



Department of Built Environment

Design Space Exploration of 3D Concrete Printed Bridges

by

D.J.C. Hollanders

MSC THESIS

Assessment committee

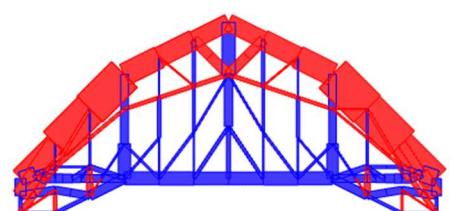
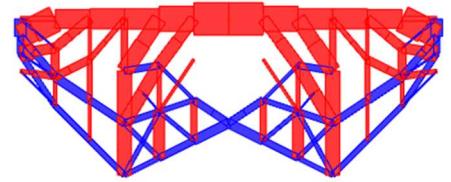
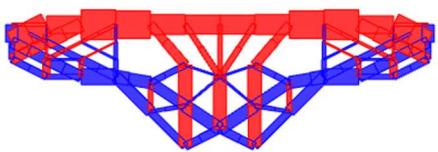
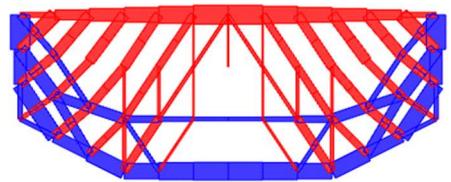
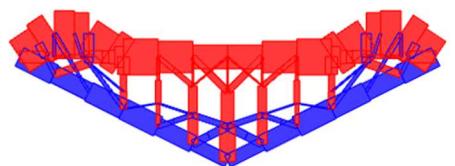
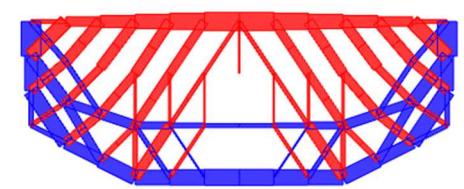
Member 1 (chair): Prof. dr.ir, T.A.M. Salet
Member 2: Dr.ir. , R.J.M. Wolfs
Member 3 Dr. P. Nourian

Graduation

Program: Built Environment
Capacity group: Concrete Structures
Date of defence: October 2, 2023
Student ID: 1255061
Study load (ECTS): 45
Track: Structural Engineering and Design

This thesis is public and Open Access.

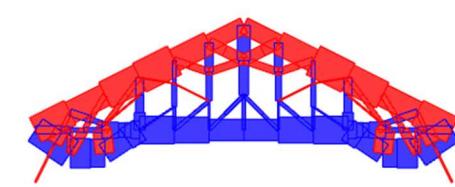
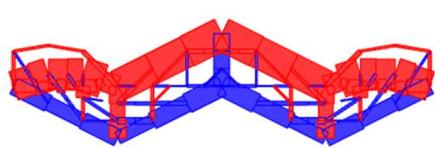
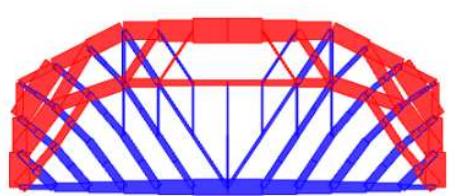
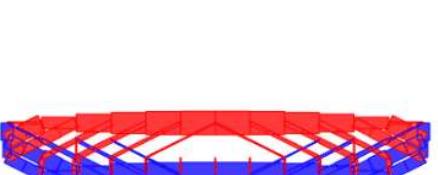
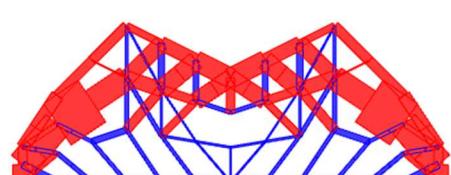
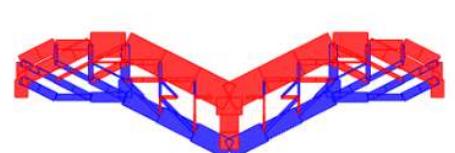
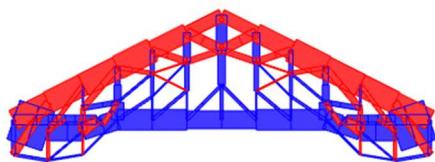
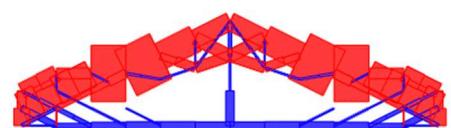
This thesis has been realized in accordance with the regulations as stated in the TU/e Code of Scientific Conduct. Disclaimer: The Department of the built environment of the Eindhoven University of Technology accepts no responsibility for the contents of MSc theses or practical training reports



Design Space Exploration

of 3D Concrete Printed Bridges

Dennis Hollanders 02-10-2023



I Abstract

3D concrete printing is rapidly emerging and numerous projects have showcased the potential of this technique. However, realizing the full scope of 3D printing properties in design applications remains challenging. This thesis therefore addresses the challenges of 3D concrete printing in design applications. To bridge this gap, a user-guided Design Space Exploration (DSE) approach is proposed. The focus lies on optimizing bridge designs, a suitable domain due to its alignment with the capabilities of 3D-printed concrete. The proposed dashboard aids users in identifying desirable topologies (i.e. “What to design?”) and guides the user in decision-making by illustrating parametric implications within the design space (i.e. “How to design”).

The general methodology comprises two steps: the creation of a parametric bridge and the development of the dashboard, used to navigate the design space. A generative design strategy is used to unify these steps, traversing from a single bridge to a design space for dashboard analysis. To elaborate, the parameters making up a single bridge, shape- and optimization parameters, will be cross-referenced. Design principles are established to ensure that the created bridges are print-specific. These principles include two-dimensional design, modular, sizing optimization, unreinforced concrete, and diversity of topologies. These are used in the parametric development of a bridge. Which starts with varying design domains, making up the line model or ground structure of each bridge, yielding diverse topologies in the generative design process. The next steps in the process include assigning structural properties to the line model, after which a structural calculation will determine the strain energies in the system. Dividing the strain energy by the total strain energy gives the strain energy ratio of each element to that of the system. This ratio is the driver behind the heuristic optimization that follows. In the further development of the optimization algorithm, tools have been added to develop its functionality. One of which is the Young's modulus penalization, which tricks the optimization into optimizing towards compressive structures by fictively reducing the Young's modulus of elements in tension. Additionally, a tool is created able to set a construction constraint to the optimization, forcing material to a location by setting a minimum strain energy bound in the optimization process. Finally, numerical examples are used to benchmark or exemplify the tools making up the design space of a single bridge.

This design space resulting from the described process, is being navigated with a dashboard and created in dash, a Plotly library. The dashboard equips users with an array of tools, enabling them to identify bridges that align with their preferences. This involves defining self-set criteria within the design space, allowing users to make informed decisions through analysis graphs. The dashboard uses a

total of six graphs to comply with this functionality: a domain constraint graph, filtering histogram, 3D objective space, t-SNE plot, parallel categories plot and a visualization tool. To ensure the ability for a comparative analysis, the inserted databases are being normalized via a rank transformation, which assigns a rank to each element based on its specific output variable.

To validate the effectiveness of the dashboard, a design case has been formulated. The scenario concerns a bridge with an eight-meter span, in which the variable domain heights are confined to a range of positive and negative three meters.

The results from the case show that the “What to design?” can be answered effectively. Regardless of the complexity of the design question, the dashboard could provide a comprehension comparison between optima designs. The index selection tool facilitates a straightforward comparison between identified bridge designs and desired outcomes. Notably, the ranking method eliminates the complication in normalization due to outliers. However, it does introduce a trade-off by diminishing the ease of comparing due to the absence of relative differences. The ability to compare how different design queries lead to different optima is perceived to be a functional method to convey decision-making with non-technical personnel. Secondly, the dashboard provides information about the design space. Through the utilization of the parallel categories plot, information can be gained regarding the distribution of input variables and their impact on output performance. The results have shown that this provided basic information about domain functioning, showing that low values for domain 3 perform poor in general.

Studying the parameters shows that prestress seemed to be a positive addition to the structures' functioning. The Young's modulus penalization tools behave very case-specific. While one penalization method occasionally outperforms the others, a slight preference is observed for non-penalized structures. The Construction constraint tool causes lower total strain energy performances, as expected, but does however proof not to be a postprocessing step; it distinctly alters the distribution of material around the constrained element. It is therefore seen as a valuable tool to control material placement throughout the optimization. In addition to this, the generative design process in combination with optimization is seen as a valuable methodology to impose significant variation to the topological design space.

To conclude, the dashboard shows to be able to address both research objectives, providing information on the “what and How to design?”. While the methodology applied shows to provide bridges tailored to the specific needs of 3D concrete printed bridges. Consequently, this underscores the potential of user-guided design space exploration as an effective tool for engineers and designers, navigating the complex design landscape of 3D concrete printed structures.

II Preface

In this thesis, I've had the opportunity to explore my passion for parametric design and expand its horizons within the innovative realm of 3D concrete printing. This thesis has led me to tackle interesting design challenges and delve into various fascinating fields such as optimization, generative design but also provided a case develop programming skills.

I would like to thank Dr.ir. R.J.M. (Rob) Wolfs a lot for his excellent guidance throughout the duration of the entire thesis, aiding in every part of this thesis. In addition to this, I would like to thank Dr. P. (Pirouz) Nourian for his expertise in the field of computational design and his guidance on this subject. I would like to thank Prof. dr.ir, T.A.M. (Theo) Salet as chair of the commission for his passion on the subject at hand which raised the interest of graduation in this field. Finally, I would like to express thanks to the other meeting members ir. D.H. (Derk) Bos, ir. I. (Idil) Gümrük, and M. (Mathijs) Vervoort for their valuable contributions.

Furthermore, I would like to thank all other friends, students and family who actively thought with me and helped me throughout this thesis.

Dennis Hollanders
Eindhoven
02-10-2023

Contents

<u>I ABSTRACT</u>	<u>III</u>
<u>II PREFACE</u>	<u>IV</u>
<u>CONTENTS</u>	<u>V</u>
<u>CHAPTER 1 INTRODUCTION</u>	<u>7</u>
1.1 PROBLEM STATEMENT	7
1.2 RESEARCH OBJECTIVE	7
1.3 THESIS OUTLINE	8
<u>CHAPTER 2 LITERATURE REVIEW</u>	<u>9</u>
2.1 WHAT IS DESIGN SPACE EXPLORATION?	9
2.2 WHY DESIGN SPACE EXPLORATION?	9
2.3 DESIGN SPACE EXPLORATION EXEMPLIFIED	10
2.4 APPLICATION IN THE BUILT ENVIRONMENT	10
2.5 DESIGN SPACE EXPLORATION METHODS SUMMARIZED	11
<u>CHAPTER 3 LEADING DESIGN PRINCIPLES</u>	<u>12</u>
<u>CHAPTER 4 METHODOLOGY</u>	<u>13</u>
4.1 WORKFLOW	13
4.2 IMPLEMENTATION DETAILS	14
<u>CHAPTER 5 CREATING THE DESIGN SPACE</u>	<u>16</u>
5.1 GENERATING THE GROUND STRUCTURE	16
5.2 FINITE ELEMENTS	17
5.3 GROUND STRUCTURE OPTIMIZATION	18
5.4 OPTIMIZATION TOOLS	19
5.4.1 CONVERGENCE	19
5.4.2 STEP SIZE DAMPING	19
5.4.3 YOUNG'S MODULUS PENALIZATION	20
5.4.4 CONSTRUCTION CONSTRAINT	22
5.5 PARAMETRICIZED TREE	23
<u>CHAPTER 6 DESIGN SPACE EXPLORATION DASHBOARD</u>	<u>26</u>
6.1 FROM DATASET TO DESIGN SPACE	26
6.2 GRAPHS MAKING UP THE DASHBOARD	26

<u>CHAPTER 7 NUMERICAL EXAMPLES</u>	<u>30</u>
7.1 GENERAL GROUND STRUCTURE	30
7.2 YOUNG'S MODULUS PENALIZATION	32
7.3 PRESTRESS	36
7.4 BEAM-TRUSS RELATIONSHIP	36
7.5 CONSTRUCTION CONSTRAINT	37
<u>CHAPTER 8 DESIGN CASE 3DCP</u>	<u>39</u>
8.1 CASE DESCRIPTION	39
8.2 "WHAT TO DESIGN?"	39
8.1.1 CASE 1 MINIMIZE VOLUME WITH PRESTRESS	40
8.1.2 CASE 2: MULTI-OBJECTIVE CONSTRAINED OPTIMIZATION	42
8.3 "HOW TO DESIGN?"	44
8.3.1 YOUNG'S MODULUS PENALIZATION	44
8.3.2 CONSTRUCTION CONSTRAINT	44
8.3.3 SHAPE ANALYSIS.	45
<u>CHAPTER 9 CONCLUSION</u>	<u>46</u>
<u>CHAPTER 10 DISCUSSION AND FUTURE RESEARCH</u>	<u>48</u>
<u>BIBLIOGRAPHY</u>	<u>49</u>
<u>APPENDICES</u>	<u>51</u>
APPENDIX A CODE: FEA, OPTIMIZATION AND DASHBOARD	51
APPENDIX B FINITE ELEMENTS CALCULATIONS	51
APPENDIX C STRAIN ENERGY CALCULATIONS	53
APPENDIX D OVERVIEW OF THE GRASSHOPPER SCRIPT.	54
APPENDIX E ADDITIONAL RESULTS	55
APPENDIX F OVERVIEW CALLBACK FUNCTIONS	61
APPENDIX G OVERVIEW OF THE DASHBOARD	61

Chapter 1 Introduction

1.1 Problem Statement

The pressing challenges of climate change and other environmental issues are driving innovation across various sectors, including the construction and infrastructure industries. The resulting necessities of sustainable solutions have driven the sector into a new era of innovative construction techniques. One of such innovative approaches is 3D concrete printing (3DCP). The layer-by-layer deposition technique is gaining momentum due to its perceived benefits. Over recent years, numerous studies have been conducted showing these intended benefits, such as reduced material waste, increased flexibility in design and faster construction times. On the other hand, highlighting limitations that have to be taken into account when adopting such an approach (El-Sayegh et al., 2020).

Meanwhile, the industry's growing interest in 3DCP is evident, and the Technology Readiness Level (TRL) of 6-7 suggests that it's time for the industry to transition from technology demonstration projects to developing systems and subsystems for practical applications (Ma et al., 2022). Therefore, the challenge of 3D printing shifts from solving the technical hurdles, to using the advantages, but also the disadvantages, in future designs. This implies that the question moves to applying 3DCP in the best manner for a specific case. Known design problems will have to be reanalyzed in order to find the most suitable options for 3D printing. Resulting in a change of codes, guiding the designs, to match the properties of 3D printing (Bos et al., 2016).

A closer examination of the current design applications reveals that most cases have been focusing on demonstrating the intended benefits, this aligns with the TRL scale. One of these benefits is geometrical freedom, by creating free-form structures that would have been difficult to construct with traditional concrete methods (Buswell et al., 2018). Additional projects have been focusing on highlighting the potential of speeding up construction and reducing manual labour (Hossain et al., 2020). Nevertheless, while these projects show some of the benefits not often design cases are established truly trying to utilize the full potential of 3DCP, some projects that do are found in the source (Liu et al., 2022). Noticeable is the amount of bridge-specific design cases. All the intended benefits and limitations can be utilized and analyzed in the design case of a bridge, making it a fitting subject for investigation within this thesis.

The complexity in designing for 3DCP arises from the intricate balance between design parameters and manufacturing constraints, each of which has significant implications on the final design. In addition to this, the high degree of geometric freedom allows 3DCP to explore a broader and often more complex design space than

conventional methods. As 3DCP continues to develop, there is an increasing need for new design strategies that can navigate these complex design spaces. Simultaneously, new methods are required for analyzing the resulting designs ensuring effective and efficient use of 3DCP. These methods need to accommodate the unique advantages and challenges of 3DCP while providing a framework for design exploration and optimization (F. Bos et al., 2016).

Currently, a lot of design solutions are sought in the field of topology optimizations. However, adding complexity to these optimizations, making them tailored to the design questions specific to 3DCP, complicates the landscape of objective and constraint spaces. Furthermore, finding true design optima would involve optimizing to more than one objective, which further deepens the complexity of this design strategy. These objectives often conflict with one another, resulting in a set of 'Pareto-optimal' solutions, where no single design can be considered the best in all respects (Marler & Arora, 2004). In addition to this, current practices on 3DCP will require new computational models and thus design methodologies while building codes will adapt to 3DCP, meaning that the black box kind of nature of an optimization does not increase the knowledge about the subject but would only steer into a tunnel-visioned approach (R. A. Danhaive & Mueller, 2015).

Design Space Exploration (DSE) can be an instrument in resolving the challenges posed by these design questions for high-dimensional design spaces in the field of 3DCP. DSE is a systematic approach that seeks to analyze, identify, evaluate, and thereby optimize a wide array of design solutions within the feasible design space (Ha & Teich, 2017). Overall, DSE could be applied as a tool for navigating the complex design spaces of 3DCP, providing additional insights into the design methodologies while on the other hand facilitating a method to pick the actual desired designs. This approach could be utilized on two levels in 3DCP: 'What to design?', the actual topology or result, the bridge, and 'How to design?', choosing suitable settings for the optimization algorithm. As such, this thesis proposes a design space exploration model specifically for 3DCP, with a particular focus on bridge design, to facilitate a more comprehensive exploitation of this technology's potential.

1.2 Research Objective

This research aims to develop a model that aids in understanding how design parameters and manufacturing constraints influence the design and optimization of 3D-printed bridges. Instead of complying with conventional practices, the study proposes a new methodology, focusing on output parameters rather than input parameters, with the intent to minimize trial and error in the design process. (R. A. Danhaive & Mueller, 2015). This approach is based on the principles of Design Space Exploration (DSE), which have been shown to yield more desirable design results in comparison to conventional optimization strategies (Kawaguchi et al., 2016).

DSE maps multiple solutions of a design space based on varied input parameters and thereby provides a

catalogue-like solution instead of a single absolute topological answer. By comparing the designs in the catalogue, a better understanding of parametric implications can be achieved, which in its terms could result in more desirable results. As is shown by Brown's research on the effect of performance feedback on the design process (Brown et al., 2016).

In this thesis, the DSE methodology is applied to the design of 3D-printed concrete bridges. Parameters from various stages of the initial design process will be mapped, and optimization tools relevant to the limitations of 3DCP design will be explored. Thereby being able to analyse how these tools influenced the output parameters of the final result and thereby the topology. However, the goal of this research is not to offer definitive answers to specific design questions but to develop a practical methodology that can be used, especially in the initial phases of design decision-making. The objective is to demonstrate that the proposed model can furnish insights into how various parameters impact the optimization process and, consequently, the overall design. Given the particular focus on the methodology, clear objectives are formulated to steer the research in this direction.

Main objectives:

- ***Develop a model capable of determining the optimal topology***, identifying desirable topological shapes for 3D-printed bridges within the context of the given DSE case and calculation methods ("What to design?").
- ***Enhance decision-making*** by providing insights into parametric implications, leading to improved decision-making within the complex design space ("How to design?").

Sub objectives:

- **Develop** an optimization script that takes into account manufacturing constraints as variables in the design space.
- **Investigate** the influence of a variable design domain on optimal topologies.
- **Explore** the possibility of designing with output parameters instead of input parameters.

By realizing these objectives, this research aims to contribute to the understanding of how to design with optimizations within the field of 3DCP. Thereby providing a platform that cannot only be used to answer what to build but also provide knowledge about how to build 3D-printed structures.

1.3 Thesis outline

In the upcoming chapter, this thesis will first elaborate on applications and different methodologies within design space exploration by exemplifying certain projects. In Chapter 3 the main design principles will be elaborated, these guide the design process that follows into the insurance to create 3D print-specific designs. Chapter 4 then

continues with the general workflow applied in this thesis containing a flowchart of the specific process. This flowchart can be seen as a guiding thread throughout this thesis. From this point onwards Chapters 5 and 6 elaborate on the general methodology in Chapter 4. Chapter 5 follows the path leading to the creation of a single bridge. Tables will be provided with different parameters making up the design space, in which a distinction is made between generative parameters, used to create the design space, and 'fixed' parameters. Chapter 6 provides an explanation of the content and development of the dashboard, which is being created as a user-guided design space exploration tool. After explaining the general methodology and components of that methodology, the thesis continues with numerical examples in chapter 7. These examples try to validate or exemplify the working of the components creating a bridge. After validating that the model is able to generatively create a design space for 3D concrete printed bridges, a design case is established. This design case will be elaborated in Chapter 8. The goal of this case is to validate the functioning of the dashboard. This chapter is subdivided into two parts, where the first part focusses on finding a desirable topology, given a certain objective of what the bridge should be. Whereas the second part tries to analyze the design space to see how input variables influence the outputs. The final chapters include the Conclusion, chapter 9, and Discussion and Future Research, Chapter 10.

Chapter 2 Literature Review

2.1 What is Design space exploration?

Design space exploration (DSE) is an approach in the field of engineering design, that focuses on the systematic analysis and process of finding potential design solutions within a defined design space. The objective of DSE is to identify desired optimal designs that meet the constraints, while simultaneously satisfying multiple objectives. Due to the nature of presenting a set of optimal results, final objectives do not only have to include performance measurements but can also include subjective criteria such as aesthetics (Ha & Teich, 2017). For the implementation of DSE, four general steps are run through. First, a design space is defined and generated, which contains all possible design variants. This design space is then evaluated to gain insights into the effect certain parameters have on the variance in the dataset. Whereafter a navigation strategy is defined to find the desired solutions, which are finally being visualized (Danhaive, 2020).

Design Space Exploration (DSE) and Design Space Analysis (DSA) are often used interchangeably, but refer to two distinct steps: DSE is about what to build, and DSA is about how to build. DSE is then again used as the general term for the methodology. Other paradigms that closely relate are ‘design by shopping’, ‘meta-modelling’, and ‘Model-Based Systems-Engineering (MBSE)’(MacLean et al., 1991; Stump et al., 2004; Timperley et al., 2023). All these concepts vary in the exact approach, but the main idea remains to provide a methodology that allows for the exploration of design solutions.

2.2 Why Design Space Exploration?

DSE's true strength lies in its capability of navigating complex design spaces. In these landscapes, single optima are unlikely to occur. This results from high dimensionality or multiple objective criteria that are being set to the optimization, which causes an array of solutions to be optimal, depending on various ratios between the objective

criteria. This is referred to as the Pareto front, which is visualized in Figure 2.1.

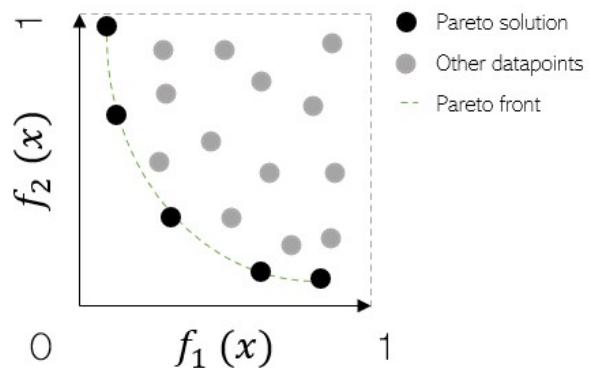
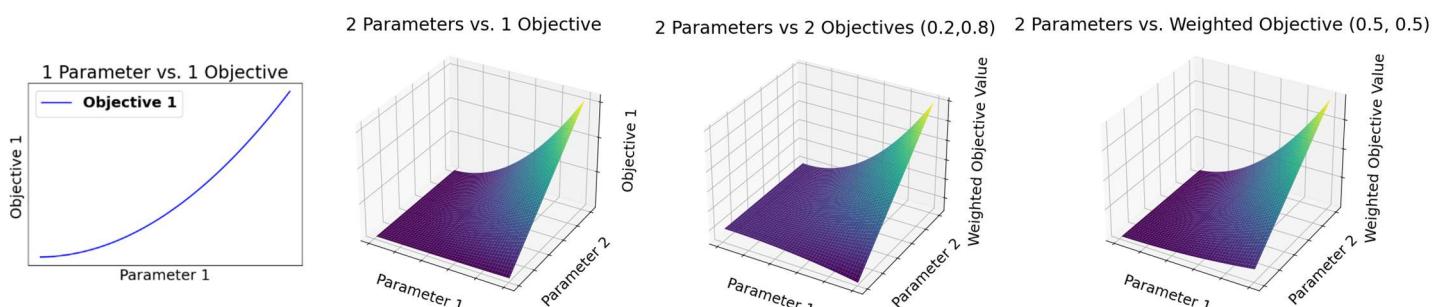


Figure 2.1 Pareto front between objectives $f_1(x)$ and $f_2(x)$

Another complexity of the design space could be due to the high dimensionality of the parameters influencing the objective space. The numerous parameters involved make the search for optimal results more difficult, which again leads to a higher likelihood of being stuck in a local minimum. DSE analyzes how these input and output parameters relate and thereby navigates more efficiently through the design space. For extremely high dimensionalities, algorithmic and surrogate techniques are applied to improve the navigability of the design space. These techniques will be further exemplified in chapters 2.1.3 and 2.1.4.

Figures 2.2 illustrates the increased complexity due to the dimensionality of an abstracted optimization problem. The first Image 2.2, describes the design question of a single parameter with a single objective. In this case, there will be no problem in finding the numerical optima. However, adding a second objective, as shown in image 2.3, and increases the dimensionality. Which causes the task of pinpointing the minimum point within this expanded parameter space to become notably intricate. To further illustrate the intricacy of complex design spaces a second incommensurable objective is added. Resulting into a Pareto-like objective space as shown in image 2.1. Picking an solution on that line is based on a certain bias by the user, basically inducing weights to the objective formulation. Images 2.4 and 2.5 show that the objective space develops with changed weights set by the user, illustrating that the weights set by the user influence the amount of design the



f.l.t.r **Figure 2.2,2.3,2.4 and 2.5** Visualization of objective spaces

single global minima that arises due to the weight settings in image 2.4 compared to multiple global minima in 2.5. This abstracted example underscores the difficulty posed by dimensionality and the bias underpinning the optimization of objective function weights, as explained by Caitlin Mueller (Tseranidis et al., 2016). Conventional techniques often tend to yield a single solution that is biased by these predefined ratios of the objectives. This means that many potential solutions are overlooked. Addressing this challenge requires effective mapping techniques for design space exploration, a fundamental aspect emphasized by (Nourian et al., 2023). In line with this, DSE offers a crucial solution, providing an explicit navigable tool for the design space, allowing users to explore and analyze all possible solutions. Thereby, DSE facilitates not only a platform for decision-making, but also enhances our understanding of how input parameters influence the design space, making it a valuable tool for designers.

2.3 Design Space Exploration Exemplified

While DSE is a rather unconventional technique in the practical field of structural engineering, it is a relevant technique in the aerospace industry. Due to conflicting constraints of aerodynamics and structural integrity and the long computation times that come with these calculations. Machine-learning techniques are often applied to a design space, training a model to understand how input and output parameters are influenced by each other. Instead of calculating every solution, a design of experiments [DoE], also known as a sampling plan, is being used to draw samples from the design space. Based on these samples a model is being trained. This surrogate or meta-model is then used to identify the set of desirable design solutions. This means that DSE in these cases not only provides the benefits of navigation and understanding but also provides a methodology that is time effective. Examples of surrogate-based design space explorations can be found in: (Han et al., 2018; Pagliuca et al., 2019; Sohst et al., 2022).

Another example of design space exploration, but on a lower level of automation, is the use of information models, a link to a user-guided DSE model to design aeroplane wings: [Aircraft Design with Dash](#) (Peter Sharpe,

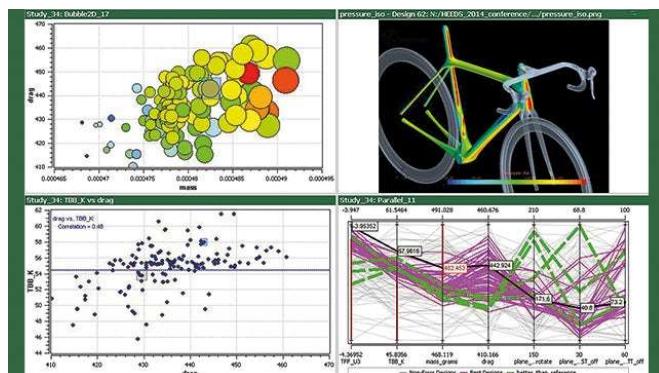


Figure 2.6 Heeds user-guided Design Space Exploration (Siemens, 2021)

2021). Similar to the aerospace industry, Siemens developed a DSE platform, HEEDS. With this platform, Siemens performed bicycle optimizations in collaboration with the pro cycling team Trek (Siemens, 2021). With this multi-objective optimization, Trek claims to have saved 182 grams of frame material, which would make their team faster by 60 seconds per hour. The display of this project in HEEDS is shown in image 2.3 (Trek, 2021). Another example of DSE in commercial programs could be Autodesk's generative design tool, Dreamcatcher. It used the level-set method to generate topological solutions based on the user's input (Autodesk). In the level-set method, the topology is minimized. Yet, several variables make up the way this approach would alter the original shape. The program calculates several alternatives and provides some minimal volume results to the user. Additionally, other functionalities, such as dream sketch allow for rapid prototyping during the sketch phase (Kazi et al., 2017). Thereby Similarly Abaqus Insight, and MATLAB Simulink, have custom integrations that could be used for design space exploration (Butt & Lavagno, 2013; Inceptra, 2023). However, a disadvantage of these programs is that the user is limited to the framework presented for exploration and analysis (Scharl & Mavris, 2001) (Inceptra, 2023; Trek, 2021).

2.4 Application In The Built Environment

DSE presents a potent approach for the built environment, both for architecture and structural design. By enabling a powerful analysis of design alternatives. DSE could extend beyond conventional generative design. It can therefore effectively be integrated into already existing software and tools such as Grasshopper and Dynamo, enhancing their capabilities. Pioneers in the field, one of whom is Caitlin Mueller, have provided significant research into the integration of this technique into structural, urban, and architectural design. Thereby exemplifying the potential of DSE in optimized design solutions for the described fields. One common methodology is cataloging the design space and creating a visual library of possible design solutions. Additionally, it illustrates relationships and dependencies between design variables. By mapping visual solutions on the objective space, designers can set their objective ratio based on their subjective preference for a certain design (C. T. Mueller, 2014). A higher-level implementation is the application of algorithmic processes in the analysis of potent designs. These range from rule-based algorithms to complex evolutionary or particle swarm algorithms. Based on predefined performance criteria these algorithms iterate over the design spaces, generating, evaluating and selecting potential solutions, see image 2.7. Promoting those with superior performance characteristics. These techniques could be meaningful for extremely large design spaces but are again more sensitive to local minima (C. Mueller & Ochsendorf, 2011). More complex design spaces have been analyzed with surrogate models (Tseranidis et al., 2016). In her work, the approximation speed of the surrogate was used for 'live designing' a topology-optimized structure. Furthermore, Caitlin Mueller has researched the potential of

reusing surrogate models, using the same model for a different design case (Whalen & Mueller, 2022) (Danhaive, 2020). Additionally, research has been conducted on creating surrogates from sketches. As a result of their research, they developed a grasshopper plugin allowing the creation of surrogate models based on parametric in and output, named Stormcloud. Likewise, Stuttgart University has created a plugin named Opossum (Mueller Caitlin et al., 2015; Wortmann, 2017).

2.5 Design Space Exploration Methods Summarized

This review of DSE applied in different fields shows that different techniques within DSE are explored to present a solution set to the user. This could be abstracted into four different types of DSE which increasingly further depend on non-human-decision-making for presenting the desired solution set. Thereby allowing for a larger design space to be explored but with an increasing risk of missing out on possible solutions. (Ha & Teich, 2017):

- **User-guided DSE:** In this methodology, DSE is merely a tool that visualizes the design space and relies totally on human decision-making. This implies that all desirable results can be explored by the user and the subjective objectives could largely influence the result.
- **Heuristic and Pseudorandom DSE:**
When design spaces increase to a certain amount it might become an impossible task to compare all results. This type of DSE randomly explores the design space to find desired results but is likely to overlook some potential promising designs.
- **Evolutionary DSE:** Genetic algorithms or swarm optimizations could decrease the dimensionality and present a set of most promising results to the user. A drawback of this approach could be that certain regions of the design space are overlooked.
- **AI implemented DSE:** For large design spaces that are computationally intensive surrogates or metamodels could be applied as tools for the user to explore the design space. Additionally, these techniques can interpolate (and extrapolate with increasing inaccuracy) through the design space covering a wider range of possible solutions.

To conclude, DSE can generally be seen as a step further than generative design, in which not just variants or options are created but also a methodology that allows one to browse through the generated dataset to pick the desired optima.

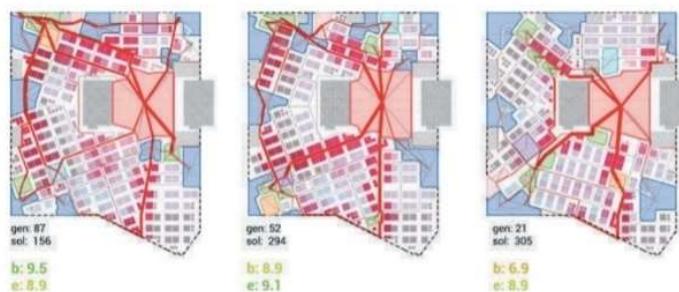


Figure 2.7 A model for Generative Space planning using Evolutionary Desing Space Exploration techniques (Danil Nagy et al., 2017).

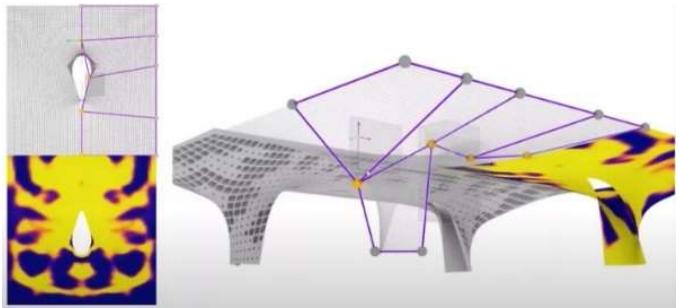


Figure 2.8 Surrogate model by (Caitlin Mueller, 2022). Allows to rapid prototype a complex topology optimization including post-processing steps.

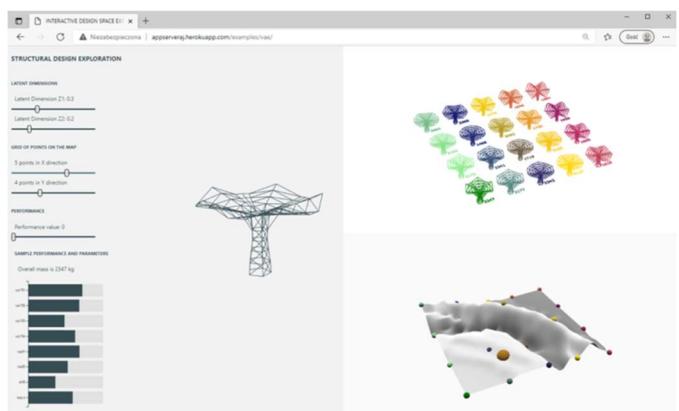


Figure 2.9 AI enhanced Design Space Exploration by IAAC, using a user-guided dashboard to browse through the solution space provided by the AI (Jastrzebska Aleksandra, 2020).

Chapter 3 Leading Design Principles

In this chapter, the leading design principles for 3D concrete printed bridges are discussed. The principles are based on the unique advantages and limitations of 3DCP. These principles have driven the development of the models to come and will referred to throughout this thesis. First, the advantages and disadvantages of 3DCP will be summed after which they will be used to establish the design principles:

Advantages:

- **[A] Geometrical freedom:** 3DCP offers full design freedom on a 2D plane, allowing the creation of any desirable shape. In the third dimension, however, the freedom is restricted with feasible angles up to approximately 80 degrees.
- **[B] Reduced material usage:** material usage can be optimized and less waste on the construction site is produced.
- **[C] Speed and efficiency:** modular construction and 24/7 fabrication increase construction efficiency.
- **[D] Cost-effective:** reducing labour requirements, material optimization and increased production times cause 3DCP to be a cost-effective alternative to conventional methods.
- **[E] Mass customization:** Each design can be unique tailored to its situation .

Limitations:

- **[F] Limited element size:** the size of individual components is limited either due to constraints of the printer size or transportation restrictions.
- **[G] Reinforcement techniques:** Current reinforcement strategies are in development, limiting the overall strength in tension or bending.

Which results in the following design principles:

- **[P1] Geometric focus :** built upon A. The geometrical freedom in the third direction is limited, therefore the analysis will focus on exploiting geometries in a 2D plane. Figures 3.1-3.4 shows that most geometrical freedom is applied in this plane. A constant unit width is assumed for each bridge.
- **[P2] Section size optimization:** Using the benefits of reduced material usage, each section's size is tailored for optimization.
- **[P3] Modularity:** Considering both the construction speed and the element size constraints [C], [F], the structure is assumed to be composed of at least two parts. In each calculation, the connection is assumed to be completely rigid.
- **[P4] Material behaviour:** Due to the limitation in reinforcement strategies [G], bi-linear elastic

unreinforced concrete will be assumed as material behaviour. For which generalized strengths of 30 N/mm² in compression and 3 N/mm² in tension is assumed.

- **[P5] Diverse topologies:** Relating the spectrum between geometrical freedom and mass customization [A], [E], various design domains will be explored and optimized yielding in a diverse array of topologies.

These design principles lay the foundation for the Design Space Exploration of 3DCP bridges. The parameters applied in the exploration tool should adhere to these principles to find feasible topologies meeting the manufacturing constraints, while also being able to analyze the impact of these parameters.



Figures 3.1 -3.4: 3D concrete printed bridges. All bridges show most shape variation within the 2D plane used in this thesis. From top to bottom: (F. P. Bos et al., 2020; de la Fuente et al., 2022; Kinomura et al., 2020; Vantyghem et al., 2020)

Chapter 4 Methodology

This chapter provides an overview of the methodology applied to explore the design space of 3DCP structures. First, the general workflow will be elaborated in which a generative design approach is being applied. This thesis generally comprises of two distinctive steps, the generation of the design space and the development of the user-guided navigation tool, which are tied together with a generative design approach, forming a third step.

- 4.1 Generative design principle
- 5 bridge design by optimization
- 6 Development of the dashboard

The workflow used to create the design space is visualized on the image on the next page, the three large squares adhere to the division described above. This chapter describes the guiding process behind each of the three components. Due to the complexity of optimization and dashboard, separate chapters, 5 and 6, will discuss these elements in more detail.

4.1 Workflow

The workflow of this thesis is presented in figure 4.2, and is globally chunked into three separated modules. These three modules contain of the generative design, the optimization, and the exploration, similar to the chapter description written above.

The first chunk covers the generative design process conducted in Grasshopper. This generative design module starts with a design idea. The design principles stated in Chapter 3 have been included as the design idea driving the workflow, denoted with the hexagonal shape. This is further elaborated into a design shape, which in this case is the parametric development of 3D concrete printed bridges. This occurs in Grasshopper where the user creates a parametric design domain with varying boundary shapes, resembling to the work of Caitlin Mueller visualized in image 4.1. The specific design domains and parameters influencing a single bridge will be further elaborated on in Chapter 5.1. A generative design approach is adopted, turning the parametric bridge into a design space. Each variation of the ground structure is cross-referenced with the parameters used in the optimization and calculation, leading to the variety of bridges making up the design space. The current application of RhinoWIP does only allow lists as input data into the component. Therefore, each bridge is generated, calculated and optimized separately.

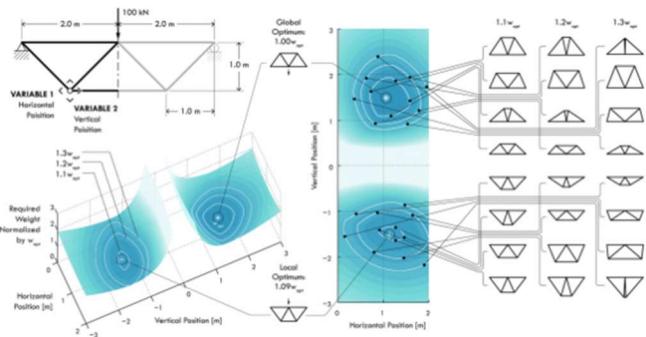


Figure 4.1 Design Space Exploration by (C. T. Mueller, 2014)

The second module, in figure 4.2, develops the ground structure into an actual topology. The ground structure created from a single iteration of the generative tool is used as input into Rhinocode, a python3 scripting environment within RhinoWIP, further described in Chapter 4.2. This script calculates the structural properties based on additional input, such as the initial guessed section size and material properties. The section sizes will always be homogeneously initialized. A heuristic optimization, based on the research of (Nefs, 2019), further explained in chapter 5.3, will then redistribute the initial material applied to the structure, resulting into a sizing optimization, from which the final topology follows. The optimization can be formulated as stated by equation (1).

Where:

- i = Index for element operations within the structure ($1 \leq i \leq n$)
- n = Amount of elements (or lines) in a structure.
- j = Index for parameters used to create different structures ($1 \leq j \leq k$) (note each j describes a unique set of parameters making up a bridge variant in the design space)
- k = Amount of bridges in the design space

$$\min \sum_{i=1}^n U_i(P_j, D_j), \quad \forall (1 \leq j \leq k)$$

Subject to: $\sum_{i=1}^n x_i L_i = V_{max}$

(1)

Where

- U_i = Strain energy of element i [J/mm^3]
- x_i = Sectional Height of element i [mm]
- L_i = Length of each element i [mm]
- V_{max} = Maximum volume constraint [mm^3]
- $P_j \in P$ = Optimization variables for structure j
- $D_j \in D$ = Design domain variables for structure j

A volume constraint is being applied to the structure, because the optimization is a redistribution of material, as chapter 5.3 will elaborate on. After finding the topology, a second optimization will change the total volume, in which each element is being assigned a percentage of the total volume. The optimization optimizes the total volume of the structure until a deflection constraint is met. This results in

the mathematical optimization formulation stated in equation (2), these principles are further explained in chapter 5.4.5.

$$\min \sum_{i=1}^n x_{ratio_i} L_i(P_j, D_j), \quad \forall (1 \leq j \leq k)$$

$$Subject to: \begin{cases} x_{ratio_i} = \frac{V_{max}}{x_i L_i(b)} \quad \forall (1 \leq i \leq n) \\ \delta_{max} = \delta_{allow} \end{cases} \quad (2)$$

Where

x_{ratio_i} = Percentage of the totale volume each element uses
[-]

δ_{max} = Maximum occurring deflection in the structure [mm]

δ_{allow} = Maximum allowable delfection set by user [mm]

(b) = Width

The data of a single bridge is then brought back into the grasshopper environment, module 1, where the data of each iteration is being stored, denoted with the triangular shaped icon in image 4.2. By doing these optimizations for every possible combination of P and D, a variety of topologies is being created which are referred to as the design space. After which the complete database will be written into a JSON file.

The final module describes the workflow leading up to the use of the navigation tool. Elaborated upon in chapter 6, the resulting database will be used as input for the design space exploration tool, created in Dash a Python library. Dash is chosen because it is built upon the graphical library of Plotly, meaning that it automatically integrates with the interactive plots created with this library. By separating the calculations from the exploration tool, a more generic use of the exploration tool is achieved. The exploration tool could be used for any dataset and is thereby not limited to the specific case of this thesis. The tool allows to explore the dataset by providing interactive graphs for filtering, setting constraints and analyzing the data. The goal is that the user can define its own objective and constraints to find the desirable bridge, resulting in the third optimization formulation, shown below in equation (3).

$$\min O_p$$

$$Subject to C_p(P, D) \quad (3)$$

Where

$O_p \in O$ = User defined objective function could be (a combination of): Volume, support forces, deflection

$C_p \in C$ = User defined Constraints (a combination of): Any variable from P or D

Rhino/Grasshopper:

The RhinoWIP (Work-In-Progress) version was selected primarily for its introduction of RhinoCode, which enables the utilization of Python3 directly within Grasshopper. While RhinoWIP brings several advantages to the project, it is worth noting that it is still under development, which led to certain limitations. Initially, these limitations impacted the use of Python libraries and prevented the integration of specific Grasshopper modules such as Karamba3D. The Grasshopper environment has been used as framework for creating the database and making up the design space. Grasshopper serves as platform for controlling the generative process and manipulating the parameters essential to development of the bridges.

Python3:

Python3 is used on multiple facets of this thesis. Inside the RhinoCode modulus is applied for the FEA-analysis and optimization of bridges. Python3 is also employed for data extraction from Grasshopper, facilitating the data flow iinto a JSON file. Furthermore, PyCharm was selected as the primary environment for coding, enabling the development and local deployment of the dashboard.

Overview of the libraries used:

- **Numpy:** This library is the main reason for switching to RhinoWIP since earlier versions of rhino only included Ironpython. The Numpy libraries are instrumental for the Finite Elements Analysis and optimization process.
- **Pandas:** For efficient data management, Pandas is employed when storing and importing datasets.
- **Scipy:** The Scipy Library is used for smaller optimization strategies applied throughout the thesis, for example in the Smooth penalization principle, chapter 5.4.3, and the volume optimization of each bridge, chapter 5.4.5.
- **SkLearn:** This machine learning library, is utilized for generating the t-SNE plot employed within the dashboard.
- **Plotly graphs object:** The main functionality of the dashboard comprises out of interactive plots. These are facilitated by this library.
- **Plotly Dash:** The dashboard framework is constructed with the Plotly Dash library. Dash seamlessly interacts with the plotly graphs object, making the dashboard very suitable for integrating interactive plots.

4.2 Implementation Details

In this thesis a variety of technologies are used in the development and execution of the research. This subchapter offers a comprehensive overview of the key technologies and tools used in the study.

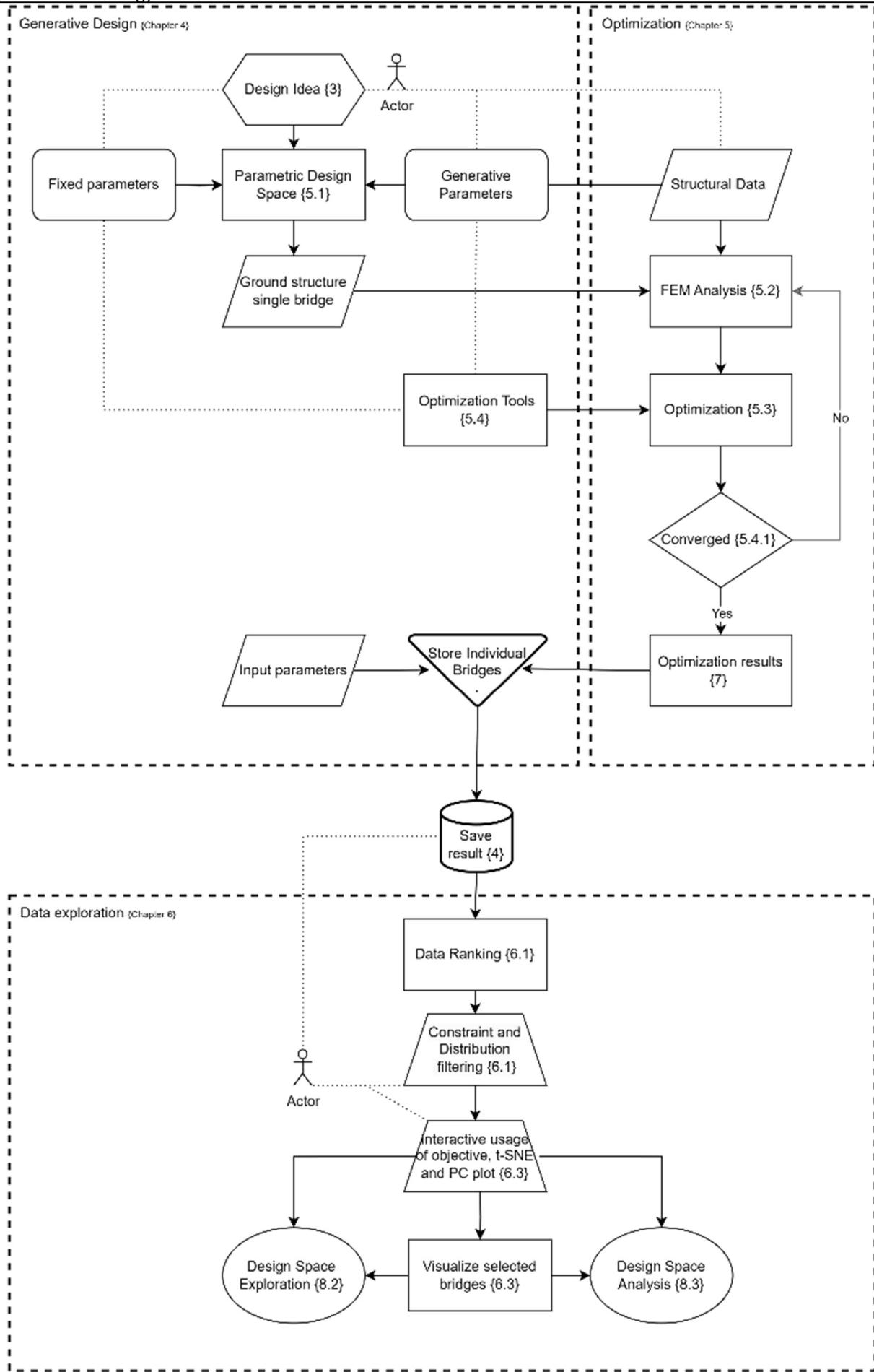


Figure 4.2 Schematization of the workflow described in Subchapter 4.1

Chapter 5 Creating The Design Space

This chapter will follow the manipulations conducted on a single bridge. The parameters influencing each bridge, which will be used for the generative creation of the database, will be discussed and shown. The parametric model resulting in an optimized bridge is based on the design principles described in Chapter 3. The text will therefore refer back to the relevant principles. Each section ends with a table providing an overview of the involved parameters, where a distinction is made between generative parameters, which make up the design space, and ‘fixed’ parameters which do not contribute towards variation within the design space. The optimization discussed in Chapter 5.3 has several additional functionalities that will be described within its subchapters. The following steps are described:

- 5.1 Generating the Ground Structure**
- 5.2 Finite Elements Calculation**
- 5.3 Ground Structure Optimization**
- 5.4 Optimization Tools**
 - 5.4.1 Convergence
 - 5.4.2 Step Size Damping
 - 5.4.3 Young’s Modulus Penalization
 - 5.4.4 Construction Constraint
 - 5.4.5 Optimize The Maximum Volume
- 5.5 Visualization of a Parametric Tree**

By examining the creation of a single bridge and describing the underlying optimization techniques. an example is set for the construction of the entire database used for design space exploration. The techniques described in this chapter are validated in Chapter 7, Numerical Examples.

5.1 Generating the Ground Structure

Before creating the ground structure, a parametric case has to be developed, in this case, specified on the bridge design. First, a bounding box is created, constituting the minimum and maximum height allowed to be used for spanning the loading path. The primary objective of the generative ground structure is to manipulate the height of the bounding box, or domain, at three points on one side of the bridge, mirroring the changes on the opposite side. By creating the ground structure in this manner, with a relatively low subdivision, each bridge is ‘forced’ into a certain shape. This creates a large variety of topological solutions to be included in the design space, which makes this step comply with the design principle [P5]. The adapted bounding box is subdivided into a set of points, which can be seen in image 5.1. The next step in the process is also visualized in the same image, namely a line model is created out of the set of points. The parameter controlling the subdivisions is fixed, because changing the subdivision does not lead to different topological solutions but only refines the solutions at hand. The points are then used to create a grid of lines. A linear mesh instead of a mesh with surfaces, or topology optimization, is applied because the linear mesh more naturally follows the manufacturing process of 3DCP. Therefore, post-processing steps are assumed to involve less additional material consumption, complying with the principle [P2]. Additionally, the assumption is made that lines control the topology better than planes. Changing the boundary box of a mesh with planes will likely have less impact than doing so on a mesh with lines, complying with the principle [P5]. Two grids have been explored, one cross-referencing all points to create a line between every point [Option 1] and one creating a structured grid by creating only lines between neighbouring points [Option 2], these are also visualized in image 5.1. The first option creates a lot more lines and therefore increases the computation times, while the resulting topologies generally perform worse in the upcoming optimization. Therefore option 2 will be used throughout this thesis. This grid of lines is used as the starting topology in the

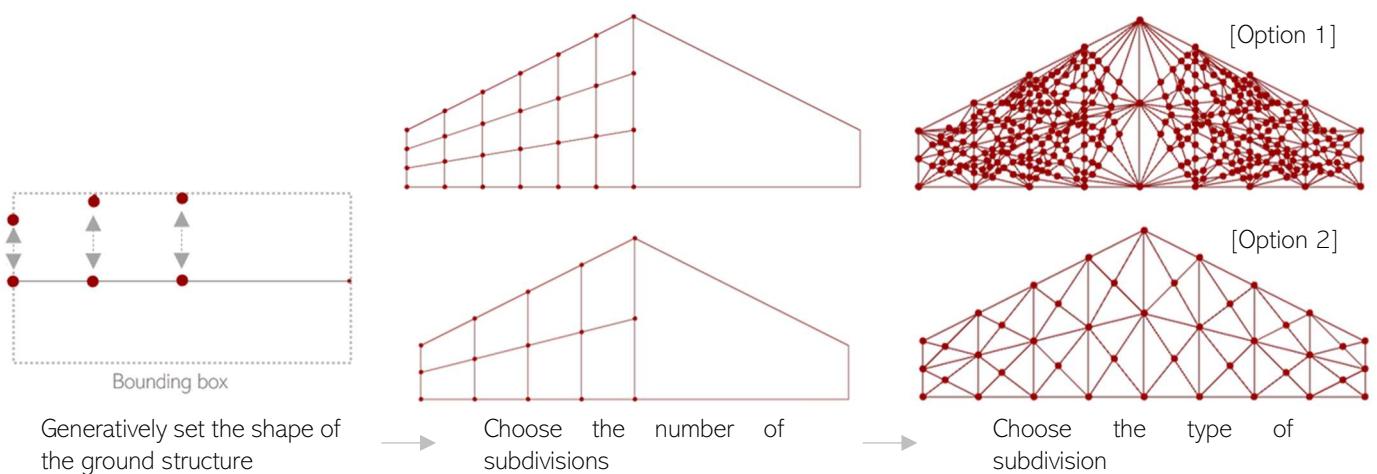


Figure 5.1 Parametric creation of the ground structure.

Chapter 5 Creating The Design Space

optimization and will therefore be called the ground structure.

Table 1 Parameters of the Ground Structure.

input	- Design Idea - Design domain heights List[float]
Output:	- Ground structure of a single bridge
Parameters:	- Bounding box size (design domain) [float] - Subdivision/step size [float] - Subdivision type [string] - Number of vertical points [float] - Step size between vertical points [float]
Generative parameters:	- Design Domain Heights 1,2 and 3 List[float]

5.2 Finite Elements Calculation

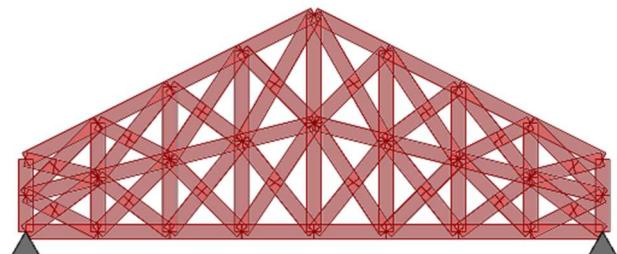
Finite element analysis (FEA) is used to analyze the ground structure generated in Grasshopper. The book “A First Course in Finite Elements” by Daryl Logan (Logan, 2007), is used as a guide for the calculations. The lines from the previous step are inserted from Grasshopper into the RhinoCode module, which is, at time of writing, only available in the WIP version of Rhino. Additionally, at the time of writing, Rhinocode does not allow any library to be inserted, this is limited by the developers of Rhino, therefore the well-known FEA code that is self-written in Python. Each line becomes a single element in the FEA model. These lines combined with material properties and an initial guess for each member, are converted into force distributions for each member. This is done using the direct stiffness method which is based on the relationship below:

$$\mathbf{F} + \mathbf{F}_{eq} = \mathbf{K} \mathbf{D} \quad (4)$$

Where

- F** = Force vector [N]
- F_{eq}** = Equivalent force vector [N]
- K** = Element stiffness matrix [N/mm]
- D** = Displacement vector [mm]

The use of the direct stiffness method is further elaborated and derived in Appendix B. It is possible to calculate the



Set initial member size to elements

structure as a truss, considering only axial forces, or as a beam, with axial, shear and bending forces in compliance with the design principle [P5]. This design option facilitates the construction of various types of bridges with varying nodal connections.

All calculations conducted will be linear elastic-The smooth Young's modulus penalization, explained in chapter 5.4.3, creates non-linear structural behaviour yet the underlying material properties are still assumed to be linear elastic concrete-, assuming unreinforced concrete, complying with the design principle [P4].

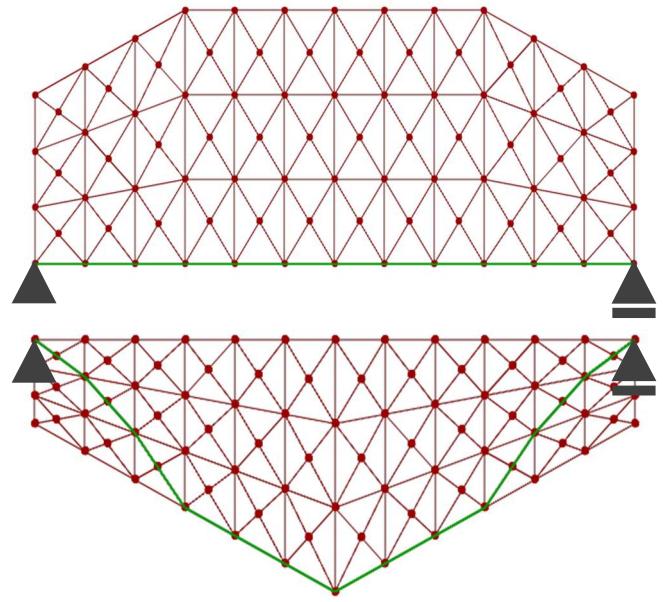
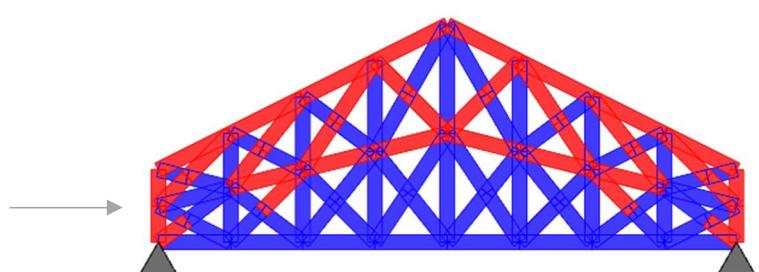


Figure 5.3 The prestress shown in green of two possible bridge shapes.

The model currently only allows one load case as input for analysis. It should be mentioned that single load cases are not representative of realistic structures, but this approach is adopted to decrease computation times, given the intended use of this tool for the early design stage. All loads are applied as resultant nodal end forces.

A single distributed load along the load path can be inserted by the user. Alternatively, a point load in the middle of the bridge at the loading path can be inserted.

The FEA model includes an option to account for gravity. Dead load is defined for each element based on the material properties set within Grasshopper. This load is switched on or off by a toggle.



Perform FEA calculations

Figure 5.2 Perform Finite Elements on bridge.

Prestress is applied to limit tensile forces and allows for changes in manufacturing conditions, complying with the design principle [P3]. The prestress is always applied from node 0 to the lowest node of domain parameter two. This design rule maximizes the counteractive force against the bending moment, for each generative alternative, while ensuring that the prestress will never be in a vertical line with the vertical support; Image 5.3 elaborates this further and provides options for prestress with structural material applied above the support line and material applied below it. The prestress load P is determined as described in equation 5. This load is then applied as an axial force vector to each prestressing element.

$$P = A_p F_{yp} \quad (5)$$

Where

- P = Prestress force [N]
- A_p = Area of prestressing cable [mm^2]
- F_{yp} = Yield strength of prestressing cable [N/mm^2]

In the model, the decision is made to allow for the insertion of a desired prestress area. The resulting prestress in each member is summed up into a prestress force vector. This vector is added to the other force vectors resulting into a change of equation (3) into:

$$\mathbf{F} + \mathbf{F}_{eq} + \mathbf{P}_{eq} = \mathbf{K} \mathbf{D} \quad (6)$$

Table 2 Parameters for finite elements

Input:	- Line model - Load case - Initial guess section sizes - Material properties
Output:	- Force distribution - Displacements
Parameters:	- Material properties (Linear elastic concrete) List[float] - Gravity [on/off] - Initial section sizes List[float] - Loadcase List[float]
Generative Parameters:	- Amount of prestress: radius of the prestressing cable [float] - Beam/Truss [string]

5.3 Ground Structure Optimization

After calculating the structural performance of each bridge, a ground structure optimization will be conducted. The ground structure approach is an optimization technique redistributing material throughout the structure in accordance with the energy present in the system (Bendsøe & Sigmund, 2004). Generally, it solves a topological design question as a sizing problem by minimizing the compliance, the inverse of stiffness, in the system and thereby maximizing the structural stiffness. The minimization of compliance is formulated as:

$$\begin{aligned} \min \mathbf{C} &= \mathbf{f}^T \mathbf{u} \\ \text{s.t. } \mathbf{K} \mathbf{u} &= \mathbf{f} \end{aligned} \quad (7)$$

Where

- \mathbf{C} = Compliance [mm/N]
- \mathbf{f}^T = Transposed force vector [N]
- \mathbf{u} = Displacement vector [mm]

Minimizing the compliance reduces the deformations under applied loads, which implies that it minimizes the potential energy stored in the system. Because a Linear elastic material behaviour is assumed [P4], all potential energy in the system is in the form of Strain energy. Additionally the ground structure problem is proven to be a convex optimization problem and existence of solutions is assured (Bendsøe & Sigmund, 2004). Therefore, optimization can be achieved through the heuristic minimisation of strain energy. The heuristic optimization can be unconstrained, as deflections affect optimal cross-sections but have no impact on optimal topologies, according to Bendsøe. Consequently, the optimization can be divided into two stages: first, finding the optimal topology given maximum volume constraints, and then determining the minimum volume required to satisfy deflection constraints. The optimization function is therefore described as equation (1) for the topological redistribution. However the heuristic application of this formula is being applied as derived by (Nefs, 2019) as:

$$A_i = \frac{\frac{U_i}{\sum_i U_i} V_{max}}{L_i} \quad (8)$$

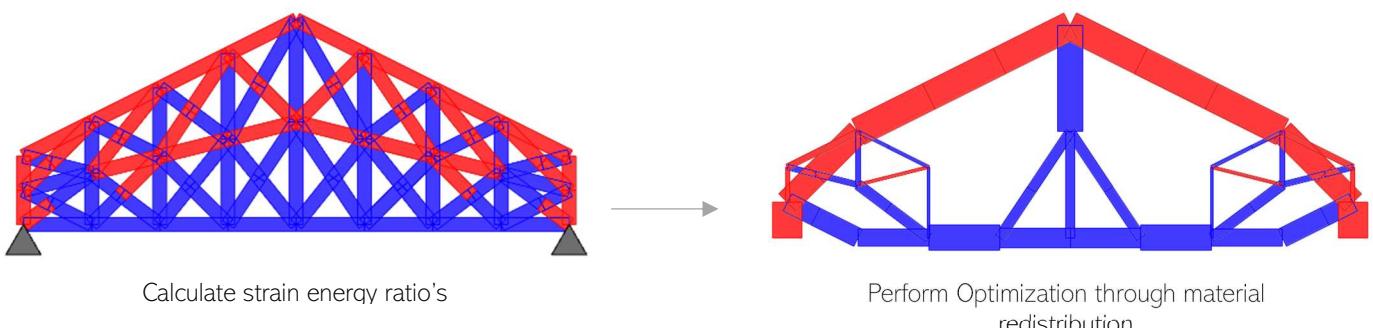


Figure 5.4 From FEA analysis to optimized ground structure.

Chapter 5 Creating The Design Space

Where

A_i	= Section of the element [mm ²]
U_i	= Strain energy of the element [J/mm ³]
$\sum_i U_i$	= Total strain energy [J/mm ³]
V_{max}	= The total volume constraint [mm ³]
L_i	= Length of the member [mm]

Table 3 Optimization parameters

Input	- FEA output {5.2} - Settings optimization tools {Chapter 5.4}
Output:	- Sectional heights for each element
Parameters:	Settings optimization tools; - Convergence [on/of] - Convergence tolerance rate [float] - Step size damping factor [float] - Optimize vmax [on/off] - Young's modulus penalization [on/off] - Nef's penalization factor [float] - Forced members [on/off] - Threshold volume [float]
Generative parameters:	- Young's modulus penalization type [string] - Construction constraint location [string]

The method for computing the strain energies is available in Appendix C. As the optimization of the 2D bridge is under consideration, the out-of-plane dimension (i.e. the cross-sectional width) remains constant for all members. Therefore, the structural height, denoted as, can be directly optimized with the GS approach, adhering to design principle [P1]. The width can be excluded from the equation as it is uniform for all members, and therefore, does not impact the material redistribution.

$$x_i = \frac{\frac{U_i}{\sum_i U_i} V_{max}}{L_i (b)} \quad (9)$$

Where

x_i	= Cross sectional height of an element [mm]
b	= Width [mm]

Image 5.4 shows the effect of the heuristic optimization of the bridge case at hand. The image on the left shows the result from the FEA analysis, whereas the image on the right depicts the resulting material redistribution from the optimization.

The Optimization does not take into account stability and could produce kinematically unstable structures. To prevent the optimization from getting stuck by these issues posed, a minimum volume constraint is being added to the optimization, as described in equation 9. This prevents singularities while calculating. This implies that no sectional volume can be reduced to zero. Therefore, a post-processing

step is being added to the optimization, which removes all lines with a volume lower than the predefined threshold value.

$$x_i l_i \geq V_{min} \quad (10)$$

Where

V_{min}	= Minimum volume for each member [mm ³]
-----------	---

5.4 Optimization Tools

In order to improve the functionality of the optimization, multiple tools have been added to it. Which in turn have their variables allowing for inspection by the DSE tool, these will be further elaborated upon in the upcoming chapter. By incorporating these elements into the optimization process, the optimization becomes more versatile and capable of exploring various design possibilities and optimizing the structural performance based on different (manufacturing) performance criteria.

5.4.1 Convergence

To ensure a proper convergence of the optimization process, it is essential to establish stopping criteria. This model incorporates two methods for determining when the optimization should stop. First, an upper bound is set to the maximum number of iterations. It could be possible that a certain bridge generated for the database has an extremely undesirable shape causing the optimizer to have trouble with converging. To address this, an upper bound can be set to any desirable. The second method considers the change in strain energy between successive iterations. If the difference in strain energy between two iterations falls below a certain percentage of the strain energy, known as the strain tolerance rate, the optimizer will be terminated at that particular iteration. The specific percentage can be adjusted to any desirable level of convergence. These criteria can be noted as:

$$n \geq n_max \text{ or } |x_n - x_{n-1}| \leq \left| \frac{x_n}{c} \right| \quad (11)$$

Where

C	= The stopping criteria (100 as initial number) [-]
n	= Number of iterations [-]
x_n	= Strain energy at one iteration [J/m ³]
x_{n-1}	= Strain energy at previous iteration [J/m ³]

5.4.2 Step Size Damping

The heuristic optimization can be sensitive to state switching. In which it keeps switching between two local optima between iterations. The global optima will then lie somewhere in between of these local optima. This state switching can have several causes, such as specific combinations of shape, loading condition and the initial guess, resulting in peak stresses and corresponding strain energies. Consequently, excessive material allocation occurs

5.4.3 Young's Modulus Penalization

Given the difference in strength in tension and compression for concrete, it is favorable for the structures to optimize towards a configuration predominantly in compression. The optimization strategy is based on the minimization of strain energy. Therefore, a decrease of the young's modulus of the elements in tension implies a decrease in stiffness for these elements, making it less favorable to distribute material to these elements for the optimizer. Because tensile elements transfer forces less efficiently than compressive elements meaning that minimizing the compliance will result in maximizing the compressive elements.

Linear Penalization

For many Linear elastic concrete applications, a bi-linear stress strain curve is applied as shown in figure 5.7 by (Nefs, 2019). Nefs proposes to adapt the bi-linear stress relation to create similar strain energy density surfaces for the compressive and tensile parts. Which practically results in reduced Young's modulus of square root 10 for the elements in tension, visualized in the same graph with the grey areas. This linear Young's modulus penalization can be formulated as shown in equations 13,14.

$$E_{pf_Nefs} = \frac{E_{cm}}{PF} \quad (13)$$

Where,

E_{pf_Nefs} = Penalized youngs modulus by Nefs [N/mm^2]
 E_{cm} = Young's modulus of concrete [N/mm^2]
 PF = Penalization factor [-]

With

$$PF = \sqrt{10} \text{ for } \sigma_i > 0 \quad (14)$$

Where

σ_i = Stress of the element [N/mm^2]

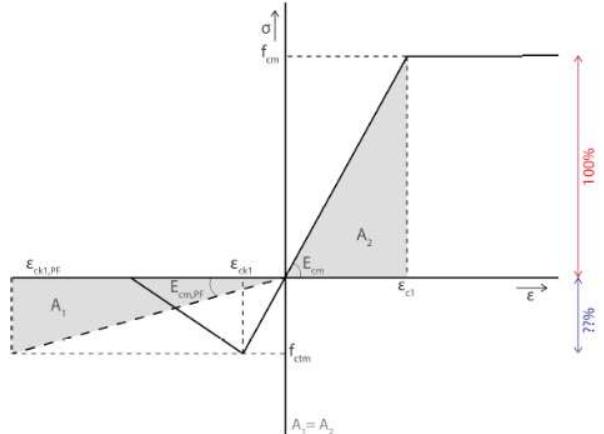


Figure 5.7 Adapted bi-linear concrete material behaviour by (Nefs, 2019).

Smooth Penalization

The nod at the origin of the material behaviour graph does, however, cause the optimization algorithm to be sensitive to state switching. For elements that are at the intersection of

at these locations, causing the next iteration to have insufficient material at another location, which in its turn again causes peaks at those locations. Thereby trapping the optimization to these local minima. This tool prevents state switching by applying a step size damper, which adds only a percentage of the change in calculated dimensions over the iterations instead of the full difference. This causes the two states to slowly converge into a single solution. An example of the application of step-size damping is visualized in images 5.5 and 5.6. These images show the same bridge without damping and two values of damping. The two graphs show the same convergence but with different scales. The plot shows that both values of damping resolve the state switching and that the value of damping influences the accuracy of the result and with that the convergence times. A high damping factor results in fast convergence with less accuracy.

$$x_{i,damped} = x_{i-1} + \frac{x_i + x_{i-1}}{x_{i-1}}(1 - \zeta) \quad (12)$$

Where

$x_{i,damped}$ = Damped sectional height [mm]

x_{i-1} = Sectional height from previous iteration [mm]

x_i = Calculated sectional height from current iteration [mm]

ζ = Dampfactor [-]

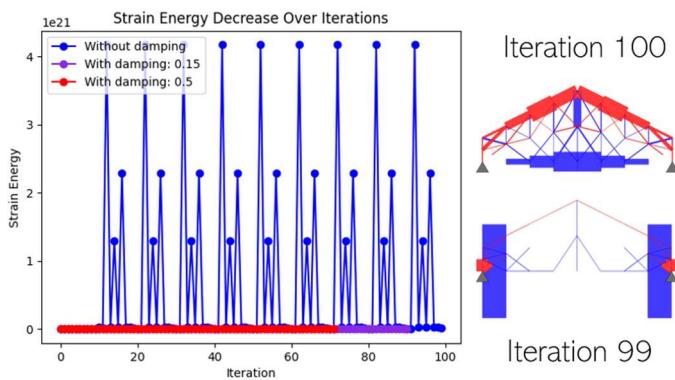


Figure 5.5 Convergence of optimization subjected to state switching, in blue. The images to the right depict the optimization results at two successive iterations.

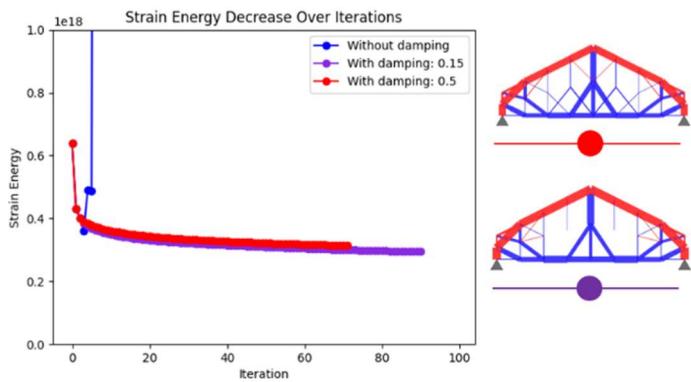


Figure 5.6 Convergence of the optimization with stepsize damping. Two different damping values are provided and visualized.

Chapter 5 Creating The Design Space

being in-between different axial stress states the change in young's modulus could cause a change in force orientation, causing to trap the optimization. A smooth curve would be preferred to resolve this issue. This can be achieved by applying a sigmoid function which is often used for smooth optimization principles and therefore applied for this specific case. The general shape of the sigmoid function is described as:

$$S(x)_{standard} = \frac{1}{1 + e^x} \quad (15)$$

The bounds of this function are set by the material yield properties. This implies that the standard sigmoid equation, 12, can be changed into equation 13.

$$S(x) = -\sigma_T + \frac{\sigma_c + \sigma_T}{1 + e^{S(x-b)}} \quad (16)$$

Where

- σ_T = Yield stress in tension [N/mm²]
- σ_c = Yield stress in compression [N/mm²]
- S = Parameter controlling steepness of the curve [-]
- b = Parameter determining horizontal offset [-]

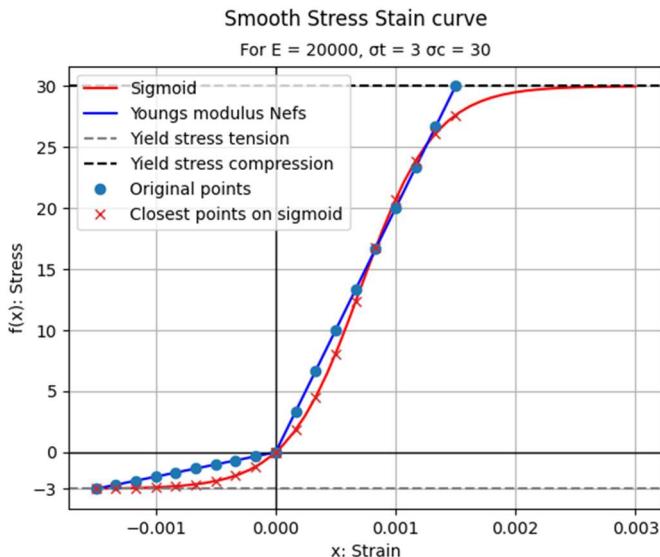


Figure 5.8 Smooth Young's modulus curve used for penalization.

In order to replicate Nefs' Young's modulus penalization, it is necessary to choose S and b values in a way that area A1 is equivalent to A2, as shown in Figure 5.7. This should be done while taking into account the stress parameters of concrete in compression and tension, and also passing through points 0,0. By setting $S(0) = 0$, a connection between S and b can be established

$$S(0) = 0 \rightarrow S = -\frac{\ln(\frac{\sigma_c}{\sigma_T})}{b} \quad (17)$$

With the relation $A_1 = A_2$ the integral of $S(x)$ below the x-axis should be equal to the integral of the function above the x-axis for the given axial stress values. This function does, however, not have a solution, because the areas A1 and A2 will never be equal given the setup described above.

Therefore, S or b will have to be calculated using a minimization function minimizing the distance between the two graphs, to maximize the similarity in behaviour, with in this case S as a parameter. The iterative process involves varying the S parameter, calculating the y location for both graphs given an input of x coordinate, and determining the difference between the y coordinates on both graphs as the objective. The standard SciPy minimization is used with the objective function described above to find this value of S. This process is performed outside of the RhinoCode module to compute and optimize the structure. This dependency is founded on fixed material properties, meaning that optimization takes place only once. After determining the curve parameters, Young's modulus can be calculated at any point on the curve by taking the secant of the respective point with respect to the origin.

$$E_{sigmoid} = \frac{S(x) - y_2(0)}{x - x_2(0)} \quad (18)$$

However, since the shape of the sigmoid function is not exact to that of the initially applied Young's modulus, the nearest point on the sigmoid curve will have to be calculated. This to switch from the normal Young's modulus to the sigmoid penalized Young's modulus. The nearest point is calculated with the assumption that every possible point on a sigmoid curve is one of:

$$x = x, y = S(x) \quad (19)$$

The distance between an arbitrary point (a,c) on the original Young's modulus curve and the sigmoid curve results in equation 17.

$$f_{distance}(x) = \sqrt{(x - a)^2 + (S(x) - c)^2} \quad (20)$$

The minimum distance is where the derivative of $f_{distance}(a, c)$ is equal to 0, which is described in equation 18.

$$\frac{df_{distance}(x)}{dx} = \frac{-a + (S(x) - c)S'(x) + x}{\sqrt{(a - x)^2 + (c - S(x))^2}} \quad (21)$$

This formula does not have any exact solutions therefore the minimization function of the SciPy library is applied to approximate the root of this function. Several methods in the library were tested on accuracy and speed. Eventually the "TNC" method was chosen due to its performance in terms of speed, with no significant decrease in accuracy. The resulting smooth Young's modulus graph is shown in image 5.8

The found value of x can then be inserted into the original sigmoid function S(x) to find the corresponding y on the sigmoid function. After which the young's modulus can be calculated by taking the Secant of this point with respect to the origin, as described in equation (18). The effect of the two penalization methods is visualized in image 5.9. The

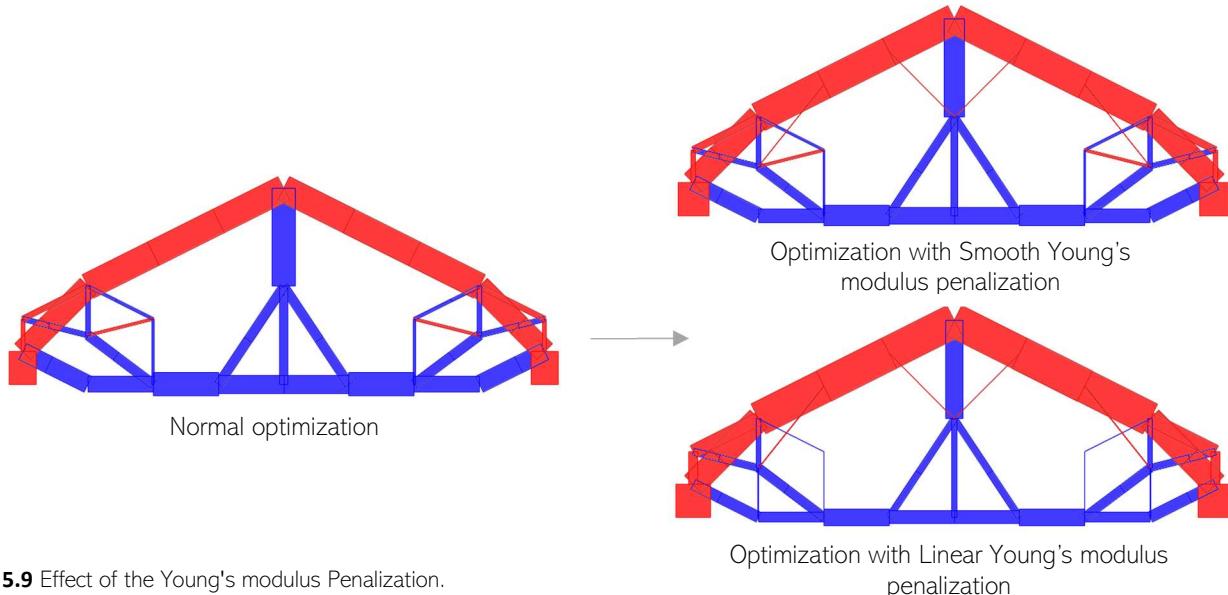


Figure 5.9 Effect of the Young's modulus Penalization.

image shows the optimization without penalization on the left. On the right the two penalized methods are shown with the image on top visualizing the linear penalization and the one below the smooth penalization.

5.4.4 Construction Constraint

In order to reach optimal designs adhering to the manufacturing constraints, a new optimization tool is introduced. The tool addresses situations where the optimization process removes material from critical locations in the manufacturing process, think of connections. Post-processing could reintroduce the removed material, but this material would then not contribute towards the objective. To tackle this problem, the construction constraint tool is being implemented complying with design principle [P3]. To use this tool the user has to predefine the member index of the elements that should adhere to the desired minimum section dimension. This is achieved by adding a fictive strain energy to the elements which exceed the minimal set section size. Based on the formula used for the ground structure optimization (5), the minimum required strain energy can be calculated, by setting the section height equal to the desired minimum height threshold.

$$x_i < \text{Threshold} \quad (22)$$

$$U_{\text{Threshold}} < \frac{\text{Threshold } \sum_i (\mathbf{U}_i) L_i}{V_{\max}} \quad (23)$$

Where

Threshold

= A minimum section height made up by the print width times the amount of print layers [mm]

$U_{\text{Threshold}}$

= Minimum strain energy for a member to stay above a sectional size complying with the set threshold [J/mm^3]

If the strain energy of the desired element is below this threshold value, it has to be updated to the threshold value. This, however, means that the total energy in the system increases, which is not in line with the principle of conservation of energy. Therefore, all non-affected elements will receive a reduced strain energy in order for the total energy in the system to be conserved. The reduction will be the summed difference between the strain energy threshold and the strain energy of the elements below the threshold. The optimization formula therefore changes into equation (24). Equation (25) describes the scalar factor that will be introduced to each non-affected cross-section.

$$x_i = \frac{\mathbf{U}_i U_{rs} V_{\max}}{\sum_i \mathbf{U}_i L_i} \quad (24)$$

$$U_{rs} = \frac{\sum (U_{\text{threshold}} - \mathbf{U}_{i,b})}{\sum_i \mathbf{U}_i} \quad (25)$$

Where

$\mathbf{U}_{i,b}$ = Strain energy of the elements that are below the threshold [J/mm^3]

U_{rs} = The reduction in strain energy applied to non-affected elements [J/mm^3]

Image 5.10 shows the effect of the construction constraint tool in which the upper example shows material being forced the loading path, located between the two supports. Whereas in the second option material is being compelled to vertical element in located in the middle between the two supports. The member indices of the relevant members are predefined into the optimization by user input from Grasshopper. In Grasshopper, four gates have been created that automatically index the right elements by using a value list. The options are: no elements, the load path, the middle vertical element and the elements on the design domain boundary. In each iteration the optimization checks the strain energy of these elements compared to the total strain energy in the system. If this ratio would yield a smaller cross-sectional dimension the lower boundary set to these elements, then the optimization pushes the strain energy of

Chapter 5 Creating The Design Space

these elements to the strain energy matching the lower bound material energy. The increase in total strain energy is monitored after which all non-affected members receive a strain energy reduction.

5.4.5 Optimize The Maximum Volume

After finding the optimum material distribution the model will continue to optimize the structure by finding the optimal value for the maximum volume. This aligns with Bendsøe's methodology of first finding a topology and then optimizing it towards a second objective, in this case, the volume. Since the DSE tool is of specific interest in the early design stage an optimization towards deflection will be conducted and the structures will therefore not be optimized towards stresses. Optimizing all bridges to the same deflection normalizes them allowing to compare them on multiple factors, such as volume, stress utilization and support forces, instead of comparing them on their difference in deflection. Resulting in the optimization formulation described in equation (2), which is repeated below for convenience.

$$\begin{aligned} \min V_{\max} \\ \text{s.t. } \sum x_i l_i(b) = V_{\max}, \delta_{\max} \leq \delta_{\text{allowable}} \end{aligned} \quad (2)$$

Where :

- V_{\max} = Maximum applied volume [mm³]
- δ_{\max} = Maximum occurring deflection [mm]
- $\delta_{\text{allowable}}$ = Maximum allowable deflection [mm]

Next to this optimization function, two constraints are added. First, the Sum of the bar volumes should always be equal to V_{\max} , which is satisfied by using the strain energy ratio resulting in the sectional heights x_i , given a certain volume. Secondly, the maximum occurring deflection should be equal to the maximum allowable deflection. Since there is only one objective and a single parameter to be tuned, this optimization can also be seen as a root-finding problem. In this root-finding problem, the calculated allowable deflection minus the maximum allowable deflection should

be equal to zero. An advantage of this approach is that each solution will automatically adhere to the second constraint.

The root finding problem is challenging since the constraint is dependent on the sectional heights x , which are themselves reliant on the value of the maximum volume, V_{\max} to be minimized. As a result, this optimization function can be expressed as follows:

$$\min v_{\max} \quad (26)$$

With

$$f(x) = \delta_{\max}$$

$$\delta_{\text{allowable}} = \frac{L_{\text{span}}}{800}$$

Therefore

$$f(v_{\max}) - \frac{L_{\text{span}}}{800} = 0 \quad (27)$$

Where

$f(x)$ = Calculation to determine the deflection

5.5 Visualization of a Parametricized Tree

By changing the parametric values of the generative parameters described in this section, a large variety of bridge can be created. These parameters in combination with design domain parameters generating the different ground structure, make up the design space to be analyzed. This subchapter contains a parametricized tree, showing how the generative optimization parameters influence the bridge designs. The trees thereby show how equations (1) and (2) are influenced by different settings of P , the optimization parameters. Two trees are shown, thereby visualizing the effect of a set of P 's on two Different D 's, the design domain variables. Note that the second tree is made with a subdivision of three whereas the first is made with a subdivision of two. This to show intended effect of this parameter, which is not used to generate varying topologies but instead can be preset as desired by the user.

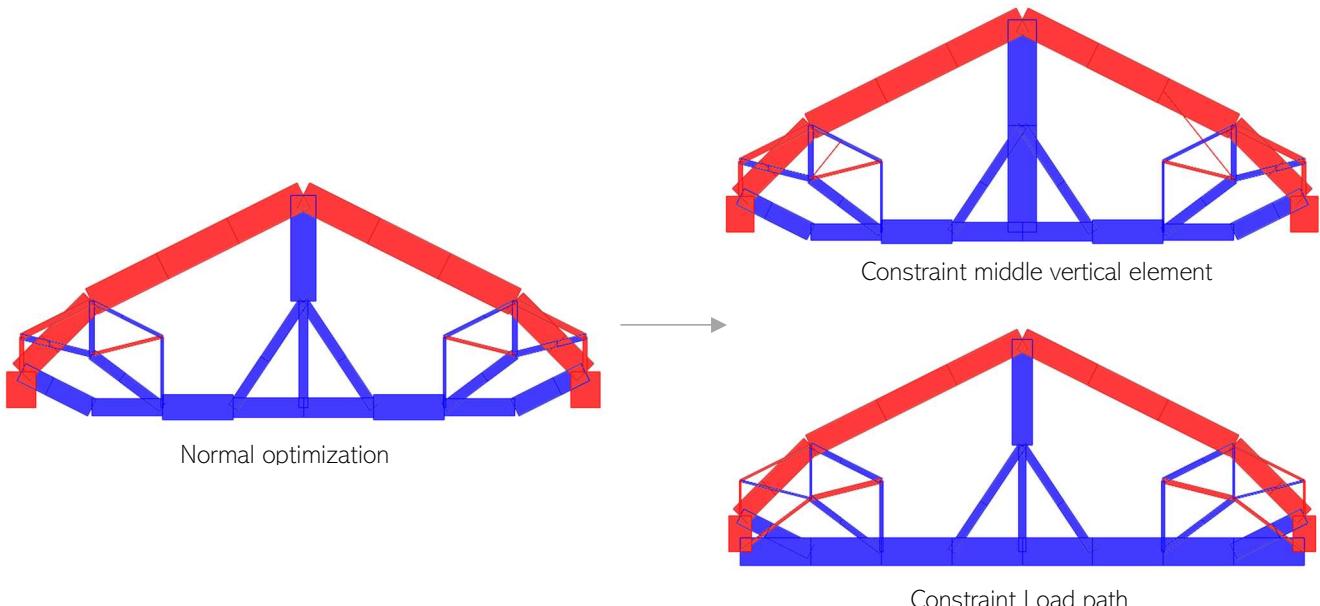
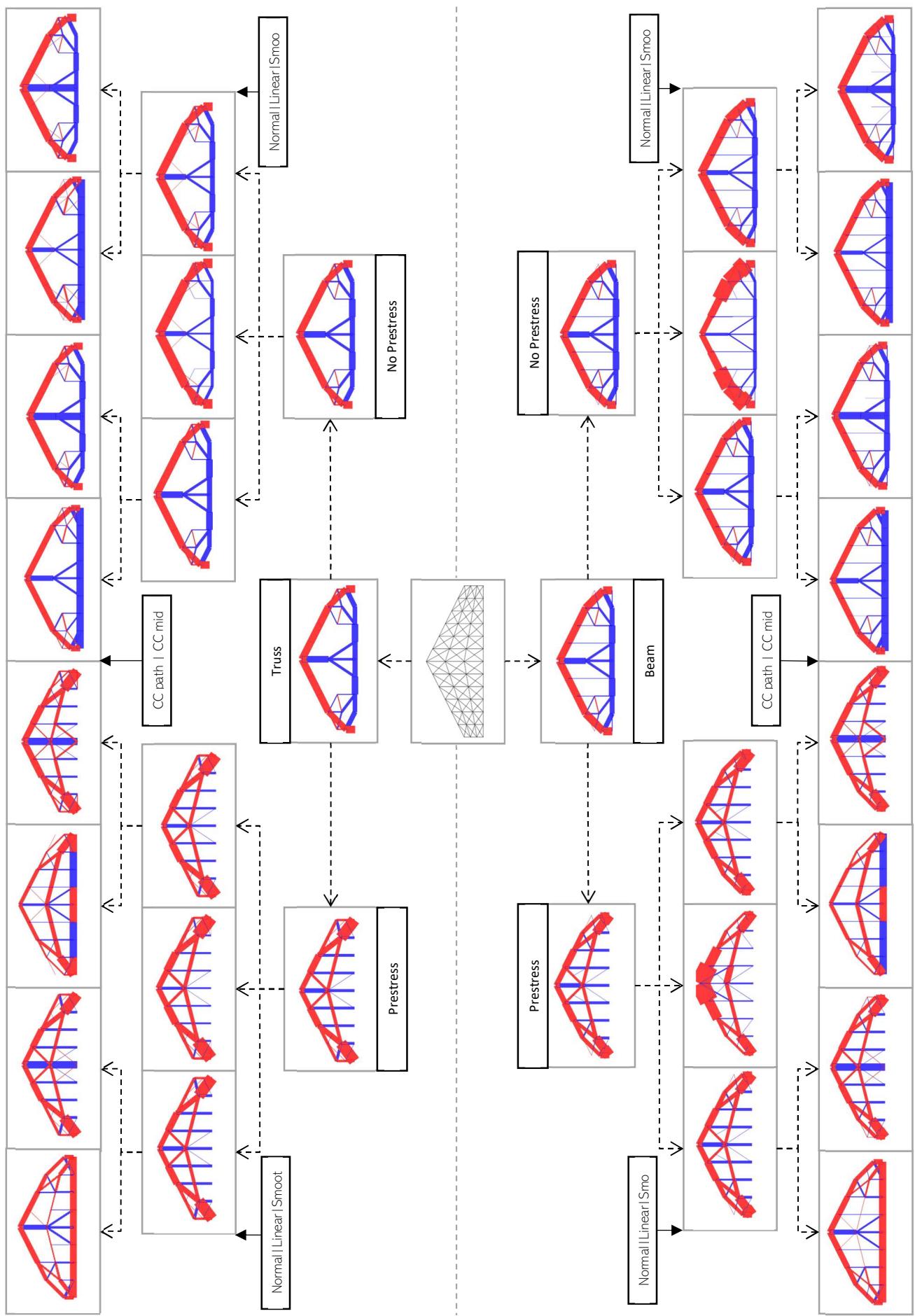
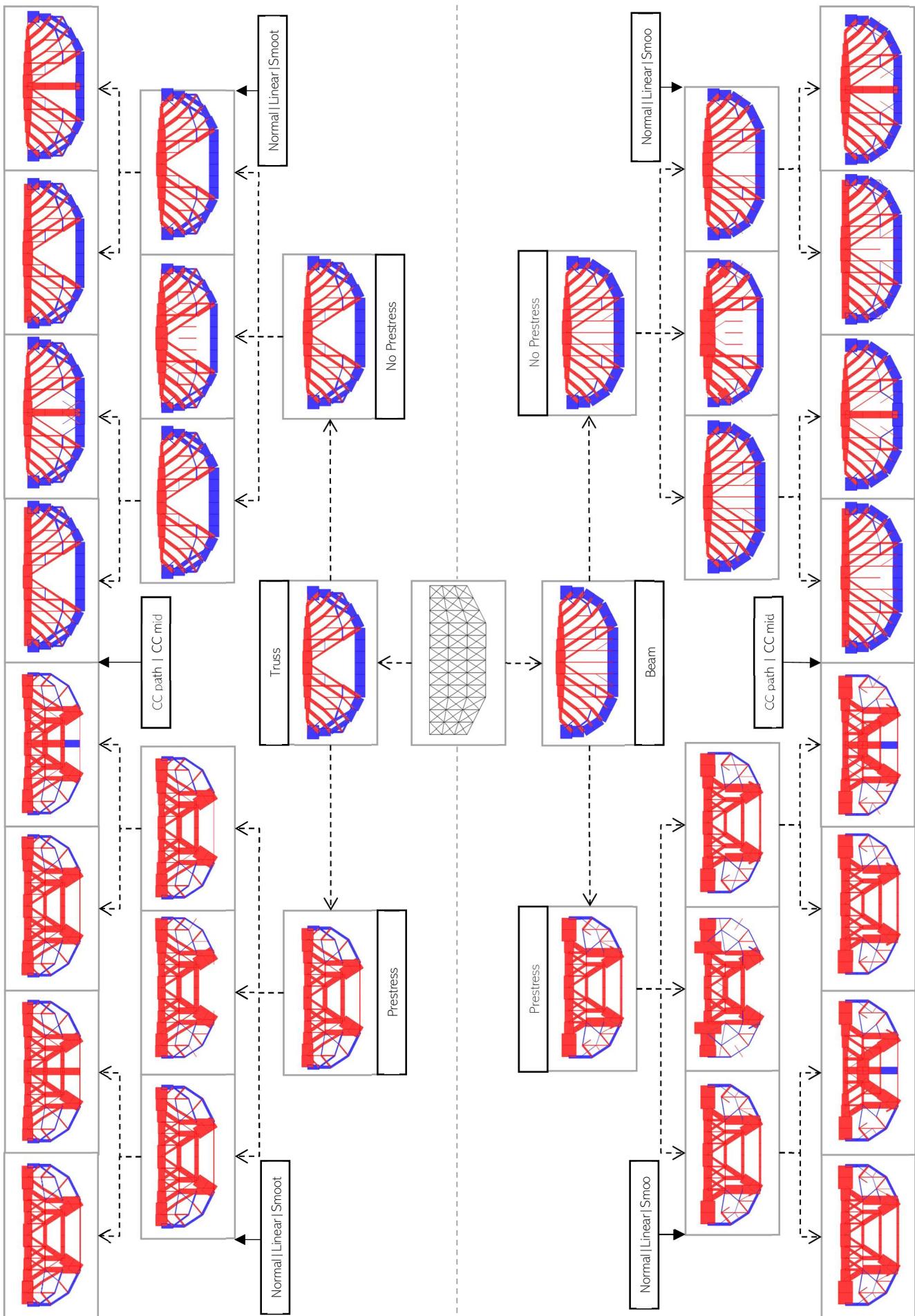


Figure 5.10 Influence of Construction constraint tool.





Chapter 6 Design Space Exploration Dashboard

The database, once established, will be navigated through a user-guided Design Space Exploration approach. A user-guided approach is being implemented because the perceived insights are supposed to be the largest. According to the steps necessary for successful DSE, the dashboard should provide tools to be able to navigate through the solution space, examine the design space and finally visualize desired solutions and allow for comparison between those solutions. The dashboard should thus be able to navigate, examine and visualize. The dashboard is created in Dash. Dash integrates with Plotly's built-in graphing libraries allowing for the implementation of interactive graphing features. One example of a "dash" interface for DSE is already provided in the literature research on aeroplane wings(Peter Sharpe, 2021).

The DSE tool has five main components:

- **Graph 1:** A feature to set domain constraints.
- **Graph 2:** A feature to filter the database based on the desired maximum volume.
- **Graph 3:** A method to locate the preferred bridge based on the visualization of an objective space.
- **Graph 4:** A graph to explore data structure using dimensionality reduction techniques.
- **Graph 5:** A parallel categories plot for exploring relationships between input and output parameters.
- **Graph 6:** A tool to visualize the selected bridge and show its corresponding performance .

In the overall design of the design board, the design was made to separate sliders from graphing features, for the sake of clarity. Sliders are therefore shown to the left of a block containing a graph. The dashboard is being visualized, in parts, on the upcoming pages, images 6.2, 6.3, 6.4 and 6.5. However, a general overview of the dashboard is presented in Appendix G. The red-blue scale in graphs is chosen, because opacity reduction visualization techniques are applied to the marker, implying a colour scale including light colours could be seen as opacity-reduced colours.

Interaction between graphs is enabled by callback functions that are triggered by changes in sliders or graphs. These callback functions then modify the corresponding graphs or dataset. A schematization of these callbacks can be found in Appendix F.

6.1 From dataset to design space

The generated dataset needs to be manipulated before being able to analyze and find desired bridges from that

dataset. The dashboard itself allows to manipulate the dataset by setting constraints on the desired shape, graph 1, and filtering based on the maximum volume, graph 2. In the back-end, however, the data is first being manipulated. The goal of the dashboard is mainly to compare the performance of different bridges. This would imply that a normalization strategy would be at hand. However, when the distribution of a representative design space is being analyzed, figure 6.1, it becomes clear that it is extremely skewed, due to the outliers posed by the generative design method. Most normalization methods are sensitive to this skewed distribution. Therefore, the data is 'normalized' using a ranking transformation. The best performing bridge in a certain category is given rank 0, the second-best rank 1 and so on. These ranks are then converted into a -1 till 1 scale. Implying that the best performing bridge has value -1 and the worst has value 1. This range is most suitable for the t-SNE plot. It is important to note that with this method the relative performance will no longer be included in the data visualization.

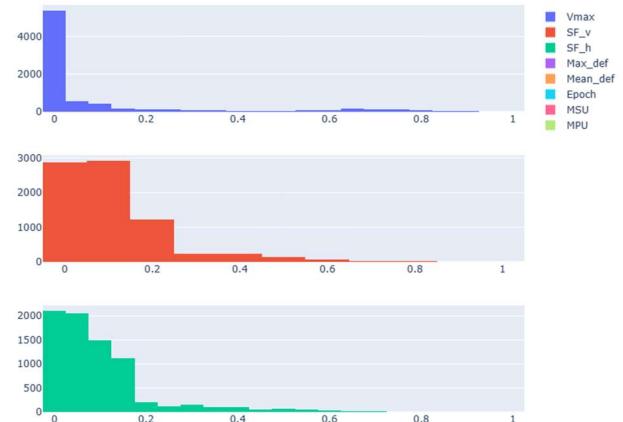


Figure 6.1 Distribution of a representative design space.

6.2 Graphs making up the dashboard

Graph 1: Constraining the domain

The first graph allows users to set constraints on the design space. By using the sliders next to the tool the used design domain can be decreased. When constraints are applied, datapoints that fall outside of the limit are de-emphasized, by reducing their opacity. Plotly offers the ability to exclude datapoints from visualization by selecting their legend entry, using this method one can remove the constraint datapoints from the graph. This helps users to filter non-compliant design.

Graph 2: Data Sampling

The second graph offers the capability to filter data from the database. Not all datapoints are relevant, when the users objective is to find the best performing bridge in a certain category. A built-in distribution plot is used for this purpose, which enables the users to select specific data ranges for

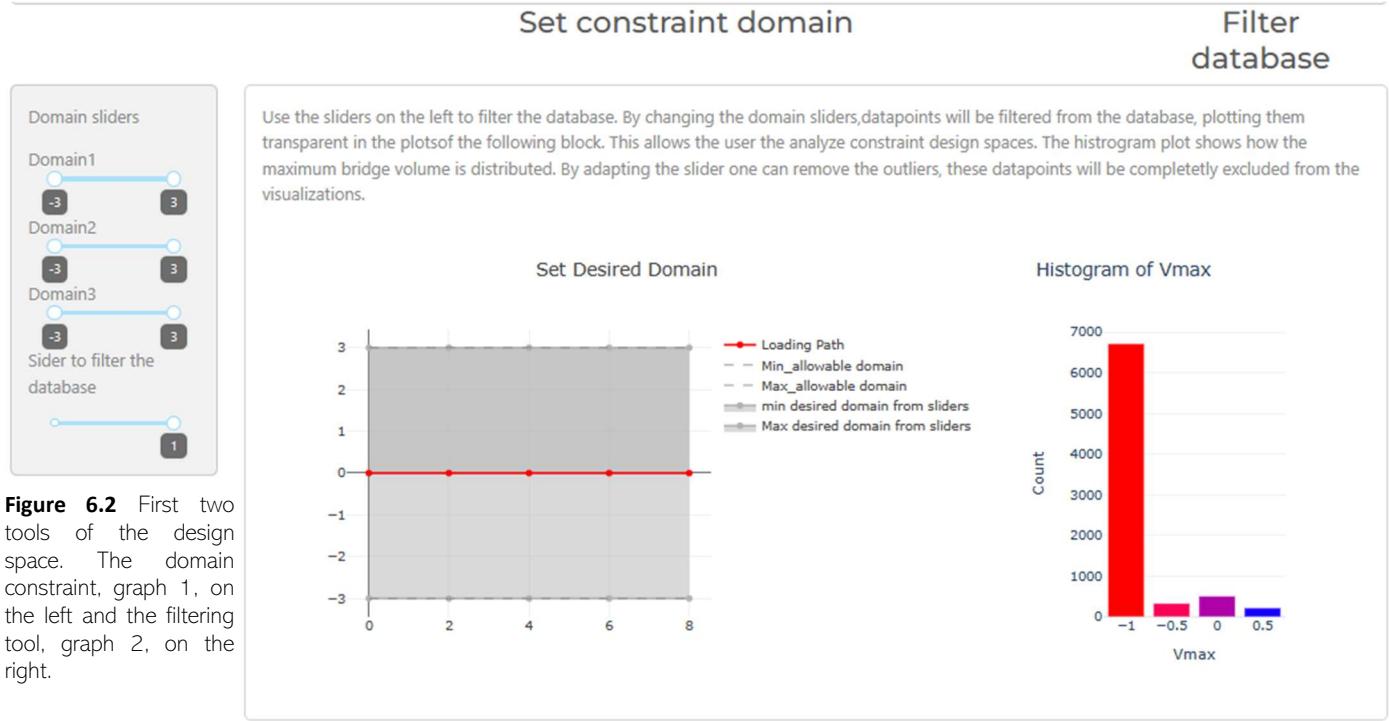


Figure 6.2 First two tools of the design space. The domain constraint, graph 1, on the left and the filtering tool, graph 2, on the right.

each parameter of interest. Designs falling outside these selected ranges will be removed from the visualization, thereby focusing only on the samples of interest. This data filtering mechanism allows users to examine and compare a manageable number of design alternatives without being overwhelmed by the full complexity of the design space.

After processing the data, the user can use three graphs to analyze and explore the dataset. All graphs allow the user to plot the output data and thereby explore their desired design.

Graph 3: Objective Space

The next graph allows the user to interactively plot a three-dimensional objective space. A table is provided next to the plot where various weights can be entered for each axis. Additionally, tools are provided to set bounds to the visualized axis, allowing it to zoom on to a specific domain. This tool can be used to quickly find a desired bridge. In addition to this, changing the marker shape of the plotted data can also help to indicate links within the dataset.

Complex optimization functions that would for example optimize towards costs, would base their cost function based a specific weights ratio between different calculated output parameters. By providing a table next to the objective space one can explore desired complex objective functions. Therefore, the axes have been named, "cost", "manu" (manufacturability) and "eco" (environmental). The interactive features allow us to analyze how the objective spaces change by creating different variations of these complex objective spaces. In the standard settings, the "cost", "eco" and "manu" sliders are set to maximum volume, combined support forces and epoch respectively.

Graph 4: t-SNE (t-distributed Stochastic Neighbor Embedding)

One of the main challenges for high-dimensional data is the incapability of human readability. High-dimensional data with more than three dimensions is difficult to visualize in a two or three-dimensional space. This is where dimensionality reduction techniques such as t-Distributed Stochastic Neighbor Embedding (t-SNE), Principal Components Analysis (PCA) and Uniform Manifold Approximation and Projection (UMAP) come into place. These techniques project the high-dimensional space into a two or three-dimensional plot. Choosing between techniques depends on their applicability to preserve local or global data structures and their handling of linear or non-linear relationships. Out of these three techniques t-SNE is chosen for its performance on local data structures and ability to handle non-linear data. UMAP can perform similarly but tends to introduce artificial clustering (Huang et al., 2022). The built-in t-SNE algorithm of Sklearn is applied into the data exploration tool.

The functionality to choose whether to plot into a two or three-dimensional space is added to the tool. Additionally, the perplexity and learning rate can be set by manipulating the visualization. Where the learning rate determines how much the optimization may change throughout each iteration. Increasing the learning rate therefore causes more spread, whereas decreasing causes denser clustering. The perplexity determines the ratio between keeping local or global data structures, where high values emphasize global structures and low values vice versa.

The t-SNE plot clusters local input data structures together. By plotting the data points in the colour scale of a certain output parameter, analysis can be conducted if certain clusters over or underperform. Based on this

conclusions could be drawn on clustered performances. It is, however, important to note that this is a stochastic method including a randomness component. Therefore, it will never reproduce exactly the same image.

For graphs 3 and 4 dropdown menus are added allowing categorical data as marker shape and numerical data as color scaling. The legend can then be used to filter undesired categories from the plot. This method allows to explore two additional dimensions in the same plot.

Graph 5 Parallel categories plot

Next to the t-SNE plot a more conventional graphing method, for exploring high-dimensional spaces is introduced; the parallel category plot, a variation on the parallel coordinates plot. The parallel categories plot is similar to the parallel coordinates plot, apart from dividing each axis into bins. This plot is applied because its built-in features allow us to explore how different input combinations result in the analyzed output. It thereby allows the user to find relations between desired parameters. The axes of the plot can be shifted, allowing to place the desired parameters next to another desirable axis. Output parameters can be added to the plot. The numerical values of the output are being transformed into bins, similar to

those of the histogram plot. In addition to this a dropdown menu is provided allowing to add the desired output parameters into the PC plot. The parameter inserted as last in the dropdown menu determines the colorscale of the PC plot

Graph 6: Visualization of the bridge & Radial bar plot

Finally, the selected elements in the scatterplots plots or dropdown menus are visualized at the bottom of the file. A finite number of elements can be selected before the page automatically refreshes. A radial bar plot is added next to the visualization showing the ranked output values of the selected bridge allowing for comparison with other selected bridges. Hovering over the bar shows the original data, this allows for relative comparison between design potentials. On the other side of the visualization the input data is being described. A table is being provided that can find the index of a bridge based on the desired input. This will output an index of the corresponding bridge. This allows for easy comparison between bridges that do or do not contain a certain variable.

Design Space Exploration

Adapt the visualization of the plots

Select category for markershape in visualization

Select an input category

Select category for colors in visualization

Vmax

Adapt the visualized bounds

Cost bound

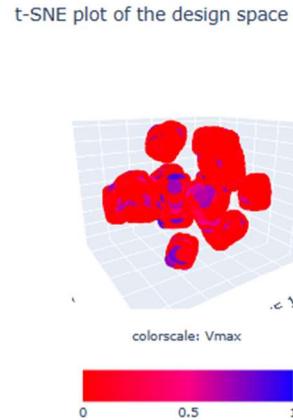
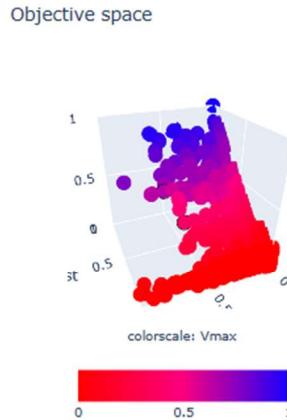
Eco bound

Manu bound

Manipulate the t-SNE plot

Dimensions of t-SNE

The scatterplots below can be used to explore parameters and find desired bridges. The plot on the left, plots the objective space according to the table below the graph. Users can thereby explore various objective spaces and the influence of parameters on the objective space. The graph on the right is a t-SNE plot, which plots all input parameters in a two or 3 dimensional space. Clusters of similar input will form, which can be analyzed by plotting various output parameters as colorscale to it. The sliders on the left allow to manipulate the visualization, by setting different colorscales, different marker types and changing the t-SNE calculation. The perplexity can be changed to change the ratio between preserving local or global structures, which leads to an increase or decrease of clusters. While the learning rate



Set weights for the objective space

Index	Cost	Manu	Eco
MSU	0	0	0
MPU	0	0	0
Vmax	1	0	0
SF_v	0	0.5	0
SF_h	0	0.5	0
Max_def	0	0	0
Mean_def	0	0	0
Epoch	0	0	1
a_p	0	0	0

Figure 6.3 Objective space, graph 3, on the right and the t-SNE plot, graph 4, on the right

Design Space Analysis

Parallel categories plot

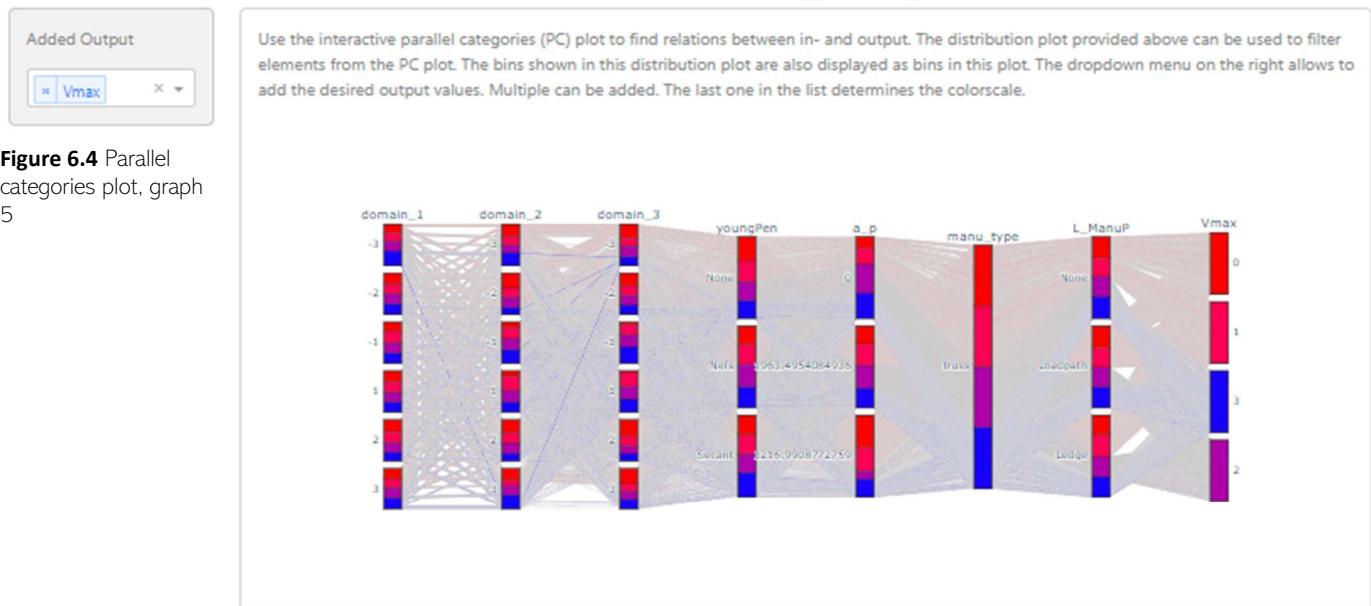
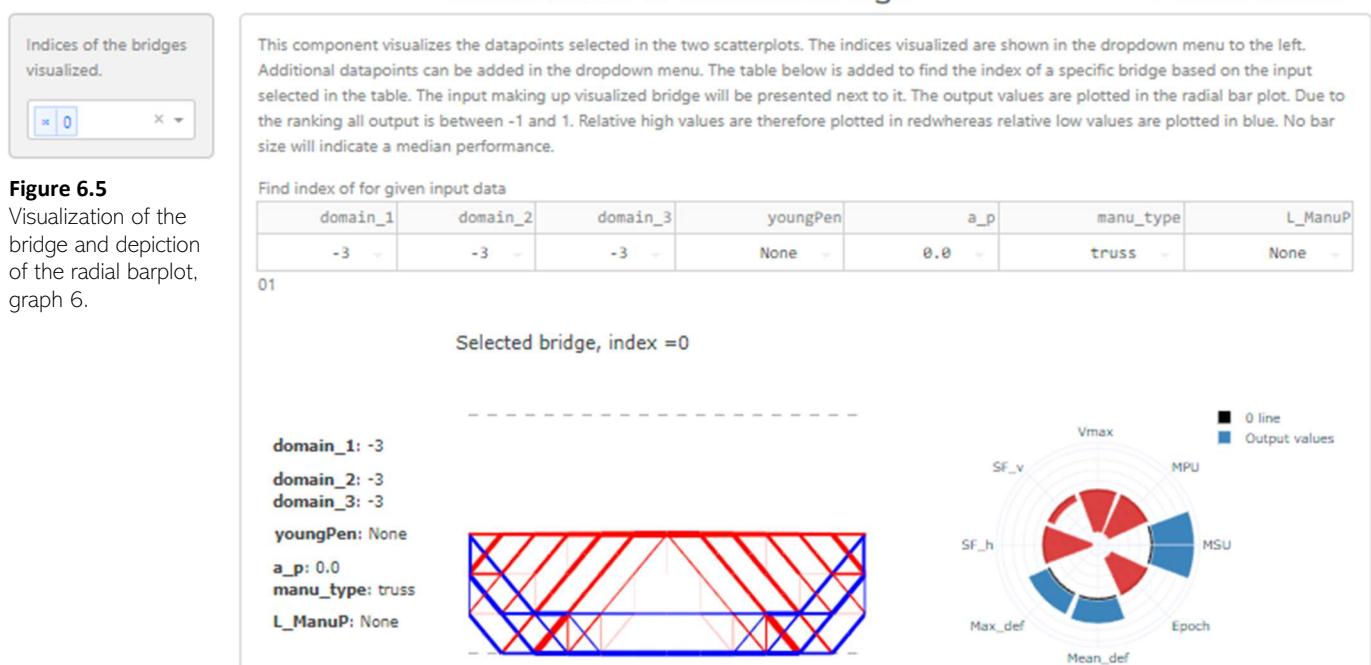


Figure 6.4 Parallel categories plot, graph 5

Performance



Chapter 7 Numerical Examples

The elements and tools making up the optimization will be validated by showing multiple benchmark problems or examples visualizing the intended effects. In addition to this the strain energy decrease will be visualized for each numerical case.

The following examples will be looked into:

- **7.1** General optimization
- **7.2** Youngs modulus penalization
- **7.3** Prestress
- **7.4** Relation beam truss
- **7.5** Forced members

By exploring the effect of individual parameters and analyzing if they perform as expected the total performance will be validated. There will be no checks on combined behaviour, it is assumed that the individual checking is sufficient.

Table 4 describes the main parameters used in all numerical examples. Where necessary additional information will be provided in tabular input or is shown next to the visualizations themselves.

Table 4: Settings numerical examples

Material variables	
Young's modulus	20000 [N/mm ²]
Yield strength	30/3 [N/mm ²]
Compression/Tension	
Optimization variables	
Threshold	1 [print width]
1 Print width	10 [mm]
Calculation type	Truss
Tolerance	1e-5 [-]
Maximum iterations	50 [-]
Structural case	
Support	Pin, roller
Load	1 [kN], or defined next to example
Width, b	100 [mm]
Starting height, x	100 [mm]

7.1 General ground structure

First the general optimization is being validated. All tools except for those essential to the convergence have been disabled. Common examples, or benchmarks, are selected as cases for validation. The examples depicted can be compared to the literature of Bendsøe(Bendsøe & Sigmund, 2004). The optimization results on the next page, show on the left the first iteration of the optimization and on the right side of the page the final converged solution. The color scheme highlights key structural attributes: red for

compression, blue for tension, and grey for elements below the minimum threshold, slated for removal in the post-processing phase.

Structure A1 is a simple case of a cantilever truss, indicating what would occur with the structure, matching one's expectation. The optimization redistributes all material into a triangular truss, resulting in a two-component structure where one element is in tension and another in compression. Case A2 builds upon case A1, the span of the cantilever is now increased. Resulting in the same topology as Case A1 at the end with a new braced cross being added to support the increased cantilever. The result found is in line with benchmark of Bendsoe. Case A3 is a scaled version of case A2 the similarity in results indicates that no scaling effect occurs in the optimization. The final case, Case A4, is the so-called MBB- benchmark. Resembling a beam supported at both ends subjected to a point load in the middle. The resulting truss-shaped structure matches the expected structure to be found. To conclude, all case results match the expectations set by the benchmarks.

The Epoch graph, displayed in Figure 7.1, provides a visual representation of the convergence process for each of the trusses. Notably, the strain energy consistently decreases over successive iterations, showcasing a smooth convergence trajectory. An interesting observation arises in the third iteration when a slight deviation occurs; this phenomenon aligns with the initialization of step size damping.

Furthermore, the graph highlights an important trend between Cases A2 and A3: scaling up the structure or refining the mesh leads to the same optimized structure with identical strain energy performance. However, it's worth noting that this increased level of detail comes at the cost of increased convergence time.

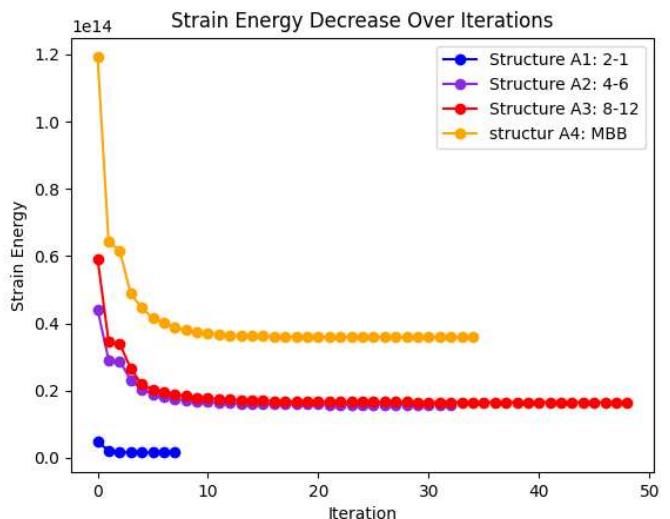
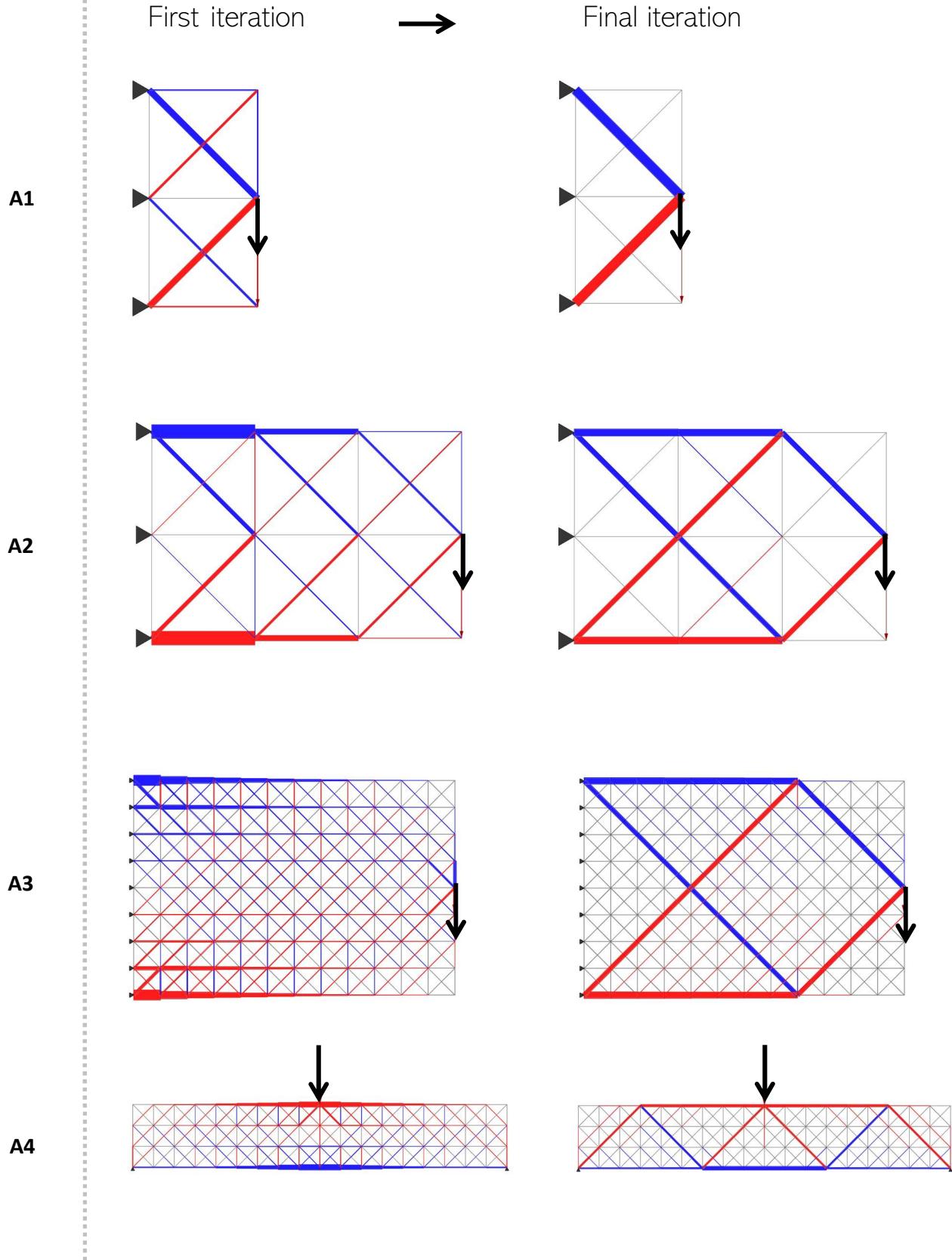


Figure 7.1 Strain energy decrease over the iterations for the cases of the general ground structure.

7.1 General Ground Structure



7.2 Young's Modulus Penalization

Table 5 Young's modulus penalization variables

Example specific parameters	
YoungPen	None, Linear, Smooth
Nef's penalization factor	$\sqrt{10}$
Force	Next to pictures

The images on the following page visualize the effect of Young's modulus penalization on simple structures. Table 5 shows the additional parameters used for this validation. In order to gain a deeper understanding of how Young's modulus penalization works, 'simple' intuitive structures are presented as there are no benchmarks for this method. The structure of the figures is as follows: The central element shows the optimized case without Young's modulus penalization, the 'normal' case. The plots on the left show the effect of smooth penalization, while the plots on the right show the effect of linear penalization.

The aim of Young's modulus penalization is to get the optimization to reassign material in tension to elements in compression where possible. However, when looking at its operation, it is important to note that the smooth penalization is dependent on the stress of the element and therefore the force applied to the system. This penalization method does not have a single Young's modulus for tension and compression but has a unique Young's modulus for each value within its yield stress range. Under- or overstressing the structure will give different results. Overstressing the structure results in designs similar to linear penalization due to the upper limit of the smooth penalization method, which causes the yield modulus to be equal to the linear modulus. On the other hand, under-stressing the structure results in limited penalization, which makes the structure similar to the normal optimization case. Preferably, in the present design case, the stresses would remain within the applied stress ranges. This optimization may therefore work best in combination with a stress constrained optimization. This is illustrated by the two cases presented.

Case 1

Case 1 describes a two-element beam with a central force. In this case the material distribution can only be assigned to one of the two elements. The case clearly shows the difference in functioning between the two optimization strategies. At both a low and high force application to the structure, the linear penalization method pushes all material to the element in compression. This, because no matter which is force is applied to the structure, the penalized material in compression will always transfer forces more efficiently. The smooth optimization however shows the impact the applied force can have on the penalization method. At forces, for this case resulting in yield stresses around than the yield strength in tension, the optimization pushes all material to the compressive elements, because penalized young's modulus is similar to that of the linear

penalization. For lower force applications, however, the penalized material properties of the elements in tension is not that significant yet, implying the stiffest structure is obtained by applying a (smaller) amount of material in tension.

Case 2

This example involves a simple structure optimized as predominantly tensile without penalty. The case demonstrates that for both structures, material usage in compression increases significantly under penalization. From these cases, it becomes evident that penalization effectively increases the allocation of material to compressive elements, without removing elements in tension that are of essence for the structure. Similarly, to Case 1 the smooth optimization behaves as an intermediate between no penalization applied and the linear penalization.

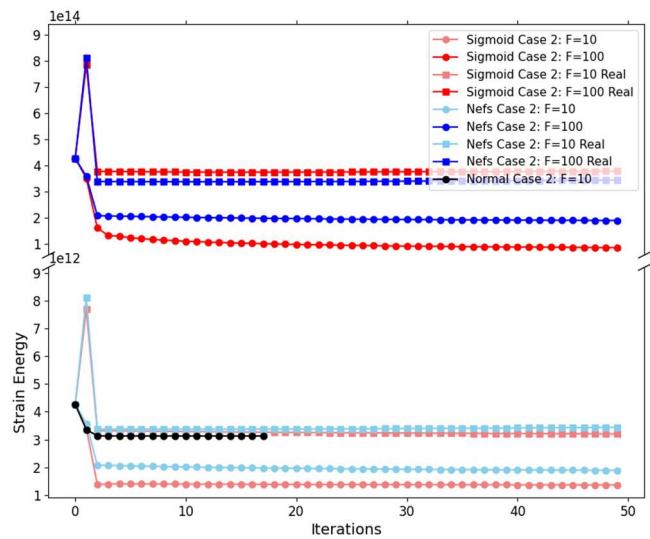


Figure 7.3 Strain energy gradient for case 1, Two-element beam.

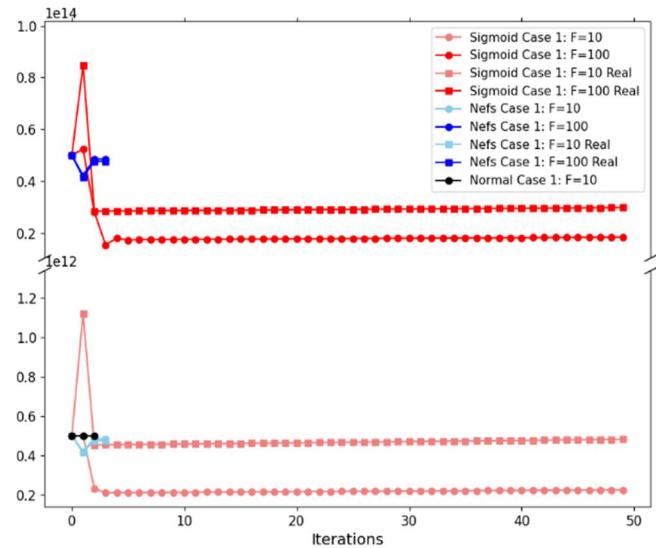


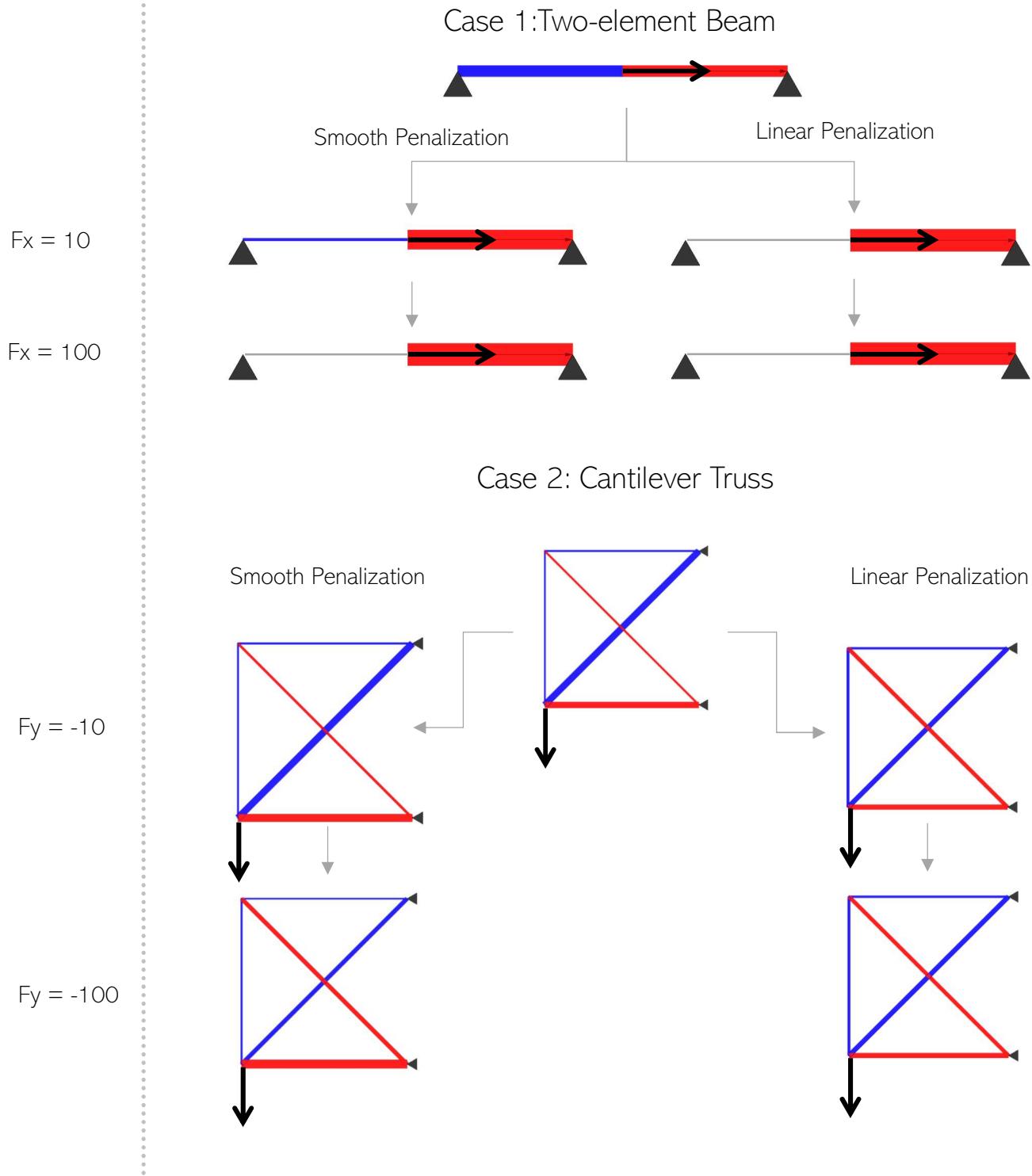
Figure 7.4 Strain energy gradient for case 2, cantilever truss.

The strain energy reduction plots shown in Figures 7.3 and 7.4, Case 1 and Case 2, contain the strain energy used

Chapter 7 Numerical Examples

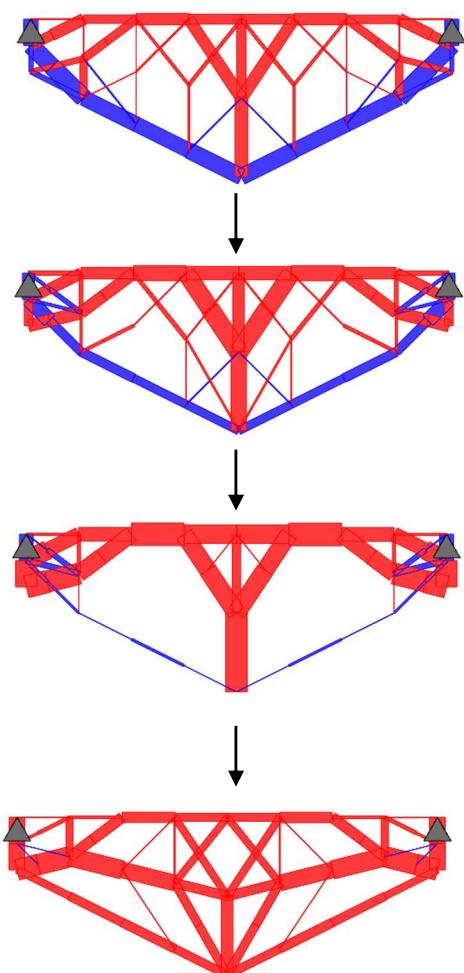
in the optimization with Young's modulus penalty and its actual strain energy with the normal Young's modulus of concrete, denoted as 'Real'. The plots show that the convergence is similar for structures with fully utilized stresses or with non-fully utilized stresses. Based on Graph 7.3, the Smooth penalization appears to outperform the linear penalization both in terms of objective strain energy and 'Real' strain energy. Unexpectedly, this difference between linear and smooth penalization behaviour becomes larger for the overstressed structure. This might however be caused by the scaling induced by the increased force. In Case 2, the smooth objective strain energy also appears to outperform the linear strain energy. However, in this graph the increase in force causes a shift in the better performance, initially the smooth penalized performs better, while at high forces the linear penalized performs slightly better. It is also noticeable that in this case the 'normal case' objective performed better than the smooth and linear penalization at low forces. However, this could be explained by the fact that the special case is a tension dominated case. Therefore, the penalized models, now with less material in tension, behave less stiffly in their tensile components, causing their real strain energy to increase compared to the normal case.

7.2 Young's Modulus Penalization

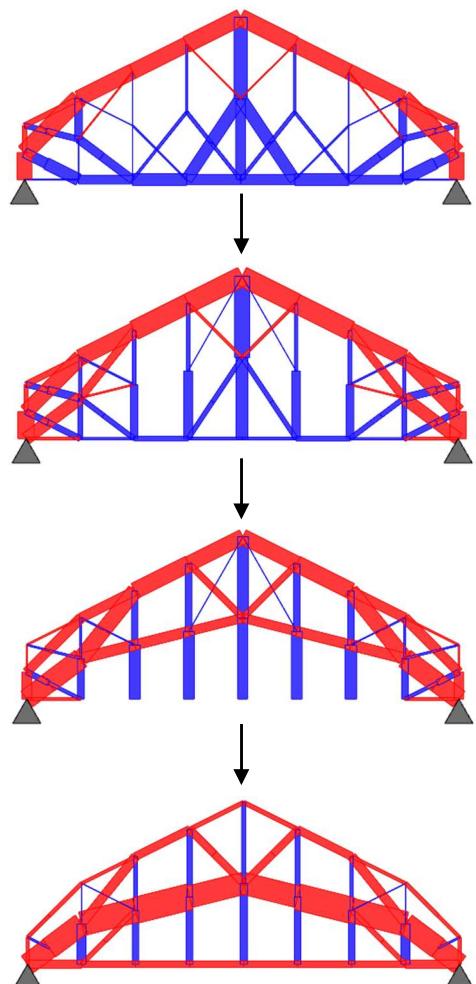


7.3 Prestress

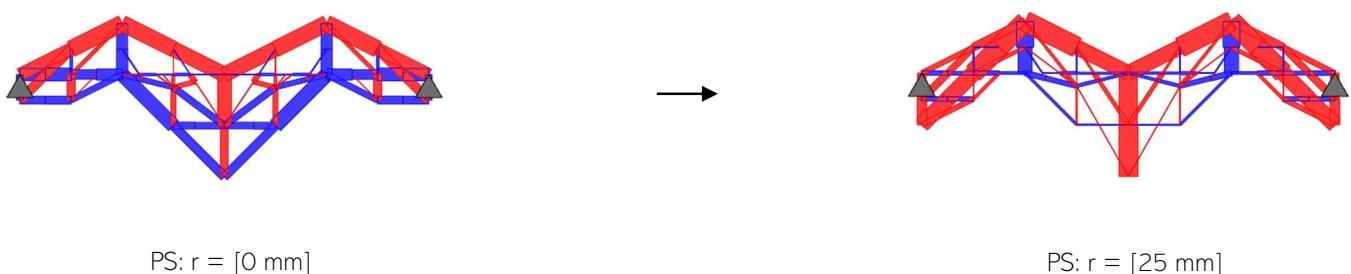
Case 1: Underspanned Prestress



Case 2: Tied-Arch



Case 3: Undesirable Prestress



7.3 Prestress

To illustrate the functionality of prestress, its application at various strengths is visualized on page 35. Two cases are presented, on the left for an under-spanned structure and on the right prestress application for an arch shaped structure. A third case is presented to explicitly show that prestressing can cause elements to appear to be disconnected.

Case 1:

In this scenario, an underspanned structure subjected to prestress is considered. As prestress is gradually applied, a significant transformation occurs. The bottom tensile element within the structure gradually fades in response to increasing prestress application. This phenomenon can be attributed to the prestress's ability to counteract and limit tensile forces within the structure. As the prestress strengthens, the compression forces along the load path become increasingly predominant. Essentially, the prestress serves as a counterbalance that redistributes forces and reduces reliance on the previously tensile-loaded bottom element. This highlights how prestressing significantly modifies the internal force distribution within a given structure.

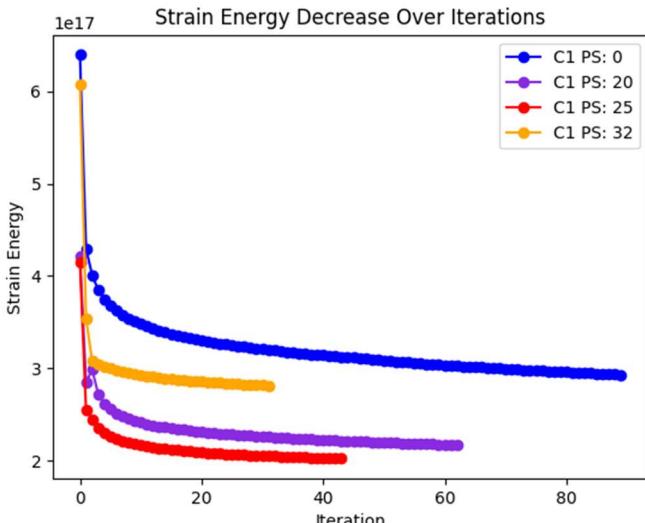


Figure 7.5 Strain energy decrease for prestressing case 1, underspanned prestress.

Case 2:

In the second case, the application of prestress effectively gives rise to two arch-like forms within the structure. One of these forms resembles a catenary curve and begins to dominate the structural configuration.

At medium levels of prestress, the prestress counteracts the tension on the elements along the loading path. Due to elements in this region appear to "disappear" or become significantly reduced in size. Essentially, the prestress force becomes sufficient to balance and neutralize the tension generated by the bending forces. At higher levels of prestress, this effect is mitigated, resulting in a completely compressed structure.

In general, the images show that applying prestressing to the structure limits the tensile forces. Furthermore, the images show that at increasingly higher force applications elements start to become disconnected. This is due to the fact that the prestress is also applied to the structure as nodal end forces, meaning that instead of the force passing through the beam, it only occurs at the end nodes. This can be clearly seen in the example on the right at a radius of 25 mm, where the prestressing cancels out the force in the beam and creates connectors that take all the force from the end nodes. On the bottom of the page Case 3, a prestressed situation of a less conventional case is shown, where the previously described effect results in a disconnected vertical element in compression.

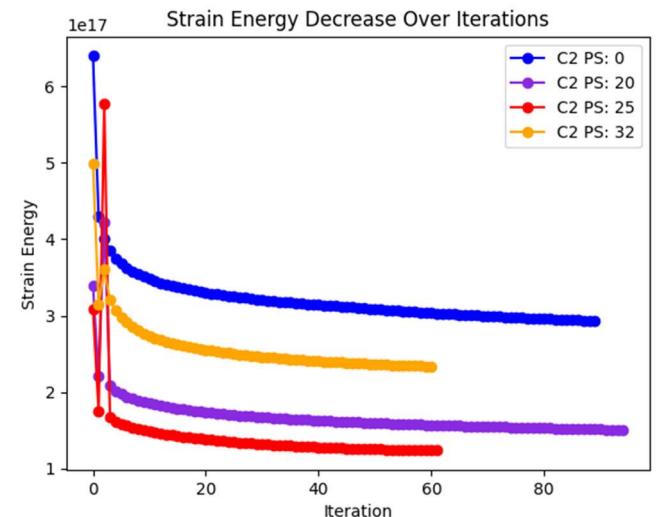


Figure 7.6 Strain energy decrease for prestressing case 2, tied-arch

Looking at the strain energy graphs shows that the prestress helps with converging. Most of the prestressed structures tend to converge faster than the normal case. From both cases the conclusion can also be drawn that more prestress application does not lead to better structures, in terms of strain energy performance. The final case of 32 mm prestress seems to overstress both structures, likely causing large horizontal displacements. These horizontal displacements are more severe than the deformation due to the distributed load on the structure, causing higher strain energy objective at the converged state.

7.4 Beam-Truss Relationship

Chapter 5.2 describes that one of the generative parameters allows to calculate the structure either as a beam or a truss. This slider does however not seem to have a large impact on the structure, as can be seen in the parametrized tree of chapter 5.5. It was expected that changing this slider would result in an entirely different global optimum. In which the solution would switch from normal force dominated structures to structures predominantly in bending. yet changing the slider only locally changes the material distribution. One reason for this could lie in the fact that the heuristic optimization principle is originally based on the

strain energy of axial elements. At the starting iteration the bending strain is of the same order of magnitude as the axial strains. The bending strains however become insignificantly smaller than the axial strains throughout the optimizations. Currently the structure is being analyzed after which the resulting strain energies are being calculated for each force type separately. The resulting strain energies are then summed up and used as driver for the optimization objective. Even though these energies have the same units, and the potential energy of a beam can be calculated as the sum of these energies, it is expected that this method does not describe the correct combined behaviour, causing the axial elements to dominate the material redistribution.

7.5 Construction Constraint

On the next page, page 38, several examples are provided of the Construction constraint implementation. In which FM indicates the location where a minimum bound is being set to the material allocation. Once again, no benchmarks are available for this method. Therefore, validation will be based on examining the structures on expected behaviour. The images on the left show the optimization without Construction Constraint whereas the images of the right show those with, at the location denoted by FM.

Case 1:

In the first case, the size of the bottom horizontal bar is constrained to being relatively large. Doing so, does however not seem to influence the material redistribution in the optimization. The increased material usage of the bottom horizontal bar seems to be covered by a scaled reduction in the material usage of all other members.

Case 2:

The second case constraints the material usage on the loading path. The effects of this case seem to minorly affect the structure similar to Case 1. The increased material usage on the load path does, however, seems to influence the bottom tensile element slightly more than others. Which could be reasoned by the load path counteracting more of the global bending moments.

Case 3:

A vertical element at half of the span is being constrained. Inducing material where the ‘normal’ optimization would not apply any material. This results in the fact that the construction constraint does not only apply the material at the desired location, but also locally changes the material distribution around the constrained element.

Case 4:

To further illustrate the possible effect of the construction constraint changing the local material behaviour a similar case to that of Case 3 is proposed. In which the same change in local material allocation can be observed.

Generally, these images show that this tool does not lead to a new global optimum, but that the optimization changes the local structure around the influenced element to meet the newly made changes. However, the addition of newly formed bars around the forced member proof that the method is not merely functioning as a postprocess step within the optimization. The method actively participates in shaping the local structure to meet the specified constraints. To further evaluate the effectiveness of this constraint, the strain energy plot is examined, depicted in Image 7.6. The plot mainly shows that the final strain energy of both structures is rather similar. However, as expected the case without the Construction Constraint consistently performs slightly better in terms of overall structural performance. Unexpectedly the constrained performance of Case 1 overperforms the unconstrained optimization. Due to the nature of the heuristic optimization, the found optima will likely never be the true global minima. It is therefore likely that the construction constraint pushed the optimization into a different local minima for this structure which exceeds the unconstrained local minima.

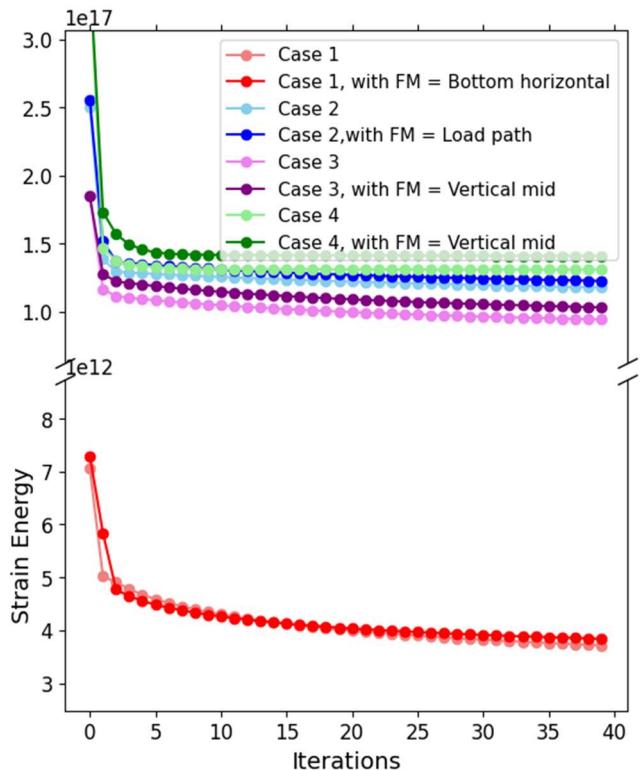


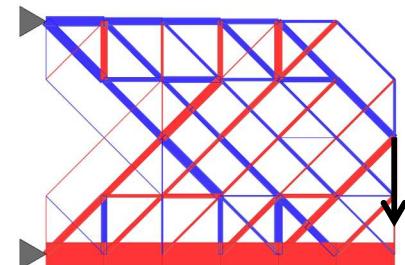
Figure 7.7 Strain energy decrease construction constraint.

7.5 Construction Constraint

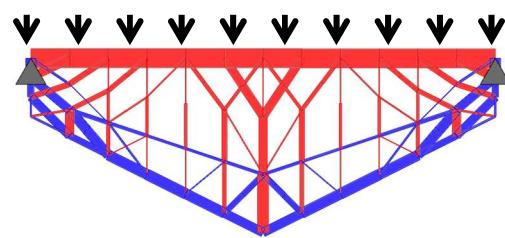
Normal optimization

C1
FM: bottom

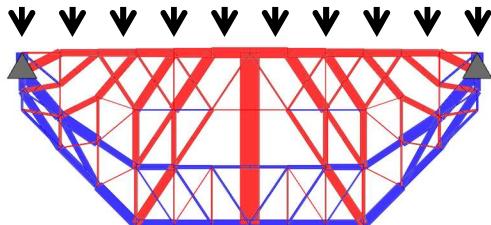
With Construction Constraint



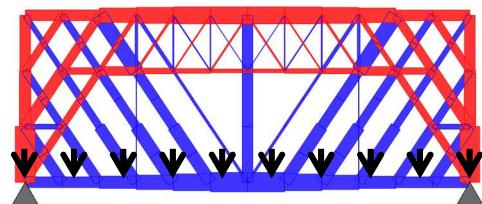
C2
FM: Load path



C3
FM: Middle



C4
FM: edges+
middle



Chapter 8 Design Case 3DCP

This chapter will provide information and results about a design case, which is created to validate the dashboard. In this design case all parameters making up the design space will be quantified and described. To validate the dashboard, it is reviewed if it can be used to answer the “What to design” and “How to design” questions stated in the research objective, in chapter 1.2. The outcomes of the dashboard will be subjected to data analysis to confirm the conclusions drawn from the dashboard. The structure of this chapter will include:

- **8.1 Description of the case**
- **8.2 “What to design?”**
- **8.3 “How to design?”**

8.1 Case Description

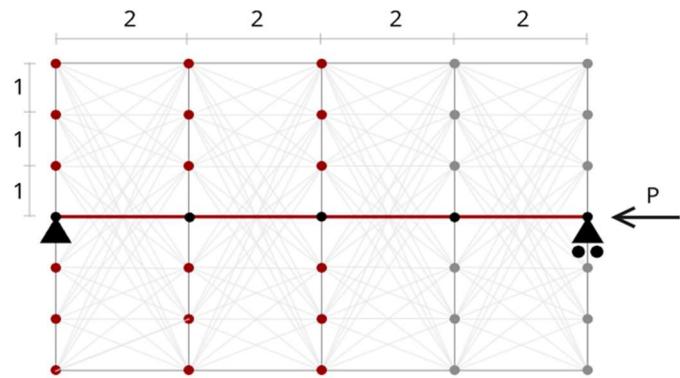


Figure 8.1 Visualization of the design space for the design case.

The bridge case that will be explored consists of a span of 8 meters. The optimization is allowed to use a maximum structural height of 3 meters and a minimum structural height of -3 meters, from the loading path. The domain heights are explicitly chosen to cover both relative stocky and slender bridge designs with varying heights. A pin-roller support condition is assumed, with a compressive force, imitating the prestress force, applied to the roller. This is all visualized in image 8.1, where the red dots display the potential domain heights that will result in the varying design domains. The structural height used in the optimization will always run from the node at the loading path to one of the red dots. The light grey lines indicate possible boundary shapes for the structures generated.

Table 6 describes the input used for generating the design space, which will be explored with the dashboard. The generative parameters described in the table create a variety of 5832 bridges. Based on the numerical examples of Chapter 7.3 the prestress is chosen where the examples performed best, 25 mm, and where the prestressed seemed to be overstressing the structure, 32 mm. For the Young's

modulus penalization, the ratio between initial applied material, the section height and width, and the load, is chosen such that the stresses remain within the yield strengths for a large variety of cases. The construction constraint tool in its turn has forced material on the loading path or edge of the structure. The loading path could be useful for the practical reason of attaching a deck to the bridge structure. Whereas the forced edge could be useful for connecting the parts together.

Table 6 Parameters of the design case

Generative variables	
Domains	[-3, -2, -1, 1, 2, 3] [m]
YoungPen	None, Linear, Smooth
Prestress	0 – 25 – 32 [mm]
Construction Constraint	None, Load path, Edge
Optimization variables	
Max iterations	30 [-]
Damp factor	0.15 [-]
Convergence tolerance rate	1e-3 [-]
Minimum section height	1 Print width
Geometrical properties	
Grid	Structured
Grid subdivision	3
Structural properties	
Density	2400 [kg/m ³]
Youngs modulus	30000 [N/mm ²]
Yield compression	30 [N/mm ²]
Yield tension	3 [N/mm ²]
Yield stress prestress	1860 [N/mm ²]
Ratio Linear penalization	$\sqrt{10}$ [-]
Starting Width (b)	500 [mm]
Starting Height (x)	100 [mm]
Poisson	0.2 [-]
Assumed print layer width	10 [mm]
Forced member size	100 [mm]
Load	1 [kN/m]
Gravity	On

8.2 “What to design?”

Within the design case, two cases will be analyzed where different objectives are set to the question “What to design”. First, a case where the objective is to find the bridges with the least volume applied to them, with the only constraint that prestress needs to be applied to the structure in order to fix the two perceived parts together. For the second case, a multi-objective nature is added to it. Not only is the minimum volume of importance but also the combined support force behaviour. A bridge is ought to be found where both of them are minimized. Additionally, only material is allowed to be placed below the loading path. The prestressing constraint is extended to be possibly replaced by forcing material to the edges which would allow for possibilities to connect the two perceived modular shapes.

Additional results, without explanation on the steps to find them, are visualized in Appendix E.

8.1.1 Case 1 Minimize volume with prestress

The case described above can be formulated into an optimization function described below.

$$\begin{aligned} \min V_{max} \\ s.t. A_p > 0 \end{aligned} \quad (27)$$

Where

V_{max} = The maximum volume

A_p = Applied prestress area

First, a description will be given of how the dashboard is used to find the final desirable bridges. After that, the final objective space will be visualized by plotting the potential bridges.

Steps applied to find the desired bridges

Histogram of V_{max}

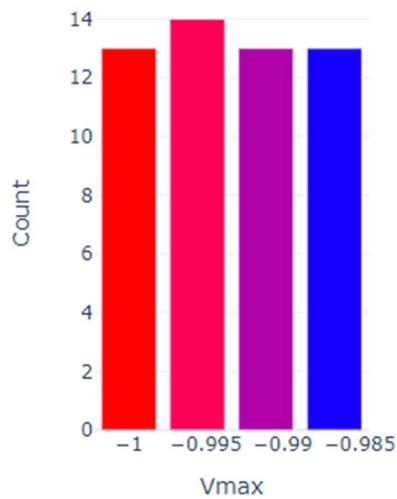


Figure 8.2 Histogram after filtering to the desired number of bridges, for Case.

Step 1) Filter design space based on volume:

Filter the design space to only include an arbitrary top percentage of the design space. The filtering can be rather extreme since there is only a single objective, in this case only the top 1.5% are shown. This results in the arbitrary amount of around 40 bridges that are being included in the analysis.

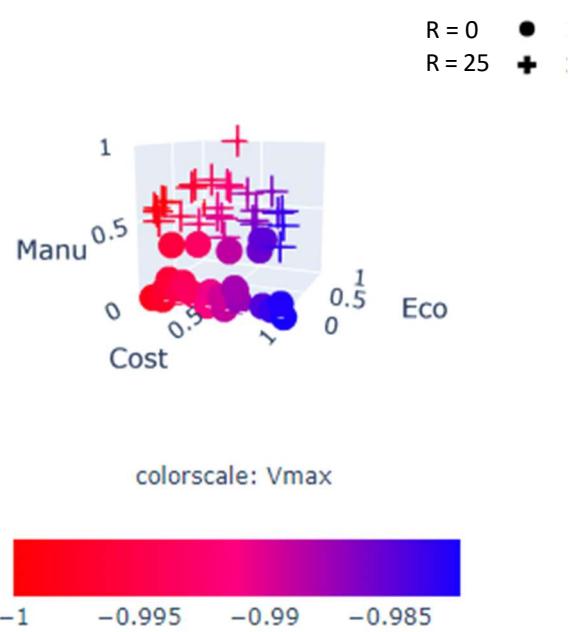


Figure 8.3 Objective space containing only bridges with prestress.

Step 2) Filter based on prestress:

After adding the prestress to the visualization, all datapoint which do not include prestress can be removed from the visualization by clicking on their legend entry. The resulting datapoints from the histogram filtering do however not include any results without prestress. The coloring is standardly set to maximum volume this therefore does not need to be changed.

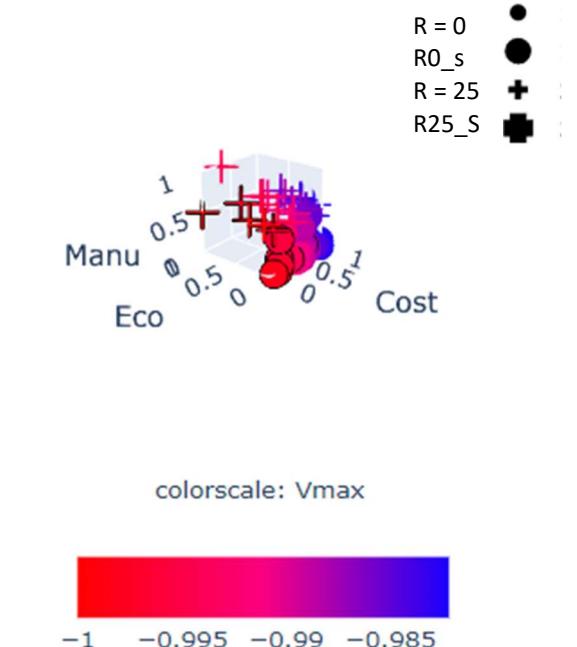


Figure 8.4 Selected points from the objective space.

Step 3) Select datapoint with lowest maximum volume

The objective space derived from step 2 now only shows a few potential designs. By clicking on the markers, the index of the relevant bridge will be sent to the visualization. Because only the volume is of concern to the datapoint which performs best on the Vmax axis, in this case the 'cost axis' can be selected.

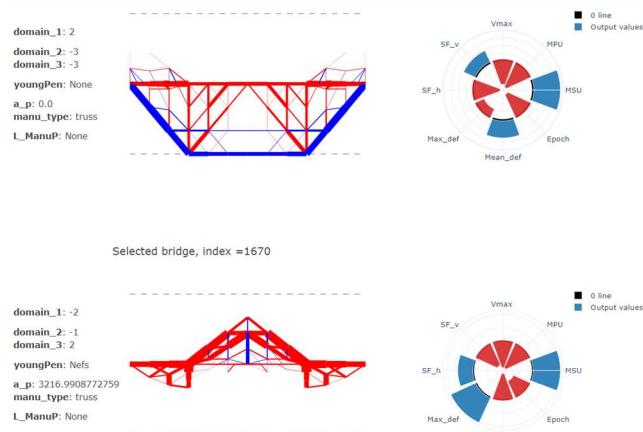
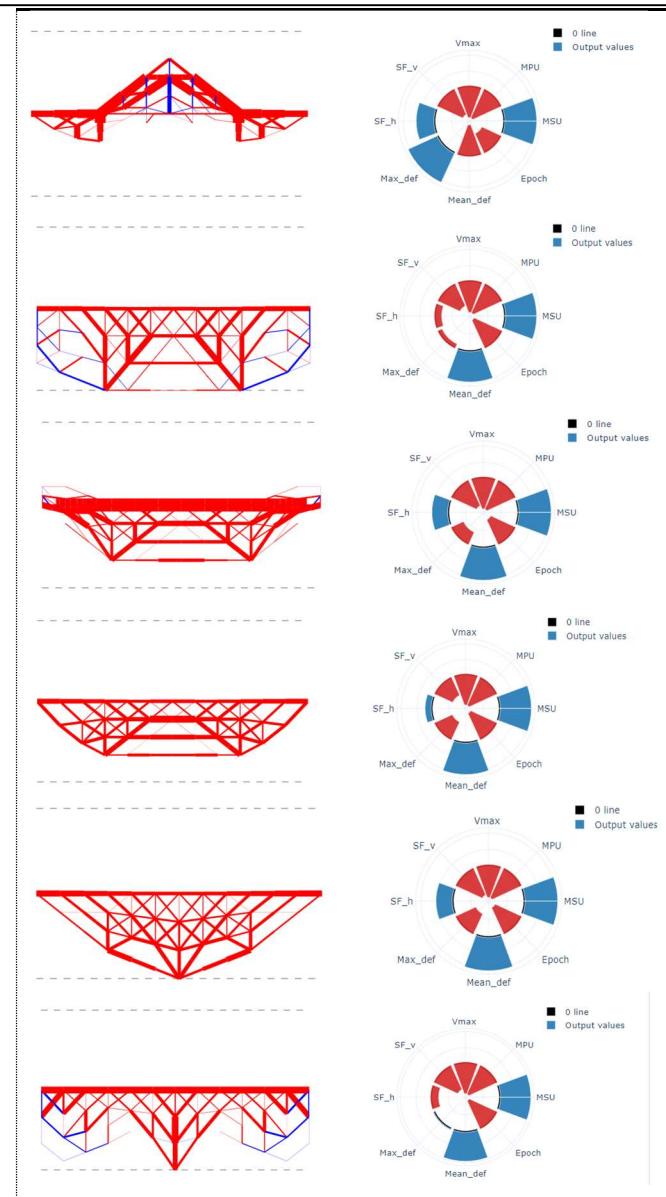
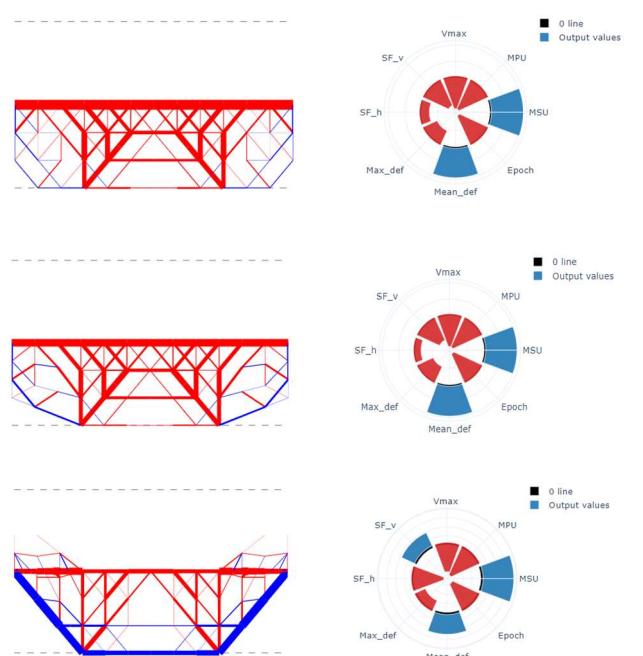


Figure 8.5 Visualization of two bridges in the dashboard.

Step 4) Analyze & Compare

The clicked datapoints will be visualized below. These resulting bridges can be compared to one another. By hovering over the radial bar plot stats, a text item appears in which the true performance of each output value is denoted. These can be used to compare relative performances. The final resulting bridges are depicted in the following rows, subjective criteria could be implemented to pick the final desired bridge topology.



Figures 8.6.1 -8.6.9 The bridges complying to the design objective.

The results show a variety of bridges being included in the objective space for the design question of minimizing the total weight given a minimum prestress constraint. All of the filtered results, the datapoint available after step 1, contain prestress so no additional filtering is required. Analyzing the resulting bridges reveals that all bridges are a variant to so-called under-spanned bridges. The reason for this could be that the prestress application works best for these under-spanned type of structures. Additionally, a note should be placed to image 8.6.4. This bridge did make it to minimum volume bridges, but the maximum deflection did not converge. The bridge showed a deflection of 107 mm and should therefore be excluded from the results. The other bridges have a deflection between 4 and 18 mm. Apart from the addition of prestress, it is striking that 4 out of 9 bridges contain the manufacturing material constraint to the loading path, whereas it was expected that this tool would reduce functionality. Therefore, it would be expected that a similar bridge without the construction constraint tool should also be included, but this is not the case. Showing that the

construction constraint tool in specific cases indeed overperforms the unconstraint optimization.

8.1.2 Case 2: Multi-Objective Constrained Optimizaiton

The objective of case two could be formulated by the notation below, in equation 28. In words, this function describes the minimization of both the maximum volume as well as the combined support forces. It is up to the user to decide the ratio between those two. Next to this the prestress constraint is being extended to be able to be replaced by the construction constraint tool, if that has been set to 'Edge'. In addition, to this only material above the loading path is allowed to be placed.

$$\min V_{max}, SF_V + SF_h \quad (28)$$

$$s.t. \begin{cases} A_p > 0 \text{ or } FM = 'Edge' \\ Domain_{1,2,3} > 0 \end{cases}$$

Where

- $SF_V + SF_h$ = The combined vertical and horizontal support forces
- FM = Forced Members
- $Domain_{1,2,3}$ = The structural height of the three domain sliders.

Steps applied to find the desired bridges

Histogram of V_{max}

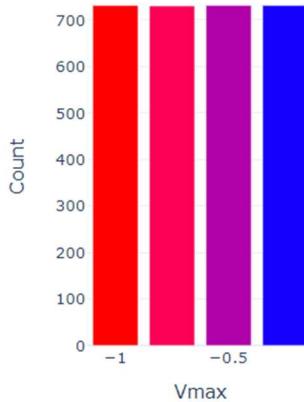


Figure 8.7 Histogram of the design space for case 2, roughly 50% filtered.

Step 1) Filter design space based on volume:

Filter approximately 50% of the design space. Filtering is only possible on maximum volume. Due to the multi objective nature filtering more than half would include

filtering designs that are likely to overperform in the combined objective function.

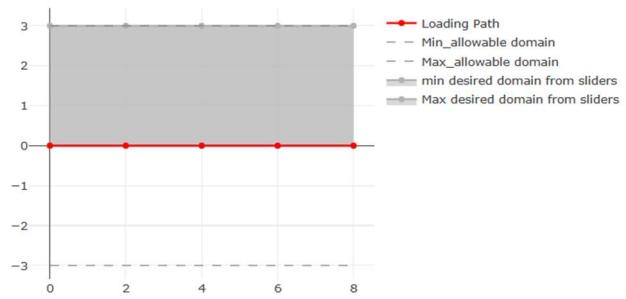


Figure 8.8 Constraint set to only include material above the load path, for case 2.

Step 2) Set constraint

Reduce the design space to only include datapoints where volume is applied below the loading path. The other datapoints will be visualized with reduced opacity. By clicking on their legend entry, they will be removed from the design space.

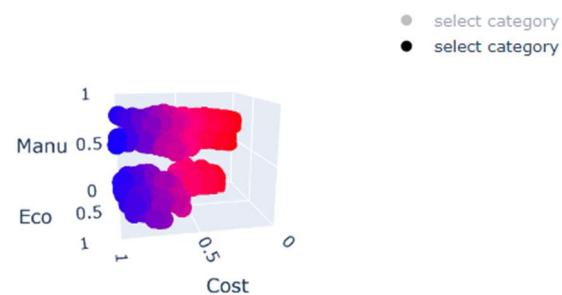


Figure 8.9 Starting point of the objective space before adapting the axes and visualization method. The reduced opacities have been removed.

Step 3) Create the objective space

Set the objective space to the desired settings. Create one axis containing the horizontal and vertical support forces, "Manu", one with prestress, "Eco", and one with maximum volume, "Cost". Set marker shape to manufacturing constraints. Decrease the bounds of the plot to low values of the axis where the volume and support forces are plotted on.

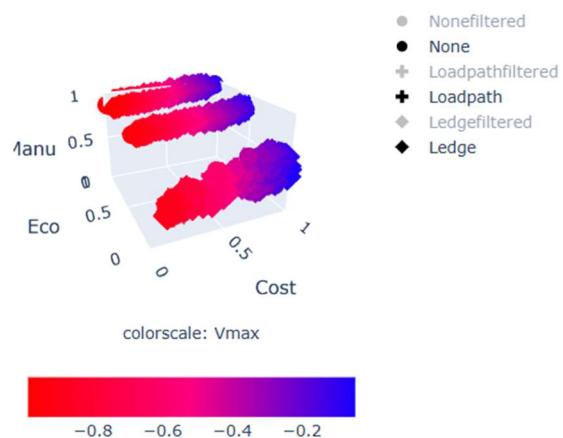


Figure 8.10 Resulting objective space after filtering and applying the described visualization method.

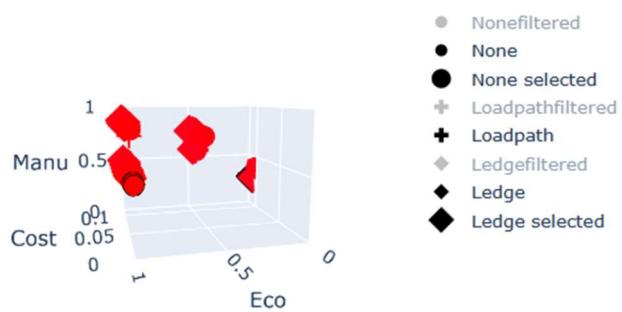


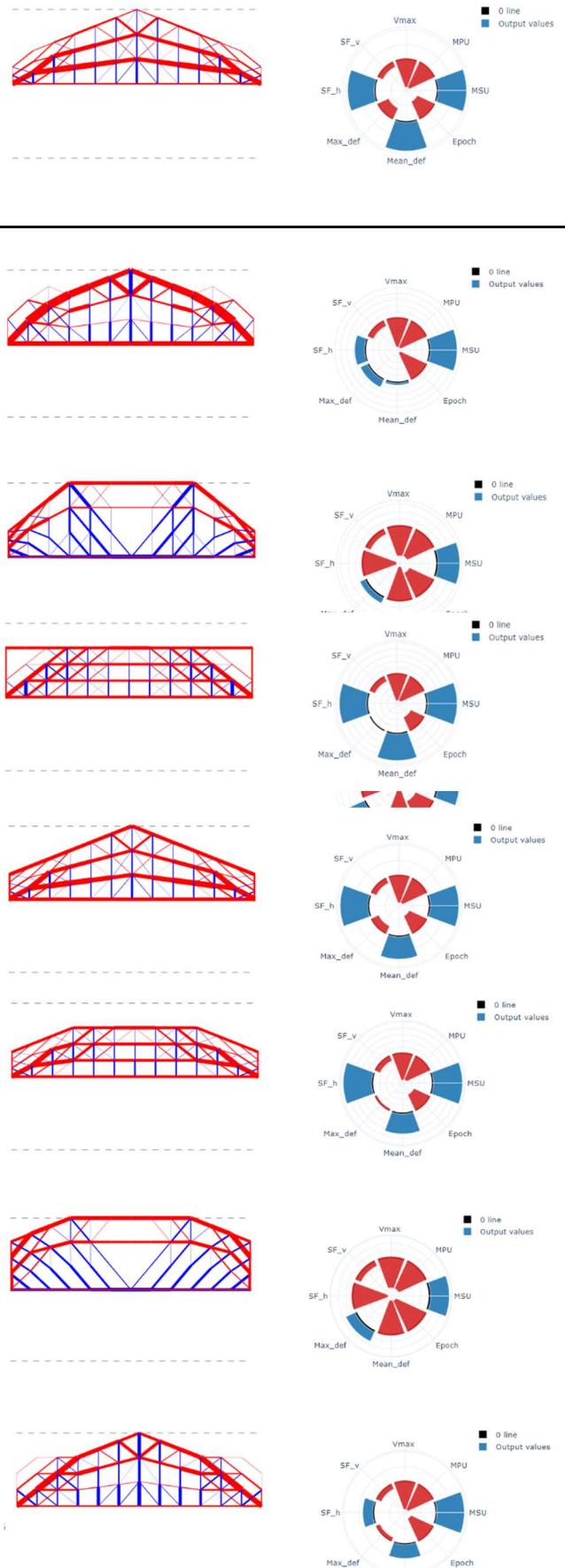
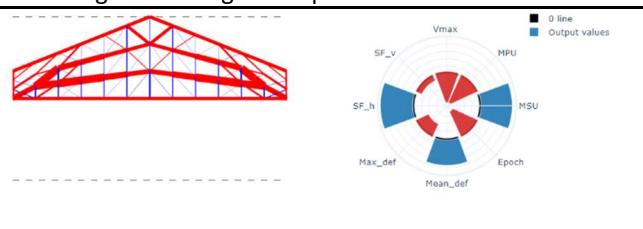
Figure 8.11 Visualization of the objective space including the selected points.

Step 4) Select datapoint with lowest maximum volume

The user should now select the point which lies on their desirable pareto front and can subjectively decide which points to select in the high-performance zone. The desired bridge should come from a row containing prestress force or should be a diamond shaped marker to comply with the objective goals.

Step 5) Analyze & Compare

The resulting bridges can be compared to one another. By hovering over the radial bar plot stats, a text item appears in which the true performance of each output value is denoted. These can be used to compare relative performances. The bridges visualized in the next rows are the bridges adhering to the process described:



Figures 8.12.1-8.12.12 Visualization of the possible bridges given the set of requirements.

Most results show a variation of a tied arch being created by prestress induced to the system. If no proper arch can be formed, due to limitation in geometry, a more truss like structure will be formed. In example 8.12.5 The construction constraint tool seems to have placed ‘useless’ material in the corners of the structure. It does, however, have one of the highest maximum volume performances. Implying that the applied material does likely contribute to the overall stiffness. The results without prestress show clear resemblances to truss structures. In these trusses you see a clear usage of the edges of the design domain in which the infill pattern distributes material from the loading path towards the nodes of the upper edge. All these bridges perform slightly worse in maximum volume applied, but they do not have a horizontal support component. It could therefore be worthwhile to analyze if the additional material that would be required to connect these elements together, without prestress, is worth the trade-off over the reduced material required for the support.

8.3 “How to design?”

The dashboard can also be used to analyze the design space and thereby analyze the influence of parametric implications on the design space. In this thesis, two new optimization tools are proposed to apply to the known ground structure optimization: smooth Young’s modulus penalization and the construction constraint during an optimization. Using the dashboard these techniques will be analyzed in order to draw conclusions on their functioning. A correlation matrix is created to verify the results drawn from the dashboard. The correlation matrix is based on linear relations between input and output variables, shown in Figure 8.13. However the only true relation t3hat can be established from this matrix, is the relation between prestress and horizontal support forces.

8.3.1 Young's modulus penalization

First the young’s modulus penalization will be analyzed in the dashboard, more specifically the smooth penalization (SP) will be compared to the linear penalization (LP). Nefs

showed in his thesis that his original penalization method increased overall performance. The numerical examples in Chapter 6. showed that overstressing the design case results in similar structural behaviour of the SP to LP. The design case, however, is chosen such that most bridges will not be overstressed, therefore being suitable to analyze how this difference in behaviour affects structural performance. It is important to note that, due to the dependency of the SP on stresses and thereby forces on the structure, comparing behaviour with only a single loading condition is insufficient for complete comparison and can therefore only be used indicative. Analyzing the design space shows that there does not seem to be a large discrepancy between the linear and smooth penalization. It is however striking that the bridge without Young’s modulus penalization outperforms both penalizations, at high levels of filtering. The reason for this is likely the fact that the maximum volume reduction is conducted with the homogeneous Youngs modulus used in the first iteration. This implies that the material is distributed with a different Young’s modulus, which could cause the minimized volume to be higher due to this change in Young’s modulus.

8.3.2 Construction Constraint

Analyzing the construction constraint shows at first that there does not seem to be any difference in performance between the construction constraint variables. However, filtering the database again shows that forcing more material results in worse performing bridges. No Construction constraint performs better than Load path, which again performs better than edge. This effect does, however, not seem to be severe. It must however be noted that the severity is dependent on the amount of material, which is been set to the lower bound, increasing this material amount will have a larger effect on the total strain energy in the structure. The effect the construction constraint tool in general has on the overall performance can however not be quantified with the tool. Striking however from Chapter 8.2, one result contained a variant including a minimum material allocation to the loading path. The variant without this material allocation did not make the final

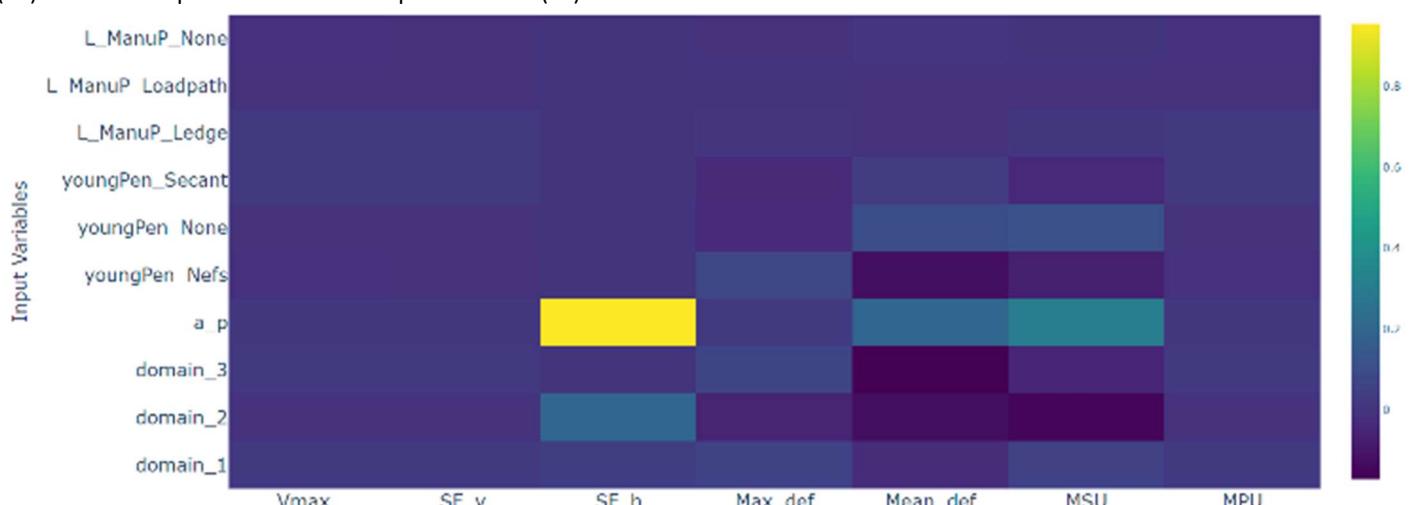


Figure 8.13 Correlation Matrix. Input variables on y-axis and output variables on the x-axis.

selection implying that the tool overperformed the design without this tool. This can be explained by the fact that heuristic optimization is applied. The optimization result is therefore heavily induced by the starting situation, where all material is equally distributed. Changing this distribution, for the forced members to include more material, pushes the optimization into a different solution space into which an optimum is found containing more material to the construction constraint variable.

8.3.3 Shape analysis

Analyzing the parallel categories plot, at two different filtered distributions, shows that the domains largely influence the maximum volume in the structure. Mainly the third domain seems of importance. High values contribute to better bridges. This is already reflected in the initial parallel categories plot, filtering the domains only further clarifies this, as shown in image 8.14. When hovering over the specific blocks, it seems that mainly the arch inducing domains perform well, for example 1,2,3, this however is visually based on the amount of highlighted lines. The plot does not clearly quantify this suspicion.

The goal of the t-SNE plot is to analyze how the different domain combinations behave. It was expected that the clustering derived would be based on the different domain combinations. One could then analyze the performance of a single cluster by viewings its coloring and thereby viewing the performance of a domain combination. However, the actual clustering is differently constructed which did not allow for the perceived analysis. Image 8.16 shows the high order of organization within a cluster. Nevertheless, the concept of this methodology can be verified with the case of prestressing and maximum horizontal support force. Each cluster on the 2D plain is one type of prestressing. Plotting the color as maximum horizontal support force then shows the expected behaviour. More prestress leads to more horizontal support force. Diving a bit further in this analysis, using the t-SNE plot shown in figure 8.15 shows that for the middle type of prestress, the horizontal support force seems

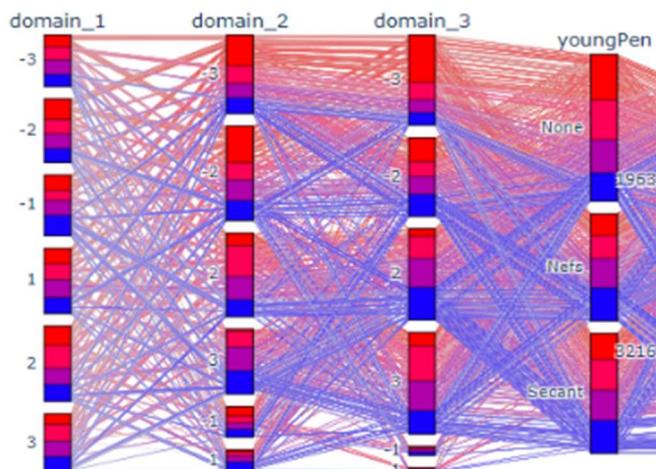


Figure 8.14 Parallel categories plot for a filtered design domain. Clearly domain 3 and the amount of prestress seem to be of large importance. The None Prestress seems to be better performing than the Secant and Nefs. The Construction constraint tool, L_ManuP, shows a slight decrease in performance for more material forced.

dependent on the height of the second domain. This can also be explained by the fact that the prestress cable is first under an angle for domain_2 values lower than 0. The height of domain_2 determines the value of the angle and thereby influences the amount of horizontal support force. This is also reflected in the correlation matrix shown in image 8.13

t-SNE plot of the design space

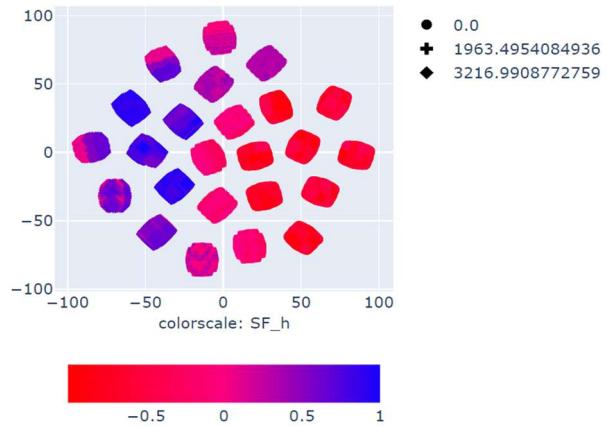


Figure 8.15 t-SNE plot where each cluster respresents a type of prestress. The colorscale is the maximum horizontal force.

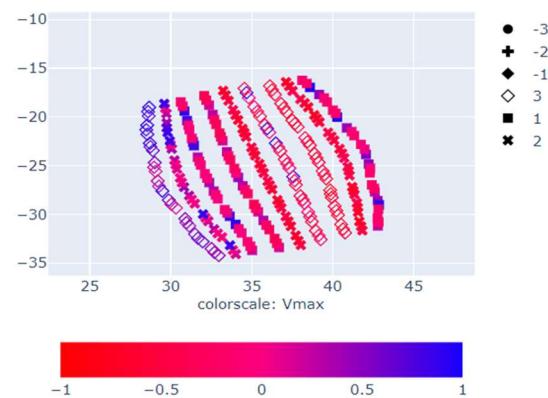
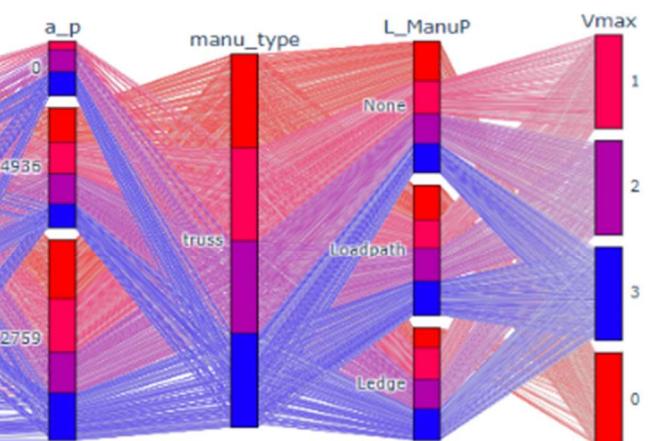


Figure 8.16 Advanced Clustering within a cluster makes interpretability difficult.



Chapter 9 conclusion

This thesis has stepped forward on the challenge of realizing the potential of 3D concrete printing in design applications. By introducing a user-guided design space exploration, dashboard insights are being provided on the “what to design?” and the “How to design?” aspects. This is all by demonstrating the application of generative design principles that adhere to the limitations and possibilities of 3D printing, leading to optimized and structurally feasible outcomes. The goal of this research was to create a model able to find desirable topologies for 3D concrete printed bridges (i.e. “What to design?”) and with that method also support decision making by providing information of parametric implications on the design space (i.e. “How to design?”). Thereby a method is proposed to analyze and compare a parametric design space suited for 3D concrete printed structures. By being able to address these questions, this thesis contributes towards the development of 3D concrete printed designs.

The developed user-guided data exploration dashboard offers an array of tools to guide the early design process. Through filters, visualization and analysis the dashboard serves as guide for informed decision-making. This is showcased by the case study addressing the dashboard’s capability to find desired bridges, whereas the analysis tools can help in steps to decision making. The results show that the dashboard can quickly be used to find desirable bridges, based on desirable output variables. Applying additional constraints or multiple objectives to the desired bridge does not lead to additional complexity in finding the desired bridges. Whereas the results have shown that it does lead to entirely different topological solutions, showing the bias designers can set upon themselves with ‘black box’ type optimizations. When complicating the design space one can easily visualize potential bridge solution, where then an exact selection can be made with subjective criteria. This selection can be based by the visualization, visualizing the outcome of the bridge, with its relevant ranking on each output domain located next to it. The tool clearly shows till what point exact or subjective choices are being made, which makes it specifically tailored to convey design choices and solutions as structural engineering to clients or architects. Thereby properly being able to answer the “what to design?” given a specific design space at hand.

Analyzing the design space with the dashboard proved to be more difficult. One can pinpoint rather simple relationships within the design space, but the tool lacks the ability to quantify these relationships. Nevertheless, the parallel categories plot offers a rapid overview of the distribution within the filtered design space, adding value to the exploration tool. The t-SNE plot, however, proves less intuitive to interpret. The clustering appeared to be in an order still not suited for analysis. Manipulating the input of the t-SNE does result in different clustering but did not provide the desired clustering perceived for analysis. The

relationship that was established between horizontal support force, prestress and the second design domain parameter did however show how the tool could be useful, when a ‘logical’ interpretable way of clustering is provided. Overall, the case study was an effective measure to test the dashboard’s functionality. It tested multiple design objectives successfully. Whereas it also provided options for analysis that are being verified by data analysis. The results on the functioning of the dashboard are mainly positive, with the side note that future development should focus on additional tools for quantitative analysis.

Zooming in the method to create the design space for the 3D concrete printed bridges shows that the leading design principles have been worked out into several tools throughout the process. To start the newly proposed optimization tools; the construction constraint, and the Young’s modulus penalization, showed that they positively contributed towards designing 3D print specific bridges. To elaborate, The Construction Constraint results showed only a slight expected decrease in performance, as expected by pushing optimization to a different path. While chapter 7.3 did show that it does locally change the structure around the constraint material. Thereby ensuring it does not act as a postprocessing step within the optimization. Furthermore, For the young modulus penalization, there does not seem to be much difference in behaviour between the linear and smooth penalization method. Applying no penalization overall leads to more high performing bridges in terms of maximum applied volume related to the desirable maximum deflection. The overall stresses in the penalized structures did however seem to be slightly lower as has been shown by plotting the mean stress utilization. In the mean stress utilization, the linear penalization did seem to overperform the smooth penalization whereas it was expected that the smooth penalization would drive structures into lower stress utilization, this because a higher stress utilization implies a lower Young’s modulus leading to a higher strain energy. The overall results did however show that it is really case dependent which penalization method works better and the reason for this is likely due to the exact force distribution in the structure leading to a different smooth penalization performance for each bridge. To generalize on this, the smooth optimization method proves more difficult, but could provide more control to the user. By playing around with the occurring stresses in the structure the user could find any solution in between no penalization applied and the linear applied penalization. Finally, prestress had a major positive effect on the structural performance of each bridge, while also ensuring manufacturability.

The varying design domains seemed to lead to a larger variety of topologies being included in the design domain. Not only by the difference in domain boundaries that were possible, but also by the different angles of the infill pattern resulting from these different domain boundaries. This, for example, caused certain design domains to allow for arch shaped structures to be resulted

Chapter 9 conclusion

from the infill pattern, due to a rather strange boundary domain. Similar performance is likely not created when merely refining design space, because refining would still imply similar angles to be suited within the design domain.

In summary, this thesis affirms that design space exploration can guide designers through the intricate landscape of 3D concrete printed structures. Through the introduction of a user-guided dashboard, generative design principles, and a focus on both "what to design?" and "how to design?", this research contributes to the development of effective methodologies for harnessing the potential of 3D concrete printing. The presented model facilitates design innovation by addressing complexities and revealing insights in a way that can enhance decision-making and optimize design outcomes. As 3D concrete printing continues to evolve, this research paves the way for a more informed, efficient, and visual approach to concrete structure design.

Chapter 10 Discussion and Future Research

Design space exploration has proven to be a versatile method for navigating the complex landscape of 3D printed concrete structures, albeit with some limitations in its current application. Future improvements lie in: Expanding the 3D printing framework, Artificial intelligence integration, refining the dashboard's analysis potential and developing current optimization and calculations. By addressing these developments the Design Space Exploration method could increasingly further be used in utilizing the full potential of 3D concrete printing.

Development of the 3D printing framework

Addressing limitations in the current design space, it's important to acknowledge that not all variables influencing design have been thoroughly analyzed due to the scope of this thesis. Yet the strength of the DSE tool is that it can compare the effect of input variables on the final design. Variables such as material mixtures, layer adhesion, and for example curing times, influence fresh and hardened state concrete. The DSE tool could be used to analyze design implications that come with variable settings between these parameters, allowing to aid in the trade-off between fresh and hardened state design properties.

Furthermore, could the to be developed reinforcement strategies algorithmically be included in the optimization process, using similar methods applied for Young's modulus penalization and the Construction constraint, to control reinforcement layout during the optimization process guaranteeing enough concrete cover. The type of reinforcement strategy could then be used as parameter and the resulting bridges could be compared.

Expanding the calculations to 3D analysis of structures would allow to take into account material orientation. Once again, the dashboard could be analyzed to view the challenges that come with this task. A difficulty in 3D design would however be the need to construct monolithic structures or an algorithm that creates parts from the structures to be designed.

Artificial Intelligence integration

Developing the framework as suggested yields various new parameters to the design space, increasing the complexity of the objective landscape. Current practice has already shown the advantages of using a filter on maximum volume to find desired bridges. This could be extended into an approach as described in the literature research. The advantages would include: The number of dimensions could be scaled further without diminishing the usability, No longer need for a database you would train the model, interpolating and extrapolating, use AI decision making to improve human decision making. The dashboard itself can be used to monitor and modify the surrogate itself.

Refining the dashboard

As explained, the dashboard's current analysis potential is rather limited. This is mainly due to the lack of qualitative analysis. The difficulty in this is that every data set is different and could therefore need different graphing and analysis strategies.

Chapter 9 concludes that the dashboard can be used as a tool to communicate a structural design problem to clients or architects. If it were to be used as such, further research into human-centered design could follow. That research will look at what is needed to communicate design decisions and whether the dashboard meets these needs. At the moment the dashboard is mainly based on personal considerations of what would be desirable.

Expanding Calculations:

Chapter 7.5 described that the beam-truss relationship did not work as intended and is therefore not considered further. Behavioral analysis showed that the rotational forces in the beam structure were significantly reduced and became irrelevant compared to the axial forces. The optimization function is therefore likely to have a bias towards axial structures. It is not known what causes this bias and how it could be avoided. However, it would be interesting to further analyze what causes this problem and how it can be solved. This would lead to an increased variety of the design space with a single slider. In addition, analyzing structures as beams would introduce bending moment distributions. This means that the section size of each element of a truss could be optimized for that bending moment. However, this would require additional integration points to be added to the calculations.

In the dashboard Bridges, the main output value against which bridges are compared is the maximum volume, due to the limitations of slicing the dataset based on this value only. As explained in Chapter 5.4.5, this maximum volume is found by optimizing the volume against the deflection constraint, which means that all structures in the database should have the same deflection. However, the convergence of this optimization is not ideal, there is a slight discrepancy in the maximum deflections in the range 0.2 - 20 when they should be optimized towards 15, meaning that slightly worse performing bridges could actually perform as well or better if their optimization converged better.

Another improvement for the maximum volume optimization could lie in the fact that some structures require such large volumes to satisfy the volume constraint that its printability becomes unrealistic due to overlapping sections. One way to solve this would be to apply the scalar volume multiplication from this deflection optimization to the currently fixed depth b instead of the section height x . This would ensure that all structures remain constructable within the plane, but the out-of-plane depth could become too large. For example, one could then filter out bridges whose out-of-plane depth would be greater than the actual width of the bridge, to ensure that all compared bridges would be constructable.

Bibliography

- Autodesk. *Fusion 360 Introduction to Generative Design* / Autodesk University. Retrieved June 8, 2023, from <https://www.autodesk.com/autodesk-university/article/Fusion-360-Introduction-Generative-Design>
- Bendsøe, M. P., & Sigmund, O. (2004). Topology Optimization. In *Topology Optimization*. Springer Berlin Heidelberg. <https://doi.org/10.1007/978-3-662-05086-6>
- Bos, F. P., Lucas, S. S., Wolfs, R. J. M., & Salet Editors, T. A. M. (2020). *RILEM Bookseries Second RILEM International Conference on Concrete and Digital Fabrication*. <http://www.springer.com/series/8781>
- Bos, F., Wolfs, R., Ahmed, Z., & Salet, T. (2016). Additive manufacturing of concrete in construction: potentials and challenges of 3D concrete printing. *Virtual and Physical Prototyping*, 11(3), 209–225. <https://doi.org/10.1080/17452759.2016.1209867>
- Brown, N., Mueller, C. T., Kawaguchi, K., Ohsaki, M., Takeuchi, T., & Brown, N. C. (2016). *The effect of performance feedback and optimization on the conceptual design process*. <https://www.researchgate.net/publication/316107726>
- Buswell, R. A., Leal de Silva, W. R., Jones, S. Z., & Dirrenberger, J. (2018). 3D printing using concrete extrusion: A roadmap for research. *Cement and Concrete Research*, 112, 37–49. <https://doi.org/10.1016/J.CEMCONRES.2018.05.006>
- Butt, S. A., & Lavagno, L. (2013). Design space exploration and synthesis for digital signal processing algorithms from Simulink models. *2013 8th IEEE Design and Test Symposium, IDT 2013*. <https://doi.org/10.1109/IDT.2013.6727109>
- Caitlin Mueller. (2022). *Creative computing for high performance architectur*.
- Danhäive, R. (2020). *Structural Design Synthesis Using Machine Learning Chair of the Department Committee on Graduate Students*.
- Danhäive, R. A., & Mueller, C. T. (2015). Combining parametric modeling and interactive optimization for high-performance and creative structural design. *Proceedings of the International Association for Shell and Spatial Structures (IASS)*.
- Danil Nagy, Lorenzo Villaggo, Dale Zhao, & David Benjamin. (2017). *Beyond Heuristics A Novel Design Space Model for Generative Space Planning in Architecture*. <https://www.semanticscholar.org/paper/A-Novel-Design-Space-Model-for-Generative-Space-in/c1b0b9646c45bbfc3c03995d61743b7fbefac25>
- de la Fuente, A., Blanco, A., Galeote, E., & Cavalaro, S. (2022). Structural fibre-reinforced cement-based composite designed for particle bed 3D printing systems. Case study Parque de Castilla Footbridge in Madrid. *Cement and Concrete Research*, 157. <https://doi.org/10.1016/j.cemconres.2022.106801>
- El-Sayegh, S., Romdhane, L., & Manjikian, S. (2020). A critical review of 3D printing in construction: benefits, challenges, and risks. In *Archives of Civil and Mechanical Engineering* (Vol. 20, Issue 2). Springer. <https://doi.org/10.1007/s43452-020-00038-w>
- Ha, S., & Teich, J. (2017). *Handbook of Hardware/Software Codesign*.
- Han, Z. H., Abu-Zurayk, M., Görtz, S., & Ilic, C. (2018). Surrogate-based aerodynamic shape optimization of a wing-body transport aircraft configuration. In *Notes on Numerical Fluid Mechanics and Multidisciplinary Design* (Vol. 138, pp. 257–282). Springer Verlag. https://doi.org/10.1007/978-3-319-72020-3_16
- Hossain, M. A., Zhumabekova, A., Paul, S. C., & Kim, J. R. (2020). A review of 3D printing in construction and its impact on the labor market. In *Sustainability (Switzerland)* (Vol. 12, Issue 20, pp. 1–21). MDPI. <https://doi.org/10.3390/su12208492>
- Huang, H., Wang, Y., Rudin, C., & Browne, E. P. (2022). Towards a comprehensive evaluation of dimension reduction methods for transcriptomic data visualization. *Communications Biology*, 5(1). <https://doi.org/10.1038/s42003-022-03628-x>
- inceptrta. (2023). *Isight*. <https://www.inceptra.com/solution/isight-the-simulia-execution-engine/>
- Jastrzebska Aleksandra. (2020). *DESIGN SPACE EXPLORATION WITH VARIATIONAL AUTOENCODERS*. <https://www.iaacblog.com/programs/design-space-exploration-variational-autoencoders/>
- Kawaguchi, K., Ohsaki, M., Takeuchi, T., Brown, N. C., & Mueller, C. T. (2016). *Spatial Structures in the 21st Century*.
- Kazi, R. H., Grossman, T., Cheong, H., Hashemi, A., & Fitzmaurice, G. (2017). DreamSketch: Early stage 3D design explorations with sketching and generative design. *UIST 2017 - Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*, 401–414. <https://doi.org/10.1145/3126594.3126662>
- Kinomura, K., Murata, S., Yamamoto, Y., Obi, H., & Hata, A. (2020). *Application of 3D Printed Segments Designed by Topology Optimization Analysis to a Practical Scale Prestressed Pedestrian Bridge* (pp. 658–668). https://doi.org/10.1007/978-3-030-49916-7_66
- Liu, J., Nguyen-Van, V., Panda, B., Fox, K., Du Plessis, A., & Tran, P. (2022). Additive Manufacturing of Sustainable Construction Materials and Form-finding Structures: A Review on Recent Progresses. [Https://Home.Liebertpub.Com/3dp, 9\(1\), 12–34. https://doi.org/10.1089/3DP.2020.0331](Https://Home.Liebertpub.Com/3dp, 9(1), 12–34. https://doi.org/10.1089/3DP.2020.0331)
- Logan, D. L. (2007). *A first course in the finite element method*. Thomson.

-
- Ma, G., Buswell, R., Leal da Silva, W. R., Wang, L., Xu, J., & Jones, S. Z. (2022). Technology readiness: A global snapshot of 3D concrete printing and the frontiers for development. *Cement and Concrete Research*, 156, 106774. <https://doi.org/10.1016/J.CEMCONRES.2022.106774>
- MacLean, A., Young, R. M., Bellotti, V. M. E., & Moran, T. P. (1991). Questions, Options, and Criteria: Elements of Design Space Analysis. *Human-Computer Interaction*, 6(3–4), 201–250. <https://doi.org/10.1080/07370024.1991.9667168>
- Marler, R. T., & Arora, J. S. (2004). Survey of multi-objective optimization methods for engineering. In *Structural and Multidisciplinary Optimization* (Vol. 26, Issue 6, pp. 369–395). <https://doi.org/10.1007/s00158-003-0368-6>
- Mueller, C., & Ochsendorf, J. (2011). *An Interactive Evolutionary Framework for Structural Design*.
- Mueller, C. T. (2014). *Computational Exploration of the Structural Design Space Submitted to the Department of Architecture in Partial Fulfillment of the Requirements for the Degree of*.
- Mueller Caitlin, Brown Nathan, & Felipe Jonathas. (2015). *design-space-exploration-tool-suite-for-grasshopper*. <http://digitalstructures.mit.edu/page/tools#design-space-exploration-tool-suite-for-grasshopper>
- Nefs, K. (2019). *Optimization of 3D concrete printed cycling bridges: a ground structure approach* [TU Eindhoven]. <https://research.tue.nl/en/studentTheses/optimization-of-3d-concrete-printed-cycling-bridges>
- Nourian, P., Azadi, S., & Oval, R. (2023). *Generative Design in Architecture: From Mathematical Optimization to Grammatical Customization* (pp. 1–43). https://doi.org/10.1007/978-3-031-21167-6_1
- Pagliuca, G., Kipouros, T., & Savill, M. A. (2019). Surrogate modelling for wing planform multidisciplinary optimisation using model-based engineering. *International Journal of Aerospace Engineering*, 2019. <https://doi.org/10.1155/2019/4327481>
- Peter Sharpe. (2021). *Solar Aircraft Design with AeroSandbox and Dash*. <https://dash.gallery/dash-aerosandbox/>
- Scharl, J., & Mavris, D. (2001). Building parametric and probabilistic dynamic vehicle models using neural networks. *AIAA Modeling and Simulation Technologies Conference and Exhibit*. <https://doi.org/10.2514/6.2001-4373>
- Siemens. (2021). *HEEDS*. <https://www.redcedartech.com/>
- Sohst, M., Vale, J., Crawford, C., Potter, G., & Banerjee, S. (2022). *A framework for multi-fidelity multi-disciplinary kriging-based surrogate model optimization of novel aircraft configurations*.
- Stump, G., Yukish, M., Simpson, T., & O’Hara, J. J. (2004). *Trade space exploration of satellite datasets using a design by shopping paradigm* (Vol. 6). <https://doi.org/10.1109/AERO.2004.1368206>
- Timperley, L., Berthoud, L., Snider, C., Tryfonas, T., Prezzavento, A., & Palmer, K. (2023). *Towards Improving the Design Space Exploration Process Using Generative Design With MBSE*. <https://doi.org/10.1109/AERO55745.2023.10116019>
- Trek. (2021). *How we made our lightest bike even faster*. https://www.trekbikes.com/ca/en_CA/emonda-aero/
- Tseranidis, S., Brown, N. C., & Mueller, C. T. (2016). Data-driven approximation algorithms for rapid performance evaluation and optimization of civil structures. *Automation in Construction*, 72, 279–293. <https://doi.org/10.1016/J.AUTCON.2016.02.002>
- Vantyghem, G., De Corte, W., Shakour, E., & Amir, O. (2020). 3D printing of a post-tensioned concrete girder designed by topology optimization. *Automation in Construction*, 112. <https://doi.org/10.1016/j.autcon.2020.103084>
- Whalen, E., & Mueller, C. (2022). Toward Reusable Surrogate Models: Graph-Based Transfer Learning on Trusses. *Journal of Mechanical Design, Transactions of the ASME*, 144(2). <https://doi.org/10.1115/1.4052298>
- Wortmann, T. (2017). Opossum - Introducing and Evaluating a Model-based Optimization Tool for Grasshopper. In T. Wortmann (Ed.), *Protocols, Flows, and Glitches - Proceedings of the 22nd CAADRIA Conference* (pp. 283–292). CAADRIA. <https://www.icd.uni-stuttgart.de/research/research-tools/opossum/>

Appendices

Appendix A Code: FEA, Optimization and Dashboard

The relevant code used in this thesis, that of the Analysis/optimization in RhinoCode and the code for the dashboard, is uploaded on the following github page. The readme file will contain more information about the use of the code.

Github-Link: <https://github.com/DennisHollanders/Design-Space-Exploration-of-3D-Concrete-Printed-Bridges.git>

Appendix B Finite elements calculations

The direct stiffness method applied derives from the principle of equilibrium for energy. The potential energy of the structure can be divided into internal, energy due to deformation, and external energy, energy due to applied loads. The potential energy is the sum of these denoted in equation (1)

$$E_{pot} = E_{pot,i} + E_{pot,e} \quad (29)$$

where

E_{pot} = Total potential energy

$E_{pot,i}$ = Internal potential energy

$E_{pot,e}$ = External potential energy

Where the internal potential energy is calculated by integrating the strain and stress vector over the volume of the element.

$$E_{pot,i} = \frac{1}{2} \int_v \{\boldsymbol{\varepsilon}\}^T \{\boldsymbol{\sigma}\} dV \quad (30)$$

Where

$\{\boldsymbol{\varepsilon}\}^T$ = Strain matrix

$\{\boldsymbol{\sigma}\}$ = Stress matrix

V = Volume

Which can be rewritten into

$$E_{pot,i} = \frac{1}{2} [d_a \ d_b] \int_v \begin{bmatrix} -\frac{1}{l} \\ \frac{1}{l} \end{bmatrix} E \begin{bmatrix} -\frac{1}{l} & \frac{1}{l} \end{bmatrix} dV \begin{bmatrix} d_a \\ d_b \end{bmatrix} \quad (31)$$

Where

$[d] = \begin{bmatrix} d_a \\ d_b \end{bmatrix}$ = Displacement vector with d_a and d_b being the nodal displacements

$[N] = \begin{bmatrix} 1 - \frac{x}{l} & \frac{x}{l} \end{bmatrix}$ = Shape function matrix, with l being the length of the element and x the location on the element

$[B] = \begin{bmatrix} -\frac{1}{l} & \frac{1}{l} \end{bmatrix}$ = Strain displacement matrix

The external potential energy includes the work done by external forces and the work done by distributed loads along the length of the

$$E_{pot,e} = -\{d\}\{F\} - \int_l \{u\}^T (q) dl \quad (32)$$

Where

$\{d\}$ = Displacement vector

$\{F\}$ = Force vector

$\{u\}^T$ = Displacement vector

(q) = Distributed load

Combining the internal and external potential energy into the total potential energy results in equation (33)

$$E_{pot} = E_{pot;i} + E_{pot,e} = \int_v 1/2[\mathbf{d}]^T [\mathbf{B}]^T [\mathbf{E}] [\mathbf{B}] [\mathbf{d}] dV - [\mathbf{d}]^T [\mathbf{F}] \quad (33)$$

Taking the derivative of this equation results into the principle of stationary potential energy.

$$\frac{\partial E_{pot}}{\partial [\mathbf{d}]} = \int_v [\mathbf{B}]^T [\mathbf{E}] [\mathbf{B}] dV [\mathbf{d}] - [\mathbf{F}] = [\mathbf{K}] [\mathbf{d}] - [\mathbf{F}] = 0 \quad (34)$$

Thus, implying that element stiffness matrix K is equal to

$$[\mathbf{K}] = \int_v [\mathbf{B}]^T [\mathbf{E}] [\mathbf{B}] dV \quad (35)$$

In practice standard element matrices are used for K depending on the element type.

$$K = \begin{bmatrix} \frac{AE}{L} & 0 & 0 & -\frac{AE}{L} & 0 & 0 \\ 0 & \frac{12EI}{L^3} & \frac{6EI}{L^2} & 0 & -\frac{12EI}{L^3} & \frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & \frac{4EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{2EI}{L} \\ -\frac{AE}{L} & 0 & 0 & \frac{AE}{L} & 0 & 0 \\ 0 & -\frac{12EI}{L^3} & -\frac{6EI}{L^2} & 0 & \frac{12EI}{L^3} & -\frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & \frac{2EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{4EI}{L} \end{bmatrix} \quad (36)$$

A Truss element has an element stiffness matrix containing only deformation due to axial loads while the beam element contains deformation due to transverse loads and rotations. Joined together describes a 'beam-column' element able to describe all translation and rotations within the plane.

The most basic principle for this is the application of a 2-node truss element with a distributed line load along the length of the element.

In this case eq (4) would result in the following equation, where the stiffness matrix k only takes into account the axial contributions.

$$AE \begin{bmatrix} \frac{1}{l} & -\frac{1}{l} \\ -\frac{1}{l} & +\frac{1}{l} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} F_1 \\ F_2 \end{bmatrix} + \begin{bmatrix} \frac{1}{2} ql \\ \frac{1}{2} ql \end{bmatrix} \quad (37)$$

In order to combine multiple elements with different local axes a Jacobian or transformation matrix is used. This matrix translates every local element matrix into the global coordination system. After which the element stiffness matrices can be combined into global stiffness matrix.

$$T = \begin{bmatrix} C & S & 0 & 0 & 0 & 0 \\ -S & C & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & C & S & 0 \\ 0 & 0 & 0 & -S & C & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (38)$$

Where

T = Transformation matrix

C = Cosine of the angle between bar and the x-axis

S = Sine of the angle between bar and x-axis

As for the local and global stiffness, the transformation matrix can be applied to the force vectors and displacement vectors to translate local into global vectors and the other way around. Therefore, both the global displacements and forces can be found by multiplying the transformation matrix with the force and displacements vectors. In this capital letters imply globally transformed matrices and vectors while lower case letters imply local matrices and vectors.

$$\mathbf{F} = \mathbf{Tf} \text{ and } \mathbf{U} = \mathbf{Tu} \quad (39)$$

And as discussed, the global element stiffness matrix can be determined with:

$$\mathbf{K} = \mathbf{T}^T \mathbf{k} \mathbf{T} \quad (40)$$

With these formulations all unknowns can be found in the following order:

Appendices

- First the global stiffness matrix is assembled by combining all K , this by summing the contributions to the same DOFs. The external forces and boundary conditions on the structure are known. Therefore, The global displacements can be solved with:

$$\mathbf{F} = \mathbf{KD}$$
 (41)
- It is known that the displacements at the boundary conditions is equal to 0, therefore the complementing DOFs can be crossed out from this matrix equation.
- With the global displacements known, the equation can now be reversed in order to solve for the internal forces in the structure.

$$\mathbf{f} = \mathbf{T}^T \mathbf{KD}$$
 (42)
- With the internal forces and the displacements known other element related information such as stresses and strains can be calculated.

Appendix C Strain Energy Calculations

The strain energy ratio of each element is used as driver in the optimization principles. This appendix will define how the strain energy is being calculated for each member and force type. A short derivation is being provided.

For normal force:

$$U_{i,N} = \frac{1}{2} F_i \Delta L_i \quad (43)$$

where

- $U_{i,N}$ = Strain energy due to axial forces
 F_i = The axial force on the element
 ΔL_i = Axial deformation of the element

With

$$\Delta L_i = \int_0^{L_i} \frac{F_i}{A_i E_i} dx \quad (44)$$

$$U_{i,N} = \int_0^L \frac{F_i^2}{2A_i E_i} dx$$

$$U_{i,N} = \frac{F_i^2 L_i}{2A_i E_i} = \frac{\sigma_i^2 V_i}{2E_i} \quad (45)$$

Where

- A_i = Section of the i'th element
 E_i = Young's modulus of the i'th element
 σ_i = Stress of the i'th element
 V = Volume of the element

For shear:

With

$$U_{i,V} = \int_0^{L_i} \frac{V_i^2}{2E_i I_i} dx \quad (46)$$

$$U_{i,V} = \frac{V_i^2 L_i}{2G_i A_i} = \frac{\tau_i^2 L_i A_i}{2G}$$

For bending:

$$dU = \frac{1}{2} M_i d\theta_i \quad (47)$$

With

$$d\theta = \frac{ds}{R}$$

$$R = \frac{EI}{M}$$

$$dU = \frac{M^2 i ds}{2E_i I_i}$$

$$(48)$$

For small deflections $ds=dx$, therefore for the entire beam:

$$U_{i,M} = \int_0^{L_i} \frac{M_i^2}{2E_i I_i} dx$$

$$U_{i,M} = \frac{M_i^2 L_i}{2E_i I_i} = \frac{\sigma_i^2 L_i I_i}{2E_i} \quad (49)$$

Where

$U_{i,M}$ = Strain energy due to bending

M_i = Moment on the element

θ_i = Rotational angle due to the moment

I_i = Moment of Inertia of the element

The total strain energy can be calculated as the sum of strain energies on the structure

$$\text{truss: } U_i = U_{i,N}$$

$$\text{beam: } U_i = U_{i,N} + U_{i,V} + U_{i,M} \quad (50) \quad (51)$$

Where

U_i = Combined strain energy of the element

Appendix D Overview of the grasshopper script.

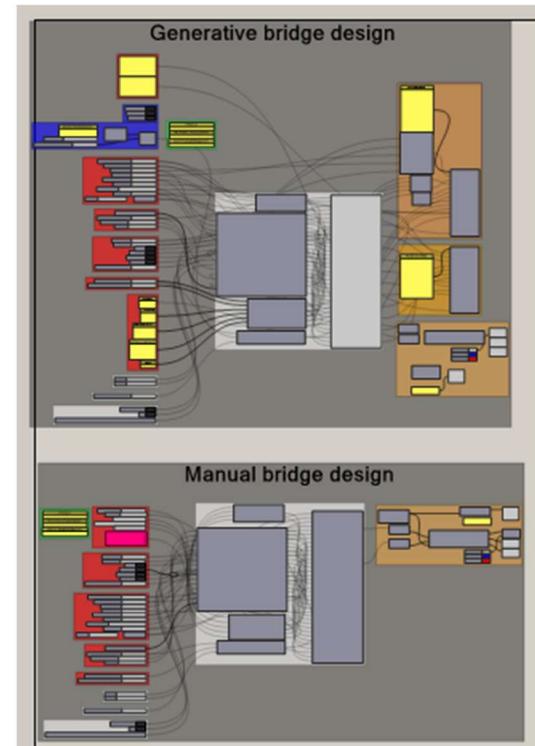
Usage of the grasshopper file

- Open the Grasshopper file for used in this thesis; You should be able to use to grasshopper code by now allowing you to generate bridge databases.

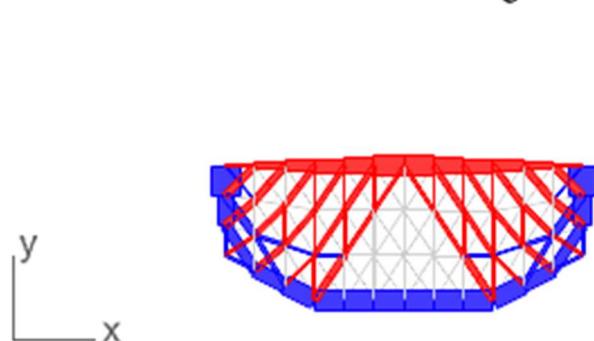
- You are able to use the generative framework in which you control which bridges are being generated by altering the panel input. Check the amount of variants this creates then set the maximum slider value to this (This slider is found on the blue background). Enable the storage, set the slider to 0, reset the storage and animate the slider (use amount of frames = amount of options) to generate a database with the variants applied by the panel input. The generative framework crossrefences the panel input options to create all available options between the presented parametric inputs. This results in lists with indices of parameter settings of which each index corresponds to a single bridge. (columns are the parameter options, rows are the parameters of a single bridge). By now indexing over this lists, you obtain the parameters making up each bridge. Therefore animating the slider used for indexing results generating all available bridges.

- The framework below the generative framework is the 'manual' framework. This framework allows you to explore individual variants, implying that you have to set each parameter and a single bridge results. The grasshopper script behind the 'manual' framework is the same as that of the generative framework only now single options are presented to the cross-referencing tool resulting in a single bridge as indexing option.

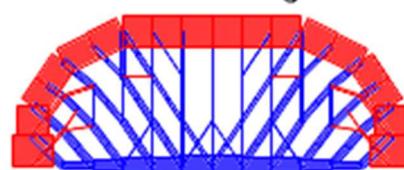
Grasshopper Top viewport



Generative bridge



Manual bridge



Appendices

Appendix E Additional results

Additional Result 1

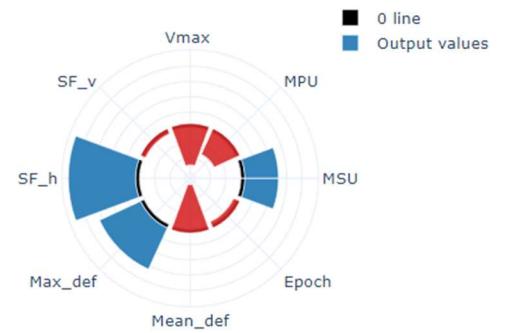
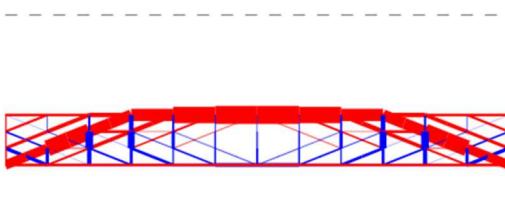
$$\begin{aligned} \min \quad & V_{max}, \quad SF_V + SF_h \\ \text{s.t.} \quad & \left\{ \begin{array}{l} A_p > 0 \text{ or } FM = 'Edge' \\ -1 < Domain_{1,2,3} < 1 \end{array} \right. \end{aligned} \quad (52)$$

Where

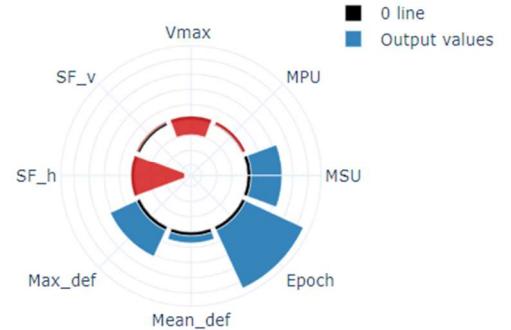
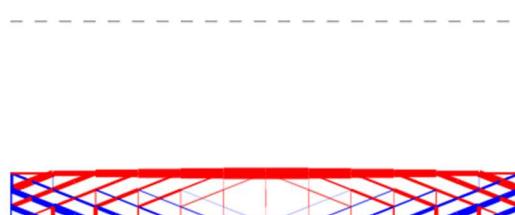
- $SF_V + SF_h$ = The combined vertical and horizontal support forced
- FM = Forced Members
- $Domain_{1,2,3}$ = The structural height of the three domain sliders.

Results:

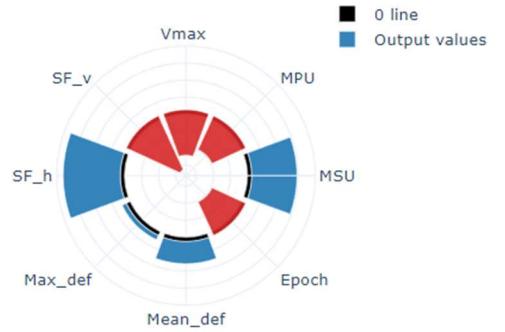
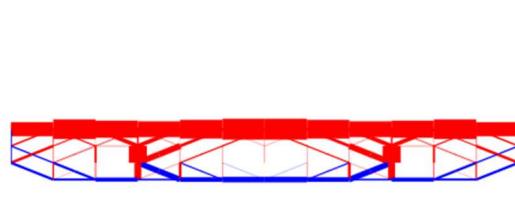
```
domain_1: 1
domain_2: 1
domain_3: 1
youngPen: None
a_p: 3216.9908772759
manu_type: truss
L_ManuP: Ledge
```



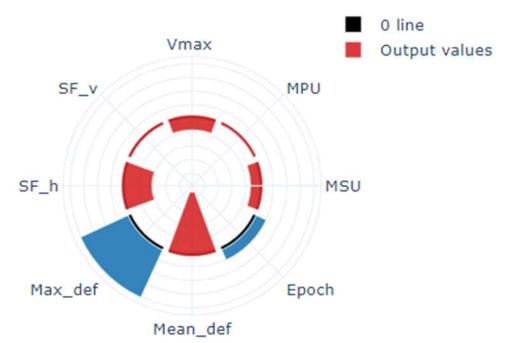
```
domain_1: -1
domain_2: -1
domain_3: -1
youngPen: None
a_p: 0.0
manu_type: truss
L_ManuP: Ledge
```



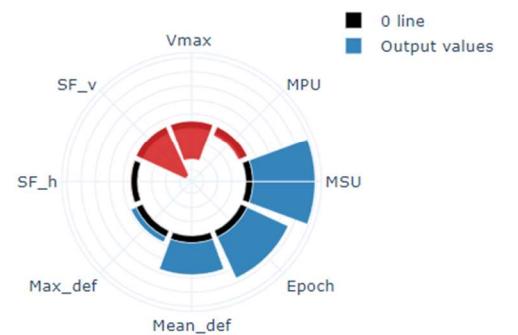
```
domain_1: -1
domain_2: -1
domain_3: -1
youngPen: None
a_p: 3216.9908772759
manu_type: truss
L_ManuP: None
```



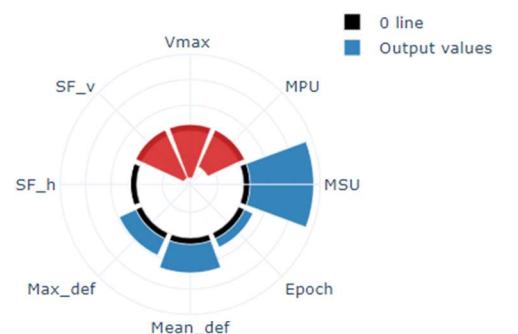
domain_1: 1
domain_2: 1
domain_3: 1
youngPen: Nefs
a_p: 0.0
manu_type: truss
L_ManuP: Ledge



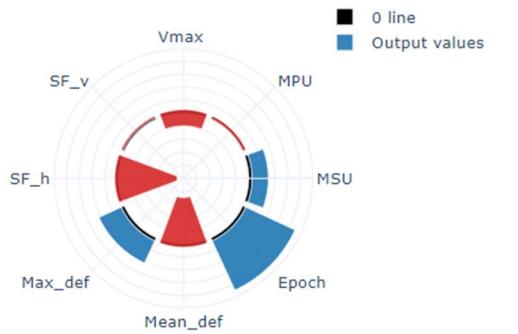
domain_1: 1
domain_2: -1
domain_3: -1
youngPen: None
a_p: 1963.4954084936
manu_type: truss
L_ManuP: Loadpath



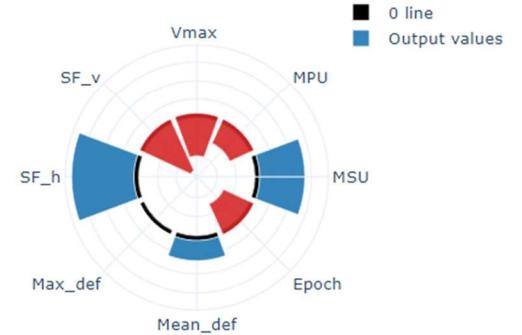
domain_1: -1
domain_2: -1
domain_3: -1
youngPen: None
a_p: 1963.4954084936
manu_type: truss
L_ManuP: Ledge



domain_1: 1
domain_2: 1
domain_3: 1
youngPen: None
a_p: 0.0
manu_type: truss
L_ManuP: Ledge



domain_1: 1
domain_2: -1
domain_3: -1
youngPen: None
a_p: 3216.9908772759
manu_type: truss
L_ManuP: Ledge



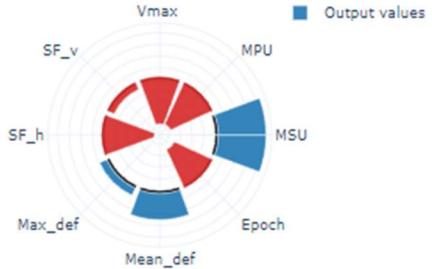
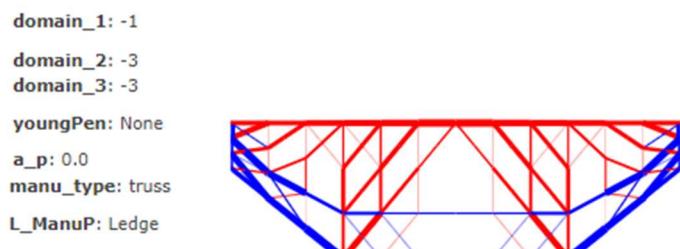
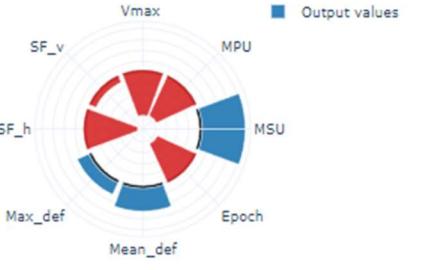
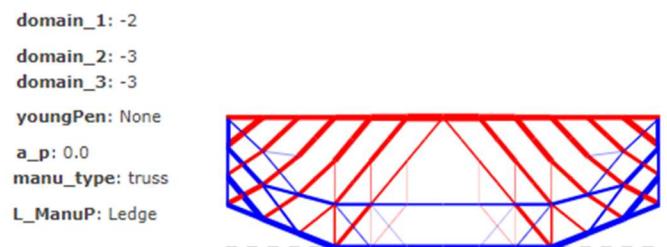
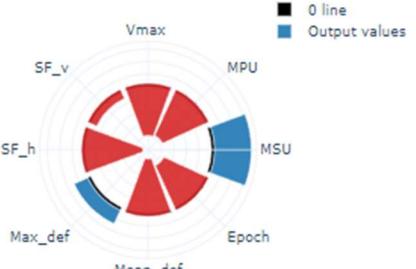
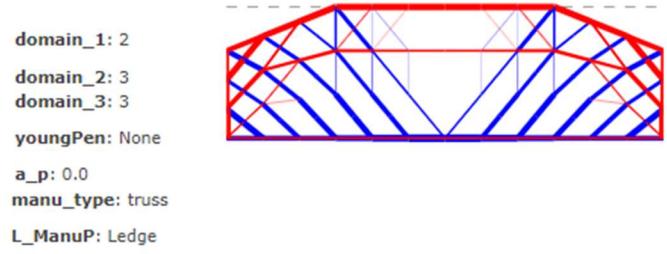
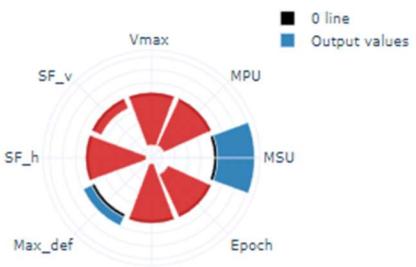
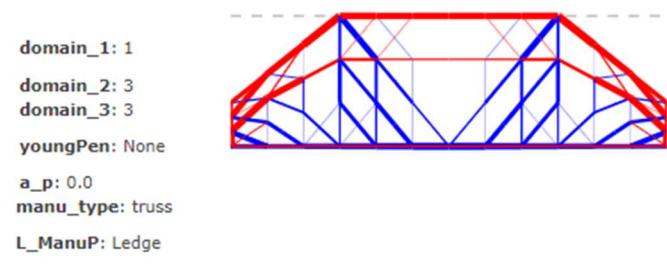
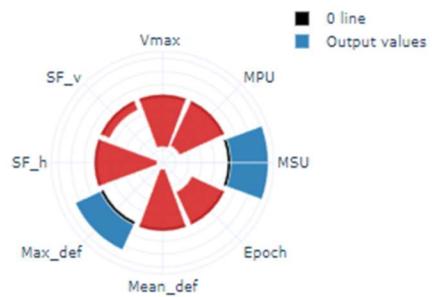
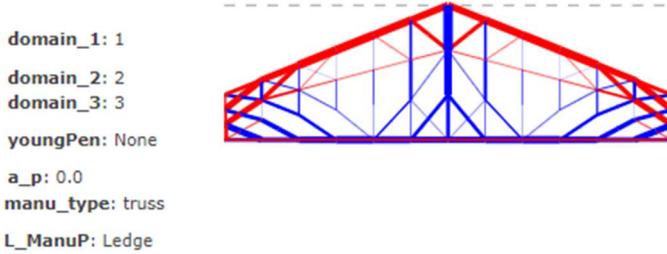
Appendices

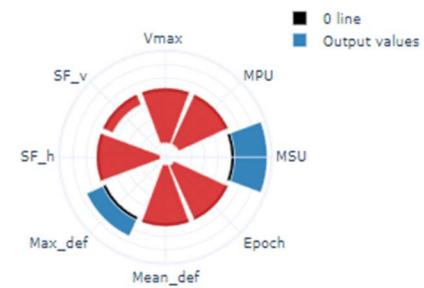
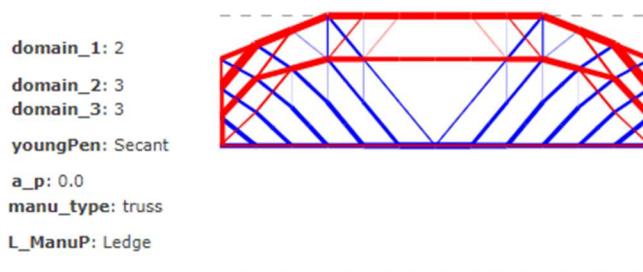
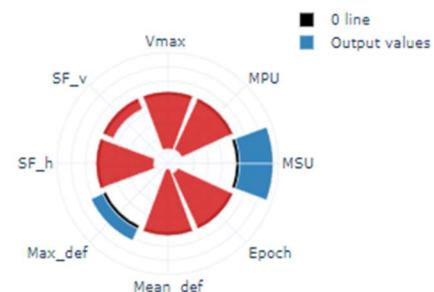
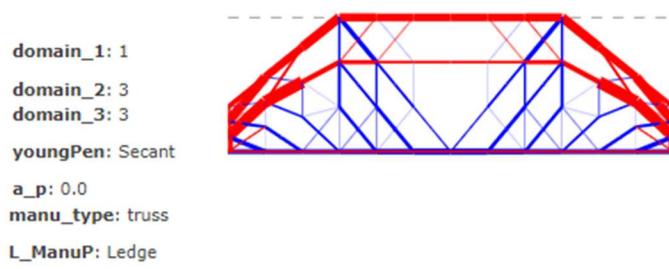
Additontial result 2

$$\min V_{max}$$

s.t. $FM = 'Edge'$

(53)

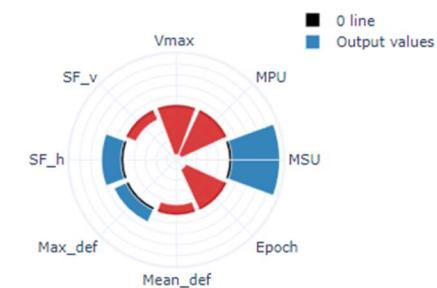
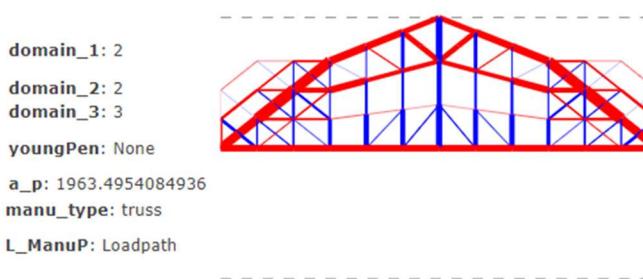
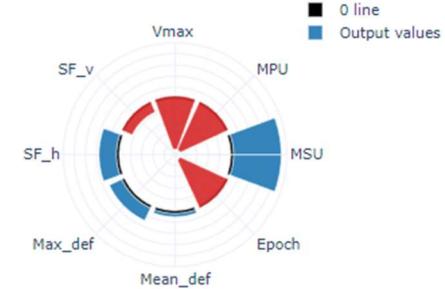




Additional Result 3

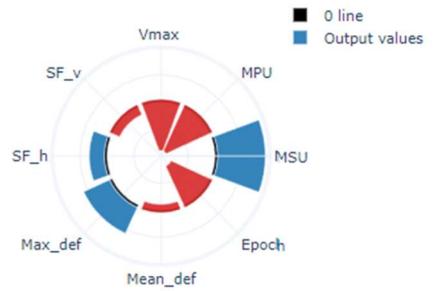
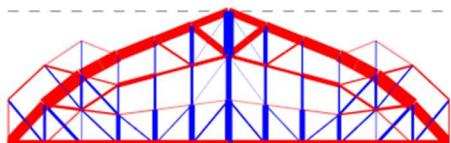
$$\begin{aligned} & \min V_{\max} \\ \text{s.t. } & \left\{ \begin{array}{l} A_p:r = 32 \text{ [mm]} (\text{high value of prestres}) \\ FM = \text{'Loadpath'} \\ Domain_{1,2,3} > 0 \end{array} \right. \end{aligned} \quad (54)$$

Results:

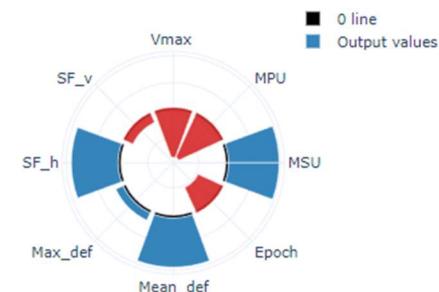
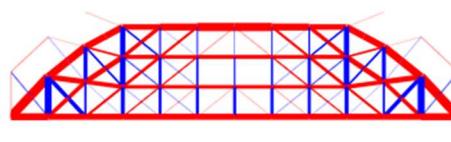


Appendices

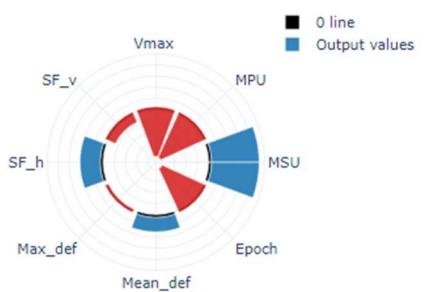
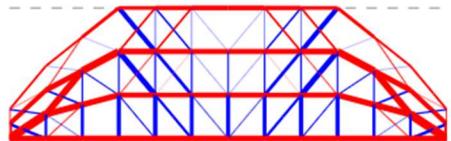
domain_1: 3
domain_2: 2
domain_3: 3
youngPen: None
a_p: 1963.4954084936
manu_type: truss
L_ManuP: Loadpath



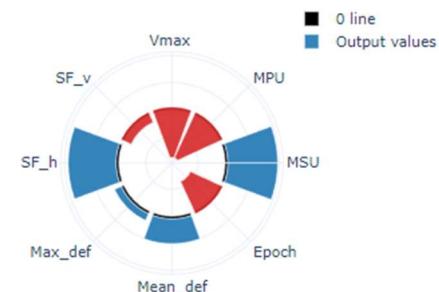
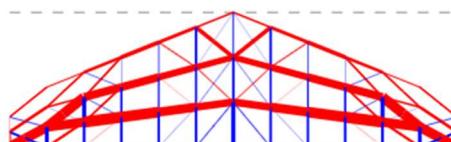
domain_1: 3
domain_2: 2
domain_3: 2
youngPen: None
a_p: 3216.9908772759
manu_type: truss
L_ManuP: Loadpath



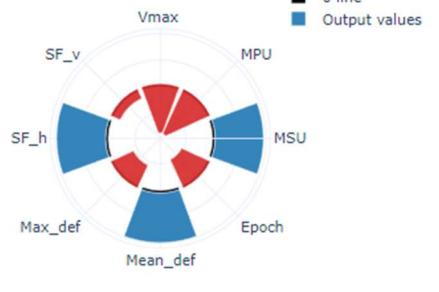
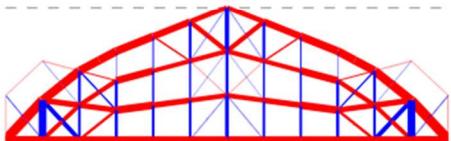
domain_1: 1
domain_2: 3
domain_3: 3
youngPen: None
a_p: 1963.4954084936
manu_type: truss
L_ManuP: Loadpath



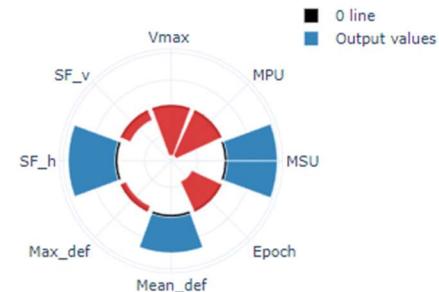
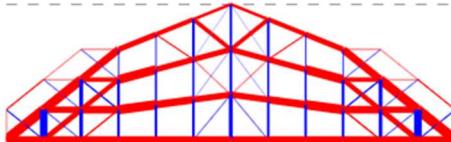
domain_1: 1
domain_2: 2
domain_3: 3
youngPen: None
a_p: 3216.9908772759
manu_type: truss
L_ManuP: Loadpath



domain_1: 3
domain_2: 2
domain_3: 3
youngPen: None
a_p: 3216.9908772759
manu_type: truss
L_ManuP: Loadpath



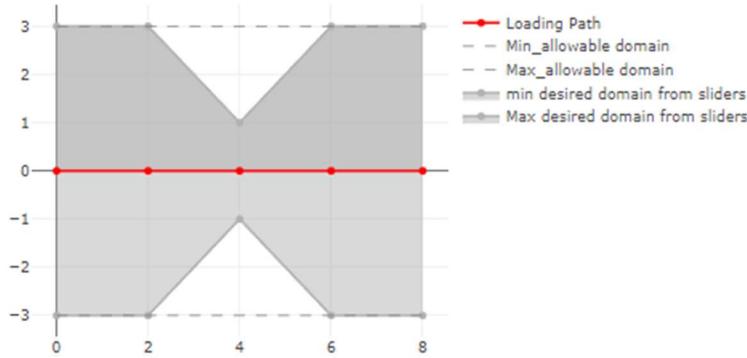
domain_1: 2
domain_2: 2
domain_3: 3
youngPen: None
a_p: 3216.9908772759
manu_type: truss
L_ManuP: Loadpath



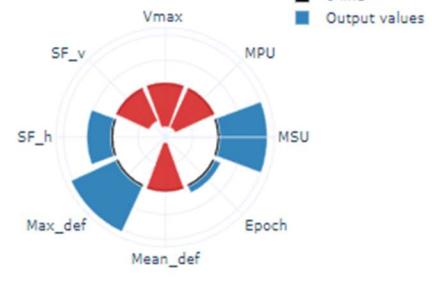
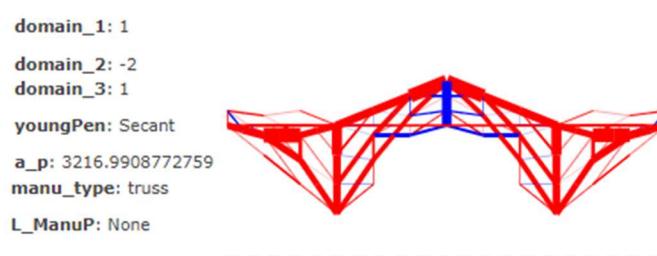
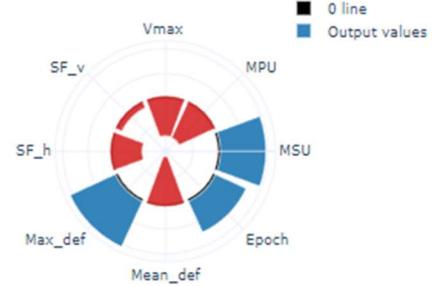
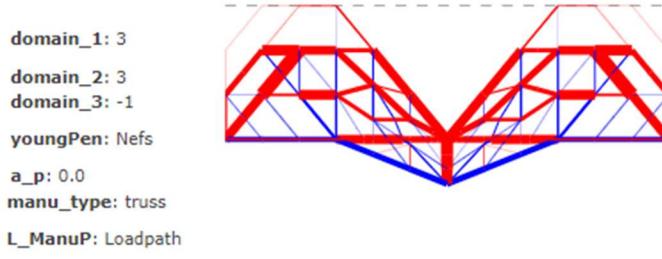
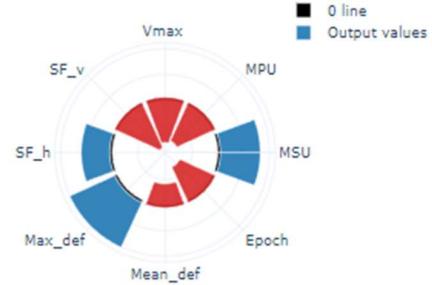
Additional Result 4

$$\min V_{max}$$

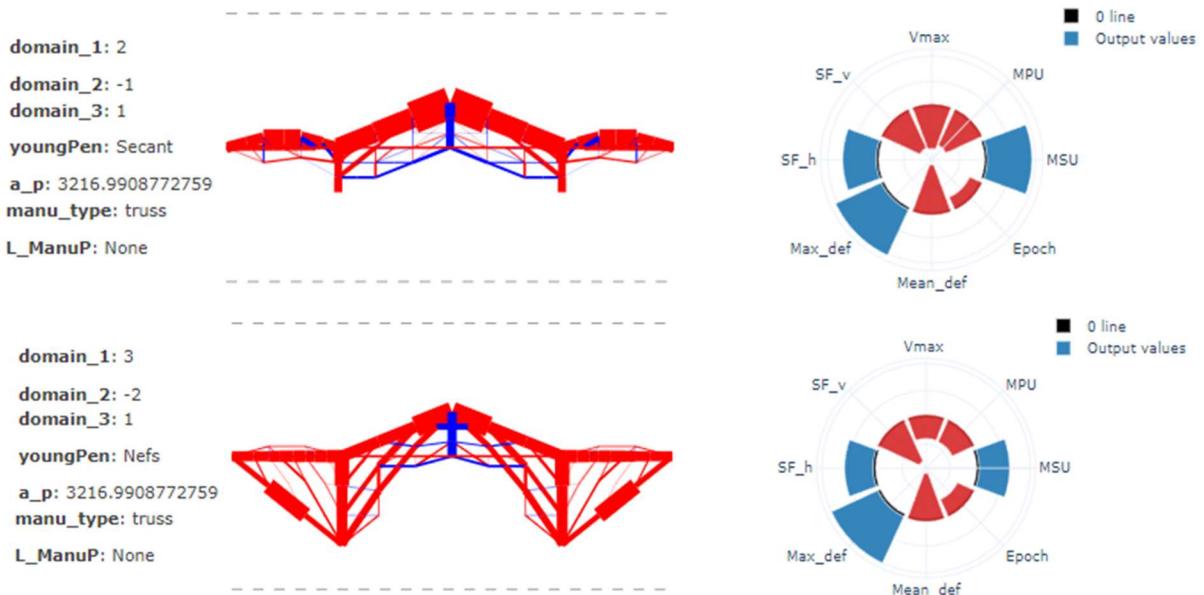
s.t: $1 < Domain_3 < 1$ (54)



Results:

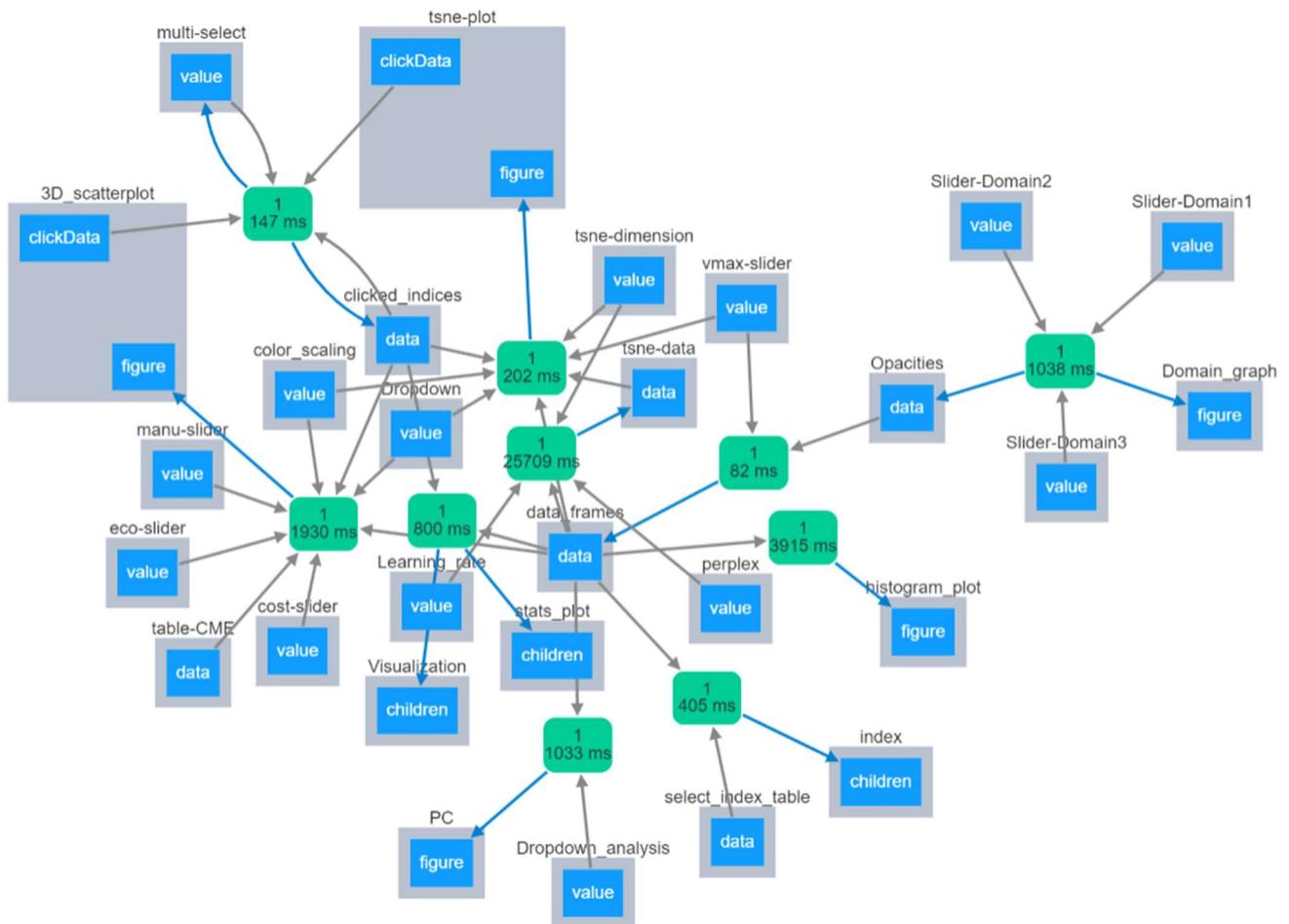


Appendices



Appendix F Overview callback functions

Dash uses callback functions to trigger events throughout the dashboard. A schematization of the interaction between callbacks is shown below:



Appendix G Overview of the dashboard

Design space analysis of a 3D printed concrete bridge

This file can be used to explore the design space of 3D concrete printed bridges. When running the dashboard it first manipulates the dataset by ranking all output variables. The following variables are included: Input Domains(1-3), amount of previous, manufacturers variable (with a minimum material constraint to certain elements), type of young's modulus realization, Output: Vmax (maximum volume), Max and mean deflection, SF_y & SF_h support force vertically and horizontally, Epoch optimization objective (summed strain energy), MSU (mean stress utilization), MPU (mean platenwidth utilization).

Set constraint domain

Filter database

Domain sliders:

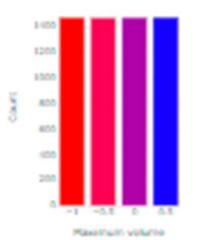
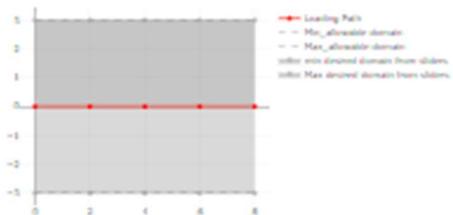
- Domain1
- Domain2
- Domain3

Select to filter the database

Use the sliders on the left to filter the database. By changing the domain sliders, datapoints will be filtered from the database, plotting them transparent in the plots of the following block. This allows the user to analyze constraint design spaces. The histogram plot shows how the maximum bridge volume is distributed after the ranking process. By adapting the slider one can remove the outliers, these datapoints will be completely excluded from the visualizations.

Set Desired Domain

Histogram of ranked maximum value



Design Space Exploration

Design Space Analysis

Adapt the visualization of the plots

Select category for markershape in visualization

Select as input c:

Select category for color in visualization

Vmax

Adapt the visualized bounds

Cost bound

Eco bound

Manu bound

Manipulate the t-SNE plot

Dimensions of t-SNE

Learning rate of t-SNE

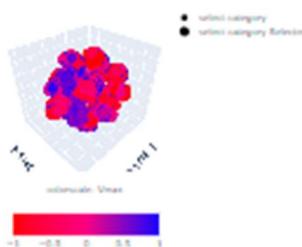
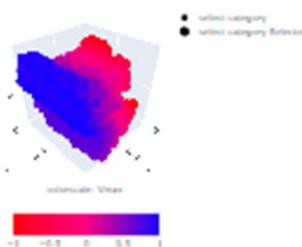
Learning rate of t-SNE: 10000

Penalty of t-SNE: 10

The scatterplots below can be used to explore parameters and find desired bridges. The plot on the left, plots the objective space according to the table below. The graph on the right is a t-SNE plot, which plots all input parameters in a two or 3 dimensional space. Clusters of similar input will form, which can be analyzed by plotting various output parameters as colorscale to it. The sliders on the left allow to manipulate the visualization, by setting different colormaps, different marker types and changing the t-SNE calculation. The perplexity can be changed to change the ratio between preserving local or global structures, while the learning rate determines the amount of change allowed to be made during each iteration. A higher learning rate causes more clustering.

Objective space

t-SNE plot of the design space



Set weights for the objective space

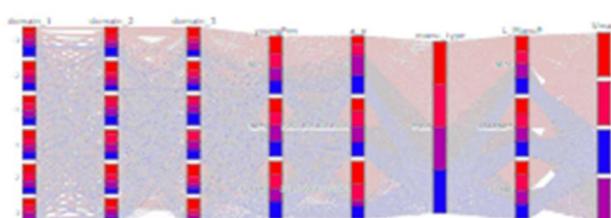
Index	Cost	Manu	Eco
PP1	0	0	0
PP2	0	0	0
Vmax	0	0	0
SF_y	0	0.5	0
SF_h	0	0.5	0
Max_def	0	0	0
Mean_def	0	0	0
Epochs	0	0	1
R_P	0	0	0

Parallel categories plot

Add Output

Vmax

Use the interactive parallel categories (PC) plot to find relations between in- and output. The distribution plot provided above can be used to filter elements from the PC plot. The bins shown in the distribution plot are also displayed as bins in this plot. The dropdown menu on the right allows to add the desired output values. Multiple can be added. The last one in the list determines the colormap.



Appendices

This component visualizes the datapoints selected in the two scatterplots. The indices visualized are shown in the dropdown menu to the left. Additional datapoints can be added in the dropdown menu. The table below it adds to find the index of a specific bridge based on the input selected in the table. The input making up visualized bridge will be presented next to it. The output values are plotted in the radial bar plot. Due to the ranking all output is between -1 and 1. Relative high values are therefore plotted in red whereas relative low values are plotted in blue. No bar size will indicate a medias performance.

Find index of for given input data:

domain_1	domain_2	domain_4	youngness	A_P	shape_type	L_Material
-8	-8	-4	None	0.0	Truss	None

(0)

Selected bridge, index = 0

domain_1: -2
domain_2: -2
domain_3: -2
youngness: None
A_P: 0.0
shape_type: truss
L_Material: None

