicted in Figure 5.1. The port basin originally was meant as a working port for the construction of the Maasvlakte. However, after construction, the port basin had little use, since the construction was over. Over time the port basin was used as a leftover spot: if an unassigned vessel required a spot, the Scheurhaven could be an option to lay berth. The position of the port basin was favourable for tugboats, since they require to be close to the Port of Rotterdam entrance to assist incoming vessels fast. Hence, the tugboats were a suitable client for the Scheurhaven. Over time the berths in the port basin grew naturally, because of the 'leftover berth' idea that reigned in port planners' minds. The consequence of this natural growth is that the basin does not have the optimal layout and it is hard to get to the optimal layout, since re-assigning berths requires contracts to either expire or to be renegotiated with several parties. The lack of strategy for the port makes it hard to optimize the port basin layout. For the Scheurhaven a plan is made to give the port basin a defined purpose, along with the creation of additional berth locations for Fairplay tugboats.

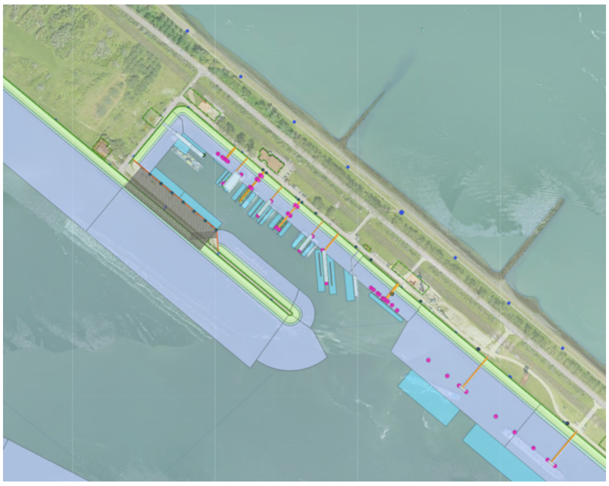


Figure 5.1: Map of the Scheurhaven

The basin now facilitates a variety of clients, thus vessel types. Most berths are assigned for tugboats from Boluda and the other berths are for Evides (drink water company), the Rowers Association KRVE, and more. The problem that came to rise recently is the tugboats 'bumping' into the slope of the Scheurhaven. Tugboats in itself are quite robust, hence only the slope of the embankment is at risk. By bumping into the embankment, the embankment loses material which can impose a risk on the stability of the slope.

A fundamental question to this problem: why is the bumping happening? The origin of the problem lies in the growing size of the tugboat vessels. In the past tugboats had a B: LOA (Beam to Length overall) ratio of 1:4. Contemporary vessels are getting closer to a 1:3 ratio and it is expected that the vessels will even reach a 1:2 ratio, resulting in so-called 'doughnuts'. Along with the ratio change, the vessel lengths have also increased, thus bumping into the embankment, because the vessels have outgrown the current berth sizes.

A growing vessel size means that it also requires more space to maneuver itself through the basin. The problem with a port basin is the stiffness of the dimensions: a port basin perimeter is expensive to adjust, since it requires construction of elements and site excavation. Both cost a lot of money and result in downtime in the port basin, which requires compensation for the contracted companies inside the port basin. A solution that requires these major interventions is not preferred by the port authority and if a solution can be found within interventions inside the port basin, without affecting the port basin perimeter and the adjacent land, it would be preferred. Changes in the port basin perimeter would result in the roads being rearranged, which affects the accessibility of the basin.

The land around the basin perimeter is occupied by roads, a small office building, a house, and a pair of wind turbines. Sacrificing these buildings would result in settlements between the port authority and local stakeholders, which drive the costs up, an unfavorable outcome for the port authority. The wind turbines impose a different type of constraint on the environment. The wind turbines have a 'safety contour' (circular area) around them in which people are not allowed to stay for longer than an hour. The circular area of the safety zone stretches to the water area, which implies constraints on the berth locations.

Apart from the safety constraints, the port handles a guideline for designing port basins. The guidelines prescribe mainly geometric constraints for port basins: berths require a certain distance from one another, a guaranteed depth, distance from embankments, and more requirements stated in the guideline. These constraints are based on operational and safety philosophies.

Tackling all the constraints without doing an intervention in the system seems nearly impossible. Thus, interventions, such as building quay walls, retaining walls, embankment slope improvements, and more, are allowed to be used. The downside is that every intervention comes at a, literal, cost, either monetary or emission-wise. Along with the direct costs is the cost of operational downtime and compensation that goes hand-in-hand with construction. The important question that arises with building these interventions is: is it worth to implement such an intervention? In most cases it is more worthwhile to adjust the system slightly instead of implementing major changes on the system when accomplishing the same result. The key concept in this question is trade-offs; how does an adjustment in the system affect the rest of the system? These trade-offs happen in multiple dimensions (cost, space, connectivity, etc.), which makes the problem hard for humans to comprehend. The more dimensions considered, the harder it is to determine what is the optimal result for the situation.

5.2. Goal of the case

The goal is to facilitate nine futureproof berths for tug boats of Boluda in the Scheurhaven. The berths are required to facilitate tug boats of size 35x15x6,5 (LxBxD). Other berths inside the port basin remain unaffected by the improved/new berths.

5.3. Port basin layout requirements

5.3.1. Geometric layout requirements

Before determining the costs and emissions, the design has to be deemed 'valid'. A design is valid when it satisfies the constraints given. For designing port basins, the validity is determined by many different factors from many disciplines. The first document for designing port basin layouts is 'Ontwerprichtlijnen havens en vaarwegen'. The relevant constraints given in this document are:

- Required width of the port basin
- Required depth for vessel
- Required turning basin diameter
- Required respective vessel distance
- Required distance from embankments
- Required distance margin for tugboat assistance (if applicable)
- Required minimal distance from wind turbines

Along with these given constraints, constraints that are deemed 'common sense' should also be set. These constraints are:

- Berths should be located inside the basin, in the water area
- Berths are not allowed to overlap
- No overlap between berths and constructions is allowed
- Berths are required to be attached to a 1) quay wall or 2) jetty 3) pontoons.
- The pathing to a berth requires a depth equal to the design depth of the vessel allocated to the berth
- The berths don't obstruct the pathway

5.3.2. Port basin construction requirements

The mentioned constructions also require some validity checks; a round quay wall or hexagonal pontoons are not realistic. So every construction has a special set of constraints, which will all be addressed. First, a quay wall should be:

- A straight line or rectangular (continuous)
- Long enough to facilitate respective design vessel
- Built on the wet perimeter of a port basin
- Built as an interface between water and land
- Built to a certain depth to provide stability

A pontoon:

- Must be built into the water
- Requires a certain design depth to not hit the bottom
- Must be a line or rectangular (continuous)
- Requires a connection (bridge) to the land head
- Should be long enough to facilitate the respective vessel

The port basin design only contains pontoons and quay walls which are of interest. Therefore other elements, such as jetties, are not considered.

5.3.3. Basin bottom requirements

Designing a port basin layout imposes changes to the environment to facilitate the design vessels. Hence, the bottom of the port basin will also be adjusted. Since the bottom profile can be seen as a function of the berths inside the basin, the major requirement is the stability of the bottom after the intervention.

5.4. Case scope for optimisation

The scope of the case is mostly equal to the research scope. It mainly concerns finding the optimal pathway for the berths in a basin as its optimisation. Thus, structures are not taken into account for the optimisation. The consequence of this assumption is that a large part of the optimisation is taken as an assumption. If the method is applied to the basin without the structures, the outcomes would not be comparable at all. Hence, the berth locations are set according to the design produced by the port of Rotterdam Authority. The only optimisation left would be the pathway optimisation.

To conclude, the case scope would be to optimize the pathway for the set of Boluda berths, which locations are set for the optimisation.

5.5. Environment modeling

Before optimisation, the environment has to be set up. To start, an environment matrix of 256 x 256 elements is created with each element representing a grid cell. In order to make the Scheurhaven fit in the matrix, the scale is set as 1 cell is 2x2 meter. This makes the Scheurhaven basin with a length of approximately 310 x 90 meters fit in the matrix. The water body in the matrix is drawn using the OpenCV fillPoly function and assigning the water body a value of 0 and the land cells a value of 1. The resulting basin:

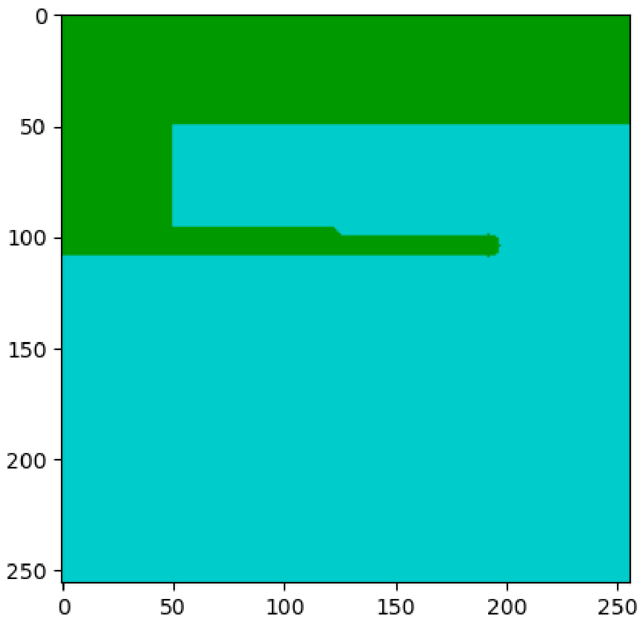


Figure 5.2: Scheurhaven modeled in the environment matrix. Green is the land side and blue is the waterside

The pathway optimisation is done based on the berths set by the Port of Rotterdam design. The design for the Boluda tugboat berths in the basin is depicted in the figure below:

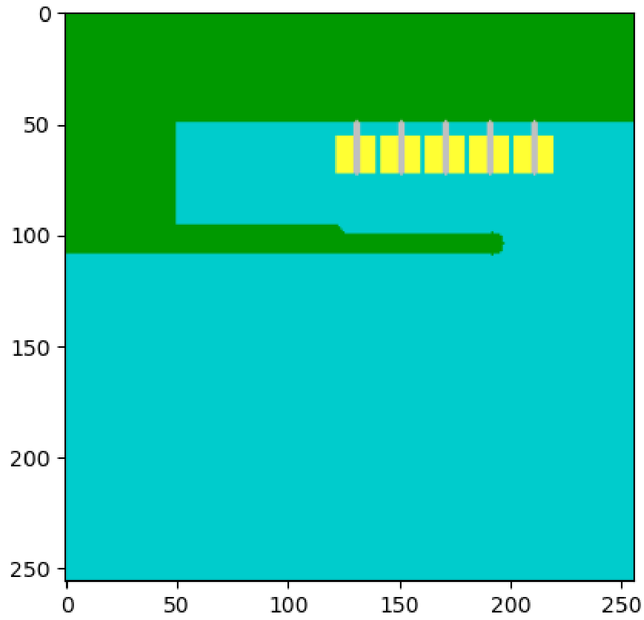


Figure 5.3: The Boluda tug berths are attached to a pontoon indicated by the gray line. The berths are in yellow

The next step is to model the basin bottom of the scheurhaven. Since the original Scheurhaven design was made in 2020, those depths will be taken into account. The basin bottom, by estimation, has the following depth profile:

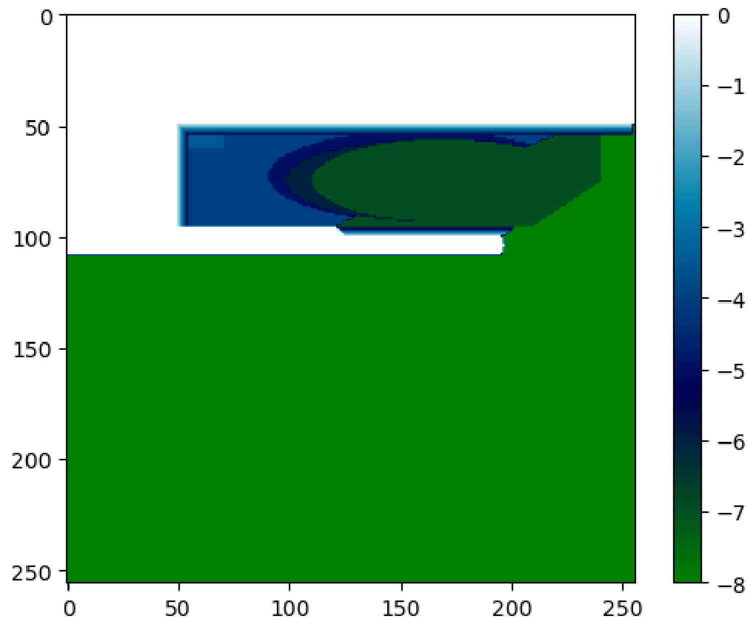


Figure 5.4: Scheurhaven depth profile in 2020, by estimation based

5.6. Pre-processing for optimisation

The created environment has to be translated into a graph for the optimisation. The graph is made by checking the Moore neighbours for each cell. The result is an adjacency matrix of dimensions 65536 x 65536 with each column and row indexed representing a grid cell. This adjacency matrix can easily be translated into a graph by means of the networkx library in python, using the 'from_scipy_sparse' function. The graph is denoted as G(V, E) with V being all the vertices and E being all the edges. Note that the weight of the edges in the graph represents the relative distance between cells. In this case, the grid cells are 2 x 2 meters, thus the unit distance has to be scaled by a factor of 2.

The depths of each vertex (grid cell center) are stored in a vector of length 65536 with each position coinciding with indices in the graph. This vector is called the depth vector, denoted as:

$$\mathbf{d_V} = \{d_{V,0}, d_{V,1}, d_{V,2}, ..., d_{V,65533}, d_{V,65534}, d_{V,65535}\}$$

The scope of the case is on the waterborne side of port planning and no adjustments are allowed to the land vertices. For the sake of the dredging algorithm and computational speed, the land nodes have to be expelled from the graph. The set of vertices consists of land vertices, V_L, and water nodes, V_W. The land nodes are the nodes with depth 0, and the water vertices have smaller depths. The depths are with respect to the water level with the negative values being below water level and positive values being above water level. Finally, the nodes are either water vertices or land vertices, no overlap in identities exists. With these characteristics, the following identities hold:

$$V_L = \{i \ : \ d_{V,i} \geq 0\}$$

$$V_W = \{i \ : \ d_{V,i} < 0\}$$

$$V_L \cup V_W = V$$

$$V_L \cap V_W = \emptyset$$

The edges have to be filtered accordingly, leading to the following notation:

$$E_W = \{(i,j) \ ; \ \forall_i \forall_j \in V_W\}$$

The considered graph is:

$$G(V_W, E_W)$$

5.7. Optimisation set-up and result

The environment and graph are set up to facilitate optimisation. Now some parameters have to be set for the optimisation, such as:

- The starting points of the berths for the pathway optimisations
- Design depth of the vessels
- Dealing with the pontoons and in between space of berths
- determining the entrance/exit point for the vessels

The starting point of the optimisation of the vessels are the centers of mass for each berth. The starting locations are depicted in figure 5.5. The location of these center points implies that this vertex has to be dredged. The normative vessel has a design depth of -9 meters w.r.t. the water level. The vessels in the scope all have this normative depth. The basin bottom is depicted in figure 5.5. Note that the bottom is not yet dredged for equilibrium, thus the gradient in values can exceed the stability slope for soil.

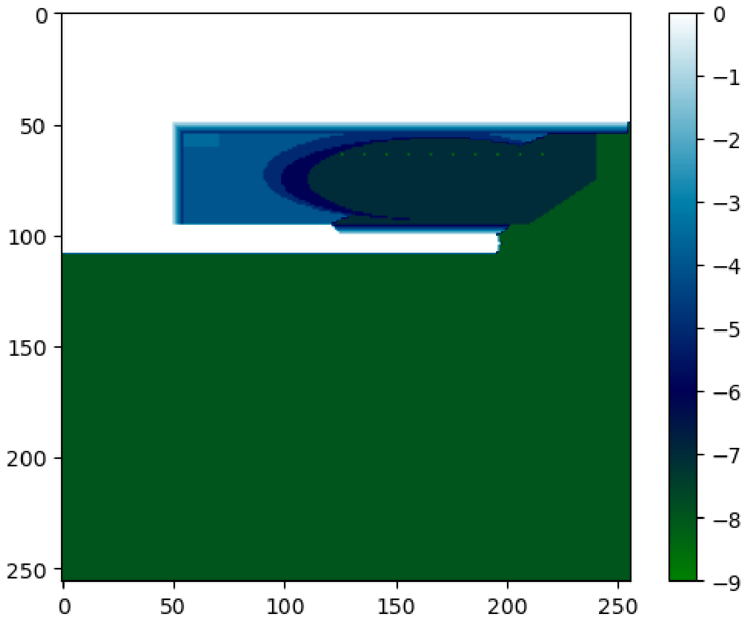


Figure 5.5: Scheurhaven depth profile with the required depth set at the berth locations (light green dots). Note that the color palette slightly shifted from the original depth colors.

The pontoons and spaces in between the berths should not be traversable. Otherwise, the pathfinding algorithm would traverse through the pontoons or use another berth for its pathway. Hence, these vertices are excluded from the considered graph.

Now the optimisation is ready and the method described in Chapter 3 is applied to find the optimal pathway. The figure below displays the final result of the optimisation:

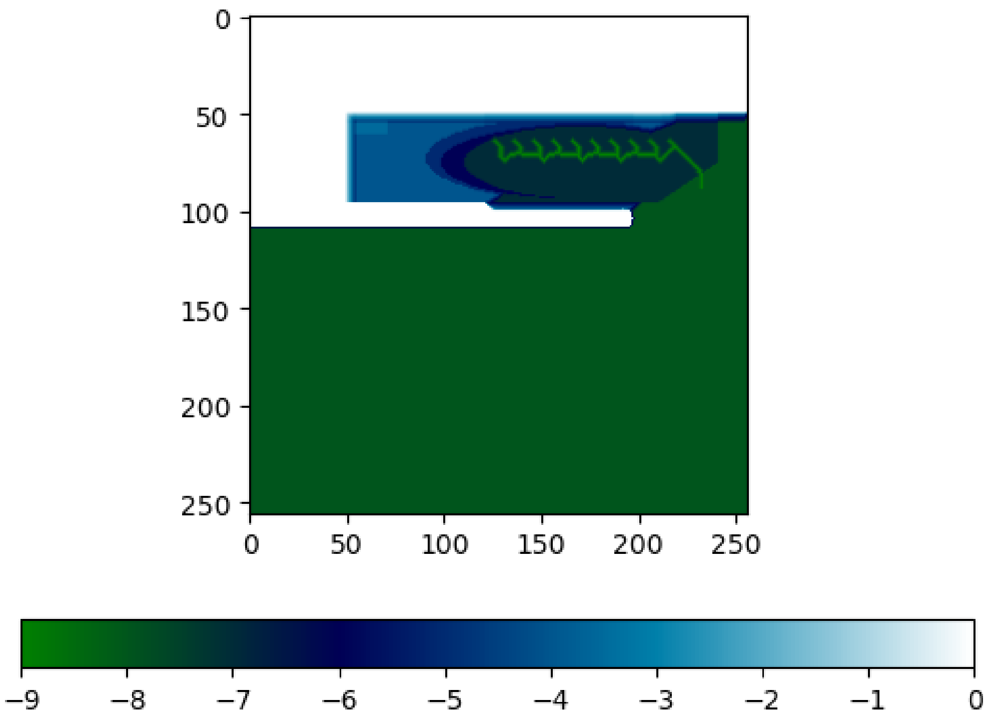


Figure 5.6: Optimal pathway for the Scheurhaven port basin using the heuristic approach

6 Discussion

This chapter discusses the limitations, applicability, and performance of the methods developed.

6.1. Limitations of heuristic berth location optimisation

Chapter 3 introduced the method to optimise berth locations for berths with a minimal required depth and a required minimal distance to other berths, assuming berths as single-cell entities. The first limitation is the lack of berth geometry. As can be seen from the literature presented in Chapter 2, berth layouts is an exercise that relies on a vessel, thus berth, geometry. The distance constraints hold specifically for the bow and aft of the vessel, which cannot be distinguished with the current method. Hence, the distance constraint measures distances in every direction from the berth. The result is that opposite berths are not valid due to being too close to other berths using the method, but in reality, the berth layout is valid. This problem also goes for berths located around corners. The distance measuring in the graph does not make a distinction between berths located on the same straight quay wall, where the distance constraint does hold, or the distance around corners, where the distance constraint does not hold anymore. The result of this flaw is that good quality variants are unjustly labeled as invalid, thus reducing the overall quality of the optimisation, thus leaving out potential optimal variants, thereby limiting support quality for decision-making and selecting the best berth layout variant.

The lack of geometry is the result of the choice to model the space discrete using a grid. A berth shape is a rectangle, which is easy to comprehend for people, but defining a rectangle in discrete space turned out to be difficult. Geometric properties of shapes are hard to define when the grid only contains information on connectivity and grid cell characteristics. The solution might be found in digital geometry or computational geometry. The named subjects have been studied slightly in order to implement geometry, but to no avail. These subjects require a mathematical skill set that goes beyond the mathematics thought in the civil engineering curriculum at TU Delft. Hence, it might be advisable to research the possibility of combining the current methods with a mathematical approach that functions in a continuous space.

Apart from the implications of the lack of geometry on berths, another result of the lack of geometry is the absence of structures in the developed methods. In a basin layout, constructions are indicated using 'simple' geometric elements. A quay wall is schematised as a line or a thin rectangle, just like jetties and pontoons. A considerable amount of time was spent to define geometric shapes in a grid, but, again, to no avail. The lack of structures restricts the port planning exercise significantly, since the possibility of building structures allows for more design variants, thus the potential added value to the decision process for selecting the best berth layout. Other studies mentioned, such as Woerlee (2019), did manage to implement multiple geometric features for breakwater design for a simple basin. The hypothesised reason could be due to the choice of modeling structures in continuous space, which did not require addressing complex disciplines such as digital geometry.

The experienced problem with the implementation of geometry and the resulting lack of structures withheld the progress made in terms of formulating structures in the optimisation formulation. Structures in the context of optimisation for minimal costs have been researched: every structure imposes a monetary cost to the objective. Implementing a structure in a port basin is not a goal in itself; it is a monetary sacrifice willing to be made in order to allow for more design possibilities. The trade-off exists between expanding the design space by building a structure versus the additional costs of building a structure. This trade-off could provide alternative insights relevant to the port basin design exercise for port planners in deciding the best variant, but this is not accomplished due to the complexity of implementing geometry in a discrete environment.

The above-mentioned problems regarding geometry came from the assumption of a discrete space and the importance of geometry in basin layout design. The root lies within the selection to use a discrete space. The reason it was picked was due to the complex shapes of large port basins, which was shortly addressed in Section 3.2.3, to avoid future problems when applying the method to larger port areas. Assuming that port basins are manifold and that analytical mathematical expressions (functions as $y = ax + b$) are not suited for describing complex port geometries of larger port basins, ends up resulting in complex mathematics for describing simple geometric elements in the port basin, such as berth and structures. However, in the recommendations concepts are introduced to possibly implement geometry.

Another limitation is a minor problem in defining the pathway from the berth to the entrance of the port basin using the predefined central pathway. The pathway from the berth to the central pathway is the shortest path between them. However, the pathway per berth has to be the shortest path from berth to entrance, instead of the shortest path from the berth to the central pathway to the basin entrance. The result is that the berths are located and evaluated based on a total pathway that does not consider optimal pathways for each berth. An example can be found in Figure 3.10b, where the pathways have sharp-cornered pathways. The result is another drop in the quality of the generated variant.

Finally, there is a limitation in finding the optimal berth layout using the meta-heuristic improved harmony search algorithm to optimise the layout. The meta-heuristic used has not been fine-tuned in terms of optimisation parameters, such as the convergence speed parameter and focus parameter provided by the pygmo python library. The parameters have been studied limited, hence the optimisation could have converged faster when these parameters were adapted. Moreover, it is suggested to look into developing a self-made heuristic to solve the problem, for instance, a genetic algorithm as done in other student theses as Schaly (2021). A tailor-made heuristic algorithm could end up generating better results in a shorter amount of time.

6.2. Discussion on benchmark performance of heuristic and exact pathway optimisation

The Chapter 4 presented two approaches to optimise pathways through the basin: an exact approach and a heuristics approach. This section discusses the outcome of the two approaches and compares them. After, the limitations are given with regard to the respective methods.

In the first place, both approaches produced better pathways than the initial assumption of a central pathway. Even though the central pathway is an underestimation, as is the heuristic approach, it is improved by the exact pathway, as well as the heuristic pathway. This implies that both approaches provide quality improvement for minimizing pathway dredging. The second result is the overlap in pathways created by heuristic and exact approach when the fleet is divers and the basin bottom is not uniform.

If the fleet is uniform and the depth is also uniform, the results between the heuristic and exact pathway could vary, depending on the basin geometry and berth locations. The pathways created by both approaches both arguably optimise the pathway their way; the heuristic approach maximises the dredging savings by moving along the quay wall, even though the method is not 'aware' of dredging savings due to its negligence of the dredging influence around the pathway, while the exact pathway prefers a central pathway. Due to the difference in measuring dredging between the methods, it is hard to conclude which one performs best, considering the qualitatively good performance of the exact approach in the other benchmarks.

Another factor to consider is the computational time it took for these optimisations. The heuristic approach could generate quality berth layouts in about 2 hours. After that, the pathway optimisations differed slightly. The heuristic pathway finished in about 10 minutes, while the exact approach in some cases required 6 hours to converge to a stable solution. This computational time differed per benchmark; benchmark 1 took 20 minutes to converge using the exact pathway, while benchmark 3 took 6 hours to produce an arguably non-converged solution. However, from a project point of view, these optimisation times are very short compared to the time it takes for designers to design a layout, which is touched upon in Appendix C.

To conclude, it is hard to say which of the approaches is superior in terms of performance. The only certainty is that the exact approach provides a more robust approach, since it is less sensible to human programming errors and it can potentially find a global optimum. Hence, the exact approach would be the more reliable approach to consider for application.

6.3. Limitations on the heuristic and exact pathway approaches

For both the heuristic and exact pathway optimisations, numerous limitations have to be addressed to give a clear view of the extent to which the approaches are applicable. The first obvious limitation is the lack of geometry of the berths, which is discussed in the first section of this chapter. Due to the lack of geometry pathway dimensions are not taken into consideration.

Furthermore, both methods do not take into account vessel maneuverability when forming a pathway through the basin. Vessels in general are less capable of taking sharp turns. This differs from vessel type as tug boats are more maneuverable than large container vessels. Thus, the pathway should be in accordance with the maneuverability of the vessels, resulting in pathways with less sharp turns and more monotonic pathway directions. The result of the negligence of maneuverability is a potentially unrealistic pathway. However, for decision making it could have small implications as the pathway can be used as an indication of optimality for selecting the best basin layout.

Apart from the geometry and maneuverability absence of both methods, each method has its specific drawbacks. A significant drawback of the heuristic approach is in the sequential pathway finding for each berth to eventually define the pathway through the basin. By finding the pathway for the berth sequentially, each berth pathway iteration the method will produce an optimal path for each berth given an environment that will be different from the previous and the next environment, due to the creation of the pathway. This means that in every berth iteration, the problem gets adapted and has its respective optimal solution. The result is that the method will produce local optima upon local optima, which could lead to a non-global optimum, given the problem of finding the best pathway through the whole basin for all berths together. Although this method will produce a better pathway than the central pathway, it can potentially reduce the quality of the pathway design, hence supporting the decision-makers in assessing the best pathway.

To reduce the negative effect of the method described in the paragraph before, the method proposes to try every sequence of pathways from berth to the basin entrance. However, this combinatorial problem increases computational expense when the number of berths increases. Specifically, the number of combinations for a given number of berths is $n_b!$. For the benchmarks with 5 berths, it gives 120 combinations, which was done in roughly 3 minutes. Keep in mind that the basins in the benchmarks were small. For larger basins and a larger fleet, the computational time can increase dramatically.

The exact pathway optimisation approach, contrary to the heuristic pathway optimisation, is capable of finding the global optimal solution. However, when solving the problem for the benchmark cases, it became clear that actual convergence to the optimal solution is hard. The runs done for the benchmark were time-limited to 3 hours. In none of the benchmark cases, the optimisation converged to the optimal. The only convergence during this research was in the sensitivity analysis for a basin width of 25 units and berths located 5 units from the basin entrance, as it converged in roughly 3 minutes. The precise reason for the lack of convergence is not assessed. Observations concerning the log gurobi (the mathematical programming solver used) show that the lower bounds didn't move or moved barely, just as the objective value of the best iteration after 2 hours. However, the optimisation can converge to an optimal solution as previously mentioned, and after 3 hours the produced pathway is stable.

6.4. Applicability of heuristic approach on Scheurhaven case

The heuristic pathway optimisation has been applied on the Scheurhaven case. The reason that no berth location optimisation is applied, is the unfair comparison due to the lack of features in the optimisation method compared to the features used in the real basin layout and therefore the redundancy of the optimisation. In the context of adding value to the potential decision-making process, the Scheurhaven case was a bridge too far for the existing methods to add value in terms of berth layout and pathways.

Even though the method cannot add substantial value to decision-making for berth layout and pathway design for the Scheurhaven case, the case itself had limited potential optimisation. The design possibilities were limited, because most of the basin was already designed and the layout for the additional berths had limited design options. Hence, the potential to add value was little. The current method could potentially add more value to problems with more available berth locations and a fleet that is diverse in terms of design depths and diverse minimal distances between berths and berth location constraints.

To conclude this section: the current case is not the ideal case for this optimisation approach, due to the lack of features in the approaches, as well as the case spatial complexity and fleet composition. Nevertheless, the case study can function as a guideline for modeling a case.

6.5. Limitations of optimisation scope

The current optimisation is to minimize the required dredging for creating the pathway through the basin. Dredging is a factor in deciding what layout is the best, but several other factors also play a role in deciding what basin berth layout to realise. A sample of the other factors is discussed in this section.

The berth layout optimisation locates a set of berths that minimized the dredging for a pathway. the result of the optmisation only takes into account dredging volumes. The result of the volumes can be translated to costs and emissions exhausted to realize the dredging pathway. The additions of these factors will not affect the optimisation, because the relation between costs, emissions and dredging volume is positive; dredging will lead to an increase in costs and emissions. However, when structures are also considered, the relation between dredging and creating structures could affect the optimal design, since a trade-off could exist in marginal costs and emissions of dredging and building structures. By limiting only to dredging volume, this trade-off could not occur.

Another factor to consider would be the safety of the layout. The current method only can impose safety restrictions by means of required distancing between berths. The safety with regard to the created pathway is not considered. The optimisation runs create pathways that run straight along a quay wall, which might not be preferred by vessel operators, because of the risk of colliding with the quay walls caused by, for example, a current. Therefore, the output generated could provide an optimal for dredging, but safety officers could deem it sub-optimal or even invalid. Incorporation of safety standards beyond the guidelines could be considered for future study.

The current scope optimises dredging costs for realising the basin berths and pathway. This optimisation is short-sighted, since costs of infrastructure also occur over its lifetime. Port basins experience the transportation of sediment in and out of the basin that might result in maintenance dredging in a part of the basin when the depth of the basin is no longer the NGD. The effects of basin sedimentation could create a new optimal layout that minimizes lifetime dredging costs.

7 Conclusion

This chapter presents the conclusion of the research. Using research done in the previous chapter the main question stated in the introduction will be answered. The main research question reads:

"How can generative design be used to optimise (or improve) design solutions of port basin layouts during the early project phase?"

In order to answer this question, a method has been developed to generate an optimal berth layouts that minimizes the required dredging path assuming a pre-defined central pathway, while satisfying distance requirements between berths in a basin with quay walls around the perimeter and constant dimensions. The location of the berths is optimised using the Improved harmony search (IHS) meta-heuristic optimisation algorithm. After locating the berths, the optimal pathway for the berth layout is generated using either the heuristic pathway method, which translates dredging into a weighted graph and sequentially formulating a pathway from each berth to the entrance, and an exact pathway method, which uses mathematical programming and the gurobi solver to generate an optimal pathway. The heuristic berth layout, heuristic pathway, and exact pathway optimisation methods were then applied to conceptual, self-developed benchmark cases to test the performance qualitatively and quantitatively of the methods and to compare the objective value for all the pathways developed. Also, the pathways formed by the heuristic and exact pathways were discussed. After that, the exact pathway method was tested on sensitivity to berth location, basin width, slope stability angle, and berth design depth. Finally, the heuristic pathway approach was applied to the Scheurhaven case.

After the research, it can be concluded that generative design using the developed method can improve design solutions by providing an optimised berth layout and pathway that could be used for reference when evaluating expert-developed berth layout designs. The current method lacks features in order to produce a berth layout variant that could compete with the expert's layouts or a layout that could be taken into consideration when selecting a preferred layout for realisation. The method as is can be used to study layouts and pathway in different basin environments and different fleets to observe the optimal behaviour, as done in the method validation and the sensitivity analysis. In the context of the early project phase, the method can give an indication of the locations that could be optimal and after the method has been applied, the experts can compare it to their intuition and use it to update their beliefs on the optimal berth layout and its pathway.

The developed method was made in an attempt to improve port basin layout design in accordance with the thesis scope. However, when doing the literature review for this research, papers were found that optimised the topology of a cantilever beam (Sigmund, 2001, Wang et al., 2002) and combined structural elements (Hofmeyer et al., 2017), which might function as inspiration to improve the structure of port basin elements as quay walls and breakwaters. Generative design could be used to study a breakwater or breakwaters that minimizes material usage, while maintaining structural integrity, which is comparable with Sigmund (2001) and Wang et al. (2002). The optimisation of port basin elements was not in the scope of this research, since it does not concern basin layouts, but application of generative design on the mentioned port basin elements might improve design solutions for port basins.

Returning to the scope of this research, in terms of optimising design solutions in the early phase of an actual project, the Scheurhaven case study indicated that the current method is not ready to improve or optimise berth layouts by means of generated optimal layouts in the early project phase by neither producing a variant that considers substantial berth layout features, such as structures, to be comparable nor by providing decision support for experts. The Scheurhaven case can be described as a bridge too far for the current methods. Hence, the added value for an actual real-life project is little.

Apart from the little added value on the Scheurhaven project using the developed method, the added value of generative design on the early project phase in particular is debatable. Providing a general conclusion of the added value of generative design in the early project phase is not realistic after this research, but factors that impact the added value can be discussed. The first factor is the project size. The Scheurhaven project was a relatively small project with limited design solutions, hence optimisation using generative design is redundant; the added value of the generative design algorithm is negligible. The added value would be greater if the design fleet for the case was greater and more diverse and the basin geometry would be more complex with a greater number of possible berth locations. I hypothesize that these cases get more complex for human designers to find the optimal layout. However the exact added value for these cases has not been studied, thus a definite conclusion cannot be drawn. However, the complexities might not be found on a project level, but on greater port areas, such as the Maasvlakte or Europoort. These areas can be evaluated on whether the current layout is the optimal layout in terms of berth locations. The evaluation could provide a basis for berth location policy aims to minimize dredging in the port areas. The scope might then be no longer on a project level.

Another factor that impacts the added value of generative design in the early project phase is the costs of developing a generative design method versus the potential added value of such generated designs for berth layouts in a project. This research took almost one full-time equivalent of work and several hours of feedback from experts. The result is a method that could be used for further research, but the added value at the moment is little. For further development, the potential added value could be assessed in comparison to the value of current basin layout methods and the resources required to fulfill the potential added value. In this comparison the amount of port basin layouts designs could be considered; if the added value is small for one berth layout design is small, the method could provide limited added value, but if the amount of berth layout designs is numerous the small added value for one design could increment to a great added value.

From a long-term perspective, this research does provide a basis for berth location and pathway optimisation that can be used for development of more features. Most of the features start by finding a way to implement geometry to the method, as the lack of features originates from the lack of geometry. Though it is possible to add geometry by studying digital geometry or find other ways to formulate basic geometry, as in Peng et al. (2016) where simple geometry is defined using mathematical programming, it is difficult due to the complex mathematics that is not granted in the civil engineering curriculum. The upside when the method can be extended with geometric features, it allows for optimisation of any basin shape, which allows for analysis of complex port environments. The method that has the most potential to do these analyses is the exact pathway approach, which has to be extended. The exact pathway optimisation, if extended to berth location optimisation and implementation of geometric features, allows for a global optimum design to be found off the port basin, which the heuristic approaches lack in finding a global optimum.

8

Recommendations

This chapter presents recommendations for further research and improvements. Along with the recommendations, some suggestions are made for implementation.

Adding geometry to the berths and pathways

As stated in the discussion, the geometries of the berths and the impact of vessel dimensions on the fairway are significant. It is a crucial step to generate designs that could be deemed valid. The solution should be to add information on orientation in the graph. A promising method for such implication lies in applying chain code. Chain code considers all possible directions in the Moore neighbourhood and giving each edge direction a value, see figure 8.1. The edge of shapes and regions can be expressed as a chain of directions from a certain starting point. This way shapes can be drawn.

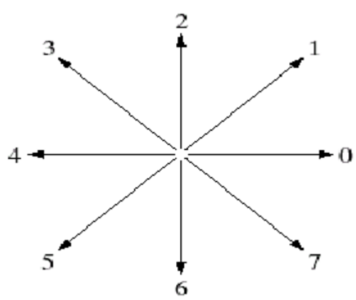


Figure 8.1: Moore neighbourhood directional chain code

Another interesting topic to add geometry would be to consider edge contraction. Berths in a grid can also be represented as one vertex in the grid by contracting all edges of the vertices that describe the berth. By contracting the edges, the berth simply consists of one node, but its surface would be equal to the sum of vertices that are contracted. Also, the notion of edge bundling could help in reducing the computational complexity of the optimisation

Considering operability of vessels in basins

Port basins are interfaces made for vessels. Vessels experience relatively bad maneuverability due to their size and inertia. The designs provided in this report do not take this into account. The result is pathways that have sharp turns that are not navigable by the vessels.

A solution could lie in the addition of a dual graph. A dual graph is a graph that stores relations in the graph. In this case, the dual graph stores relations between the edges and their respective angle.

By tracking graph traversal over the edges, the turns can be measured and then be constrained or be made an objective in the form of a relaxed constraint. A method to formulate this is provided by Nourian (2016), where this is referred to as 'the easiest path'.

Adding structures to the optimisation

As stated in the discussion, not considering structures cuts out a large part of port planning. Implementing structures into the exact formulation would be hard, but rewarding. The formulation for the construction does not have to be of great detail; a set of connected vertices and their impact on the optimisation and environment could help out. In later stages, other factors could be taken into account, such as considering the feasibility of a structure given a soil profile.

A way to formulate the structures in the system would be to consider a structure as a penalty to the objective value and as a disruption in the graph. The penalty is due to the construction cost of such a structure. A structure in the system does provide additional design options, but its counterweight is the cost of building. The disruption in the graph is due to the fact that structures are solid, physical objects: there is no possibility to move through them. In the graph, this could be achieved by deleting all edges going in and out of the structure, or by adding an infinite weight to the edges for graph traversing algorithms to be discouraged from using them. The edge removal would be more logical for the exact approach, while the infinite edge weight is applicable for the heuristic approach.

Extending the exact optimisation formulation

The exact optimisation formulation seems promising in finding the exact optimal solution. If geometric constraints and distance constraints are added to the formulation, the layout optimisation problem would be promising in determining optimal berth layout solutions. A way to address the distance constraint is to pre-process the distances between the possible berth locations and store them in a matrix. The matrix can be used as parameters for the optimisation problem and constraints can be set using it.

Application of heuristic algorithms on the exact optimisation formulation

The exact formulation has yet to be used by applying for the exact optimisation programs. It is also possible to apply heuristic algorithms on the exact problem formulation to produce near-optimal results. The pygmo library is also suitable to use for this cause (Hagberg et al., 2008). As the exact optimisation formulation gets more sophisticated, the application of the heuristic optimisation algorithms on the exact problem could improve computational time.

Designing for value instead of costs

The method proposed only considers costs for optimisation. However, port planning is a measure to obtain a larger strategic goal, which would be to create value. It would be of interest to add parameters to berths which could estimate its revenue and then design a basin for maximal value in terms of placing a number of berths and its location.

If the goal is to add as much valuable berths as possible in a limited port basin area, the problem has close resemblance to a knapsack problem (Cacchiani et al., 2022), which could be combined with the exact approach.

Sensitivity analysis on optimisation parameters

The optimisation script contain optimisation parameters, which can be adapted to get to converge to the solution quicker. This report only used the standard optimisation settings in pygmo and gurobi. For example, the gurobi optimizer can pre-solve the problem more aggressive, which makes the upfront solving more thorough, to give better solutions. A study is required for the parameters and the model's sensitivity to the resulting convergence speed.

List of Figures

List of Tables

Bibliography

Al-Hashemi, H. M. B., & Al-Amoudi, O. S. B. (2018). *A review on the angle of repose of granular materials*. Elsevier B.V. https://doi.org/10.1016/j.powtec.2018.02.003

Alvita, R. (2020). *A tool for container terminal design developing a tool for container terminal design on a concept design phase while taking into account area limitation*. http://repository.tudelft.nl/.

Bakker, H. (2008). *Management of projects: A people process*. https://www.researchgate.net/publication/27345499

Brameier, M., & Banzhaf, W. (2007). *Evolutionary algorithms for solving multi-objective problems*. Springer.

Cacchiani, V., Iori, M., Locatelli, A., & Martello, S. (2022). Knapsack problems — an overview of recent advances. part ii: Multiple, multidimensional, and quadratic knapsack problems. *Computers and Operations Research*, *143*. https://doi.org/10.1016/j.cor.2021.105693

Chan, F. T. S., & Chung, S. H. (2004). Multi-criteria genetic optimization for distribution network problems. *International Journal of Advanced Manufacturing Technology*, *24*, 517–532. https://doi.org/10.1007/s00170-002-1445-5

Cheong, C. Y., & Tan, K. C. (2008). *A multi-objective multi-colony antalgorithm for solving the berthallocation problem*. Springer.

Correcher, J. F., den Bossche, T. V., Alvarez-Valdes, R., & Berghe, G. V. (2019). The berth allocation problem in terminals with irregular layouts. *European Journal of Operational Research*, *272*, 1096–1108. https://doi.org/10.1016/j.ejor.2018.07.019

Cortezón, J. A. R., & de la Peña, J. A. J. (2007). *Rom 5.1-05 quality of coastal waters in port areas*. Puertos del Estado.

Cubukcuoglu, C., Nourian, P., Sariyildiz, I. S., & Tasgetiren, M. F. (2022). Optimal design of new hospitals: A computational workflow for stacking, zoning, and routing. *Automation in Construction*, *134*. https://doi.org/10.1016/j.autcon.2021.104102

Diestel, R. (2017). *Graph theory*. http://www.springer.com/series/136

Dijkstra, E. W. (1959). A note on two problems in connexion with graphs. *Numerische Mathematik*, 269–271.

Doran, J., & Kendall, F. (1966). Experiments with the graph traverser program, 235–259. https://royalsocietypublishing.org/

Drira, A., Pierreval, H., & Hajri-Gabouj, S. (2007). Facility layout problems: A survey. *Annual Reviews in Control*, *31*, 255–267. https://doi.org/10.1016/j.arcontrol.2007.04.001

Erdelsky, P. J. (n.d.). Elementary set theory [Accessed: 2023-05-15]. http://www.efgh.com/math/algebra/sets.html.

FICO. (2023). Xpress [Accessed: 2023-05-10]. https://www.fico.com/en/products/fico-xpress-optimization

Geotechdata.info. (2013). Angle-of-friction. http://geotechdata.info/parameter/angle-of-friction

Gigerenzer, G. (2008). Why heuristics work. *Perspectives on Psychological Science*. https://doi.org/https://doi-org.tudelft.idm.oclc.org/10.1111/j.1745-6916.2008.00058.x

GNU. (2023). Gnu linear programming kit (glpk) [Accessed: 2023-05-10]. https://www.gnu.org/software/glpk/

Gurobi. (2023). Gurobi [Accessed: 2023-05-10]. https://www.gurobi.com/

Hagberg, A. A., Schult, D. A., & Swart, P. J. (2008). Exploring network structure, dynamics, and function using networkx. In G. Varoquaux, T. Vaught, & J. Millman (Eds.), *Proceedings of the 7th python in science conference* (pp. 11–15).

Haralick, R. M., Sternberg, S. R., & Zhuang, X. (1987). Image analysis using mathematical morphology. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *PAMI-9*, 532–550. https://doi.org/10.1109/TPAMI.1987.4767941

Hofmeyer, H., Schevenels, M., & Boonstra, S. (2017). The generation of hierarchic structures via robust 3d topology optimisation. *Advanced Engineering Informatics*, *33*, 440–455. https://doi.org/10.1016/j.aei.2017.02.002

IBM. (2023). Cplex [Accessed: 2023-05-10]. https://www.ibm.com/products/ilog-cplex-optimization-studio/cplex-optimizer

Imai, A., Nagaiwa, K., & Tat, C. W. (1997). Efficient planning of berth allocation for container terminals in asia. *Journal of Advanced Transportation*, *31*, 75–94. https://doi.org/10.1002/atr.5670310107

Klette, R. (2004). *Digital geometry*. https://doi.org/https://doi.org/10.1016/B978-1-55860-861-0.50026-2

Mahdavi, M., Fesanghary, M., & Damangir, E. (2007). An improved harmony search algorithm for solving optimization problems. *Applied Mathematics and Computation*, *188*, 1567–1579. https://doi.org/10.1016/j.amc.2006.11.033

Ministerie van Infrastructuur en Waterstaat. (2021). *Adn*. Rijksoverheid. https://www.rijksoverheid.nl/documenten/publicaties/2015/05/21/adn

Mittelman, H. (2011). Mixed integer linear programming benchmark (serial codes) [Accessed: 2023-05-10].

Nourian, P. (2016). *Configraphics graph theoretical methods for design and analysis of spatial configurations*. TU Delft. https://doi.org/10.7480/abe.2016.14

Nourian, P. (2023). *Generative design in architecture: From mathematical optimization to grammatical customization* (P. Kyratsis, A. Manavis, & J. P. Davim, Eds.). Springer International Publishing. https://doi.org/10.1007/978-3-031-21167-6

Peng, C. H., Yang, Y. L., Bao, F., Fink, D., Yan, D. M., Wonka, P., & Mitra, N. J. (2016). Computational network design from functional specifications. *ACM Transactions on Graphics*, *35*. https://doi.org/10.1145/2897824.2925935

PIANC. (2004). Guidelines for the design of fender systems (2002-2004). https://www.pianc.org/publications/marcom/guidelines-for-the-design-of-fender-systems

PIANC. (2014). Harbour approach channels - design guidelines (2014). https://www.pianc.org/publications/marcom/harbour-approach-channels-design-guidelines

PIANC. (2016). Criteria for the selection of breakwater types and their related optimum safety levels (2016). https://www.pianc.org/publications/marcom/criteria-for-the-selection-of-breakwater-types-and-their-related-optimum-safety-levels

PIANC. (2019). *Ports on greenfield sites : Guidelines for site selection and masterplanning*.

Port of Rotterdam. (2022). *Ontwerprichtlijnen havens en vaarwegen*.

Purss, M. (2017). *Open geospatial consortium internal reference number of this ogc ® document: 15-104r5*. http://www.opengeospatial.org/legal/.

Rardin, R. L., & Uzsoy, R. (2001). *Experimental evaluation of heuristic optimization algorithms: A tutorial*.

Rijksoverheid. (2022). Binnenvaartpolitieregelement. https://wetten.overheid.nl/BWBR0003628/2017-01-01

Rosenfeld, A., & Klette, R. (2002). Digital geometry. *Information Sciences*, *148*, 123–127. www.elsevier.com/locate/ins

Schaly, R. M. (2021). *Increasing the value of a genetic algorithm based breakwater layout model*. http://repository.tudelft.nl/.

Sigmund, O. (2001). *A 99 line topology optimization code written in matlab*. Springer-Verlag. http://www.topopt.dtu.dk.

Snyman, J. A., & Wilke, D. N. (2010). *Springer optimization and its applications 133 practical mathematical optimization basic optimization theory and gradient-based algorithms second edition*. http://www.springer.com/series/7393

Sörensen, K. (2015). Metaheuristics-the metaphor exposed. *International Transactions in Operational Research*, *22*, 3–18. https://doi.org/10.1111/itor.12001

Sun, Y., Chu, S.-C., Hu, P., Watada, J., Si, M., & Pan, J.-S. (2022). Overview of parallel computing for meta-heuristic algorithms eurasip journal on wireless communications and networking view project a human behavior analysis system. view project overview of parallel computing for meta-heuristic algorithms. *Taiwan Ubiquitous Information*, *7*. https://www.researchgate.net/publication/362732572

Taccari, L. (2016). Integer programming formulations for the elementary shortest path problem. *European Journal of Operational Research*, *252*, 122–130. https://doi.org/10.1016/j.ejor.2016.01.003

Taneja, P., Walkert, W. E., Ligteringent, H., Schuylenburg, M. V., & Plas, R. V. (2010). Implications of an uncertain future for port planning. *Maritime Policy and Management*, *37*, 221–245. https://doi.org/10.1080/03088831003700637

Teeling, J. (2020). *Improving applicability of a parametric model for breakwater layout design*. http://repository.tudelft.nl/.

Thoresen, C. (2014). *Port designer's handbook (3rd edition)*. ICE Publishing. https://doi.org/10.1680/pdh.60043.001

Tsinker, G. (1997). *Handbook of port and harbor engineering*. Springer US. https://doi.org/10.1007/978-1-4757-0863-9

van Koningsveld, M., Verheij, H., Taneja, P., & de Vriend, H. (2021). Ports and waterways navigating the changing world. https://doi.org/10.5074/T.2021.004

Verruijt, A. (2001). *Grondmechanica*. http://geo.verruijt.net/.

Wang, M. Y., Wang, X., & Guo, D. (2002). *A level set method for structural topology optimization*. www.elsevier.com/locate/cma

Wikipedia. (2023). Z-order curve — Wikipedia, the free encyclopedia [[Online; accessed 14-May-2023]].

Woerlee, S. (2019). *Breakwater layout optimisation using a parametric model development of a decision-making tool for the conceptual design of breakwaters*. https://www.pagineazzurre.com/.

Yazdi, A., & Teshnizi, E. S. (2021). Effects of contamination with gasoline on engineering properties of fine-grained silty soils with an emphasis on the duration of exposure. *SN Applied Sciences*, *3*. https://doi.org/10.1007/s42452-021-04637-x

Zhang, T. Y., & Suen, C. Y. (1984). A fast parallel algorithm for thinning digital patterns. *Image processing and Computer Vision*.

A UKC policy

Havengebied, varend	UKC beleid
Nieuwe Waterweg, Nieuwe Maas en Oude Maas	10% van de diepgang
Nieuwe Maas: containerschepen Eem- en Waalhaven	1,0 m
Inloop centrale geul Botlek, Eemhaven en Waalhaven	1,0 m
Havens regio Botlek/Stad	0,5 m
Caland-/Beer-/Yangtzekanaal en Arianehaven - *Container Vessels L>350m*	1,0 m - *10% van de diepgang*
Havens grenzend aan Caland-, Beer- en Yangtzekanaal en Arianehaven, T < 17,4m - *Container Vessels L>350m*	0,5 m - *8% van de diepgang*
Havens grenzend aan Caland-, Beer- en Yangtzekanaal en Arianehaven, T ≥ 17,4m - *Container Vessels L>350m*	1,0 m - *8% van de diepgang*
Hartelkanaal	10% van de diepgang en 0,5 m in sluizen
Krabbegeul en Zeehaven Dordrecht, inkomend	10% van de locale diepgang
Krabbegeul en Zeehaven Dordrecht, uitgaand	5% van de locale diepgang
Dordtsche Kil en Overloop	10% van de locale diepgang
Havens Moerdijk en Zuid Hollandsch Diep	5% van de locale diepgang
Noord via Bolnes	10% van de locale diepgang
Noord via Dordrecht	10% van de locale diepgang
Binnenvaart (varend én ligplaats i.v.m. boegschroef)	0,7 m
Havenslepers (varend en ligplaats i.v.m. grote vermogens en straalbuizen)	1,0 m

Figure A.1: Static UKC policy

B

ALAT policy

Figure B.1: ALAT regions, where the numbers indicate the height of the NAP reference level with respect to ALAT

Deelgebieden	ALAT t.o.v. NAP	HMT	FWA
Prinses Amalia-, Alexia- ,Ariane- en Margriethaven	-115 cm	30 cm	1,0%
Palen 90 en 91	-110 cm	30 cm	1,0%
Yangtzekanaal	-105 cm	30 cm	1,0%
Maasvlakte 1 en Europoort	-100 cm	30 cm	1,0%
Beergat/Hudsonhaven	-95 cm	30 cm	1,0%
Nieuwe Waterweg - raai 1031 tot 1028	-95 cm	30 cm	1,0%
Nieuwe Waterweg - raai 1028 tot 1021	-90 cm	30 cm	1,0%
Dintelhaven/Hartelkanaal (tot Pionierkade)	-90 cm	30 cm	2,5%
Hartelkanaal tot Hartel 2 terminal t.h.v. Donauhaven	-85 cm	30 cm	2,5%
Scheur – raai 1021 tot 1017	-85 cm	30 cm	2,5%
Scheur – raai 1017 tot 1014	-80 cm	30 cm	2,5%
Hartelkanaal Rozenburgsesluis – Oude Maas	-80 cm	30 cm	2,5%
Botlek, Oude Maas tot Botlekbrug en Nieuwe Maas tot Schiemond	-75 cm	30 cm	2,5%
Nieuwe Maas van Schiemond tot raai 997	-70 cm	30 cm	2,5%
Nieuwe Maas – raai 997 tot 994	-65 cm	30 cm	2,5%
Krabbegeul en Zeehaven Dordrecht	-5 cm	30 cm	2,5%
Moerdijk	+15 cm	30 cm	2,5%

Figure B.2: Normative ALAT and FWA in port regions

C

Scheurhaven case design process

The research aims to compare the current port planning process for the Scheurhaven project to the production of the algorithm. The design process is stored in the table below. Each step is provided with an estimation of time spend on the step:

Task	Time spent [hours]
Intake	4
Modelling status quo	8
Exploratory research	8
Create variant 1	6
Evaluate variant 1	3
Adapt variant 1	3
Create variant 2	6
Evaluate variant 2	3
Adapt variant 2	3
Create variant 3	6
Evaluate variant 3	3
Adapt variant 3	3
Create variant 4	6
Evaluate variant 4	3
Adapt variant 4	3
Create variant 5	6
Evaluate variant 5	3
Adapt variant 5	3

Table C.1: Time estimation of the port basin design of the Scheurhaven project for a port designer.

The table is the result of an interview with a port planner from the port of Rotterdam. The tasks in the table describe:

- Create variant: shaping a spatial layout, given constraints given by the experts on the project team. The variant produced is deemed possibly optimal or of particular interest.
- Evaluate variant: the variant is evaluated by the project team on the quality (objective performance) and validity (requirements satisfaction).
- Adapt variant: feedback provided by the evaluation is processed in the variant design.

The intake, modelling status quo and the exploratory research are all tasks done at the start of the project to inventory what the situation is and what the situation should be. During this phase people also get to know each other and set out expectations for the project. The interesting part of the process is the repetitive parts, since an algorithms is meant for repetitive tasks. The variant design, evaluation and adaption provide an opportunity for improvement. The current man hours it takes for one design is 80 hours. This is only one member of the project team, which consists of 8 people.

The time in itself does not give complete insight in the process. Information is required in the timeframe these steps were taken during the early project phase. The early project phase started approximately in April 2022 and the end of the early project phase ended in December 2022.

D
Dredging algorithm

Placing a berth requires adjustments in the basin bottom profile to guarantee a certain depth for vessel to navigate and berth safely. Keep in mind that the environment is discrtised into a mesh with cells. When the pathing and berth are at guaranteed depth by 'removing' soil from a cell, the profile will not be in equilibrium. The reason is that the height difference between the dredged cells and non-dredged cells can give non-feasible basin bottoms. Thus, a function is required which:

1. Dredges the pathway and berth;
2. Creates the new bottom equilibrium profile after dredging to the required depth in path and berths;

In that order, The process is schematised in figure E.21. The value for the stability slope ϕ_{ss} depends on the soil in the basin. For now, the soil is assumed to be saturated sand, resulting in a stability slope angle of 40°

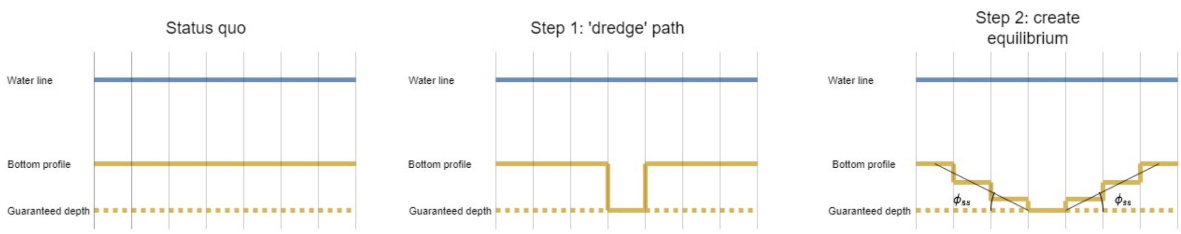


Figure D.1: Dredging function

The calculation for the equilibrium profile starts from the cells where dredging is required and sprawls to the adjacent cells. This process keeps going until the neighbouring cells are stable compared to the original cells and the given slope stability angle ϕ_{ss}. In graph theory such algorithm is called a 'breadth-first search (BFS)' algorithm. The process of the BFS algorithm is displayed in figure D.2

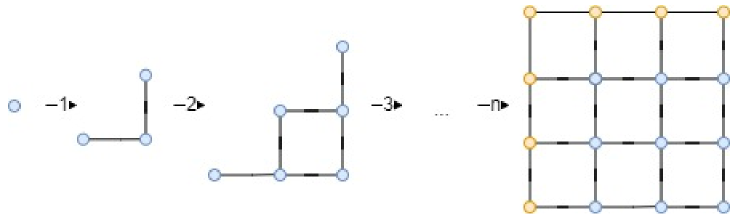


Figure D.2: BFS algorithm on schematised port basin

This process is done for every node in the pathway to create an equilibrium for the respective pathway. However, in the process some vertices in the pathway are affected and are no longer the required depth. Hence, the vertices that are no longer the required depth have to be dredged again. This process is repeated until every pathway vertex has satisfactory depth and the bottom is in equilibrium.

E

Geometric approach

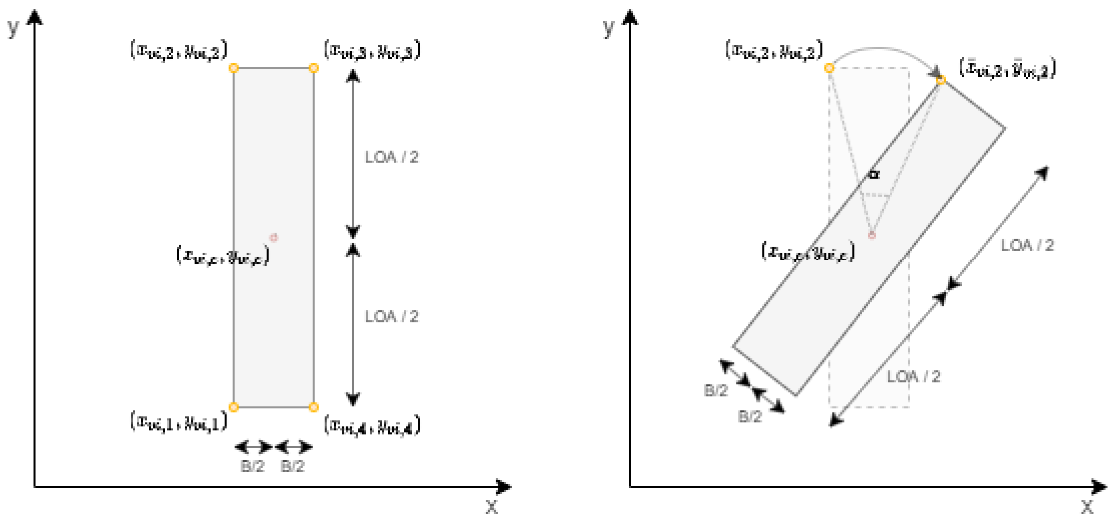
This appendix contains the geometric approach. Important is the fact that this method is not worked out completely, only for a part. Evaluation of the method and advice from the committee caused this approach to be abandoned. The appendix is meant to display the complexity constraints and accompanying notations.

E.1. Input variables

The input for the problem will be an amount of vessels. The vessel input will consist of:

- Length overall (LOA)
- Width (B)
- Draught (D)
- Vessel centre coordinates ($x_{vi,c}$, $y_{vi,c}$)

From the input all relevant parameters can be calculated. In figure E.1 depicts the coordinates of a vessel in a rotated and un-rotated state.



(a) Non-rotated vessel coordinates

(b) Rotated vessel coordinates

Figure E.1: Vessel coordinates

The coordinates of the non-rotated vessels are:

$$x_{vi,1} = x_{vi,c} - B_{vi}/2 \;\; ; \;\; y_{vi,1} = y_{vi,c} - LOA_{vi}/2$$
$$x_{vi,2} = x_{vi,c} - B_{vi}/2 \;\; ; \;\; y_{vi,2} = y_{vi,c} + LOA_{vi}/2$$
$$x_{vi,3} = x_{vi,c} + B_{vi}/2 \;\; ; \;\; y_{vi,3} = y_{vi,c} + LOA_{vi}/2$$
$$x_{vi,4} = x_{vi,c} + B_{vi}/2 \;\; ; \;\; y_{vi,4} = y_{vi,c} - LOA_{vi}/2$$

The coordinates of the rotated vessels are:

$$\bar{x}_{vi,1} = x_{vi,1} * cos(\alpha_{vi}) - y_{vi,1} * sin(\alpha_{vi}) \;\; ; \;\; \bar{y}_{vi,1} = x_{vi,1} * sin(\alpha_{vi}) + y_{vi,1} * cos(\alpha_{vi})$$
$$\bar{x}_{vi,2} = x_{vi,2} * cos(\alpha_{vi}) - y_{vi,2} * sin(\alpha_{vi}) \;\; ; \;\; \bar{y}_{vi,2} = x_{vi,2} * sin(\alpha_{vi}) + y_{vi,2} * cos(\alpha_{vi})$$
$$\bar{x}_{vi,3} = x_{vi,3} * cos(\alpha_{vi}) - y_{vi,3} * sin(\alpha_{vi}) \;\; ; \;\; \bar{y}_{vi,3} = x_{vi,3} * sin(\alpha_{vi}) + y_{vi,3} * cos(\alpha_{vi})$$
$$\bar{x}_{vi,4} = x_{vi,4} * cos(\alpha_{vi}) - y_{vi,4} * sin(\alpha_{vi}) \;\; ; \;\; \bar{y}_{vi,4} = x_{vi,4} * sin(\alpha_{vi}) + y_{vi,4} * cos(\alpha_{vi})$$

with:

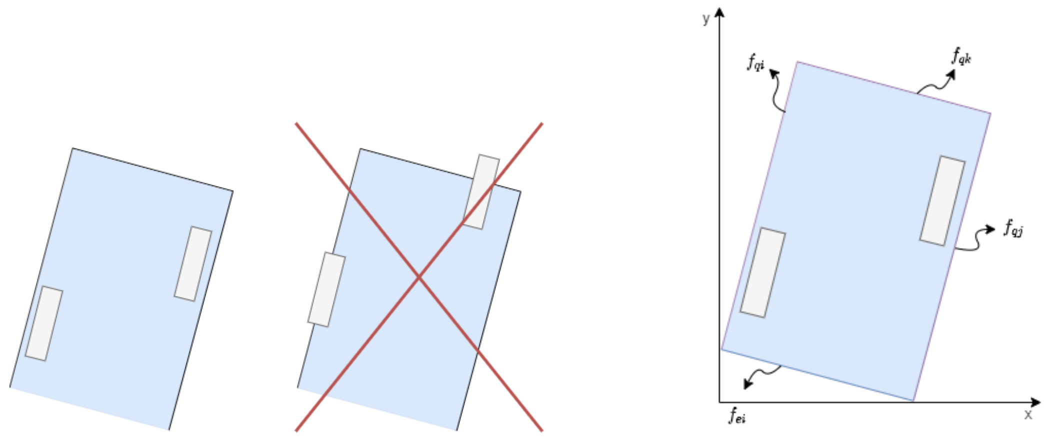
$$-90 \leq \alpha \leq 90$$

The reason for this domain for α_{vi} is the repetition of orientation for a larger domain.

E.2. Constraints

In this section presents the imposed restrictions. The imposed restrictions are obtained from the document "Ontwerprichtlijnen havens en vaarwegen". The section shows the practical depiction and description of the restriction and the mathematical depiction and description. The base case is a simple rectangle port basin.

1. Port basin domain The vessels lay completely inside the port basin domain.



(a) Practical representation (b) Mathematical representation

Figure E.2: Vessel domain

The vessel coordinates have to satisfy all the following mathematical expressions:

$$\bar{y}_{vi,m} \geq f_{el} \;\; for \;\; x_{el,0} \leq x_{vi,m} \leq x_{el,1}$$
$$\bar{y}_{vi,m} \leq f_{qi} \;\; for \;\; x_{qi,0} \leq x_{vi,m} \leq x_{qi,1}$$
$$\bar{y}_{vi,m} \leq f_{qk} \;\; for \;\; x_{qk,0} \leq x_{vi,m} \leq x_{qk,1}$$
$$\bar{y}_{vi,m} \geq f_{qj} \;\; for \;\; x_{qj,0} \leq x_{vi,m} \leq x_{qj,1}$$

In which i, j, k = 1, 2, .., o, with o being the number of basin boundaries. l = 1, 2, .., p, with p being the number of entrances.

2. vessel overlap The vessels are not allowed to overlap with each other in each others 'operations zone'. The operations zone is the zone around the vessel where no other objects are allowed, unless for a quay wall or jetty. The zone is 10% of the vessel length around the whole vessel.

The coordinates for the non-rotated vessel operations zone are:

$$x_{vi,1,oz} = x_{vi,c} - \frac{5B_{vi} + LOA_{vi}}{10} \;\; ; \;\; y_{vi,1} = y_{vi,c} - 0.6LOA_{vi}$$

$$x_{vi,2,oz} = x_{vi,c} - \frac{5B_{vi} + LOA_{vi}}{10} \;\; ; \;\; y_{vi,2} = y_{vi,c} + 0.6LOA_{vi}$$

$$x_{vi,3,oz} = x_{vi,c} + \frac{5B_{vi} + LOA_{vi}}{10} \;\; ; \;\; y_{vi,3} = y_{vi,c} + 0.6LOA_{vi}$$

$$x_{vi,4,oz} = x_{vi,c} + \frac{5B + LOA_{vi}}{10} \;\; ; \;\; y_{vi,4} = y_{vi,c} - 0.6LOA_{vi}$$

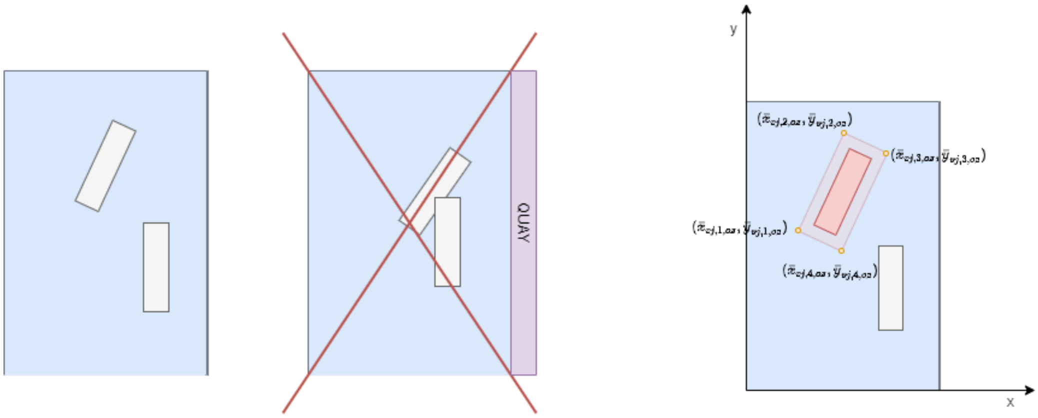
The coordinates for the rotated vessel operations zone are

$$\bar{x}_{vi,1,oz} = x_{vi,1,oz} * cos(\alpha_{vi}) - y_{vi,1,oz} * sin(\alpha_{vi}) \;\; ; \;\; \bar{y}_{vi,1,oz} = x_{vi,1,oz} * sin(\alpha_{vi}) + y_{vi,1,oz} * cos(\alpha_{vi})$$

$$\bar{x}_{vi,2,oz} = x_{vi,2,oz} * cos(\alpha_{vi}) - y_{vi,2,oz} * sin(\alpha_{vi}) \;\; ; \;\; \bar{y}_{vi,2,oz} = x_{vi,2,oz} * sin(\alpha_{vi}) + y_{vi,2,oz} * cos(\alpha_{vi})$$

$$\bar{x}_{vi,3,oz} = x_{vi,3,oz} * cos(\alpha_{vi}) - y_{vi,3,oz} * sin(\alpha_{vi}) \;\; ; \;\; \bar{y}_{vi,3,oz} = x_{vi,3,oz} * sin(\alpha_{vi}) + y_{vi,3,oz} * cos(\alpha_{vi})$$

$$\bar{x}_{vi,4,oz} = x_{vi,4,oz} * cos(\alpha_{vi}) - y_{vi,4,oz} * sin(\alpha_{vi}) \;\; ; \;\; \bar{y}_{vi,4,oz} = x_{vi,4,oz} * sin(\alpha_{vi}) + y_{vi,4,oz} * cos(\alpha_{vi})$$



(a) Practical representation (b) Mathematical representation

Figure E.3: Vessel overlap

Overlapping for rotated vessels requires extra attention since the restrictions set are more complex. To tackle the restrictions, regions A to F are introduced to set the restrictions piece wise. Figures E.3a and E.3b display the regions. First, the case for -90°<α<0°is addressed (figure E.3).

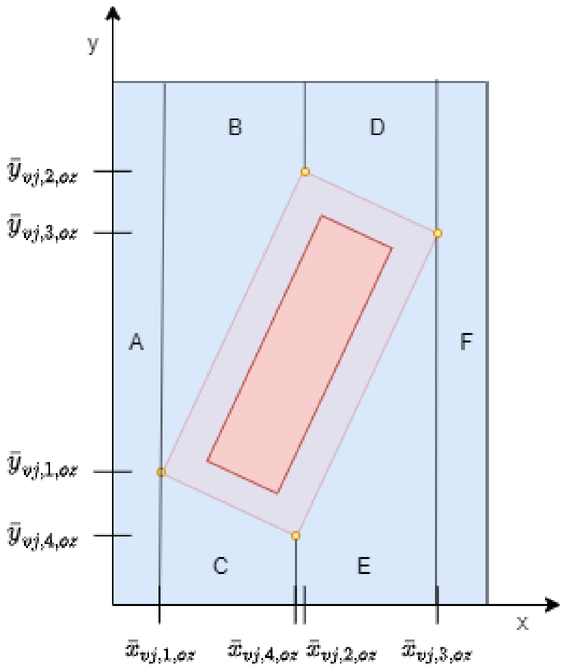


Figure E.4: Vessel domain for -90°<α<0°

Each of the regions is addressed in the list below. All points of vessel vi should at least comply to one of the criteria:

$$A: \ x_{vi,m} < \bar{x}_{vj,1,oz}$$

$$B: \ y_{vi,m} > \bar{y}_{vj,1,oz} + \frac{\bar{y}_{vj,2,oz} - \bar{y}_{vj,1,oz}}{\bar{x}_{vj,2,oz} - \bar{x}_{vj,1,oz}} * (x_{vi,m} - \bar{x}_{vj,1,oz}) \ \ with \ \ \bar{x}_{vj,1,oz} \le x_{vi,m} \le \bar{x}_{vj,2,oz}$$

$$C: \ y_{vi,m} < \bar{y}_{vj,1,oz} + \frac{\bar{y}_{vj,4,oz} - \bar{y}_{vj,1,oz}}{\bar{x}_{vj,4,oz} - \bar{x}_{vj,1,oz}} * (x_{vi,m} - \bar{x}_{vj,1,oz}) \ \ with \ \ \bar{x}_{vj,1,oz} \le x_{vi,m} \le \bar{x}_{vj,4,oz}$$

$$D: \ y_{vi,m} > \bar{y}_{vj,2,oz} + \frac{\bar{y}_{vj,3,oz} - \bar{y}_{vj,2,oz}}{\bar{x}_{vj,3,oz} - \bar{x}_{vj,2,oz}} * (x_{vi,m} - \bar{x}_{vj,2,oz}) \ \ with \ \ \bar{x}_{vj,2,oz} \le x_{vi,m} \le \bar{x}_{vj,3,oz}$$

$$E: \ y_{vi,m} < \bar{y}_{vj,4,oz} + \frac{\bar{y}_{vj,3,oz} - \bar{y}_{vj,4,oz}}{\bar{x}_{vj,3,oz} - \bar{x}_{vj,4,oz}} * (x_{vi,m} - \bar{x}_{vj,4,oz}) \ \ with \ \ \bar{x}_{vj,4,oz} \le x_{vi,m} \le \bar{x}_{vj,3,oz}$$

$$F: \ x_{vi,m} > \bar{x}_{vj,3,oz}$$

$$with \ \ m = 1,2,3,4; \ \ i = 1,2,..,n; \ \ j = 1,2,..,n; \ \ i \ne j$$

n = number of vessels

For the vessel domain when 0°<α_{vj}<90°the situation is shown in the figure below:

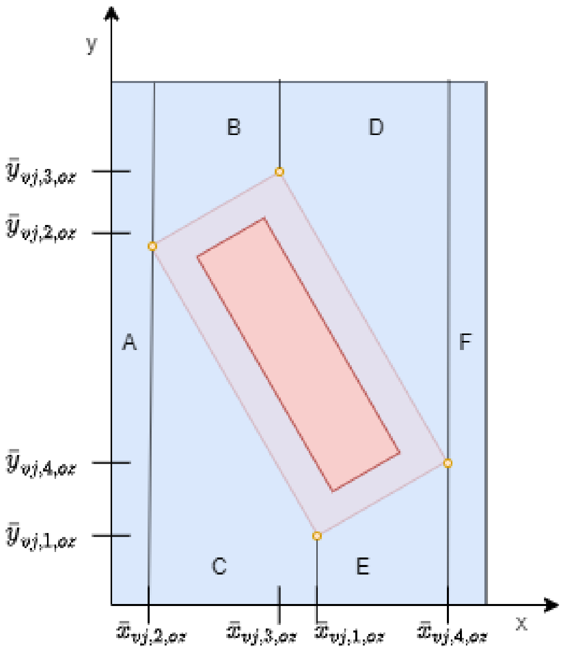


Figure E.5: Vessel domain for 0°<α_{vi}<90°

Each of the regions is addressed in the list below. All points of vessel vi should at least comply to one of the criteria:

$$A: \; x_{vi,m} < \bar{x}_{vj,2}$$

$$B: \; y_{vi,m} > \bar{y}_{vj,2} + \frac{\bar{y}_{vj,3,oz} - y_{vj,2,oz}}{\bar{x}_{vj,3,oz} - \bar{x}_{vj,2,oz}} * (x_{vi,m} - \bar{x}_{vj,2,oz}) \;\; with \;\; \bar{x}_{vj,2,oz} \leq x \leq \bar{x}_{vj,3,oz}$$

$$C: \; y_{vi,m} < \bar{y}_{vj,2,oz} + \frac{\bar{y}_{vj,1,oz} - \bar{y}_{vj,2,oz}}{\bar{x}_{vj,1,oz} - \bar{x}_{vj,2,oz}} * (x_{vi,m} - \bar{x}_{vj,2,oz}) \;\; with \;\; \bar{x}_{vj,2,oz} \leq x_{vi,m} \leq \bar{x}_{vj,1,oz}$$

$$D: \; y_{vi,m} > \bar{y}_{vj,3,oz} + \frac{\bar{y}_{vj,4,oz} - \bar{y}_{vj,3,oz}}{\bar{x}_{vj,4,oz} - \bar{x}_{vj,3,oz}} * (x_{vi,m} - \bar{x}_{vj,3,oz}) \;\; with \;\; \bar{x}_{vj,3,oz} \leq x_{vi,m} \leq \bar{x}_{vj,4,oz}$$

$$E: \; y_{vi,m} < \bar{y}_{vj,1} + \frac{\bar{y}_{vj,4} - \bar{y}_{vj,1}}{\bar{x}_{vj,4} - \bar{x}_{vj,1}} * (x_{vi,m} - \bar{x}_{vj,1,oz}) \;\; with \;\; \bar{x}_{vj,1,oz} \leq x_{vi,m} \leq \bar{x}_{vj,4,oz}$$

$$F: \; x_{vi,m} > \bar{x}_{vj,4}$$

Another case is when α_{vj}=-90°, shown in the figure below:

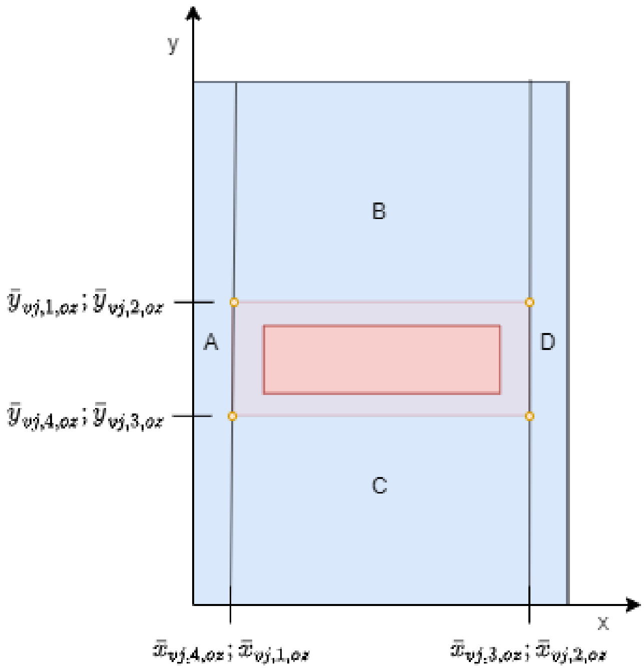


Figure E.6: Vessel domain α_{vj}=-90°

Each of the regions is addressed in the list below. All points of vessel vi should at least comply to one of the criteria:

$$A: \; x_{vi,m} < \bar{x}_{vj,1,oz}$$
$$B: \; y_{vi,m} > \bar{y}_{vj,1,oz}$$
$$C: \; y_{vi,m} < \bar{y}_{vj,3,oz}$$
$$D: \; x_{vi,m} > \bar{x}_{vj,3,oz}$$

The case for α_{vj}=90°is:

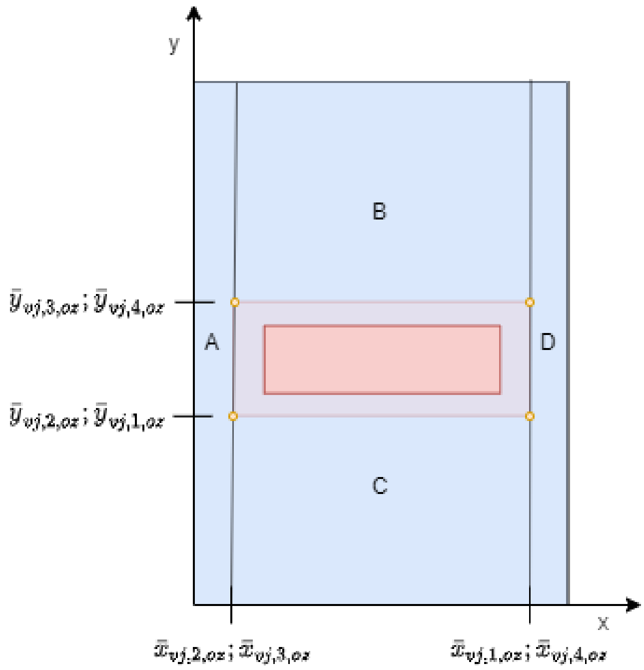


Figure E.7: Vessel domain α_{vj}=90°

Each of the regions is addressed in the list below. All points of vessel vi should at least comply to one of the criteria:

$$A: \; x_{vi,m} < \bar{x}_{vj,2,oz}$$
$$B: \; y_{vi,m} > \bar{y}_{vj,4,oz}$$
$$C: \; y_{vi,m} < \bar{y}_{vj,2,oz}$$
$$D: \; x_{vi,m} > \bar{x}_{vj,4,oz}$$

The final case is when α=0°:

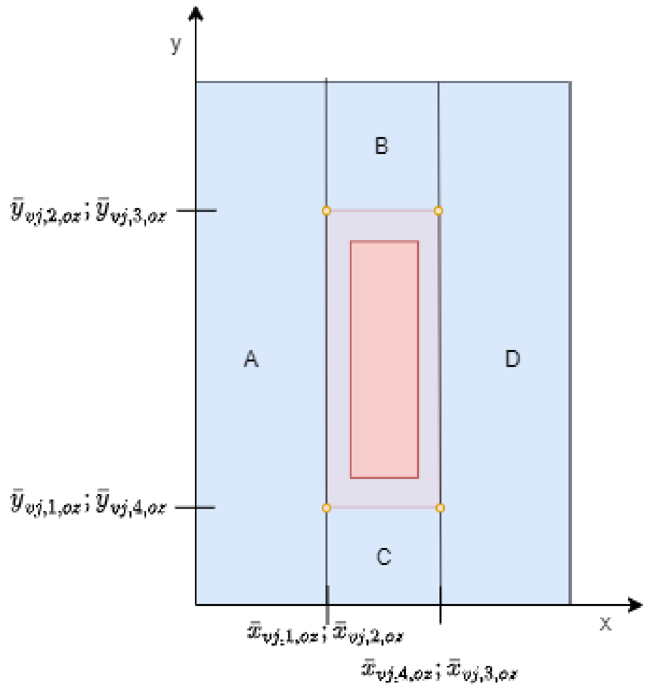
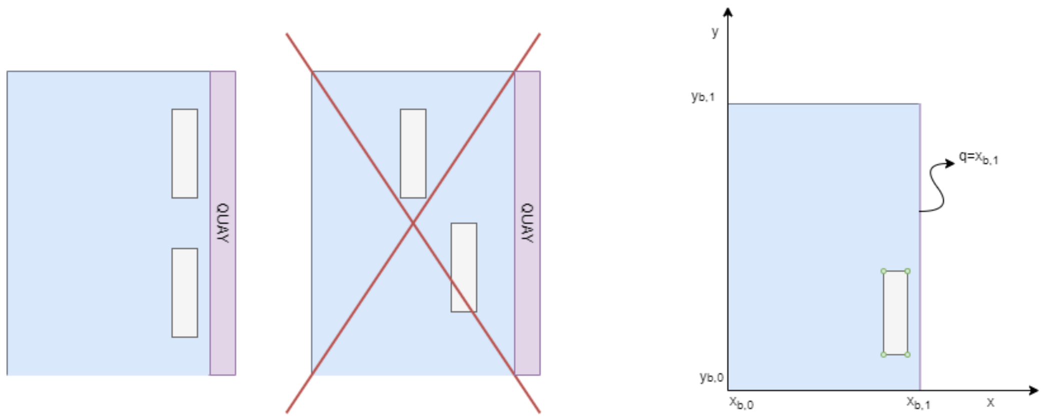


Figure E.8: Vessel domain α_{vj}=0°

Each of the regions is addressed in the list below. All points of vessel vi should at least comply to one of the criteria:

$$A: \; x_{vi,m} < \bar{x}_{vj,1,oz}$$
$$B: \; y_{vi,m} > \bar{y}_{vj,3,oz}$$
$$C: \; y_{vi,m} < \bar{y}_{vj,1,oz}$$
$$D: \; x_{vi,m} > \bar{x}_{vj,3,oz}$$

3. Quay attach The vessels in the basin have to be attached to a quay wall or jetty (figure E.9).



(a) Practical representation

(b) Mathematical representation

Figure E.9: Quay wall attach

Mathematically speaking, the quay wall is a line in space. The vessels modelled have to be attached with the long side of the vessel to that exact line. The exact side of attachment of the vessel does not matter, as long as it is one of either sides. Mathematically this is expressed as:

$$(x_{vi,1} = f_{qk}) \cap (x_{vi,2} = f_{qk}) \cap (y_{vi,1} = f_{qk}) \cap (y_{vi,2} = f_{qk}) \;\; or;$$
$$(x_{vi,3} = f_{qk}) \cap (x_{vi,4} = f_{qk}) \cap (y_{vi,3} = f_{qk}) \cap (y_{vi,4} = f_{qk})$$

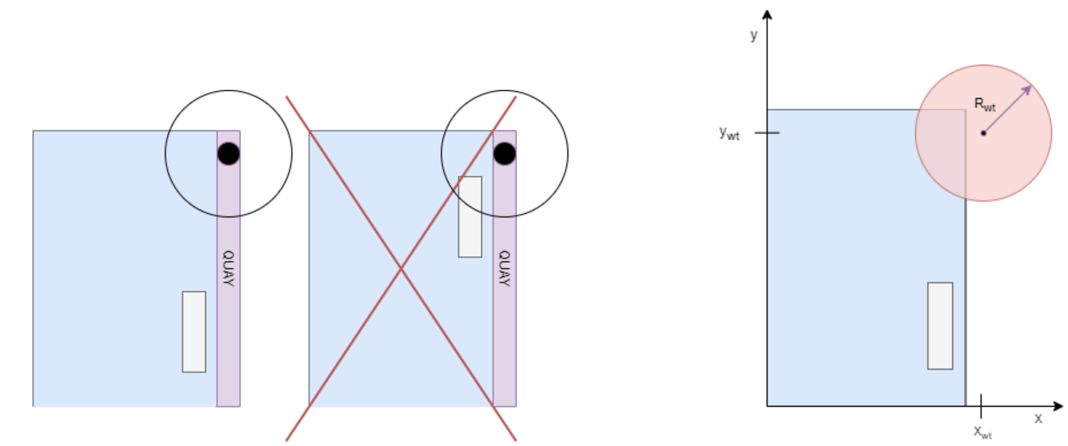
for

$$x_{qk,1} \leq x_{vk,m} \leq x_{qk,2}$$
$$y_{qk,1} \leq y_{vk,m} \leq y_{qk,2}$$

with m = 1,2,3,4 ; k = 1,2,..,p ; p being the amount of quay walls in the port basin.

4. Quay wall clash Vessels in the basin should not clash with the quay wall. If correct, restrictions set at paragraph 1 and 3 prevents this clash along with the formulation of the vessel geometry.

5. Safety contours In port basins safety contours play a role in the layout of the basin. The safety contours could be due to terminals, storage tanks or residential areas. An example of for safety contours is the region around a wind turbine (figure E.10)



(a) Practical representation

(b) Mathematical representation

Figure E.10: Safety contours of surrounding objects

The safety contour is a circle with an accompanying radius. The formula for a circle, translating the radius for Cartesian coordinates, is:

$$(x - a)^2 + (y - b)^2 = r^2$$

in which a and b are the coordinates of the centre (a, b). Translating the formula to the port planning case:

$$(x - x_{obj,l})^2 + (y - y_{obj,l})^2 = R^2_{obj,l,sc}$$

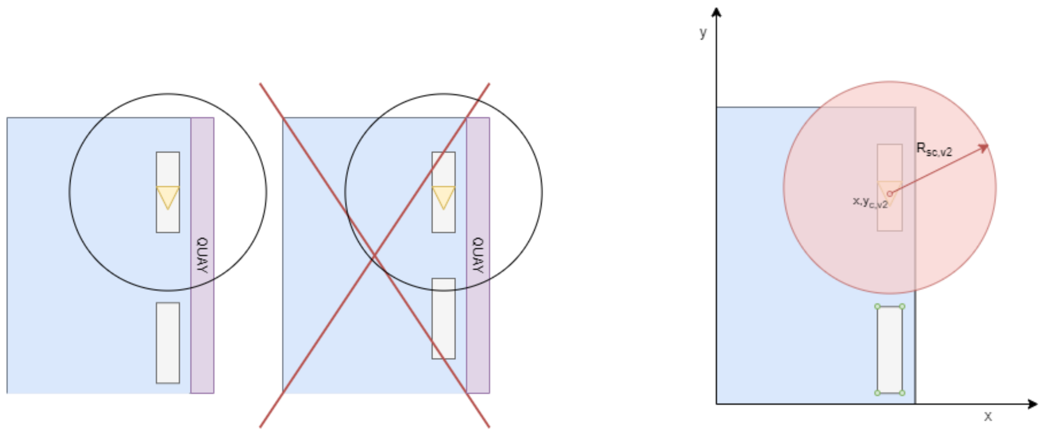
The vessels are not allowed to be in the circle, thus the following mathematical restriction is imposed. along a rewriting of the equation above is done:

$$y = y_{obj,l} \pm \sqrt{R^2_{obj,l,sc} - (x - x_{obj,l})^2}$$

$$(y_{vi,m} > y_{obj,l} + \sqrt{R^2_{obj,l,sc} - (x_{vi,m} - x_{obj,l})^2}) \cap (y_{vi,m} < y_{obj,l} - \sqrt{R^2_{obj,l,sc} - (x_{vi,m} - x_{obj,l})^2})$$

with m = 1,2,3,4 and $y_{obj,l}$ being object number l, with l=1,2,..,q; q being the number of objects with safety contours.

6. Vessel safety contours Some vessels carry dangerous goods with them, imposing a potential treat for surrounding vessels. Hence, a safety contour is imposed around the vessel. The safety contour for the vessel is depicted in Figure E.14.



(a) Practical representation (b) Mathematical representation

Figure E.11: Safety contours of vessels

The principle for determining the vessel safety contour is the same as for objects with safety contours. The domain restrictions for vessel safety contours is:

$$y = y_{vj,c} \pm \sqrt{R^2_{vj,c,sc} - (x - x_{vj,c})^2}$$

$$(y_{vi,m} > y_{vj,c} + \sqrt{R^2_{vj,c,sc} - (x_{vi,m} - x_{vj,c})^2}) \cap (y_{vi,m} < y_{vj,c} - \sqrt{R^2_{vj,c,sc} - (x_{vi,m} - x_{vj,c})^2})$$

6. Fairways The port basin requires a certain width in order to be operational. Two cases are presented in the design guide:

- Short basins
- Long basins

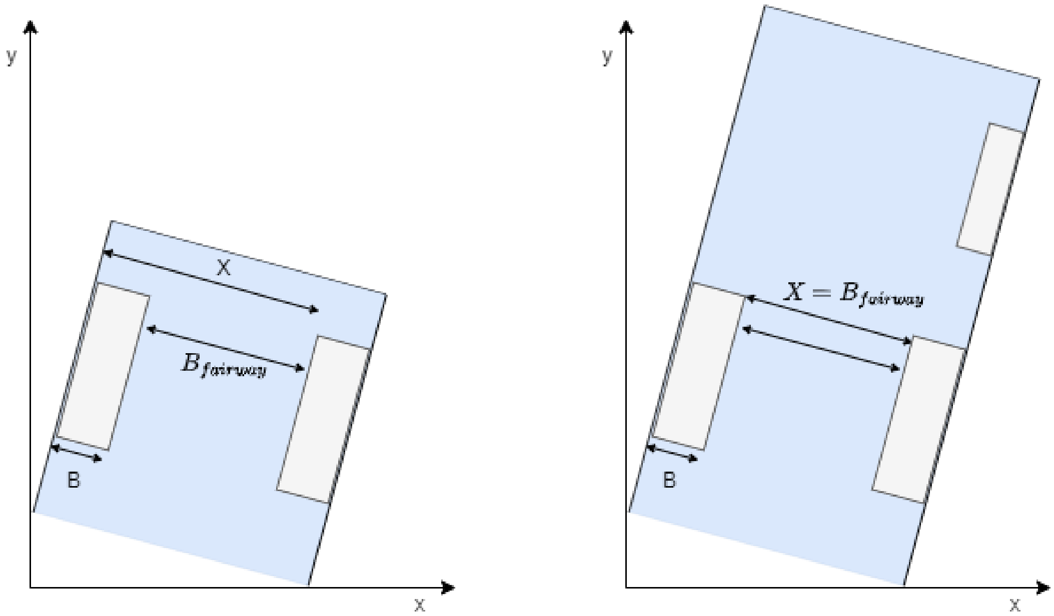
Short basins have a required width determined by the normative vessel berth, while long basins require a certain distance between two vessel berths. The guideline does not define what the difference between a long and short basin. Looking at the figures in the design guide, the assumption is made that a long basin can fit two normative vessels in sequence, while a short basin can only fit one. The following fairway width are assumed:

$$B_{fairway} \geq 2 * B_{vi} \;\; if \; L_{fairway} \leq 2.5 * LOA_{vi}$$
$$B_{fairway} \geq 3 * B_{vi} \;\; if \; L_{fairway} > 2.5 * LOA_{vi}$$

With B_{vi} being the normative vessel width and $L_{fairway}$ being:

$$L_{fairway} = \sqrt{(y_{qi,2} - y_{qi,1})^2 + (x_{qi,2} - x_{qi,1})^2}$$

The factor 3B is the general width for port basins, independent from location. In a further stadium this width could be made variable depending on location. The situations are depicted in figure E.12a and figure E.12b.



(a) Short basin fairway width (b) Long basin fairway width

Figure E.12: Basin width dimensions

The given fairway widths are not the design widths for the basins; the formulae are restrictions on the fairway. The width of the fairway is dependent on whether berths are on both quay sides. If both quay sides contain berths, the fairway width is equal to the distance between the berths. If only one of the quay wall sides has a berth, the fairway width is two times the width of the normative vessel B_{vi}.

The fairway should be accessible for the vessels entering the basin, which means that the depth of the fairway should be equal to the design depth. As a consequence the dredged volume in the basin is affected.

In order to mathematically express the domain for the fairway, the following assumptions are made:

- The fairway goes from entrance to the final vessel's operation zone;
- The fairway width is the perpendicular distance between two parallel quay walls;
- The width of the fairway is determined by the normative vessel width, if the minimum width fits between the berths, or;
- The fairway width is equal to the distance between parallel berths on opposite sides

In order to calculate the perpendicular distance between the quay walls, the equations for the quay wall have to be assumed. Quay walls are most of the time straight lines, which translates in a 2D plane to:

$$f_{qi} = m_i x + b_i$$
$$f_{qj} = m_j x + b_j$$

The two lines are assumed to be parallel, which makes m_i=m_j. The resulting perpendicular distance is:

$$d_{qi,qj} = \frac{|b_j - b_i|}{\sqrt{1 + m^2}}$$

From this distance the fairway width $B_{fairway}$ for long basins is calculated:

$$B_{fairway} = d_{qi,qj} - B_{vi} - B_{vj} \quad if \quad B_{fairway} \geq 3 * B_{vi}$$

$$else \ \ B_{fairway} = 3 * B_{vi}$$

and for short basins:

$$B_{fairway} = d_{qi,qj} - B_{vi} - B_{vj} \ \ if \ \ B_{fairway} \geq 2 * B_{vi}$$
$$else \ \ B_{fairway} = 2 * B_{vi}$$

The resulting fairways widths have to be plotted in a 2D plane. The fairway is in essence a transformed function with respect to the quay wall. The transformation done is perpendicular to the quay wall. The transformation can mathematically be described as a transformation along the x-axis or the y-axis. In this case the transformation is done along the y-axis.

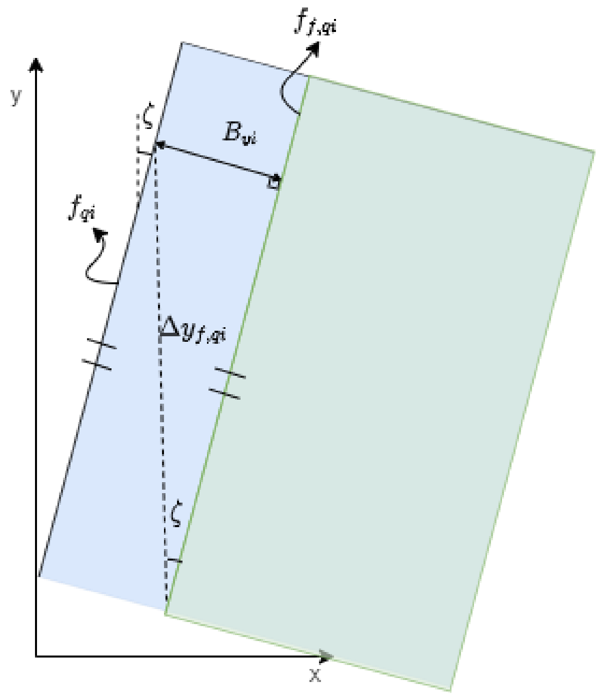


Figure E.13: Scheme for line transformation to find fairway boundary function (fairway indicated in green)

The transformation is done using geometry rules. In order to find the distance $\Delta_{f,qi}$, the angle ζhas to be determined. To get ζ, the linear equation is considered:

$$f_{qi} = mx + b$$

The formula for slope m is:

$$m = \frac{\Delta y}{\Delta x}$$

Te angle ζHowever, the angle ζis the inverse of the slope:

$$\zeta = arctan(1/m)$$

From geometry the distance $\Delta y_{f,qi}$ comes from geometry:

$$\Delta y_{f,qi} = \frac{B_{vi}}{sin(\zeta)}$$

The resulting transformed function is:

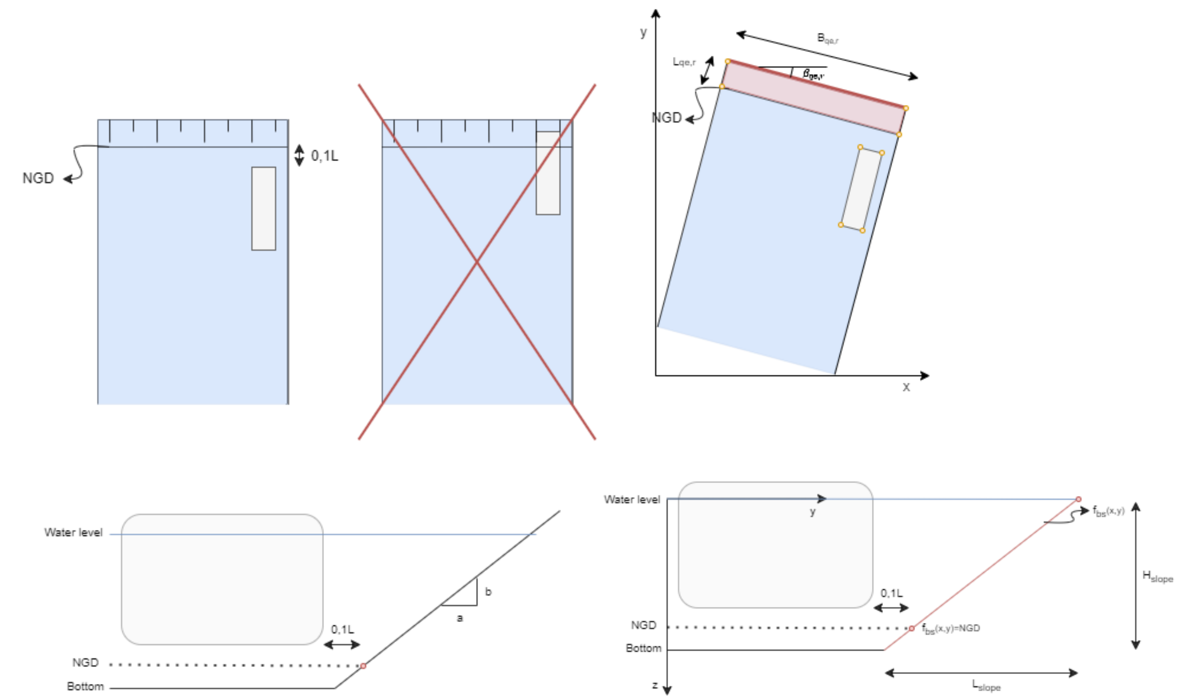
$$f_{f,qi} = f_{qi} - \Delta y_{f,qi}$$

The fairway consists of 2 boundary lines. Hence, the fairway boundary equation on the other basin side is:

$$f_{f,qj} = f_{qj} + \Delta y_{f,qj}$$

in which $f_{f,qj}$ uses the vessel width B_{vj} from the berth on the opposite quay side. Both function have a domain and reach starting at the entrance of the basin till the end of the basin.

Slope distance To prevent the vessels from penetrating the bed slope, restrictions are imposed for the distance between the vessel and the slope. The distance is parameterised to be 10 percent of the vessel length of the vessel nearest to the slope. The representation of the situation is shown below:



(a) Practical representation

(b) Mathematical representation

Figure E.14: Quay embankment distance vessels

In real life, the embankment goes from one basin edge to the opposite embankment. This means that when the basin edged have a different slope, the embankment gets wider or smaller as it goes in the water. However, in most cases the edges are in the same angle. Hence, it is assumed that the width is constant over the length of the embankment. The result is a rectangular zone.

The quay embankment is assumed to be modeled as a line, which is also the edge of the basin. The input for the line is two coordinates which are the intersections with the other quays, depicted in the figure below as ($x_{qe,r,1}$, $y_{qe,r,1}$) and ($x_{qe,r,2}$, $y_{qe,r,2}$). From these coordinates the quay embankment angle can be computed:

$$\beta_{qe,r} = arctan(\frac{y_{qe,r,2} - y_{qe,r,1}}{x_{qe,r,2} - x_{qe,r,1}})$$

The other coordinates are calculated from the angle $\beta_{qe,r}$. The formulae are:

$$x_{qe,r,4} = x_{qe,r,1} + L_{qe,r} * sin(\beta_{qe,r})$$
$$y_{qe,r,4} = y_{qe,r,1} + L_{qe,r} * cos(\beta_{qe,r})$$
$$x_{qe,r,3} = x_{qe,r,2} + L_{qe,r} * sin(\beta_{qe,r})$$
$$y_{qe,r,3} = y_{qe,r,2} + L_{qe,r} * cos(\beta_{qe,r})$$

The restrictions for the quay embankment are the same type of restrictions as the vessel operations zone. All the restrictions are described below. First the restrictions are set for -90<$\beta_{qe,r}$<0, depicted in figure E.16:

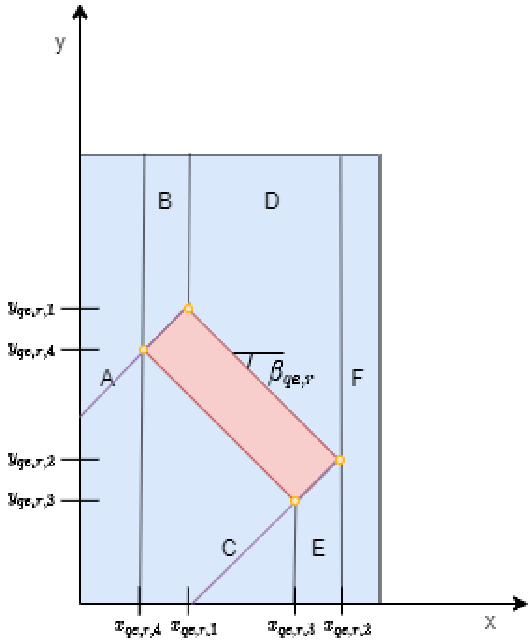


Figure E.15: Slope zone restrictions -90<$\beta_{qe,r}$<0

The restrictions are:

$$A:\ \bar{x}_{vi,m,oz} < x_{qe,r,4}$$

$$B:\ \bar{y}_{vi,m,oz} > y_{qe,r,4} + \frac{y_{qe,r,1} - y_{qe,r,4}}{x_{qe,r,1} - x_{qe,r,4}} * (\bar{x}_{vi,m,oz} - x_{qe,r,4})\ \ with\ \ x_{qe,r,4} \le \bar{x}_{vi,m,oz} \le x_{qe,r,1}$$

$$C:\ \bar{y}_{vi,m,oz} < y_{qe,r,4} + \frac{y_{qe,r,3} - y_{qe,r,4}}{x_{qe,r,3} - x_{qe,r,4}} * (\bar{x}_{vi,m,oz} - x_{qe,r,4})\ \ with\ \ x_{qe,r,4} \le \bar{x}_{vi,m,oz} \le x_{qe,r,3}$$

$$D:\ \bar{y}_{vi,m,oz} > y_{qe,r,1} + \frac{y_{qe,r,2} - y_{qe,r,1}}{x_{qe,r,2} - x_{qe,r,1}} * (\bar{x}_{vi,m,oz} - x_{qe,r,1})\ \ with\ \ x_{qe,r,1} \le \bar{x}_{vi,m,oz} \le x_{qe,r,2}$$

$$E:\ \bar{y}_{vi,m,oz} < y_{qe,r,3} + \frac{y_{qe,r,2} - y_{qe,r,3}}{x_{qe,r,2} - x_{qe,r,3}} * (\bar{x}_{vi,m,oz} - x_{qe,r,3})\ \ with\ \ x_{qe,r,3} \le \bar{x}_{vi,m,oz} \le x_{qe,r,2}$$

$$F:\ \bar{x}_{vi,m,oz} > x_{qe,r,2}$$

For 0<$\beta_{qe,r}$<90 the restrictions are depicted in the figure below:

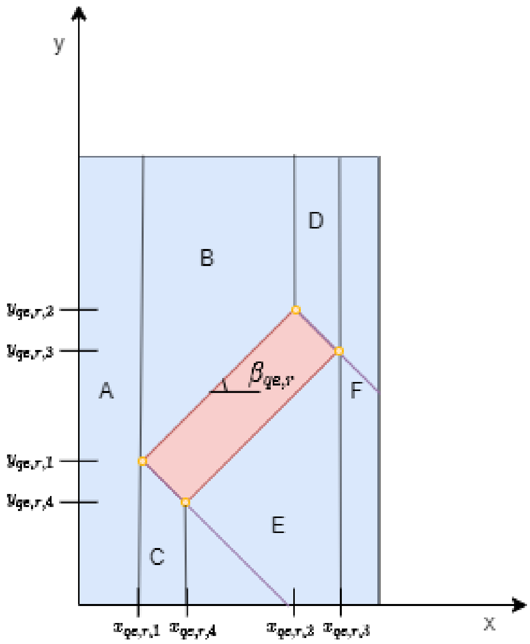


Figure E.16: Slope zone restrictions 0<$\beta_{qe,r}$<90

The restrictions are:

$$A: \; \bar{x}_{vi,m,oz} < x_{qe,r,1}$$

$$B: \; \bar{y}_{vi,m,oz} > y_{qe,r,1} + \frac{y_{qe,r,2} - y_{qe,r,1}}{x_{qe,r,2} - x_{qe,r,1}} * (\bar{x}_{vi,m,oz} - x_{qe,r,1}) \;\; with \;\; x_{qe,r,1} \leq \bar{x}_{vi,m,oz} \leq x_{qe,r,2}$$

$$C: \; \bar{y}_{vi,m,oz} < y_{qe,r,1} + \frac{y_{qe,r,4} - y_{qe,r,1}}{x_{qe,r,4} - x_{qe,r,1}} * (\bar{x}_{vi,m,oz} - x_{qe,r,1}) \;\; with \;\; x_{qe,r,1} \leq \bar{x}_{vi,m,oz} \leq x_{qe,r,4}$$

$$D: \; \bar{y}_{vi,m,oz} > y_{qe,r,2} + \frac{y_{qe,r,3} - y_{qe,r,2}}{x_{qe,r,3} - x_{qe,r,2}} * (\bar{x}_{vi,m,oz} - x_{qe,r,2}) \;\; with \;\; x_{qe,r,2} \leq \bar{x}_{vi,m,oz} \leq x_{qe,r,3}$$

$$E: \; \bar{y}_{vi,m,oz} < y_{qe,r,4} + \frac{y_{qe,r,3} - y_{qe,r,4}}{x_{qe,r,3} - x_{qe,r,4}} * (\bar{x}_{vi,m,oz} - x_{qe,r,4}) \;\; with \;\; x_{qe,r,4} \leq \bar{x}_{vi,m,oz} \leq x_{qe,r,3}$$

$$F: \; \bar{x}_{vi,m,oz} > x_{qe,r,3}$$

For $\beta_{qe,r}$=90, shown in the figure below, the restrictions are:

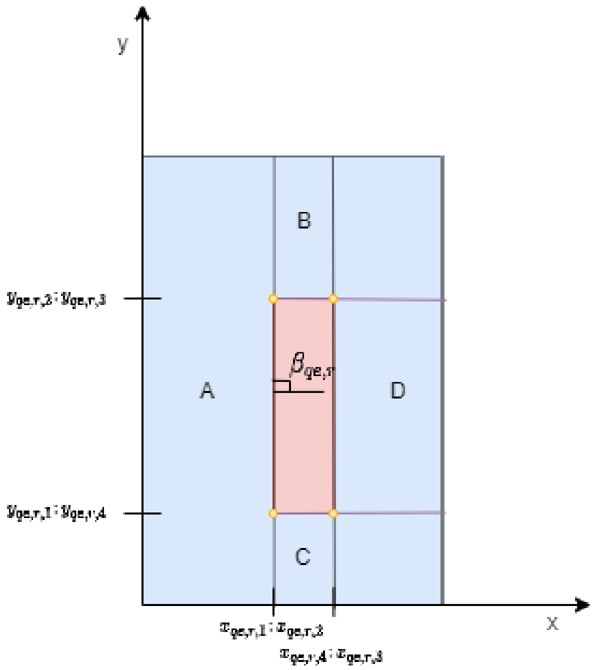


Figure E.17: Slope zone restrictions $\beta_{qe,r}$=90

The restrictions are:

$$A: \; \bar{x}_{vi,m,oz} < x_{qe,r,1}$$
$$B: \; \bar{y}_{vi,m,oz} > y_{qe,r,3}$$
$$C: \; \bar{y}_{vi,m,oz} < y_{qe,r,1}$$
$$D: \; \bar{x}_{vi,m,oz} > x_{qe,r,3}$$

For the case 90<$\beta_{qe,r}$<180, shown in figure E.18, the restrictions are:

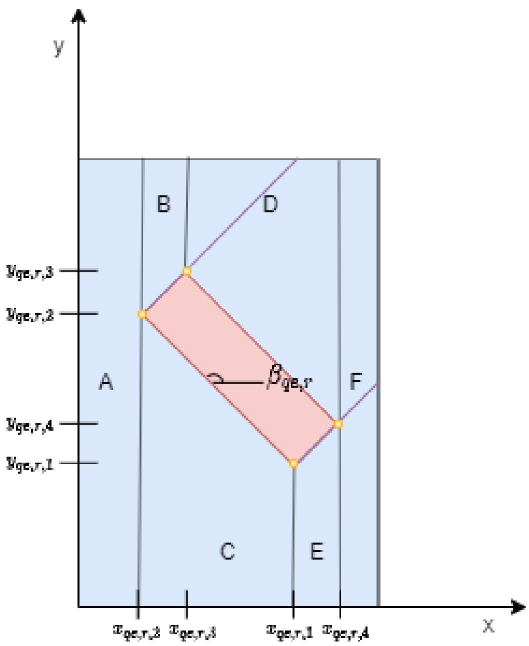


Figure E.18: Slope zone restrictions 90<$\beta_{qe,r}$<180

The restrictions are:

$$A: \ \bar{x}_{vi,m,oz} < x_{qe,r,2}$$

$$B: \ \bar{y}_{vi,m,oz} > y_{qe,r,2} + \frac{y_{qe,r,3} - y_{qe,r,2}}{x_{qe,r,3} - x_{qe,r,2}} * (\bar{x}_{vi,m,oz} - x_{qe,r,2}) \ \ with \ \ x_{qe,r,2} \leq \bar{x}_{vi,m,oz} \leq x_{qe,r,3}$$

$$C: \ \bar{y}_{vi,m,oz} < y_{qe,r,2} + \frac{y_{qe,r,1} - y_{qe,r,2}}{x_{qe,r,1} - x_{qe,r,2}} * (\bar{x}_{vi,m,oz} - x_{qe,r,2}) \ \ with \ \ x_{qe,r,2} \leq \bar{x}_{vi,m,oz} \leq x_{qe,r,1}$$

$$D: \ \bar{y}_{vi,m,oz} > y_{qe,r,3} + \frac{y_{qe,r,4} - y_{qe,r,3}}{x_{qe,r,4} - x_{qe,r,3}} * (\bar{x}_{vi,m,oz} - x_{qe,r,3}) \ \ with \ \ x_{qe,r,3} \leq \bar{x}_{vi,m,oz} \leq x_{qe,r,4}$$

$$E: \ \bar{y}_{vi,m,oz} < y_{qe,r,1} + \frac{y_{qe,r,4} - y_{qe,r,1}}{x_{qe,r,4} - x_{qe,r,1}} * (\bar{x}_{vi,m,oz} - x_{qe,r,1}) \ \ with \ \ x_{qe,r,1} \leq \bar{x}_{vi,m,oz} \leq x_{qe,r,4}$$

$$F: \ \bar{x}_{vi,m,oz} > x_{qe,r,4}$$

For the case $\beta_{qe,r}$=180 or $\beta_{qe,r}$=-180, shown in the figure below, the restrictions are:

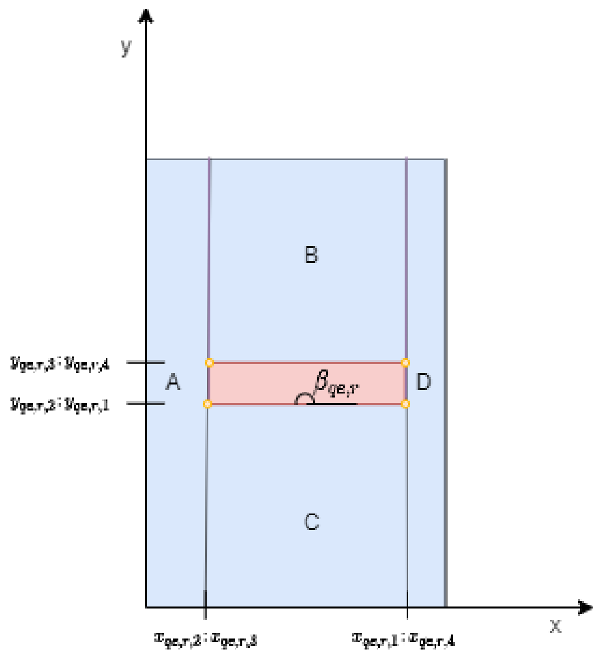


Figure E.19: Slope zone restrictions $\beta_{qe,r}$=180

$$A: \ \bar{x}_{vi,m,oz} < x_{qe,r,3}$$

$$B: \ \bar{y}_{vi,m,oz} > y_{qe,r,3}$$

$$C: \ \bar{y}_{vi,m,oz} < y_{qe,r,1}$$

$$D: \ \bar{x}_{vi,m,oz} > x_{qe,r,1}$$

For the case -180<$\beta_{qe,r}$<-90, shown in the figure below, the restrictions are:

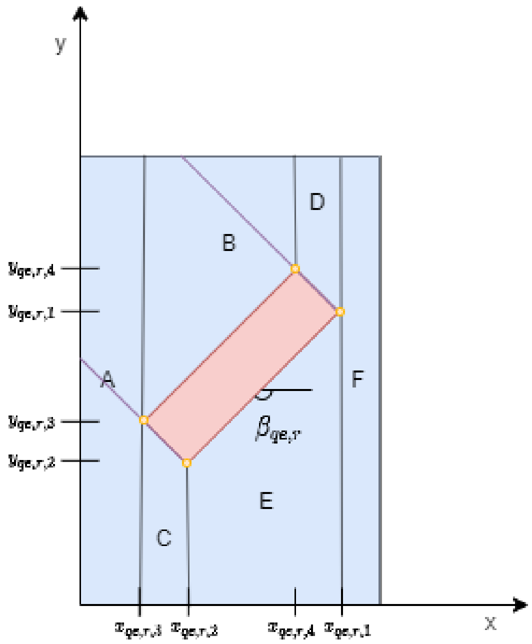


Figure E.20: Slope zone restrictions -180<$\beta_{qe,r}$<-90

$$A: \bar{x}_{vi,m,oz} < x_{qe,r,3}$$

$$B: \bar{y}_{vi,m,oz} > y_{qe,r,3} + \frac{y_{qe,r,4} - y_{qe,r,3}}{x_{qe,r,4} - x_{qe,r,3}} * (\bar{x}_{vi,m,oz} - x_{qe,r,3}) \; with \; x_{qe,r,3} \le \bar{x}_{vi,m,oz} \le x_{qe,r,4}$$

$$C: \bar{y}_{vi,m,oz} < y_{qe,r,3} + \frac{y_{qe,r,2} - y_{qe,r,3}}{x_{qe,r,2} - x_{qe,r,3}} * (\bar{x}_{vi,m,oz} - x_{qe,r,3}) \; with \; x_{qe,r,3} \le \bar{x}_{vi,m,oz} \le x_{qe,r,2}$$

$$D: \bar{y}_{vi,m,oz} > y_{qe,r,4} + \frac{y_{qe,r,1} - y_{qe,r,4}}{x_{qe,r,1} - x_{qe,r,4}} * (\bar{x}_{vi,m,oz} - x_{qe,r,4}) \; with \; x_{qe,r,4} \le \bar{x}_{vi,m,oz} \le x_{qe,r,1}$$

$$E: \bar{y}_{vi,m,oz} < y_{qe,r,2} + \frac{y_{qe,r,1} - y_{qe,r,2}}{x_{qe,r,1} - x_{qe,r,2}} * (\bar{x}_{vi,m,oz} - x_{qe,r,2}) \; with \; x_{qe,r,2} \le \bar{x}_{vi,m,oz} \le x_{qe,r,1}$$

$$F: \bar{x}_{vi,m,oz} > x_{qe,r,1}$$

E.3. Objectives

The quality of the solution is determined by the degree of objective satisfaction. The objectives in port planning range from dredging to X. the objectives are defined and mathematically described in this section.

Dredging The volume of dredged sediment determines the cost of the layout. The berth location determines the amount of required dredged sediment, because of the NGD provided by the port. In this section the dredging is done in two parts: the initial berth dredging and the profile equilibrium dredging.

The initial dredging is the dredging required to provide the vessel with the described NGD below the vessel. Mathematically speaking, the analytical way the bottom can be described determines the initial berth dredging function. Two situations are relevant:

1. The bottom is irregularly shaped and can't be expressed by an analytical function
2. The bottom can be expressed as an analytical function

In case 1) the dredged volume is calculated by means of a numerical approach. The bottom gets divided in small cells forming a grid together. For each cell where the vessel berth is placed, the cell depth has to be equal to the design depth.

In case 2) an integral provides the amount of sediment dredged to facilitate the vessel the integral is in the form of:

$$\iint_D f_b(x,y)\,\mathrm{d}x\mathrm{d}y$$

in which f_b is the bottom function and D is the dredging domain, dependant on the orientation of the vessel. Function f_b is a fitted function of the bottom data points. Before using this function, the function has to be tested whether it produces a function that fits the bottom well.

The presented manner of the initial berth dredging is not realistic, since it produces an unrealistic profile with straight vertical edged in the bottom profile. Therefore the profile equilibrium dredging is introduced: this the additional sediment that has to be removed in order to create a realistic bottom profile post initial berth dredging. The assumption for this profile is that the sediment requires a certain bed slope in order to obtain stability. This slope is now called the stability slope γ_{ss} (angle $\gamma$of slope stability). The angle for now is assumed to be 40°, which is the angle of internal friction for wet sand Geotechdata.info, 2013. The depiction of the dredging types is shown in figure E.21

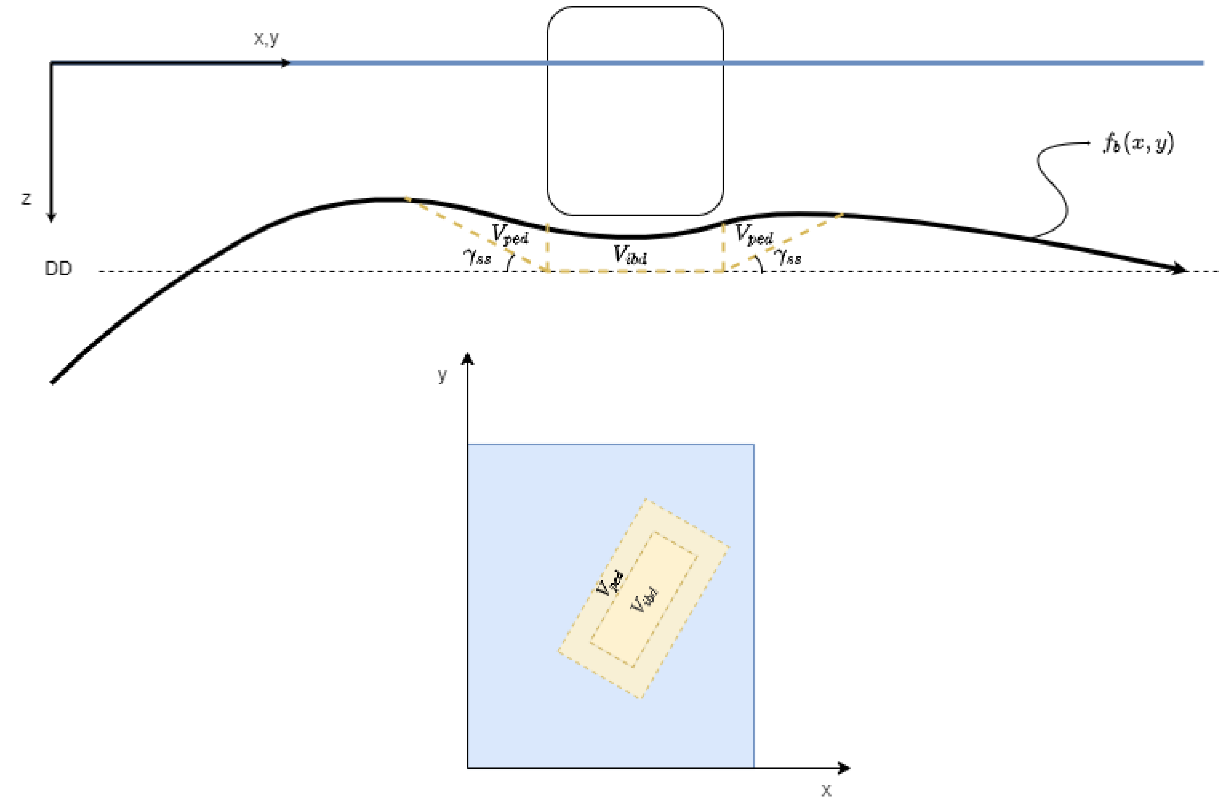


Figure E.21: Dredging function

E.4. Preliminary conclusion on geometric method

This approach is an approach which requires plenty of parameter definition. It is easy to lose oversight using many parameters and notations are prone to errors. Although the geometric approach is easy to comprehend, the execution can get tedious and hard to oversee.

Another conclusion regards the geometry of the basin. The basin is a combination of linear functions, which is unrealistic when applying a method to a real basin. Real basins, or port environments, have shapes and curves that cannot be described as a combination of linear functions. It is more realistic to assume that the port basin is a manifold, an abstract topological space. Creating an approach that works on manifolds is more widely applicable.

Finally, the geometric approach is not widely used in literature. This research is not about inventing the wheel and creating a new approach, it is about applying a design method to another problem type. Using the topological approach (meshes, integer space) is more common and more information about the approach is available.