The background of the entire page is a dense, intricate red truss structure, resembling a complex network of interconnected lines forming a mesh or lattice pattern. This pattern is consistent across all sections of the cover.

From least-weight to least-environmental- impacting truss structures

A method to minimize the total environmental cost with the Ground Structure Method for the
welding, coating and construction material

B. van Rhijn

MSc. Thesis, December 2022

From least-weight to least-environmental-impacting truss structures

*A method to minimize the total environmental cost with the Ground Structure Method for the
welding, coating and construction material*

by

Bart van Rhijn

to obtain the degree Master of Science
at the Delft University of Technology

Student number: 4750896
Project duration: September 2020 – December 2022

Graduation committee:

TU Delft

Prof. Dr. ir. M.A.N. Hendriks (chair)

Prof. dr. ir. J.G. Rots

Dr. ir. A.A. Mehrotra

Iv-Infra

Ir. W.P.J. Langedijk (supervisor)

Ir. P.J.C. van Lierop

Preface

This Msc. thesis is the final product to conclude my study on Structural Mechanics at the faculty of Civil Engineering and Geosciences. At the start of this thesis I did have a global understanding of the field of structural optimization, and I imagined a goal to work towards: going from least-weight to least-environmental-impacting truss structures. The thesis is started off with the sentence: *Climate change is changing the world*. At the start of my study at TU Delft I was a bit sceptical about the field of climate engineering, but now at the end of my study, now I am entirely sure of the fact that this change won't stop until we fix this pressing issue.

Writing the thesis was a bit like your average bus trip in the public transit. I did know the end goal, but along the way there were stops and some bumps. Some stops did appear to be unnecessary in the end, and some were mandatory. A necessary dive was the field of mathematical programming, which was completely new to me. And it was interesting to see how this is a distinct field within mathematics. I hope that this work may inspire others to start building constructions with a lower environmental impact.

First and foremost, I would like to thank Walter Langedijk for all his advice, patience and help to steer me in the right direction. Further I want to thank all the members of the committee: Pieter Langedijk, Max Hendiks, Jan Rots and Anjali Mehrotra for showing curiosity and for providing insightful feedback. Lastly, I am very grateful for having a warm and supporting family.

*Bart van Rhijn,
7 December 2022*

Abstract

Climate change is changing the world. Where previously efficient use of materials was mainly applied to realize cost savings, this can also help to reduce the footprint of a construction. There is a rather direct and obvious relationship between the amount of material required in a construction and its environmental impact. Of course, the environmental impact is not solely dependent on this metric, but minimizing the weight of the structure is a great starting point. This thesis focusses on minimizing the environmental impact of a welded truss bridge. In addition to the required volume of construction material, the environmental costs for the welding and conservation are also taken into account. A case study is performed on a bicycle bridge crossing a highway that is built in the Netherlands.

The thesis starts with a review of multiple methods that minimize the weight of a structure. Within the field of structural optimization, The Ground Structure Method (GSM) appears to be the most suitable method for large structures that consist of slender structural elements. Making utterly high refinements in the GSM-model will result in a structure with definitely the lowest volume possible. However, this structure will have lots of smaller and shorter elements that will require in total more welding and conservation. This will not lead to a least-environmental-impacting structure. Thus the main question arises:

Will, within the ground structure method, minimizing on the environmental impact result in a significantly different structure than a minimization on weight?

The objective function for the environmental impact consists of the three considered contributing factors: the construction material, welding and the conservation. The environmental impact for the three factors is determined with a Life Cycle Assessment (LCA). Finally, every contributing factor is unified into a single indicator value through the Environmental Cost Indicator (ECI) method.

The objective function also consists of three variables which represent: the volume of construction material, the welding volume and the surface of the structure. The volume is already known, since it is the regular GSM minimization. The welding volume can be determined through the joint-cost method, which is adding an artificial length to each member. The surface

area of the structure is harder to determine. The relation of the volume of a construction element and its surface area is generally speaking non-linear. Multiple implementations were investigated. The aim is to perform the optimization on a fully connected ground structure, and so the assumption is made to make the relation between the volume and surface area linear. A circular hollow cross section with a variable radius and a constant wall thickness is implemented into the optimization method. The final objective function to minimize the ECI costs is a mixed-integer linear programming problem (MILP).

This method is tested on the established benchmark for a cantilever structure and on a case study for a bicycle bridge. The shape of the optimal structure is dependent on the amount of nodes within the design domain. The results for the cantilever structure does clearly reflect this. Depending on the amount of nodes in the design domain, the minimization of the environmental impact is decreased between 0 and 37%, while the weight is at most 2.5% higher. The difference of the environmental impact between the least-weight and least-environmental-impacting structure keeps increasing as the node density increases.

The bicycle bridge is optimized in a 2D and 3D design domain. The design domain of the bicycle bridge appeared to be too big to be solved by the MILP formulation optimization, thus the domain is reduced to a single span of the bridge. Furthermore, the amount of nodes in the design domain is limited to improve the computability of the problem. In both the 2D and 3D variant the regular minimization on weight requires only a fraction of the time to solve the problem successfully. For the 2D case there is a difference between the minimization of the weight and ECI. The number of members in the least-environmental-impacting structure is reduced by 40%, which results in a 1% lower environmental impact. The MILP could not converge properly in 4 hours in the 3D design domain. This is mainly because the model size did increase a lot compared to the 2D design domain. Going from 2D to 3D adds a third axis, which increases the amount of constraints by 50%. Likewise, the number of nodes in a 3D domain increase more rapidly than in a 2D domain.

All in all, the method is implemented successfully and validated with the cantilever structure. The proposed method will result in a structure with an equal or lower environmental impact compared to the regular least-weight minimization. However the minimization of the

environmental cost with the proposed optimization method is able to solve problems with around 5,000 variables. To solve larger models successfully it is advised to either reduce the connectivity of the ground structure or to apply the joint-cost method with an LP.

Keywords: Ground Structure Method, Environmental Impact, Environmental Cost Indicator, Least-weight structures.

List of symbols

The following variables are being used:

\underline{a}	Area vector
a_i	Area of member i
A_{out}	Outer surface of the structure
$\underline{\underline{B}}$	Nodal equilibrium matrix
e_c	ECI cost for the conservation
e_v	ECI cost for the main construction material
e_w	ECI cost for the welding
\underline{f}^k	External load vector of load combination k
j	Joint cost
\underline{l}	Length vector
l_i	Length of member i
\mathbf{M}	big-M
$\min ECI$	Objective function to minimize the total Environmental Cost Indicator costs
$\min V$	Objective function to minimize the volume
\underline{q}^k	Internal force vector of load combination k
\underline{s}	Vector containing binary indicator variables
s_i	Binary indicator variable for member i
t	Wall thickness
t_w	Weld thickness
V	Volume construction material
V_w	Volume welding material
σ^-	Maximum allowable compressive stress
σ^+	Maximum allowable tensile stress

Contents

Preface.....	ii
Abstract	iii
List of symbols.....	vi
Chapter 1 Introduction.....	1
1.1 The case	2
1.2 Structural optimization.....	3
1.3 Environmental impact.....	6
1.4 Problem statement.....	8
1.5 Research objective	8
1.6 Research question	8
1.7 Reading guide	9
Chapter 2 More on the Ground Structure Method.....	11
2.1 Mathematical Programming.....	11
2.2 Michell structures.....	12
2.3 Ground structure	12
2.4 Mathematical description.....	15
2.5 Extensions.....	17
2.5.1 Joint cost	17
2.5.2 Adaptive member adding	18
2.5.3 Stability	18
2.5.4 Geometric optimization	19
2.6 Software for the Ground Structure Method	19
Chapter 3 Environmental Cost Indicator.....	21
3.1 Life Cycle Assessment.....	21
3.2 Goal and Scope of the ECI values	21

3.2.1 Goal.....	22
3.2.2 Scope.....	22
3.3 Environmental Impact Categories.....	23
3.4 Weight factors	25
3.5 Data collection	25
3.5.1 Structural Hollow Sections	26
3.5.2 Welding	27
3.5.3 Conservation	28
3.6 Conclusion	29
Chapter 4 Minimization of the environmental impact.....	30
4.1 Volume	30
4.2 Volume welding	30
4.3 Surface area.....	31
4.4 Main optimization method – hollow cross-section with a constant wall thickness	33
4.5 Implementation	35
Chapter 5 Running the optimization method.....	36
5.1 Cantilever beam	37
5.2 Case study	40
5.2.1 General information.....	40
5.2.2 As-built load bearing system.....	41
5.2.3 Design domain	42
5.2.4 Loads.....	42
5.3 2D optimization as-built structure	44
5.4 2D optimization of the bicycle bridge	46
5.5 3D optimization of the bicycle bridge	48
5.6 Summary.....	53

Chapter 6 Discussion	55
6.1 Environmental Cost Indicator	55
6.2 Case study	56
6.3 Optimization method	57
Chapter 7 Conclusion	59
Chapter 8 References	61
APPENDIX	64
A – Data sources environmental impact	64
B – Loads for the case	66
C – Code	72
D – Joint-cost method	84

Chapter 1 Introduction

Climate change is a global crisis that regularly fills the headlines of newspapers. Today, the construction industry is responsible for approximately 40% of material use and 30% of greenhouse gas emissions worldwide (Lagaros, 2018). The European Union has made a commitment to strive to be climate neutral by 2050 and have a fully circular economy, as stated in the European Green Deal¹. An enormous amount of work still needs to be done to achieve this. Every step in the right direction, regardless of size, brings us closer to this goal.

Where previously efficient use of materials was mainly applied to realize cost savings, this can also help to reduce the footprint of a construction. A number of methods have been developed to generate structures wherein the construction material is placed as efficient as possible. These methods fall under the subject Structural Optimization. There is a rather direct and obvious relationship between the amount of material required in a construction and its environmental impact. Of course, the environmental impact is not solely dependent on this metric, but minimizing the weight of the structure is a great starting point.

This thesis focusses on minimizing the environmental impact of a truss bridge. In addition to the required volume of construction material, other factors are also included in the calculation of the environmental impact. In this chapter the case is described, structural optimization in more detail, and the frame for the environmental impact is explained. This to come to a problem statement and the research methodology at the end of this chapter.

¹ Website: https://ec.europa.eu/info/strategy/priorities-2019-2024/european-green-deal_en

1.1 The case

A truss bridge was built over the highway A2 near Maarssen in 2012. This structure will be used as a case study. It is a slender steel structure with welded joints. The bridge is intended for pedestrians and cyclists and occasionally a maintenance vehicle drives over it. It has two main spans of approximately 40 meters each. The bridge deck is separated from the main supporting structure. In this thesis, the environmental impact generated by the main load-bearing structure is discussed in more detail. The foundation and the bridge deck are not taken into account.



Figure 1. This bicycle bridge over the highway A2 will be used as a case study.

1.2 Structural optimization

Formally stated, structural optimization is done by means of an *objective function* which contains certain *design variables*, e.g. cost of materials and labour, structural strength, weight, etc. The objective function is minimized (or maximized), while being subjected to (Rozvany, 1992):

- i. Geometrical constraints; e.g. design domain or limited variations of the cross-sectional shapes.
- ii. Behavioural constraints; e.g. restrictions on stresses, displacements, stability, natural frequencies.

The structural optimization problems can be classified as:

- i. *sizing optimization* where the design variables are most frequently chosen as the cross-sectional area of the elements of the structure, or
- ii. *shape optimization* where the boundaries of the design domain are optimized, which can be applied to continua (e.g. a hull), or the interfaces between composite materials, and
- iii. *layout or topology optimization*, which aims to assign material in the most effective way within the design domain. Hence, layout optimization consists also of sizing and shape optimization.

In Figure 2 an example is shown for the different types of optimizations. For the sizing optimization (a) the size of the beams is optimized. With the shape optimization the boundaries of the white circles within the beam are optimized to reduce the material (b). In this example (c) topology optimization is applied where material from the design space is removed such that the stiffness of the beam is maximized with a targeted volume ratio. Note that this is a fairly simple example and that there are a lot of other formulations and methods available.

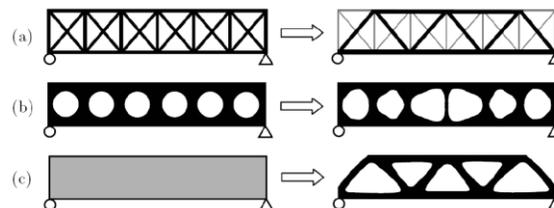


Figure 2. Three structural optimization categories: a) sizing, b) shape and c) topology optimization. The initial design domain is shown on the left and the final optimized design is shown on the right (Mozumder, 2010).

Layout optimization is the most complex task in SO because there are in theory an infinite number of possible topologies, and simultaneously the size of nonvanishing members should be determined (Rozvany, 1992). During the last few decades, layout and topology optimization has become a preferred tool that is used for the automotive and aerospace industries, but it is less so in civil engineering (Baandrup, et al., 2020); (Lagaros, 2018). At first sight these industries seem to have a lot of similarities with civil engineering: they also build relatively large objects. However, there are some major fundamental differences in the design process in relation to the other industries. Some of the challenges to apply layout optimization within the design process in civil engineering are:

- i) The production volume in the other industries is higher. Steel bridges are unique structures thus the investments that are needed for a thorough design is spread over only one product, and therefore the budget is under more pressure.
- ii) In the other industries there is generally a lead producer that is more leading in the design process and production than clients or contractors in the construction industry. Due to the fragmentation the investments for development do rely on a company that is being part of a small fraction of the whole process (Coenders, 2011).
- iii) In the construction industry there is a lot less prototyping or even none at all. The first prototype is usually the building itself (Coenders, 2011).
- iv) Low volume fractions in the design space are occupied with material. It is therefore challenging to cover multiple length scales: e.g. bridges that can span over a kilometer has widths in the order of 20-40 meter are built from plates with a thickness as low as 8 millimeter (Baandrup, et al., 2020).
- v) Most methods optimize based on maximizing stiffness or minimizing the volume. It is challenging to also include the cost, complexity or other less tangible parameters in the optimization problem.

Above are some concerns noted which may make it difficult to apply layout optimization within the construction industry. It is important to note that the layout optimization can be categorized in two major classes: a continuum or discrete mechanical formulation. The continuum formulation is based on solid elements, whereas the discrete formulation is based on

higher order elements (e.g. a truss or beam). Due to this difference not all the mentioned challenges (i to v) weigh as heavy on every optimization method.

One of the best-known continuum methods is the Solid Isotropic Microstructure with Penalisation (SIMP), which aims to maximize the stiffness while the volume is kept constant. This method can be applied to design an optimal bridge, but a gigantic amount of computational power is required to get results with a good resolution (Baandrup, et al., 2020).

A discrete layout optimization method is the Ground Structure Method (GSM). In Figure 3 are the solutions for an optimal cantilever shown with the discrete GSM model on the left and a continuum SIMP model on the right (He., et al., 2018). It can be seen that the continuum model will converge towards the same type of solution as the discrete model, but the latter uses much more CPU time. The GSM is more suitable for large scale structures with slender elements. The discrete elements make the low volume occupancy and the challenging length scales less of a problem.

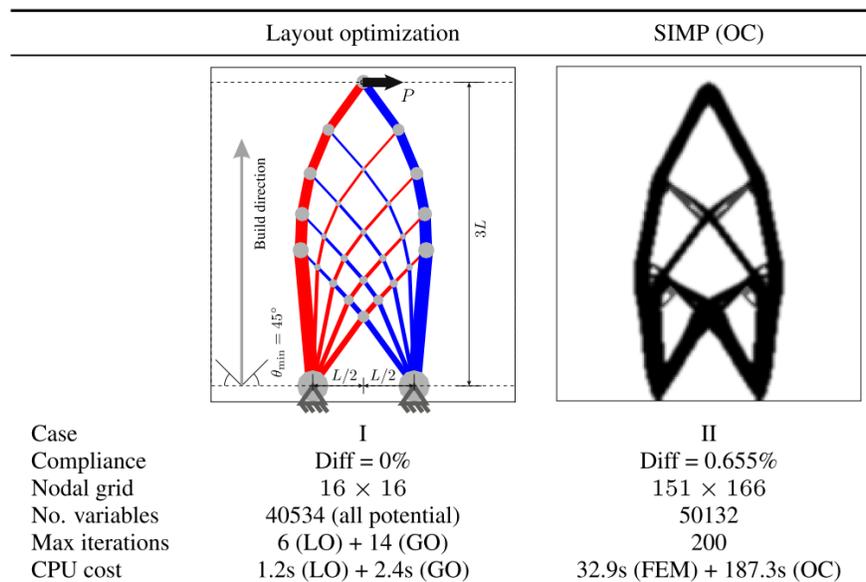


Figure 3. A comparison of the solutions obtained via layout optimization and continuum topology optimization. (He., et al., 2018)

1.3 Environmental impact

The CO₂ emissions are often at the centre of the discussions in the global climate debate. In the Netherlands, this is supplemented by a nitrogen crisis and a PFAS debacle. These emissions are being discussed, but these are by no means all the emissions that affect the climate and the environment. A method that provides good insight into the environmental impact of a construction is a Life Cycle Assessment (LCA). The LCA takes all kind of environmental aspects into account, including the CO₂ and nitrogen emissions. The industry standard NEN-EN 15804:2012 is used to define the scope, as shown in Figure 4.

When a bicycle bridge is being built, a whole chain of processes is started. Raw materials are mined, transported to blast furnaces, forged into steel and finally finished under a dynamometer to be rolled into CHS (stage A1-3). Then it is transported to the fabrication shed, assembled and then driven to the bridge and place it (stage A4-5). As maintenance, the bridge construction is repainted once every 20 years (stage B2). These are a few examples. LCA is a method that provides insight into the environmental impact of the entire chain.

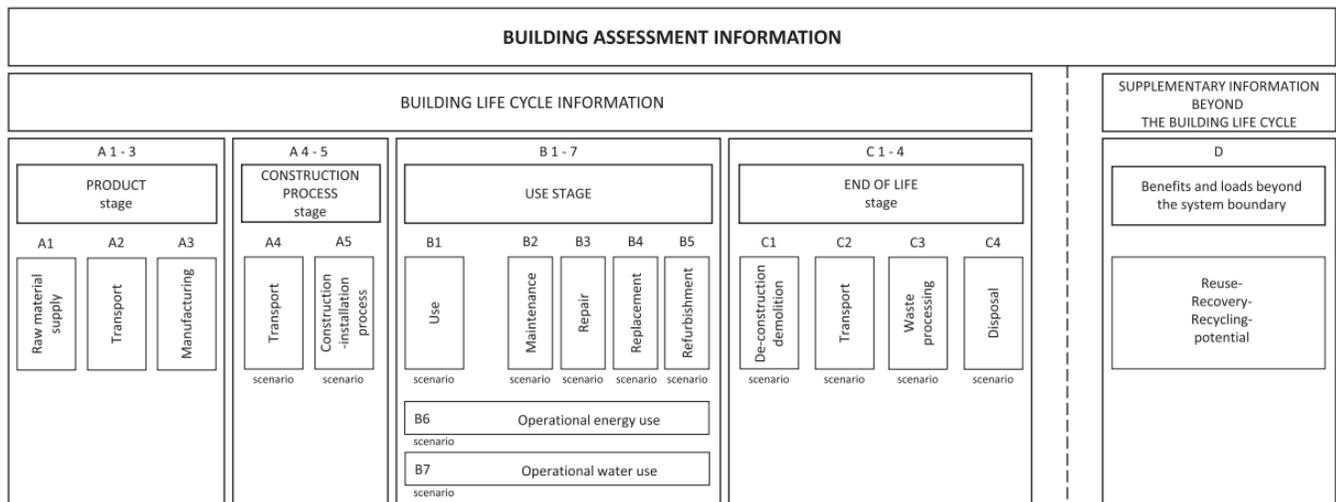


Figure 4. The specific life cycle stages considered in LCA according to standard NEN-EN 15804:2012

In this thesis are, for the calculation of the environmental impact, the making of the semi-finished products (A1-3), the welding from the construction phase (A5) and having to repaint as part of the maintenance (B2) are included into the scope. The transport in the construction phase (A4) is not considered. It is assumed that the distribution of the required materials is a fairly constant factor and is therefore independent of the size of the beams used. During the user phase,

only regular maintenance is considered (B2). During use, no energy is used by the bridge structure (B1; B6-7). Repair and replacement (B3-5) is not included, because this is something that occurs incidentally and is therefore unpredictable. The type of material used (steel) is constant. Therewith it is assumed that the results from the end-of-life (C1-4) are also constant.

Environmental cost indicator

The environmental impact is divided up into multiple impact categories. All the emissions are expressed in an equivalent unit of that category. For example, a frequently used impact category is the global warming potential (GWP), which is expressed in kg CO₂ equivalent units. Methane is also a greenhouse gas with more warming potential than CO₂, thus the conversion factor for methane to CO₂ equivalent units is bigger than one.

Still, it is not straight forward to compare the difference of the environmental impact between two products, because of those several impact categories. One way to weigh the environmental impact is through the Environmental Cost Indicator (ECI). With this method every impact category is assigned a monetary value, that should represent the costs to make the environmental impact undone or to make it sustainable. The Environmental Cost Indicator is used in this thesis to express the environmental damage of a product.

Comparison CO₂ emissions between construction steel and welded steel.

Welded steel has a higher Global Warming Potential (measured in kg CO₂ eq./kg) than hot- or cold-rolled steel. According to the ICE database, the world wide average for steel construction material has a Global Warming Potential (GWP) ranging between 1.27 - 3.01 kg CO₂ eq. per kilo steel depending on the production method and the required quality. The GWP of metallic arc welding varies between from 9 to 19 kg CO₂ eq per 1 meter of welding a 25 mm thick plate with a V-joint depending on the type of arc welding applied, which is equal to 2.5 - 5.33 kg CO₂ eq. per kilo of welding (Favi, et al., 2019).

1.4 Problem statement

It is clear that a bridge made from mainly hot-rolled steel has a lower environmental impact than exactly the same bridge manufactured through additive manufacturing. The girders are made in mass production and the process is well optimized. Additive manufacturing can't compete with this efficiency, and as a result, the environmental impact for a cubic meter of hot-rolled steel is lower than a cubic meter of welding. Making utterly high refinements in the GSM-model will result in a structure with definitely the lowest volume possible. However, this structure will have lots of smaller and shorter elements that will require in total more welding and conservation, whom have their own contribution to the total environmental impact, which isn't taken into account in the current GSM-model. Therefore, it is clear that least-weight structures does not automatically lead to least-environmental-impacting structures, but how significant is the difference?

1.5 Research objective

The goal of this research is to minimize the environmental impact of a bicycle bridge. As a starting point the minimal required amount of construction material will be used, which is the least-weight structure. Then there will be looked into how the objective function has an influence on the environmental impact. The environmental impact will be expressed with the Environmental Cost Indicator, as discussed in *1.3 Environmental impact*. Three contributing factors are taken into account, namely the required amount of construction material, welding and the coating. The choice of material is constant throughout the optimization. The least-environmental-impacting structure is benchmarked against the least-weight structure.

1.6 Research question

First of all, the general Ground Structure Method does minimize only the total volume of construction material. A method needs to be derived which is expanding the minimization to also take the welding and conservation into account. And so, the first question is:

How to minimize the ECI costs for the construction material, welding and conservation through the Ground Structure Method?

And subsequently, for the given method derived from this initial question one may wonder what the difference in the outcome is for the generated structures, thus:

Has the minimization on the environmental impact with the Ground Structure Method an significant influence on the generated truss structures, compared to the usual GSM with weight minimization?

1.7 Reading guide

The approach of this research is given in this paragraph with a reading guide.

Chapter 2 More on the Ground Structure Method

The GSM seems to be the best optimization method for slender structures. In the second chapter more information is gathered on the Ground Structure Method. Relevant questions are:

What is the background of the optimization method used for the GSM?

What are the characteristics of the GSM?

What type of extensions are there for the GSM to enhance its usability?

What software is useful for the research?

Chapter 4 Environmental Cost Indicator

The Environmental Cost Indicator (ECI) is used as pivot values in the optimization method. The main characteristics of the bicycle bridge (section 1.1) are used to determine estimates of these pivot values. Of course, the exact ECI cost can vary, among other things, on the location and project. As mentioned in section 1.3, the boundaries are chosen as such that these project dependent variations are limited as much as possible. The main question is:

What is the ECI cost for the construction material, welding and the conservation?

Chapter 3 Minimization of the environmental impact

With the knowledge on GSM and MP methods the minimization ECI is determined.

What assumptions have to be made?

How to minimize the ECI cost for a single member?

How to minimize the ECI cost for the whole model?

How to implement the method in practice?

Chapter 5 Running the optimization method

Does the ECI contribution of the welding and the conservation have a significant influence on:

- the environmental impact?
- the layout of the optimal structure?

Chapter 6 & 7 Discussion, Conclusion and recommendations

Finally, the main question of the research can be answered and there is some room to reflect on the optimization method, the usage of ECI and the case study.

Chapter 2 More on the Ground Structure Method

The background and the fundamentals of the Ground Structure Method are investigated in this chapter. It is started off with a brief introduction into mathematical programming, which is where the Ground Structure Method (GSM) is built on. Next, the Ground Structure Method is investigated, and finally relevant software is examined.

2.1 Mathematical Programming

A few key concepts of Mathematical Programming (MP) are required to be able to understand the GSM. MP is a mathematical optimization problem, which has the general form of:

$$\begin{aligned} &\text{minimize} && f_0(x) \\ &\text{subject to} && f_i(x) \leq b_i \quad i = 1, \dots, m \end{aligned}$$

In this function the x is a vector contains the decision variables $x = x_1, \dots, x_n$ which represents the choice made. It is optimized through the objective function f_0 , while being subjected to (in-)equality constraints that are given in the functions $f_i(x) \leq b_i$. The values of b_i are the bounds for the constraints. The solution from a MP represents the minimum cost, or maximum utility, for the vector x .

There is not a closed analytical solution available for the general optimization problem, however, for specific types of optimization problems are very effective methods at hand that can be used. The optimization problem can be categorized based on the type of variables, functions and bounds used. The most important step is recognizing whether the MP is convex or not. For convex problems efficient algorithms are available that can optimize a system with thousands, or even millions, of variables. Due to the convexity, global optimality is also assured. A special type of convex problem is the Linear Programming (LP) problem where all the functions are linear. If the objective function or one of the constraints is non-linear then it is called a non-linear programming problem (NLP). NLP which are not known to be convex can be extremely hard to solve, such that even for smaller systems with a few hundreds of variables can become intractable (Boyd, 2004).

Software

There is a vast amount of software packages available that are able to solve various types of MPs. Professional packages, such as MOSEK (MOSEK.com), Gurobi (gurobi.com) or IBM ILOG CPLEX (ibm.com/analytics/cplex-optimizer), run very fast and do come with loads of options. There are also free packages available (e.g. cvxpy), however these have always less options and do run most of the time slower.

2.2 Michell structures

Much research has already been carried out into the design of a framework. The optimal truss with minimal use of materials is defined in the 1904 work *Limits of Economy Material in Frame-structures* published by A. Michell (Michell, 1904). These minimal trusses now also bear the name Michell Trusses. Figure 5 shows the optimal truss for a cantilever beam. The solution for the minimal truss is two continuous spirals that meet orthogonally on the line AB. As a consequence of the fact that it is a continuous solution, this solution contains an infinite number of bars. The image only shows the flow lines (strain-rate field) that the bars to be placed must follow.

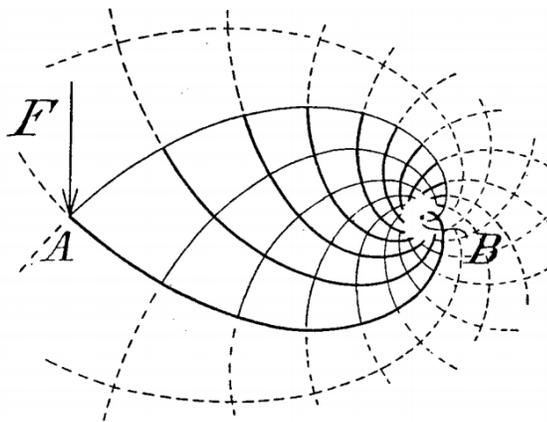


Figure 5. A force F at location A is balanced by a moment $F \times AB$ and an equal opposing force on B . The minimal truss is formed by two spirals with B as origin that meet orthogonally in A . The truss consists purely of connections with a right angle (Michell, 1904).

2.3 Ground structure

To make this solution more useful, a discrete solution for the minimal truss has also been created, first introduced by W. Prager in *A note on discretized Michell structures* in 1974. Over time, this method has been further developed and continues now under the name Ground

Structure Method (GSM). The naming is derived from the ground structure that is created when the model is initiated. The number of possible elements is, by making the ground structure, reduced from an infinite to a finite amount. The number of potential bars used in the ground structure depends on the number of nodes and the connectivity between the nodes. Figure 6 shows an example of a ground structure with a constant number of nodes and different levels of connectivity between the nodes.

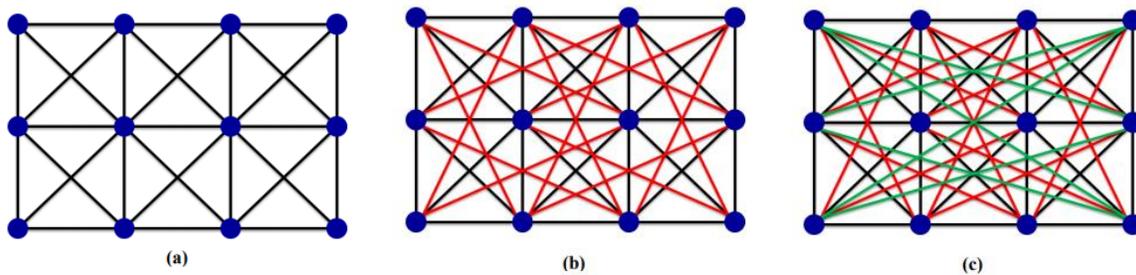
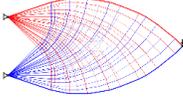
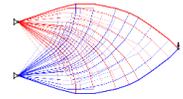
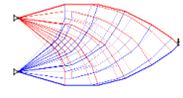
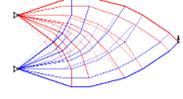
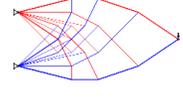
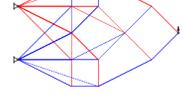
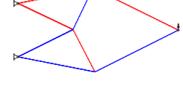
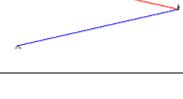


Figure 6. An example of a ground structure with a) level 1 connectivity, b) level 2 connectivity and c) level 3 connectivity (Zhao, 2014).

The distance between the nodes has an influence on the outcome of the solution. This makes the solution mesh-dependent, i.e. placing the nodes closer together and adding more potential bars results in a solution with a mesh that is more refined than one with a mesh with a larger distance between the nodes. In Table 1 the same cantilever is calculated, varying only the node density. The first solution comes very close to the exact solution for a continuous Michell truss due to its fine mesh. Lowering the node density has a greater impact on the number of bars needed than on the volume of the structure.

Table 1. The effect of various refinements of a discrete Michell truss on the total volume and the number of bars required.

Refinement	Volume	Number of bars	Relative amount of bars
	1.000	424	1.000
	1.002	330	0.778
	1.005	162	0.382
	1.008	114	0.269
	1.017	52	0.123
	1.030	44	0.104
	1.030	34	0.080
	1.051	8	0.019
	1.420	2	0.005

2.4 Mathematical description

In the previous paragraphs relation between the continuous Michell truss and discrete ground structure is explained. This paragraph elaborates on the mathematical aspects to generate such structures. In Figure 7 the steps in the optimization process are shown. After the design domain with the loads and supports is defined (a), the design domain is populated with nodes (b). Members are placed between every node to create a network of potential members (c). In this example every single node is connected with every other node in the domain, or in other words, the nodes are fully connected. This will result in $n(n - 1)/2$ number of members, with n being the number of nodes. A lower number of potential members could be used, but this should be done with care, since the optimal structure can only be formed from members that are available in the initial potential member set.

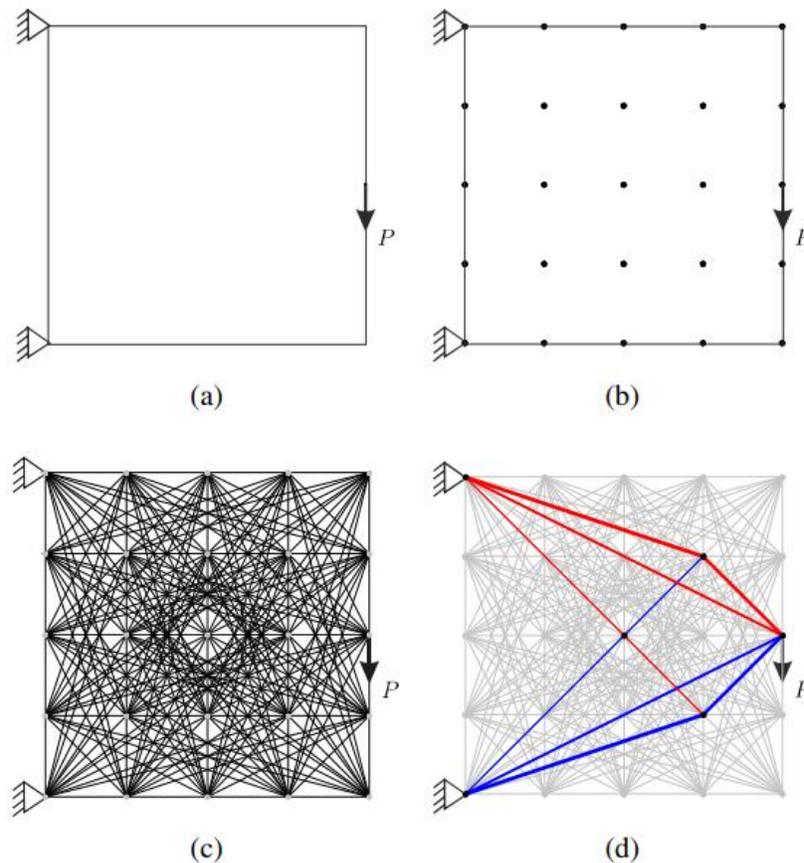


Figure 7. Steps in layout optimization: a) specify design domain, loads and supports; b) place the nodes within the design domain; c) make the set of all potential members, which is in this case full connectivity between the nodes; d) use the optimization to come to the optimal layout.

The next step is to optimize the members through the GSM. In this thesis the plastic formulation is used (Pritchard, et al., 2005), because it has some advantages over the elastic formulation. For designs with multiple load cases, the elastic formulation can converge to local optima, while the plastic formulation guarantees to be globally optimal (Zegard, et al., 2014). In the ground structure are $n = 1 \dots N$ nodes and $m = 1 \dots M$ members:

$$\min_{a, q} V = \underline{l}^T \underline{a} \quad (\text{eq. 2.1a})$$

$$\text{s.t.: } \underline{B} \underline{q}^k = \underline{f}^k \quad (\text{eq. 2.1b})$$

$$\underline{q}^k \geq -\sigma^- \underline{a} \quad (\text{eq. 2.1c})$$

$$\underline{q}^k \leq \sigma^+ \underline{a} \quad (\text{eq. 2.1d})$$

$$\underline{a} \geq \underline{0} \quad (\text{eq. 2.1e})$$

With:

V	Volume
σ^+, σ^-	Maximum allowable tensile and compressive stress
$\underline{l} = [l_1, l_i, \dots, l_M]$	Member lengths
$\underline{a} = [a_1, a_i, \dots, a_M]$	Member areas
$\underline{q} = [q_1, q_i, \dots, q_M]$	Internal member forces
$\underline{f} = [f_{x1}, f_{y1}, f_{xj}, f_{yj}, \dots, f_{xN}, f_{yN}]$	External forces
\underline{B}	Nodal equilibrium matrix $[2N \times M]$
$k = 1, 2, \dots, p$	Load cases

The objective is to minimize the volume V through variables \underline{a} and \underline{q} . Of course, the area must be larger than or equal to zero. If an area a_i takes the value 0, the member is unavailable and absent in the optimal structure. The internal member forces are intermediary variables that can be either positive (tension) or negative (compression). The equilibrium matrix \underline{B} is assembled from the local nodal equilibrium matrixes \underline{B}_j . It is written as a 2D problem, thus the equilibrium matrix \underline{B} and force vector \underline{f} have $2N$ rows. For a single load case ($k = 1$) the structure is only optimal if the upper or lower internal member force inequality is active for all members. Optimizing the problem given in Figure 7c with the objective (eq. 2.1a) and

constraints (eq. 2.1b-e) will result in the optimal structure as shown in Figure 7d. The grey members have zero area, and are not used in the optimal structure. Red represents members in tension and blue are the members in compression.

2.5 Extensions

The given minimization in equation 2.1a-e is a bare bone method that will result in optimal structures for that particular case, which is not directly usable for real world applications. Numerous research papers have been published on the subject. In the last two decades the GSM has gained a lot more attention, mainly because of the computational advantages of using the adaptive member adding method (2.5.2). In the subparagraphs below are a few important extensions explained that do increase either the utility applicability of the GSM.

2.5.1 Joint cost

Adding an additional cost to every member is an effective method to alter the outcome of the least-weight truss, which embodies large amounts of joints. The paper *Joints in optimum frameworks* (Parkes, 1975) aimed to reduce the amount of nodes through adding a joint cost per member. Calling this the joint radius would be a closer description, because all the members have a constant joint cost j added to each member length (see Figure 8). In this case only the objective function is altered. The constraints can be kept the same as defined in equation 2.1b-e.

$$\min V = \sum_i^M (l_i + 2j)a_i \quad (\text{eq. 2.2})$$

With:

j joint cost

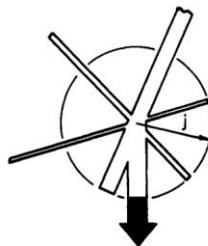


Figure 8. Multiple members are connected to a single node. Every member that connects to the node has a joint cost j added to their length. In this figure it is illustrated as a joint radius (Parkes, 1975).

2.5.2 Adaptive member adding

The GSM does work great on problems with a limited scale. However, the problem size grows quadratically with the number of nodes. The problem size is the number of members in the ground structure. This makes the method unpractical to use for medium to large scale problems, because of the sheer number of members, which results in high computational costs. The adaptive member scheme solves this problem (Gilbert, et al., 2003). Instead of calculating the whole problem with all its potential members a subset is taken, the active member set. A good starting point is using a subset that has level 1 connectivity (see Figure 6). Members from the potential member set that have the highest virtual strain, are added iteratively to the active member set. Optimality is reached when all the potential members have a virtual strain equal or lower than 1 (Pritchard, et al., 2005).

2.5.3 Stability

The optimal ground structure will often result in a structure containing very slender members under compression, making the structure sensitive to various types of instabilities, as shown in Figure 9. Including these is meaningful to be able to generate more practical and realistic designs. For example, the first mechanism (a) is Euler buckling. To include this type of instability into the formulation the critical buckling stress n_{cr} is added in the constraints of the GSM (Zhao, 2014). Euler buckling has nonlinear behaviour, but the optimization problem is still a LP, because in the constraint the critical stress is used instead of the critical buckling load. The two other types of stabilities require different methods and are addressed in: (Zhao, 2014); (Weldeyesus, et al., 2019); (Weldeyesus, et al., 2020).

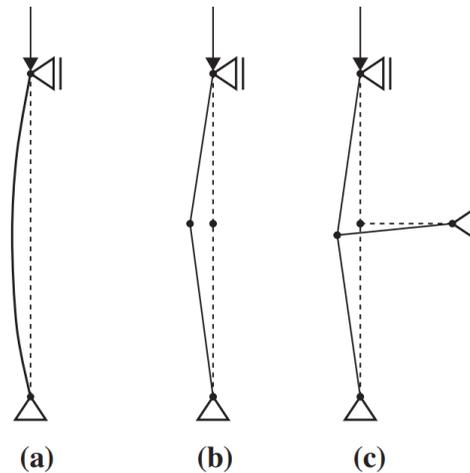


Figure 9. Three types of instability. a) local instability, b) nodal instability and c) global instability (Descamps, et al., 2014).

2.5.4 Geometric optimization

Geometric optimization is a post processing step that can be done after the layout optimization (He., et al., 2018). The nodes are the variables in this optimization step, instead of using the area and internal forces. The nodes can be moved around with the objective to minimize the volume of the truss. The initial location of the nodes becomes less important due to the rationalization of its location in this post processing step. However, the node density is still a factor that plays a role. A high node density will likely result in an improved solution in terms of volume (He, et al., 2015).

2.6 Software for the Ground Structure Method

There is software readily available that can perform the optimization. On the web is the GSM available on www.layopt.com. It can do live animated optimizations, but it is closer to being a cool toy than tool.

Another option is Peregrine. Peregrine is a software package that can be used as a plug-in in Grasshopper, a visual programming environment for the 3D CAD software Rhino. It is a powerful tool that is able to generate least-weight structures for a given set of loads, supports and material properties. The workflow of the program consists of pre-processing, solving and post-processing. In order to start the minimization process in Peregrine the following four items must be set: The design domain, loads, supports and material properties. Additionally, Euler Buckling

can be included as a post-processing step. The solver minimizes the volume of the ground structure. The solver has the possibility to add additional functionality in the objective function, such that the minimization includes:

1. Euler buckling (as a constraint)
2. Self-weight
3. Joint Cost

There are multiple options to post-process the least-weight structure. Two important ones are the ‘Stabilize’ and ‘Geometric Optimization’. In Geometric Optimization the location of the nodes are optimized. This makes the required accuracy of the initial ground structure less important. The stability may be increased by adding members, increasing cross-sectional area or adding supports. The aim of all the other post-processing components is to increase the constructability of the structure, which comes with a trade-off that increases the volume. Therefore, all these components have also a user defined parameter ‘maximum volume increase’ to be able to control how much the volume may increase due to the trade-off that are being made.

Within Grasshopper most plug-ins are developed to minimize on the volume (including Peregrine). It is a closed environment and the constraints and objective can’t be altered in any useful way. Peregrine will therefore not be used.

The final option is Python. A paper about the GSM with a 99-line Python code is available (He, et al., 2019) and also one in MATLAB (Zegard, et al., 2014) and (Zegard, et al., 2015). Multiple MP solvers are available that can be controlled through Python. The biggest disadvantage is that all the additional methods will not be available, which does decrease the usability of the GSM significantly.

Chapter 3 Environmental Cost Indicator

For the optimization the environmental impact for the construction material, the welding and the paintwork needs to be determined. This will be done by determining the Environmental Cost Indicator (ECI) value for each of these products. To be able to make a fair comparison some insight in the Life Cycle Assessment is required.

3.1 Life Cycle Assessment

The environmental impact is assessed through a Life Cycle Assessment (LCA). The methodology for a LCA is formally defined in ISO 14040 and consists of the following four phases:

1. Definition of goal and scope of the LCA
2. Life Cycle Inventory phase
3. Life Cycle Impact Assessment phase
4. Interpretation phase which can consist of reporting, critical review and limitations of the LCA

The first step in a LCA is defining the goal, scope, functional unit and the system boundaries. The Life Cycle Inventory (LCI) phase consists of collecting information about the in- and output of materials and emissions for making a product. This data should be related to the specific life cycle stages, as shown in Figure 4. All this information is summarized in the Life Cycle Impact Assessment (LCIA), and an Environmental Product Declaration (EPD) can be created. The last phase is to interpretate the obtained information and give a recommendation. In this thesis a complete LCA will not be performed, since performing a LCI can take months to collect all the data. What will be done is calculating the ECI values based on data that is provided by third parties, which can be a research paper, LCIA or an EPD.

3.2 Goal and Scope of the ECI values

The system boundaries are already defined in section 1.3. However, because it is such an important step in the LCA procedure, it is reviewed with additional details in the coming subparagraphs.

3.2.1 Goal

The goal is to make a fair comparison between the environmental impact of the welding, paintwork and construction, such that it can be used in the optimization process. To arrive to these results an ECI value will be calculated for each of these products. The values should be representative for the Dutch construction industry.

3.2.2 Scope

The scope is the main load bearing structure of a bridge, excluding the foundations. The truss configuration is optimized and it is assumed that this will have a small impact on the foundation, since the support conditions are kept the same.

Functional unit

The environmental impact is calculated over the whole service life of the bridge, which is 100 years. The functional unit is defined as such that it is practical to use in the Ground Structure Method. Due to the fact that these values are required for the optimization process, the functional unit for the conservation differs from the other two parameters. In the model the construction material and welding are a volume, therefore the environmental impact of these two will be expressed as shadow costs per cubic meters [$\text{€}/\text{m}^3$]. The amount of required conservation is defined as shadow costs per square meter [$\text{€}/\text{m}^2$].

Life span of products

The reference service life (RSL) and the resulting replacement frequency is given in Table 2. For the conservation the Dutch traffic and marine administration Rijkswaterstaat (RWS) requires at least a lifespan of 20 years. However, to extend the RSL of the bridge intermediary maintenance is often performed to repaint some deteriorated parts earlier. In the context of this LCA it is assumed that the whole structure is repainted every 20 years.

Table 2. Lifespan and replacement frequency of the subproducts.

Product	RSL [year]	Amount of replacements
Circular Hollow Section	100	0
Weld	100	0
Conservation	20	4

System boundaries

The system boundaries are defined following the specific life cycle stages, as shown in Figure 4. In this thesis are, for the calculation of the environmental impact, the making of the semi-finished products (A1-3), the welding from the construction phase (A5) and having to repaint as part of the maintenance (B2) are included into the scope. The transport in the construction phase (A4) is not considered. It is assumed that the distribution of the required materials is a fairly constant factor and is therefore independent of the size of the beams used. During the user phase, only regular maintenance is considered (B2). During use, no energy is used by the bridge structure (B1; B6-7). Repair and replacement (B3-5) is not included, because this is something that occurs incidentally and is therefore unpredictable. The type of material used (steel) is constant. Therewith it is assumed that the results from the end-of-life (C1-4) are also constant.

The environmental data given in an EPD are the stages A1-A3, C and D. However, the data quality of those numbers is not equal. For the production process (A1-A3) the local production process is investigated and the data in the report is a result of the process the producer has. Stage D covers the reuse, recycling and recovery of the product its end life, which is harder to determine. It is often seen that for these values a global average is used. Of the three EPDs that are examined, all are all using a global average for this parameter. This can result in non-realistic results where the benefit of stage D is larger than the burden from stage A. Furthermore, the other data sources do not report information about this life stage. Life stage D will therefore not be used in the calculation of the ECI costs.

3.3 Environmental Impact Categories

During the manufacture, use and end-of-life stages of a product the surroundings may experience some negative environmental impacts. These negative impacts can be grouped into categories, which all contribute to the same type of environmental effect. In a LCA such a group is named Environmental Impact Category (EIC). These individual environmental impact categories are also called midpoints. There are lots of different categories. According to the Dutch building regulation ‘Bouwbesluit 2012’, the eleven basic environmental impacts which

must be included in the environmental impact assessment for construction products are listed in Table 3. An EPD requires to assess 7 EIC, which is four less than the Dutch regulations require.

Table 3. Environmental Impact Categories. (Bouwkwaliteit, 2019)

Environmental Impact Category	Unit equivalent	Dutch impact assessment	EPD
Global warming potential (GWP)	kg CO ₂ eq.	X	X
Ozone layer depletion potential (ODP)	kg CFC-11 eq.	X	X
Acidification potential (AP)	kg SO ₂ eq.	X	X
Eutrophication potential (EP)	kg PO ₄ ³⁻	X	X
Photochemical oxidant formation potential (POCP)	kg C ₂ H ₄ eq.	X	X
Abiotic depletion for non-fuel resources (ADPE)	kg Sb eq.	X	X
Abiotic depletion for fossil resources (ADPF)	kg Sb eq.	X	X
Human toxicity potential (HTTP)	kg 1.4 DCB eq.	X	
Fresh water ecotoxicity (FAETP)	kg 1.4 DCB eq.	X	
Marine water ecotoxicity (MAETP)	kg 1.4 DCB eq.	X	
Terrestrial ecotoxicity (TAETP)	kg 1.4 DCB eq.	X	

Variation in EIC unit equivalents

The two most used environmental analysing methods are from the Institute of Environmental Sciences (CML) from the Leiden University and the ReCiPe method. Both do show lots of similarities, however there are some differences that should be addressed. For the photochemical oxidant formation potential (POCP) the unit equivalent for CML is ethene, and for ReCiPe it is NMVOC (non-methane volatile organic compound). The conversion between the two is: 1 kg NMVOC is equal to 0.416 kg ethylene equivalent (Goedkoop, 2000). For EPDs the usage of fossil resources is expressed in Megajoules, and a conversion factor of 4.81E-4 kg Antimony (Sb) eq./MJ may be used (Bouwkwaliteit, 2019).

Fossil resource depletion may also be expressed as kg oil eq. The amount of energy a kg oil contains depends on where the oil is coming from, and it is assumed that it is Brenton oil.

Following the definitions set by the International Energy Agency, the conversion factor is 45 MJ per kg oil equivalent (IEA, 2021).

3.4 Weight factors

Comparing the environmental impact between two products is not straight forward, because of those several impact categories. An effective way to arrive to a weighted sum of these categories is the Environmental Cost Indicator (ECI). The one-value indicator is perfect for the optimization process, because of its clarity and the fact that the current optimization process also uses the one-value objective V . In this case the shadow costs are an estimation of the required costs to make the environmental impact undone (CIE4100). The costs for each EIC can be found in Table 4.

Table 4. Environmental cost for the 11 impact categories (Bouwkwality, 2019).

Impact category	Unit	Weighting Factor (€/ unit)
Global warming potential	kg CO ₂ eq.	0.05
Depletion potential of the ozone layer	kg CFC-11 eq.	30.00
Acidification potential	kg SO ₂ eq.	4.00
Eutrophication potential	kg PO ₄ ³⁻ eq.	9.00
Photochemical oxidant creation	kg C ₂ H ₄ eq.	2.00
Abiotic depletion potential for non-fossil resources	kg Sb eq.	0.16
Abiotic depletion potential for fossil resources	kg Sb eq.	0.16
Human toxicity potential	kg 1.4 DBC eq.	0.09
Fresh water ecotoxicity	kg 1.4 DBC eq.	0.03
Marine water ecotoxicity	kg 1.4 DBC eq.	0.0001
Terrestrial ecotoxicity	kg 1.4 DBC eq.	0.06

3.5 Data collection

The data collected is coming from various sources. In the coming sub-paragraphs for every product the ECI value is given based on the defined functional unit. The amount of reviewed EPDs and papers is limited. The source of all the environmental data can be found in: appendix A – Data sources environmental impact.

3.5.1 Structural Hollow Sections

The bridge is built up from circular hollow sections. In the production process around 90% of the emissions are coming from the production of the hot rolled coil. Specifically, the conversion from iron ore to liquid steel is very energy intensive.

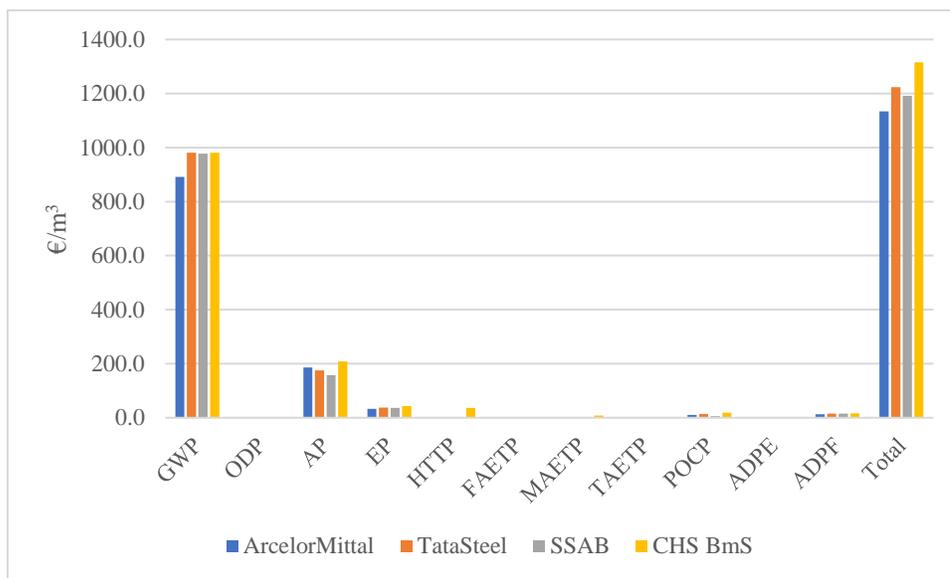


Figure 10. Environmental costs per EIC for the production of structural hollow sections in life stage A1-A3

For the steel cold-rolled hollow sections (SHS) three European producers are selected. All the environmental product declarations from these producers are conform EN 15804. However, the various environmental impact categories for toxicity are not included in the EDP; e.g. HTTP, FAETP, MAETP and TAETP. The Dutch steel association ‘Bouwen met Staal’ did perform a complete LCA review, and their represented values will be used (BmS, 2013). Their results for the structural hollow sections for the environmental impact categories are similar to those represented in the EPDs from the manufacturers, but with additional information about the toxicities. It should be noted that the results are an average comprising several products. The variation between products can lead to a spread of over 20% in one or more environmental categories. Summing all the individual environmental impact categories results in an ECI value of 1314 Euro per cubic meter of SHS.

3.5.2 Welding

Various production methods are available for welding and it is a very energy-intensive process in manufacturing. Welding is not an end-product, such as a CHS, thus an EPD is not readily available. A thorough study was performed (Favi, et al., 2019) to determine the attributional life cycle assessment for various methods of welding. The given data is representative for Italy. This has foremost an impact on the origin of the energy used in the welding process. The life stage A2 is excluded from the assessment.

The assessed welding methods are: gas metal arc welding (GMAW or MIG), shielded metal arc welding (SMAW), gas tungsten arc welding (GTAW or TIG), submerged arc welding (SAW) and flux-cored arc welding (FCAW). The results are based on engineering documentation for the welding and material information from the manufacturer of the filler material. The functional unit is defined as: 1 meter seam welding of a 25-millimeter-thick plate with a V-bevel configuration. The toxicities FAETP, MAETP, and TAETP aren't included in the research. The high human toxicity potential (HTTP) is due to the welding fumes that are released during the process.

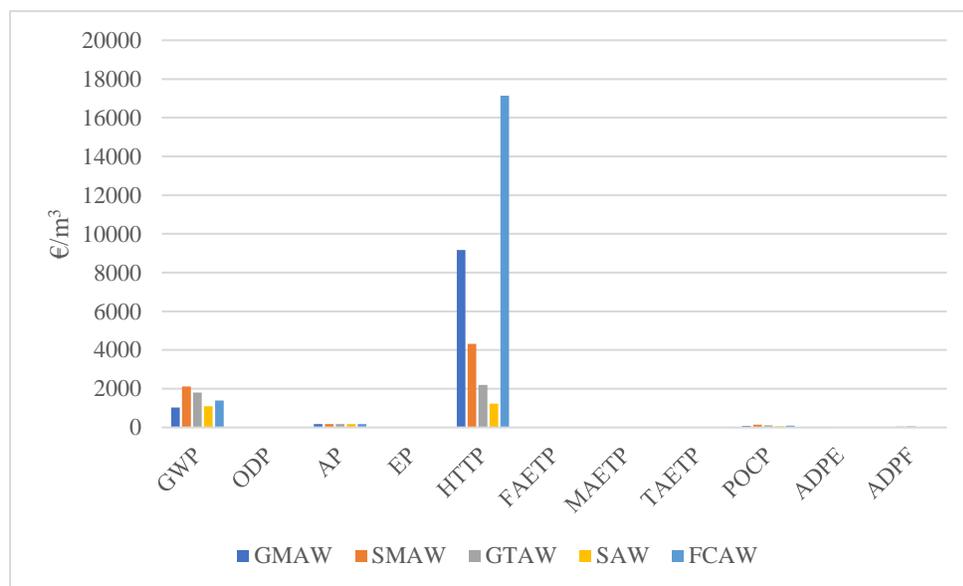


Figure 11. Environmental costs per EIC for the production of a cubic meter of welding in life stage A1 and A3

SAW is the most used welding technique in bridge construction, because of the high deposition rates and deep penetration of the filler material. Welding large and straight welds, such as flanges, plates or girder boxes is main application for SAW. SAW is not suitable for vertical or overhead welding because the flux cannot be kept in place over the arc due to gravity.

This limits the possibility for full penetration welding for a bridge consisting of CHS, because for SAW the work must be moved such that the welder points downwards. Where SAW can't be applied, bridge fabricators typically turn to GMAW for the shorter production welds (FHWA, 2019). Summing all the individual environmental impact categories results in an ECI value of 10496 Euro per cubic meter of gas metal arc welding.

3.5.3 Conservation

There is a vast amount of conservation methods available. A classical finishing is one layer of primer and two top layers, which is used in this thesis. In this case for the top layers an acrylic paint is used. A study (Paiano, et al., 2021) investigated two basic acrylic resin paints from Acrylux and Mastercolor. HTTP, FAETP, MAETP and TAETP are four types of toxicities, and are missing in the data sheet. For the primer the EPD from the brand International of the company Akzonobel is used as input data. The average value of the acrylic paint is 0.09 Euro/m² and for the primer it is 0.18 Euro/m² (Figure 12). To arrive at the total ECI cost this should be multiplied by 5, since it is reapplied 4 times during its lifespan.

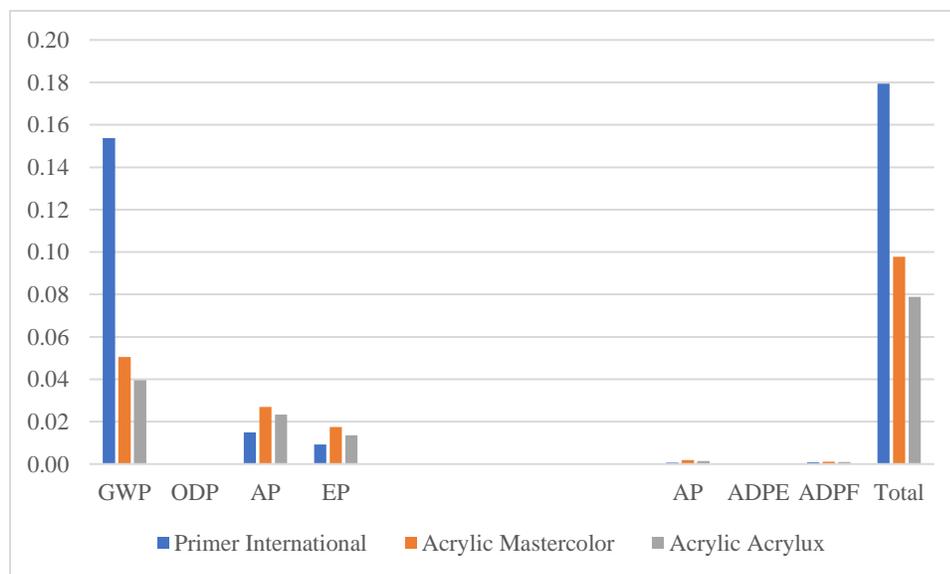


Figure 12. Environmental cost per EIC for painting per square meter for life stages A1-A3

3.6 Conclusion

In short, the Environmental Cost Indicator (ECI) values per product are represented in Table 5. These values do take into account the life span of each product. The functional unit is expressed as $\text{€ECI}/\text{m}^3$ for the CHS and welding and for the conservation it is in $\text{€ECI}/\text{m}^2$, because then the values can be used within the optimization process.

The Dutch building regulation requires information about Environmental Impact Categories (EIC) that are not available in an Environmental Product Declaration. Only the information about various toxicities are missing for the conservation and welding (see 3.5.2 Welding and 3.5.3 Conservation). Further, the information on life stage A2, transportation, is missing for the welding.

The amount of data assessed is limited to 3 EDPs, 2 papers and a research summary. The variation in the EDPs for the circular hollow sections can vary to up to 20 percent, depending on which specific section is selected. The variation for the welding is even higher, namely up to a factor 6x depending on which welding method is chosen. However, this comes with the side note that not every method is applicable for every weld job.

The potential variation of the ECI factor e_v is 20 percent, and the ECI factors e_w and e_c are even higher.

Table 5. ECI values for each parameter for a service life of 100 years.

Parameter	Product	ECI	Unit
e_v	CHS	1310	$\text{€}/\text{m}^3$
e_w	Welding	10500	$\text{€}/\text{m}^3$
e_c	Conservation	1.78	$\text{€}/\text{m}^2$

Chapter 4 Minimization of the environmental impact

The goal is to minimize the environmental cost for the structure by calculating the Environmental Cost Indicator (ECI) for the volume construction material V , the welding volume V_w and the conservation area A_{out} . The optimization method is an expansion of the full-scale Ground Structure Method, as mentioned in 2.4. Thus, the two variables readily available are the area \underline{a} and the internal member force \underline{q} . The total ECI value for the main load bearing structure can be expressed as:

$$ECI = e_V V + e_w V_w + e_c A_{out} \quad (\text{eq. 4.1})$$

With:

V	Volume [m^3]
V_w	Volume welding [m^3]
A_{out}	Surface area of all structural members [m^2]
e_V	ECI value of hot- or cold-rolled steel [$\text{€}/m^3$]
e_w	ECI value for welding [$\text{€}/m^3$]
e_c	ECI value for the paint [$\text{€}/m^2$]

V , V_w and A_{out} are the three variables that are considered in the optimization. The first two can be implemented directly, but the outer area of the structure does have implications on the MP formulation. This will become clear in the coming paragraphs, which elaborate on the optimization method itself. More details on how the pivot values for the environmental impact are determined is explained in the previous chapter, Chapter 3 .

4.1 Volume

The volume is the sum of all the member area times its length ($V = \underline{l}^T \underline{a}$). In this work the cross-sectional area is constant over the member lengths. It is the same definition as (eq. 2.1a).

4.2 Volume welding

It is assumed that the members are connected together with full penetration welds on both sides. Most of the elements do connect with an angle, however this will be neglected. The volume of a weld is approximated by having a constant thickness t_w and an area which is the

same as the member, as is shown in Figure 13. This makes the welding volume: $V_w = 2t_w \sum_i a_i$. This is a quantification of the joint cost method from section 2.5.1.

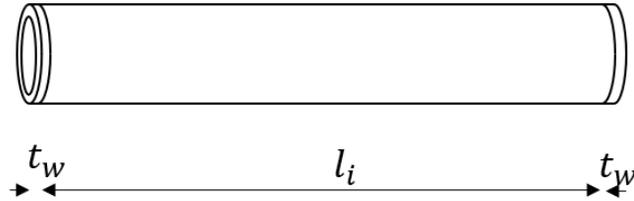


Figure 13. Dimensions of a single element.

4.3 Surface area

The area on the outside of the structure is coated to prevent corrosion. The goal is to establish a relation between the cross-sectional area and the circumference of that area. The shape of the cross section determines the circumference of the outer area. For now, it is assumed that the area and shape of the cross-sections are constant over the length of a member. The outer area of the structure can therefore be defined with $A_{out} = \underline{l}^T \underline{C}$ with \underline{C} being the circumference of the members.

Various cross sections are listed in Table 6. The circumference of the first 4 shapes is expressed in terms of area, and if applicable, also in wall thickness t . The hollow cross sections do have a constant wall thickness, which makes the relation between the area and circumference linear. The goal is to minimize the circumference, thus a logical shape would be a circle.

Solid cross section

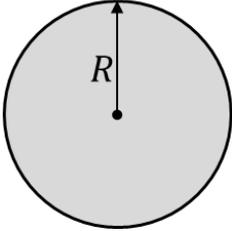
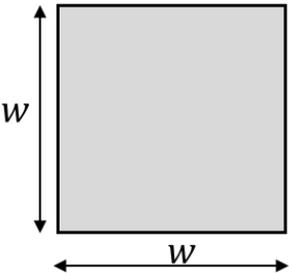
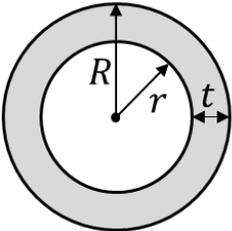
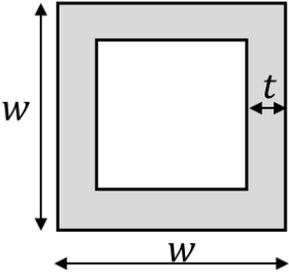
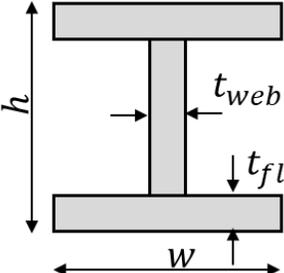
Using a solid cross-section is an option, but using a fractional polynomial in the objective function is a bad idea, because then the objective is not concave anymore. It could be done by introducing a new variable \underline{C} and, if using the circular solid cross-section, it would result in:

$$\min ECI = e_v V + e_w V_w + e_c \underline{l}^T \underline{C}$$

With the additional constraint that:

$$\underline{C}^T \underline{C} = 4\pi a$$

Table 6. Various cross-sectional shapes with its area and circumference.

Cross-sectional shape	Area	Circumference
	$a = \pi R^2$	$C = 2\sqrt{\pi a}$
	$a = w^2$	$C = 4\sqrt{a}$
	$a = \pi R^2 - \pi r^2$	$C = \frac{a}{t} + \pi t$ For $a \geq \pi t^2$
	$a = 4bt - 4t^2$	$C = \frac{a}{t} + 4t$ For $a \geq 4t^2$
	$a = 2wt_{fl} + (h - 2t_{fl})t_{web}$	$C = 2b + 2h + 2(b - t_{web})$

Hollow cross section

The circumference of the hollow cross section has a linear relation with the area, giving the benefit that a non-linear constraint can be omitted. However, there are additional requirements that must be added to the formulation, namely:

1. the area must be zero or bigger than πt^2 , and likewise
2. the circumference must be zero when the area is zero.

Beam-shaped profile

The H-profile its circumference in Table 6 is not expressed in terms of the continuous variable a , as it would not make much sense. In this case it would make more sense to optimize with a catalogue of profile types. However, limiting the area to a pre-assigned set makes it difficult to solve a fully connected ground structure. It can be solved with a mixed-integer linear program. Another approach was investigated by (Jiang, et al., 2017) where the optimization process is alternating between continuous variables and the discrete set of members.

4.4 Main optimization method – hollow cross-section with a constant wall thickness

The profile that is being implemented, is the circular hollow cross-section with a constant wall thickness. Thus, the relation between the area and circumference is: $C = a/t + \pi t$. To calculate the outer area accurately, the bound for the area: $\underline{a} \geq \underline{0}$, must be replaced accordingly with:

$$\underline{a} = \underline{0} \vee \underline{a} \geq \pi t^2$$

Otherwise, the last term in the objective function won't measure the ECI of the paintwork accurately. This can be done with the big- \mathbf{M} method (Fairclough, et al., 2019) in combination with a binary indicator variable s_i . Substituting everything into (eq. 4.1) gives:

$$\min_{a,q,s} ECI = e_v \underline{l}^T \underline{a} + \sum_i^M 2e_w t_w a_i + \sum_i^M e_c l_i \frac{a_i}{t} + e_c l_i \pi t s_i \quad (\text{eq. 4.2a})$$

stated that:

$$\underline{B} \underline{q}^k = \underline{f}^k \quad (\text{eq. 4.2b})$$

$$\underline{q}^k \geq -\sigma^- \underline{a} \quad (\text{eq. 4.2c})$$

$$\underline{q}^k \leq \sigma^+ \underline{a} \quad (\text{eq. 4.2d})$$

$$\underline{a} \geq 0 \quad (\text{eq. 4.2e})$$

$$a_i \leq \mathbf{M}s_i \quad (\text{eq. 4.2f})$$

$$a_i - \pi t^2 \geq -\mathbf{M}(1 - s_i) \quad (\text{eq. 4.2g})$$

With:

i index of member $i = 1, 2, \dots, M$

k index of the load combination

\underline{l}, l_i length vector, length of member i

\underline{a}, a_i area vector, area of member i

s_i binary indicator variable

\underline{B} Equilibrium matrix [$m \times n$]

\underline{q} Internal force vector [m]

\underline{f} External force vector [n]

σ^+, σ^- Maximum and minimum allowable stress

t_w Weld thickness

t Wall thickness of the tube

The constraints from eq. 2.1b-e can be used without making any alterations and are presented as eq. 3.2b-e. The step function is added through eq. 3.2f-g. Adding these two constraints do give supplementary bounds for the area, but do not make eq. 3.2e redundant. The big- \mathbf{M} value must be chosen carefully. A big- \mathbf{M} too small will result in a too constrain problem, and making it too big will result in a numerically unstable system. The big- \mathbf{M} is set as twenty times the maximum force of any point load divided by the maximum allowable stress:

$$\mathbf{M} = 20 \frac{\max \underline{f}}{\sigma}$$

In the case that there are multiple loads in the model this may be too constraint. After an optimization it can be checked whether the big- \mathbf{M} is too constraint or not by comparing it with the member with the biggest area. If that area is equal to or very close to the value of the big- \mathbf{M} , the big- \mathbf{M} was too constraint and should be increased.

This is a Mixed Integer Linear Programming (MILP) problem and can be solved with the correct software.

4.5 Implementation

As mentioned in section 2.6, the approach to implement this optimization is done in Python. The implementation is based on the 99-line script in (He, et al., 2019). The script is able to perform the basic GSM. Unfortunately, any additional functionality, such as flexural buckling, geometric optimization and checks on global stability are not available (see 2.5 and 2.6).

The given script uses the open-source LP solver `cvxpy`, which has not the ability to solve MILP formulations. Therefore, the solver is replaced with IBM ILOG CPLEX 12.9.0 Optimization Studio. The Python module `DOCplex` is used to load the model from Python into the solver.

The script runs on a desktop with Windows 10 installed with 16GB RAM and a AMD FX-8730 eight-core processor that runs at 4.00 GHz. The computational load for solving the MILP is substantial. Thus the default settings of CPLEX are used², except for:

- The default branch and cut method keeps the tree in RAM. This could cause stack overflow error for larger problems, thus this is changed such that it is able to save a part of the tree on the hard drive.
- The default MIP gap tolerance is 10^{-4} , which would be too tight for larger problems. It is changed to 0.1%.
- There is no time limit set in the default settings. It can happen that the optimizer has trouble with reaching the MIP gap tolerance, thus adding a time limit is in that case beneficial.

The CPLEX solver comes with around 100 settings that can be tuned to improve the performance of the solver. What the actual best settings would be for this optimization problem isn't extensively investigated in this thesis. The complete implementation in Python is presented in appendix C – Code.

² All the setting of CPLEX can be found online at:

<https://www.ibm.com/docs/en/icos/12.9.0?topic=cplex-list-parameters>

Chapter 5 Running the optimization method

The optimization method to minimize on the environmental costs, as presented in Chapter 4 is performed multiple times in this chapter. The results of this optimization is compared with the GSM that minimizes on volume, as presented in section 2.4. However, the proposed method has bounds on the area, namely $a = 0 \vee a > \pi t^2$. These bounds are implemented by adding additional constraints to the formulation. Of course, changing the constraints does have an influence on the outcome. Therefore, the following optimizations are performed:

- *min ECI*: minimize the environmental cost through eq. 3.2a-g.
- *min V*: minimize the volume through eq. 2.1a-e supplemented with constraints for the area, which are eq. 3.2f-g.
- *min V unbounded*: minimize the volume through eq. 2.1a-e.

From now on, the methods referred to the names as mentioned here. The *min V unbounded* is just the regular GSM as it is given in 2.4. When calculating the environmental cost for the unbounded method, the area can be smaller than the minimal required area of a cross section, which is πt^2 . If that's the case, then the surface area of member i is then calculated as if it is a solid rod with $A_{out,i} = 2l_i\sqrt{\pi a_i}$.

The material properties are kept constant throughout the optimization runs and are:

- $\sigma^+ = \sigma^- = 235 \text{ N/mm}^2$
- $e_v = 1310 \text{ €/m}^3$
- $e_w = 10500 \text{ €/m}^3$
- $e_c = 1.78 \text{ €/m}^2$

The wall thickness t and weld thickness t_w are 10 mm for the cantilever (section 5.1) and the 2D case (section 5.4) and 8 mm for the 3D case (section 5.5)

The results in the coming paragraphs do contain figures with an optimal structure. The variable area (\underline{a}) can contain numerical errors. The cut-off for the area is set to 10^{-6} , and if it is smaller it is set to zero. The sizes of the bars do vary in thickness, which represents the area of the bar. The thickness of the bars is relative to each other and it does not display the actual physical size. The colors in the figures have the following meaning:

- Blue is compression
- Red is tension
- Grey is in the case when there are multiple load combinations. Then grey is in either compression or tension, depending on the load combination.

5.1 Cantilever beam

The first optimization is performed with a design domain that will result in a cantilever beam. The optimal structure in a continuous domain is given in Figure 5. For the discrete domain, which is the Ground Structure method, the optimal shape for various node densities is given in Table 1. Likewise, this optimization run is executed multiple times where the node density in the design domain is increased at every run.

Setup

The design domain is a 1.0 by 1.0 meter square domain. On the left side are fixed supports placed over the height of the domain, and a vertical load F is placed on the right side in the center (see Figure 14). Two objective functions have a minimum requirement for the area, which is 314.1 mm². Thus, the minimal capacity of each member is 73.8 kN. The size of the point load F is set at 1000 kN, which is 13.5 times larger than the minimal required area.

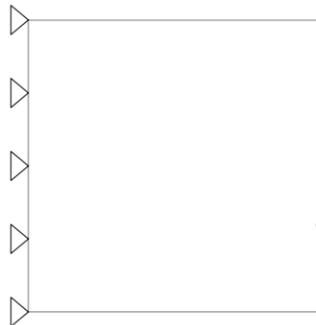


Figure 14. Design domain with a fixed support on the left edge and a load at the left center of the domain

The design domain is populated with nodes to create the ground structure. In the optimization runs the amount of nodes within the domain is varied. The node density is in each iteration doubled. The final optimal structure is dependent on the location of the nodes, and by doubling the density there are nodes added, while keeping the nodes from the previous iteration at the same location. The least amount of nodes possible is a 3x3 mesh, and it is scaled up to a mesh of 17x17 nodes.

Results

The optimal structures are exactly the same when the domain is populated with a nodes in a 3 by 3 grid, as is shown in Figure 15). Because there are only 9 nodes in the design domain, the bandwidth of options, with only 36 potential members, is in that case really small. As the amount of nodes increase, the variations in the optimal structure becomes clear, and are depicted in Figure 16.

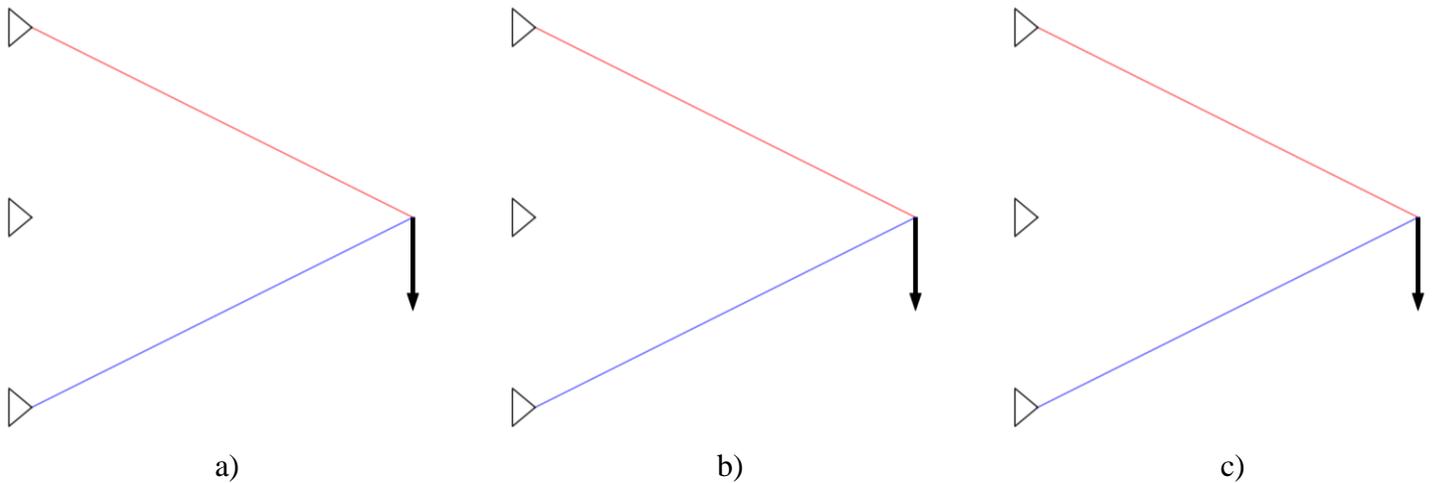


Figure 15. The results for the three optimization methods with 2 nodes per meter: a) min ECI, b) min V and c) min V unbounded.

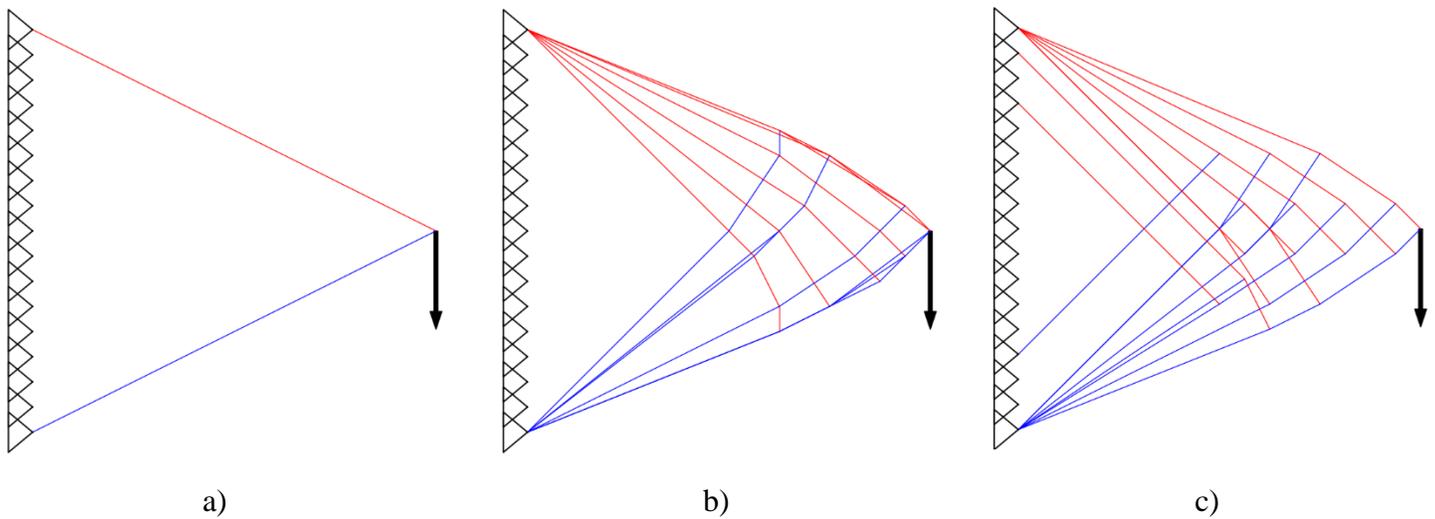


Figure 16. The results of the three optimization methods with 16 nodes per meter: a) min ECI, b) min V and c) min V unbounded.

The optimal structure for a minimal ECI cost doesn't change when the amount of nodes in the domain is increased. The result of the min *ECI* method is plotted in red in Figure 17 and Figure 18, and it is completely flat. The shape of the optimal structure for the two methods that minimize on volume do change. In the run with the most nodes the ECI cost for these least-weight structures is around 37% higher, and the volume is 2.5% lower.

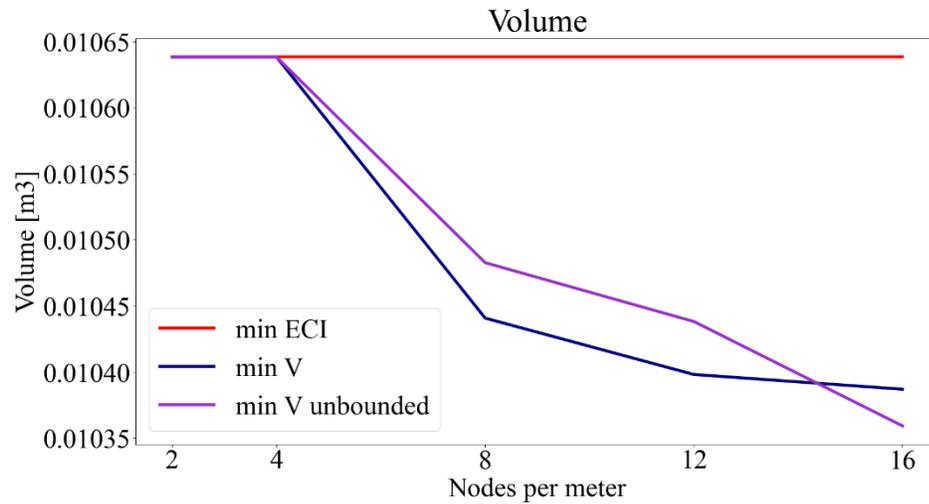


Figure 17. Volume of the optimal structure for the three optimization methods

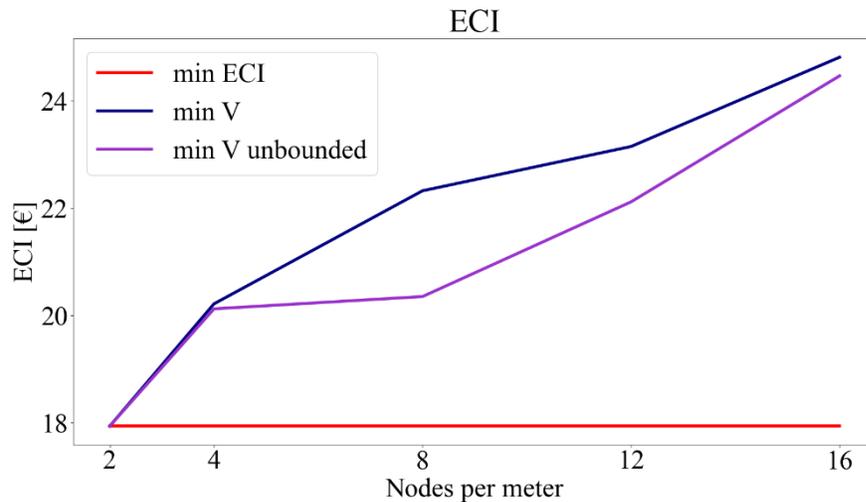


Figure 18. Total ECI cost for the optimal structure for the three optimization methods

5.2 Case study

As a case study a steel truss bicycle bridge is used, as was mentioned in section 1.1. In this paragraph a general overview is given, and in paragraphs 5.4 and 5.5 the results are presented for the 2D and 3D domain.

5.2.1 General information

The bicycle bridge crosses the A2 and connects the residence Maarsen with the rural area Haarzuilens, which also connects to Leidsche Rijn of the municipality Utrecht. Information over this project is provided by Iv-Infra. The key characteristics of this structure will form the input for the minimalization process. A visual render of the bridge is given in Figure 1. The connection has a length of roughly 466 meter and consists of the following parts:

- Eastern run-up of 176 meter,
- Center part over the A2 with a length of 97.5 meter, and
- Western run-up of 179 meter.

The bridge has a walkway of 1.75 meter and a bicycle path of 3.25 meter wide, and it has an overall width of 7.0 meter. The focus in this thesis is on the central section, which consists of a main span of 91.0 meter with an intermediate support. The left half has a span of 39 meter (Figure 19) and the right half has a span of 45.5 meter (Figure 20). The bridge is constructed as a truss construction made of tubes. The tubes in the lower edge of the bridge are filled with concrete, so that in the event of a collision there is also sufficient capacity in the lower edge tubes and in the nodes.

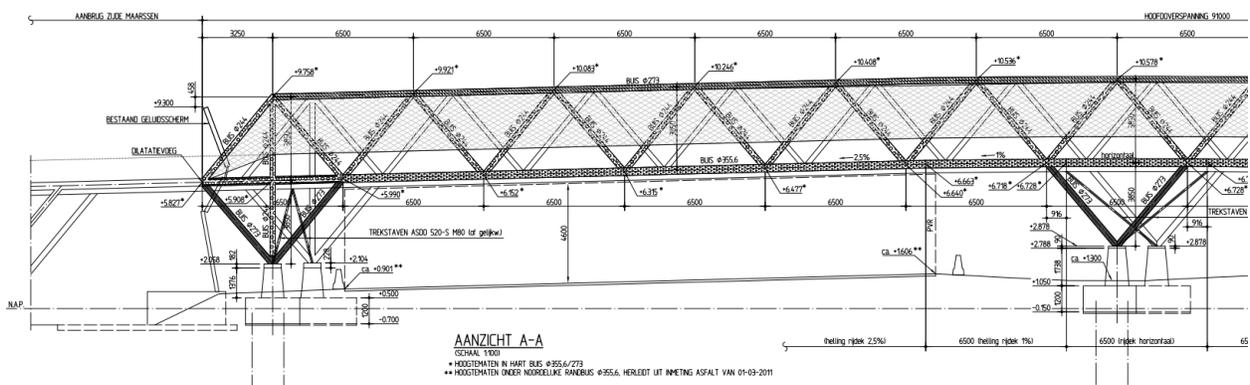


Figure 19. Left span of the bridge.

5.2.3 Design domain

The dimensions of the design domain are in the optimization run simplified. This is mainly done to get practical dimensions for the 3D case. The height from the support to the bridge deck is set to 4.0 meter. From the bridge deck to the upper truss is also 4.0 meter. In the as-built structure the elevation of the bridge deck between the edges and the center is around 900 mm. The foundation in the center is raised likewise. This elevation is not added into the design domain. It is assumed that the supports are able to transfer compressive and tensile loads. Ideally, the design domain would be over the full span of the bridge, as is illustrated in Figure 22.

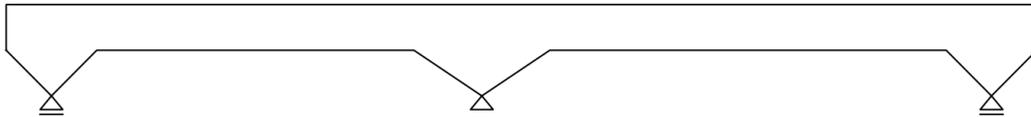


Figure 22. Ideal 2D design domain for the case.

If the nodes are placed on a 1 by 1 grid, then there are 828 nodes in total. With a fully connected ground structure it will result in 111220 potential members. The original optimization method, $\min V$, with no bounds on the area, and with its member adding method is able to solve this easily, but it is too much for the $\min ECI$ optimization. Therefore, only the left span is optimized. Figure 23 shows the design domain that is used for the optimization and it has a length of 32 meters. The nodes are placed on a 2 by 2 meter grid.

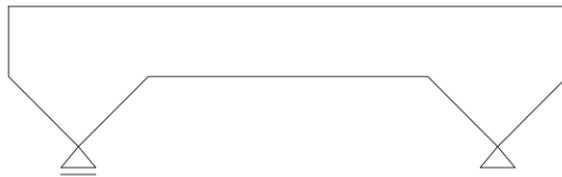


Figure 23. Used 2D design domain.

5.2.4 Loads

The loads from the case study are used in the optimization model, and details of these loads can be found in appendix B – Loads for the case. In the design calculations of Iv-Infra of the bicycle bridge is accounted for 19 load cases, some with multiple sub-load cases, and this leads to 26 load combinations. Implementing this into the optimization model would be impractical due to the immense amount of computational power required to solve such a system.

The optimization model is using only load combinations in the ultimate limit state (ULS). The GSM is a plastic model, and it is not designed to determine elastic displacements. Thus, the

serviceability limit state (SLS) is omitted. Further, it is unclear how to add loads due to temperature differences within the structure, and so these are also omitted.

The loads that are being considered are:

- Permanent load of the bridge deck
- Vertical live loads
- Wind load orthogonal to the structure

Every load combination adds additional constraints to the model, namely via $\underline{B}\underline{q}^k = \underline{f}^k$.

Therefore, the amount of load combinations are reduced even further by combining the permanent, wind and vertical live loads in to one load combination. The load factors and simultaneity factors, Ψ_i , are applied as if there is one load case in the system. The distributed design loads along the top and bottom chords of the bridge are given in Table 7. The units of all the given loads are in kN/m. The difference between the two load combinations is the direction of the wind load. Of course, the 2D optimization is on the xz-plane, thus then only the vertical live load is considered.

Table 7. The load combinations that are considered and are located at the top and bottom chord of the bridge.

Load combination	Load location	$q_{z,k}$		$q_{y,k}$		$q_{z,d}$	$q_{y,d}$
		permanent	variable	permanent	variable		
LC1	Bridgedeck At top of the bridge	16.6	12.5		2.4	45.5	4.0 kN/m
					2.4		4.0 kN/m
LC2	Bridgedeck At top of the bridge	16.6	12.5		-2.4	45.5	-4.0 kN/m
					-2.4		-4.0 kN/m

The loads are defined as a distributed load, however within the model the loads are applied only at nodes. Every point load does represent the connection between the bridge deck and the main load bearing truss structure. How the point loads are placed does have a huge influence on the resulting optimal structure. For example, a straight forward approach would be to apply this load to every single node. This would result in an optimal structure that requires to have members connected to all the nodes in the bottom and top chord of the bridge, which would

be an unpractical structure. Also, so many connections between the bridge deck and the truss structure are not necessarily required.

The current bridge deck connects to the truss structure every 3.25 meters. The ideal approach would be to add the bridge deck to the optimization. A longer span of the bridge deck would increase the required deck height, but it would require less connections to the truss structure at the same time. The latter reduces the weight of the truss structure, and so there is an optimal span for the connection between the bridge deck and truss structure. For now the connections of the bridge deck is set to 4 meters. Thus, the distributed loads should be multiplied by 4 to get the load on each node.

5.3 2D optimization as-built structure

The as-built structure has to comply with various design aspects, such as stability, deflection and strength. Some of these important engineering aspects are dealt with through the extensions given in paragraph 2.5. However, these are not implemented in this thesis, thus the as-built structure will be simplified such that it matches structural integrity of the optimal structures that are presented in the two coming paragraphs. The as-built structure has the shape of a Warren Truss. The design domain is given in Figure 24. The loads at the edges are 273 kN and the three loads in the middle are 364 kN. To make sure that the GSM converges towards a Warren truss the ground structure is reduced, as is shown in Figure 25.

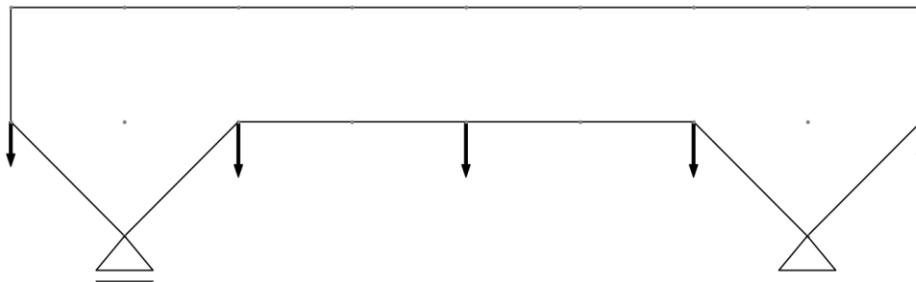


Figure 24. The design domain, loads and supports of the as-built bicycle bridge

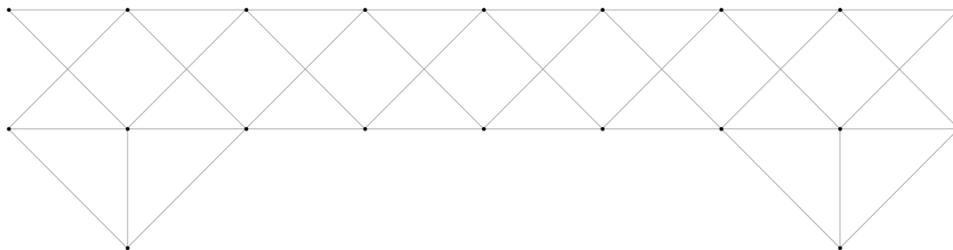


Figure 25. The ground structure for the Warren Truss

The configuration of the members in the optimal structure in Figure 26 is equivalent to that of the as-built structure, which is shown in Figure 19 and Figure 20. The total volume of this optimal structure is 0.21 m^3 and it has a total ECI cost of 297 Euro, when the yield stress is 235 N/mm^2 . The as-built structure does use S355, which would reduce the amount of steel to 0.14 m^3 with an ECI cost of 199 Euro. Applying the same profiles of the as-built structure in this ‘optimal structure’ would result in a total construction volume of 1.42 m^3 .

The most significant load case is the live load on the bridge deck, even though this load case is used in the optimization, the difference of the ‘optimal structure’ is huge compared to the as-built one. Even if the maximum allowable stress would be limited to 50% of the yield, which gives a volume of 0.28 m^3 , the difference is still a factor 5. It clearly showcases that this optimization is only a partial analysis of the complete structural analysis required to make sound structures. Firstly, because the physical model of the implemented GSM is a simplification of the real world, and secondly, because not the same design requirements are used for the optimization. For example, the profile used in the bottom chord of the ‘optimal structure’ has an area of $2,323 \text{ mm}^2$, whereas the as-built structure uses a $\text{Ø}355.6 \times 16$ with an area of $17,070 \text{ mm}^2$. It is relatively big compared to the other members of the as-built structure and it certainly has overcapacity under normal circumstances. However, since the bridge runs over an highway, a truck might collide with the bridge. When that happens all the overcapacity is gone.

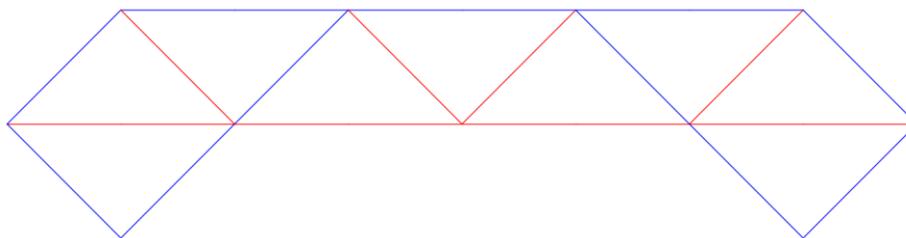


Figure 26. The ‘optimal structure’ for the as-built situation

5.4 2D optimization of the bicycle bridge

The setup and results for the 2D variant are presented in this paragraph. This optimization run does include one load combination, namely the vertical live load. The design domain is one span of the bridge, as is mentioned in section 5.2.3, and is illustrated in Figure 23. The load that is being considered is the live load, as was mentioned in section 5.2.4, and is listed in Table 8. The nodes are placed on a 2 by 2 meter grid, and are connected together with 1218 potential members. The load is applied every 4 meter. which results in the optimization problem given in Figure 27. For this optimization the MIP gap is set to 0.1% with a time limit of 3,600 seconds.

Table 8. The loads used in the 2D case.

Load combination	Load location	$q_{z,k}$ [kN/m]		$q_{z,d}$ [kN/m]	$F_{z,d}$ [kN]
		permanent	variable		
LC1	Bridgedeck	16.6	12.5	45.5	182

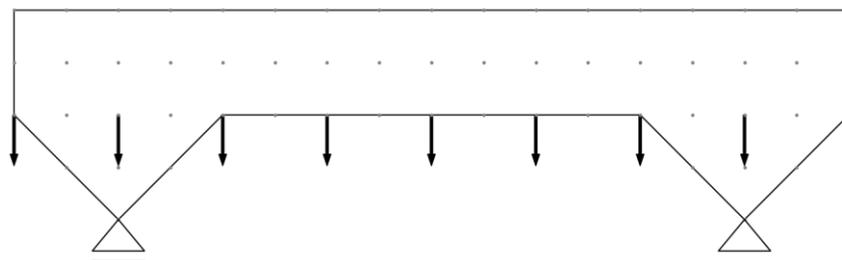


Figure 27. The design domain with 85 nodes and 9 point loads

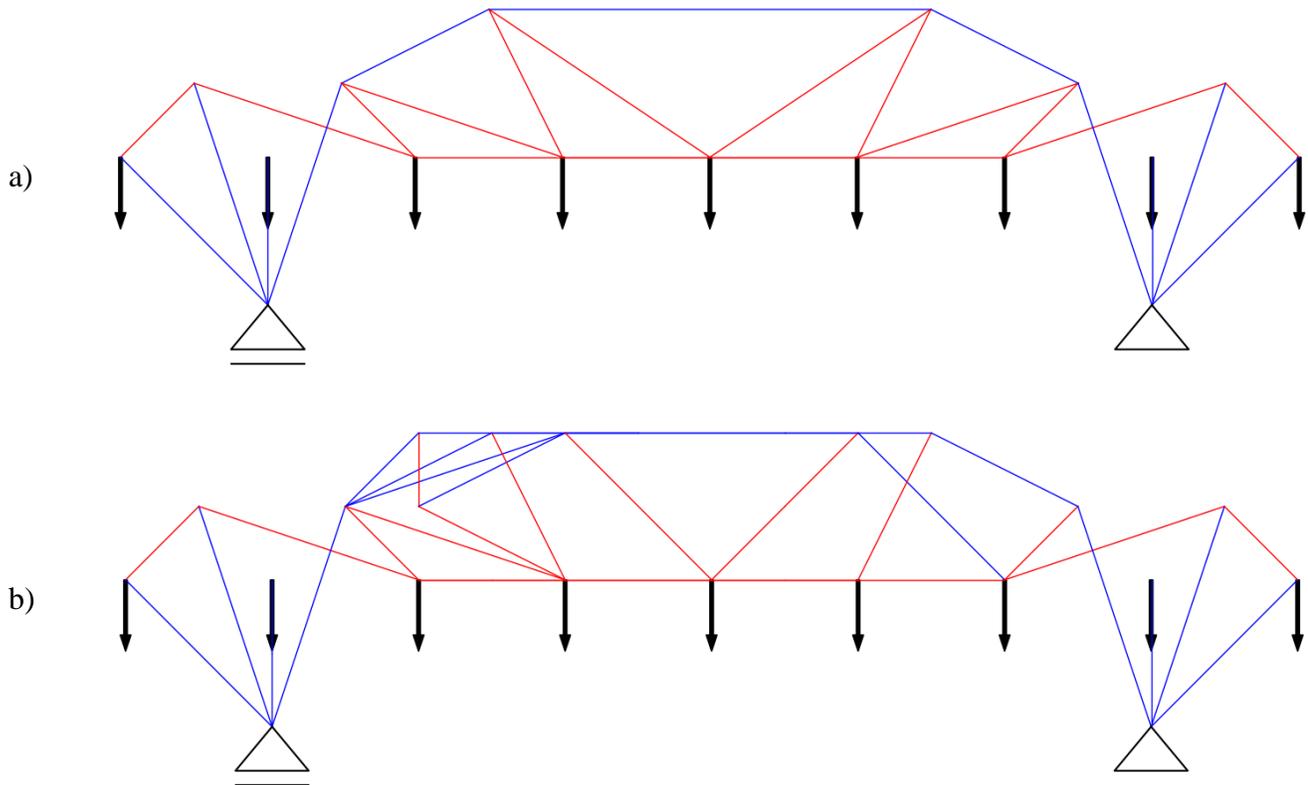
Results

The solver is able to solve this problem successfully, and the results are presented in Table 9. The min ECI optimization reached in an hour an optimality that is maximal 0.82% from the global optimum. The optimizations that minimize on volume outperformed it heavily, since it reached the global optimality within seconds. The difference in the last column of the table is the result of min ECI divided over min V . The environmental cost for the welding is reduced by 30%, due to the fact that there are 40% less members in the optimal structure for min ECI . However, this cost saving is negligible compared to the environmental cost of the construction material, ECI_v , which is in this case 85% of the total ECI cost.

Table 9. Results for the case with the 2D optimization.

Objective	min V unbounded	min V	min ECI	diff.
Volume (m ³)	0.1601	0.1611	0.1611	1.0000
ECI (€)	249	249	246	0.99
ECI _v (€)	210	211	211	1.00
ECI _w (€)	7.4	7.4	5.16	0.70
ECI _c (€)	32.3	31.0	30.1	0.97
Members in optimal structure	44	42	25	0.60
Solve time (s)	0.3	12.5	3600	288
MIP gap	-	0.00%	0.82%	-

The optimal structures are given in Figure 28 and they do show lots of similarities. It is clear that the stability of the structure isn't a criterium in the basic GSM method. Furthermore, from these figures it becomes clear that the min ECI method is limiting the number of members in the optimal structure.



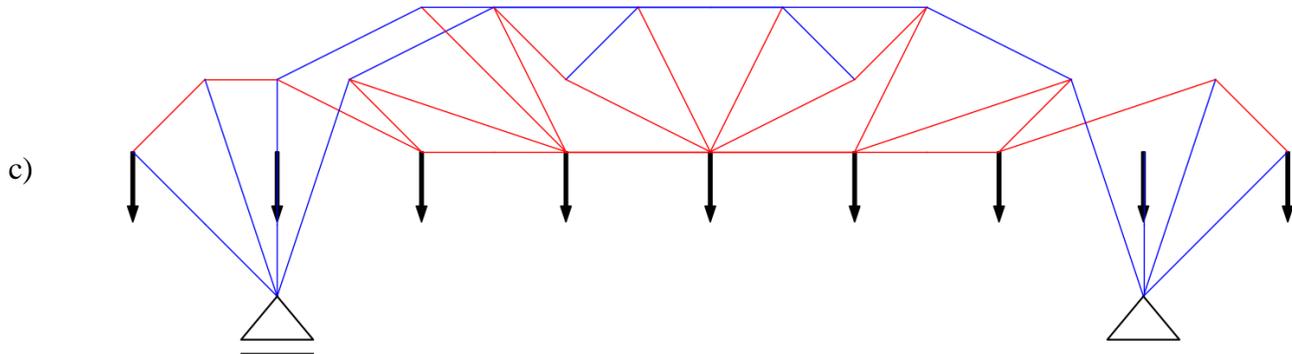


Figure 28. Optimal structures for a) *min ECI*, b) *min V* and c) *min V unbounded*

5.5 3D optimization of the bicycle bridge

The setup and results for the 3D variant is presented in this paragraph. This optimization run does include the vertical static and live load and the horizontal wind load.

Design domain

In a 3D design space the optimal structure for the interaction between the wind load and the vertical live loads can be tested. The 3D design domain will consist of only a quarter span. Of course, this will change the layout of the optimal structure, but the mixed-integer LP is so very computationally expensive that it is unfeasible to optimize over the whole span.

The domain is being built up from multiple polygons, as is shown in Figure 29, and it is the left quarter of the 2D design domain in Figure 22. The x-axis is along the bridge, the y-axis is along the width and the z-axis along the height. The span is cut in half, thus 4 supports are placed at the 4 corners of the cut-off, where only the degree of freedom in the x-direction is set to zero. Since it is modelled as a complete truss structure, the members can only transfer axial loads. Therefore, there are planes added around the support at $x=4.0$ meter at both sides of the bridge.

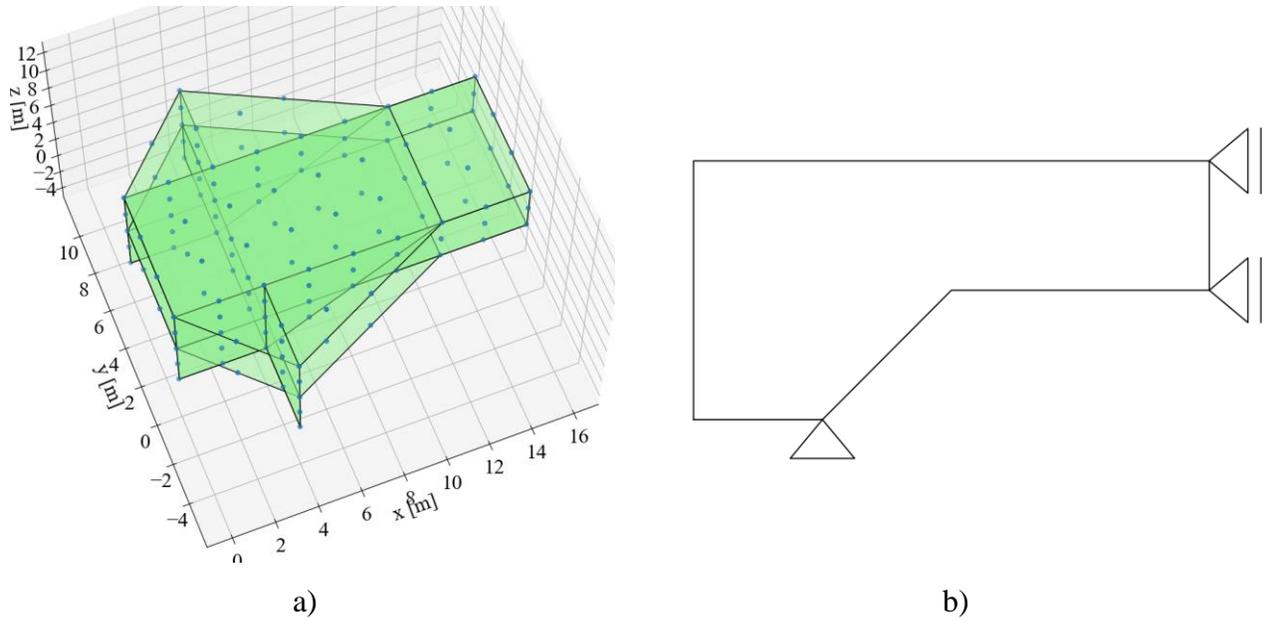


Figure 29. 3D design domain with a) a view from the top and b) the schematic side view

Optimization problem

The nodes are placed 2 meters apart within the design domain, which results in 3122 potential members. There are two load combinations for the optimization, and these are given in Table 7 of section 5.2.3, and are displayed in Figure 30 and Figure 31 respectively. Both load combinations do have at the two bottom chords a vertical load. Depending on the load combination the wind load is either on the chords on the left or right. For this optimization the MIP gap is set to 0.1% with a time limit of 16,000 seconds.

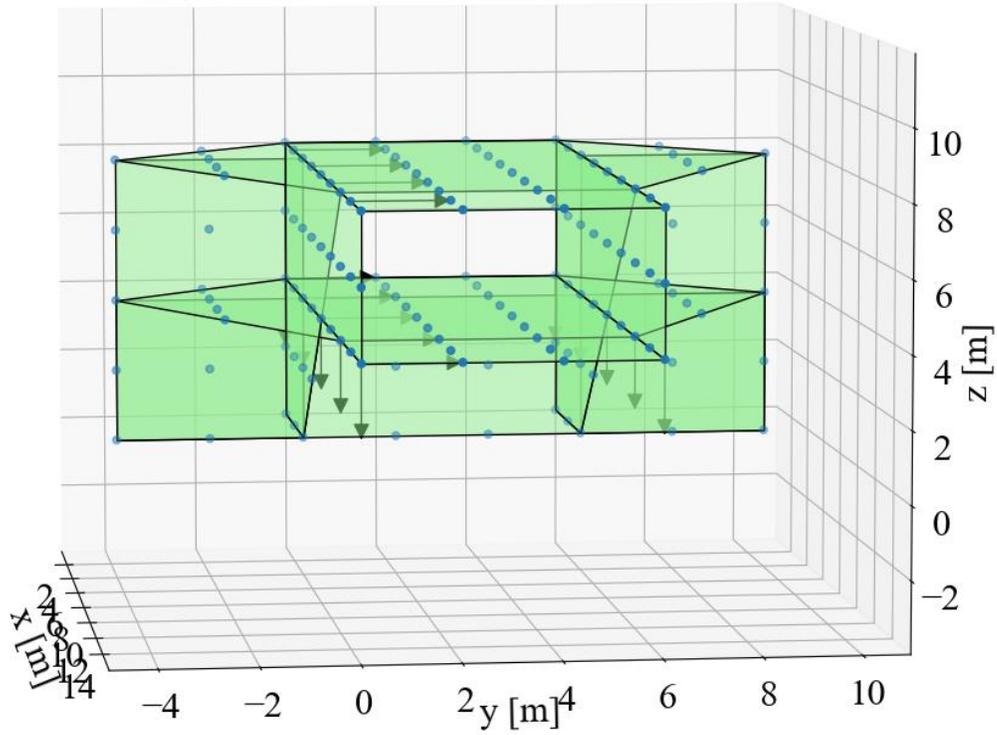


Figure 30. Load combination 1 with at the two bottom chords a vertical load and at the two left chords a horizontal wind load.

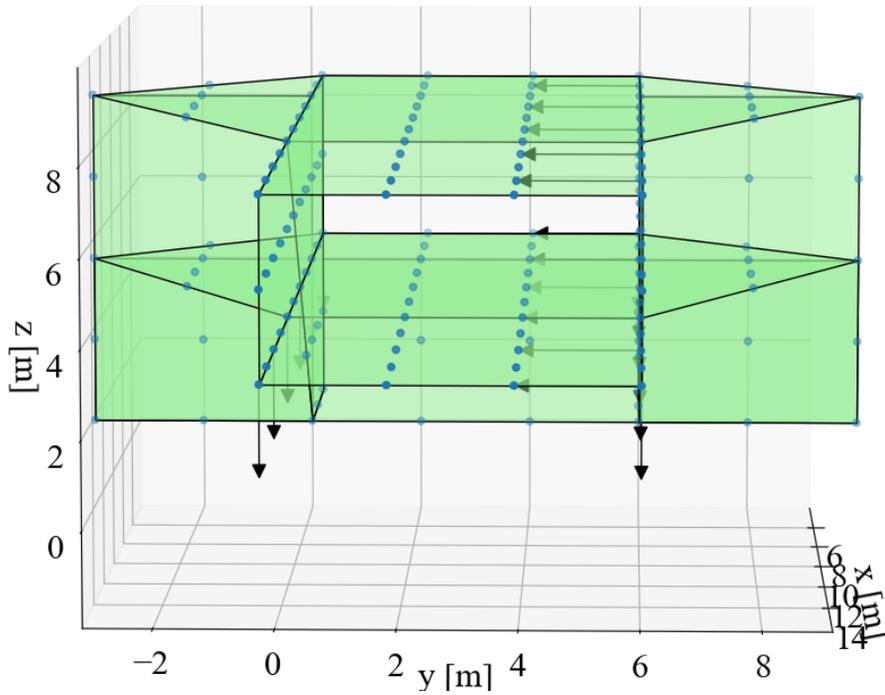


Figure 31. Load combination 2 with at the two bottom chords a vertical load and at the two right chords a horizontal wind load.

Results

The solver is able to solve the optimization problem $\min V$ and $\min ECI$, but in 16,000 seconds the convergence is between 5-14%. The unbounded $\min V$ method is solved successfully within 30 seconds. The total environmental cost for the $\min ECI$ method is 1 percent lower than the $\min V$ method. But the optimality gap is a multiple of this saving, so no meaningful conclusions can be drawn on this particular point.

Table 10. Results for the 3D optimization

Objective	$\min V$ unbounded	$\min V$	$\min ECI$	diff.
Volume (m ³)	0.1579	0.1718	0.1724	1.0036
ECI (€)	268	276	274	0.99
ECI _v (€)	207	225	226	1.00
ECI _w (€)	7.9	8.2	6.51	0.79
ECI _c (€)	52.7	43.1	42.1	0.98
Members in optimal structure	178	116	117	-
Solve time (s)	26.9	16000	16000	-
MIP gap	-	5-10%	13.18%	-

Looking at the optimal structure in Figure 32 it is clear that the GSM will result in unstable members. Interestingly, as is shown in Figure 33, the bracing for the horizontal wind load is placed at both sides in the unbounded $\min V$ method, while with the two other methods the bracing is placed on only one side. In the figure the diagonal bracing is shown for the bridge deck and for the top of the bridge at the same time.

The wall thickness and weld thickness are in this optimization run reduced to 8 mm. The minimal required area gives at least 47 kN capacity in each member, while the wind load is only 16 kN. It seems that in the case when the design load is smaller than the minimal required area, the variation between the $\min V$ and $\min ECI$ method will be minimal.

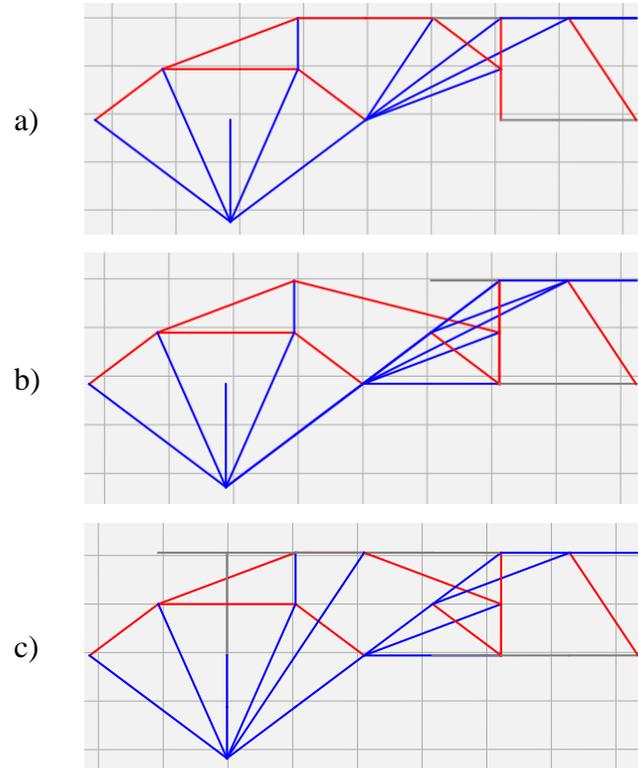


Figure 32. The side view at $y=0$ of the optimal structure for a) min ECI, b) min V and c) min V unbounded.

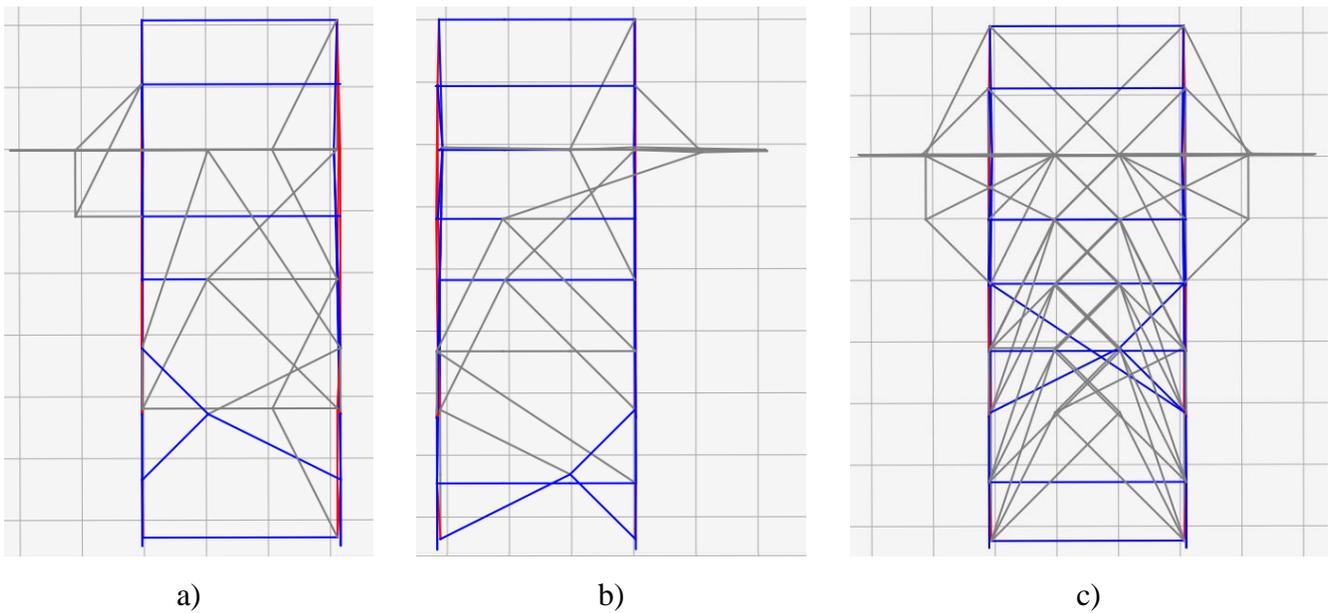


Figure 33. Top view of the optimal structure for a) min ECI, b) min V and c) min V unbounded.

5.6 Summary

The minimization of the environmental impact for a least-environmental-impacting structure is compared to the regular least-weight structure for a fully connected ground structure. To be able to solve the optimization, the relation between the area and the circumference is assumed to be linear supplemented with an additional constraint to have a minimal required area (see paragraph 4.3 and 4.4).

The cantilever beam

The example in paragraph 5.1 does show that, as the number of potential members increases, the difference between the min ECI and min V method becomes clearer. The cantilever in a continuous domain will have an infinite number of members. The discrete domain will approach this more and more as the node density in the domain is increased, as is shown in Table 1. The min V method, with the constraints on the area, will follow this same behaviour if the load is larger than the minimum member capacity, which follows from the minimal required area. However, no matter how much the node density is increased, the min ECI method will always converge to a truss with 2 bars, as is illustrated in Figure 16a.

This cantilever example does have members that do connect directly from the support to the load. In a slender structure, such as a bridge, a direct load path won't be available and the optimal least-environmental-impacting structure will be less obvious.

The bicycle bridge

A case study on a bicycle bridge is performed from paragraph 5.2 to 5.5. The length of the main span over the highway is 91 meters. This size is unfortunately too big to solve for the MILP formulation. Trying to optimize over this domain results in either a stack overflow, numerical errors or an almost indefinitely long run time to get a meaningful convergence. Thus, the design domain is reduced to one span in the 2D optimization, as is illustrated in Figure 23. For the 3D optimization that span is halved again, which results in a domain given in Figure 29. To improve the convergence rate even further a coarse node density is applied where the nodes are placed on a 2 by 2 meter grid. Likewise, the amount of load combinations is reduced to the bare essentials, which is the vertical live load in the 2D case and a rough combination of the vertical live load and the wind load in the 3D case.

With the taken precautions the 2D case is able to converge to 0.82% for the min *ECI* method. The 3D case converges to an optimality gap of 13% after more than 4 hours, which isn't ideal. Especially if it is compared to the unbounded min *V* method, that converges completely in only 26.5 seconds. For the 2D case the least-environmental-impacting truss structure does result in 40% less members, which results in a 1% lower environmental impact. The effect on the total environmental impact is low, because the main contributing factor in this *ECI* calculation is the *ECI* cost for the construction material, which is in this case 86% of the total environmental cost.

The comparison between the of the as-built structure (section 5.3) and the min *ECI* its optimal structure (section 5.4) is presented in Table 11. Both are optimized for a single load combination. Of course, the structural integrity of the as-built structure is better, or rather, the structural integrity of the optimal structure is so bad that it can't be built. Nevertheless, it does show that the min *ECI* method is capable of determining efficient load paths to create very efficient structures, since the required volume is only 78% and the environmental impact is 83% of the as-built structure.

Table 11. Comparing the as-built structure with the min *ECI* its optimal structure.

Objective	as-built	min <i>ECI</i>	diff.
Volume (m3)	0.2076	0.1611	0.7760
<i>ECI</i> (€)	297	246	0.83
<i>ECI_v</i> (€)	272	211	0.78
<i>ECI_w</i> (€)	9.4	5.16	0.55
<i>ECI_c</i> (€)	15.6	30.1	1.92
Members in optimal structure	19	25	1.32
MIP gap	-	0.82%	

Chapter 6 Discussion

The research done in this thesis is discussed in this chapter. It is summarized into keynotes that are split into the minimization method itself, the case study and the Environmental Cost Indicator.

6.1 Environmental Cost Indicator

The minimization is based on the Environmental Cost Indicator (ECI) for the material. The ECI costs for the materials are generalised to uniform units that fit into the optimization model. Four notes on ECI are:

- The minimization of the environmental impact in this thesis is only a partial analysis of the complete environmental cost for constructing a welded truss structure. When other impacting factors are considered, such as the handling of material during the construction phase, then the number of members in the optimal structure will probably be even more limited.
- The environmental cost for the welding is much lower than the labour cost to produce such a joint. Thus, the most cost efficient structure will have even less members.
- The ECI is determined for the construction material, welding volume and the coating. The volume of a weld is really low compared to the volume of construction material. Even though the ECI cost for a cubic meter of welding is a multiple of that of hot rolled steel, the total contribution is between 1 and 15% of the total ECI cost in this particular model. The ECI cost for the conservation is between 10 and 20% of the ECI cost of the construction material.
- The variation of the indicator values is very high. The ECI value for the CHS category can vary up to +/- 20% depending on what specific profile is chosen. The steel grades S235 up to S420 do fall within the same category in the Environmental Product Declaration. The amount of data assessed for the ECI cost of the welding and coating is limited, but it is expected that the spread is as high or even higher than for the CHS profiles. Furthermore, the weighted monetary cost for the Environmental Impact Categories, which is the basis for determining the ECI cost, does also contain uncertainties of what the actual monetary value should be.
- The field of environmental studies is still developing. The Environmental Product Declarations (EPD) and various papers may use different units for the Environmental Impact Categories (EIC). Further, the Dutch building regulations requires information on 11 EIC,

while an EDP only needs to provide information on 7 of them. Harmonizing these two points further would be beneficial for all the users.

6.2 Case study

Chapter 5 starts with a well-known design domain for a cantilever. The case is optimized twice, namely with a 2D and 3D domain. The following notes are made:

- There aren't any design requirements set in this thesis, because the generated structures are not yet sophisticated enough. Since the optimization is done in Python it doesn't have access to any pre- or post-processing options, thus it will generate structures with members that are:
 - crossing each other,
 - overlapping each other,
 - unstable in plane,
 - unstable out of plane,
 - unstable under compressive loading, and
 - dependent on the location of the nodes during the initialization.

These issues can be prevented by adding additional constraints to the objective function or through analysing the optimal structure afterwards through post-processing steps. Some useful methods to solve these issues are described in section 2.5.

- Adding more than one load combination is crucial to determine an accurate shape for the optimal structure. The loads do give the shape to the structure. However, extra load combinations do generate extra equality constraints to the optimization problem. This will result in a balancing act between accuracy and computability. The MILP formulation has trouble to solve problems over 10,000 variables, and so the accuracy will be limited for larger problems.
- The design case has load cases due to wind loading and due to temperature differences within the structure. It is unclear how the stress due to temperature difference could be implemented within the main optimization. Wind is a transient load, so it is only there where the construction is. In the case study it is modelled as a distributed load over the surface area. As with a discrete FEM, the load can only be applied at the nodes, but the difficulty with the GSM is that the structure isn't there yet. It is generated based on the loads in the nodes. For the 3D case the wind load is placed at the nodes such that it will form a Warren Truss. This is

the same type of truss configuration as the as-built structure. Another option would be to place the wind loads right above the vertical loads of the bridge deck, which might push the optimization method into a truss structure that is more comparable to a Pratt Truss or K Truss.

6.3 Optimization method

The aim is to minimize the environmental impact of the construction material, welding and coating of the truss structure. The general approach for the minimization is formulated as: $\min ECI = e_v V + e_w V_w + e_c A_{out}$. The optimization method chosen in section 4.4 is applied to a fully connected ground structure. The members of the structure are schematized as circular hollow sections (CHS) with a constant wall thickness. The radius of the CHS is variable and depends on the area of the cross-section.

The coating is applied on the outside of the structure, and generally speaking, the relation between the outer surface and the volume of a structure is non-linear. This schematization of the CHS is made based on the computability and usability of the optimization problem. This given assumption results in a minimal required area, because if the area would be too small the linearization would not make sense.

As a result, the minimization formulation becomes a Mixed-Integer Linear Programming problem, while the original Ground Structure Method is a Linear Programming problem. From the results in the 2D and 3D case it becomes clear that this MILP is more complex to solve. Furthermore, the MILP formulation has only a weak dual, while a strong dual is required for the member adding method. Therefore, it is questionable whether a mixed-integer linear programming problem is usable for a fully connected ground structure. In the two coming subparagraphs alternative approaches are discussed which may overcome this issue.

Reduce size of the ground structure

In the proposed method the fully connected ground structure is used. Using the maximum connectivity may not be the best approach for an optimization method that is not a LP, because it is computationally harder to solve and the member adding method cannot be applied.

A logical step forward would be to reduce the size of the ground structure. The total number of members do scale quadratically with the number of nodes, and so this reduction can reduce the computational cost significantly. However, this comes with the downside that the

shape of the optimal structure is more restricted. In some cases this can be justified. For example, the research of (Ranalli, et al., 2018) and (Havelia, 2016) both do use a ground structure with level 1 connectivity. Because the number of members are so limited, the optimization is able to minimize the total installed cost of steel frame structures while also performing checks on stiffness and connection types between members. Also, because of the connectivity level 1, the issue of overlapping and crossing members is vanished.

Apply the joint-cost method

The general minimization problem $\min ECI$ could also be approached differently. It is clear that the basic GSM as formulated in 2.4 has two major advantages over the proposed minimization. Namely, it is an LP, which is easier to solve, and secondly, the member adding method can be applied. As mentioned in paragraph 2.5.1, the joint cost method is adding a cost to each member length, while keeping the optimization problem a LP. The given objective function in (eq. 4.2a) can be approximated through a relaxation with the joint cost method, as is explained in APPENDIX D – Joint-cost method. The LP is:

$$\min_{a,q} \sum_i^M (l_i + j)a_i$$

With:

$$j = \frac{2e_w t_w}{e_v + e_c/t}$$

This j is the optimal joint cost for this particular model. In this case the outer area of the structure is not calculated accurately anymore. Nevertheless, this approximated quantification of the environmental cost of the welding, construction material and the conservation will result in a LP. The member adding method can be applied too, so larger problems can be solved.

Using this optimization method with the optimal joint cost will result in a good approximation. In the case of the cantilever it will result in the same structure as the $\min ECI$ method, as is demonstrated in APPENDIX D.

Chapter 7 Conclusion

The aim of this thesis is to generate structures with the Ground Structure Method that are least-environmental-impacting instead of being least-weight. The two main research questions are answered. The chapter ends with a commentary on the usage of the Environmental Cost Indicator and a recommendation for future research.

How to minimize the ECI costs for the construction material, welding and conservation through the Ground Structure Method?

The ECI costs for construction material, welding and conservation can be generalized as such that it fits in the optimization model. The general idea of the optimization, as it is formulated in (eq. 4.1), is quite straight forward, but implementing less so. It is necessary to define a profile type, such that the surface area of the structure can be calculated. A circular hollow cross-section is chosen for the implementation. Another profile type could be implemented by changing the relation between the circumference and the area.

In paragraph 4.4 the implementation of this profile is done, and it results in a Mixed-Integer Linear Programming problem. The method is capable of making the least-environmental impacting truss structure. However, the MILP formulation starts to have issues to converge when more than 5,000 potential members are in the system. The LP formulation, combined with the member adding method, is capable of solving problems with over 1,000,000 potential members.

Two options are given in paragraph 6.3, which can help to solve design domains with more nodes. Either reduce the size of the ground structure or apply the joint cost method.

Has the minimization on the environmental impact with the Ground Structure Method an significant influence on the generated truss structures, compared to the usual GSM with weight minimization?

It can have a significant impact on the shape of the optimal structure. Whether it will make a big difference or not does depend on the configuration of the design domain and the node density. In the example of the cantilever the least-environmental-impacting truss structure does

have a lower ECI between 0 and 37% depending on the number of nodes in the domain, while the extra weight is at most only 2.5%. The number of nodes in the case study with the bicycle bridge is limited over the height of the structure, and so the various possible optimal structures is also more limited. As a result, the environmental cost saving is limited in the case study.

Finally, a conclusion for the usage of the Environmental Cost Indicator in the optimization method. The optimization requires ECI costs in units that are heavily generalized quantities, since they are based on the ECI cost of a cubic or square meter. Thus it can be expected that these unit costs do have a high variation of environmental cost. The min *ECI* method is capable of generating a least-environmental-impacting truss structure, but it should kept in mind that the decisions made in the final design can have a bigger impact on the final ECI cost of the structure. For example, if the strength of the members are critical in the design, then increasing the steel grade would result in a reduction of the ECI cost. Or in other words, using steel grade S420 instead of S235 would result in a 70% reduction of the ECI cost. This reduction is bigger than the potential variation that is given in the EDP, which is 20%, thus it would be a guaranteed decrease.

Recommendation for further research

In the case study the main load bearing truss structure is optimized. The bridge deck is connected to the truss at every 3.25 meter. For future research it could be interesting to add the bridge deck to the design domain. A longer span of the bridge deck would increase the required deck height, but it would require less connections to the truss structure at the same time. The latter reduces the weight of the truss structure, and so there is an optimal span for the connection between the bridge deck and truss structure. This would require to combine the truss optimization with a grillage optimization (Bolbotowski, et al., 2018) of the bridge deck.

Chapter 8 References

- Baandrup M. [et al.]** Closing the gap towards super-long suspension bridges using computational morphogenesis [Journal] // Nature communications. - 2020.
- BmS** BmS mrpi-blad 2013 Light Construction engels [Online]. - Bouwen met Staal, 2013. - 2021. - <http://www.dutchhall.nl/wp-content/uploads/2012/06/BmS-mrpi-blad-2013-Light-Construction-engels.1.pdf>.
- Bolbotowski K., He L. and Gilbert M.** Design of optimum grillages using layout optimization [Journal]. - [s.l.] : Structural and Multidisciplinary Optimization, 2018. - 3 : Vol. 58.
- Bouwkwaliteit Stichting** Bepalingsmethode Milieuprestatie Gebouw en GWW-werken [Online] // <https://milieudatabase.nl/wp-content/uploads/2019/05/SBK-Bepalingsmethode-versie-3.0-1-januari-2019.pdf>. - Stichting Bouwkwaliteit, 2019. - 2021.
- Boyd S.** Convex Optimization [Book]. - New York : Cambridge University Press, 2004.
- Coenders J.L.** NetworkedDesign, next generation infrastructure for computational design [Report]. - Delft : TU Delft, 2011.
- Descamps B. and Coelho R.F.** The nominal force method for truss geometry and topology optimization incorporating stability considerations [Journal]. - [s.l.] : International Journal of Solids and Structures, 2014. - 13 : Vol. 51.
- Fairclough H. and Gilbert M.** Layout optimization of simplified trusses using mixed integer programming with runtime generation of constraints [Journal]. - [s.l.] : Springer, 2019. - Vol. Structural and Multidisciplinary Optimization.
- Favi C., Campi F. and Germani M.** Comparative life cycle assessment of metal arc welding technologies by using engineering design documentations [Journal]. - [s.l.] : Springer Verlag, 2019. - 24 : Vol. The International Journal of Life Cycle Assessment.
- FHWA** Bridge Welding Reference Manual [Book]. - [s.l.] : Federal Highway Administration, 2019.
- Gilbert M. and Tyas A.** Layout optimization of large-scale pin-jointed frames [Journal]. - [s.l.] : Engineering Computations, 2003. - Vol. 20.
- Goedkoop M.** The Eco-indicator 99 [Report]. - Amersfoort : Pre-consultants, 2000.

- Havelia P.** A ground structure method to optimize topology and sizing of steel frame structures to minimize material, fabrication and erection cost [Report]. - Stanford : Stanford University, 2016.
- He L. and Gilbert M.** Rationalization of trusses generated via layout optimization [Journal]. - [s.l.] : Structural and Multidisciplinary Optimization, 2015. - 677-694 : Vol. 52.
- He L., Gilbert M. and Song X.** A Python script for adaptive layout optimization of trusses [Journal] // Structural and Multidisciplinary Optimization . - 2019. - pp. 835-847.
- He. L. [et al.]** Conceptual design of AM components using layout optimization and geometry optimization [Journal]. - [s.l.] : Elsevier, 2018. - 78 : Vol. Computers and Mathematics with Applications.
- IEA** Conversion factor [Online] // <https://www.iea.org/reports/unit-converter-and-glossary>. - International Energy Agency, 2021. - 2021. - <https://www.iea.org/reports/unit-converter-and-glossary>.
- Jiang C. [et al.]** Design and Volume Optimization of Space Structures [Journal]. - [s.l.] : ACM Transactions and Graphics, 2017. - 4 : Vol. 36.
- Lagaros N.D.** The environmental and economic impact of structural optimization [Journal]. - [s.l.] : Structural and Multidisciplinary Optimization, 2018. - 58.
- Michell A.G.M.** The limits of Economy of Material in Frame-structures [Journal]. - [s.l.] : The Philosophy Magazine , 1904. - 47 : Vol. 8.
- Mozumder C.** Topometry Optimization of Sheet Metal Structures for Crashworthiness Design Using Hybrid Cellular Automata [Report]. - Notre Dame, Indiana : University of Notre Dame , 2010.
- Paiano A. [et al.]** Sustainable options for paints through a life cycle assessment method [Journal]. - [s.l.] : Journal of Cleaner Production, 2021. - Vol. 295.
- Parkes E.W.** Joints in optimum frameworks [Journal]. - [s.l.] : International Journal of Solids and Structures, 1975. - 1017-1022 : Vol. 11.
- Pritchard T.J., Gilbert M. and Tyas A.** Plastic Layout Optimization of Large-Scale Frameworks Subject to Multiple Load Cases, Member Self-Weight and with Joint Leght Penalties [Conference] // 6th World Congresses of Structural and Multidisciplinary Optimization. - Rio de Janero : [s.n.], 2005.

- Ranalli F., Flager F. and Fischer M.** A Ground Structure Method to minimize the total installed cost of steel frame structures [Journal]. - [s.l.] : International Journal of Civil and Environmental engineering, 2018. - 2 : Vol. 12.
- Rozvany G.I.N.** Shape and layout optimization of structural systems and optimality criteria methods [Book]. - New York : Springer-Verlag Wien, 1992. - pp. 1-5.
- Weldeyesus A.G. [et al.]** Adaptive solution of truss layout optimization problems with global stability constraints [Journal]. - [s.l.] : Structural and Multidisciplinary Optimization, 2019. - 2093-2111 : Vol. 60.
- Weldeyesus A.G. [et al.]** Truss geometry and topology optimization with global stability constraints [Journal]. - [s.l.] : Structural and Multidisciplinary Optimization, 2020. - 1721-1737 : Vol. 62.
- Zegard T. and Paulino G.H.** GRAND — Ground structure based topology optimization for arbitrary 2D domains using MATLAB [Journal]. - [s.l.] : Journal of Structural and Multidisciplinary Optimization, 2014. - 861-882 : Vol. 50.
- Zegard T. and Paulino G.H.** GRAND3 — Ground structure based topology optimization for arbitrary 3D domains using MATLAB [Journal]. - [s.l.] : Journal of Structural and Multidisciplinary Optimization, 2015. - 1161-1184 : Vol. 52.
- Zhao T.** An implementation of the ground structure method considering buckling and nodal instabilities [Report]. - Urbana : University of Illinois, 2014.

APPENDIX

A – Data sources environmental impact

Data type	Owner / written by	Year of publication	Assessment of	Representative for production in:
EPD	ArcelorMittal	2020	Structural Hollow Section	- Lexy (FR), - Iasi (RO), - Krakow (PL), - Karvina (CZ), - Rettel (FR), - Roman (RO).
EPD	TataSteel	2017	Structural Hollow Section	- Port Talbot (UK) - Corby (UK) - Hartlepool (UK) - Maastricht (NL) - Zwijndrecht (NL) - IJmuiden (NL)
EPD	SSAB	2016	Structural Hollow Section	- Pulkkila (FI) - Hämeenlinna (FI) - Lappohja (FI) - Toijala (FI) - Oulainen (FI) - Virsbo (SW)
Summary	Bouwen met Staal	2013	Structural Hollow Section	Netherlands
Paper published in 'The International Journal of LCA'	Claudio Favi, Federico Campi, Michele Germani	2019	LCA of metal arc welding	Italy

Paper published in 'Journal of Cleaner Production'	Annarita Paiano, Teodoro Gallucci, Andrea Pontrandolfo, Giovanni Lagioia, Paolo Piccinno, Amedeo Lacalamita	2021	LCA of acrylic paint	Italy
EPD	Akzonobel	2020	International Primer	Worldwide

B – Loads for the case

In this appendix the load bearing system of the as-built structure is described. The loads on the structure are calculated and the load combinations are described .

B1 Load bearing system

The load transfer of the bridge in the as-built model runs from the deck, via cross beams and consoles to the steel construction, as is shown in Figure 34. The steel truss structure transfers the load to the foundation. The truss structure is made from elements with steel grade S355, and the diagonal crossings at the supports are pretensioned steel bars made of S520.

For each support a concrete foundation has been made, which is founded on prestressed concrete piles. The center supports are translation-fixed in all directions. To prevent the bridge from becoming "locked up", the end supports are translation-free in the longitudinal direction of the bridge. The foundations of the end supports cannot bear tensile forces. The stability perpendicular to the main direction of the bridge is ensured by applying a cross bracing of tension bars at each pair of support points, as depicted in Figure 34.

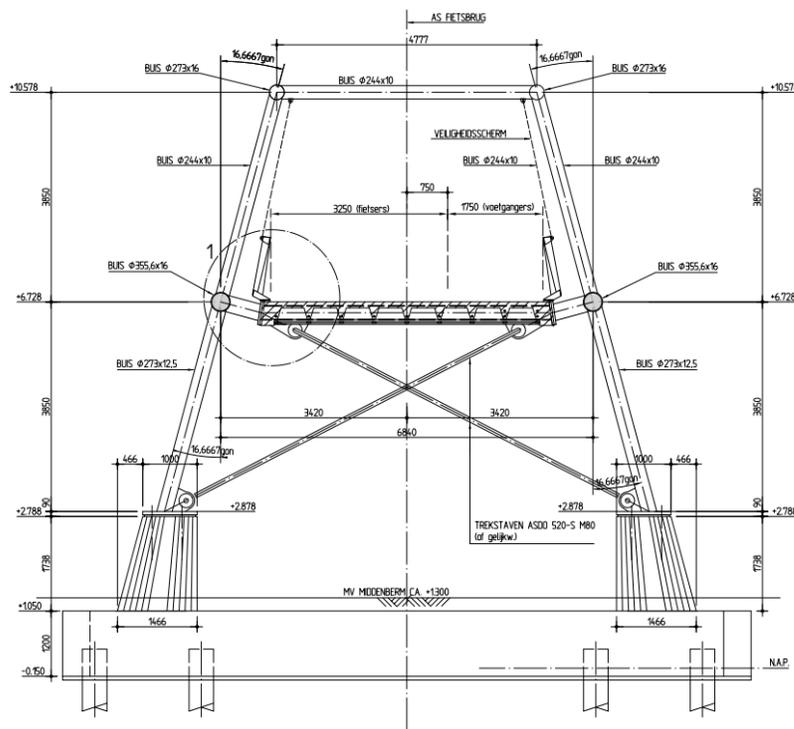


Figure 34. Cross section of the bridge at the center support.

Interestingly, the bridge deck is connected to the bottom rim tubes with a constant distance of 3.25 meters between each console. Thus in the structure the bottom rim tubes do have a dual purpose, namely i) it is a part of the truss structure, ii) act as a beam to transfer the load from the console to the nodes of the truss.

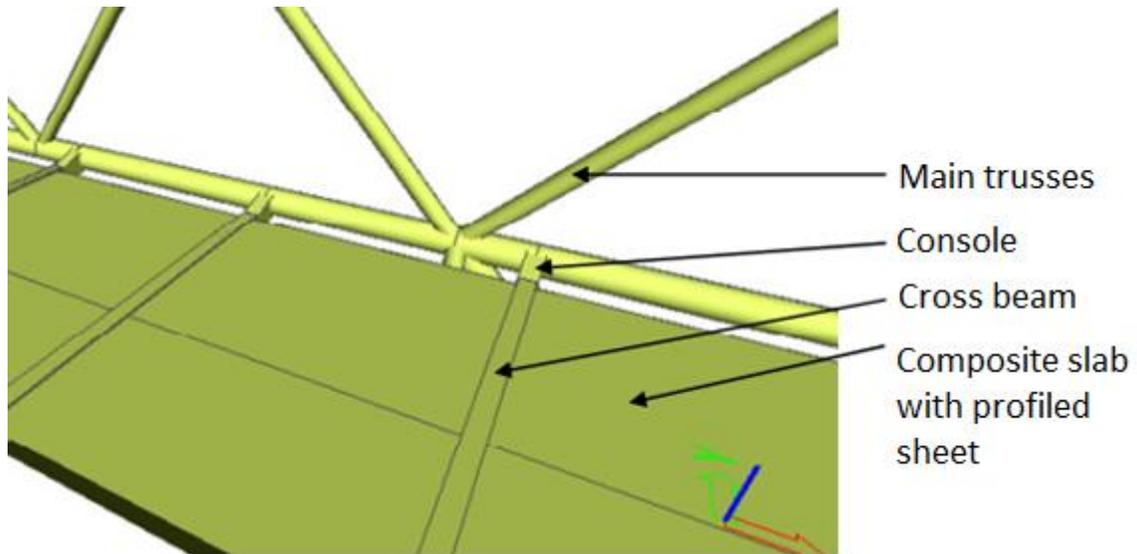


Figure 35. Load transfer to main load-bearing trusses.

B2 Loads

In the design calculations are 19 load cases considered. Not all the elements of the structure can bear a combination of tensile and compressive forces, which leads to adding non-linear load combinations. In total 26 load combinations are used.

B2.1 Load cases

The bridge has on both sides of the deck a truss capable of transferring the loads to the supports, thus where applicable the load is divided between the two. The load cases due to temperature differences in the structure and wind loads not orthogonal to the structure are omitted in this paragraph. Also, the bridge is crossing a highway, causing a risk that the pillars of the structure can be damaged due to a collision. This analysis is also not written out in this paragraph.

B2.1.1 Load case 1: self-weight

The bridge deck is a composite slab made of a profiled sheet topped off with a layer of reinforced concrete and with a width of 5.5 meter its cross-section has an area of approximately

1.00 m². This makes the slab weight 4.6 kN/m². Every 3.25 meter a crossbeam (280ASB74) is placed in the deck with a weight of around 7.5 kN.

The total weight of the deck per meter is $G_k = 13.8 \text{ kN/m}$. The self-weight of the truss structure is not considered in the calculations.

B2.1.2 Load case 2: resting load

Additional permanent items that are not part of the main load bearing system are represented in Table 12.

Table 12. Resting load

Item	Load [kN/m ²]	Load [kN/m]
Wear layer	0.16	0.48
Railing	-	0.50
Safety net	-	1.40
Ceiling cladding	0.14	0.42
Total		2.80

B2.1.3 Load case 3: temporary mobile loads vertical

The bridge has two spans of 45.5 and 39.0 meter with 4 load cases. The goal of splitting up the temporary loads has to do with combining these load cases, because by omitting a part of the temporary load can lead to higher forces at some other places within the structure.



Figure 36. Schematic representation of the temporary load cases

According to NEN-EN 1991-2 (5.1) the load on the deck is $q_{fk} = 3.74 \text{ kN/m}^2$. This results in a load of 9.35 kN/m on one side of the bridge.

Note: The GSM optimization results in a design that can be used in an early design phase. In the design q_{fk} is reduced from 5.0 to 3.74 kN/m². In the optimization model this reduction isn't used and the full 5.0 kN/m² is applied, resulting in a distributed load of 12.5 kN/m.

B2.1.4 Load case 4: temporary mobile loads horizontal

The horizontal loads occur simultaneously with the vertical temporary load. The horizontal load is 10% of the vertical load q_{fk} and 60% for the load of a vehicle.

B2.1.5 Load case 5: extraordinary loads – accidental presence of vehicle

The possible presence of a vehicle is accounted as a special load case. The load configuration is shown in Figure 37. The first axis load is 80 kN and the second axis is 40 kN. No other live loads have to be added with this extraordinary load, if a service vehicle is defined with the LC3. This is not the case, thus this LC is used as an service vehicle in combination with LC3. If it is combined with the live loading in LC3 the vehicle has 5 meter clearance around it.

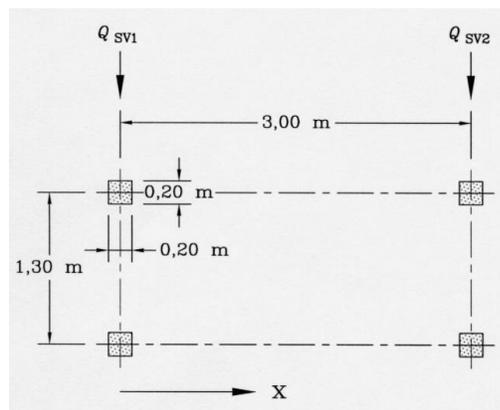


Figure 37. Configuration for the extraordinary load according to NEN-EN 1991-2 C1:Figuur 5.2

When the vehicle is driving on one side of the bridge the load on the truss structure is more than half the load, namely the first axis is 60 kN and the second axis is 30 kN.

Note: The live load LC3 is higher than the single vehicle, thus this load case LC5 can be omitted.

B2.1.6 Loads of the runup structure

The bridge is supporting the runup on both ends of the bridge, as shown in Figure 38. The runup is stabilizing itself for loads perpendicular to main direction of the structure. However, the

runup is transferring longitudinal and vertical loads to the bridge, and Table 13 has the summary for the loads.

Table 13. Loads of the runup structure

Load case	$F_{k,x}$ [kN]	$F_{k,z}$ [kN]
LC1 – self weight	210.5	47.8
LC 2 – permanent load	7.2	2.8
LC 3 – temporary load vertical	217	43.7
LC 4 – temporary load horizontal	70.2	16.3

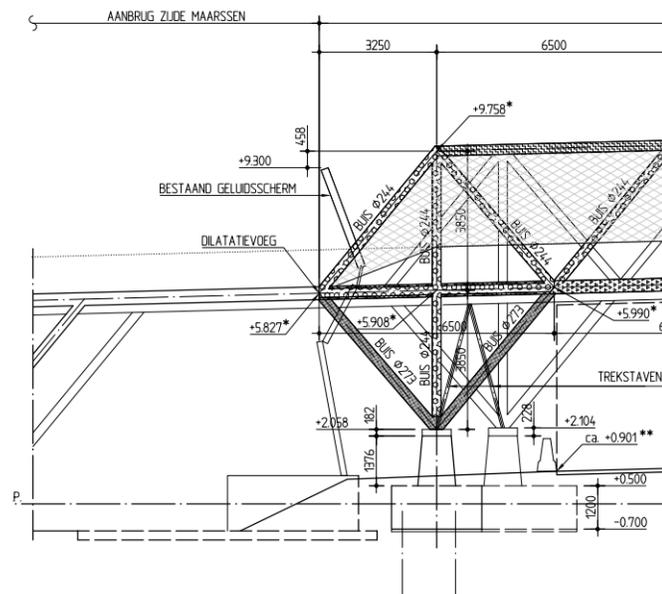


Figure 38. Schematic view of the connection between the runup and the bridge

B2.1.7 Wind load orthogonal to the structure

The wind load is 1.50 kN/m^2 . In the calculation for the wind load it is stated that 20% of the wind load passes through, because it is a skeletal structure. This results in a load of 2.4 kN/m in the bottom and top chord of the bridge.

B2.2 Load combinations

The formulas for combining the forces in the ULS are, according to NEN-EN 1990 6.10a & 6.10b, the least favorable of:

$$i) \quad E_d = \sum_{j \geq 1} \gamma_{G,j} G_{rep} + \gamma_{Q,1} \Psi_{0,1} Q_{k,1} + \sum_{i > 1} \gamma_{Q,i} \Psi_{0,i} Q_{k,i}$$

$$ii) \quad E_d = \sum_{j \geq 1} \xi_j \gamma_{G,j} G_{rep} + \gamma_{Q,1} Q_{k,1} + \sum_{i > 1} \gamma_{Q,i} \Psi_{0,i} Q_{k,i}$$

With the partial factors γ according to Table 14 and Ψ -values according to Table 15. A consequence class 3 must be applied to the bridge with a lifespan of 100 years.

Table 14. NEN-EN 1990 C1/NB Tabel NB.5. Partial factors for CC1 and CC2

CC	Blijvende en tijdelijke ontwerpsituaties	Blijvende belastingen		Overheersende veranderlijke belasting	Veranderlijke belastingen gelijktijdig met de overheersende	
		Ongunstig	Gunstig		Belangrijkste (indien aanwezig)	Andere
1	(Vgl. 6.10a)	1,2 $G_{k,j,sup}^a$	0,9 $G_{k,j,inf}$		1,35 $\psi_{0,1} Q_{k,1}$	1,35 $\psi_{0,i} Q_{k,i} (i > 1)$
	(Vgl. 6.10b)	1,1 $G_{k,j,sup}^b$	0,9 $G_{k,j,inf}$	1,35 $Q_{k,1}$		1,35 $\psi_{0,i} Q_{k,i} (i > 1)$
3	(Vgl. 6.10a)	1,5 $G_{k,j,sup}^a$	0,9 $G_{k,j,inf}$		1,65 $\psi_{0,1} Q_{k,1}$	1,65 $\psi_{0,i} Q_{k,i} (i > 1)$
	(Vgl. 6.10b)	1,3 $G_{k,j,sup}^b$	0,9 $G_{k,j,inf}$	1,65 $Q_{k,1}$		1,65 $\psi_{0,i} Q_{k,i} (i > 1)$

Table 15. NEN-EN 1990 C1/NB Tabel NB.17

Belasting	Belastingscombinaties									
	gr1	gr2	Q_{fwk}	Onbedoeld voertuig ^b	W^c		T^d		S	A1 ^{a,c}
Gelijkmatig verdeelde belasting	1	0,8	0	0	0,4	0,32	0,4	0,32	0	0,4
Horizontale belasting	1	1,0	0	0	0,4	0,4	0,4	0,4	0	0,4
Dienstvoertuig Q_{serv}	0	1,0	0	0	0	0,4	0	0,4	0,8	0
Geconcentreerde belasting Q_{fwk}	0	0	1	0	0	0	0	0		0
Onbedoeld voertuig	0	0	0	1	0		0		0	0
Wind F_{wk}	0,3	0,3	0	0	1,0	1,0	0,3	0,3	0,3	0
Temperatuur	0,3	0,3	0	0	0,3	0,3	1,0	1,0	0,3	0
Sneeuw	0	0	0	0	0	0	0	0	1,0	0
Impact op of onder de brug	0	0	0	0	0	0	0	0	0	1
Aardbevingsbelasting	0	0	0	0	0	0	0	0	0	0

^a A1 = aanrijding op of onder de brug en aanvaring.
^b Te beschouwen als gewoon belastingsgeval (geen calamiteit); zie 5.6.3.
^c Bij deze combinatie is in eerste kolom de verticale belasting vermenigvuldigd met Ψ_0 en de horizontale met Ψ_0^2 ; in de tweede kolom is dat omgekeerd; dit is gedaan om consistent te zijn met het gebruik van de groepen verkeersbelastingen.

C – Code

Within the thesis a 2D variant and 3D variant was made. Inhere the 2D variant is displayed. The code consists of three files: `build.py`, which creates the domain with the ground structure; `display.py`, which handles creating the figures; and `gsm2D.py`, which contains the optimization to find the optimal structure. The code of the optimization is a bit longer than the original 99-line code (He, et al., 2019), but it comes with additional flexibility, which was extensively used throughout this research. The code is written for testing purposes, and the run time of the member adding method could be improved significantly if the functions would reuse the nodal equilibrium matrix instead of creating a new one every iteration.

`gsm2D.py`

Note: the last three imports, `build`, `display` and `Tools`, are three other files that were placed in the same directory. The other libraries are either a standard library or available through `pip`. The script given here is from the optimization run of the 2D bicycle bridge. The member adding method is removed, since that can't be applied in the MILP.

```
import os
import numpy as np
from docplex.mp.model import Model
import shapely.geometry as geo
import time
import copy
import build
import Tools
from display import Plotting

def solve(active_member_ids: list):
    """Solve the LP problem
    Input:      nodes, supports, loads, base members (C1) and additional active
    members (C2+)
    Output:     updated additional members
               virtual displacements of the nodes
    """
    global SUPPORTS, LOADCASES, NODES, POTENTIAL_MEMBERS
    global ev, ew, ec, sigma, A_MIN
    global time_solve_start, time_solve_end

    time_solve_start = time.time()

    "make the vectors"
    l = [POTENTIAL_MEMBERS[nij][0] for nij in active_member_ids]
    f = []
    for k, loadcase in enumerate(LOADCASES):
        f.append(np.zeros(2 * len(NODES)))
        for load in loadcase:
            for i, xz in enumerate(NODES.values()):
                if (load.x, load.z) == xz:
```

```

        f[k][2*i] = load.px
        f[k][2*i+1] = load.pz
dof = np.ones(2*len(NODES))
for support in SUPPORTS:
    for i, xz in enumerate(NODES.values()):
        if (support.x, support.z) == xz:
            dof[2*i] = support.dofx
            dof[2*i+1] = support.dofz

"calculate the equilibrium matrix B"
B = build.calc_b(NODES, POTENTIAL_MEMBERS, active_member_ids)
"REMOVE THE DOF==0 ROWS"
dofzeros = []
for i, d in enumerate(dof):
    if d == 0:
        dofzeros.append(i)
B = np.delete(B, dofzeros, axis=0)
for k, fk in enumerate(f):
    f[k] = np.delete(fk, dofzeros, axis=0)

print(f'    Shapes: B: {B.shape} ; q: {len(active_member_ids)}')

"build the model"
model = Model()
# model.parameters.read.scale = 1
# model.parameters.mip.tolerances.mipgap = 0.0001
# model.parameters.mip.strategy.file = 2
# model.set_time_limit(3600)

a = model.continuous_var_list(len(active_member_ids), name='a', lb=0)
q = []
# s = model.binary_var_list(len(active_member_ids), name='s')
for k, fk in enumerate(f):
    qname = f'q{k}'
    qk = model.continuous_var_list(len(active_member_ids), name=qname, lb=-10 **
10, ub=10 ** 10)
    model.add_constraints(model.sum(model.dot(qk, B[i,:])) == fk[i] for i in
range(len(fk)))
    model.add_constraints(qk[i] <= sigma * a[i] for i in range(len(qk)))
    model.add_constraints(qk[i] >= -sigma * a[i] for i in range(len(qk)))
    q.append(qk)
# model.add_constraints(a[i] <= M * s[i] for i in range(len(a)))
# model.add_constraints(a[i] - A_MIN >= -M * (1 - s[i]) for i in range(len(a)))
"""min V:"""
model.set_objective('min', model.dot(a, l))
"""My model"""
# model.set_objective('min', (ev + ec / tw) * model.dot(a, l) + ew * 2 * tw *
model.sum(a) + ec * 3.1415 * tw * model.dot(s, l))

model.print_information()

"solve model and update results"
sol = model.solve(log_output=True)
print(sol.solve_details)

res_a = sol.get_values(a)
sol_q = list(sol.get_values(qk) for qk in q)
res_q = np.array(sol_q).T.tolist()

V = sum(res_a[i] * l[i] for i in range(len(res_a)))
Vs1 = sigma * V
ECIv = ev * V
ECIw = ew * 2 * tw * sum(res_a)

```

```

ECIc = 0
for ai, li in zip(res_a, l):
    if ai <= 0:
        continue
    if ai < A_MIN:
        ECIc += ec * li * 2 * (3.1415 * ai) ** 0.5
    else:
        ECIc += ec * (li * ai / tw + 3.1415 * tw)

print(f'A_MIN: {A_MIN}, A_MAX: {max(res_a)}, volume: {V}, volume(sigma=1): {Vs1}
number of members: {len(POTENTIAL_MEMBERS)}')
print(f'ev: {ECIv}')
print(f'ew: {ECIw}')
print(f'ec: {ECIc}')

for m, nij in enumerate(active_member_ids):
    POTENTIAL_MEMBERS[nij][3] = res_a[m]
    POTENTIAL_MEMBERS[nij][4] = res_q[m]

sol_data = [ECIv, ECIw, ECIc, V, Vs1, max(res_a), sol.solve_details]

model.clear()

return POTENTIAL_MEMBERS, sol_data

def run():
    global SUPPORTS, LOADCASES, NODES, POTENTIAL_MEMBERS
    global solve_data, solve_members
    global ev, ew, ec, sigma, A_MIN, tw

    print(f'Problem initialized with {len(POTENTIAL_MEMBERS)} potential members')
    all_active_members = list(POTENTIAL_MEMBERS.keys())
    print(f'Doing one iteration with {len(all_active_members)} members')
    POTENTIAL_MEMBERS, soldata = solve(active_member_ids=all_active_members)
    solve_data.append([*soldata, len(all_active_members)])
    solve_members.append(copy.deepcopy(POTENTIAL_MEMBERS))
    return

if __name__ == '__main__':
    time_start = time.time()
    time_solve_start = 0
    time_solve_end = 0

    "load variables"
    F = 182
    ev = 1310 # EUR/m3
    ew = 10500 # EUR/m3
    ec = 1.78 # EUR/m2
    sigma = 235000 # kN/m2
    M = 20 * F / sigma
    t = 10 / 1000 # m
    tw = 10 / 1000 # m
    A_MIN = 3.1415 * (tw ** 2) # m2

    "Save location"
    root = './data_2Dcase/minVunbounded'

    ""Init GSM""
    length = 32
    height = 8

```

```

DOMAIN = build.domain(width=length, height=height)

DOMAIN = DOMAIN.difference(
    geo.Polygon([[0, 4], [4, 8], [0, 8]])
)
DOMAIN = DOMAIN.difference(
    geo.Polygon([[length, 4], [length, height], [length - 4, height]])
)
DOMAIN = DOMAIN.difference(
    geo.Polygon([[4, height], [8, 4], [length - 8, 4], [length - 4, height]])
)
MESH = {'nodes_x': 2, # number of nodes in the x-direction
        'nodes_z': 2,
        'nodes_pm': 0.5, # number of nodes per meter in x and z-direction
        'type': 'nodes_pm'
        }

NODES = build.nodes(DOMAIN, MESH)
POTENTIAL_MEMBERS = build.ground_structure(DOMAIN, NODES, max_length=None)

SUPPORTS = [
    build.support(4, height, True, False, nodes=NODES),
    build.support(length - 4, height, False, False, nodes=NODES)
]

lc1 = []
q_stepsize = 4
for i in range(0, length+1, q_stepsize):
    print(f'stepsize {q_stepsize} at {i}')
    lc1.append(build.load(i, 4, px=0, pz=F, nodes=NODES))

LOADCASES = [lc1]
MAX_LOAD = build.get_max_load(loadcases=LOADCASES)

plot = Plotting(supports=SUPPORTS, loadcases=LOADCASES, max_load=MAX_LOAD,
domain=DOMAIN, nodes=NODES)
length_frac = 1 / MESH['nodes_pm']
for _m in POTENTIAL_MEMBERS.values():
    # if _m[0] <= 1.42 * length_frac: # activate members with connectivity level
1
    _m[1] = True # since no member adding can be applied, set all to True

"""Storing some intermediate results for analyzing later on."""
solve_data = []
solve_members = []

"""start sol"""
run()
"saving the data"
path = root # optionally add an extra subpath
if not os.path.exists(path):
    os.makedirs(path)

Tools.save(path+'/solve_members.pickle', solve_members)
Tools.save(path+'/solve_data.pickle', solve_data)
Tools.save(path+'/supports.pickle', SUPPORTS)
Tools.save(path+'/loadcases.pickle', LOADCASES)
Tools.save(path+'/nodes.pickle', NODES)
Tools.save(path+'/domain.pickle', DOMAIN)

for i in range(len(LOADCASES)):
    plot.plot_loadcase(POTENTIAL_MEMBERS, NODES, i)

```

```
plot.plot_members(POTENTIAL_MEMBERS, NODES)
plot.show()
```

build.py:

```
import numpy as np
import shapely.geometry as geo
from collections import namedtuple

Support = namedtuple('Support', ['x', 'z', 'dofx', 'dofz'])
Load = namedtuple('Load', ['x', 'z', 'px', 'pz', 'nid'])

def support(x, z, dofx, dofz, nodes: dict):
    load_nid = None
    for nid, xz in nodes.items():
        if xz[0] == x and xz[1] == z:
            load_nid = nid
            print(f'support added at node id {nid} {xz}')
            break
    assert load_nid is not None
    return Support(x, z, dofx, dofz)

def load(x, z, px, pz, nodes: dict):
    load_nid = None
    for nid, xz in nodes.items():
        if xz[0] == x and xz[1] == z:
            load_nid = nid
            print(f'load added at node id {nid} {xz}')
            break
    assert load_nid is not None
    return Load(x, z, px, pz, load_nid)

def get_max_load(loadcases):
    load_forces = [load_i
                   for lc in loadcases
                   for load in lc
                   for load_i in (load.px, load.pz)]
    return abs(max(load_forces, key=abs))

def domain(width, height) -> geo.Polygon:
    """build the domain here. extra info to make a complexer shape is handled in the
    main file"""
    d = geo.Polygon([(0, 0), (width, 0), (width, height), (0, height)])
    return d

def nodes(domain: geo.Polygon, mesh: dict):
    n = {} # nodes in dictionary: n = {node number: (x, z) , ... }
    l = domain.bounds[2]
    h = domain.bounds[3]
    convex = True if domain.convex_hull.area == domain.area else False
    """Create mesh"""
    if mesh['type'] == 'nodes_pm':
        # find best fit based on length, height and nodes pm
        pm = mesh['nodes_pm']
        nx = round(l * pm) + 1 # number of nodes on x-axis
```

```

    nz = round(h * pm) + 1
    dx = 1 / (nx - 1)
    dz = h / (nz - 1)
    ni = nx * nz # total number of nodes
    print(f'    Making mesh with {ni} nodes. nx:{nx}, dx:{dx}, nz:{nz}, dz:{dz}')
    cnt = 0
    if convex:
        for i in range(nx):
            for j in range(nz):
                n[cnt] = (i * dx, j * dz)
                cnt += 1
    else:
        "non-convex domain. check if point is in/on domain"
        for i in range(nx):
            for j in range(nz):
                pt = geo.Point(i * dx, j * dz)
                if domain.contains(pt) or domain.boundary.contains(pt):
                    n[cnt] = (i * dx, j * dz)
                    cnt += 1
    else:
        raise NotImplementedError
    return n

def ground_structure(domain: geo.Polygon, nodes: dict, max_length=None):
    """Create the ground structure with all the potential available members
    The GSM is fully connected. """
    pms = {} # potential_members = { (node1, node2): [length, on/off, mem_index,
sol_a, sol_q] , ...}
    cnt = 0
    convex = True if domain.convex_hull.area == domain.area else False
    if convex:
        for i, xzi in nodes.items():
            for j, xzj in nodes.items():
                if i < j:
                    dx = xzi[0] - xzj[0]
                    dz = xzi[1] - xzj[1]
                    length = (dx ** 2 + dz ** 2) ** 0.5
                    if max_length is not None:
                        if length > max_length:
                            continue
                    pms[(i, j)] = [length, False, cnt, 0, []]
                    cnt += 1
    else:
        "non-convex domain: check if line is within the design domain"
        for i, xzi in nodes.items():
            for j, xzj in nodes.items():
                if i < j:
                    line = geo.LineString([xzi, xzj])
                    if max_length is not None:
                        if line.length > max_length:
                            continue
                    if domain.contains(line) or domain.boundary.contains(line):
                        pms[(i, j)] = [line.length, False, cnt, 0, []]
                        cnt += 1
    return pms

def calc_b(nodes, members, members_using, alpha=1, jc=0, mod_l=1):
    """Calculate the equilibrium matrix B. Full matrix is being built."""
    """All nodes must be present in this calculateion. Indices based on node_id ni and
nj. """
    B = np.zeros((2 * len(nodes), len(members_using)))

```

```

for m, nij in enumerate(members_using):
    ni, nj = nij
    l = members[nij][0]
    dx = nodes[nj][0] - nodes[ni][0]
    dz = nodes[nj][1] - nodes[ni][1]
    B[2 * ni, m], B[2 * ni + 1, m], B[2 * nj, m], B[2 * nj + 1, m] = + -dx /
(alpha*l + jc)*mod_l, -dz / (alpha*l + jc)*mod_l, dx / (alpha*l + jc)*mod_l, dz /
(alpha*l + jc)*mod_l
return B

```

display.py:

```

import matplotlib.pyplot as plt
import shapely.geometry as geo
import Tools
plt.rcParams["font.family"] = "Times New Roman"
plt.rcParams.update({'font.size': 36})

class Plotting(object):
    def __init__(self, domain: geo.Polygon, nodes, supports, loadcases, max_load):
        self.domain = domain
        self.nodes = nodes
        self.supports = supports
        self.loadcases = loadcases
        self.max_load = max_load

    def show(self):
        plt.show()

    def design_domain(self):
        plt.figure()
        plt.axis('equal')
        x, y = self.domain.exterior.xy
        plt.plot(x, y, color='black')
        for interior in list(self.domain.interiors):
            pts = interior.coords
            plt.plot(*zip(*pts), color='red')

        domain_scaling = 1 / 2

        for support in self.supports: # HORIZONTAL SUPPORTS
            size = 2*domain_scaling # scaling factor for the drawing
            xval = [support.x, support.x + size, support.x - size, support.x]
            zval = [support.z, support.z + 1.2 * size, support.z + 1.2 * size,
support.z]
            if support.dofx == True:
                plt.plot([support.x - size, support.x + size], [support.z + 1.6 *
size, support.z + 1.6 * size],
                        color='black', )
                plt.plot(xval, zval, color='black', )

            _, _, zmin, zmax = plt.axis()
            plt.ylim(zmax, zmin)

    def plot_nodes(self, nodes, plt_domain=False):

        fig, ax = plt.subplots()
        plt.axis('equal')

        if plt_domain:

```

```

        x, y = self.domain.exterior.xy
        plt.plot(x, y, color='grey')
        for interior in list(self.domain.interiors):
            pts = interior.coords
            plt.plot(*zip(*pts), color='red')

    for (nx, nz) in nodes.values():
        plt.plot(nx, nz, '.', color='black')

    for k, (nx, nz) in nodes.items():
        ax.annotate(k, (nx, nz), fontsize=8)

    _, _, zmin, zmax = plt.axis()
    plt.ylim(zmax, zmin)

    def plot_members(self, members, nodes, plt_domain=False):
        plt.figure()
        plt.axis('equal')

        "scale lines based on the max load and supports+loads based on dimensions of
        the design domain"
        line_scaling = 1 / self.max_load
        _bounds = Tools.get_bounds(nodes)
        domain_scaling = 1 / max([_bounds['x']['max'] - _bounds['x']['min'],
        _bounds['z']['max'] - _bounds['z']['min']])

        for ninj, mem_info in members.items():
            c = 'tab:gray'
            n1, n2 = ninj[0], ninj[1]
            x_values = [nodes[n1][0], nodes[n2][0]]
            z_values = [nodes[n1][1], nodes[n2][1]]
            plt.plot(x_values, z_values, c, linewidth=0.6)

        for (nx, nz) in nodes.values():
            plt.plot(nx, nz, '.', color='black')
        for support in self.supports: # HORIZONTAL SUPPORTS
            size = 1.2*domain_scaling # scaling factor for the drawing
            xval = [support.x, support.x + size, support.x - size, support.x]
            zval = [support.z, support.z + 1.2 * size, support.z + 1.2 * size,
            support.z]

            if support.dofx == True:
                plt.plot([support.x - size, support.x + size], [support.z + 1.6 *
                size, support.z + 1.6 * size],
                color='black', linewidth=domain_scaling*7)
                plt.plot(xval, zval, color='black', linewidth=domain_scaling*7)

        if plt_domain:
            x, y = self.domain.exterior.xy
            plt.plot(x, y, color='red')
            for interior in list(self.domain.interiors):
                pts = interior.coords
                plt.plot(*zip(*pts), color='red')

        _, _, zmin, zmax = plt.axis()
        plt.ylim(zmax, zmin)

    def plot(self, members, title=None, update=True, plt_members=True,
    plt_nodes=False, plt_supports=False, plt_loads=False):
        plt.figure()
        plt.axis('equal')
        if title is not None:
            plt.title(title)

```

```

_bounds = Tools.get_bounds(self.nodes)

if plt_members:
    for ninj, mem_info in members.items():
        pa = mem_info[3]
        if pa > 10 ** -6:
            pq = mem_info[4]
            if all(pqk >= 0 for pqk in pq): # tension
                c = 'r'
            elif all(pqk <= 0 for pqk in pq): # compression
                c = 'b'
            else:
                c = 'tab:gray'
            n1, n2 = ninj[0], ninj[1]
            x_values = [self.nodes[n1][0], self.nodes[n2][0]]
            z_values = [self.nodes[n1][1], self.nodes[n2][1]]
            plt.plot(x_values, z_values, c, linewidth=1)

scaling = 1

if plt_nodes:
    for (nx, nz) in self.nodes.values():
        plt.plot(nx, nz, '.', color='black')

if plt_loads:
    for loads in self.loadcases:
        for load in loads:
            # print(load.__repr__())
            plt.arrow(load.x, load.z, 1.5 * load.px / self.max_load, 1.5 *
load.pz / self.max_load,
                    width=0.1,
                    color='black')

if plt_supports: # cantilever
    for support in self.supports: # VERTICAL SUPPORTS
        size = 0.1 * scaling # scaling factor for the drawing
        xval = [support.x, support.x - 1.2 * size, support.x - 1.2 * size,
support.x]
        zval = [support.z, support.z + size, support.z - size, support.z]
        if support.dofx == True:
            plt.plot([support.x - size, support.x + size], [support.z + 1.6 *
size, support.z + 1.6 * size],
                    color='black', linewidth=20 * .1)
            plt.plot(xval, zval, color='black', linewidth=20 * size)

_, _, zmin, zmax = plt.axis()
plt.ylim(zmax, zmin)

def plot_loadcase(self, members, nodes, k: int, title='', plt_supports=True,
plt_domain=False, plt_nodes=False):
    """
    :param members: potential members
    :param nodes: all nodes
    :param k: load case k=..
    :param title: default is 'Loadcase {k}'
    :param plt_domain: draw the design domain
    :return:
    """
    if k > len(self.loadcases):
        print(f'loadcase {k} is out of index')
        return
    loads = self.loadcases[k]

```

```

plt.figure()
plt.axis('equal')
if title == '':
    title = f'Loadcase {k+1}'
plt.title(title)

_bounds = Tools.get_bounds(self.nodes)
domain_size = max(_bounds['x']['delta'], _bounds['z']['delta'])

domain_scaling = 1 / 2

if plt_domain:
    x, y = self.domain.exterior.xy
    plt.plot(x, y, color='black')
if plt_nodes:
    for (nx, nz) in nodes.values():
        plt.plot(nx, nz, '.', color='grey')

if plt_supports:
    for support in self.supports: # HORIZONTAL SUPPORTS
        size = domain_size / 20 # scaling factor for the drawing

        "horizontal supports"
        size = 2 * domain_scaling # scaling factor for the drawing
        xval = [support.x, support.x + size, support.x - size, support.x]
        zval = [support.z, support.z + 1.2 * size, support.z + 1.2 * size,
support.z]

        if support.dofx == True:
            plt.plot([support.x - size, support.x + size], [support.z + 1.6 *
size, support.z + 1.6 * size],
                    color='black', )
            plt.plot(xval, zval, color='black', )

    for load in loads:
        # print(load.__repr__())
        plt.arrow(load.x, load.z, 1.5 * load.px / self.max_load, 1.5 * load.pz /
self.max_load,
                width=0.1,
                color='black')

    _, _, zmin, zmax = plt.axis()
    plt.ylim(zmax, zmin)

```

Tools.py

```

import numpy as np
import json
import pickle

# -----
#
# Tools dump data
# -----
#

def json_load(filename):
    with open(filename, 'r') as f:
        return json.load(f)

```

```

def json_save(filename, data):
    with open(filename, 'w') as f:
        json.dump(data, f)

def save(filename, data):
    assert filename.endswith('.pickle') # only pickle files
    with open(filename, 'wb') as file:
        pickle.dump(data, file)

def load(filename):
    assert filename.endswith('.pickle')
    with open(filename, 'rb') as file:
        return pickle.load(file)

# -----
#
# Tools lookup
# -----
#

def search_val_in_dict(d: dict, value=None) -> dict:
    # for dict in d
    for k, val in d.items():
        if val == value:
            return k

def get_dict_in_list(lst: list, key=None, value=None) -> dict:
    for d in lst:
        if d[key] == value:
            return d
    print('didnt find the dict in list!!!')
    raise NotImplementedError

def is_dict_in_list(lst: list, key=None, value=None) -> bool:
    for d in lst:
        if d[key] == value:
            return True
    return False

# -----
#
# Tools model
# -----
#

def get_active_nodes(member_ids) -> list:
    """
    :param member_ids: (ni, nj)

```

```

: return: node_ids [ni, nj, ni+1, ..]
"""
node_ids = []
for (ni, nj) in member_ids:
    if ni not in node_ids:
        node_ids.append(ni)
    if nj not in node_ids:
        node_ids.append(nj)
return node_ids

def get_bounds(nodes: dict) -> dict:
    _init_vals = nodes[list(nodes.keys())[0]]
    bounds = { # initialize
        'x': {
            'min': _init_vals[0],
            'max': _init_vals[0]
        },
        'z': {
            'min': _init_vals[1],
            'max': _init_vals[1]
        }
    }
    x = 'x'
    z = 'z'
    min = 'min'
    max = 'max'
    for (nx, nz) in nodes.values():
        if nx < bounds[x][min]:
            bounds[x][min] = nx
        if nx > bounds[x][max]:
            bounds[x][max] = nx
        if nx < bounds[z][min]:
            bounds[z][min] = nx
        if nx > bounds[z][max]:
            bounds[z][max] = nx
    bounds['x']['delta'] = bounds['x']['max'] - bounds['x']['min']
    bounds['z']['delta'] = bounds['z']['max'] - bounds['z']['min']
    return bounds

```

D – Joint-cost method

In Figure 39 the cost for a single member is given with a length of 1 meter. The left axis is the cost for the member for the min ECI and the right axis for the min V . From this graph it is directly clear that the ECI method is a linear combination of the original GSM method. A minimal area is required for the minimization of the ECI, because of the assumptions made in Chapter 4, while this isn't required for the original method.

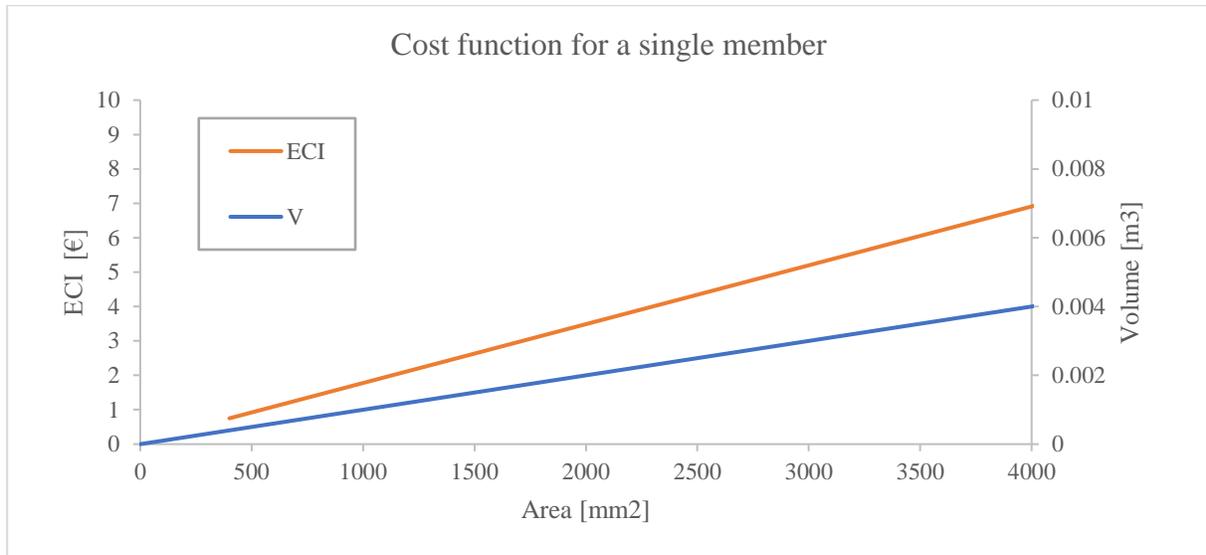


Figure 39. Comparing the cost of a single member with a length of 1 meter, a wall thickness $t=12$ mm and a joint length $t_w = 12$ mm

Applying relaxation to the objective function by removing the indicator variable s_i , and rewriting it using the distributive property for summation the objective becomes:

$$\min ECI = \sum_i^M \left(\left(e_v + \frac{e_c}{t} \right) l_i + 2e_w t_w \right) a_i + \pi e_c t l_i \quad (\text{eq.D1})$$

The additional constraints for the area are removed likewise. Because the indicator value s_i is removed, the last term in the function is not doing much in this relaxed optimization, and removing it makes:

$$\min ECI = \tilde{l}^T a \quad (\text{eq. D2})$$

where:

$$\tilde{l}_i = \left(e_v + \frac{e_c}{t} \right) l_i + 2e_w t_w \quad (\text{eq. D3})$$

It should be clear that the objective function of the relaxed min *ECI* is a parametrization of the original objective function min *V*. However, in this relaxed form the outer area of the structure isn't calculated accurately anymore. The min *ECI* formulation, as proposed in paragraph 4.4, is a stricter version, because of the required minimal area.

The joint cost method is:

$$\min_{a,q} \sum_i^M (l_i + j)a_i \quad (\text{eq. D4})$$

When modified length \tilde{l}_i is normalised to the length, the optimal joint cost is:

$$j = \frac{2e_w}{e_v + e_c/t} t_w \quad (\text{eq. D5})$$

Using this optimization method with the optimal joint cost will result in a good approximation.

Another way to think about this is that the joint cost can be used to configure the optimal refinement of the structure. The structures in Table 1 are made by varying the amount of nodes in the design domain, and it is clear that in the general GSM more nodes does result in a more refined structure. This is clearly not the structure with the least *ECI* cost. A similar effect can be achieved by varying the joint cost. The benefit of using the joint cost method is that it is that, if sufficient nodes are available in the design domain, the optimal refinement will be determined automatically. Just like the min *ECI* method does.

Applying the joint cost method to the cantilever (see paragraph 5.1) with a 1 by 1 meter domain meshed with 16 nodes per meter does result in the same optimal structure as the min *ECI* method. The optimization method is performed for various joint costs and are plotted in Figure 41 and Figure 42. The optimal structure with a joint cost of 0.0, 0.02, 0.028 and 0.1 meter are given in Figure 43. The two graphs can be divided into four sections. An initial 'unstable' section where the joint cost is between 0.0 and 0.005, and

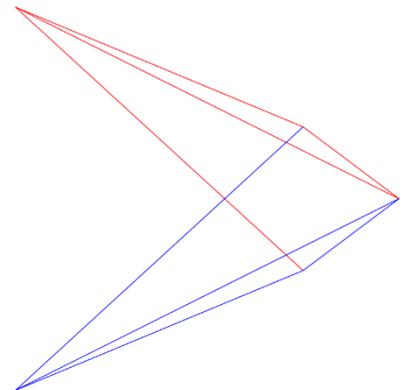


Figure 40. Optimal structure with a joint cost=0.03 meter

the optimal structure are slight variations of Figure 43a. The three other sections are stable and will result in the structures given in Figure 43b-d. Note that at the last transition, at a joint cost = 0.03 meter, there is some ‘unstable’ behaviour, which leads to a structure that is a combination of Figure 43c and Figure 43d, as is shown in Figure 40.

With the given ECI costs from Chapter 3 the optimal joint cost would be $j = 0.141$ meter, which would indeed result in the optimal structure of Figure 43d.

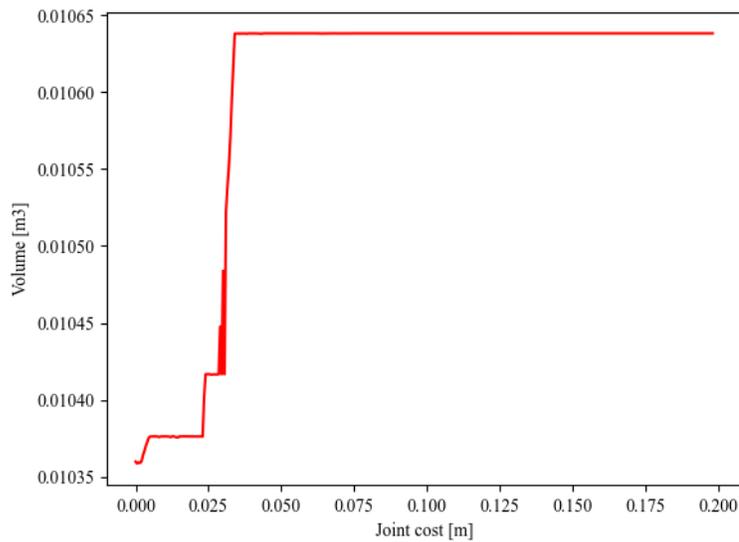


Figure 41. Volume of the optimal structure for various joint costs

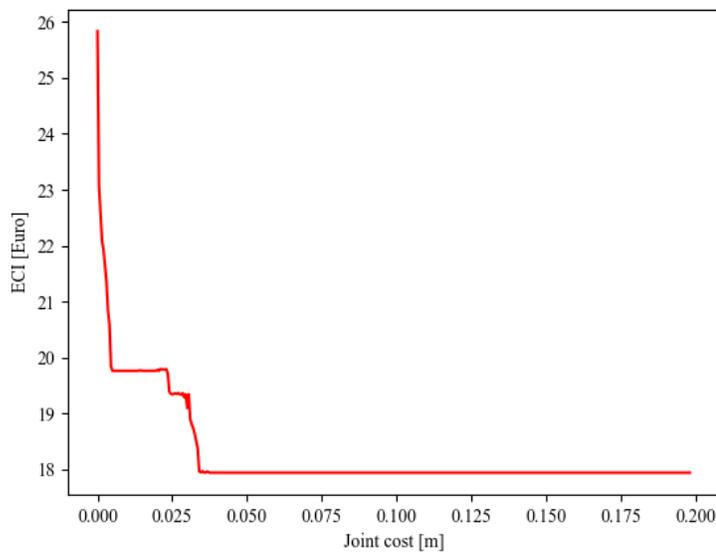


Figure 42. ECI cost of the optimal structure for various joint costs

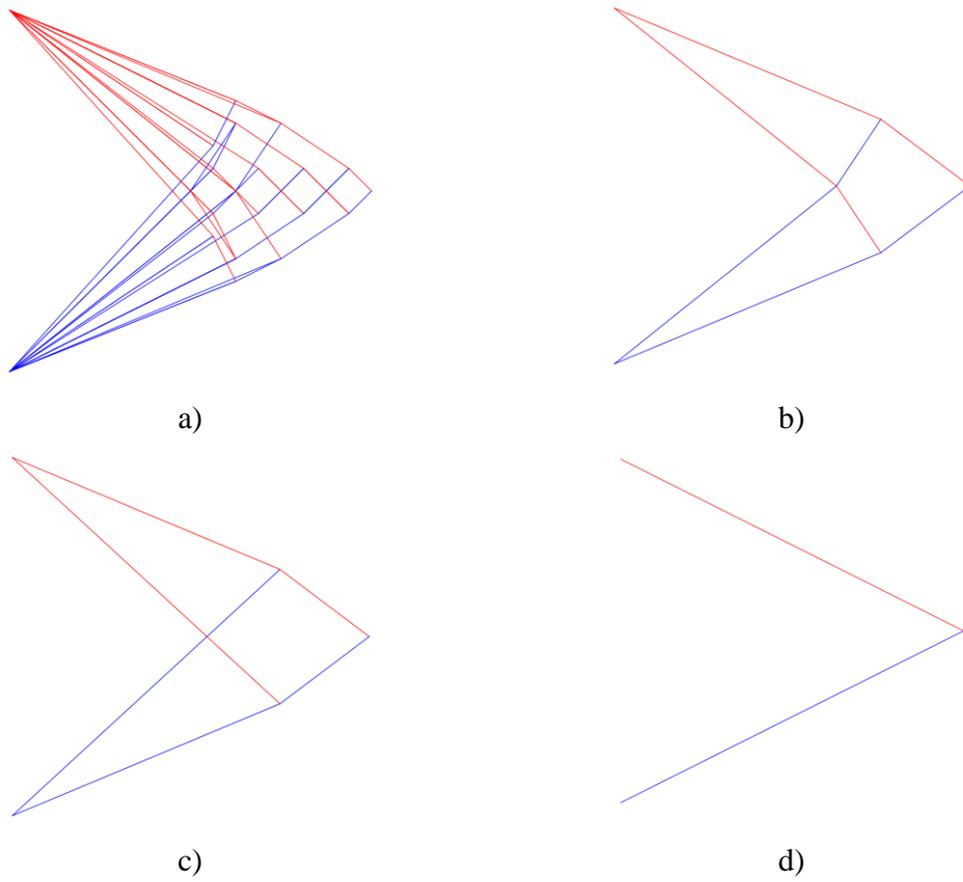


Figure 43. Optimal structures with a joint cost set to a) 0.00, b) 0.020, c) 0.028 and d) 0.10 meter