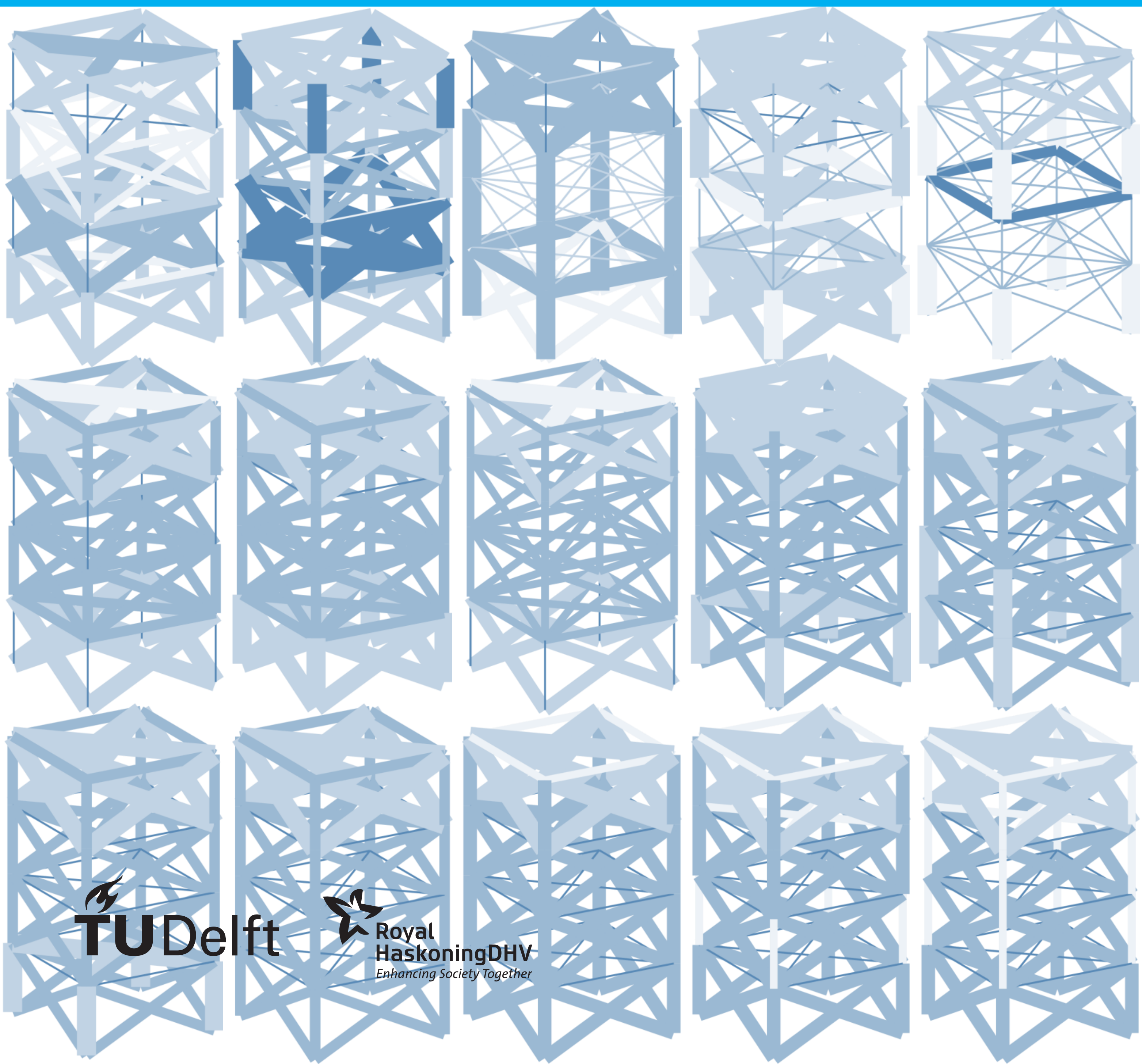
Buildable Design in Optimisation of Steel Skeletal Structures: A Comparison of Existing and New Methods for Finding the Best Solution with Low Diversity

T.R. van Woudenberg



TU Delft

Royal HaskoningDHV
Enhancing Society Together

Figure 1 on cover page: iteration results for cardinality constraints method in grouped optimisation of 72-bar truss structure.

Buildable Design in Optimisation of Steel Skeletal Structures: A Comparison of Existing and New Methods for Finding the Best Solution with Low Diversity

T.R. van Woudenberg

in partial fulfilment of the requirements for the degree of

Master of Science
in Civil Engineering

at the Delft University of Technology,
to be defended publicly on 2nd of July 2020 at 10:00.

Thesis committee:	dr.ir. F.P. van der Meer	TU Delft (chair)
	ir. R. Crielaard	TU Delft
	dr.ir. J.L. Coenders	White Lioness technologies
	ir. S.B. Cox	Royal HaskoningDHV
	ir. J.M. Houben	TU Delft

An electronic version of this thesis is available at:
http://resolver.tudelft.nl/uuid:a977598d-700f-4738-b750-4006a9712f64

The input data and datasets generated during this thesis are available at:
http://doi.org/10.4121/uuid:4e32b29f-6647-4a36-9ea1-8931c88f8864

Preface

Ever since the start of my study in civil engineering, I have had a great interest in the mechanics of civil structures. Civil structures are masterpieces in their combination of structural performance with architecture and functionality; the most world-famous structures conquer the forces of nature with an elegant and light structural design, incorporated in daring architectural creativity. Unfortunately, these structures seem to be rare, and the practical field of civil engineering appears to be dominated by building guidelines, budgetary limitations and replication of older designs. A structural engineer should not focus on these aspects, but should be allowed to use his engineering knowledge to the best. Therefore, my goal is to ease the structural design process, so that the focus of the structural engineer can be on the fundamental mechanics of a design. While nowadays, a civil engineer heavily relies on his personal design experience for generating ideas, the future civil engineer should use his experience as a guide on generative design tools. The development of these tools requires high expertise, not only in the field of structural design, but also in applied physics, mathematics and computer science.

In this thesis I took the opportunity of diving into the tools of generative design. It started as a research project in an, for me, unfamiliar field of science. Nonetheless, I received much knowledge, joy and satisfaction from this process. In the end, this thesis led to the developed of a new generative design tool, which generates both buildable and optimum designs in terms of weight, costs and sustainability. I hope that this tool is of valuable use and an inspiration for further research.

I would like to thank my thesis committee for their support. Thank you, Simon Cox, for your enthusiasm and our discussions at the coffee table. Frans van der Meer, you were a very pleasant chairman of the committee, providing me with practical guidance and asking decisive and critical questions. Roy Crielaard and Jeroen Coenders were my experts on optimisation; your inspirational ideas brought me to the thesis' topic, and the in-depth reviewing of many draft versions steered me into the right direction. Lambert Houben, thank you for your flexibility and the coordination of the administrative parts of my thesis.

Furthermore, I would like to thank all my colleagues and fellow interns at Royal HaskoningDHV for the great experience I had at the office, although my internship ended prematurely because of the coronavirus. In particular, I would like to thank Erik, Tianxiang, Robin, Harm, Tijl, Thomas, Geert, Geerte and Robin. Finally, I would like to thank my family and friends, especially my girlfriend Saskia and friends Fleur, Annelies and Wouter, for their feedback on the report and their support and encouragement throughout the process.

Tom van Woudenberg
June 2020

Summary

In the design of steel structures, optimisation methods promise cheap, light and sustainable structures. However, the resulting designs tend to have a high diversity of profiles, making them unbuildable. Furthermore, the optimisation problem is mathematically complex, leading to a long and potentially unsolvable optimisation process. Grouping methods solve both issues by finding the optimum solution, for which the number of distinct profiles is limited.

Multiple grouping methods exist in literature, and it is not known which is the best: the methods have not been applied on the same problems, and the computational effort has not been compared. This gap in literature leads to the following question: “Which method for grouping can find the lightest and cheapest steel structure with minimal computational effort?”

To answer this question, a comparison of the existing grouping methods is made on their theoretical and numerical performance. The theoretical comparison comprises the size and properties of the search space, and the number of additional calculations. The numerical comparison consists of weight optimisation of eight benchmark problems. For each structure and method, the weight of the solution and corresponding computational effort is evaluated.

Manually grouping of members, which is the most popular grouping method, relies on the engineer’s expertise and rules of thumb. This method requires no additional calculations but in general fails to find the optimum grouping for a light or cheap structure. Other existing methods include the geometry, axial force distribution or an ungrouped result in their grouping process, or adapt the optimisation problem. Of these methods, only the cardinality constraints method is guaranteed to potentially find the lightest grouped design, while reducing the search space for a small number of groups. However, it creates many local optima, which increases the complexity of the search space.

In the aim of finding a grouping method which creates a simple and small search space and has low computational effort, the fully stressed combinatorial search method is proposed. In this method, the grouping is found by a combinatorial search, which evaluates the estimated weight or costs of a restricted set of groupings based on the weight per unit length of the members of a fully stressed design. Then, optimisation of a small and simple search space finds the corresponding optimum profiles. These steps are repeated, in which the fully stressed design uses the result of the previous optimisation as its reference design. The loop repeats until the grouping is unchanged, or the result diverges.

In all numerical experiments, the new method gave results with a low weight, while it kept the computational effort to an acceptable level. It gave the lightest design for four out of eight problems and showed high certainty for converging to the lightest design in two problems. For the other two problems it performed second best. Conclusively, this method is the best available grouping method for steel structural optimisation.

In case of cost optimisation, the new grouping method can efficiently find the optimum design including the optimum number of groups; the new method converges to cheaper design with less computations than in the case no grouping is applied. For a real-life case-study, the costs of a design were reduced with 7.3% compared to a manually grouped design and with 19.6% compared to the conventional design process.

I suggest that further research focusses on further development of the new grouping method as proposed improvements can be made on the initial design, and the number of computations in the combinatorial search and fully stressed design. Moreover, the effectiveness of a suggested simplification of the new method should be investigated. This would allow application for engineers who are not able to apply a mathematical optimisation method. For practical application, incorporation of building codes and cost functions with a well-defined scope are desired. Finally, utilisation of the grouping methods in other applications is possible, but the performance of the methods should be evaluated per application.

Table of contents

1 Introduction

The design of steel structures is everyday work for the civil engineer, but it has room for improvement. In a structural design, an engineer aims at making the lightest, cheapest or most sustainable design. This task can be performed and improved by an optimisation method. However, this is rarely done in practice, because optimisation has some practical drawbacks: the high diversity of the solution and the inability to limit this diversity. This issue must be solved to allow the design of buildable structure.

To do so, the basics of optimisation are introduced first. This chapter explains the functioning and added value of optimisation, starting from its history. Subsequently, its application to steel structures is introduced, and the relation between weight, costs and diversity is analysed. Then, the research question and subquestions are presented, together with the scope of the research project. Finally, an outline of this report is given.

1.1 Background

Optimisation is a large field of research, of which this chapter only covers the relevant aspects. For a more in-depth introduction, I would recommend the course *ME46060 Engineering Optimization*, given by M. Langelaar, and the book *Introduction to optimum design*, written by J.S. Arora (2017).

1.1.1 History

Optimisation is the mathematical process of finding the best solution of a problem, in which the definition of best solution can be, for example, minimum weight or costs. It is an active field of research, and well-established methods have been developed. Nonetheless, the application to the industry of civil engineering is limited. This is a pity, as this design process has high potential for a more optimal design of both simple and complex civil structures. Other design sectors have incorporated optimisation into their design process, with impressive results. Applications have ranged from weight savings on an aircraft wing to the reduction of material use of packaging products (Langelaar and van Keulen 2019). The industry of civil engineering should aim for optimality too, as the structural performance of civil structures is closely linked to safety and global well-being. This close link demands for the best possible design and thereby the application of optimisation. Yang et al. (2016) stated this demand as follows:

"In civil engineering, …, the best possible balance between security and economy must be found without risking lives. …"

Until now, optimisation in civil structures has primarily been restricted to research. Optimisation of structures started in 1960 with the famous research by Schmit (1960). He took the three-bar truss structure shown in Figure 2, subjected to two independent load cases. The thinking at the time was (and nowadays still is to some extent) that the best structure would be a structure in which each bar is loaded to its limit in at least one load case. Using optimisation concepts, he showed that in the lightest possible structure, the vertical bar is not fully stressed in any of the two load cases (Schmit 1960; Salajegheh and Vanderplaats 1993).

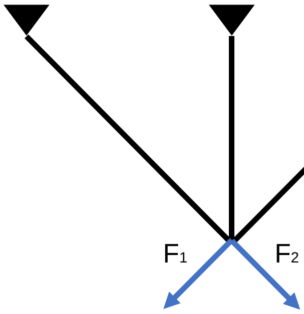


Figure 2 – Three-bar truss Schmit subjected to two independent load cases.
Adaptation of original figure in literature (Vanderplaats 1993).

Optimisation has been present in the science of engineering since then. These methods have found impressive structures and design concepts too, which would not have been thought of by an engineer. An example is the split-pylon concept bridge, as shown in Figure 3. This new type of bridge design was discovered with the use of an optimisation method. The bridge design can span much greater distances than conventional bridges, compared to their weight (Fairclough et al. 2018)

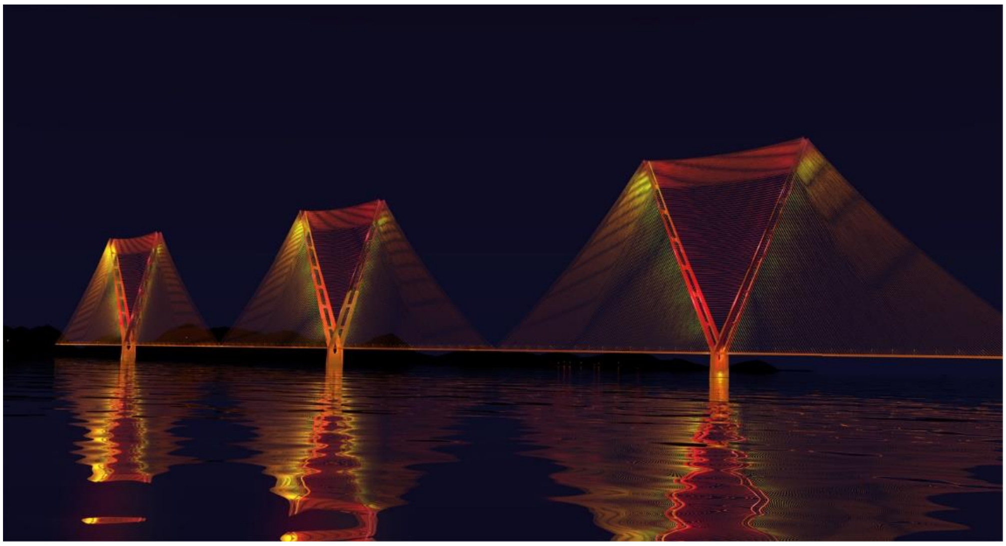


Figure 3 - Example optimised weight structure, split-pylon concept bridge which can span high distances relative to its weight.
Figure taken from literature (Fairclough et al. 2018).

1.1.2 Conventional and optimisation design process

Optimisation methods change the conventional design process by incorporating mathematical procedures, which reduces the required input of an engineer. In the conventional design process, the experience of a structural engineer is vital: after an engineer defines his problem, he estimates an initial design, performs a structural analysis and checks the results on his demands. If the engineer thinks the design can be improved, he adapts the design based on heuristics. The engineer repeats this until he is satisfied. This process is shown in Figure 4. The blue-shaded blocks show in which phase an engineer provides input based on his experience.

It should be noted that more sophisticated design models exist for the conventional design process. These models show, for example, the interaction between multiple phases of design and interaction between different stakeholders. However, the experience of the engineer is crucial in all conventional models.

For the optimisation design process, the repetitive tasks of the engineer are automated: mathematical optimisation methods replace the engineer in altering the design. As a result, the influence of the engineer shifts to defining the optimisation problem. Mathematics takes over the iterative task of finding the best solution, and this process is stopped when mathematical convergence criteria are met. As the optimisation design process requires no input from an engineer during the iterations, many solutions can be evaluated. The optimisation design process is shown in Figure 4 as well. Again, the steps in which the experience of the engineer is needed are shaded blue.

For simple problems, the conventional design process can be good enough and might results in a proper design. However, for more complex problems, an engineer cannot oversee all possible solutions, or his engineering heuristics fall short. Therefore, he cannot give a guarantee for finding the optimum result.

It should be noted that all optimisation methods require a mathematical description which describes optimality. As a result, some problems might not be well suited for this optimisation design process as mathematics cannot formulate all aspects of a design. For instance, the beauty of a building is difficult to express in a mathematical format. Other design criteria are more suitable for optimisation, like weight, costs or environmental impact.

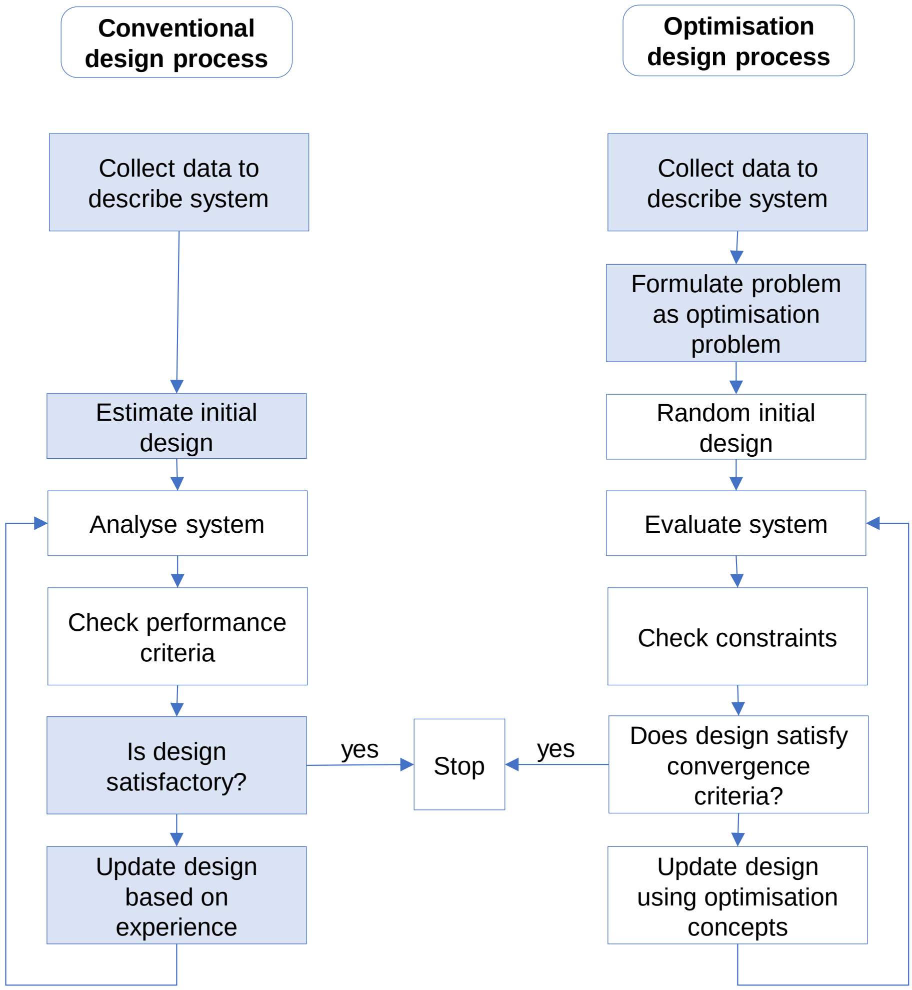


Figure 4 - Conventional design process versus optimisation design process: the repetitive tasks are automated. Blue shading shows the need for experience of an engineer.
Adaptation of original figure in literature (Arora 2017).

1.1.3 Basics of optimisation

A few basics of optimisation are required to understand the functioning and added value of optimisation methods: the standard design optimisation model, local and global minima, and search method (Papalambros and Wilde 2000; Arora 2017).

Standard design optimisation model

The first essential tool is the standard design optimisation model. This model is the mathematical description of optimisation problems in design. It is shown in equation (1.1) to (1.3):

Minimise the objective function:

$$f(x_i) \tag{1.1}$$

With design variables:

$$x_i = \{x_1, x_2, \dots, x_n\}$$

And equality and inequality constraints:

$$g_j(x_i) \leq 0 \qquad \text{for } j = 1 \dots m \tag{1.2}$$

$$h_l(x_i) = 0 \qquad \text{for } l = 1 \dots p \tag{1.3}$$

The standard design optimisation model includes an objective function f which has to be minimised. This objective function can be everything expressible in a mathematical formula: weight, cost, embodied energy, environmental impact and many more. This function is dependent on design variables x_i, *which* are the options available for all n parts of the design like material, geometry or size. The set of possible solutions of all design variables is called the design space or search space. The equality g (1.2) and inequality constraints h (1.3) are defined to be equal to zero or negative. These constraints bound possible designs to demands on the design, like maximum weight, strength limits or minimum dimensions. The set of solution of design variables which meet all constraints is called the feasible set, which is a subset of the search space. An infeasible solution lies outside this set and is not a valid solution to the optimisation problem.

Local and global minima

The second important concept of optimisation is the definition of a minimum. A global minimum is the solution of design variables which evaluates to the minimum value for the objective function while meeting the constraint conditions. On the other hand, a local minimum is a solution of design variables for which only the objective function values in a small neighbourhood are equal or bigger. Figure 5 shows an example of a local and global minimum for one design variable.

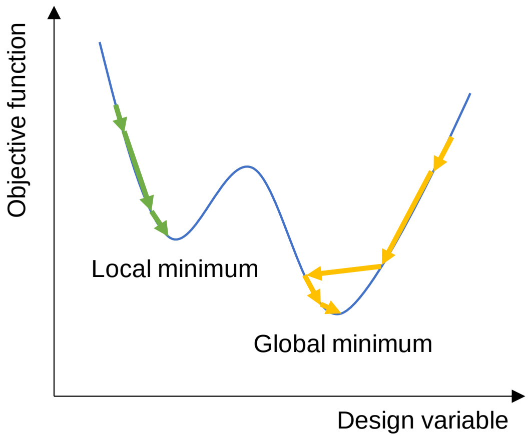


Figure 5 - Example of local and global minima, and iterative search by search methods. Individual searches can find a different minimum.

Search methods

The final basic concept of optimisation is the search method. As the search space is in general too big to evaluate for all options, search methods iteratively explore the search space by searching in the neighbourhood of their best-known solutions. The solution gradually become better and better during the process, and minima can be found without evaluating all options. However, many search methods rely on stochastics. This causes search methods to show inconsistent results when applied multiple times on the same problem. Figure 5 shows the iterative behaviour of search methods and the possibility of converging to different minima depending on a random start point.

Many different strategies and variations are available for search methods, which are case-specific and not generally applicable. However, all these methods have one thing in common: the characteristics of the search space influences the ability to find the global minimum. In general, the bigger the search space and the more local minima there are, the harder the problem is to solve.

1.1.4 Optimisation problem steel structural design

The standard design optimisation model is written in a specific form for steel structural design (Arora 2017). In most applications, the objective function is defined as the weight or cost of the structure. The design variables control sizing of the members, influencing both weight and costs. The available member sizes are not free, but in most cases this choice is limited to a set of standard steel profiles. Finally, design constraints are set by inequality constraints on stresses and displacements, while equality constraints are not used. The stress constraints can include, for example, yield constraints and buckling constraints. Maximum displacements at midspan might be a possible displacement constraint. These stresses and displacement are calculated with a structural analysis.

Weight optimisation

In weight optimisation, the objective function is the weight of the structure. In general, it has few local optima, which is best explained with a simplified representation of the search space, as is shown in Figure 6. The plot shows both the weight of a structure and the constraint function as a function of the design variable.

The design variable represents the choice of profiles for the members: a low value indicates light members, and a high value indicates heavy members. The weight of the total structure gradually increases with heavier members, as indicated with graph of the weight of the structure.

The constraint function is shown with the green line, of which the side with the thin line represents the feasible and negative domain, so the top right part of the graph with heavy profiles. For the bottom left part of the graph the designs correspond to a constraint function which is bigger than zero; for infeasible design the structure fails the beams are sized too light. The line represents the case in which the constraint function is zero, and the design is on the limit of feasibility.

The yellow cross indicates the global optimum. This solution is a local optimum because moving to left or right makes the design infeasible or heavier, and it is a global optimum because it is the only optimum. As only one local optimum exists, this problem is easy to solve.

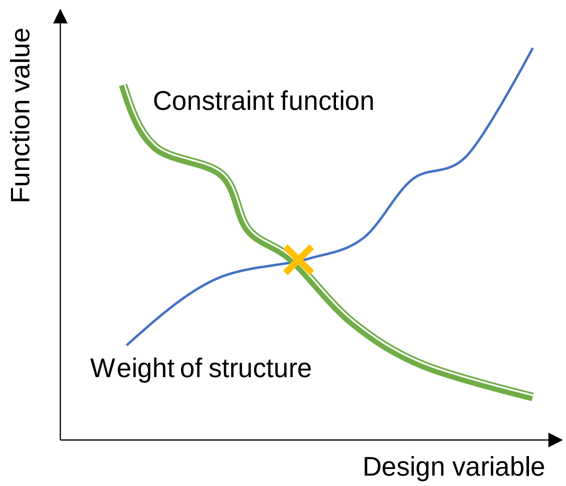


Figure 6 - Example search space weight optimisation. Yellow cross indicates global optimum, no local optima are present.

Cost optimisation

In the case of the objective function being a cost function, the problem converts in a cost minimisation problem. This cost function expresses the total building costs of the structure as a function of the structural design. The weight is one of the variables in this cost function, but other factors like fabrication, transport and erection costs can be included in this cost function as well.

Now, the choice for a heavier profile, might reduce the final costs, as it might result in a solution with less distinct profiles. Therefore, these cost functions result in a complex search space with many local optima, which causes struggles for an optimisation method (Adeli and Sarma 2006). This complexity is shown in Figure 7, in which multiple local optima appear; each local optima is in a local minimum of the cost function, or at a border with the constraint function.

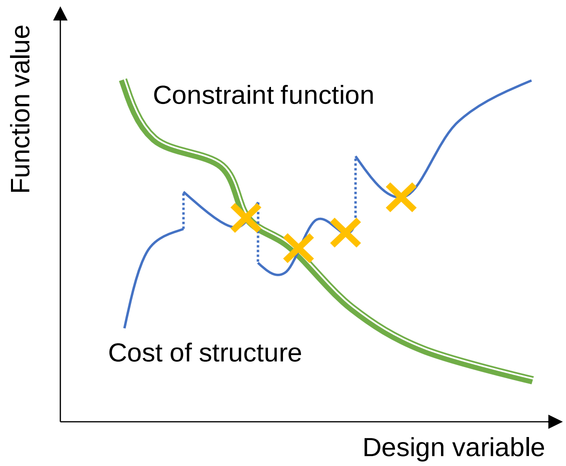


Figure 7 - Example search space cost optimisation. Yellow crosses indicate multiple local optima.

The cost function should contain all aspects which contribute to the costs of a structure. These factors are depending on for example, place, time and economy (Tizani et al. 1996). Because these cost factors are hard to quantify, the perfect cost function does not exist. However, attempts have been made to find it, as the minimum cost structure is the primary goal in industry (Pavlovčič et al. 2004; Haapio 2012; Ajouz 2018).

Difference in solution

In general, the optimum minimum weight solution of a structural has a different profile for every bar, as each bar is loaded differently. Some bars might have the same profile, because the diversity of the profile database is not big enough to provide each individual bar with the exact required resistance. The solution of the weight optimisation problem for a cantilever truss beam is shown in Figure 8. The thickness of each member stands for the weight of the profile.

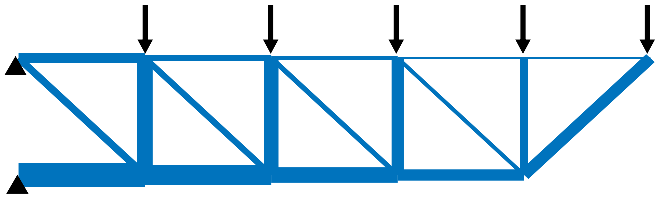


Figure 8 – Example minimum weight design: almost every member has a different profile.

The minimum cost design can be different than the minimum weight design. For the cantilever truss beam, one of the local cost optima might be the design shown in Figure 9. In this design, the number of distinct profiles reduces, which reduces costs.

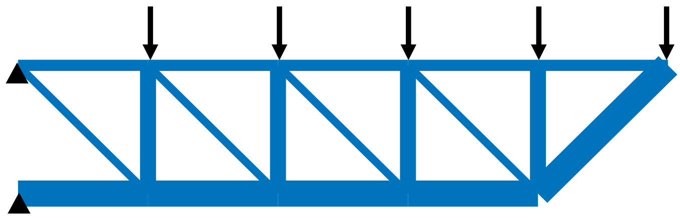


Figure 9 – Example minimum cost design: few unique profiles.

1.1.5 Principle of commonality

Both the optimum weight and optimum cost solution are used to evaluate a design. However, as shown by the cantilever truss beam, the optimal design of both procedures is different. A popular assumption is that the main difference is caused by the diversity of profiles in the optimum weight design. This observation is known as the principle of commonality: The fewer distinct components a structure has, the lower the costs are. This principle has been referred to in literature with different names: cardinality, location-allocation problems, standardisation theory, and minimisation of wasted material from overdesign (Templeman 1988; Reitman and Brent Hall 1990; Chan 1992; Biedermann and Grierson 1996; Gutkowski 1997).

The principle of commonality is true because limiting the number of distinct profiles results in a decrease in the costs of purchasing, storing, fabrication and detailing. Some examples of origins of the decrease in costs are:

- Bulk discounts
- Easier administration of stock
- Easier fabrication
- Reduce of chance on errors
- Easier quality control
- Less unique connections

The principle of commonality does not hold for very few distinct components; in the limit case that one profile is chosen, it is a heavy and expensive profile. Consequently, an optimum exists for a certain number of distinct profiles for which light profiles can be chosen, but the diversity is low enough to prevent high costs. Figure 10 shows conceptually the raw material costs, the additional costs for a high diversity and the sum of these two for a range of distinct sizes. For the total costs, an optimum number of groups is visible.

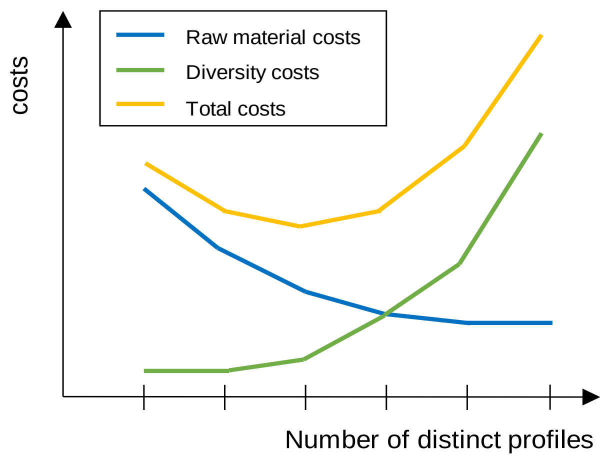


Figure 10 - Comparison of costs, adaptation of original figure (Gutkowski 1997). Total costs are sum of diversity and raw material costs, which creates an optimum for a limited number of distinct profiles.

Although the principle of commonality is logical, it is hard to model; diversity costs are hard to specify explicitly. This complication was acknowledged by Templeman (1988) too:

> *"…whereas material costs are easy to calculate accurately, the savings afforded by bulk purchasing and the simplified fabrication of a restricted set of sizes are far more difficult to quantify with the same accuracy. …"*

1.1.6 Grouping methods

As it is hard to model the exact costs which obey the principle of commonality, but another possibility is to force the solution to a specific number of distinct profiles. This is called a grouping problem, which aims at grouping individual members and profiles to a specified number of groups. It consists of two strongly linked subproblems. The first subproblem is to select a distinct number of profiles to use, and the second subproblem is to define which groups of members share the same profile. These two problems are shown in Figure 11. The top figure indicates the possible profiles, of which the colours beams represent a selection of 4 out of 25 problems. In the bottom figure, the members are grouped with corresponding colours.

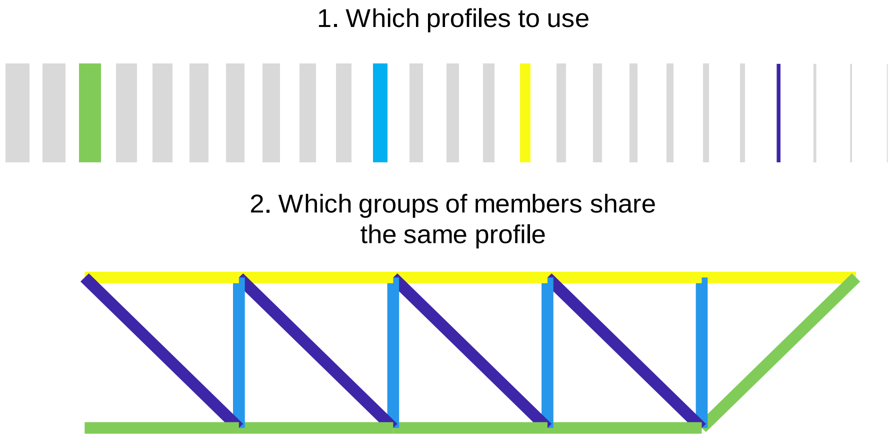


Figure 11 – Two subproblems of the grouping problem. These problems are strongly linked.

In design practice, these problems are treated separately and by hand: an engineer first uses his engineering experience to solve the second subproblem. After doing so, he selects the best profile for those groups. The ability to do this optimally is strongly influenced by the experience of the engineer, as restricting the number of options in solving the first subquestion might rule out the possibility to find the optimal result. Furthermore, the complexity of this problem is immense; the number of possible options can be very high.

Grouping methods are methods which can solve the grouping problem (Barbosa and Lemonge 2005; Walls and Elvin 2010a). Multiple methods have been proposed in literature, but as far as I am aware, no comparison is performed on all available methods. Furthermore, the computational effort of these methods should be part of this comparison, as some methods might be easy to apply but lead to a poor result, while other methods find a better result but require more computations.

Besides the ability to limit the number of distinct profiles, grouping methods have an additional benefit: the mathematical complexity of the optimisation problem can be simplified, reducing the necessary computational effort. That is because grouping methods may reduce the search space, which is beneficial or even necessary for search methods to converge to a global minimum (Templeman, 1988; Krishnamoorthy, Prasanna Venkatesh and Sudarshan, 2002; Toğan and Daloğlu, 2006; Mashayekhi *et al.*, 2012).

In most design problems, an elementary grouping is applied, which should not be confused with the grouping of the grouping problem. In elementary grouping, beams are grouped based on geometrical considerations, like symmetry, continuous beams, or architectural demands. These beams are no longer treated independently in the problem description. It should be noted that according to Stolpe (2010), a symmetric problem might have an asymmetric optimal result; symmetry should only be enforced if there is proof that the optimal design should be symmetric.

1.1.7 Implicit and explicit optimisation

But why to apply grouping methods if normal cost optimisation anyway leads to a grouped solution? The answer is in the complication of defining a complete cost function and the possibility to reduce the mathematical complexity. Therefore, a distinction can be made between implicit cost, explicit cost and weight optimisation. Implicit optimisation does not include a grouping method, but explicit cost and weight optimisation do.

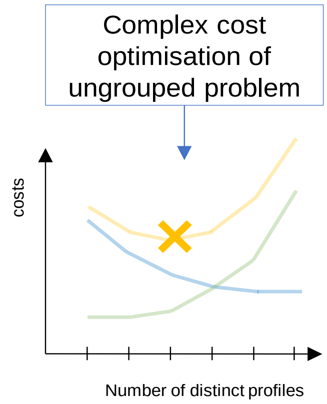


Figure 12 – Implicit cost optimisation. One optimum defines both costs and the number of distinct profiles.

Implicit cost optimisation

For implicit cost optimisation, no grouping method is applied, but as the cost function has a low value for solutions that have few distinct profiles the optimisation method finds the optimum number of groups and the corresponding design; the number of groups is not fixed but a result of the optimisation.

As no grouping method is used to simplify the mathematical complexity, the search space is big. Furthermore, the unknown cost function has to be defined, but will probably contain many local optima.

However, if these problems are solved, the global cost optimum can be identified in one analysis, represented by the single cross in Figure 12. However, this process is expected to be too complicated with the currently available optimisation methods, computational resources and cost functions. Furthermore, the questionable existence of the perfect cost function reduces the value of this solution.

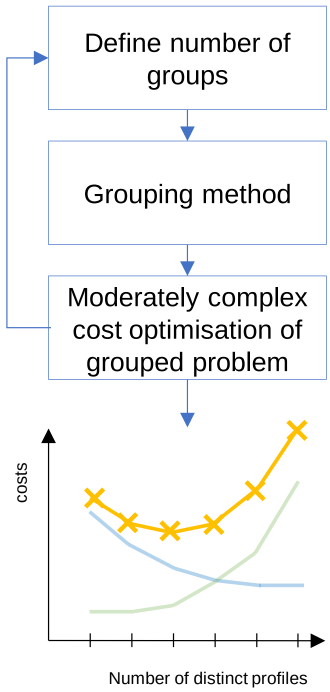


Figure 13 – Explicit cost optimisation. For each number of distinct profiles, a cost optimum is found.

Explicit cost optimisation

For explicit cost optimisation, a grouping method is applied. This grouping method reduces and simplifies the complex search space. This procedure is repeated for all possible number of groups, which are optimised for every number of groups separately. From the minimum value of all grouped optimisations, the global grouped optimum can be identified, as shown in Figure 13.

Although the problem is simplified by a grouping method, a cost function is needed anyhow. Again, this cost function has a complex search space and is hard to define.

Nevertheless, I expect the total computational effort to be reduced compared to implicit cost optimisation. The same global optimum as the implicit cost optimum can be found. As with implicit cost optimisation, the value of this solution is questionable.

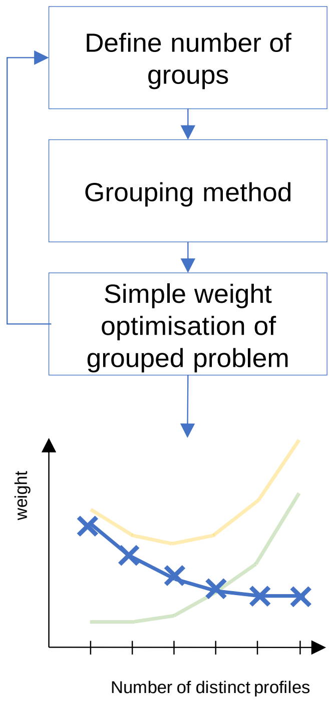


Figure 14 – Weight optimisation. For each number of distinct profiles, a weight optimum is found.

Weight optimisation

Finally, weight optimisation does not include the costs of the structure. Therefore, a grouping method is required to reduce the diversity of the solution. The optimisation would converge in the solution with high diversity otherwise. By repeatedly applying a grouping method for all possible number of groups, a descending optimum weight curve is found, as shown in Figure 14.

This optimisation problem is simple compared to implicit and explicit cost optimisation as it does not use a cost function and small subproblems are solved.

However, the final graph gives no information on which number of groups is desired. An engineer now has to make that choice. Nonetheless, the graph provides useful information to make a fair trade-off between weight and diversity.

Because of the simplification of the optimisation problem and the non-existence of the perfect cost function, I regard weight optimisation as most advantageous in the design of structures.

1.2 Research question

The research question of this thesis is defined as:

Which method for grouping can find the lightest and cheapest steel structure with minimal computational effort?

1.3 Subquestions

The main research question is solved by answering five subquestions:

1. Which methods exist for grouping in steel frame optimisation in literature?
2. How do the grouping methods perform theoretically on weight and cost optimisation?
3. How do the grouping methods perform on benchmark problems for minimum weight?
4. Do the currently available grouping methods perform well enough for application in research and practice? If not, can a new method be developed, outperforming current methods?
5. How do grouping methods improve the minimum cost design of a real-life case-study?

1.4 Scope

The scope of this research project is set to the following six boundary conditions:

- Steel truss and frame structures are considered with a fixed geometry. The analysis of a fixed geometry is representative; in many steel structures the geometry is fixed by demands like storey-height and fire safety.

- Members are chosen from sets of standard steel profiles. This assumption is close to reality in which most steel constructions are built with standard available profiles, which allows for economical bulk production. The use of custom members is possible as well, but these are only used for special applications in which normal beams cannot be applied.

- The structural analyses are evaluated on basic constraints which provide similar design constraints as the design codes. The expansion of constraint functions to these design codes is an elaborate task, more useful for industrial application than for science.

- The structural design is on a global level; the detailed design, including the design of connections, is not considered. This approach is similar with the process in practice, in which the detailed design is made after a first global design. Usually, some margin is taken on the unity check of the global design to allow flexibility in the detailed. As this margin is case-specific, it is not taken into account in this study.

- The structural analysis is a linear elastic analysis. This is chosen because nonlinear analyses are rarely applied in practice for the design of regular buildings, and are not required by most building codes.

- Grouping methods are compared on their performance in weight optimisation problems. Because of the complication and variability in defining a generally valid cost function, the performance of each individual grouping methods on cost models is not evaluated. However, the applicability to cost optimisation is demonstrated for the best method.

1.5 Thesis outline

This thesis answers the subquestions chapter by chapter. For sake of compactness, the new method, which was found in this study, is introduced together with existing methods. This allows comparing both the existing methods and the new method in one overview.

Chapter 2 introduces the grouping methods present in literature, answering the first subquestion. Next, Chapter 3 presents the new method. Chapter 4 compares all methods on their theoretical performance, answering the second subquestion. Based on this comparison, a selection of the most promising methods is presented. This selection is analysed numerically in Chapter 5, answering the third subquestion. Both Chapter 4 and 5 evaluate the need and requirements of a new method, and the proposed new method is matched with these requirements, as part of the fourth subquestion. Chapter 6 treats the last subquestion on cost optimisation of a real-life case. Subsequently, Chapter 7 discusses various limitations of the research. Conclusively, Chapter 8 answers the main research question and Chapter 9 provides recommendations for further research and applications. In the Appendices, examples and detailed descriptions of elements of the research project are given.

2 Overview grouping methods

Grouping methods are essential for effective optimisation of steel structures. These methods reduce the number of distinct profiles in the final solution of an optimisation problem. This chapter shows the grouping methods available in literature, thereby answering the first subquestion:

1. Which methods exist for grouping in steel frame optimisation in literature?

The methods for grouping present in literature are listed in Figure 15 and are explained in detail in this chapter. The grouping methods are arranged by the moment at which the groups are formed: before, during or after the optimisation process. This chapter describes each method in detail with an example.

A priori based on the geometry

- Rules of thumb (ROT)
- Neural network (NN)
- Member length (ML)
- Profile selection (PS)

A priori based on an additional analysis

- Axial force (AF)
- Axial force and slenderness (AF+S)
- Axial force and weakening (AF+W)

During optimisation process

- Cardinality constraints (CC)
- Additional constraints (AC)
- Multi-objective (MO)

After ungrouped optimisation

- Rounding to selection (RS)
- Ungrouped combinatorial search (UCS)

Figure 15 - Overview grouping methods which categorisation on moment at which the groups are formed.

A reference case for all methods is the case in which no grouping is applied (NG). In this case the optimisation method finds the profiles for individual members, and all available profiles are available to do so; no profiles are selected, and no groups are assigned to the members.

An example of no grouping is shown in Figure 16. It shows the 18-bar cantilever truss benchmark problem, of which its properties are attached in Appendix F. The colours in this figure represent different groups.

Appendix C.1 shows how optimisation for an ungrouped problem converges to the optimal design.

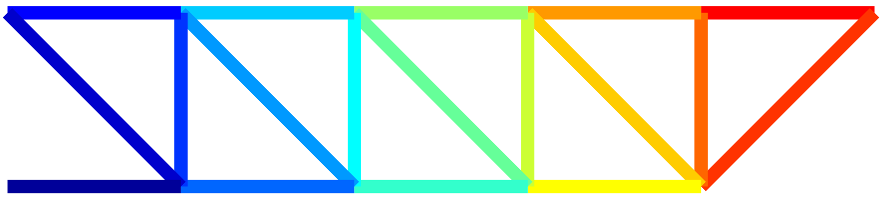


Figure 16 – Example NG with for each member the choice for all profiles.

2.1 A priori based on geometry

This category incorporates all methods which involve grouping before the start of the optimisation problem without relying on an additional analysis; in these methods, the grouping is solely based on the geometry of the structure. An exception is the profile selection method. Nonetheless, this method is regarded as part of this category because it is applied before the optimisation starts, and it does not use a structural analysis.

2.1.1 Rules of thumb

A popular choice is to group members manually, based on experience, personal preferences and rules of thumb. By doing so, an engineer (implicitly) applies the rules of thumb method (ROT). Examples of rules of thumb for storey buildings are that interior beams of similar span and diagonals are each grouped per storey, and exterior columns are grouped over two adjacent storeys. For truss beams, top and bottom chords are grouped in the middle and outer bays, as well as verticals near the supports. These rules of thumb can be applied in regular structures like storey buildings and truss beams. However, for more complex structures, rules of thumb might not be applicable.

In most steel frame optimisation problems in literature, this method has been used (Arora and Govil 1977; Templeman 1988; Chan 1992; Biedermann and Grierson 1996; Walls and Elvin 2010a, b). As it has been the most used method in practice as well, engineers are experienced with this procedure. However, for complex structures and multiple loading conditions this procedure can be complicated and requires professional engineering expertise. Therefore, this method does not guarantee an optimal result.

An example of the rules of thumb method is shown in Figure 17. According to the rules of thumb, this method groups members for top and bottom chords, diagonals, and verticals.

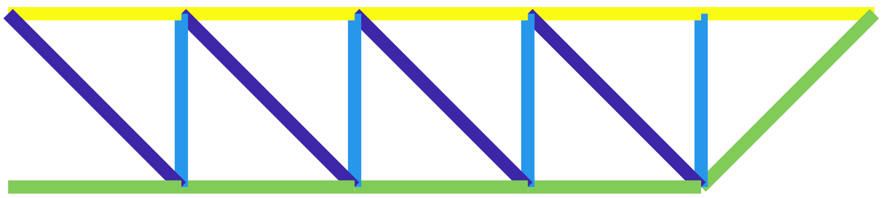


Figure 17 – Example ROT, grouping of diagonals, vertical, upper and lower truss.

2.1.2 Neural network

The neural network method (NN), prescribed by Biedermann and Grierson (1996), aims at using artificial intelligence for the grouping process. Neural networks can model knowledge which is difficult to represent algorithmically. By doing so, the rules of thumb are not explicitly specified, but these follow from the neural network. The neural network is trained by the result of grouping using existing designs. Furthermore, an engineer can set up this neural network in many ways by defining the input variables and neural network lay-out. For example, the engineer can choose from multiple input variables: the relative location of members, irregularities, the number of members, and many more.

A limitation of this procedure is that a neural network is only able to mimic the input knowledge on a new model. To do that properly, the new model should be similar to the training model. This desired similarity cannot guarantee an optimal result for the wide range of types of steel structures. Furthermore, this method heavily depends on the quality of the input knowledge; if poor grouped design are used for training the neural network, new design are poor as well. Finally, high expertise in neural network design is required to design and train these neural networks.

For the example problem, the result of the neural network method depends on the input data on which it has been trained. If the input data would be the grouping of Figure 17, the result of this method has the same grouping.

2.1.3 Member length

Biedermann and Grierson (1995) proposed another method in which members with similar member lengths (ML) are grouped. As the span of members is only one of many variables influencing the grouping procedure, this approach cannot guarantee an optimal result.

An example of the member length method is shown in Figure 18. As the diagonals are different in length than the other elements, these are grouped separately. Grouping to more than two groups is not possible in this case.

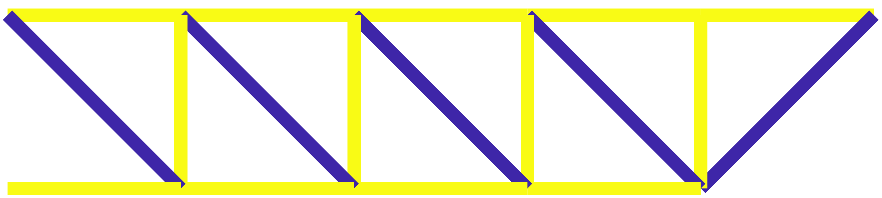


Figure 18 - Example ML, grouping of members with similar length.

2.1.4 Profile Selection

The profile selection method (PS) was proposed by Templeman (1988), and it is fundamentally different from other methods: instead of a selection of members to be grouped, an engineer selects a reduced set of profiles. The number of selected profiles is equal to the number of desired groups. The limitation of profiles reduces the search space and leaves the optimisation algorithm the task of finding the optimal steel profile of this reduced set for each design variable.

However, no guidelines exist on selecting the optimum set of profiles. An optimal result is thus not guaranteed.

For the example of the 18-bar structure, this method could select four possible profiles as shown in Figure 19. Now, the optimisation process must determine for each member which of these four profiles is optimal. The result has maximum four groups.

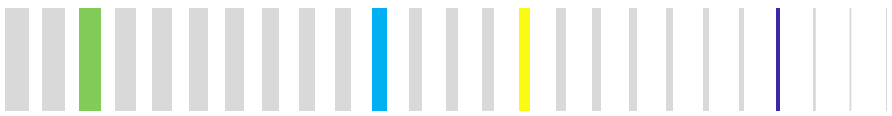


Figure 19 – Example PS, selection of four possible profiles.

2.2 A priori based on an additional analysis

This category consists of all methods which involve grouping before the start of the optimisation problem by relying on an additional analysis.

2.2.1 Axial force

Many authors have used the axial force method (AF) for truss structures (Krishnamoorthy et al. 2002; Toğan and Daloğlu 2006, 2008; Yang et al. 2016). In this method, members are grouped based on the axial force from one preliminary analysis. This preliminary structural analysis is performed with an initial design in which all members have the same cross-section. As long as all members have the same cross-section, the choice of this profile does not influence the force distribution. Based on this analysis, each member is placed in a group according to their axial force. For problems with multiple load cases, a choice must be made which internal forces are used. The maximum absolute force was used as the maximum value in this study. A suggestion for repeating this procedure after several iterations in the optimisation process was suggested by Krishnamoorthy et al. (2002). However, the paper stated that only very few members jump groups during this process, so this suggestion was not applied in this study.

Once the axial force range is known, an automatic division can be performed by one of four procedures. This division finds a bandwidth of axial range for each group. These procedures are referred to as axial force method 1, 2, 3 and 4. A manual division is possible as well, but as this requires additional input, it was regarded as not suitable for optimisation in this study. The four procedures are as follows:

1. The full range of internal forces is divided into equally spaced intervals.

2. Both the compressive and tensile range of internal forces are divided into equally spaced intervals. The number of groups for the compressive and tensile range is linearly related to the absolute size of these two intervals.

3. Both the compressive and tensile range of internal forces are divided into equally spaced intervals, and a separate group is created for an interval around zero internal force. The engineer must choose the size of this last interval. I took it as 10% of the total range in this study. The number of groups for the compressive and tensile range is dependent on the relative absolute size of these intervals. This procedure can be useful for structures in which some members are only required for stability reasons and do not carry much load.

4. The range of absolute values of the element internal forces is divided into equally spaced intervals. This method allows for a grouping of both members in tension and compression in one group, as opposed to axial force method 1, 2 and 3. However, the absolute values of the axial force cannot represent the feasibility of both compressive and tensile members; members in compression show buckling behaviour, so the absolutes values cannot guarantee an optimal result.

Figure 20 shows the results of these procedures for the example problem. In each subfigure the internal axial forces for all members is shown. The red lines represent the division limits. The colours of the bars represent the grouping. The resulting geometric groups is shown as well.

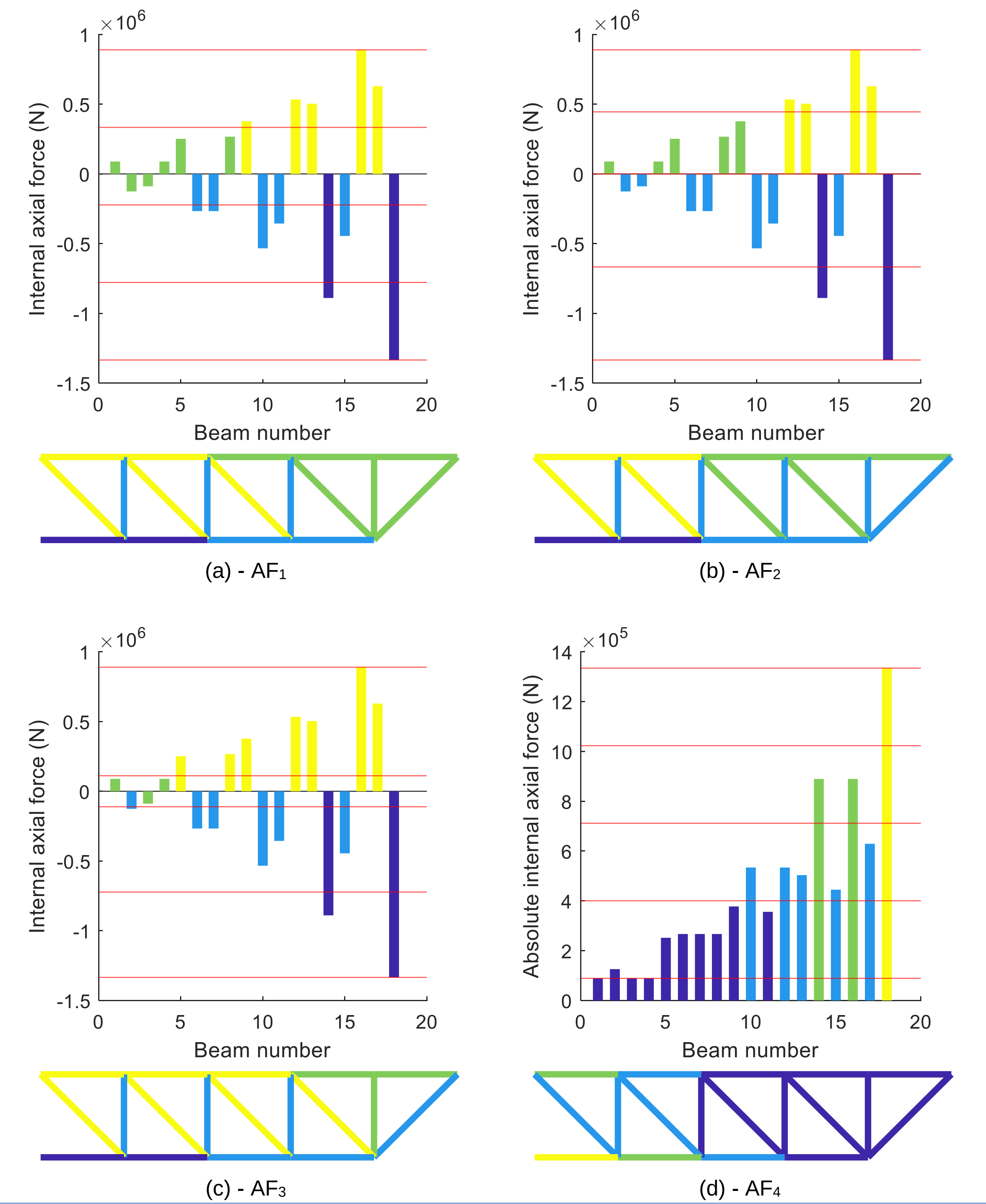


Figure 20 - Comparison axial force distribution and grouping procedures AF.

2.2.2 Axial force and slenderness

The axial force and slenderness method (AF+S) is an adaption of the axial force method (Toğan and Daloğlu 2008; Yang et al. 2016). The difference is that for a compressive member, the slenderness ratio is used to determine member groups. As the slenderness is dependent on the design, its value is estimated before the optimisation starts. Therefore, this method takes the slenderness from the lightest possible profile which satisfies the buckling stress criterion with a given initial design for the other members. This estimated slenderness is dependent on this initial design for statically indeterminate structures because the force distribution is changed in case another initial design is chosen.

The grouping is performed by one of two procedures 1 and 2, similar to the methods axial force method 2 and 3. The first procedure divides both the range of axial force and radius of gyration into separate groups. In the second procedure, a separate group is created for zero force members.

Figure 21 shows the results of these procedures for the 18-bar truss problem. For members in tension and low internal force, the left vertical axis shows the internal axial force in blue. For members in compression which are not part of the low internal force group, the right vertical axis shows the radius of gyration. The horizontal lines show the limits of division.

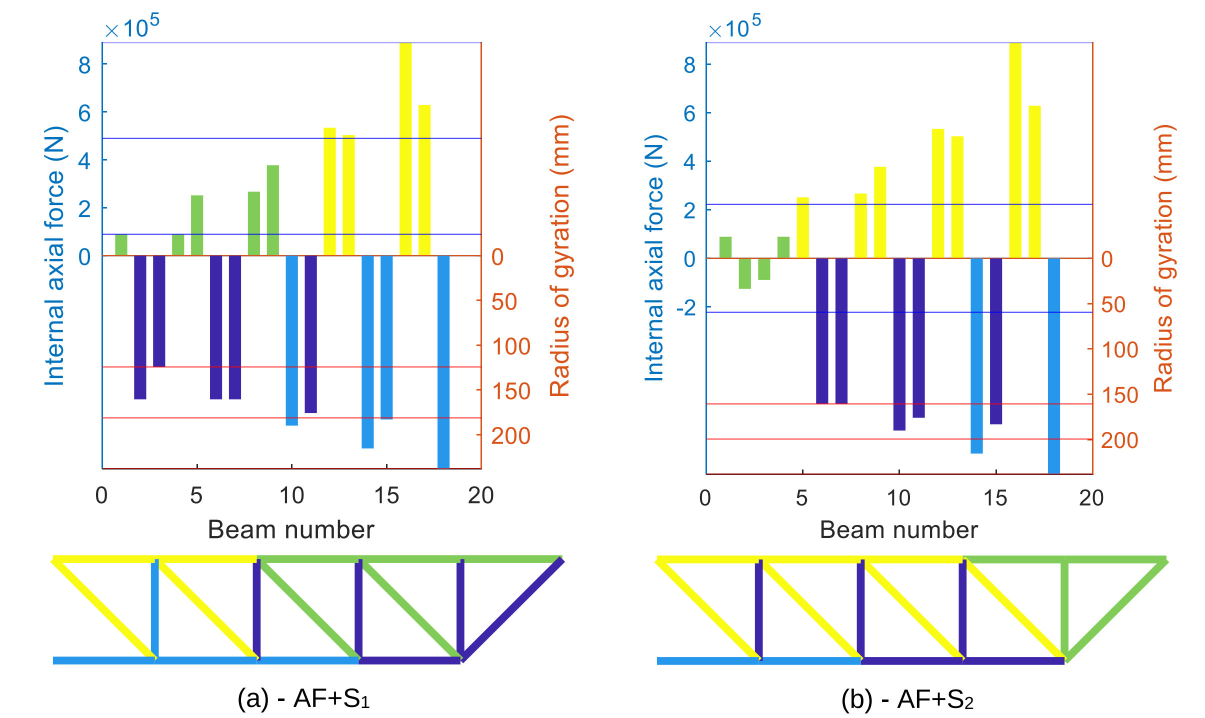


Figure 21 - Comparison axial force and radius of gyration distribution, and grouping procedures AF+S.

2.2.3 Axial force and weakening

Mashayekhi et al. (2016) used the axial force method, but propose to assign a slightly weaker profile after optimisation to the members which are loaded the least in a member group, thereby weakening the design (AF+W). The method assumes that these profiles are overdesigned in their initial group and the resulting design is still feasible after weakening. Thereby two specific percentages of the members in each group are treated separately: a small percentage of the least loaded beams is given a profile one steps weaker than the optimised solution, and another percentage is assigned a profile two steps weaker. The percentages and definition of weakness have to be defined by an engineer.

This method is not applicable to the example problem because the desired number of groups is too low to add two groups for both the compressive and tensile range. Therefore, the example shown in Figure 22 adds only one subgroup per group. For both the nine compressive and nine tensile members separately, the two members which are loaded the least are grouped separately.

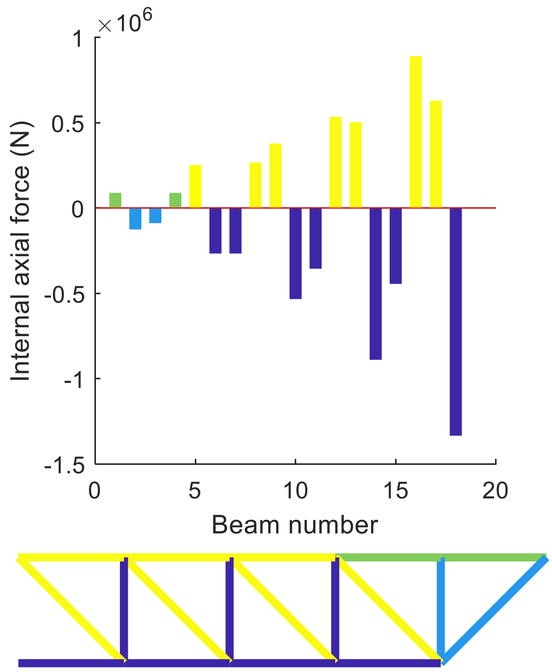


Figure 22 - Axial force distribution AF+W and weakening of two least-loaded members in each group.

2.3 During the optimisation process

The methods of this category vary both the optimal steel profiles as the distribution of groups during the optimisation process, as opposed to all former methods which vary only one of both.

2.3.1 Cardinality constraints

The method of cardinality constraints (CC) limits the search space to solution in which the number of groups is equal or smaller than the desired number of groups. To do so, the encoding of the problem is altered. This concept was claimed to be introduced by Barbosa and Lemonge (2008), but shows close resemblance with the research performed by Reitman and Hall (1990).

The original encoding points directly each member to a profile type. The encoding of the problem is changed by introducing two parts, a type and pointer part. The type part selects the profiles to use for the number of desired groups, solving the first subproblem of the grouping problem introduced in Chapter 1.1.6 and shown in Figure 23. The pointer part assigns for every independent beam one of the groups of the type part, solving the second subproblem shown in Figure 23.

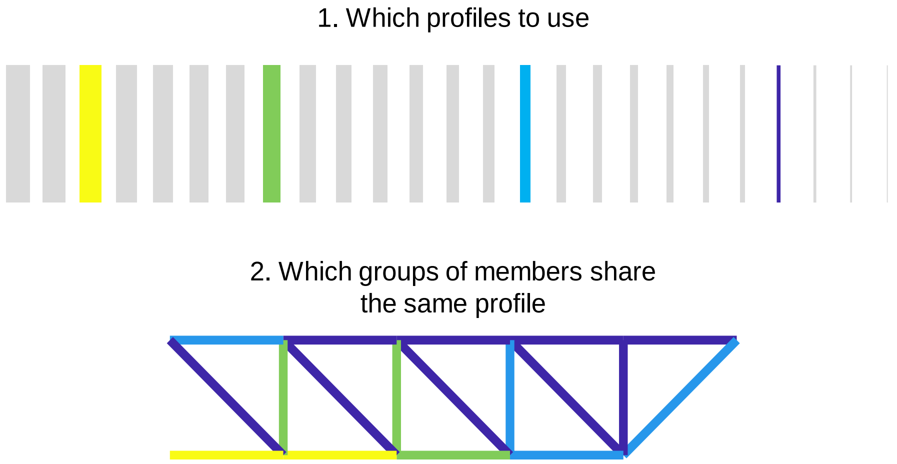


Figure 23 – Two subproblems of the grouping problem

To illustrate both mappings, Figure 24a shows a standard encoding and Figure 24b shows the adapted encoding for the 18-bar example problem. The solution of both encodings is shown in Figure 24c. The first beam is represented by the first cell of the normal encoding and the first pointer cell of the CC encoding. In normal encoding it directly maps to the 6th (thin) profile, while in CC the pointer maps to the 1st (dark blue) group of the type part, which has the 6th (thin) profile. Similarly, the last beam is represented by the last yellow cell of the type part in the CC encoding. This value maps to the 4th (yellow) group of the type part, which is the 22nd (thick) profile, while the normal encoding points directly to the 22nd profile. In essence, the pointer part of the CC encoding varies the indices to groups or colours, while the type part varies the profile indices or thicknesses.

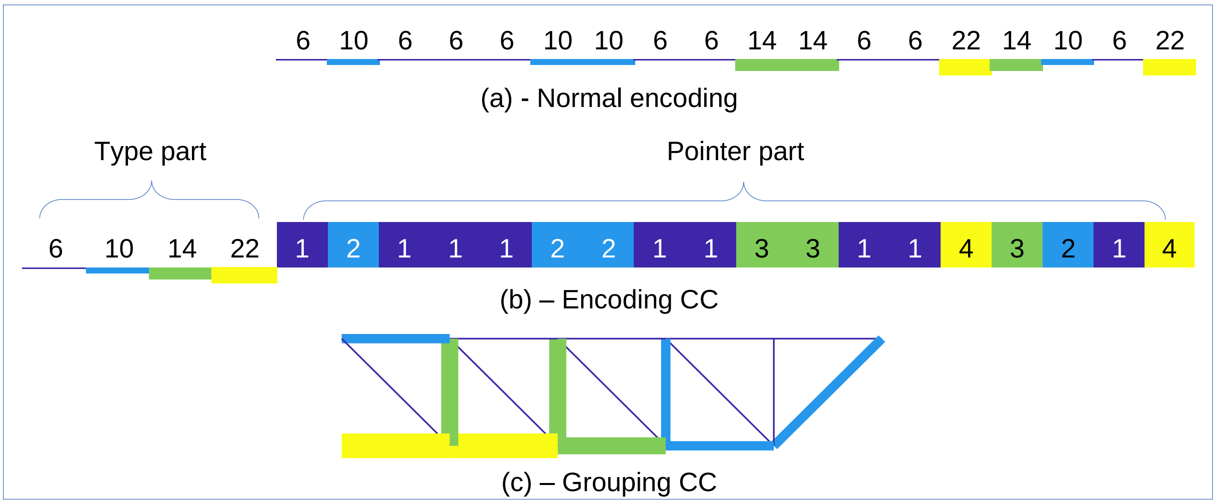


Figure 24 – Example normal encoding and encoding CC for same grouping.

Although this method adapts the encoding so that the number of groups is maximum to the desired number of groups, this method can give a solution with less number of groups. This can happen in two ways. The first possibility is if two of the indices in the type part are equal, as shown in Figure 25a. The second possibility is that one of the group indices in the pointer part is missing, as it is in Figure 25b.

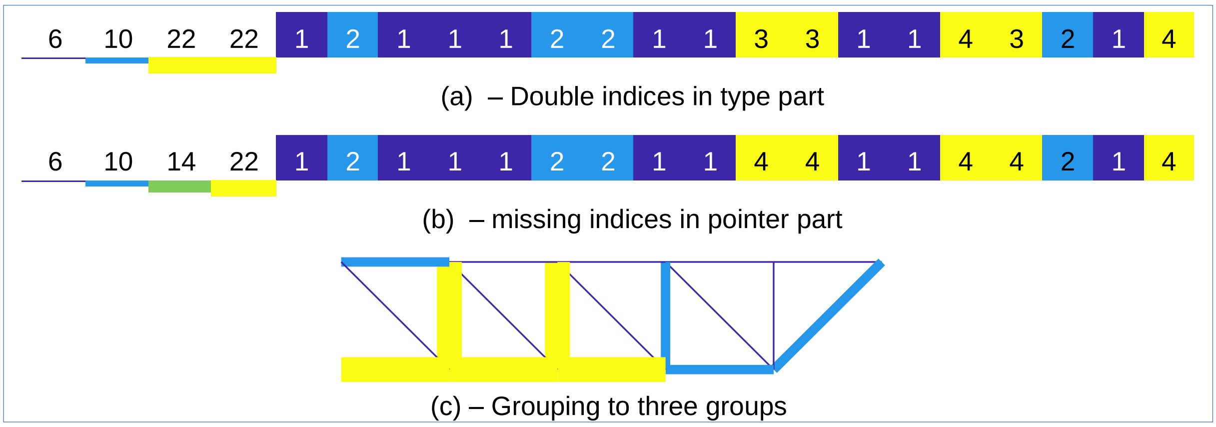


Figure 25 – Encoding CC to less than the desired number of groups.

Appendix C.2 shows how optimisation with the cardinality constraints method converges to the optimal design method for the 18-bar cantilever truss.

CC has been applied often in literature by the research group of Barbosa and Lemonge (Barbosa and Lemonge, 2005; Barbosa, Lemonge and Borges, 2008; Liu et al., 2012), but I do not know applications of this method in practice.

2.3.2 Additional constraints

Another option to limit the number of different profile sections during optimisation is by setting additional constraints (AC) in the standard design optimisation model. Barbosa et al. (2008) proposed this in the form of the following inequality constraint $g(x_i)$:

$$g\left(x_i\right) = \frac{m\left(x_i\right)}{k} - 1 \leq 0 \tag{2.1}$$

In which m is the number of distinct steel profiles in a design, which is a function of the design x_i, and k is the desired number of profiles. This constraint is added to the standard design optimisation model shown in Chapter 1.1.3. As it introduces a maximum on the number of groups, a solution with less than the desired number of groups is feasible as well. An equality condition can also be applied, as was done by Kanno (2016) for truss topology optimisation. However, optimisation methods struggle with equality constraints (Langelaar and van Keulen 2019), so only inequality constraints were applied in this study.

In application of this method in literature, the final solution did not satisfy this constraint (Barbosa et al. 2008). However, this might be caused by the method of constraint handling.

In this study, a constraint handling technique was used in which a feasible solution can be found more easily. Nonetheless, the optimisation method struggled with finding the optimum because of the large search space and small feasible part of it. In the preliminary phase of this research project, experiments showed that the optimisation method converges to design with a high weight. Furthermore, many different solutions were found when repeating the analysis.

2.3.3 Multi-objective

One of the first published methods for grouping members changes the type of optimisation: from a minimum weight problem to a multi-objective problem (MO) for both minimum weight and minimum number of distinct profiles (Galante and Oñate 1996). Shea et al. (1997) used this method in combined size and shape optimisation of truss structures. For application of a multi-objective optimisation problem, a weighted objective function *f* is proposed (Arora 2017):

$$f(x_i) = \lambda_1 \frac{\sum_{i=1}^{n} A_i(x_i) L_i \rho_i - W_{NG}}{W_{\max} - W_{NG}} + \lambda_2 \frac{m(x_i) - 1}{N_{vars} - 1} \tag{2.2}$$

In which λ_1 and λ_2 are weight parameters which are adjusted to control the number of groups in the final design indirectly. x_i are the design variables, which enter the objective function in the weight of the design $\sum_{i=1}^{n} A_i L_i \rho_i$ and the number of profiles in the design *m*. The weight and number of groups are normalised to have values between 0 and 1. Therefore, W_{NG} is the minimum weight of the ungrouped result, and W_{max} is the weight of the structure if the heaviest profile is chosen for all profiles. The minimum number of profiles is 1. N_{vars} is the maximum number of profiles, which is equal to the number of design variables. This objective function replaces the original objective function shown in Chapter 1.1.3.

By using the weighted objective function, the number of member groups is indirectly altered in this method and cannot be specified a priori. To find a specific number of groups, the optimisation must run multiple times until the desired number of groups is found. Instead of using a weighted objective function as equation (2.2), other methods are available as well. These other methods were not treated in this study.

Multi-objective problems do not have a unique global solution. Instead, a set of solutions is found, of which each solution is Pareto optimal. Pareto optimality is defined for the solution set for which no other design exist which satisfies all the objectives better. The Pareto optimal set is typically plotted in the criterion space, in which the axes represent objective functions (Arora 2017). In theory, the Pareto set shows for all number of groups the grouped optimum. However, the grouped optima are hard to find in practice.

As an example, this method was applied in a preliminary phase of this study for the 18-bar cantilever truss. In 40 optimisation analyses, the multi-objective problem was solved. In these analyses, the weight factors were varied and 40 different solutions were found. Figure 26 shows the criterion space of these results. The horizontal axis shows the number of groups in the final solution and the vertical axis the corresponding weight. The colouring of the dots shows the corresponding weight factors. It should be noted that the weight parameters do not guarantee a specific number of groups in the final solution. The blue line represents the approximation of the Pareto optimal set.

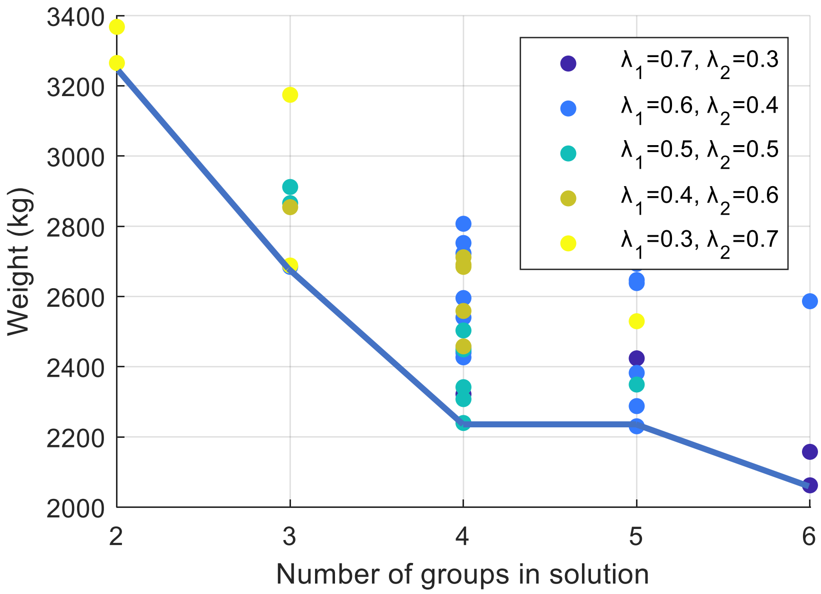


Figure 26 - Example result MO in criterion space for different weight values.

From these 40 solutions, 20 solution contained four groups. For these solutions, the method gave diverse results in both the distribution of groups and weights of the solutions as well. Comparison of these results with results from other methods showed that the global optimum for four groups was not part of these solutions. The 20 results which contain four groups in their final solution are shown in Figure 27. In this figure, all 20 solutions for each beam are plotted as part of their length. The thickness of a beam segment represents the unit weight of the profile. The colour of the beam segment represents the group.

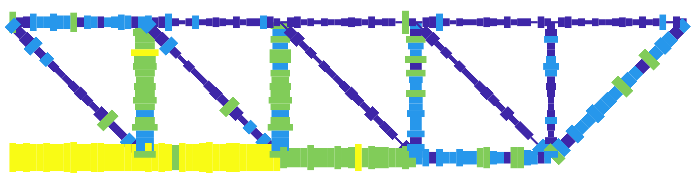


Figure 27 – Example MO, 20 diverse solutions plotted as part of each beam's length.

As a result of altering the objective function, this method converts the explicit optimisation problem into an implicit optimisation problem. As stated in Chapter 1.1.7 and demonstrated by the example, this leads to a highly complex optimisation problem. Furthermore, the best result per group might not be the grouped optimum.

2.4 After ungrouped optimisation

The methods of this category group members based on the ungrouped optimal result. This need comes at a cost, as finding the ungrouped optimum requires an additional optimisation analysis.

2.4.1 Rounding to selection

This method performs grouping according to their cross-sectional area or weight per unit length of the ungrouped optimal result, by rounding this result to a selection of profiles (RS). However, no guidelines are available on the division of the groups and whether profiles are rounded up or down. Templeman (1988) proposed a procedure in which on the result of continuous ungrouped optimisation a local search is performed. This local search should evaluate local rounding options to find the optimal rounding. Still, the definition of local and rounding is not generalised. Provatidis and Venetsanos (2006) grouped members based on their mean value and standard deviation of their unit weight per length. This procedure requires a user-defined tolerance. Adeli and Sarma (2006) proposed a similar procedure using a fuzzy discrete multi-criteria cost optimisation model.

For all implementation of this method, the result of this is strongly dependent on the user input and the optimal grouped structure might not be similar to the ungrouped optimum design. Consequently, rounding of the ungrouped design does not guarantee to find the grouped optimum.

An example of the solution of the cross-section area for the ungrouped problem is shown in Figure 28. No grouping is shown for this example as no decisive guidelines are available to do so.

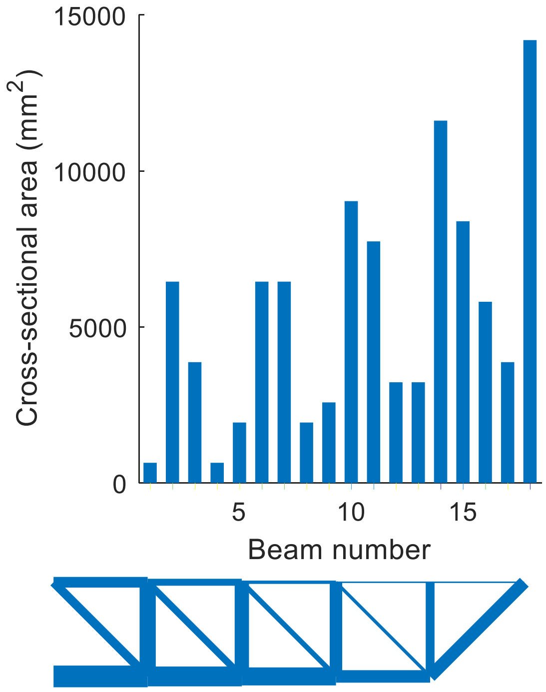


Figure 28 - Solution cross-sectional area ungrouped solution, no grouping is shown as no decisive guidelines are available to do so.

2.4.2 Ungrouped combinatorial search

The final method which has been proposed in literature is the ungrouped combinatorial search (UCS) method (Walls and Elvin 2010a). This method uses a combinatorial search for grouping of the ungrouped result of the minimum weight problem. The combinatorial search evaluates all possible grouping obeying certain assumptions. Furthermore, the design from the combinatorial search is in general overdesigned and forces may be redistributed due to the grouping. Therefore, the structure is optimised once more in an optimisation in which the members are grouped.

Combinatorial search

The combinatorial search groups members according to their weight per unit length, and finds the optimum weight limits for each group. This technique is taken from the optimum standardisation problem, also known as catalogue optimisation or location-allocation problems (Reitman 1989; Reitman and Brent Hall 1990).

The combinatorial search evaluates the weight of a restricted set of combinations based on the weight per unit length of the members of the ungrouped optimum result. This restricted set included all possible combinations with as many limit profiles as the desired number of groups. In these combinations the profiles are replaced by the nearest heavier limit profile. Subsequently, the grouping of the combination with the lowest objective function is regarded as optimal. Following this procedure, the number of combinations to be evaluated for the combinatorial search N_{CS} is:

$$N_{CS} = \binom{m-1}{k-1} = \frac{(m-1)(m-2)\cdots(m-k+1)}{(k-1)(k-2)\cdots 1} \tag{2.3}$$

In which m is the number of profiles in the design and k the number of groups. This formula originates from the binomial coefficient $\binom{m}{k}$. The binomial coefficient gives the number of possibilities for selecting m items from a set of k items without repetition. In the combinatorial search, profiles are only allowed to be made heavier. Therefore, the heaviest profile represents the upper limit for first group, reducing both k and m by one. Subsequently, k - *1* profiles must be selected out of m - *1* profiles, representing the upper limit for the remaining m - k profiles. The algorithm for defining the resulting grouping is shown in Appendix A. The real number of possible combinations, without the restriction of increase the weight to the nearest limit profile, is more: $N_{exact} = \binom{m}{k}\left\{ {n \atop k} \right\}$, which is explained in detail in Chapter 4.1.

In the case of statically determinate structure without displacement constraints, the upper limits are the optimal values for all the beams in that group. That is because the forces are not redistributed, and the applied constraints are always met with a heavier beam. In that case, no grouped optimisation in the second step is needed.

Example bridge problem

As the standardisation problems were studied extensively in the former USSR, an illustrative example of the combinatorial search is the construction of bridges in the former USSR. Imagine seven bridges to be built. These bridges have to be of minimum length: 10, 30, 40, 55, 100, 120 and 150m. The USSR-governments would like to build all necessary bridges as cheap as possible. However, the government has resources to design and build only four distinct bridges. The combinatorial search solves this problem. It groups the 7 bridges to 4 distinct bridges with as little extra bridge length as possible. In this case, bridges are allowed to be lengthened, but shortening is not permitted.

The combinatorial search reduces the number of possible grouping reduces from $N_{exact} = \binom{7}{4}\left\{ {7 \atop 4} \right\} = 12250$ to $N_{CS} = \binom{7-1}{4-1} = 20$, when obeying the principle of 'a bigger bridge satisfies'. Table 1 shows these 20 combinations. The last row shows the added bridge length for each combination. The biggest length in each group can be used for all bridges in that group. Combination 5 with bridge lengths of 150, 120, 55 and 40 proves to be optimal in this case, as only 50 m of unnecessary bridge length has to be constructed.

Table 1 – Resulting bridge length of all combinations in the combinatorial search for the USSR-bridges example problem. The bottom row shows the total added length per combination, and the shaded column shows the optimum grouping.

<table>
<tr><th>Comb. / Length</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th>9</th><th>10</th><th>11</th><th>12</th><th>13</th><th>14</th><th>15</th><th>16</th><th>17</th><th>18</th><th>19</th><th>20</th></tr>
<tr><td>**150**</td><td>150</td><td>150</td><td>150</td><td>150</td><td>150</td><td>150</td><td>150</td><td>150</td><td>150</td><td>150</td><td rowspan="2">150</td><td rowspan="2">150</td><td rowspan="2">150</td><td rowspan="2">150</td><td rowspan="2">150</td><td rowspan="2">150</td><td rowspan="3">150</td><td rowspan="3">150</td><td rowspan="3">150</td><td rowspan="4">150</td></tr>
<tr><td>**120**</td><td>120</td><td>120</td><td>120</td><td>120</td><td rowspan="2">120</td><td rowspan="2">120</td><td rowspan="2">120</td><td rowspan="3">120</td><td rowspan="3">120</td><td rowspan="4">120</td></tr>
<tr><td>**100**</td><td>100</td><td rowspan="2">100</td><td rowspan="3">100</td><td rowspan="4">100</td><td>100</td><td>100</td><td>100</td><td rowspan="2">100</td><td rowspan="2">100</td><td rowspan="3">100</td></tr>
<tr><td>**55**</td><td rowspan="4">55</td><td>55</td><td rowspan="2">55</td><td rowspan="3">55</td><td>55</td><td rowspan="2">55</td><td rowspan="3">55</td><td>55</td><td>55</td><td rowspan="2">55</td></tr>
<tr><td>**40**</td><td rowspan="3">40</td><td rowspan="3">40</td><td>40</td><td rowspan="2">40</td><td rowspan="3">40</td><td>40</td><td rowspan="2">40</td><td>40</td><td rowspan="2">40</td><td>40</td></tr>
<tr><td>**30**</td><td rowspan="2">30</td><td rowspan="2">30</td><td rowspan="2">30</td><td>30</td><td rowspan="2">30</td><td rowspan="2">30</td><td>30</td><td rowspan="2">30</td><td>30</td><td>30</td></tr>
<tr><td>**10**</td><td>10</td><td>10</td><td>10</td><td>10</td><td>10</td><td>10</td><td>10</td><td>10</td><td>10</td><td>10</td></tr>
<tr><td>**Added length**</td><td>85</td><td>85</td><td>125</td><td>175</td><td>50</td><td>55</td><td>60</td><td>105</td><td>95</td><td>165</td><td>70</td><td>65</td><td>70</td><td>95</td><td>85</td><td>135</td><td>100</td><td>90</td><td>95</td><td>175</td></tr>
</table>

Assumptions

In the USSR-bridge problem, the length of one bridge does not influence the length required of another bridge. However, in structural design problems, the force distribution is changed by a modification in the design of statically indeterminate structures. Therefore, if one beam is made bigger, another one could be made smaller than initially assumed. This characteristic makes the problem of grouping in structures a unique standardisation problem. Nonetheless, the combinatorial search of the standardisation problem is used as an approximation.

To do this, instead of the assumption ‘a bigger bridge satisfies’, three other assumptions are made:

1. Members with a similar mass per unit length have comparable properties

2. A heavier member than the ungrouped solution will always satisfy the design constraints

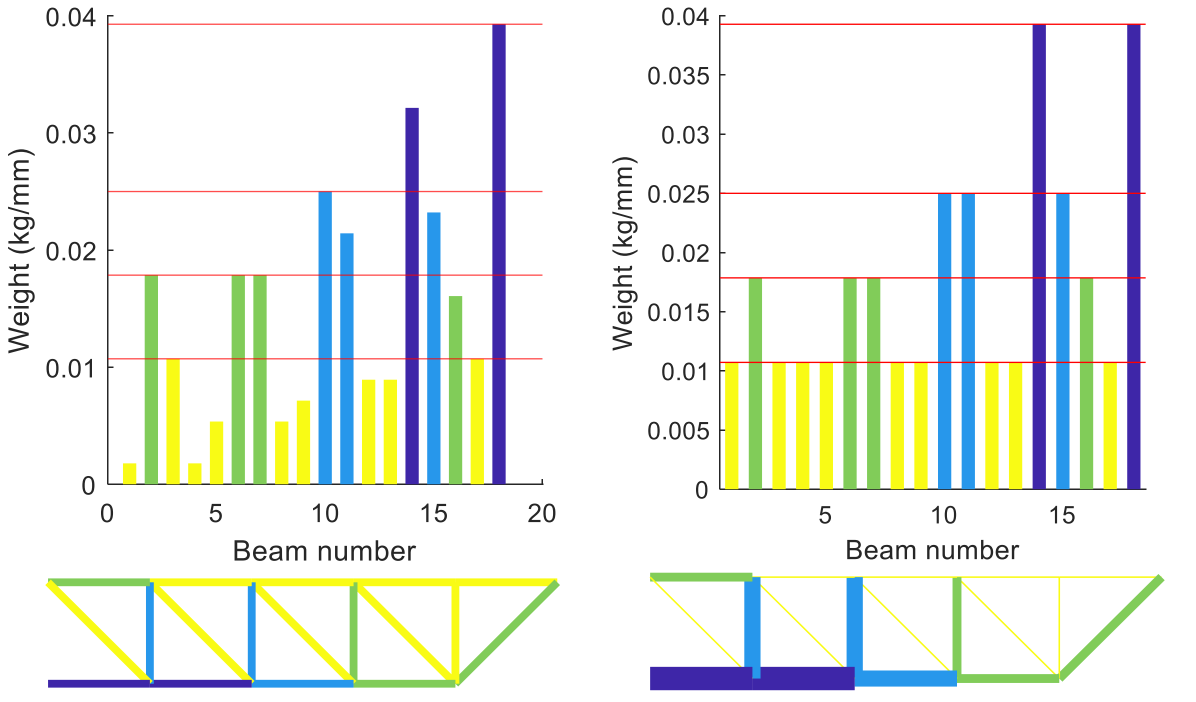
3. A lighter member than the ungrouped solution will never satisfy the design constraints

These assumptions are only generally valid for statically determinate structures without displacement constraints, which prevents this method of finding the true global grouped optimum in other structures. In other structures, a force redistribution might disprove the second and third assumption. If a profile database is used consisting of multiple different kind of profiles, like both I-shaped and angle-shaped profiles, the first assumption might not be true.

In practice, it could be the case that multiple parts of structures are grouped separately in a subset; for example, the grouping of columns and beams to other profile types. In that case, each of the subsets is grouped separately with the combinatorial search and it assumed the optimum is the sum of the subset’s optima. This is not necessarily true, as each alteration of the design of one of the subsets changes the force distribution in the total structure.

Example steel structure

A graphical representation provides further insight into the procedure of the combinatorial search. Figure 29a shows the results of the combinatorial search on the ungrouped 18-bar truss structure. The mass per unit length of the ungrouped solution is plotted for all beams. The combinatorial search finds a grouping of this ungrouped solution, shown by the height of the red lines, for which the added total mass is minimal. The added mass is equal to the white area, between the ungrouped solution and the red limits, multiplied with the corresponding member length. The resulting groups are shown with coloured bars.



(a) – General case

(b) – Statically determinate structures without displacement constraints

Figure 29 - Example weight distribution and grouping ungrouped result UCS

The 18-bar cantilever truss is a statically determinate structure without displacement constraint. In this case, the force distribution does not change by altering the profiles, and assumption two and three are guaranteed to be true. Therefore, the upper limits represent the optimal weight of all beams in that group. Therefore, the result can directly be obtained, and no grouped optimisation is needed. The result is shown in Figure 29b.

As a further explanation, an animated example of the combinatorial search is shown in Appendix C.4.

2.5 Conclusion

This chapter has presented the twelve grouping methods proposed in literature, thereby answering the first subquestion:

---

1. Which methods exist for grouping in steel frame optimisation?

The grouping methods which have been proposed in literature are:

1. Rules of thumb (ROT), in which an engineer manually groups members.
2. Neural network (NN), in which the engineer’s rules of thumb are automatised in a neural network.
3. Member length (ML), which groups members based on their length.
4. Profile selection (PS), in which an engineer selects a number of profiles, equal to the desired number of groups.
5. Axial force (AF), which groups members based on their axial force.
6. Axial force and slenderness (AF+S), which is similar to axial force, but compressive members are grouped on slenderness
7. Axial force and weakening (AF+W), which is similar as axial force, but it treats the least-loaded members of a group as a separate group.
8. Cardinality constraints (CC), which reduces the search space to all possible grouping in which the number of groups is equal or less than desired.
9. Additional constraints (AC), which adds an inequality constraint on the number of groups in the constraint function.
10. Multi-objective (MO), which adds the number of groups in the objective function.
11. Rounding to selection (RS), which rounds the result of an ungrouped optimisation to a result with the desired number of groups.
12. Ungrouped combinatorial search (UCS), which find the lightest combination of the ungrouped solution, in which members can only be combined with a member of equal or heavier weight.

These methods differ in the amount of user input needed and generality to different problems.

User input

User input is needed from an engineer when choosing group layout, number of groups, group bandwidth, initial design, reduction of profile library and on additional calculations. The group layout is the choice of which members should share the same profile. The group bandwidth includes the choice for one of the procedures proposed for AF and AF+S, or a manual selection of the normal force bandwidth per group. Table 2 shows the input required in the grouping process for all methods.

The methods which include grouping before the optimisation process starts, require more input. For the neural networks and rounding to selection method, the additional calculations demand high expertise. The additional constraints and multi-objective methods require adaptation of the optimisation problem. This requires knowledge and choices on how these adaptations can be made, for which many options are possible.

Table 2 - Comparison user input

Methods		**Group Layout**	**Number of groups**	**Reduction of profile library**	**Group bandwidth**	**Initial design**	**Additional calculations**
A priori based on geometry	**ROT**	Yes	Yes	No	No	No	
	NN	Yes, in training phase	Yes, in training phase	No	No	No	Neural network design
	ML	No	Yes	No	Yes	No	
	PS	No	Yes	Yes	No	No	
A priori based on an additional analysis	**AF**	No	Yes	No	Yes	No influence	
	AF+S	No	Yes	No	Yes	Yes	
	AF+W	No	Yes	No	Yes	No influence	
During optimisation process	**CC**	No	Yes	No	No	No	
	AC	No	Yes	No	No	No	Constraint handling technique
	MO	No	No	No	No	No	Multi-objective optimisation technique
After ungrouped optimisation	**RS**	No	Yes	Yes	Yes	No	Rounding of ungrouped solution
	UCS	No	Yes	No	No	No	

Generality

Not all methods are applicable to all types of structures: statically determinate, truss and frame structures. Besides the structure-type, some methods allow optimisation to a cost function, while others can only be used for weight optimisation. For cost optimisation, methods are regarded as applicable if they adapt their grouping to the cost function. Table 3 shows the input required in the grouping process for all methods.

All methods can be applied to statically determinate and truss structures. Furthermore, for statically determinate structures without displacements constraints, the UCS method simplifies because the grouped optimisation can be dropped.

The methods which rely on an axial force distribution are not applicable to frame structures. As in frame structures bending moments and shear forces can be dominant in the design, evaluation of only axial force is not enough.

CC, AC, MO and UCS are applicable to both cost and weight objective function as these methods evaluate the objective function in the grouping process. The objective function can be both a weight and cost function. However, the assumptions in UCS are not always satisfied in cost optimisation and the effectiveness of this method depends on the characteristics of a specific cost function. The methods ROT and NN rely on the experience of the engineering to consider cost considerations. The other methods do not include costs in their grouping process.

Table 3 - Comparison applicability

Methods		**Statically determinate structures without displacement constraints**	**Truss structures**	**Frame structures**	**Weight optimisation**	**Cost optimisation**
A priori based on geometry	**ROT**	Yes	Yes	Yes	Yes	Yes
	NN	Yes	Yes	Yes	Yes	Yes
	ML	Yes	Yes	Yes	Yes	No
	PS	Yes	Yes	Yes	Yes	No
A priori based on an additional analysis	**AF**	Yes	Yes	No	Yes	No
	AF+S	Yes	Yes	No	Yes	No
	AF+W	Yes	Yes	No	Yes	No
During optimisation process	**CC**	Yes	Yes	Yes	Yes	Yes
	AC	Yes	Yes	Yes	Yes	Yes
	MO	Yes	Yes	Yes	Yes	Yes
After ungrouped optimisation	**RS**	Yes	Yes	Yes	Yes	No
	UCS	Yes, no grouped optimisation needed	Yes	Yes	Yes	Yes

3 New grouping method

In this chapter, a new grouping method is introduced: the fully stressed combinatorial search (FSCS). This method is most familiar with UCS, as it groups members based on their weight per unit length. However, this grouping is repeated multiple times in FSCS and is based on the fully stressed design instead of the ungrouped optimum. The loop repeats until the grouping of the combinatorial search is unchanged, or the solution diverges. This framework is shown in Figure 30.

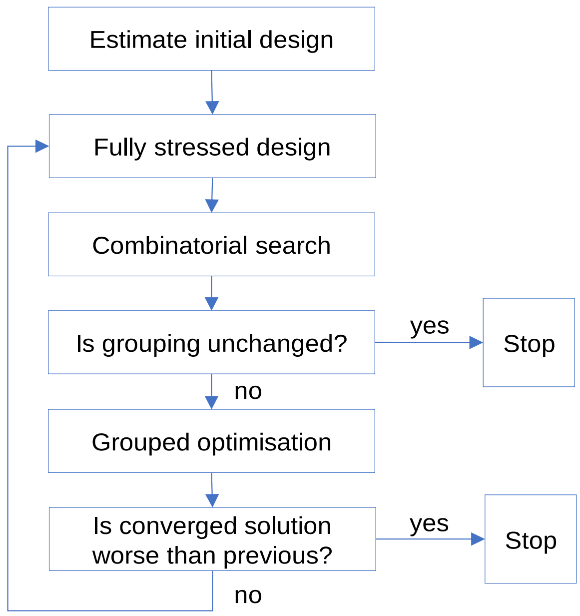


Figure 30 – Framework FSCS, grouping is based on weight per unit length of the fully stressed design, of which the limit values are found in a combinatorial search. The reference design of the fully stressed design changes during iterations.

The individual components of FSCS are prescribed in this chapter. An in-depth example of the new method on the 117-bar benchmark problem is given in Appendix A.

3.1 Fully stressed design

The fully stressed structure is the design with for each member individually the lowest weight, while still complying with the design constraints. The procedure of the fully stressed design is the same in cost and weight optimisation; the objective function has no influence on the fully stressed design. It should be noted that the fully stressed design has a high diversity of profiles, and the solution is not suitable as a final design because of its dependency on a reference design.

The fully stressed design is found by performing maximum $m \cdot n$ constraint evaluations, with m the number of independent beams and n the number of profiles. In each evaluation, one member is changed, starting from the lightest profile, while keeping the other members equal to an initial reference design. As soon as the constraint functions are met, the analysis continues with the next beam. The fully stressed design takes the lightest feasible result per beam and combines these into one design.

For statically determinate structures, the force distribution does not change by altering the design. Therefore, only one finite element evaluation for all $m \cdot n$ constraint evaluations is needed. However, the displacements do vary by a change in the design and this requires a new finite element calculation for problems with displacement constraints, just like for statically indeterminate structures.

An example is shown in Figure 31a, b and c. The figures show the design in blue in which the thickness of members represent the cross-sectional areas. In this figure, the green beam is analysed. It shows the reference design and two options. For each option, the thickness of the green beam is changed, while keeping the rest of the structure equal to the reference design. As the beam in option 1 is lighter, this option is preferred, but only if it does not violate the stress and displacement constraints. This procedure is repeated for all beams.

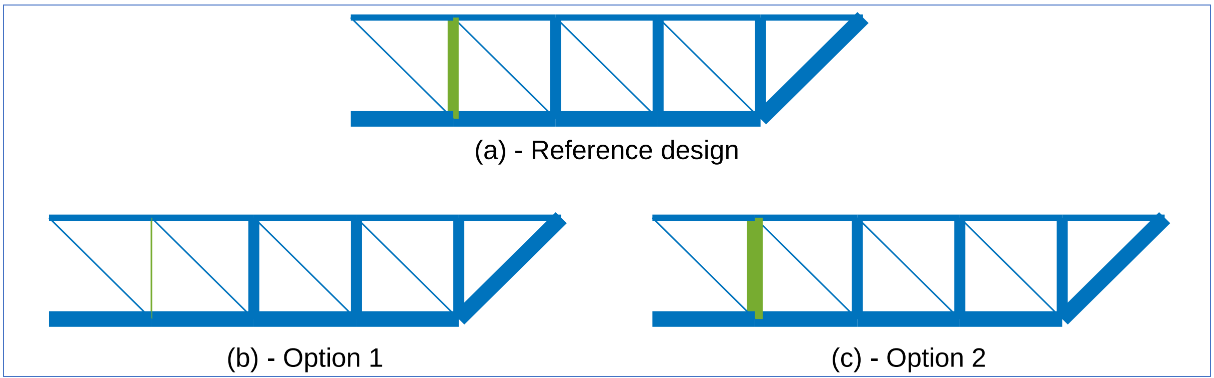
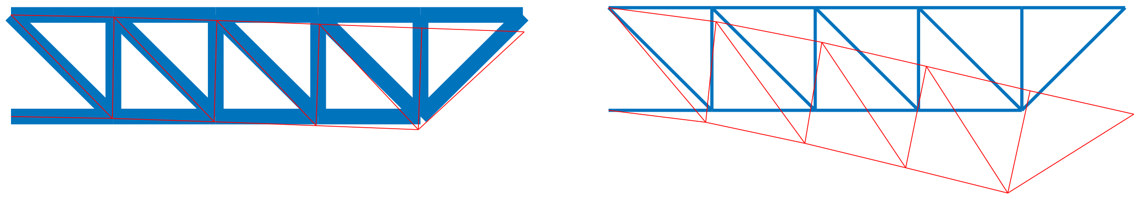


Figure 31 – Example fully stressed design, repeatedly one member of reference designs is altered, for which the lightest feasible member is preferred.

As a further explanation, an animated example of the fully stressed design is shown in Appendix C.1.

As the reference design represents all, except for one, beams in each structural analysis of the fully stressed design, it has a high influence on the force distribution and global displacement. If the reference design is infeasible, no fully stressed design can be found. Then, another reference design must be chosen. A safe choice is to select the heaviest profiles for all members. In that case, constraints are met.

An example of the influence of the reference design is shown in Figure 32a and b. These figures show two possible reference design of the 18-bar cantilever truss. If for all beams a heavy profile is chosen as a reference design, the structure deforms little as shown in Figure 32a. On the contrary, the light reference design shown in Figure 32b gives high global displacements. For the fully stressed design based on the second reference design, the displacements are strongly governed by the reference design. Therefore, the solution of the fully stressed design is different for both cases.



(a) - Heavy reference design (b) - Light reference design

Figure 32 – Example influence reference design. The global displacement is strongly influences by the references design, thereby influence the result of the fully stressed design.

In the FSCS method, the reference design changes during the repetitions of the framework: in the first iteration, the reference design is set manually, and in the following iterations, the reference design is equal to the result of the grouped optimisation. The influence of the initial manual reference design was investigated in a preliminary phase of the study. This analysis showed that no universal guideline can be given on which initial design leads to the optimal grouped structure. Therefore, three procedures were proposed, $FSCS_1$, $FSCS_2$ and $FSCS_3$:

1. The first procedure is only applicable to statically determinate structures without displacement constraints, as the choice of initial design does not influence these problems.

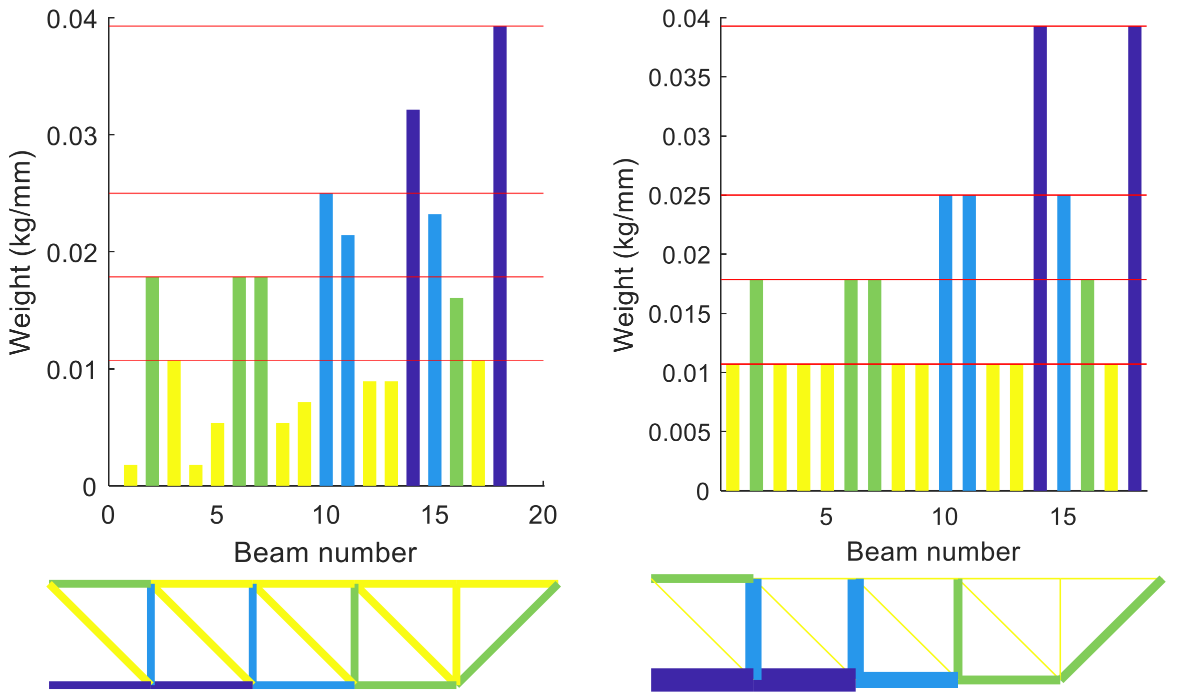
2. In the second procedure, the heaviest profiles are chosen as an initial design. This initial design will always be feasible.

3. For the third procedure, the choice for the initial design is up to the engineer. This might result in a more optimal design than $FSCS_2$. In this study, all feasible uniform initial designs were tried, requiring multiple analyses.

Of these options, $FSCS_1$ should be applied if the problem is a statically determinate structure without displacement constraints. $FSCS_2$ and $FSCS_3$ can be applied to other problems. $FSCS_3$ might find a better solution than $FSCS_2$, but it requires than an engineer specifies an initial design.

3.2 Combinatorial search

The combinatorial search takes the result of the fully stressed design and groups the members based on their unit weight per length, based on the procedure shown in Chapter 2.4.2. The result of the combinatorial search is shown in Figure 33a. The mass per unit length of the fully stressed design is plotted for all beams. The red lines represent the unit weight limits for each group. As the fully stressed design is the same as the ungrouped optimum for the this static determinate structures without displacement constraints, the result is identical to UCS.



(a) – General case

(b) – Statically determinate structures without displacement constraints

Figure 33 - Example weight distribution fully stressed design FSCS.

As a further explanation, an animated example of the combinatorial search is shown in Appendix C.4.

In the case of $FSCS_1$, the upper limits are the optimum values for all the beams in that group as shown in Figure 33b. In that case no optimisation is needed as the combinatorial search solves the entire grouping problem; both a selection of profiles is made, and the members are grouped. As a result, the framework reduces to the framework shown in Figure 34.

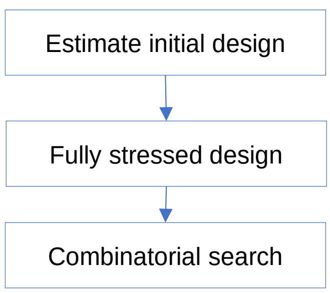


Figure 34 – Framework $FSCS_1$ for static determinate structures without displacement constraints

The combinatorial search adopted in FSCS is slightly changed with respect to the combinatorial search in UCS to allow grouping in cost optimisation. For weight optimisation, the approach of the combinatorial search in both methods is the same.

The first change for cost optimisation is that the combinations are not evaluated on total weight, but on total costs. However, the order of beams is unchanged and is based on unit weight per length. This implies the assumption that a heavier member has higher costs. The feasibility of this assumption depends on the cost function.

The second change is that additional combinations are added. This is strongly dependent on the cost function which is used. More details on which combinations should be added are provided in Chapter 6.2, in which an example is given with a cost function from literature.

3.3 Grouped optimisation

The grouping of members which follows from the combinatorial search is taken as input for the optimisation. Because the number of independent variables is reduced, the optimisation is easy to perform. It would also be possible to take the upper limits from the combinatorial search, and use these profiles in a similar way as the PS method.

For the 18-bar cantilever truss problem, the number of independent variables reduces from 18 to 4. The optimisation method now finds the optimal profiles for these 4 groups. It should be noted that finding the global optimum is not necessary as the grouped optimisation does only supply an estimation of a proper reference design for the fully stressed design. Still, at the final iteration, the global optimum should be found.

Figure 35 shows how the grouped optimisation assigns profiles to each group of members for the 18-bar cantilever truss structure.

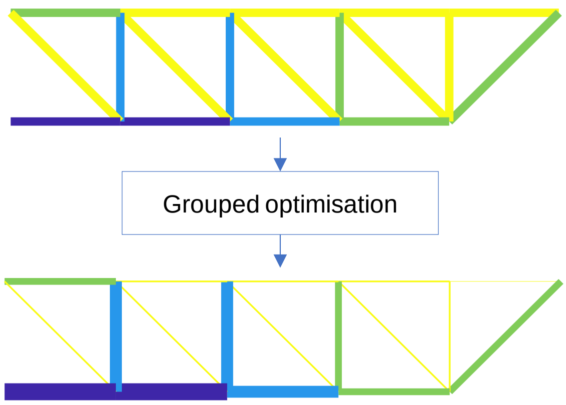


Figure 35 – Grouped optimisation on result combinatorial search in which for each member group the optimal profiles are searched for.

3.4 Loop on reference design

The result of the grouped optimisation is optimal for the grouping provided by the combinatorial search. However, that combinatorial search is based on the fully stressed design which uses an estimated reference design. To reduce the impact of the initial estimation, the process is repeated with the result of the grouped optimisation as a reference design. In most cases, this results in a more optimal design. An example of how the method converges to an optimum is given in Appendix A.

The loop steers the design in a certain direction of handling the global stiffness and global distribution of forces. Because this is an evolutionary process, the optimal way of handling the global behaviour of the structures might not be found; as the global constraints and force distribution are most influenced by the reference design of the fully stressed design, the change of individual members might not lead to a better global behaviour. The ability to solve problems with global constraints is analysed in Chapter 5.

The convergence criteria in the FSCS method, not to be confused with the convergence criteria of an optimisation method, dictate when to stop the analysis. In each iteration, starting from the second iteration, two convergence criteria are checked:

1. The grouping which results from the combinatorial search is unchanged compared to the previous iteration. If the grouping is the same, the grouped optimisation problem and its optimum are unchanged as well. In that case, the grouped result of the previous optimisation is taken as the grouped optimum.

2. The result of the grouped optimisation has a higher weight or cost than the previous optimum result. In that case, the result of the previous iteration is regarded as the optimum result. This divergence arises from a grouping which is less effective than a previous grouping in handling the global behaviour of the structure.

3.5 Conclusion

This chapter has presented the new grouping method: the fully stressed combinatorial search (FSCS). In this method, the grouping is found by a combinatorial search, which evaluates the estimated weight or costs of a restricted set of groupings based on the weight per unit length of the members of a fully stressed design. An optimisation on a reduced search space finds the corresponding best profiles. A loop is introduced to alter the reference design of the fully stressed design, allowing convergence to the optimal structure.

Three procedures are possible: $FSCS_1$, $FSCS_1$ and $FSCS_3$. For statically determinate structures without displacement constraints $FSCS_1$ should be applied, which avoids the use of a computationally demanding optimisation method. For other structures $FSCS_2$ is needed to find an optimal result, while $FSCS_3$ might find an even better result, depending on its additional user input. The added value of $FSCS_3$ is investigated with numerical experiments in Chapter 5.

Few user input is needed compared to other methods, as shown in Table 4: the number of groups is required for all procedures and a choice on the initial design for $FSCS_3$.

Table 4 – Comparison user input FSCS

Methods	Group Layout	Number of groups	Profile library	Group bandwidth	Initial design	Additional calculations
$FSCS_1$	No	Yes	No	No	No influence	n/a
$FSCS_2$	No	Yes	No	No	No	n/a
$FSCS_3$	No	Yes	No	No	Yes	n/a

This method is widely applicable to all considered problem types: statically determinate and statically indeterminate structures, truss and frame structures, and weight and cost optimisation. This is shown in Table 5. For statically determinate structures without displacement constraints, $FSCS_1$ guarantees to find the global grouped optimum without the need for optimisation. For other structures, this guarantee is not given.

Table 5 – Comparison user input FSCS

Methods	Statically determinate structures without displacement constraints	Truss structures	Frame structures	Weight optimisation	Cost optimisation
$FSCS_1$	Yes	No	No	Yes	Yes
$FSCS_2$	Yes, but apply $FSCS_1$	Yes	Yes	Yes	Yes
$FSCS_3$	Yes, but apply $FSCS_1$	Yes	Yes	Yes	Yes

For problems with displacement constraints or static indeterminate structures, this method might not be able to find an proper grouping, which is investigated in the numerical experiments of Chapter 5.

4 Theoretical comparison

In this chapter, the grouping methods listed in Chapter 2, together with the new method proposed in Chapter 3, are compared on their theoretical performance. Although the new method is a result of this comparison, this chapter includes it as well to provide a complete overview. With this comparison, the second and fourth subquestion are treated:

---

2. How do the grouping methods perform theoretically on weight and cost optimisation?

4. Do the currently available grouping methods perform well enough for application in research and practice? If not, can a new method be developed, outperforming current methods?

This chapter evaluates the grouping methods on their effect on the efficiency and accuracy of optimisation. This includes the size and complexity of the search space, potential exclusion of the global grouped optimum, and the computational effort of additional calculations. Based on this comparison, a selection of the most promising grouping methods is made. Chapter 5 compares this selection in numerical experiments.

4.1 Size search space

The search space of the problem is altered by the grouping. This influences the optimisation process; the smaller the search space, the faster the optimisation process proceeds. Furthermore, for a large search space, an optimisation algorithm may not be able to find the global optimum (Templeman 1988). The perfect method limits the search space to speed up optimisation.

For methods in which optimisation is performed on the ungrouped problem, each of the n beams has m available profiles, leading to a total search space of m^n options. If the beams are grouped to k groups, the number of options for each beam group is the same, but the total number of independent beams reduces. Therefore, the search space reduces to m^k options. For the PS method, the number of available profiles for each member reduces to the number of groups, thus leading to the search space of k^n options. The method CC, which groups both beams and profiles, has for each group m available profiles available. Simultaneously, each beam can be part of one of the k groups, leading to a search space of $m^k \cdot k^n$ options.

For all methods, the search space includes solutions which have less than the k groups. Therefore, all methods, including CC, might converge to a solution which does not have the desired number of groups. The actual number of solutions for a certain number of groups is N_{exact}:

$$N_{exact} = \binom{m}{k} \left\{ \begin{matrix} n \\ k \end{matrix} \right\} \tag{4.1}$$

In which $\left\{ \begin{matrix} n \\ k \end{matrix} \right\}$ is the Stirling number of the second kind (Knuth 2011). This number gives the number of possibilities to group n members in k non-empty groups. It is defined as:

$$\left\{ \begin{matrix} n \\ k \end{matrix} \right\} = \frac{1}{k!} \sum_{i=0}^{k} (-1)^i \binom{k}{i} (k-i)^n \tag{4.2}$$

Furthermore, $\binom{m}{k}$ and $\binom{k}{i}$ are binomial coefficients. $\binom{m}{k}$ gives the number of ways to select k distinct profiles from a profile library with k and is defined as:

$$\binom{m}{k} = \frac{m(m-1)\cdots(m-k+1)}{k(k-1)\cdots 1} \tag{4.3}$$

Table 6 shows how the grouping methods influence the size of the search space for each of the eight benchmark problems. Details of the benchmark problems can be found in Appendix E.

Table 6 – Search space benchmark problems in powers of 10

	Ungrouped	**Grouped beams**	**Grouped profiles**	**Grouped beams and profiles**	**Grouped optimum**
	NG, RS (step 1) UCS (step 1), total AC, total MO,	**ROT, NN, ML, AF, AF+S, AF+W, FSCS UCS (step 2)**	**PS**	**CC feasible AC**	**desired MO**
Theoretical	m^n	m^k	k^n	$m^k \cdot k^n$	$\binom{m}{k}\left\{{n \atop k}\right\}$
18-bar truss	25	5	10	16	13
65-bar truss	53	6	19	26	23
72-bar truss	22	5	9	15	12
112-bar truss	26	4	7	12	10
160-bar truss	61	9	29	39	33
15-bar frame	14	4	1	6	6
117-bar frame	42	7	17	25	21
147-bar frame	50	8	28	36	30

Table 6 shows that the ungrouped search space is much bigger than the grouped search space for the benchmark problems, while PS and CC result in a search space size in between those two. However, if the number of groups is high, the size of the search space of CC might exceed the ungrouped search space. That is the case for $k > \frac{n \cdot \mathrm{W}\left(\frac{m \cdot \log(m)}{n}\right)}{\log(m)}$, with W the Lambert-W-function (Wolfram Research 2020). For the 18-bar truss, this is for $k > 8$. Nonetheless, it is assumed a low number of groups is desired in the final design. In that case, the search space of CC is smaller than the ungrouped search space. The search space of PS has a maximum size equal to the ungrouped search space when as much profiles are selected as there are members in the structure, that is when k is equal to m.

It should be noted that for AC, only a small part of the total search space is feasible; a large portion of the search space of AC has more than the maximum number of groups.

For MO, the entire ungrouped search space is explored by the optimisation method, but only solution which have the specified number of groups are desired. As with AC, this leads to evaluating many undesired options.

Furthermore, for UCS, the search space of the first step is very high, while the second step has a reduced search space. The same holds for the first step of RS. The second step of RS is not shown as no general guideline is given for this step.

As an example of the size of the search spaces, Figure 36 shows the relative size of the search spaces when applying grouping methods. The areas of the circles represent the size of the search space. However, the picture is not to scale with the values shown in Table 6. If, for the 18-bar truss problem, the biggest circle would be the size of the earth's surface, the CC-circle would be the size of TU Delft Campus, the blue shaded circle would be the size of two tennis courts, the PS-circle would be the size of 8 A4-papers, and each of the smallest circles would be the size of the head of a pin. The optimum solution has the size of the cross-sectional area of a corona virus. For the 160-bar truss problem, the search space of the ungrouped problem exceeds the surface area of the observable universe if the same coronavirus represents the optimal solution.

In Figure 36, FSCS is shown as multiple circles because it performs multiple grouped optimisations. Futhermore, the full, feasible and desired search spaces of AC and MO are shown as well. Similarly, for UCS the first and second step are indicated separately. The search space of ROT and NN might partly overlap, as NN is an automatisation of ROT. Other search space might overlap as well, depending on the problem.

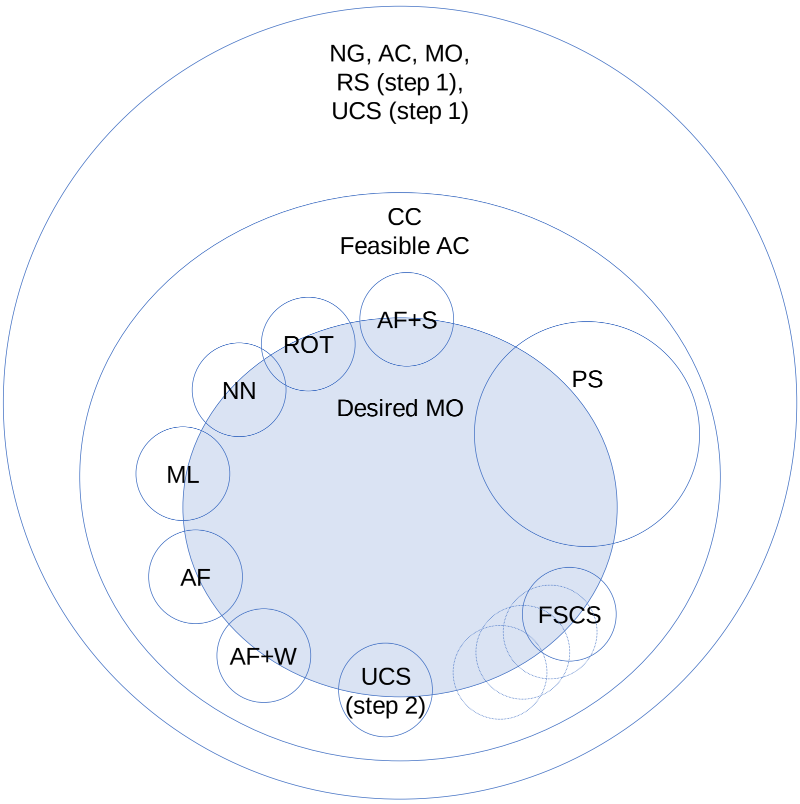


Figure 36 – Relative size and relation search space, blue shading shows potential location of grouped optimum.

4.2 Exclusion global optimum

By altering the search space, grouping methods not only alter its size, but they might exclude the global grouped optimum in doing so. In the case of exclusion of the global optimum, the optimisation method has no other possibility than to converge to a suboptimal result. However, a suboptimal result might be close to the optimal result, as is investigated in the numerical experiments of Chapter 5. Therefore, the ideal grouping method should include the global or a near-global optimum in its search space.

Figure 36 not only shows the relative size, but also the relation of the search spaces of all grouping methods. The global grouped optimum is located somewhere in the blue shaded circle. Each method is represented by a circle to show its potential of finding the grouped optimum.

AC and MO have a search space which includes the global optimum, but their search space is large. Similarly, the search space of CC guarantees to include the global grouped optimum. However, this search space can exceed the ungrouped search space.

The first step of RS and UCS does not search for the grouped optimum directly, but first for the ungrouped solution which might be located anywhere in in the ungrouped search space. Subsequently, this result is adapted to find the grouped solution. However, the ungrouped solution might not be similar to the grouped optimum, so the second step of these methods does not guarantee to include the global grouped optimum.

All other grouping methods search a small portion of the search space and might exclude the global grouped optimum in doing so. FSCS iteratively searches a small portion of the blue shaded area, which increases the possibility of including the global optimum.

As it has been shown, all grouping methods might lead to a search space in which some solution, which can be their local optima, are not part of the blue shaded area; these solutions have less groups than specified. However, in weight optimisation, more distinct profiles are in general beneficial for the weight, which increases the chance of finding a solution in the blue shaded area.

It is not known beforehand whether the resulting grouped solutions approach the real grouped optimum. Neither is it known whether the number of groups is equal or less than desired. Both aspects are investigated in Chapter 5 with numerical experiments.

Furthermore, the global optima for cost and weight are not necessarily the same but both optima are surely located in the blue shaded area of Figure 36. As a result of this unknown location, a grouping method may exclude the global grouped optimum in cost optimisation but include it in weight optimisation, or the other way around. For UCS and FSCS, the search space shifts in the case of cost optimisation because the search space is changed. For UCS this is the case for the second step, as it performs a combinatorial search of the cost optimum. For FSCS the cost influences the adapted combinatorial search, and influences the result of the optimisation in each loop. Therefore, the chance to include the cost optimum, as with the weight optimum, is higher for these methods than for the grouping methods which group members before the optimisation starts.

For statically determinate structures without displacement constraints, the location of the solution for the global grouped optimum is known because it is the solution of $FSCS_1$ and UCS and no optimisation is required to find it. This is shown in Figure 37 in a similar way as in Figure 36. The smallest circle represents the optimum grouped solution for a specified number of groups.

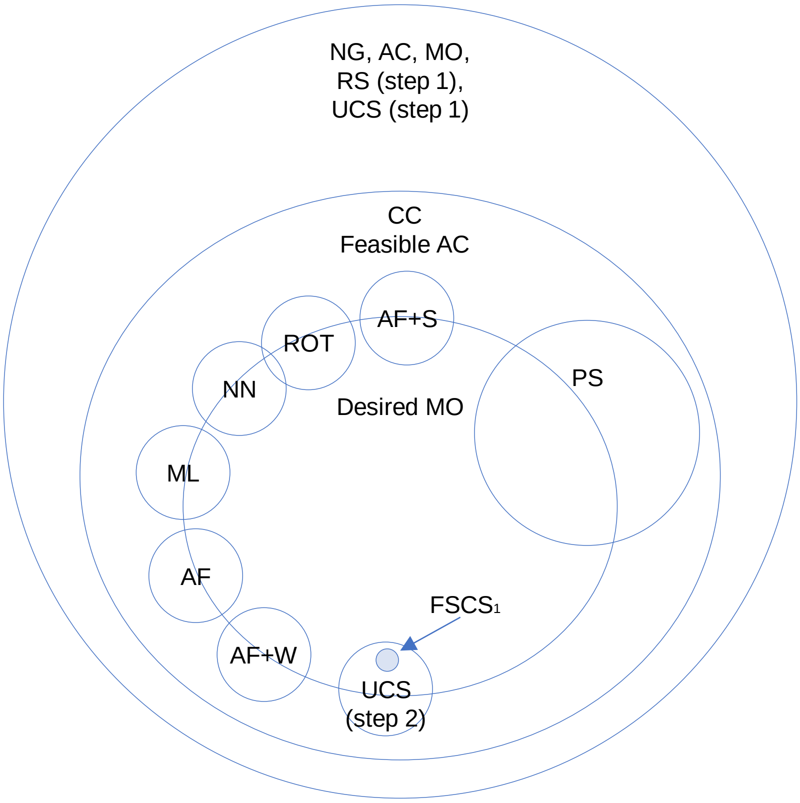


Figure 37 –Relation search space for statically determinate structures without displacement constraints, blue shading shows location of grouped optimum.

4.3 Complexity search space

Besides changing the size and potential exclusion of the global optimum, the format of the search space can be changed. This can cause a more complex search space if more local optima are introduced. Conceptual graphs illustrate this change. In these graphs, the design variable on the horizontal axis represents all design variables: a low value indicates light members, and a high value indicates heavy members. The objective function shows the corresponding weight or cost of the structure and constraint function the yield, buckling and displacement limits. In reality, the design variables cannot be shown along one dimension, and the objective and constraints functions are a function of all design variables; the graphs in this chapter are a two-dimensional simplification of the multi-dimensional search space.

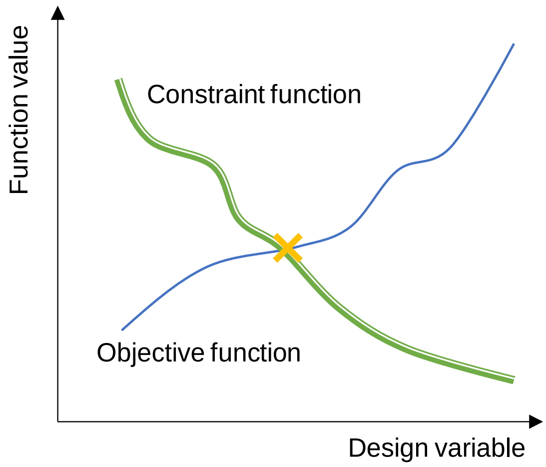


Figure 38 - Example simple search space: only one minimum is present.

Simple search space

To understand the complex search space, the representation of a simple search space is shown in Figure 38. This figure represents the search space of an ungrouped weight optimisation problem. The graph shows the objective function in blue and the constraint function in green. The constraint function is an inequality constraint for which the side of the thin green line is valid, so the top right part of the graph. The optimum solution is shown as a yellow cross. This point is optimal because a lower design variable gives an infeasible result, and a higher design variable a higher weight. As it is the only local optimum, it is the global optimum. Therefore, an optimisation method can easily solve this problem.

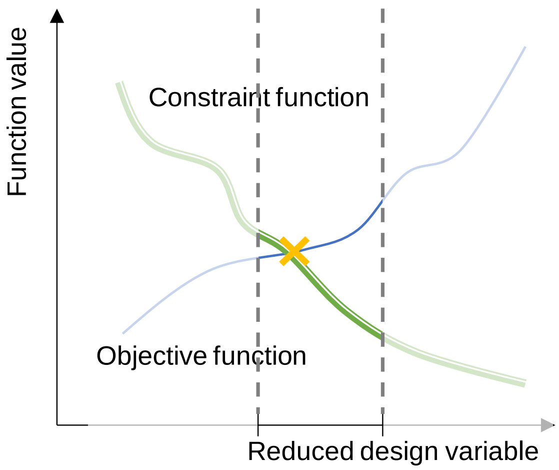


Figure 39 - Example search space grouping a priori a selection of the design variables reduces the search space but does not create more minima.

Grouping a priori

All methods which apply grouping before the optimisation starts, reduce the options for the design variables, without changing their order. This is the case for ROT, NN, ML, PS, AF, AF+S, AF+W and FSCS, as these methods reduce or the profile, or the geometric diversity. This is shown in Figure 39. As clearly visible, the complexity of the search space is not altered as the number of optima is still one. In this case the global optimum is included. But it might be excluded, if the bounds move left or right.

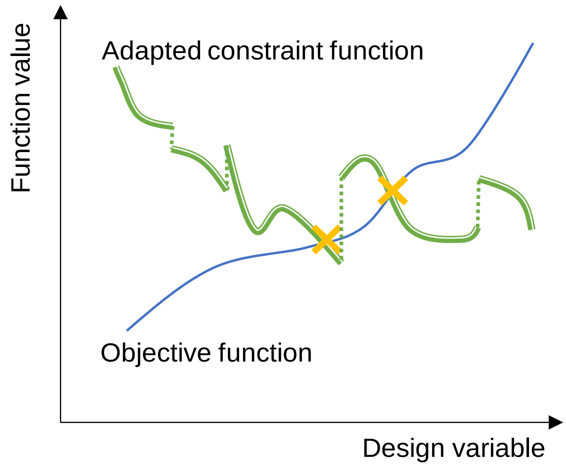


Figure 40 - Example altered constraint function in search space AC: creation of additional local optima.

Changing constraint function

AC adapts the constraint function, by adding a term for the number of groups. This is represented by Figure 40. By adapting the constraint function, new local optima appear; for both yellow crosses, a small change in the design variable leads to a worse or infeasible design. The addition of these local optima complicates the process of finding the global optimum for an optimisation algorithm.

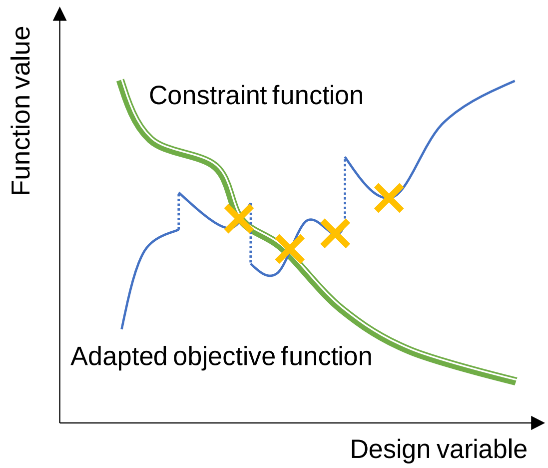


Figure 41 - Example altered objective function in search space MO: creation of additional local optima.

Changing objective function

MO adapts the objective function, in a similar way as AC influences the constraint functions. This is represented by Figure 41. Now, local optima not only appear at the edge of the feasible domain, but also in the feasible domain, as indicated by the two right yellow crosses. Again, the process of finding the global optimum is difficult due to the presence of many local optima.

It should be noted that in the case of implicit cost optimisation, the objective function is adapted in a similar manner.

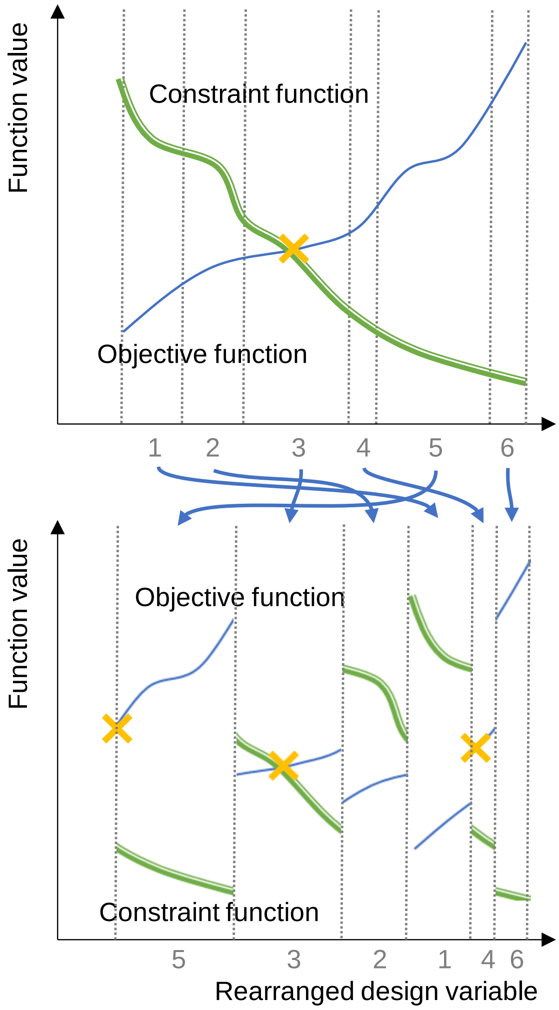


Figure 42 - Original and rearranged search space CC: creation of additional local optima.

Changing coding design variable

Finally, CC does not adapt the constraint or objective function, but the encoding of the design variables. This influences the search space too, as shown in Figure 42. The objective and constraint functions have similar values, but the design variable encoding for those value is changed. This leads to a reordering of parts of the design variable and creation of local optima.

Conclusively, the search space is only kept to the same complexity if a grouping is applied before the optimisation starts. On the other hand, it is made more complex by all methods which simultaneously determine which profiles to use and where those profiles should be located. Chapter 5 investigates the impact of the complexity on the optimisation process in numerical experiments.

4.4 Additional calculations

While the optimisation procedure takes a lot of computational effort, the calculation of the grouping methods cannot be neglected. The computational effort needed for additional calculations with NN, RS, AC, MO, FSCS, UCS is significant. On the other hand, the structural analysis needed for AF, AF+S, AF+W has neglectable computational impact: these grouping methods require one structural analysis evaluation, while the optimisation process itself requires many thousands of structural analysis evaluations.

The efficiency of NN is strongly dependent on the design of the neural network. Nonetheless, the training of a neural network requires extensive training with optimised solutions, which might not be available either.

For RS, the computational effort of the rounding procedure of the ungrouped result is unknown, as no general guidelines are available. UCS provides a method of handling with the ungrouped result by its combinatorial search. This search does not include any structural evaluations for UCS, but for large problems this search can require significant computational effort. Table 7 shows how many objective function evaluations are needed for the benchmark problems described in Chapter 5.2.

FSCS requires a similar combinatorial search as UCS, performs a fully stressed design, and it does both in multiple iterations. The fully stressed design is found by evaluating the constraints function, including a structural analysis, for maximum $m \cdot n$ options in the search space until the constraint functions are satisfied. For statically determinate structures without global constraints, this only requires one structural analysis. Table 7 shows the total number of finite element and constraint function evaluations per iterations. Chapter 5 evaluates the relative influence of these calculations on the total computational effort in numerical experiments. This influence is expected to be marginal as the optimisation methods itself uses many more finite element evaluations.

Table 7 – Computational effort additional calculations UCS and FSCS

	Combinatorial search UCS and FSCS	**Fully stressed design FSCS**	
	Number of objective function evaluations	**Maximum number of FEM evaluations**	**Maximum number of constraint function evaluations**
Theoretical	$\binom{m-1}{k-1}$	Statically determinate without global constraints: *1* Other: *max(m·n)*	*max(m·n)*
18-bar truss	680	1	450
65-bar truss	4960	1386	1386
72-bar truss	455	400	400
112-bar truss	105	688	688
160-bar truss	435897	1596	1596
15-bar frame	6	552	552
117-bar frame	353	515	515
147 bar frame	324632	900	900

4.5 Conclusion

This chapter has presented the theoretical comparison of the existing grouping methods and potential for a new grouping method, thereby answering subquestion 2 and 4.

---

2. How do the grouping methods perform theoretically on weight and cost optimisation?

The size and simplicity of the optimisation problem, possible inclusion of the global optimum, and the ease of additional calculations have been compared. The relative performance of all methods is shown in Figure 43. This figure conceptually summarises the theoretical comparison. If a bar is filled completely, its performance is good, while a poor performance is represented by a partly filled. Furthermore, the green bars represent an acceptable, yellow a questionable and red a poor performance. The perfect method would have all bars filled completely in green. For the methods RS, UCS, $FSCS_2$ and $FSCS_3$, the properties of the search space are shown for each step or iteration separately. However, for $FSCS_2$ and $FSCS_3$ the number of iterations is not known beforehand. For RS, the procedure for the second step is unknown, as no general applicable procedure has been proposed in literature. For PS and CC, the search space is strongly dependent on the number of groups, which is shown with the light yellow bars.

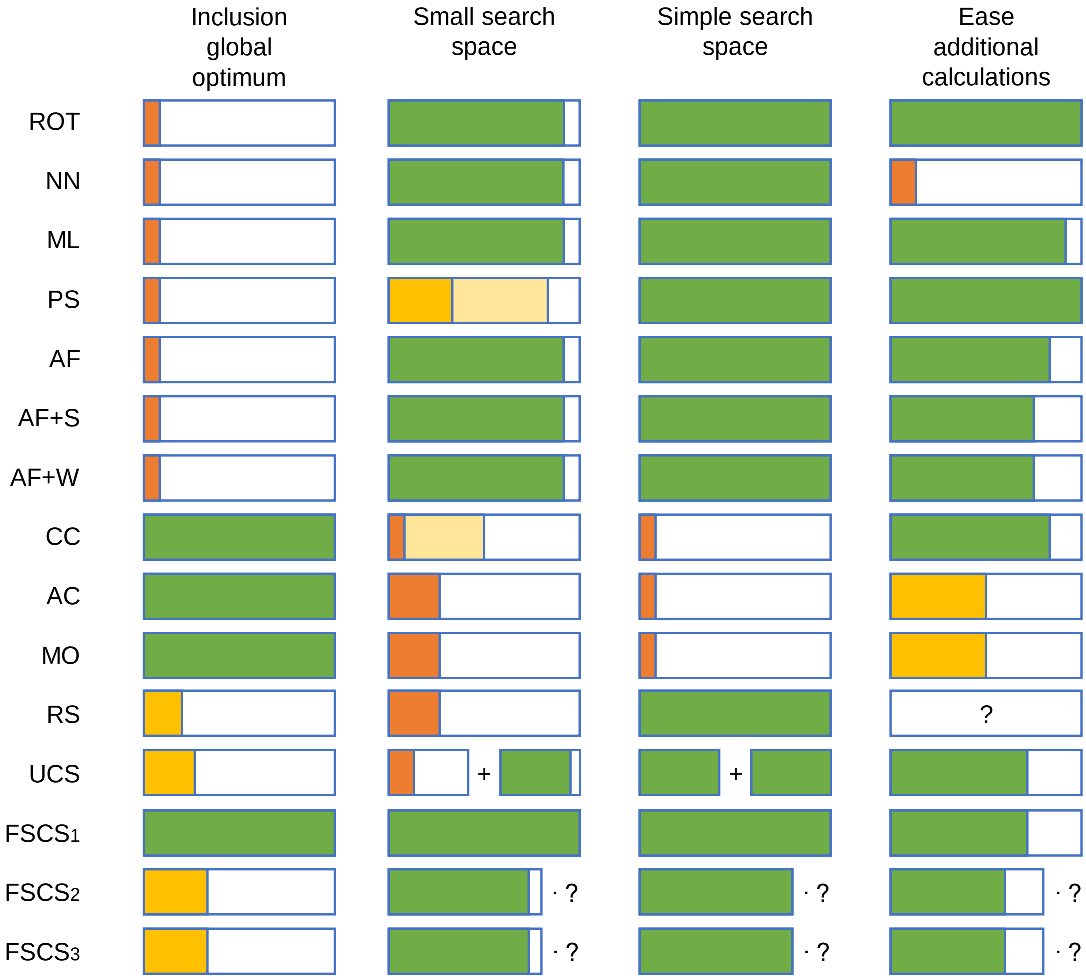


Figure 43 – Overview theoretical comparison grouping methods

The comparison shows that CC is the only method which reduces the search space while guaranteeing the inclusion of the global grouped optimum for all kind of structures. On the contrary, this is only true for a reduced number of groups. Furthermore, the search space is altered which makes it harder to find this optimum. The methods MO and AC include the global grouped optimum in their search space as well, but are ineffective in doing so because the search space is big. As with CC, the search space is made more complex by these methods. Finally, $FSCS_1$ includes the global optimum in its grouping too. Moreover, it finds the optimum grouped solution without the need for optimisation. However, this method is only applicable to statically determinate structures without displacement constraints.

All other methods do not guarantee the inclusion of the global grouped optimum in reducing the search space. Nonetheless, if the solution of these methods is close to the global grouped optimum, these methods are of highly practical use. However, this cannot be proven theoretically and is investigated in Chapter 5 with numerical experiments. Still, $FSCS_2$ and $FSCS_3$ search the search space iteratively, so these methods have a higher chance of including the global optimum in the search space. Similarly, the grouping performed by UCS on the ungrouped solution might be close to the grouped solution. For the other methods, the optima of their reduced search spaces have a low chance of being close to the global optimum. This is the result of assumptions made in their grouping process.

The number of computations needed in additional calculations is insignificant for most methods. Exceptions are UCS and RS, of which the required ungrouped search makes these methods impractical, as well as the high complexity of NN.

Finally, I expect the number of computations of the combinatorial search and fully stressed design of FSCS to be insignificant, but this is verified in the numerical experiments of Chapter 5. If a lot of iterations are required, the total computational effort of these methods might become high. Furthermore, for problems in which the combinatorial includes many combinations, the total computational effort of this computationally cheap individual analyses can become high.

4. Do the currently available grouping methods perform well enough for application in research and practice? If not, can a new method be developed, outperforming current methods?

From the theoretical comparison it can be concluded that potential is available for a new method. This new method should be able to reduce the search space significantly. In doing so, it should include the global optimum or a local optimum close to this value. Furthermore, it should not vary both profiles and groups of members at the same time, or alter the objective and constraint functions; these adaptations make the search space more complex. Finally, additional calculations are allowed, as long as the number of computations is insignificant compared to the optimisation itself.

The new FSCS method obeys these demands. It searches multiple smaller search spaces to approach the global grouped optimum. Although it does not theoretically enclose the global grouped optimum, the iterative behaviour allows convergence to the global grouped optimum. In doing so, this method only varies profiles during the optimisation, thereby not making the search space more complex. The required computations of the additional calculations, the number of required iterations, and the ability to find the (global) optimum, are investigated in the numerical experiments of Chapter 5.

Selection for numerical experiments

Not all grouping methods were compared in the numerical experiments, as the theoretical performance of some methods was judged poorly. Therefore, the methods NN, ML, PS, AF+W, MO, AC and RS were excluded. Furthermore, the methods AF and AF+S were only applied for truss-structures, as these methods are based on the axial force distribution, which is the only internal force for truss-structures.

NN was excluded because this grouping method automates ROT by using a neural network. It was expected that the results from ROT are not optimal, so an automatisation would be unnecessary. I expected ROT to perform poorly because the complexity of the grouping problems is too high to be optimally solved by an engineer.

ML was excluded because it relies on the assumption that member length is the main factor influencing the structural behaviour of a structure. This assumption was regarded as too simplistic to cover a wide range of structures.

PS was excluded because selecting the optimum set of profiles, independent of their location in the structure, was regarded as very difficult. No examples and rules of thumb are available in literature to do so.

AF+W was excluded because it is an adaption of AF with additional user input needed, for which no general guidelines are available. The use of an automatic procedure for grouping was preferred in this study.

AC was excluded because the feasible search space is much smaller than the total search space, which makes the search space highly complex. In literature and in a preliminary analysis on the 18-bar truss, it was concluded that its highly infeasible search space prevents the optimisation algorithm of finding the global optimum.

MO was excluded because the it gives many solutions which do not have the right amount of groups. In essence, this method generates an implicit cost optimisation problem, with an artificial cost function. Such a problem is hard to solve, as demonstrated in a preliminary analysis of the 18-bar truss.

Finally, RS was excluded because it does not provide general guidelines to round the ungrouped solution. UCS does have a procedure of adapting the ungrouped solution, therefore that method was preferred.

Conclusively, the methods ROT, AF, AF+S, CC, UCS and FSCS were investigated in the numerical experiments.

5 Numerical comparison

In this chapter, the numerical experiments are presented, thereby answering the third and fourth subquestion. Although the new method is a result of this comparison, it is included as well.

3. How do the grouping methods perform on benchmark problems for minimum weight?

4. Do the currently available grouping methods perform well enough for application in research and practice? If not, can a new method be developed, outperforming current methods?

As an introduction, it is explained how the numerical experiments were performed. Then, an overview is given of the benchmark problems and the genetic algorithm, which was chosen as the optimisation method. Subsequently, the results of the experiments are shown and discussed.

5.1 Method

For each benchmark problem, all grouping methods were applied multiple times. This was done because of the stochastic behaviour of the genetic algorithm. If multiple analyses found the same solution, that solution was regarded as the optimum solution for that method. Furthermore, the options of the optimisation method were varied in these analyses. This was done to find the lightest design, for every case individually; for each case, a custom set of options showed to be effective. The final weight and a selection of performance criteria were extracted from the full set of solutions per grouping method and benchmark problem. A flow-chart of this procedure is shown in Figure 44.

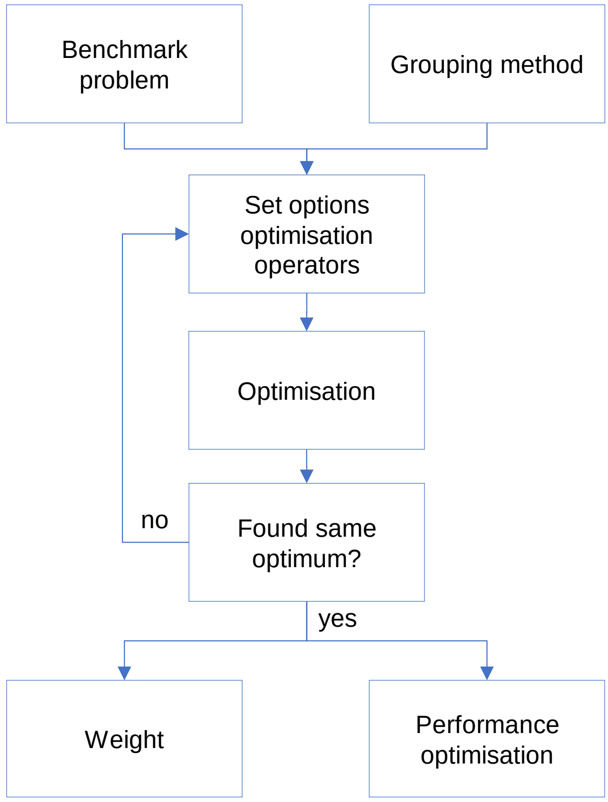


Figure 44 – Flow chart numerical experiments. For every grouping method and benchmark problem, the optimisation parameters are altered.

The truss and frames were modelled using the matrix stiffness method, an implementation of the finite element method. A code by H. Rahami (2019) was used, but it was adapted it to maximise efficiency in optimisation. Therefore, the element stiffness matrix was defined in terms of the stiffness properties of the profiles per member. This allowed to alter only the stiffness properties during the optimisation, without defining new element stiffness matrices. This implementation was validated with results from MatrixFrame in a preliminary phase of the study.

5.2 Benchmark problems

The performance of the grouping methods was evaluated on eight benchmark problems. The benchmark problems were taken from literature on optimisation methods. However, diversity in these problems is large, especially in the field of frame structures. Therefore, a selection was made to cover a wide range of structures in terms of statically determinacy, truss and frame structures, 2D and 3D structures, application of stress and displacement constraints, and number of beams. An overview of the chosen benchmark problems is given in Table 8. Appendix E describes the benchmark problems in detail.

Table 8 - Overview benchmark problems

Benchmark problems	**Statical determinacy**		**Structure**		**Dimensions**		**Constraints**	
	Deter-minant	**Inde-termi-nant**	**Truss**	**Frame**	**2D**	**3D**	**Stress**	**Displa-ce-ment**
18-bar cantilever truss	X		X		X		X	
65-bar truss beam	X		X		X		X	X
72-bar truss tower		X	X			X	X	X
112-bar truss dome	X		X			X	X	X
160-bar truss tower		X	X			X	X	X
15-bar 3-storey frame		X		X	X		X	
147-bar 3-storey frame		X		X		X	X	X
117-bar 9-storey frame		X		X	X		X	X

The problems were adapted from the original problem in literature, so that similar constraints functions and profile databases were used. Therefore, results from optimisation in this study differed from results for the same geometries in literature.

For all except for the 18-bar cantilever truss, elementary grouping was applied; some bars were grouped as part of the problem description.

5.3 Genetic algorithm

The genetic algorithm, developed by J. Holland (1975), was chosen as the optimisation method for multiple reasons:

1. The genetic algorithm can consider discrete variables and can incorporate all grouping methods.

2. The genetic algorithm has been known to be little influenced by local optima in finding the global optimum (Arora 2017).

3. The genetic algorithm has been the most well-known nature-inspired search method and has been used often as a reference case for comparison with new optimisation methods in structural engineering (Saka and Geem 2013; Stolpe 2016).

The genetic algorithm is based on the evolutionary theory of Darwin and is best explained using biological populations, for example, a herd of cows. These cows should be healthy to survive. As this herd of cows mate and get calves, parts of their DNA are mixed, and small mutations arise in the copying of this DNA. These variations alter the calves slightly compared to their parents, and some variations may give more healthy calves than their parents. Survival of the fittest dictates that the healthiest animals survive and are more likely to get offspring. Consequently, in each generation the herd of cows become healthier.

The genetic algorithm mimics this evolutionary process and applies it to mathematical problems (Holland 1975; Goldberg and Samtani 1986). The cow's DNA is replaced by a collection of number for each design. These numbers encode for the profile of the members in the design, just like DNA encodes for the health of cows. The amount of designs, equivalent to the number of cows in a herd, is set by a parameter, the population size. Mathematical procedures mimic the mating and mutating behaviour by mixing the sets of numbers and creating random variations. Moreover, some of the best designs, of which the number of designs is set by a ratio of the total population size, are regarded as elite solutions. These designs are unaffected by mixing and mutating but are directly copied to the next generation. This prevents the population from losing its best solution. In the new generation of design, the heaviest designs do not survive, while the lightest design get a lot of offspring with the same procedure. This continues until a stopping criterion is reached. Frequently, this is chosen to be a certain number of generations with an unchanging best solution.

Various mathematical implementations exist for the genetic algorithm. The genetic algorithm implementation from the Global Optimization Toolbox of MATLAB was used in this study, which is based on the work of Deep and Deb (Deb 2000; Deep et al. 2009).

The mathematical implementation of the genetic algorithm is explained in more detail in Appendix D. Appendix E gives an example for the steps of the genetic algorithm in the 18-bar cantilever truss problem, while Appendix C.1 shows the result in an animation.

5.4 Results

For each benchmark problem and grouping method, this chapter shows the following weight and performance criteria in tables:

1. Minimum weight of the optimised structures. The weight is rounded to an integer value. This value shows the ability to find or approach the global grouped optimum.

2. Dominant constraint for the final solution. Which is the constraint function which has the maximum value in the final design. It can either be a strength constraint in the case of stress constraint of an individual member, or displacement in the case a global displacement constraint. This criterium is included because FSCS and UCS might not be able to correctly group displacement dominated problems.

3. Number of groups in optimum solution. As the grouping methods allow for maximum the desired number of groups, this number shows the certainty on finding the desired number of groups.

4. Percentage of feasible analyses that found the same minimum optimum. The percentage is shown in two significant digits and shows the certainty of finding the grouped optimum of the corresponding method. If this percentage is high, the optimisation problem is easy to solve. For $FSCS_3$, this percentage includes all feasible uniform initial designs for the fully stressed design. For UCS, the percentage does only show the certainty of the second step of this method, but not the certainty of the required ungrouped optimisation. If the percentage of the optimisations in which the global optimum is found is low, the estimated density function of the final weights is plotted using a kernel estimation (Peter D. 1985). In that case, a narrow distribution indicates a simple optimisation problem and a wide distribution marks a complex search space with many local optima.

5. Mean computational time of optimisation. The time is shown in two significant digits. This value shows the ease of finding the grouped optimum, just like the former criterium. A short computational time is preferred. For UCS, the mean computational time of NG is included, as UCS uses the result of NG in its analysis.

6. Mean computational time of additional calculations. The time is shown in two significant digits. This value shows the computational time of the additional calculations of AF, AF+S, FSCS and UCS. For FSCS, this computational time is the total calculation time of additional calculations in all iterations.

7. Mean number of finite element method evaluations (FEM). This number is rounded to an integer value. It is closely related to the computational time, but this number gives an indication of the required resources for optimisation. It includes both the FEM evaluations during optimisation and during the additional calculations. For UCS, the mean number of FEM evaluations of NG is included as well. For the fully stressed design, it includes the maximum amount of FEM evaluations for all iterations.

8. Mean number of iterations required for $FSCS_2$ and $FSCS_3$. This number is rounded to two significant numbers. As the number of required iterations of $FSCS_2$ and $FSCS_3$ is not known beforehand, the computational effort may be high.

Furthermore, the resulting groupings are discussed in this chapter. Therefore, the best grouped solution is graphically shown for every grouping method and benchmark problem. The colours in these graphical representations represent groups in the final solution, while the thickness of the member represents the relative weight per unit length for the profile solution. The ungrouped solution shows grouping as well, following from elementary linking and as the ungrouped optimisation might find members with the same profile.

It should be noted that the order of colours and thickness of different figures have a different scale. As a result, a thin beam in one figure might have a higher weight per unit length than a thick beam in another figure.

For the exact solutions and performance criteria per analysis, the reader is referred to the data files: http://doi.org/10.4121/uuid:4e32b29f-6647-4a36-9ea1-8931c88f8864.

5.4.1 18-bar cantilever truss

As the 18-bar cantilever truss is a statically determinate structure without displacement constraints, no optimisation is required for methods which group member before the optimisation starts. Instead, a fully stressed design can find the optimum result, which was done in this study. This fully stressed design requires just one finite element evaluation.

From the results, shown in Table 9, it was found that $FSCS_1$ and UCS gave the lowest weight solution with less computational effort than CC. ROT, AF and AF+S converged to a structure with a higher weight. Of these methods, AF_1 performed best and AF_4 worst. Finally, all methods converged to the specified number of groups.

Table 9 – Results 18-bar truss

Method	**Weight of optimum solution (kg)**	**Dominant constraint in optimum solution**	**Number of groups in optimum solution (-)**	**Number of times optimum found (%)**	**Mean time optimisation (s)**	**Mean time additional calculation (s)**	**Mean number of FEM evaluations (-)**
NG	1855	Strength	12				1
ROT	2836	Strength	4				1
AF_1	2704	Strength	4			0.33	1
AF_2	2433	Strength	4			0.31	1
AF_3	2605	Strength	4			0.33	1
AF_4	3113	Strength	4			0.31	1
$AF+S_1$	2492	Strength	4			0.45	1
$AF+S_2$	2677	Strength	4			0.46	1
$FSCS_1$	2201	Strength	4			1.3	1
CC	2201	Strength	4	15	33		26250
UCS	2201	Strength	4			0.84	1

The optimum grouped solution of $FSCS_1$, CC and UCS, presented in Figure 45i, showed that these methods efficiently group member in both tension and compression. For example, while the top left member was loaded to a high tensile force, it was grouped to the bottom right members in compression. These compressive members were loaded less in absolute sense but buckling required additional capacity. For the AF and AF+S methods, compressive and tensile members were not combined by these methods, except for AF_4. However, AF_4 gave a heavy solution.

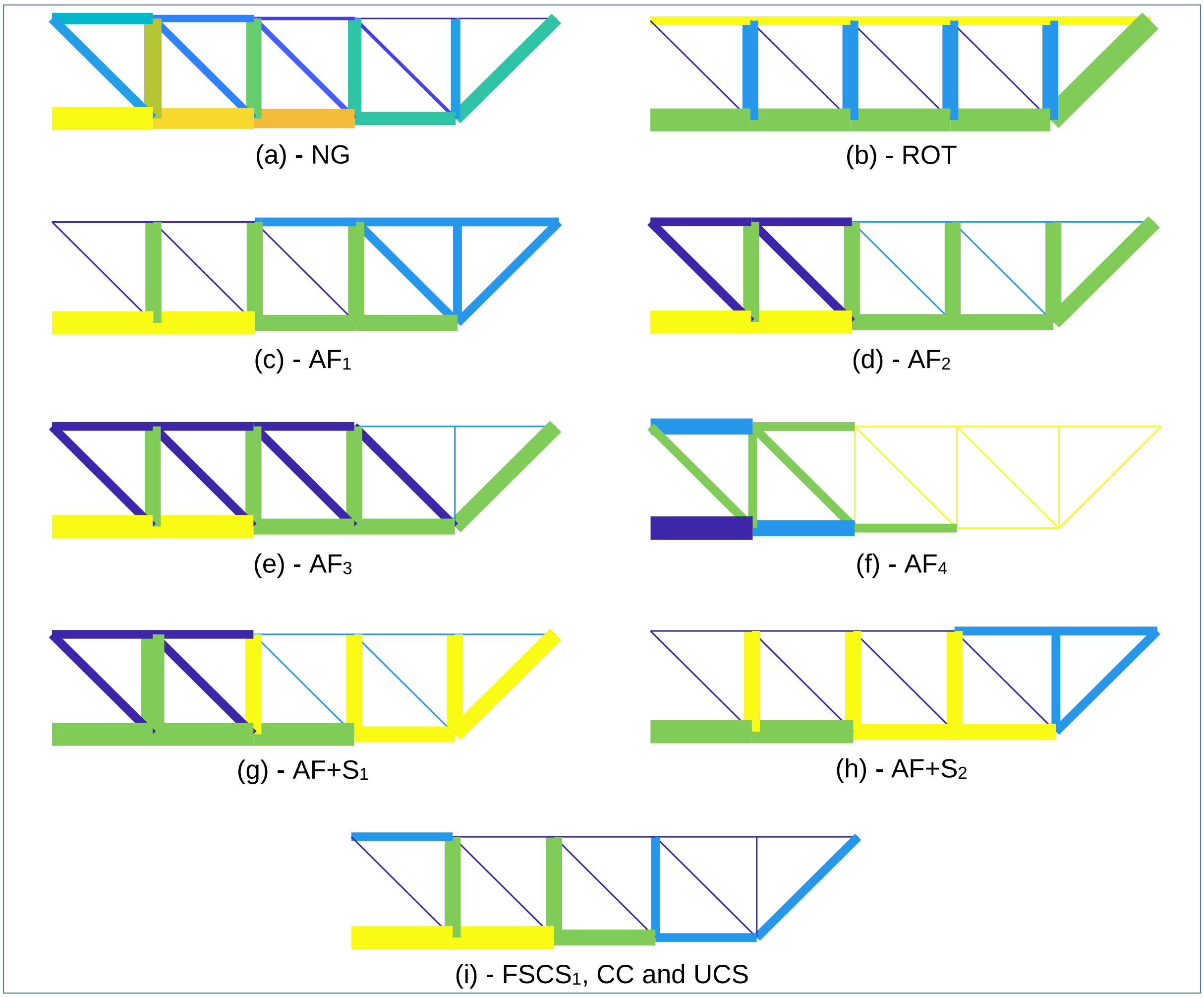


Figure 45 – Results groups and weight per unit length 18-bar problem

5.4.2 65-bar truss beam

For the 65-bar truss beam, UCS and CC converged to a solution with a similar weight, as shown in Table 10. However, UCS required more time and FEM evaluations because of the need of an ungrouped calculation. Besides that, the certainty of NG was low, as shown in Figure 46, which indirectly affects the certainty of UCS as well. Similarly, CC had a high chance of converging to a high weight. On the other hand, both $FSCS_2$ and $FSCS_3$ provided a decent approach of low-weight solution with less FEM evaluations and higher certainty. The effect of the dominant global displacement constraint did not seem to cause issues for these methods, and the optimum of $FSCS_2$ had a slightly higher weight than $FSCS_3$, but had higher certainty. Furthermore, the additional time and number of FEM calculations required for $FSCS_2$ and $FSCS_3$ was small but not neglectable to the optimisation itself. The solution of ROT, AF and AF+S converged to higher weights than the other methods. Of these methods, AF_1 performed best and $AF+S_1$ worst. The computational time of these methods was neglectable. Furthermore, for AF_4, the optimum solution led to three groups in the final solution.

Table 10 – Results 65-bar truss

Method	Weight of optimum solution (kg)	Dominant constraint in optimum solution	Number of groups in optimum solution (-)	Number of times optimum found (%)	Mean time optimisation (s)	Mean time additional calculation (s)	Mean number of FEM evaluations (-)	Mean number of iterations (-)
NG	1270	Displacement	20	0.40	160		156024	
ROT	1577	Displacement	4	80	12		5021	
AF_1	1429	Displacement	4	77	14	0.88	6743	
AF_2	1445	Displacement	4	100	16	0.81	8017	
AF_3	1568	Displacement	4	40	17	0.82	9025	
AF_4	1491	Displacement	3	100	16	0.82	9718	
$AF+S_1$	1658	Displacement	4	100	18	0.93	8995	
$AF+S_2$	1621	Displacement	4	100	16	0.87	8281	
$FSCS_2$	1395	Displacement	4	80	21	2.6	9803	1.2
$FSCS_3$	1381	Displacement	4	4.8	20	8.1	7365	1.3
CC	1375	Displacement	4	0.52	80		96560	
UCS	1374	Displacement	4	45	170	1.6	159015	

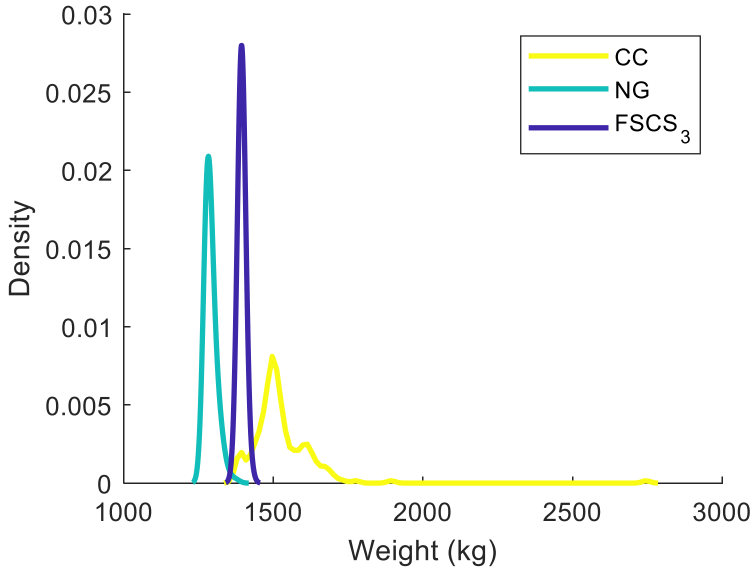
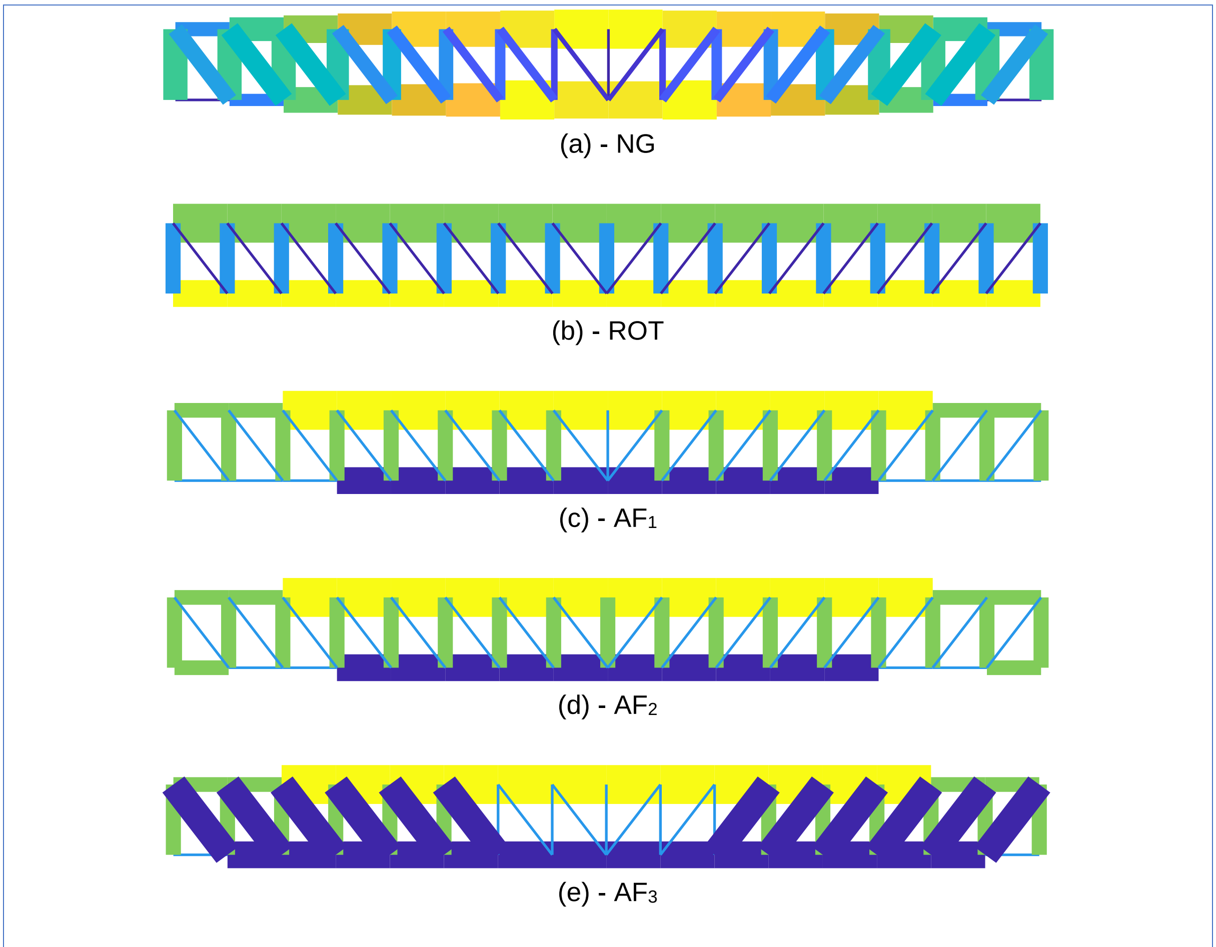


Figure 46 - Estimated density function optimum weight 65-bar truss

As presented in Figure 47, the results of $FSCS_2$, $FSCS_3$, CC and UCS gave an efficient grouping, combining compressive members in buckling and tensile member without buckling effects in one group. On the other hand, the AF and AF+S methods showed a separate grouping of compressive and tensile members, except for AF_4. But again, the resulting weight of AF_4 was high.



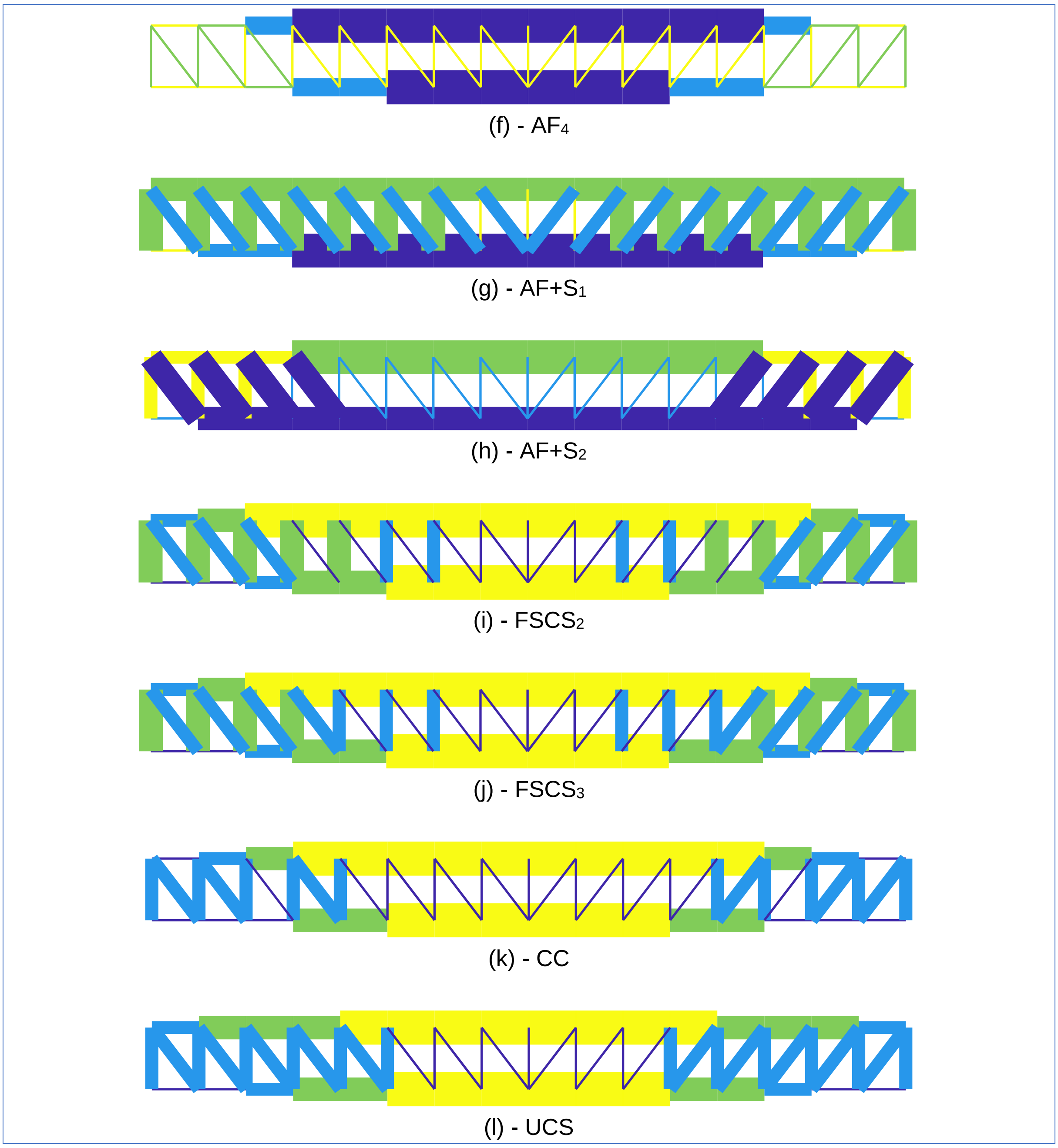


Figure 47 – Results groups and weight per unit length 65-bar problem

5.4.3 72-bar truss tower

For the 72-bar truss tower, of which the results are shown in Table 11, CC, $FSCS_2$ and $FSCS_3$ converged to the same solution. $FSCS_2$ did that in the least time and number of FEM evaluations. The additional calculations in FSCS required a non-neglectable but low amount of time and number of FEM evaluations. ROT, AF and AF+S converged to a structure with a high weight. Of these methods, $AF+S_2$ performed best and AF_4 worst. However, ROT, AF_1, AF_3, AF_4 and $AF+S_2$ converged to less than the desired number of groups.

Table 11 – Results 72-bar truss

Method	Weight of optimum solution (kg)	Dominant constraint in optimum solution	Number of groups in optimum solution (-)	Number of times optimum found (%)	Mean time optimisation (s)	Mean time additional calculation (s)	Mean number of FEM evaluations (-)	Mean number of iterations (-)
NG	269	Strength	8	38	81		112053	
ROT	401	Strength	3	100	10		3414	
AF_1	381	Strength	3	100	15	1.7	5330	
AF_2	401	Strength	4	100	14	1.6	5938	
AF_3	381	Strength	3	100	14	1.8	4912	
AF_4	413	Strength	2	100	17	1.7	5824	
$AF+S_1$	371	Strength	4	100	8.4	0.70	5596	
$AF+S_2$	315	Strength	3	100	9	0.78	5216	
$FSCS_2$	286	Strength	4	100	8.5	1.4	3391	1
$FSCS_3$	286	Strength	4	79	17	4.6	6271	1.6
CC	286	Strength	4	100	150		156322	
UCS	301	Strength	4	100	97	1.1	117496	

The grouped optimum of $FSCS_2$ and $FSCS_3$ and UCS, shown in Figure 48, revealed an unexpected grouping of bracings to resist the horizontal force; the bracings at the bottom had a low capacity while bracings at the top had a high capacity. Apparently, this allowed an efficient force redistribution which would probably not be thought of by an engineer: an engineer might the same capacity at the bottom as at the top, as the horizontal force applied at the top-level looks similar to an upright cantilever truss.

Furthermore, this benchmark problem showed how the force distribution of this statically indeterminate structure prevented UCS in finding the lightest grouped design; some members of the grouped optimum of FSCS and CC were lighter than the NG design, while UCS was only able to increase the weight of members of the NG design.

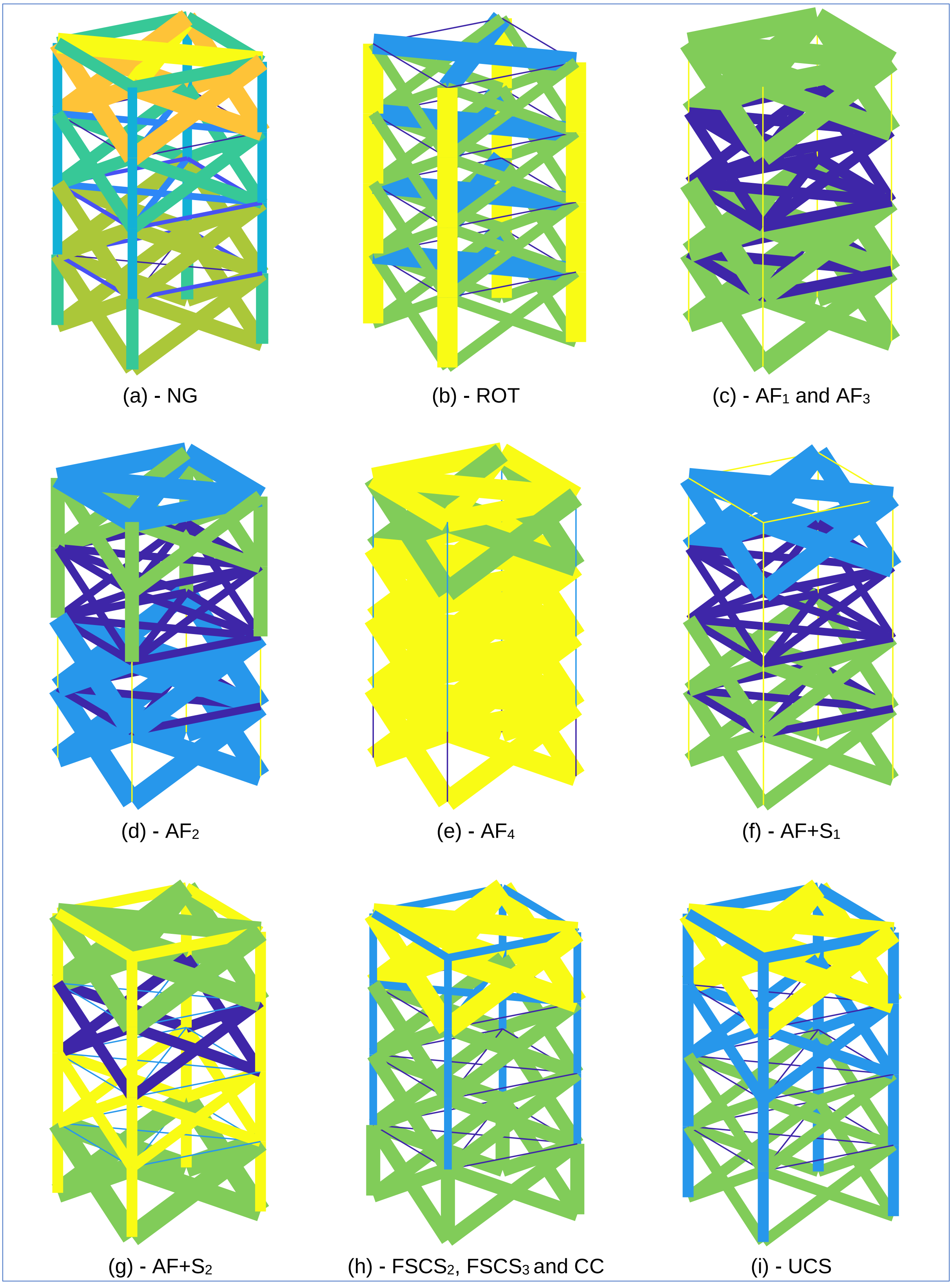


Figure 48 – Results groups and weight per unit length 72-bar problem

5.4.4 112-bar truss dome

For the 122-bar truss dome, of which the results are shown in Table 12, CC performed best. However, the distribution of the result, shown in Figure 49, indicated that this method found many solutions comparable to the results of $FSCS_3$. Furthermore, $FSCS_2$ converged to a slightly higher weight, but with high certainty and low computational effort. Again, the computational effort of the additional calculations was not neglectable for $FSCS_2$ and $FSCS_3$, but low enough to outperform other methods. As in the 65-bar truss beam, both FSCS methods found a good solution, although the displacement constraint was dominant. ROT, AF and AF+S converged to a structure with a higher weight. Of these methods, AF_4 performed best and ROT worst. Moreover, ROT and AF_3 converged to two instead of three groups.

Table 12 – Results 112-bar truss

Method	Weight of optimum solution (kg)	Dominant constraint in optimum solution	Number of groups in optimum solution (-)	Number of times optimum found (%)	Mean time optimisation (s)	Mean time additional calculation (s)	Mean number of FEM evaluations (-)	Mean number of iterations (-)
NG	2163	Displacement	11	0.50	190		62266	
ROT	3595	Displacement	2	50	10		1692	
AF_1	2587	Displacement	3	50	19	2.7	2887	
AF_2	2713	Displacement	3	75	15	3.4	3036	
AF_3	3592	Displacement	2	100	15	2.9	3064	
AF_4	2580	Displacement	3	100	14	2.8	2183	
$AF+S_1$	2673	Displacement	3	80	14	0.73	3030	
$AF+S_2$	3536	Displacement	3	100	14	0.69	3264	
$FSCS_2$	2457	Displacement	3	60	22	4.7	4206	2.1
$FSCS_3$	2413	Displacement	3	1.9	23	4.7	3737	1.9
CC	2310	Displacement	3	1.0	60		45974	
UCS	2501	Displacement	3	78	210	2.2	63996	

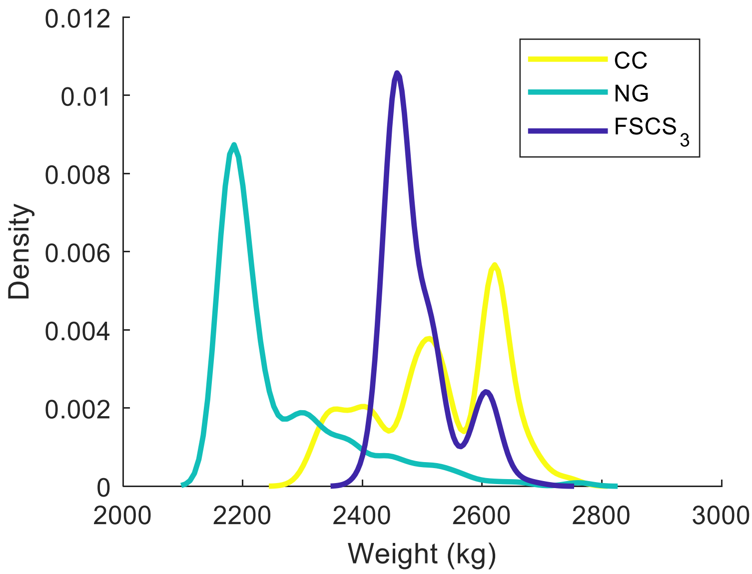
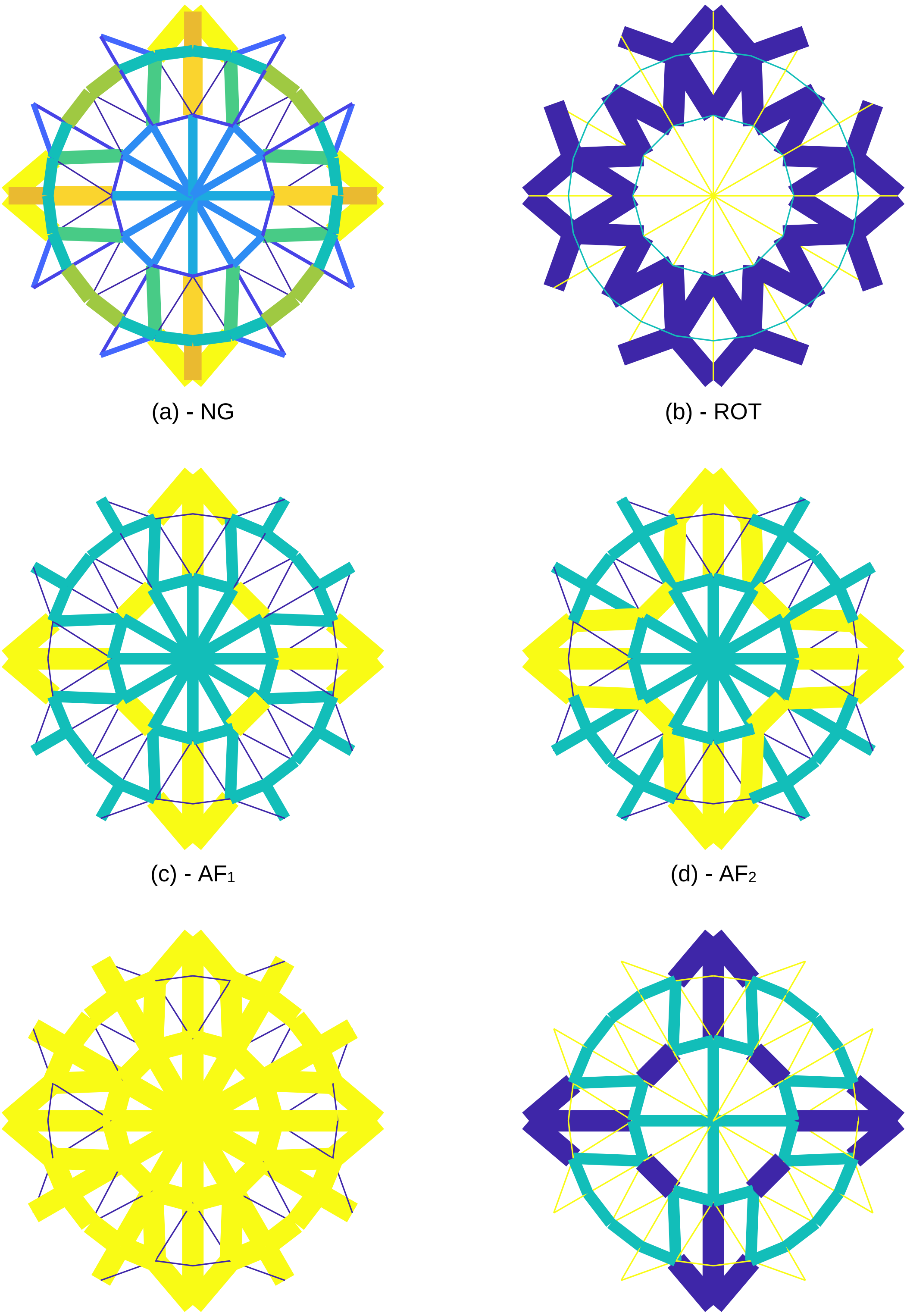


Figure 49 - Estimated density function optimum weight 112-bar truss

The subfigures in Figure 50 show a top view of the 3D-structure. The designs shows an unexpected force distribution. An engineer might expect that the vertical load is diverted in a radial direction to the supports, in which the diagonals carry a small part of the load. Hoop forces might be expected to be taken by the circular beams. However, their relative influence differed in the results: for $FSCS_2$ and ROT the circular beams were designed relatively light, while in other methods these beams showed an increased relative weight.



(a) - NG

(b) - ROT

(c) - AF_1

(d) - AF_2

(e) - AF_3

(f) - AF_4

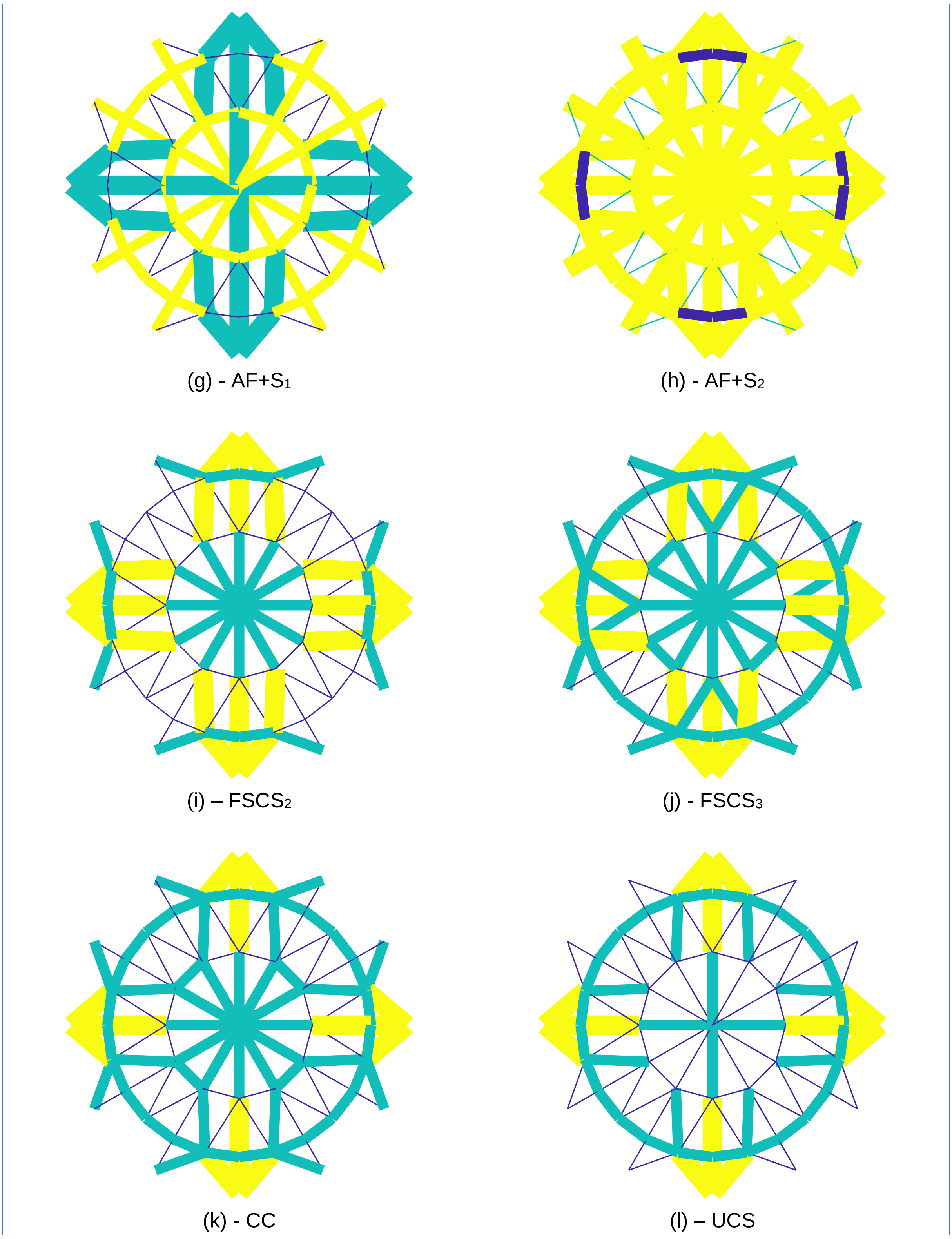


Figure 50 – Results groups and weight per unit length 112-bar problem

5.4.5 160-bar truss tower

Although the dominant constraint for most methods was the displacement in the 160-bar truss tower, $FSCS_3$ was able to converge to the lowest weight, as shown in Table 13. CC became second, but $FSCS_2$ converged to only a slightly higher weight with lower computational effort and with more certainty, observable in Figure 51. UCS found a slightly higher weight but required much more computational effort. ROT, AF and AF+S converged to a structure with a higher weight. Of these methods, AF_3 performed best and ROT worst. All of these methods, except for $AF+S_2$, converged to a solution with less than the desired six groups.

Table 13 – Results 160-bar truss

Method	**Weight of optimum solution (kg)**	**Dominant constraint in optimum solution**	**Number of groups in optimum solution (-)**	**Number of times optimum found (%)**	**Mean time optimisation (s)**	**Mean time additional calculation (s)**	**Mean number of FEM evaluations (-)**	**Mean number of iterations (-)**
NG	871	Displacement	14	0.58	940		1767347	
ROT	1015	Displacement	4	86	33		37411	
AF_1	982	Displacement	5	43	49	4.3	56359	
AF_2	982	Displacement	5	55	43	4.4	49218	
AF_3	943	Displacement	5	83	42	4.4	47469	
AF_4	988	Displacement	4	100	37	4.2	38016	
$AF+S_1$	1164	Displacement	5	60	28	1.2	26432	
$AF+S_2$	975	Displacement	6	67	34	1.3	33069	
$FSCS_2$	883	Strength	6	3.4	310	56	367705	5.6
$FSCS_3$	878	Displacement	6	0.62	260	54	238505	3.8
CC	881	Displacement	6	0.37	720		1253812	
UCS	893	Strength	6	50	980	4.8	1813070	

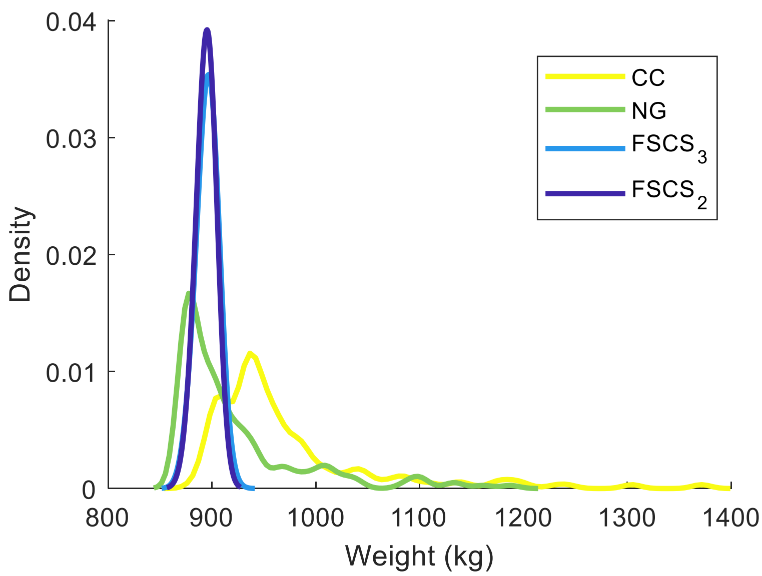
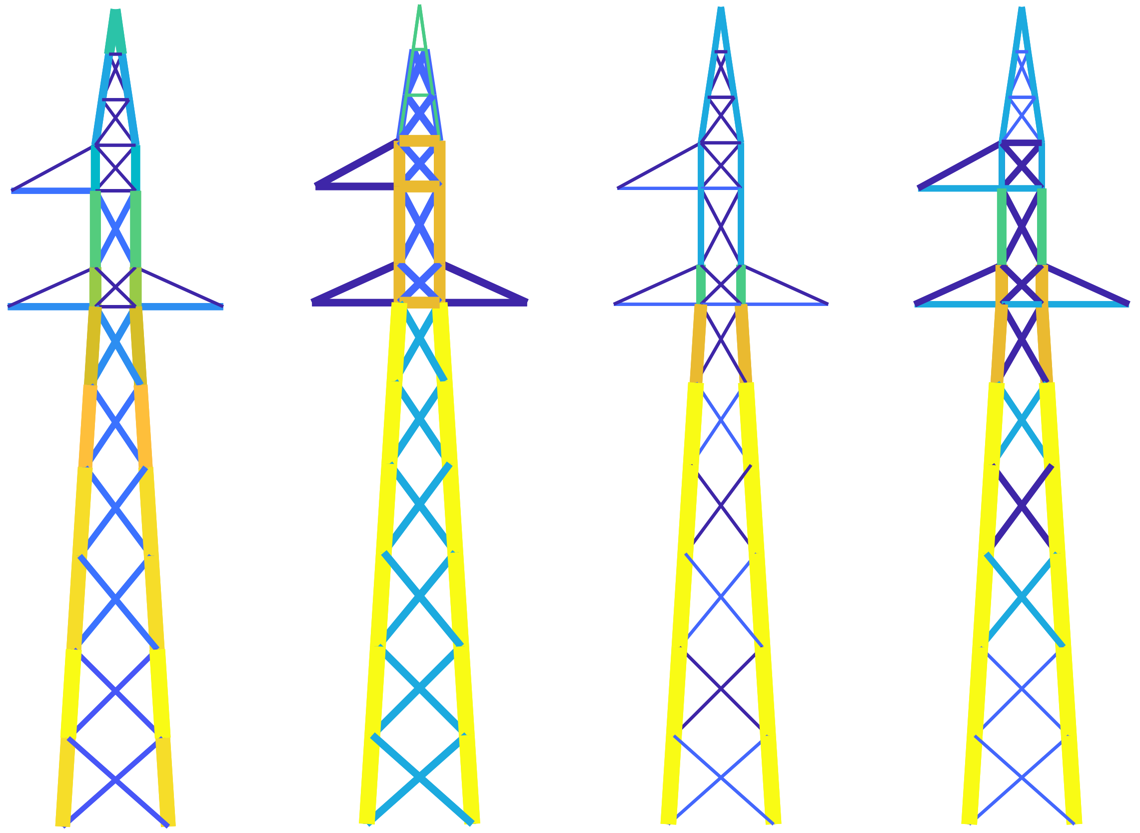


Figure 51 - Estimated density function optimum weight 160-bar truss

Figure 52 shows a side-view of the 3D-structure. The grouping of $FSCS_2$ and $FSCS_3$, CC and UCS indicated an efficient grouping of bracings; where forces were introduced from the cables, bracings showed a higher capacity. As in the 72-bar truss tower, bracings were dimensioned lighter near the base than higher in the structure, disagreeing with general structural intuition.



(a) - NG (b) - ROT (c) - AF_1 and AF_2 (d) - AF_3

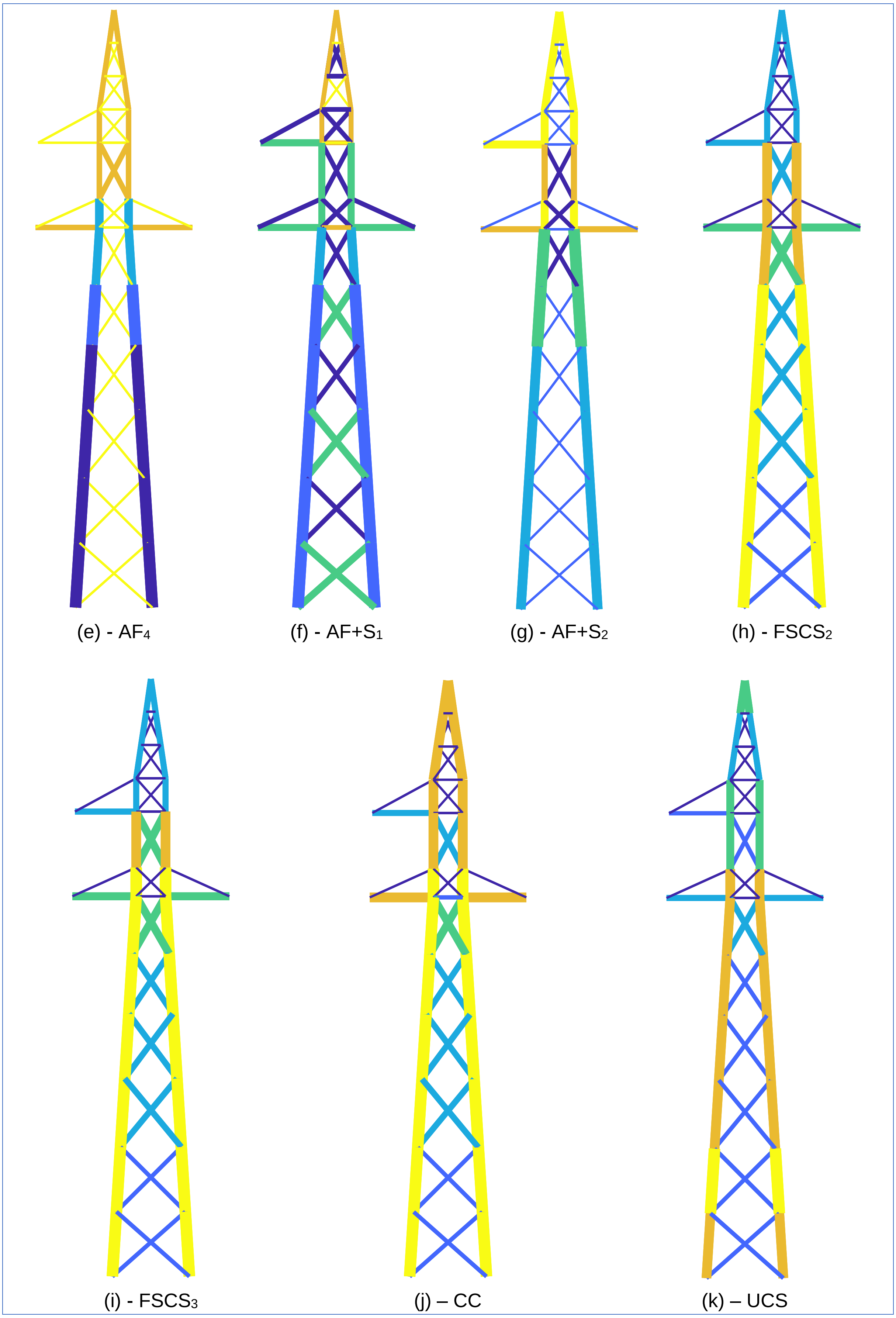


Figure 52 – Results groups and weight per unit length 160-bar problem

5.4.6 15-bar 3-storey frame

The results of the 15-bar 3-storey frame in Table 14 showed that the CC performed best with low computational effort. $FSCS_2$ and $FSCS_3$ found a slightly heavier solution with more certainty and slightly lower computational effort. UCS found a suboptimal solution and required more computational effort. ROT gave the worst solution and all methods converged to the desired three groups.

Table 14 – Results 15-bar 3-storey frame

Method	Weight of optimum solution (kg)	Dominant constraint in optimum solution	Number of groups in optimum solution (-)	Number of times optimum found (%)	Mean time optimisation (s)	Mean time additional calculation (s)	Mean number of FEM evaluations (-)	Mean number of iterations (-)
NG	7699	Strength	8	14	36		28948	
ROT	8124	Strength	3	100	8.4		3122	
$FSCS_2$	8081	Strength	3	100	18	1.9	5998	2
$FSCS_3$	8053	Strength	3	37	25	3.3	5037	1.6
CC	8014	Strength	3	21	27		12843	
UCS	8093	Strength	3	57	47	0.24	32230	

The distribution of steel in the structure by CC, as shown in Figure 53, gave a similar megabrace structural configuration as reported by Walls and Elvin (Walls and Elvin 2010b). Moreover, the ungrouped optimum suggested a grouped design like UCS in which on a higher storey-level the outer columns had a high capacity. However, the results of CC and FSCS showed that choosing another grouping allows for another force distribution with a lower weight.

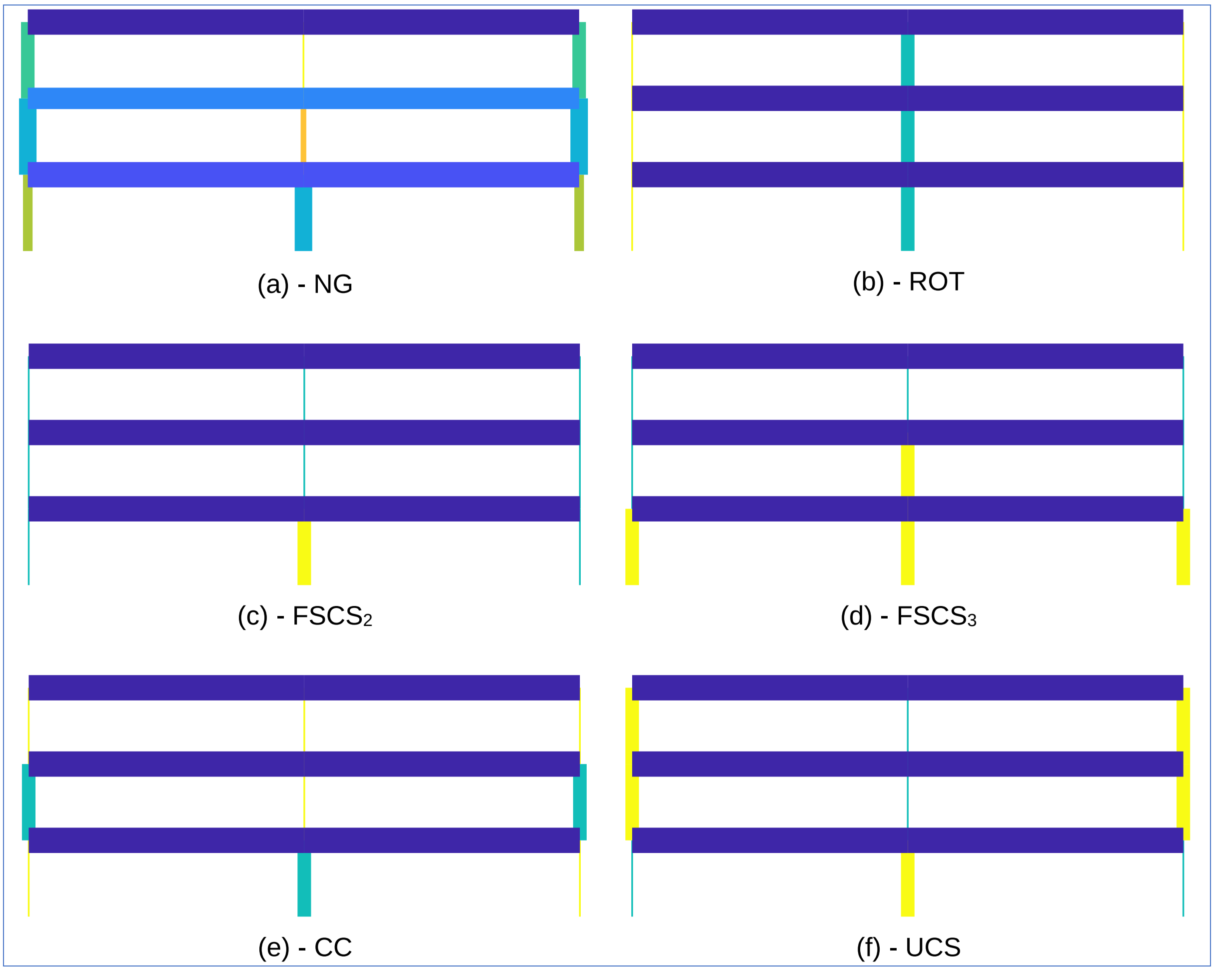


Figure 53 – Results groups and weight per unit length 15-bar 3-storey frame problem

5.4.7 117-bar 9-storey frame

$FSCS_2$ and $FSCS_3$ found the same and the lightest solution for the 117-bar 9-storey frame, as shown in Table 15. These two methods required a low computational effort. CC converged to a higher weight, with higher computational effort. Besides that, CC found a wide range of weights, as shown in Figure 54. UCS found the seconds best solution in term of weight, but required long computational time and many FEM evaluations. Again, ROT found the worst solution with six instead of seven groups in the final solution.

Table 15 – Results 117-bar 9-storey frame

Method	**Weight of optimum solution (kg)**	**Dominant constraint in optimum solution**	**Number of groups in optimum solution (-)**	**Number of times optimum found (%)**	**Mean time optimisation (s)**	**Mean time additional calculation (s)**	**Mean number of FEM evaluations (-)**	**Mean number of iterations (-)**
NG	22657	Strength	19	0.98	930		182181	
ROT	26190	Strength	6	100	45	0.40	7251	
$FSCS_2$	24779	Strength	7	100	120	14	21732	3.0
$FSCS_3$	24779	Strength	7	91	120	13	21533	2.8
CC	25004	Strength	7	0.72	550		114166	
UCS	24793	Strength	7	100	980	1.4	189099	

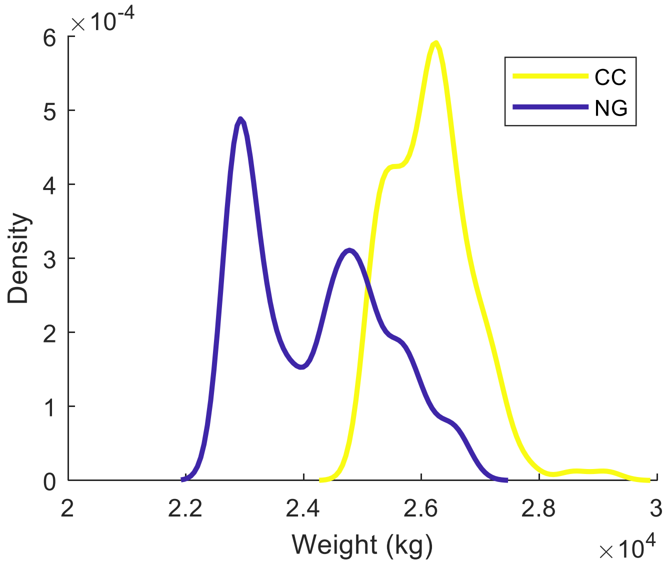
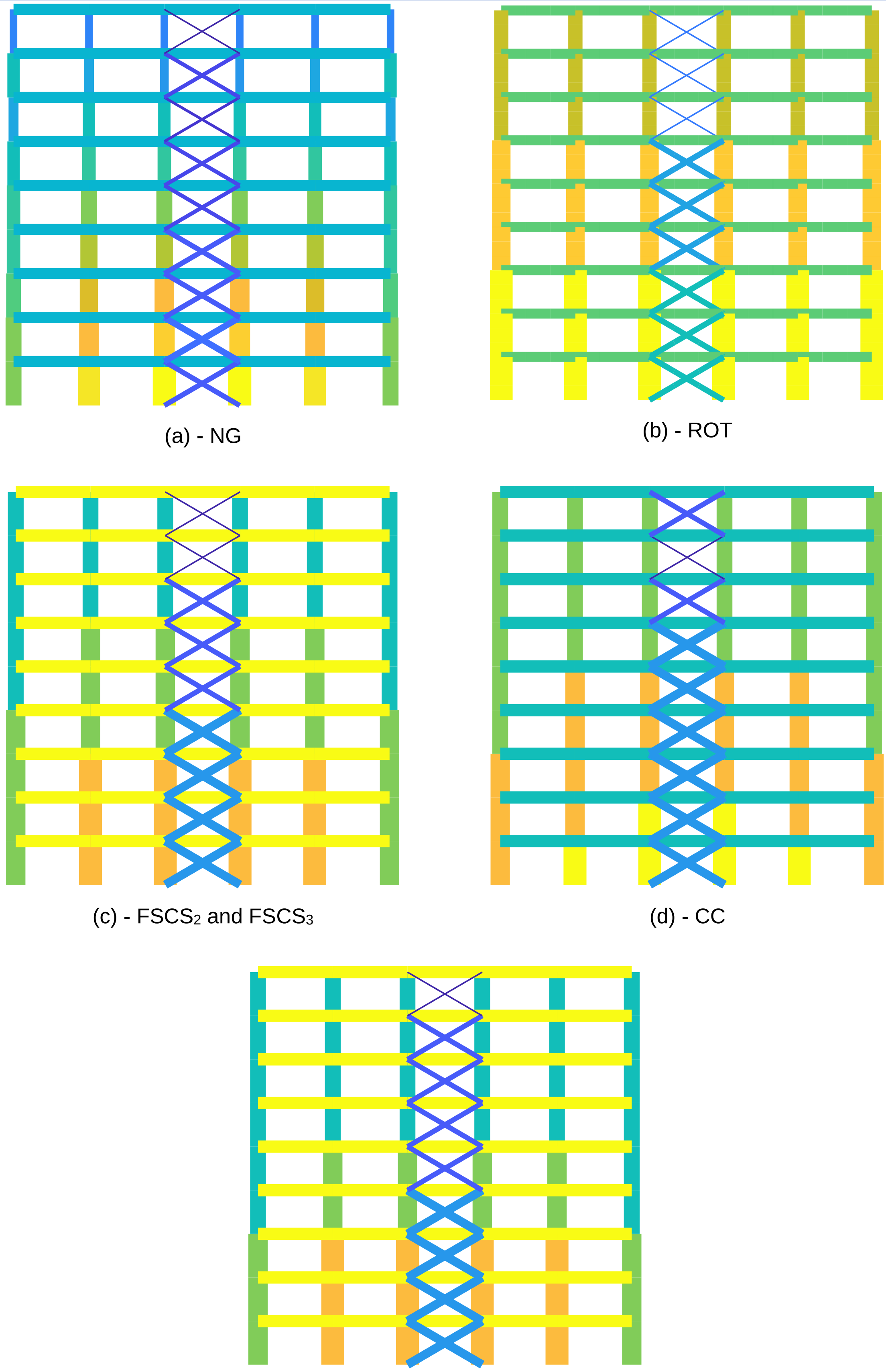


Figure 54 - Estimated density function optimum weight 117-bar 9-storey frame

The solutions of the grouped optima of $FSCS_2$, $FSCS_3$, CC and UCS in Figure 55 showed that columns are logically grouped in different column rows. It showed that the side columns on a low storey level were loaded less than middle columns at the same storey-level. This caused the low storey level columns on the side to be grouped with high storey level columns in the middle of the structure.



(a) - NG

(b) - ROT

(c) - FSCS$_2$ and FSCS$_3$

(d) - CC

(e) – UCS

Figure 55 – Results groups and weight per unit length 117-bar 9-storey frame problem

5.4.8 147-bar 3-storey frame

Table 16 shows that CC converged to the lightest solution for the 147-bar 3-storey frame. On the other hand, the uncertainty of reaching this solution was high, as shown in Figure 56, and a high computational effort was required. In contrast, the computational effort of both FSCS methods was much lower than CC, and the full range of possible solution had a low weight. UCS found a solution with high certainty during its second step of optimisation, but low certainty for NG, which required long computational time and many FEM evaluations as well. As in the other frame problems, ROT found the worst solution. UCS found a solution with five instead of six groups.

Table 16 – Results 147-bar 3-storey frame

Method	Weight of optimum solution (kg)	Dominant constraint in optimum solution	Number of groups in optimum solution (-)	Number of times optimum found (%)	Mean time optimisation (s)	Mean time additional calculation (s)	Mean number of FEM evaluations (-)	Mean number of iterations (-)
NG	24464	Strength	13	0.96	1900		203038	
ROT	30380	Strength	6	67	39		4076	
$FSCS_2$	26020	Strength	6	1.3	390	46	38911	6.0
$FSCS_3$	26066	Strength	6	2.5	530	30	33615	4.8
CC	25656	Strength	6	1	1200		122616	
UCS	26976	Strength	5	100	2000	4	207193	

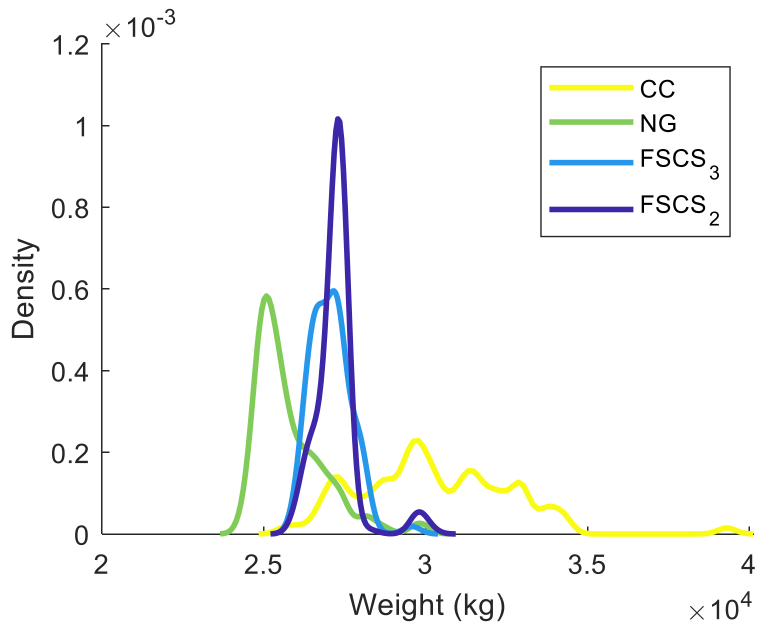


Figure 56 - Estimated density function optimum weight 147-bar 3-storey frame

The solutions of $FSCS_2$, $FSCS_3$, CC and UCS in Figure 57 showed how these grouping methods were able to group columns, bracings and beams in different parts of the structure, in a way that probably could not be thought of by an engineer. For example, the solution of CC grouped bracings on the lowest level with beams on the lowest level and middle level in one direction, and beams in both directions on the top level.

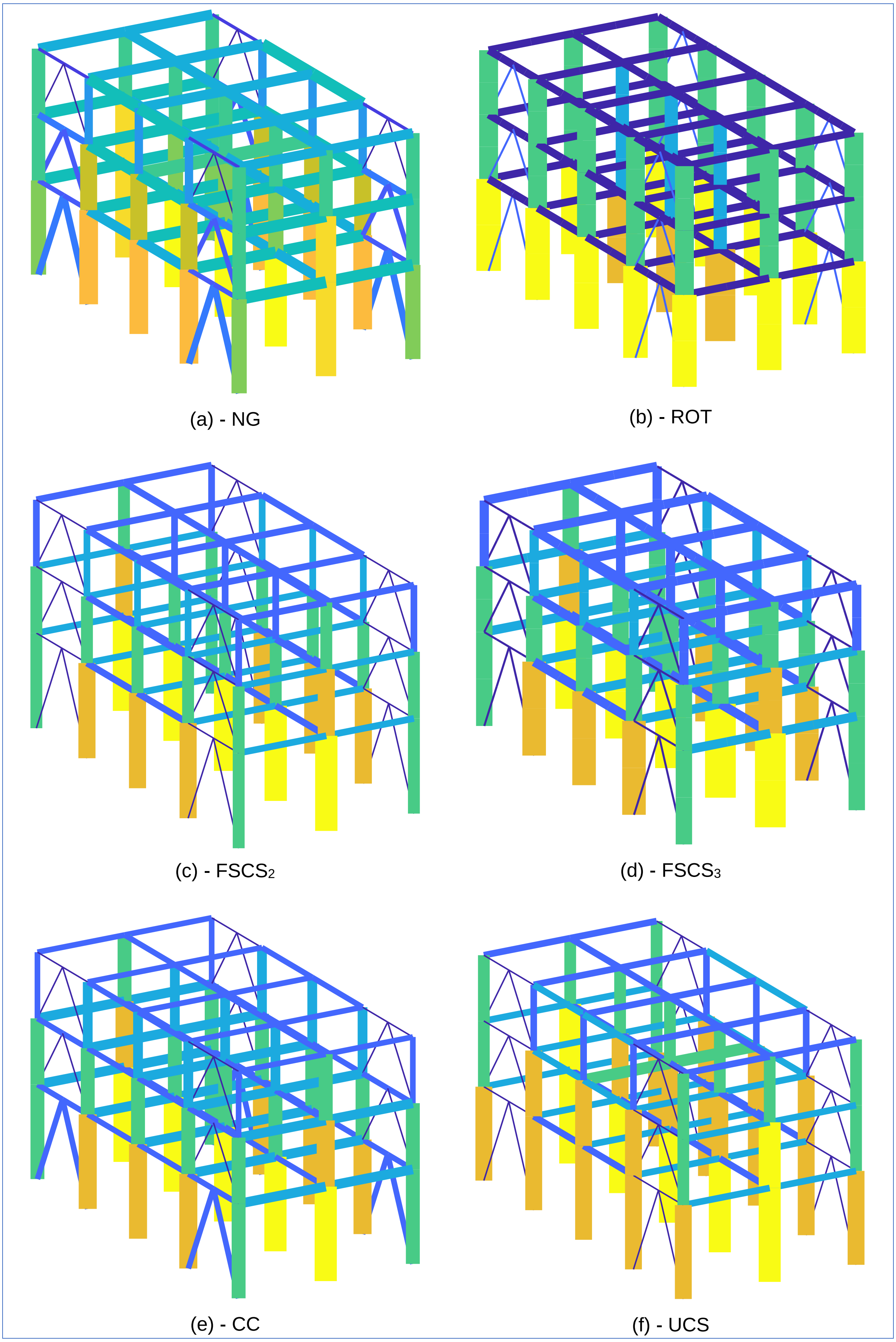


Figure 57 – Results groups and weight per unit length 147-bar 3-storey frame problem

5.5 Conclusion

This chapter has numerically investigated a selection of existing grouping methods and the potential for a new grouping method, thereby answering subquestion three and four.

3. How do the grouping methods perform on benchmark problems for minimum weight?

From the numerical experiments it is concluded that grouping methods can group members in different part of the structure to their optimal profile. This allows grouping of members to a similar profile in a more efficient way than an engineer might be able to do. Furthermore, in more complex structure, grouping methods can efficiently group beams, columns and bracings. In addition, it generates low diversity solutions with a force distribution which could possibly not be thought of by an engineer.

From the numerical experiments, conclusions can be drawn on the relative performance of the grouping methods. The relative performance of all methods is shown in Figure 58. it shows the ability to find the lowest-weight grouped optimum, computational ease and certainty of finding a good solution. This figure conceptually summarises the numerical results from all experiments. If a bar is filled completely, its performance is good and vice versa. Furthermore, the green bars represent acceptable, yellow questionable and red poor performance. The perfect method has all bars filled completely in green. For the methods UCS, $FSCS_2$ and $FSCS_3$, the properties of the search space are shown for each step or iteration. For FSCS, the mean number of iterations is shown. The performance of the six procedures of AF and AF+S is evaluated jointly.

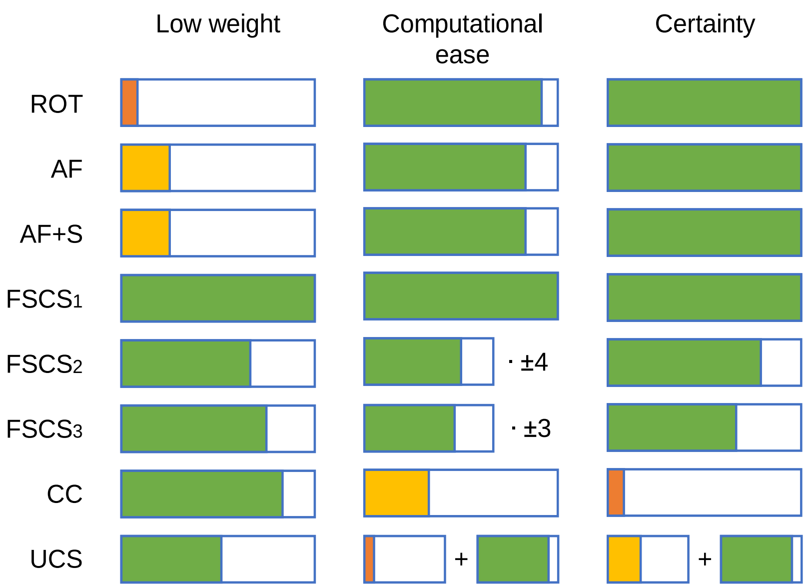


Figure 58 – Overview comparison methods numerical experiments

It can be concluded that the grouping method ROT performs worst of all methods, as it results in high-weight solutions and easily converges to solutions with less than the desired number of groups in the experiments. However, it does not require any additional calculations, and because of its small and simple search space the certainty was found to be high. Nonetheless, the ability to find a light grouped solution is regarded as more important.

Furthermore, AF and AF+S perform slightly better than ROT. These methods prove to find lighter solutions than ROT and the computational effort of its additional calculations is neglectable. However, as ROT, these methods can easily converge to less than the desired number of groups. Furthermore, these methods are only applicable to truss structures and the inability of properly grouping tensile and compressive members leads to suboptimal solutions. For the different procedures possible on AF and AF+S, none can be identified as the best.

It should be noted that the convergence to less than the desired number of groups is not necessarily bad; the optimisation finds a local optimum which happens to have less groups than desired. If an engineer does not want to find the optimum for that specific number of groups, this local optimum might be satisfactory.

From the numerical experiments it can also be concluded that, with higher computational effort than the former methods, lighter results can be expected with FSCS, CC and UCS. Of these methods, CC was able to identify the lightest solution in five of eight benchmark problems. However, it requires high computational effort and the certainty is low for complex problems.

UCS gave the lightest solution in two of eight problems in the numerical experiments. However, it requires high computational effort because of the need for an ungrouped optimisation. Besides that, as the certainty is low for an ungrouped optimisation, the certainty of the UCS method decreases indirectly too; although the certainty of the second step of UCS itself is in general high.

Finally, FSCS found the lightest solution in four out of eight benchmark problems. For two problems it had a high certainty of finding the best solution. For the other two benchmark problems it performed second best. $FSCS_1$ undoubtedly finds the grouped optimum for statically determinate structures without displacement constraint with minimal computation effort. $FSCS_2$ can find a proper solution for other problem with high certainty, and the approach to handle dominant global displacement constraints of this method seems good enough to find light-weight results. Conclusively, this method requires more computational efforts than ROT, AF and AF+S, but much less than CC and UCS. $FSCS_3$ can find a slightly better solution than $FSCS_2$, but has higher uncertainty. Therefore, I regard the added value of $FSCS_3$ as marginal. For both $FSCS_2$ and $FSCS_3$, the number of iterations is limited, keeping the total computational effort low.

---

4. Do the currently available grouping methods perform well enough for application in research and practice? If not, can a new method be developed, outperforming current methods?

For application in research, a grouping method should be able to theoretically find the true global grouped optimum, and many trials may be allowed to find it. CC is best method to do so. However, the optimisation includes a complex search space, which demands a high computational effort to find a light design. Furthermore, the complexity of the search space might prevent the optimisation from finding the global grouped optimum.

To apply grouping methods in practice, a grouping method should be able to find the global or a near-global grouped optimum with low computational effort and high certainty. The new FSCS method shows to do this, as it can find similar low-weight as CC, requires few computations and shows high certainty.

6 Application in practice

In this chapter, the influence of grouping methods on the minimum cost design is investigated to answer the final subquestion:

5. How do grouping methods improve the minimum cost design of a real-life case-study?

This question is answered on multiple aspects. First, the basics of cost optimisation and cost functions are considered. Subsequently, the procedure of explicit cost optimisation is shown in an example, in which the possible difference of cost and weight optimisation is illustrated. Finally, explicit cost and weight optimisation is applied to a real-life case-study of Royal HaskoningDHV to show the potential cost and weight savings of optimisation with grouping methods.

6.1 Cost models of steel structures

To model the cost of a structure in an optimisation procedure, all relevant costs have to be written in a cost function. This cost function can be inserted as the objective function in the standard design optimisation model. However, cost functions do not solely depend on the structural design and its weight, but other factors change the costs during the design, fabrication, use, and demolition or reuse of a structure, like:

- Raw material costs. These costs are closely linked to the weight of the structure. However, the costs of specific profiles and length might not scale linearly to the weight of those profiles.
- Production costs profiles, including the costs for cutting, painting and welding
- Bulk discount on high quantities of single profiles.
- Costs of connections, as each connection requires engineering work and material
- Foundation costs, which is a function of reaction forces at the supports.
- Costs of fabrication at height. Construction at height may require safety measure and other equipment, especially for big and heavy members.
- Costs of quality checking, which depends on the complexity of the structure.
- Engineering costs, which is a function of the complexity of the structure and design methods used. An optimisation process itself introduces costs as well, like computation power, licenses and working hours of engineers.
- Costs following from scheduling of the project and availability of partners
- Sustainability costs. The introduction of CO_2 taxes increases costs, while reusing and recycling of parts of the structure reduce costs
- Unforeseeable events. The coronavirus is a clear example which influences economy and prices of building projects.

As a result of the variety and unpredictability of many cost aspects, it is hard to quantify the costs of a design (Tizani et al. 1996).

In this study the cost function proposed by Watson et al. (1996) was chosen because it provides an estimate for costs in four aspect of construction and as it is applicable to the global design. This cost function separates costs in steel supply, fabrication, surface treatment and erection costs. These costs are expressed in unit prices and manhours. The steel supply costs are linearly related to the weight of the structure. An overview of this cost function is shown in Figure 59 and the details of this cost function are listed in Appendix G.

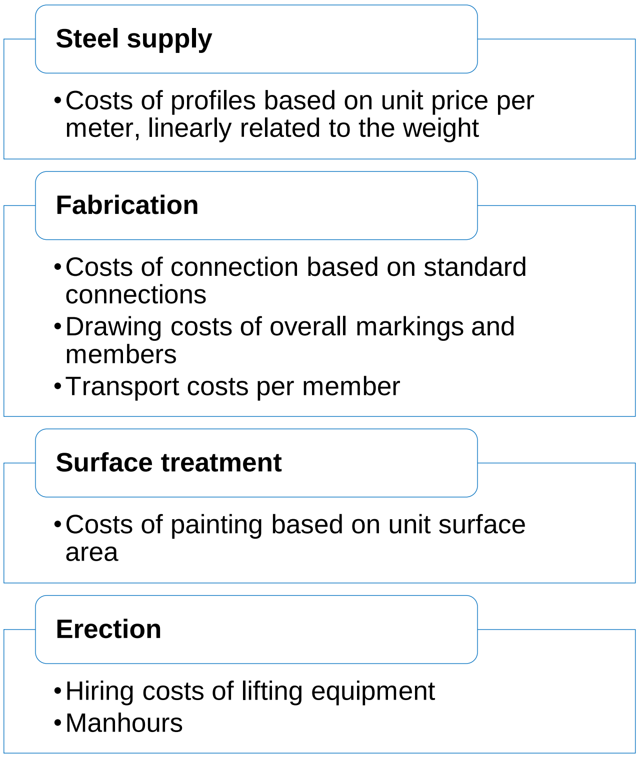


Figure 59 – Components cost function Watson et al. (1996)

6.2 Difference cost and weight optimisation

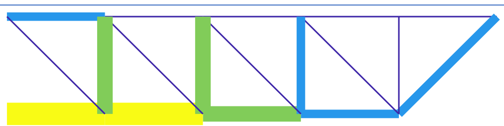
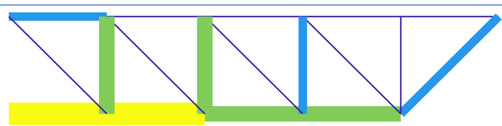
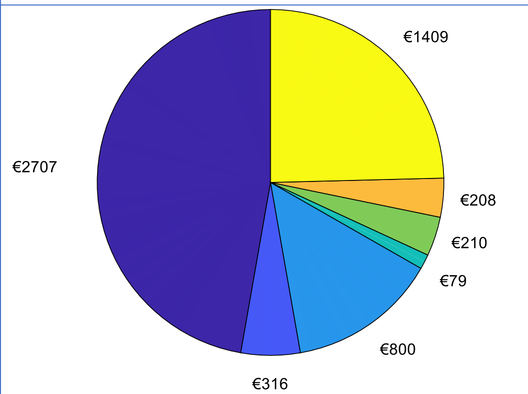
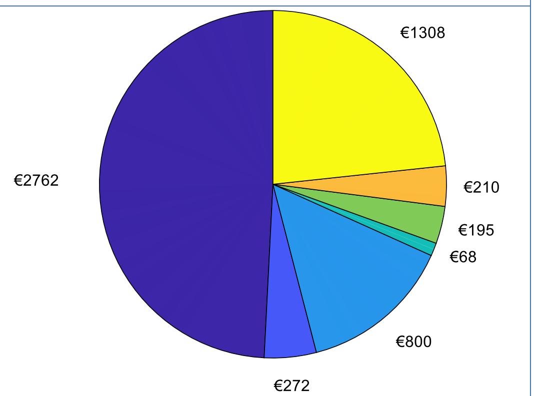
The optimum weight structure might not be the same as the optimum cost structure. The difference for the 18-bar truss structure was investigated in this research project.

Difference for four groups

For the 18-bar structure, both the optimum weight and costs of the optimal weight and optimal cost structure were evaluated. This was done for four groups, using $FSCS_1$ as grouping method and the cost function of Watson et al. (1996).

The results are shown in Table 17, which present the weight, costs, grouping of the beams and costs per category for both the weight and cost optimum. The weight optimum was found to be more expensive than the cost optimum. For the cost optimum, the weight was higher, therefore increasing the supply and surface treatment costs as well. However, fewer beam splices and beams were needed as in the bottom truss only one beam splice was needed in the cost optimum. The reduced number of beam splices reduced both erection, member drawing and transport costs, and costs for the fabrication of connections.

Table 17 – Comparison weight and cost optimisation 18-bar cantilever truss for 4 groups. Weight optimum is lighter but has higher costs than cost optimum.

	Weight optimum 4 groups	Cost optimum 4 groups
Weight	2201 kg	2246 kg
Total costs	€ 5730	€ 5617
Structure		
Costs per category		
	Supply steel Connections Marking drawings Member drawings	Transport Surface treatment Erection

Optimum number of groups

The added value of grouping methods in determining the optimum number of groups was also investigated in this study for the 18-bar cantilever truss. Figure 60 shows the optimum weight and costs of this structure for different number of groups, plus the results of implicit cost optimisation. For explicit weight and cost optimisation, the optimum structure for each number of groups was calculated. On the other hand, for implicit cost optimisation 96 independent optimisations were performed. In these calculations, the number of groups was not fixed by a grouping method, but it was a result of the optimisation.

The explicit cost optimisation showed both a decrease and increase in costs. The resulting cost optimum had 10 groups with costs of €5345, respectively. The corresponding structure is shown in Figure 61a.

The explicit weight optimisation showed a decrease in cost for one to five groups, then an increase in cost until seven groups and again a decrease for more than seven groups. For one to three groups, the supply costs were dominant in the cost function, thus leading to the same optimum for weight and cost optimisation. For four or more groups, the weight optimum was more expensive than the cost optimum.

The implicit cost optimisation found many different results, ranging from 7 to 12 groups and from €5366 to €5864. The density function of the cost distribution, for all number of groups, is shown in Figure 62. The cost of the cheapest solution was €5366. This result was different in one beam with the explicit cost optimum, which is marked with a red circle in Figure 61b.

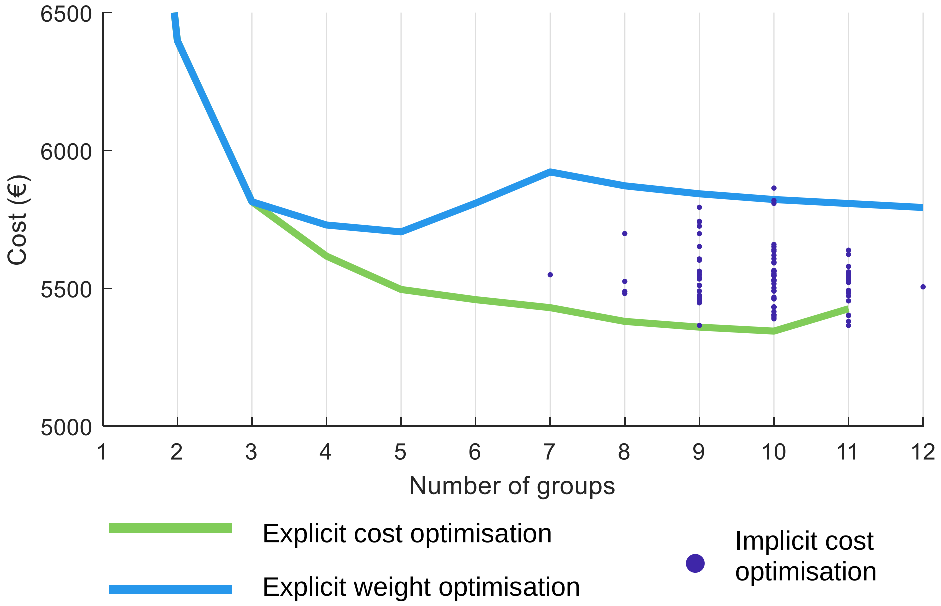


Figure 60 - Results analysis on optimum number of groups for 18-bar cantilever truss. Different optimum number of groups is found for cost and weight optimisation, and implicit cost optimisation finds many suboptimal results.

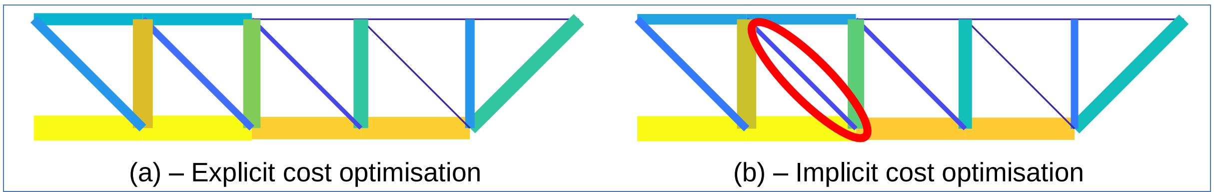


Figure 61 – Results groups and weight per unit length cost optimisation 18-bar cantilever truss. Designs are different in one bar, which is marked with a red circle.

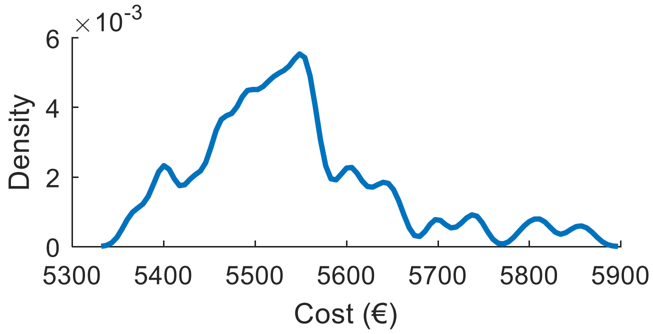


Figure 62 - Estimated density function optimum implicit cost optimisation 18-bar cantilever truss with a wide range of results.

Conclusively, cost and weight optimisation lead to different results if the supply costs are not governing in the cost function. Furthermore, explicit cost optimisation, which makes use of the new grouping method, can find the global cost optimum with higher certainty than implicit cost optimisation. This solution can be found with few computations, as multiple simple grouped problems are solved instead of solving the ungrouped problem directly.

Combinatorial search in cost optimisation

In a preliminary phase of the cost analysis, it was found the amount of beam splices in potential continuous beams is a major influencer of costs. This should be covered by a grouping method. Most grouping methods do not consider this influence. CC, AC and MO evaluate a search space in which all allowable number of beam splices are included, thus allow for finding the global cost optimum.

FSCS and UCS can include the influence of the number of beam splices by an expansion of the combinatorial search. This combinatorial search includes all combinations of grouping beam segments to all or some of the adjacent beam segments of a higher weight. As a result, this method is applicable to cost models which include the number of beam splices as well.

6.3 Real-life case-study

As a second example of cost optimisation using grouping methods, the roof of the Feyenoord stadium was taken as a case-study. Royal HaskoningDHV worked on the structural design for this new stadium (Kraaijenbrink et al. 2019). The design of the architect OMA is shown in Figure 63 and Figure 64 (OMA 2018). This chapter treats the original design by Royal HaskoningDHV, and the improved design using the new grouping method.

Figure 63 – Render new Feyenoord stadium from inside
Image courtesy OMA (2018)

Figure 64 – Render new Feyenoord Stadium in aerial view
Image courtesy OMA (2018)

The roof structure of the Feyenoord stadium consists of different substructures, shown with different colours in Figure 65. An additional movable roof structure is not shown in the picture. The dark blue truss beams are analysed in this structure, as the two identical beams together represents 63% of the total weight of the full roof structure.

The design consists of only statically determinate beams, which allows to optimise the substructures separately, following the load path. As this study focussed on optimising just the dark blue truss beams, the design of other substructures was copied from the final design of Royal HaskoningDHV (2019).

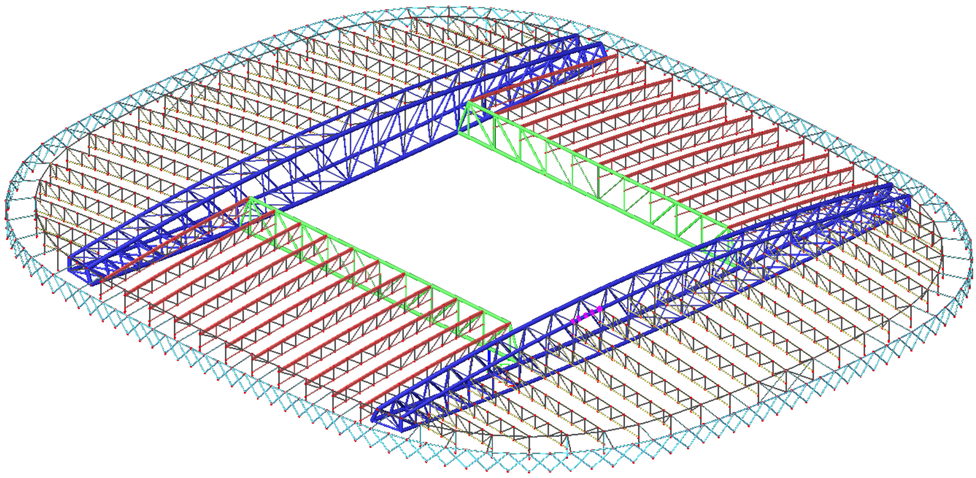


Figure 65 – Parts roof structure, dark blue truss beams were analysed in this study. Picture from report Royal HaskoningDHV (2019)

This design was originally designed by Royal HaskoningDHV with the Grasshopper plug-in Karamba3D and SCIA Engineer. The calculation of Karamba3D included an algorithm which automatically selects profiles (Tam 2020). A manual grouping was applied in this process. The result of Karamba3D was manually fine-tuned with respect to the Eurocode in SCIA.

For this case-study, this problem was slightly altered in terms of the available profile database, constraint function, load cases and geometry.

The profile database of the automatic profile selection algorithm in Karamba3D evaluated 665 German RO-profiles. In this study, a uniform selection of 50 profiles was made, taken from the entire weight range of the original profile database. A more optimal result might be found when the full database is included.

The final design of Royal HaskoningDHV was checked on the Eurocode requirements. Furthermore, in the unity check some margin was taken for connections in the detailed design, and future variations in the design. However, in this study, a simplified check was made on yield and buckling stress, and global displacement constraints. It was assumed that this simplified constraint function represented the constraints set by the Eurocode adequately.

Originally, the problem consisted of 29 load cases. The dominant five load cases for the curved truss beams were used in this study. However, the result of this study might violate constraints in one of the other original load cases.

Finally, the geometry of the structure was slightly changed, as the original structure included some bars which provided stability to elements of other parts of the roof structure. Furthermore, the green truss beams in Figure 65 had a torsional-limiting effect on the dark blue truss beams. To mimic this behaviour, spring supports were added.

All properties of the final structure are shown in Appendix F.9

6.3.1 Manual grouped design

The design with manual grouping from Royal HaskoningDHV divided all 270 beams into 21 groups. Optimisation to both cost and weight with this grouping, led to a weight of 820 ton and costs of €1.26 million in this study. This result is shown in Figure 66 and Figure 67.

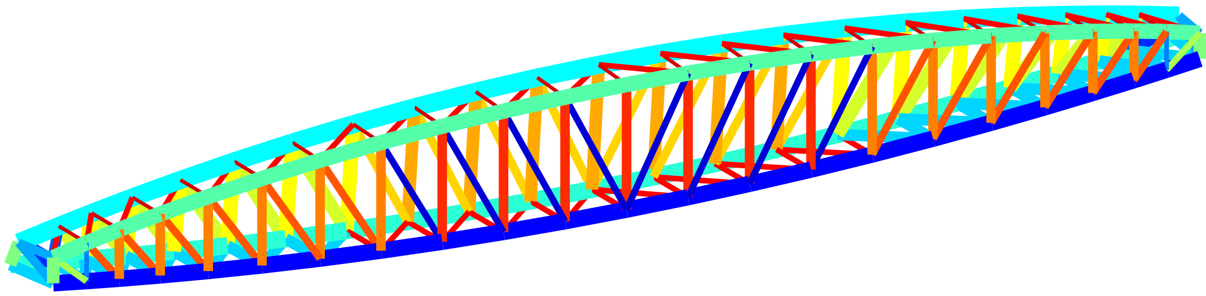


Figure 66 – Result Feyenoord stadium ROT

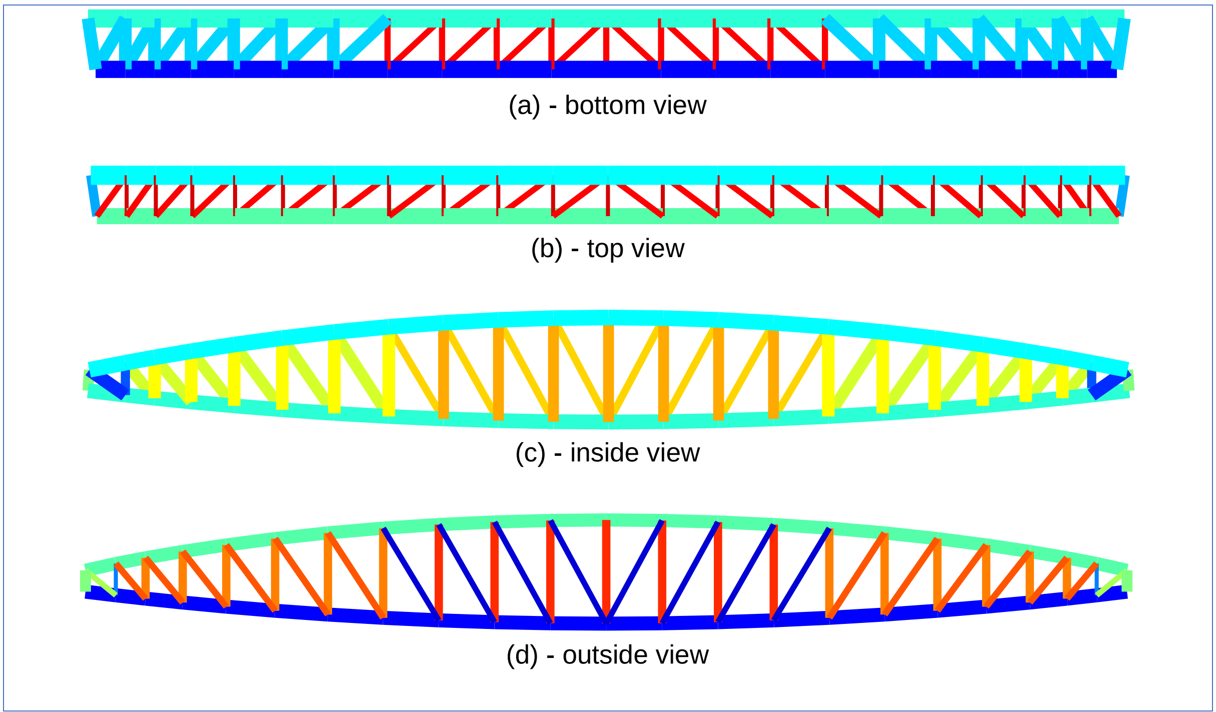


Figure 67 – Result Feyenoord stadium ROT side views

It should be noted that, as the supply cost were dominant in this design with 92% of the total costs, weight and cost optimisation gave the same solutions for this case-study.

Furthermore, this design was not equal to the original design of Royal HaskoningDHV; in the actual final design the weight was 944 ton and the costs were €1.46 million. From this actual final design of Royal HaskoningDHV, only the grouping of the original design was copied, not the choice of profiles. This difference was considerable, because of Royal HaskoningDHV did not use an optimisation method in finding these profiles and there were differences in modelling, as explained on the previous page.

6.3.2 Optimal grouped design

For the improved design, $FSCS_2$ was applied as a grouping method. It was applied for 1 to 24 groups, of which the results are shown in Figure 68. In these analyses, the grouped design with $FSCS_2$ gave a design with lower weight and costs than the manual grouping of Royal HaskoningDHV for 7 groups or more. In comparison with the original design of Royal HaskoningDHV, $FSCS_2$ gave an improved design for 4 groups and more. For 21 groups, as much as the manual grouping had, the weight and costs were decreased with 63 ton (7.8%) and €92.000 (7.3%), respectively. In comparison with the original design of Royal HaskoningDHV, the weight and costs decreases were 188 ton (19.8%) and €285000 (19.6%), respectively.

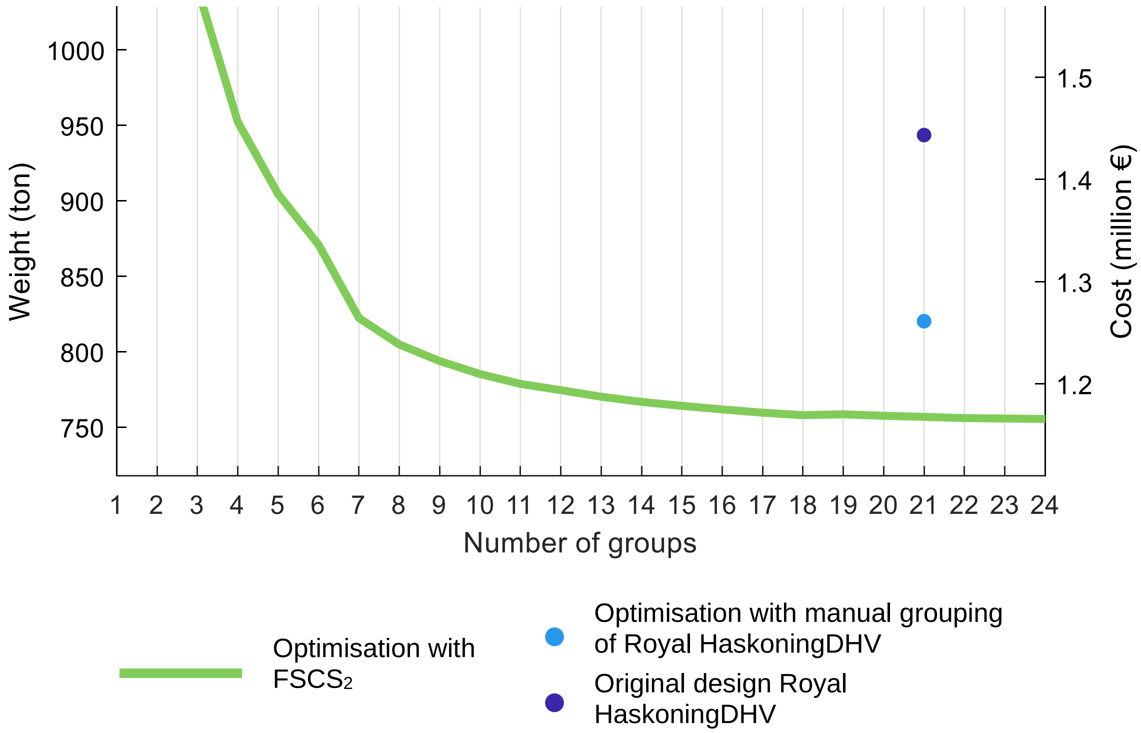


Figure 68 – Optimum design using $FSCS_2$ versus design Royal HaskoningDHV. Optimisation with grouping method shows significant weight and cost reduction, or reduction of the diversity of profiles in the design.

Result for seven groups

With the application of a $FSCS_2$, a design with similar weight and cost as the manual grouped design was found, having less groups. This was the case for seven groups, for which the improved design gave a weight of 816 ton and cost of €1.256 million. The result is shown in Figure 69. These designs resulted in a grouping which would be hard to find from engineering's experience. For example, the curved beams in the outside, as shown in Figure 69d, had the same profile as some vertical and diagonals on the inside plane, of which a few are indicated with red circles in Figure 69c.

It should be noted, that in this optimisation problem, a less extensive linking was applied than for the other analyses on the case-study: only symmetrical beams and each of the curved continuous beams were linked. By doing so, the grouping method evaluated more possible groupings. Consequently, this result was slightly lighter and cheaper than the result for 7 groups in Figure 68 with a difference of 6000 kg and €12000.

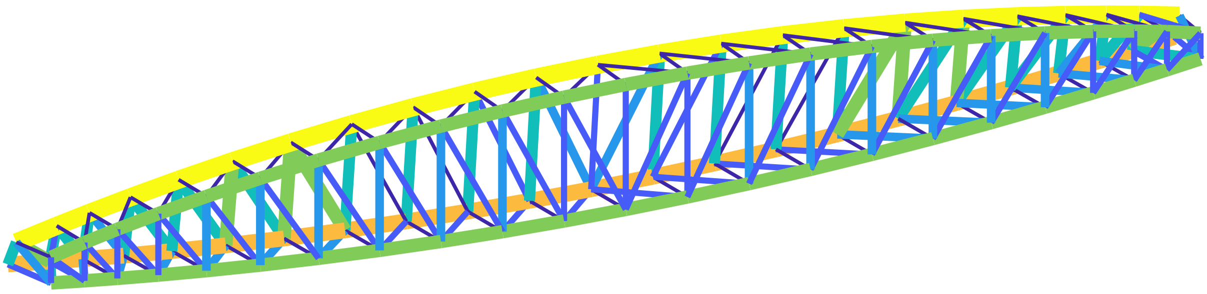


Figure 69 – Result Feyenoord stadium $FSCS_2$ 7 groups

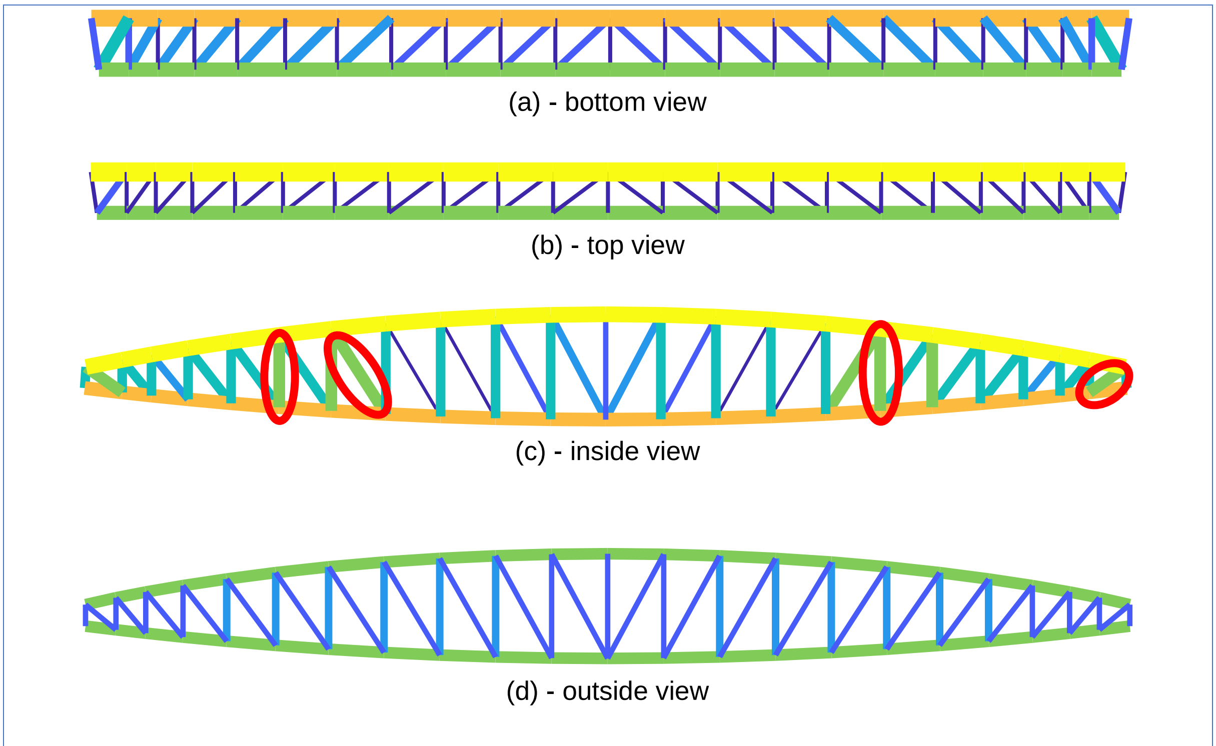
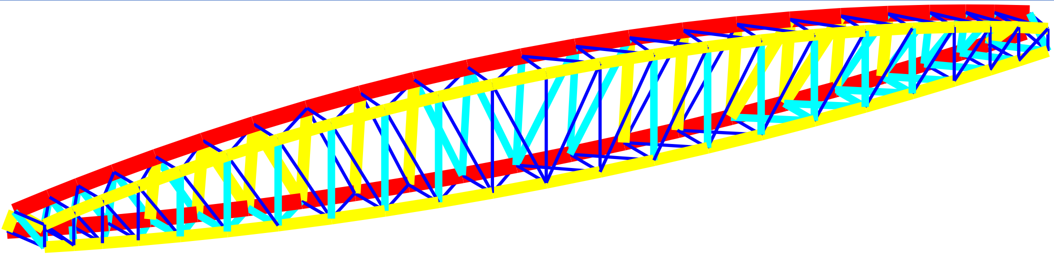


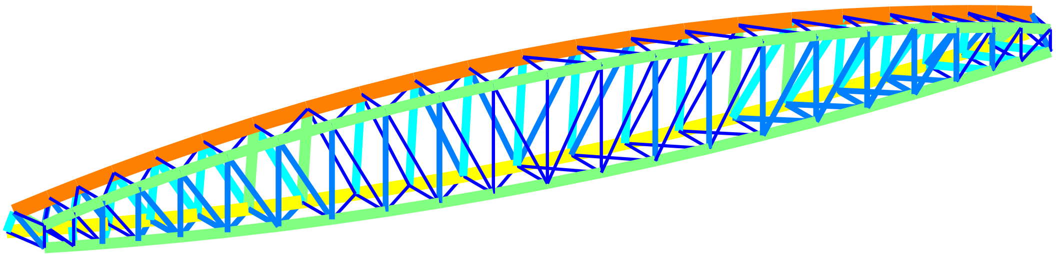
Figure 70 – Result Feyenoord stadium $FSCS_2$ 7 groups side views

Result for diverse number of groups

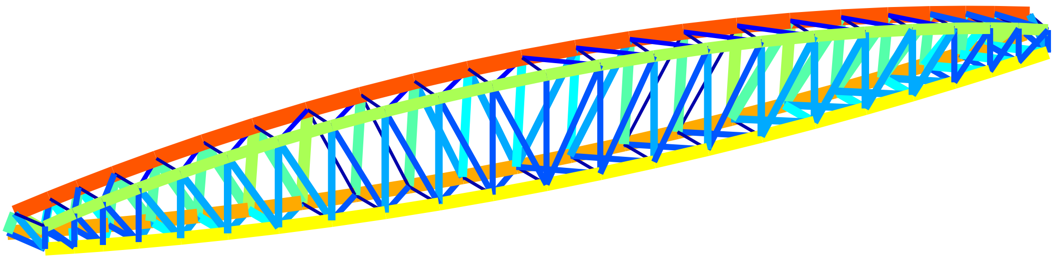
The optimum grouped designs for 4, 6, 10, 15 and 21 groups are shown in Figure 71. Again, these solutions contained a grouping which would be hard to design manually.



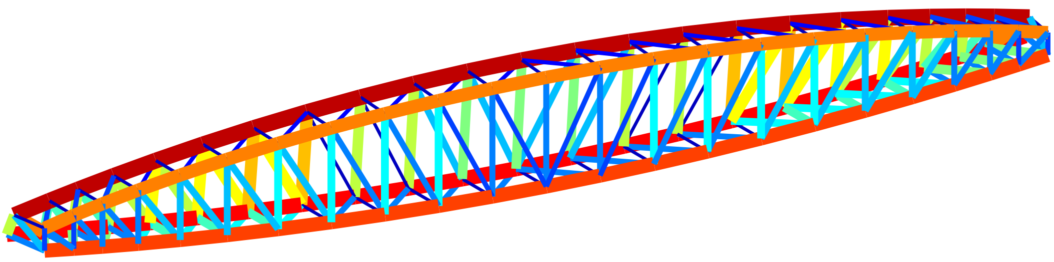
(a) – 4 groups



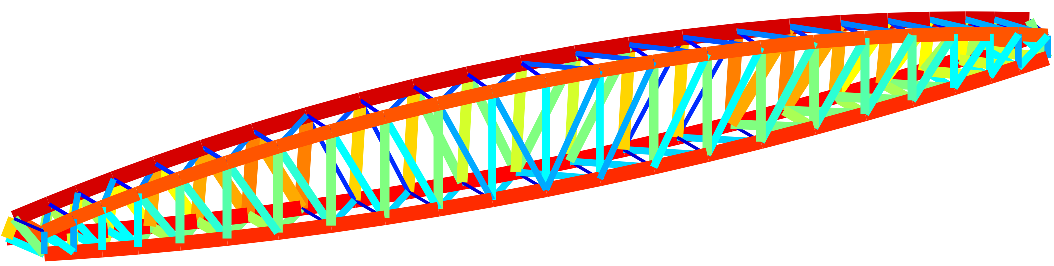
(b) – 6 groups



(c) – 10 groups



(d) – 15 groups



(e) – 21 groups

Figure 71 - Optimum grouped designs

6.4 Conclusion

This chapter has investigated the added value of grouping methods on a minimum cost design, answering the fifth subquestion:

5. How do grouping methods improve the minimum cost design of a real-life case-study?

From the application of cost optimisation for the 18-bar cantilever truss, it is concluded that grouping methods ease the process of finding the minimum cost design. Whereas the implicit cost optimisation problem is complicated to solve by optimisation, grouping methods solve the problem in simpler subproblems. By doing so, the global grouped optimum can be found with high certainty. Furthermore, cost and weight optimisation give the same optimum solution if the supply costs are dominant in the cost function. This was observed in both the 18-bar cantilever truss for one to three groups, and for the Feyenoord stadium for all number of groups.

The cost optimisation of the Feyenoord stadium shows the ability to reduce the costs and weight of a design significantly compared to a manual grouping. The use of a grouping methods allowed a weight and costs decrease of the analysed truss beam of 7.8% and 7.3%, respectively, compared to manual grouping by Royal HaskoningDHV. A decrease of weight and costs of 19.8 and 19.6%, respectively, was found when comparing the new method to the original design of Royal HaskoningDHV in which no optimisation was performed.

Finally, application of grouping methods can be used to reduce the diversity of profiles in a design significantly. In the Feyenoord roof design, the new grouping methods showed a reduction of the number of groups from 21 to 7 groups, while keeping the weight and cost approximately the same.

7 Discussion

In this chapter, the limitations of this research project are discussed by evaluating the scope, the new grouping method, the comparison of the numerical experiments and application in practice.

7.1 Scope

In the scope, some limitations on the kind of problems and a certain way of analysing the benchmark problems were proposed. Nonetheless, application outside the scope is possible as well. Table 18 shows the effect on the optimisation problem, when the problem is applied outside the limitations of the scope. Each of these effects, and the resulting performance of grouping methods, is described in detail.

Table 18 – Effect on grouping methods when applied outside scope

	Variable geometry	**Custom set of profiles**	**Building code check**	**Detailed design**	**Nonlinear analysis**	**Other cost function**
Design variables	More variables	Other options		More variables		
Objective function						More complex
Constraint function			More complex	More complex	More complex	
Resulting performance grouping methods	Unknown	Similar, except for PS method	Dependent on method	Unknown	Dependent on problem	Dependent on method

Variable geometry

In this research project, only problems with a fixed geometry were analysed. Nonetheless, all grouping methods are also applicable to problems in which the geometry of the problem is allowed to change, although this introduces a much bigger search space and many local optima: the coordinates of the nodes, the number of nodes, and the connectivity of the nodes are added to the design variables. As this introduces a new optimisation problem, bigger in size and complexity, the effectiveness of the grouping methods on these problems is unknown. I expect that only the methods which adapt the grouping multiple times are effective, as a changing geometry has significant influence on the force distribution.

Custom set of profiles

The use of a set of standard steel profiles can be replaced by a custom set of profiles without consequences on the performance of the grouping methods. The order of section properties of a custom set of profiles might be less smooth than a set of standard steel profiles, but this does not affect the grouping methods, only the optimisation process. Therefore, the grouping methods are applicable to custom profiles too. The method which manually selects a reduced set of profiles is an exception, as it might be more complex to manually select the optimum profiles of a custom set. However, in this study, the performance of this grouping method was assumed to be poor for a set of standard steel profiles as well.

Building code check

The benchmark problems included a simplified structural check on yield stress, buckling and displacement. For the design of real structures, building codes include a check on more requirements. These extensive requirements may not be suitable for the optimisation design process due to high number of needed calculations. Nonetheless, if all requirements of a building code are included in an efficiently programmed constraint function, the optimum result probably changes compared to application of simple structural check.

This new constraint function has no influence on the grouping of the grouping methods which are applied before the optimisation starts, only on the optimum profiles of that grouping. I expect the resulting designs to be suboptimal, as a building code has many requirements which cannot be covered adequately by the simple guidelines of the grouping methods.

The methods which include grouping during the optimisation process evaluate the constraint function. Therefore, these methods can be applied to a building code constraint function, with the side note that the constraint function might introduce additional local optima, influencing the effectiveness of the optimisation itself.

For the new fully stressed combinatorial search method, a building code constraint function influences the grouping because the fully stressed design is changed. However, as this fully stressed design is iteratively changed, I expect this method to adequately cope with the new constraints.

Finally, for the methods which are based on an ungrouped optimisation, I expect the global grouped optimum to be less alike with the ungrouped optimum: as more demands on the structure are set, I think the optimum of individual members is strongly dependent on other members. Therefore, any adaption of the ungrouped design will have a strong effect on the feasibility, and therefore optimality, of the final design.

Detailed design

The structural design was evaluated on a global level in this study and the detailed design was not considered. In current practice, the detailed design is performed after completion of a global design with some safety margin. This margin allows for a feasible detailed design, without the need for changes in the global design. A similar approach can be adopted with the optimisation design process, by including this margin in the optimisation of the global design as well. It changes the result of the optimum design, but the relative performance of the grouping methods is unaffected.

The applicability of grouping methods on the detailed design is unknown because the detailed design does not only depend on the choice of profiles; details like bolts, welds and cuts are added to the design variables as well. Although it is probably beneficial to have a low number of distinct bolts and welds as well, the current grouping methods are not directly applicable for grouping of other aspect than sizing of beam elements.

In theory, the global and detailed design can be implemented in one optimisation problem, and remove the need of the margin in the global design. However, this will drastically increase the size and complexity of the problem, because profiles and details of the entire structure are included in one problem. The resulting problem is probably too big to solve in its totality.

Nonlinear analysis

Instead of linear analysis, a nonlinear analysis can be adopted, thereby changing the constraint function. This increases the calculation time of each structure analysis, but gives a more exact evaluation of the structural behaviour, if that is desired. The grouping methods are as applicable as to the linear analysis, but the effect of the nonlinearities on the resulting optimum design and grouping is strongly dependent on the problem.

Cost function

As the perfect cost function does not exist, I chose a cost function from literature in this study. If another cost function would be chosen, another optimum result is found. As was described in Chapter 4.2, the grouping is only dependent on the objective function for a few methods grouping; the methods which perform grouping before the optimisation starts do not take into account the objective function, while the grouping is changed for the methods which are applied in a loop, or during or after optimisation. As with the more specific constraint function from building code requirements, I expect that only the new method, and the methods which change the optimisation process, are able to adequately find the grouped cost optimum.

Furthermore, as the optimality in cost optimisation is strongly dependent on the cost function and the problem itself, the performance of grouping methods can only be evaluated with respect to a specific cost function, not to all possible cost functions. Nonetheless, for cost functions in which the weight is dominant, as it was for the 18-bar cantilever truss up to three groups and for the case-study for all groups, the result of the comparison on weight optimisation is representative.

7.2 New grouping method

High computational effort

The new grouping method uses a fully stressed design and combinatorial search in its grouping process. For big problems with many members or a big profile database this can result in many computations, reducing the efficiency of this method. However, optimisation methods are proposed in literature to prevent evaluation of all combinations of the combinatorial search (Reitman and Brent Hall 1990). Similarly, single-variable optimisation methods can be applied for each beam in the fully stressed design (Langelaar and van Keulen 2019). Implementation of these optimisation can be implemented in the existing framework, as shown in Figure 72.

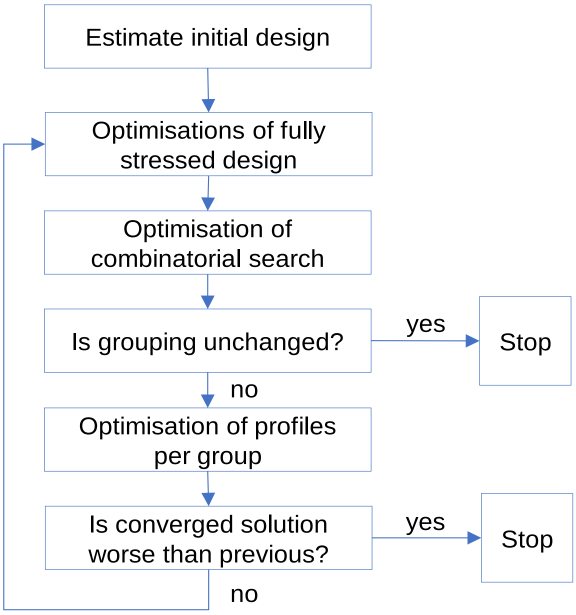


Figure 72 – Framework fully stressed combinatorial search with optimisation of subproblems

It should be noted that the number of combinations was no limitation for the benchmark problems. Furthermore, the objective function evaluations in the combinatorial search are cheap computational evaluations; for weight optimisation the objective function evaluation only involves a summation and multiplication of the weight per unit length of members with their lengths.

Memory issues

While the time to evaluate a big number of combinations was no limitation in this study, memory issues were encountered during the case-study of the Feyenoord stadium. These memory issues were solved by a more extensive elementary linking, but this prevented the grouping method from finding the optimum grouping. This adapted elementary linking led to a reduced number of combinations in the combinatorial search; for 21 and 22 groups, the combinatorial search evaluated a maximum of $2.7 \cdot 10^{11}$ combinations with additional elementary links. For more combinations, the size of a single matrix from the algorithm of the combinatorial search shown in Appendix A, exceeded the allowed matrix size in MATLAB. If the combinations are stored differently, this problem can be solved without compromising on the optimum result.

It should be noted that the change in elementary linking had a small effect on the optimal result in this study; only a difference of 6 ton (on a total of 816 ton) was observed for the result with seven groups.

Initial design

Whereas the second implementation of the fully stressed combinatorial search takes the heaviest possible profiles as an initial reference design, adapting a different uniform initial reference design in the third implementation gave a lighter optimum design in some benchmark problems. Conclusively, although the loop in this method changes the reference design during its analysis, the initial reference design has some influence on the final result. This difference was found to be marginal in this study, but only uniform designs were regarded as an initial reference design. A non-uniform design might result in an even lighter optimum design.

Perfect grouping method

The perfect grouping method was not found in this study. This ideal method should require little or no additional calculations and is guaranteed to include the grouped in a simple and small search space. A possibility of reducing effectively while guaranteeing inclusion of the grouped optimum would be to create an optimisation method with two design variables, x_1 and x_2, of which x_1 refers to the member grouping and x_2 to the profile selection. Therefore, a list of all $\left\{ {n \atop k} \right\}$ member groupings has to be created in which the groupings have exactly the required *k* number of groups for all *n* beams. This list could be generated by 'Algorithm U', as proposed by Knuth (2014). Secondly, a list should be created of all $\binom{m}{k}$ combinations of selecting the specified k number of profiles from the total profile database of size *m*. The resulting design variable is two-dimensional, with an index $x_1 = \left\{ 1, \ldots, \left\{ {n \atop k} \right\} \right\}$ to one of the possible member groupings, and an index $x_2 = \left\{ 1, \ldots, \binom{m}{k} \right\}$ to one of the possible selection of profiles. Such a method would reduce the search space into a subspace which guarantees inclusion of the grouped optimum, with no overlap with subspaces for other number of groups. In terms of the size of the search spaces and inclusion of the grouped optimum, such a method would be perfect. However, I expect the resulting optimisation problem to have a complex search space with many local optima, as the problem is converted in a two-dimensional problem with an order which follows from the two lists of member and profile grouping. Consequently, such a method will not be able to find the global grouped solution, as all existing optimisation methods are sensitive to local optima. If a similar grouping method would be invented in which the number of local optima does not increase, better performance can be achieved.

Perfect optimisation method

A generally applicable optimisation method which is less sensitive to local optima is desired. However, this is a complex and unsolved mathematical problem. Furthermore, the 'No free lunch theorem' by D.H. Wolpert and W.G. Macready (1997) states the if the performance of an optimisation is elevated for one class of problems, it will be at the cost of the performance in another class of problems. Conclusively, no generally applicable optimisation method can be found. As the perfect optimisation algorithm does not exist, the combination of the proposed new fully stressed combinatorial search method with one of the available existing optimisation methods is the best available choice in the practical field of civil engineering.

7.3 Numerical comparison

Manual grouping

For the manual grouping method, only one grouping possibility per benchmark problem was used. These groupings were taken from literature, or I came up with those using my own engineering experience. Figure 73 shows the proposed manual grouping in this study, in which each group has a different colour. Another engineering might choose to group verticals and diagonals in one group, and assigns two groups for the bottom truss. However, as the grouping problem has many solutions, I expect the other grouping methods to outperform most engineers with this manual method.

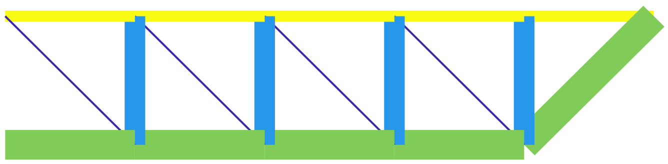


Figure 73 – Manual grouping 18-bar cantilever truss as it was used in this study. Other manual groupings are possible are well.

Reduction profile database

Simplifications were made in this study by reducing the available profile database in the benchmark problems. This adaption simplified the optimisation problem of all methods, except for the profile selection method which reduces the profile database even more as part of its grouping approach. A bigger and more diverse profile database would allow for a more optimal grouped optimum, but increases the size and complexity of the problem. However, I assume that a representative comparison is made for small and big profile database sizes, because of the variety in the benchmark problems.

Elementary linking

In the benchmark problems, an elementary linking was applied. For the 160-bar truss tower, 117-bar 9-storey frame and the case-study, this elementary linking did not only include symmetry constraints. This was done to reduce the size of the optimisation problem, especially for the ungrouped optimisation. However, such a linking might exclude the true global grouped optimum. For example, for the 160-bar truss tower, the elementary grouping assumes that all four sides of the tower are identical, while on one size two electricity cables are attached, one another size one cable, and on the other two size no cables. Therefore, the global grouped optimum might not be symmetric, which is enforced by the elementary grouping. Nonetheless, this elementary grouping was applied for all grouping methods, so a fair comparison was made.

Dominant constraints

In the numerical experiments, the dominancy of strength or displacement constraints was evaluated as the constraints with the highest value in the final solution. However, multiple constraints contribute to the optimum design, and the constraint with the highest value does not necessarily indicate that this constraint has a dominant effect. Therefore, this dominancy provides an estimation, of which the result should be interpreted with care.

Stochastic optimisation methods

In this study, I made a comparison of grouping methods for which the resulting weight was found with a stochastic optimisation method. The variance in the result of this optimisation method was judged on their certainty and distribution of the optimum weights. However, the realisation of this stochastic process can be interpreted in diverse ways, influencing the comparison of the grouping methods. In literature, many different interpretations have been used; another option would be to extract the minimum, mean and standard deviation from the results. Nonetheless, I think the performance of the grouping methods was representatively compared, supported additionally by the comparison on the theoretical performance. This theoretical comparison included the complexity of the search space, providing an indication of the resulting variance after optimisation.

Options optimisation method

The ability to find the optimum value in all grouping methods is strongly influenced by the choice of the optimisation method and options for that optimisation method. I chose to use the implementation of the genetic algorithm of Deb (2009) and Deep et al. (2009), and varied the population size, the elite ratio, and the number of generations to convergence. Another optimisation method or change in options changes the results and the performance. However, the variety of optimisation methods and variations is endless, and each problem and grouping method requires varying the options. This makes it impossible to numerically compare the grouping methods independent of the options for the optimisation method. Still, I expect that the comparison in this study represented the performance of grouping methods good enough because all stochastic optimisation methods are based on the same principles. Furthermore, the numerical comparison was supported by a method-independent theoretical performance as well.

Implementation

Part of the optimisation method and grouping methods were programmed by the author of this study. In the implementation of the optimisation, this was done as efficiently as possible to reduce computational time. However, for grouping methods which chance the optimisation itself, the way of implementing influences the comparison; if the implementation can be made more effective, the computational time of these methods reduces. I do not expect that this had significant effect on the comparison, as the number of finite element evaluations does not change by a more effective implementation of these methods. These finite element evaluations showed a similar trend as the computational time in the results.

For the grouping methods, the computational time is dependent on the implementation as well. In this case, I did not focus on developing a fast and efficient code, as these methods are only applied one or several times during one analysis. Still, the computational time of the additional calculations was low compared to the optimisation. Conclusively, the implementation did not influence the comparison.

7.4 Application in practice

Simplification new method

In practice, most engineers do not have the tools or knowledge to apply optimisation. However, the grouping problem could be of valuable use in the conventional design process as well. The new method can be applied without an optimisation method, which might lead to satisfactory results. In that case, the profiles of the groups should be selected manually in every iteration, as shown in Figure 30. As the profiles are not needed to be fully optimal during iterations, and selecting only profiles is a simpler task than selecting both profiles and groups, I expect the resulting grouping of members to be satisfactory. However, the manual selection of profiles is probably not as good as when an optimisation is applied, and it slows down the process.

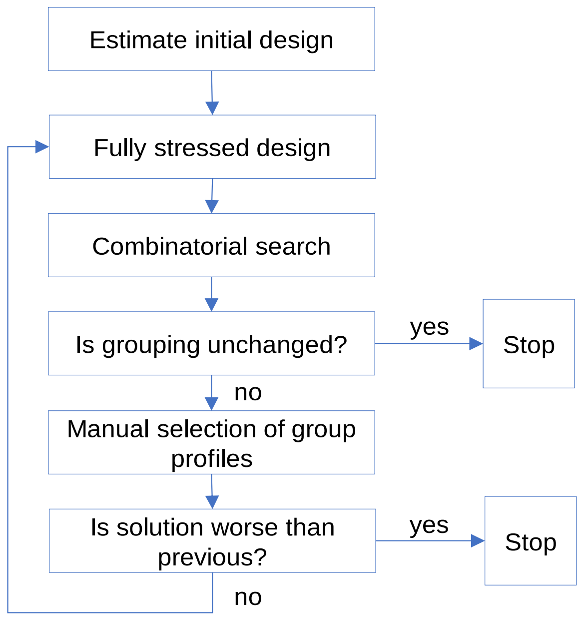


Figure 74 – Framework fully stressed combinatorial search without optimisation method but with manual selection of profiles for member groups.

Cost function

In the application of grouping methods to cost optimisation, a cost function from Watson et al. (1996) was used. It should be noted that this cost function is only applicable to medium sized steel projects, which were defined by Watson et al. (1996) as projects with costs greater than a $150,000 steel contract for supply, fabrication and erection. The 18-bar cantilever structure does not meet the application requirements of the costs, as the costs were too low. Therefore, the result of this analysis should be taken with care. Furthermore, the roof structure of the Feyenoord stadium is unique in size and shape. Consequently, although this structure is in the scope of the cost function, the costs given by this cost model may deviate from the actual costs of the structure.

Still, a perfect cost function might not be needed for design in practice in which requirements change during the process, constantly varying the optimum design. In that case, engineers would benefit highly from a cost function which approximates the cost good enough so that reasonable decision-making is possible. The cost function of Watson et al. might prove well enough in some cases, while another cost function might be more suitable in other cases. In order to know that, the scope of cost functions should be precisely defined and validated, in terms of structure type, design phase and which cost and sustainability aspects are included.

8 Conclusions

This chapter presents the conclusions of the research project on the problem of unbuildable design resulting from optimisation methods in civil engineering. With the use of grouping methods, optimum solutions with a low diversity of profiles can be found. However, the best method to do so is not known, which led to the following research question:

> Which method for grouping can find the lightest and cheapest steel structure with minimal computational effort?

This study concludes that the new method, the fully stressed combinatorial search, is the best available method to find the optimum grouped design with acceptable computational effort. Other main findings of this study are briefly described in the following statements:

1. The cardinality constraints method is the only method which reduces the search space while guaranteeing the inclusion of the grouped optimum for all kind of structures. However, this is not true for a high number of groups. Furthermore, the search space is altered by this method which makes it harder for an optimisation method to find the grouped optimum. The methods which do not reduce the search space, make the search space more complex as well. On the contrary, the other methods which reduce the search space have a simple and small search space, but might exclude the global optimum.

2. The number of computations in additional calculations is insignificant to the number of computations in optimisation, except for methods which use of the ungrouped optimum solution and the neural networks method. For the new fully stressed combinatorial search method, the number of additional computations is acceptable if the problem is not too big.

3. Manual grouping is the simplest method to apply, but its performance is strongly dependent on the experience of the engineer. The methods which group members based on axial force perform better but cannot find the global optimum due to the inability of combining compressive and tensile members. Lighter designs are found by the cardinality constraints method and ungrouped combinatorial search, although these methods require many finite element evaluations. Finally, the new method finds similar light structures as the previous two methods even though it requires less computational effort.

4. The optimum solution, including the optimum number of groups, can be found efficiently with the use of grouping method; it is possible to find lighter and cheaper designs with less computations if grouping methods are used. When the weight is dominant in the cost of a structure, an engineer can use the grouped designs to make a trade-off between cost or weight, and the diversity.

5. The costs of a structure can be reduced by applying a grouping method, in comparison with a manual grouping. In comparison with the conventional design process, the costs reduce even more. Alternatively, with the use of grouping methods the diversity of the solution can be decreased, keeping the weight and costs constant.

9 Recommendations

This chapter shows the recommendations for further research and application of grouping methods.

Research on grouping methods

1. Application of proposed optimisation methods to solve the subproblems of the new fully stressed combinatorial search method should be investigated. The effectiveness of these optimisation techniques may provide improved performance and applicability to problems with a bigger profile database and more members.

2. The effect of using a non-uniform initial design in the new fully stressed combinatorial search method should be investigated. This might allow for better results.

3. More research is needed on the development of a grouping method which is guaranteed to include the global grouped optimum in its search space, while not creating too much local optima. Such a method can outperform the available grouping methods.

4. Further research is needed on the performance of grouping methods in the case that the geometry is allowed to change. This allows for a more optimal generative grouped design, in which the engineer does not have to define the geometry manually.

5. The applicability of the available grouping methods to the detailed steel design, concrete structures, multi-material structures and composite structures should be investigated. Applications beyond civil engineering to the fields of mechanical, aerospace and maritime engineering are possible as well. All these applications benefit from a cheap, light and sustainable design with a low diversity of its components.

Application in practice

1. The performance of the new fully stressed combinatorial search method should be investigated when the sizing of member groups is performed manually by an engineer instead of with the use of optimisation methods. If the performance is good enough, the new method is a valuable tool for engineers who do not have the possibility to apply optimisation methods, but who desire an optimal grouping in their design.

2. If an engineering firm wishes incorporation of the building codes, it should focus on how to implement those codes efficiently so that the computation time can be kept minimal. Or, it should be investigated to what extent simple checks on yield stress, buckling and displacement, cover the demands from the codes.

3. As in the practical field of engineering the cost and sustainability of a design is of high importance, more research is needed on development of cost functions which are validated on a well-defined scope. These cost functions should be exact enough to allow reasoned decision-making on a design.

10 Bibliography

Adeli H, Sarma KC (2006) Cost Optimization of Structures. John Wiley & Sons, Ltd, Chichester, UK

Ajouz R (2018) Optimising production costs of steel trusses. TU Delft

Alberdi R, Khandelwal K (2015) Comparison of robustness of metaheuristic algorithms for steel frame optimization. Eng Struct 102:40–60. https://doi.org/10.1016/j.engstruct.2015.08.012

American Institute of Steel Construction (2017) AISC Shapes Databases V15.0

Arora JS (2017) Introduction to Optimum Design, Fourth edi. Elsevier, London

Arora JS, Govil AK (1977) An efficient method for optimal structural design by substructuring. Comput Struct 7:507–515. https://doi.org/10.1016/0045-7949(77)90015-3

Barbosa HJC, Lemonge ACC (2005) A genetic algorithm encoding for a class of cardinality constraints. In: GECCO 2005 - Genetic and Evolutionary Computation Conference

Barbosa HJC, Lemonge ACC, Borges CCH (2008) A genetic algorithm encoding for cardinality constraints and automatic variable linking in structural optimization. Eng Struct. https://doi.org/10.1016/j.engstruct.2008.06.014

Biedermann JD, Grierson DE (1995) A generic model for building design. Eng Comput 11:173–184. https://doi.org/10.1007/BF01271283

Biedermann JD, Grierson DE (1996) Training and using neural networks to represent heuristic design knowledge. Adv Eng Softw 27:117–128. https://doi.org/10.1016/0965-9978(96)00017-8

Chan CM (1992) An optimality criteria algorithm for tall steel building design using commercial standard sections. Struct Optim. https://doi.org/10.1007/BF01744692

Deb K (2000) An efficient constraint handling method for genetic algorithms. Comput Methods Appl Mech Eng 186:311–338. https://doi.org/10.1016/S0045-7825(99)00389-8

Deep K, Singh KP, Kansal ML, Mohan C (2009) A real coded genetic algorithm for solving integer and mixed integer optimization problems. Appl Math Comput. https://doi.org/10.1016/j.amc.2009.02.044

Deep K, Thakur M (2007a) A new crossover operator for real coded genetic algorithms. Appl Math Comput 188:895–911. https://doi.org/10.1016/j.amc.2006.10.047

Deep K, Thakur M (2007b) A new mutation operator for real coded genetic algorithms. Appl Math Comput. https://doi.org/10.1016/j.amc.2007.03.046

Dumonteil P (1992) Simple Equations for Effective Length Factors. Eng J 29:111–115

Fairclough HE, Gilbert M, Pichugin A V., et al (2018) Theoretically optimal forms for very long-span bridges under gravity loading. Proc R Soc A Math Phys Eng Sci 474:20170726. https://doi.org/10.1098/rspa.2017.0726

Galante M, Oñate E (1996) Genetic algorithms as an approach to optimize real-world trusses. Int J Numer Methods Eng. https://doi.org/10.1002/(SICI)1097-0207(19960215)39:3<361::AID-NME854>3.0.CO;2-1

Goldberg DE, Samtani MP (1986) Engineering Optimization Via Genetic Algorithm. In: Proceedings of 9th conference electronic computation. ASCE, New York, pp 471–82

Groenwold AA, Stander N (1997) Optimal discrete sizing of truss structures subject to buckling constraints. Struct Optim 14:71–80. https://doi.org/10.1007/BF01812508

Gutkowski W (1997) Discrete Structural Optimization, NV-1 onl. Springer Vienna, Vienna

Haapio J (2012) Feature-based costing method for skeletal steel structures based on the process approach. Tampere University of Technology

Holland JH (1975) Adaptation in natural and artificial systems : an introductory analysis with applications to biology, control, and artificial intelligence

Imai K, Schmit LA (2010) Configuration Optimization of Trusses. In: Optimization of Finite Dimensional Structures. CRC Press, pp 159–183

Kanno Y (2016) Global optimization of trusses with constraints on number of different cross-sections: a mixed-integer second-order cone programming approach. Comput Optim Appl 63:203–236. https://doi.org/10.1007/s10589-015-9766-0

Kaveh A, Zolghadr A (2011) A multi-set charged system search for truss optimization with variables of different natures; element grouping. Period Polytech Civ Eng. https://doi.org/10.3311/pp.ci.2011-2.01

Kazemzadeh Azad S, Hasançebi O (2015) Computationally efficient discrete sizing of steel frames via guided stochastic search heuristic. Comput Struct. https://doi.org/10.1016/j.compstruc.2015.04.009

Knuth DE (2011) The art of computer programming: v.4A: Combinatorial algorithms, Pt. 1. Choice Rev Online. https://doi.org/10.5860/choice.48-6329

Kraaijenbrink HG, van der Have RC, Luttmer J (2019) Nieuwbouw Stadion Feyenoord - Ontwerpberekening dakconstructie. Rotterdam

Krishnamoorthy CS, Prasanna Venkatesh P, Sudarshan R (2002) Object-oriented framework for genetic algorithms with application to space truss optimization. J Comput Civ Eng. https://doi.org/10.1061/(ASCE)0887-3801(2002)16:1(66)

Langelaar M, van Keulen F (2019) Lecture slides ME46060 Engineering Optimization. Delft

Lee KS, Geem ZW (2004) A new structural optimization method based on the harmony search algorithm. Comput Struct. https://doi.org/10.1016/j.compstruc.2004.01.002

Liu X, Cheng G, Yan J, Jiang L (2012) Singular optimum topology of skeletal structures with frequency constraints by AGGA. Struct Multidiscip Optim 45:451–466. https://doi.org/10.1007/s00158-011-0708-x

Mashayekhi M, Salajegheh E, Dehghani M (2016) Topology optimization of double and triple layer grid structures using a modified gravitational harmony search algorithm with efficient member grouping strategy. Comput Struct 172:40–58. https://doi.org/10.1016/j.compstruc.2016.05.008

Mashayekhi M, Salajegheh E, Salajegheh J, Fadaee MJ (2012) Reliability-based topology optimization of double layer grids using a two-stage optimization method. Struct Multidiscip Optim. https://doi.org/10.1007/s00158-011-0744-6

Murren PC (2011) Development and implementation of a design-driven harmony search algorithm in steel frame optimization. Notre Dame

OMA (2018) Feyenoord City. https://oma.eu/projects/feyenoord-city. Accessed 20 May 2020

Papalambros PY, Wilde DJ (2000) Principles of Optimal Design

Pavlovčič L, Krajnc A, Beg D (2004) Cost function analysis in the structural optimization of steel frames. Struct Multidiscip Optim 28:286–295. https://doi.org/10.1007/s00158-004-0430-z

Peter D. H (1985) Kernel estimation of a distribution function. Commun Stat - Theory Methods 14:605–620. https://doi.org/10.1080/03610928508828937

Provatidis CG, Venetsanos DT (2006) Cost minimization of 2D continuum structures under stress constraints by increasing commonality in their skeletal equivalents. Forsch im Ingenieurwesen/Engineering Res. https://doi.org/10.1007/s10010-006-0026-4

Rahami H (2019) Matrix Structural Analysis. In: MATLAB Cent. File Exch. https://nl.mathworks.com/matlabcentral/fileexchange/27012-matrix-structural-analysis?s_tid=FX_rc3_behav%0D. Accessed 13 Dec 2019

Reitman MI (1989) Optimal structural design in the USSR. Appl Mech Rev. https://doi.org/10.1115/1.3152419

Reitman MI, Brent Hall W (1990) Optimal structural optimisation. Eng Optim 16:109–128. https://doi.org/10.1080/03052159008941167

Saka MP (1990) Optimum design of pin-jointed steel structures with practical applications. J Struct Eng (United States). https://doi.org/10.1061/(ASCE)0733-9445(1990)116:10(2599)

Saka MP, Geem ZW (2013) Mathematical and Metaheuristic Applications in Design Optimization of Steel Frame Structures: An Extensive Review. Math Probl Eng 2013:1–33. https://doi.org/10.1155/2013/271031

Salajegheh E, Vanderplaats GN (1993) Optimum design of trusses with discrete sizing and shape variables. Struct Optim 6:79–85. https://doi.org/10.1007/BF01743339

Schmit LA (1960) Structural Design by Systematic Synthesis. In: Proceedings, 2nd Conference on Electronic Computation, ASCE

Shea K, Cagan J, Fenves SJ (1997) A Shape Annealing Approach to Optimal Truss Design With Dynamic Grouping of Members. J Mech Des 119:388–394. https://doi.org/10.1115/1.2826360

Steel Construction Institute (2020) Section dimensions & properties. In: Interact. “Blue Book.” https://www.steelforlifebluebook.co.uk/. Accessed 26 Jan 2020

Stolpe M (2016) Truss optimization with discrete design variables: a critical review. Struct Multidiscip Optim 53:349–374. https://doi.org/10.1007/s00158-015-1333-x

Stolpe M (2010) On some fundamental properties of structural topology optimization problems. Struct Multidiscip Optim 41:661–670. https://doi.org/10.1007/s00158-009-0476-z

Tam M (2020) User Manual Karamba3D. https://manual.karamba3d.com/3-in-depth-component-reference/3.5-algorithms/3.5.8-optimize-cross-section. Accessed 11 May 2020

Templeman AB (1988) Discrete optimum structural design. Comput Struct. https://doi.org/10.1016/0045-7949(88)90284-2

Tizani WMK, Nethercot DA, Davies G, et al (1996) Object-oriented fabrication cost model for the economic appraisal of tubular truss design. Adv Eng Softw 27:11–20. https://doi.org/10.1016/0965-9978(96)00016-6

Toğan V, Daloğlu AT (2006) Optimization of 3d trusses with adaptive approach in genetic algorithms. Eng Struct 28:1019–1027. https://doi.org/10.1016/j.engstruct.2005.11.007

Toğan V, Daloğlu AT (2008) An improved genetic algorithm with initial population strategy and self-adaptive member grouping. Comput Struct 86:1204–1218. https://doi.org/10.1016/j.compstruc.2007.11.006

Vanderplaats GN (1993) Thirty years of modern structural optimization. Adv Eng Softw 16:81–88. https://doi.org/10.1016/0965-9978(93)90052-U

Venkaya VB, Khot NS, Reddy VS (1969) Energy distribution in an optimum structural design. CFSTI, Springfield

Walls R, Elvin A (2010a) An algorithm for grouping members in a structure. Eng Struct 32:1760–1768. https://doi.org/10.1016/j.engstruct.2010.02.027

Walls R, Elvin A (2010b) Mass and stiffness distributions in optimized ungrouped unbraced frames. Int J Steel Struct 10:233–242. https://doi.org/10.1007/BF03215833

Watson KB, Dallas S, Van der Kreek N, Main T (1996) Costing of steelwork from feasibility through to completion. J Aust Steel Constr 30:9

Wolfram Research (2020) Lambert W-function. https://mathworld.wolfram.com/LambertW-Function.html. Accessed 20 May 2020

Wolpert DH, Macready WG (1997) No free lunch theorems for optimization. IEEE Trans Evol Comput. https://doi.org/10.1109/4235.585893

Wright AH (1991) Genetic Algorithms for Real Parameter Optimization. pp 205–218

Yang X-S, Bekdas G, Nigdeli SM (2016) Metaheuristics and Optimization in Civil Engineering. Springer International Publishing, Cham

Appendices

The appendices include:

Appendix A	Combinatorial search algorithm
Appendix B	Example fully stressed combinatorial search
Appendix C	Animations
Appendix D	Mathematical description genetic algorithm
Appendix E	Example genetic algorithm
Appendix F	Description benchmark problems
Appendix G	Cost model steel structures

Appendix A Combinatorial search algorithm

The algorithm for defining the groups in the combinatorial search is shown in Table 19, in which the profiles are sorted such that $p_1 > p_2 > \dots p_m$. Table 20 shows the corresponding profiles for which the objective function is evaluated. These profiles are the heaviest profiles in the corresponding group. In these tables, m is the number of distinct profiles, k is the number of groups and N_{CS} is the number of combinations following from Equation (2.3). It should be noted that one profile can represent multiple beams; the number of unique profiles is equal or less than the number of independent beams. The tables are an adaption from the original table from Walls and Elvin (2010).

Table 19 - Resulting groups for combinatorial search algorithm of determining N_{CS} possible combinations with grouping of m profiles $\{ p_1, p_2, \dots p_m \}$ to k groups, with $p_1 > p_2 > \dots p_m$. Adaptation of table from Walls and Elvin (2010)

Combination / Profile	1	2	···	*m-k+1*	*m-k+2*	*m-k+3*	···	N_{CS}
p_1	1	1	···	1	1	1	···	1
p_2	2	2	···	2	2	2	···	1
⋮	⋮	⋮		⋮	⋮	⋮		⋮
p_{k-2}	*k-2*	*k-2*	···	*k-2*	*k-2*	*k-2*	···	1
p_{k-1}	*k-1*	*k-1*	···	*k-1*	*k-2*	*k-2*	···	1
p_k	*k*	*k-1*	···	*k-1*	*k-1*	*k-1*	···	1
p_{k+1}	*k*	*k*	···	*k-1*	*k*	*k-1*	···	1
p_{k+2}	*k*	*k*	···	*k-1*	*k*	*k*	···	1
⋮	⋮	⋮		⋮	⋮	⋮		⋮
p_{m-2}	*k*	*k*	···	*k-1*	*k*	*k*	···	*k-2*
p_{m-1}	*k*	*k*	···	*k-1*	*k*	*k*	···	*k-1*
p_m	*k*	*k*	···	*k*	*k*	*k*	···	*k*

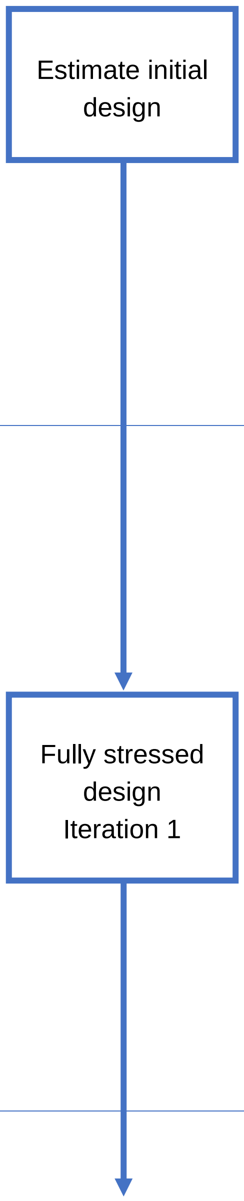
Table 20 - Resulting profiles for combinatorial search algorithm of determining N_{CS} possible combinations with grouping of m profiles $\{p_1, p_2, \ldots p_m\}$ to k groups, with $p_1 > p_2 > \ldots p_m$. Adaptation of table from Walls and Elvin (2010)

Combination / Profile	**1**	**2**	⋯	***m-k+1***	***m-k+2***	***m-k+3***	⋯	N_{CS}
p_1	p_1	p_1	⋯	p_1	p_1	p_1	⋯	p_1
p_2	p_2	p_2	⋯	p_2	p_2	p_2	⋯	p_1
⋮	⋮	⋮		⋮	⋮	⋮		⋮
p_{k-2}	p_{k-2}	p_{k-2}	⋯	p_{k-2}	p_{k-2}	p_{k-2}	⋯	p_1
p_{k-1}	p_{k-1}	p_{k-1}	⋯	p_{k-1}	p_{k-2}	p_{k-2}	⋯	p_1
p_k	p_k	p_{k-1}	⋯	p_{k-1}	p_k	p_k	⋯	p_1
p_{k+1}	p_k	p_{k+1}	⋯	p_{k-1}	p_{k+1}	p_k	⋯	p_1
p_{k+2}	p_k	p_{k+1}	⋯	p_{k-1}	p_{k+1}	p_{k+2}	⋯	p_1
⋮	⋮	⋮		⋮	⋮	⋮		⋮
p_{m-2}	p_k	p_{k+1}	⋯	p_{k-1}	p_{k+1}	p_{k+2}	⋯	p_{m-2}
p_{m-1}	p_k	p_{k+1}	⋯	p_{k-1}	p_{k+1}	p_{k+2}	⋯	p_{m-1}
p_m	p_k	p_{k+1}	⋯	p_m	p_{k+1}	p_{k+2}	⋯	p_m

The groups and profiles of the 18-bar cantilever truss are shown in an animation in Appendix C.4.

Appendix B Example fully stressed combinatorial search

As an example of the proposed fully stressed combinatorial search method, the 117-bar 9-storey frame is analysed by applying $FSCS_2$. Details of this structure are provided in Appendix F.7. In this example, only the grouping of the columns is explained in detail. The beams and bracings are grouped simultaneously.



First, for all columns the heaviest profile is chosen as an initial reference design, this is shown in Figure 75. For this problem, that is UC 305×305×283. The colours in this figure show members with the same profiles. For beams, columns and bracings, a separate heaviest profile is provided.

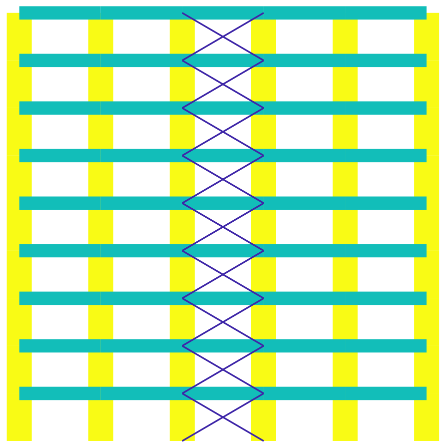


Figure 75 – Initial reference design

Based on the reference design, the fully stressed design is found. The result is shown in Figure 76. Again, colours represent members with the same profile.

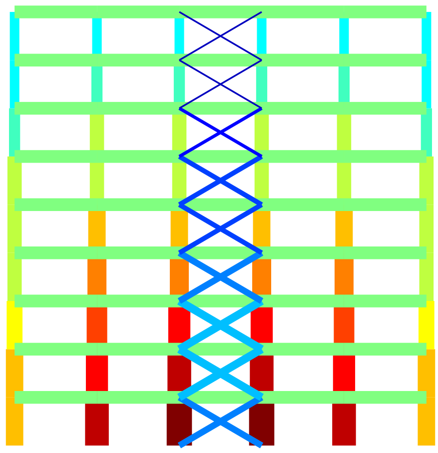


Figure 76 – Fully stressed design iteration 1

The result of the fully stressed design is combined in a combinatorial search. This is shown in Figure 77, in which each bar represents a column from the fully stressed design. The column colours correspond to the colours from Figure 76. The red horizontal lines represent the optimal limits of grouping to 3 groups.

Combinatorial search
Iteration 2

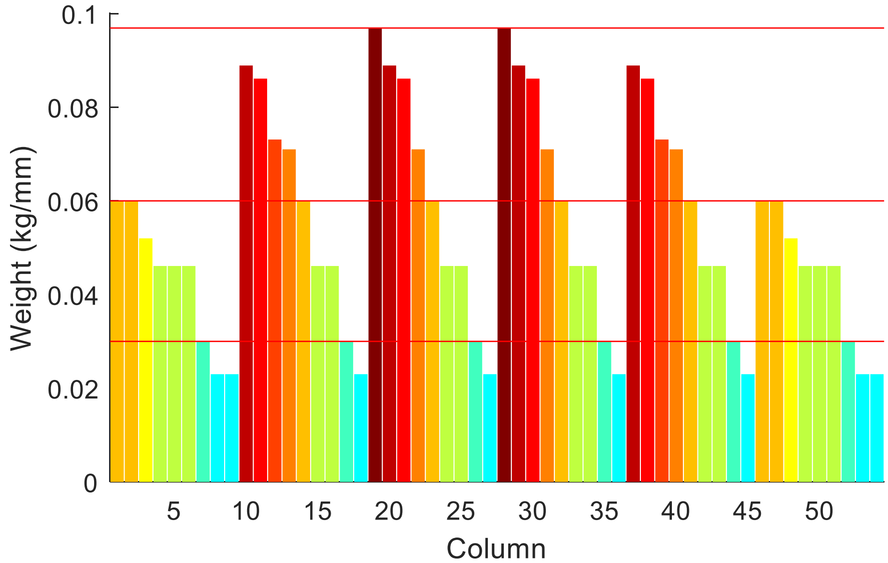


Figure 77 - Combinatorial search iteration 1

The resulting groups of the combinatorial search are shown in Figure 78. The thickness of the beams has no meaning in this figure. The colour now shows which members are grouped and are forced to the same (still unknown) profile.

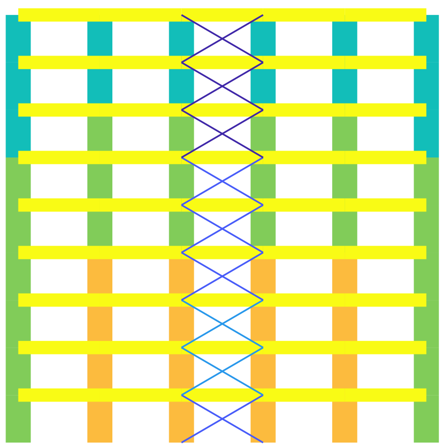


Figure 78 – Result grouping combinatorial search iteration 1

Is grouping unchanged?
Iteration 1

During the first iteration the convergence criteria are not checked yet, so this step is skipped.

Grouped optimisation Iteration 1

The resulting grouping of the combinatorial search is applied in a grouped optimisation. This process finds the optimum members for each of the 7 groups. This result is shown in Figure 84. It has a weight of 25598 kg.

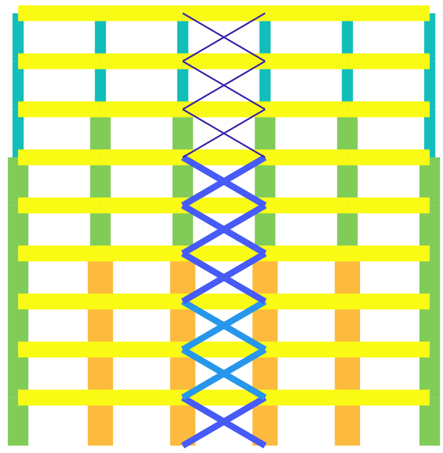


Figure 79 – Result grouped optimisation iteration 1

Is converged solution worse than previous? Iteration 1

During the first iteration the convergence criteria are not checked yet, so this step is skipped.

The first iteration has finished, now all steps are repeated for the second iteration

Fully stressed design Iteration 2

The result of the grouped optimisation from the first iteration is taken as reference design for the fully stressed design. The result is shown in Figure 80

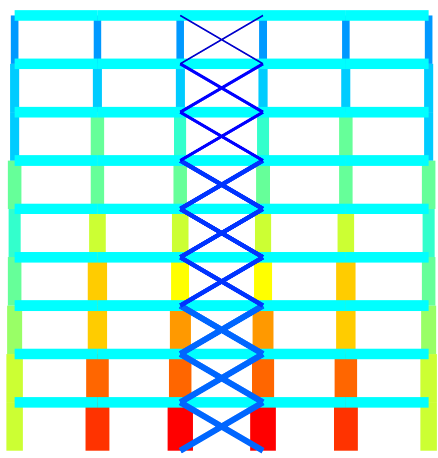
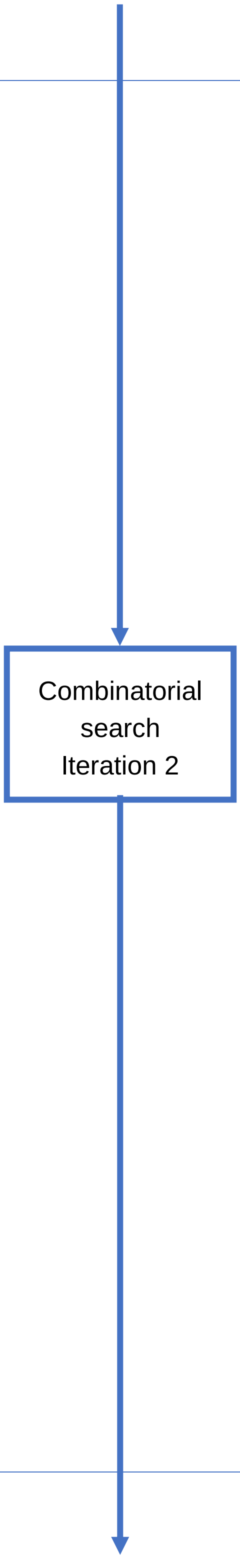


Figure 80 – Fully stressed design iteration 2



Again, the result of the fully stressed design is combined in a combinatorial search, as shown in Figure 81.

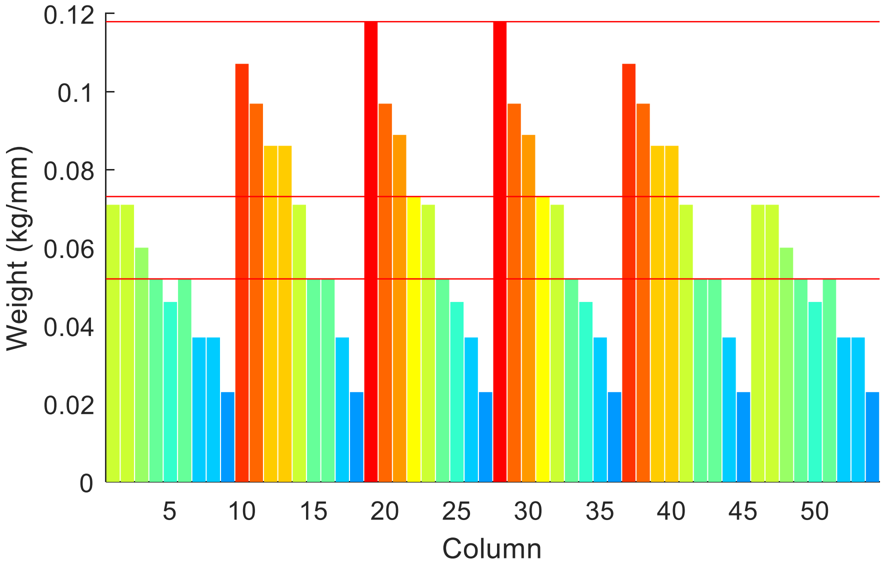


Figure 81 – Combinatorial search iteration 2

The resulting groups of the combinatorial search are shown in Figure 44.

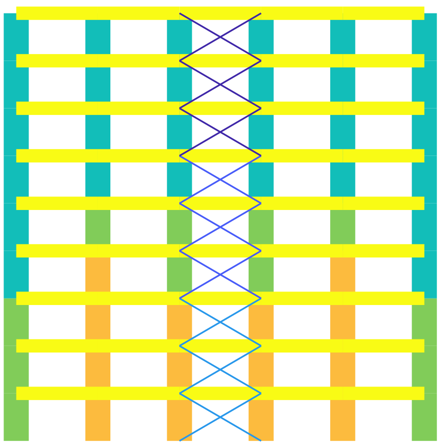
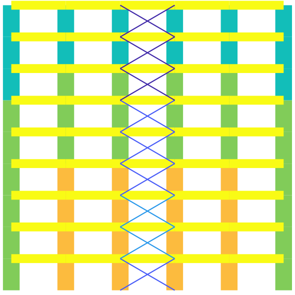


Figure 82 – Result grouping combinatorial search iteration 2

Is grouping unchanged? Iteration 2

No

Now, the first convergence criterium is checked. Therefore, the grouping of this iteration is compared to the previous grouping. This comparison is shown in Figure 83. As these groupings are not similar, the analysis continues



(a) – Iteration 1 (b) – Iteration 2

Figure 83 – Comparison grouping iteration 1 and 2

Grouped optimisation Iteration 2

The resulting grouping of the combinatorial search is applied in a grouped optimisation. This result is the structure shown in Figure 89 with a weight of 25162 kg.

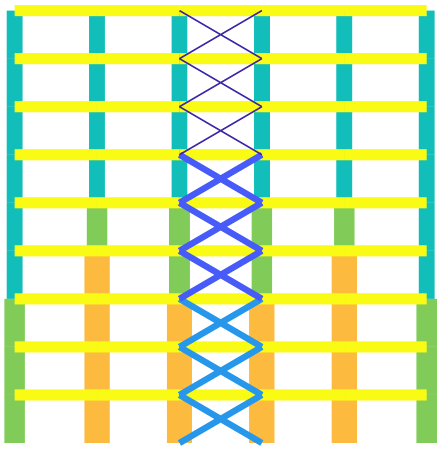


Figure 84 – Result grouped optimisation iteration 2

Is converged solution worse than previous? Iteration 2

No

The second convergence criterium is checked with the result of the optimisation. The weight of the optimum solution in the second iteration is 25162 kg, which is lighter than 25598 kg of the first iteration. Therefore, the analysis has not diverged and can continue.

The second iteration has finished, now all steps are repeated for the third iteration.

Fully stressed design Iteration 3

The result of the grouped optimisation from the second iteration is taken as reference design for the fully stressed design. The result is shown in Figure 85.

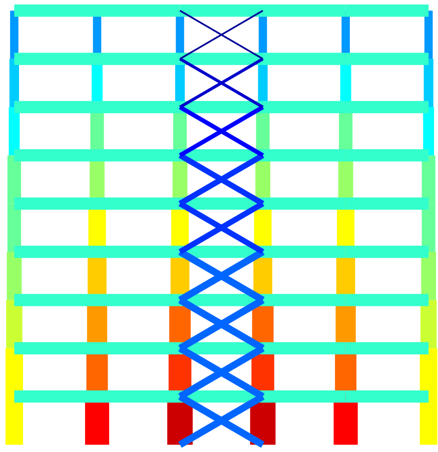


Figure 85 – Fully stressed design iteration 3

Combinatorial search Iteration 3

Again, the result of the fully stressed design is combined in a combinatorial search, as shown in Figure 86.

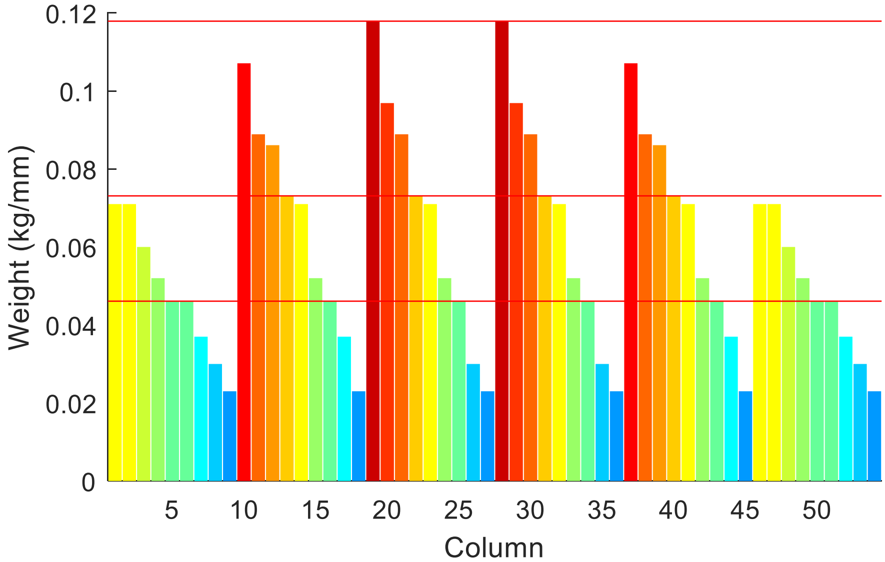


Figure 86 – Combinatorial search iteration 3

The resulting groups of the combinatorial search are shown in Figure 100.

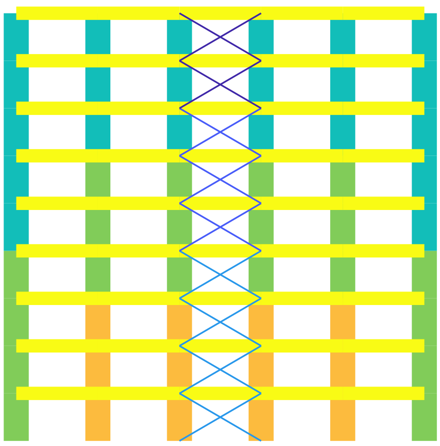


Figure 87 – Result grouping combinatorial search iteration 3

Is grouping unchanged? Iteration 3

No

Again, the first convergence criterium is checked. Therefore, the grouping of this iteration is compared to the previous grouping. This comparison is shown in Figure 101. As these groupings are not similar, the analysis continues.

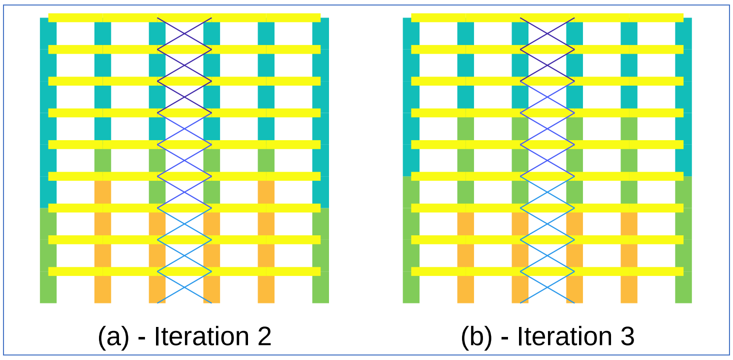


Figure 88 – Comparison grouping iteration 2 and 3

Grouped optimisation Iteration 3

The resulting grouping of the combinatorial search is applied in a grouped optimisation. This results in the structure shown in Figure 88 with a weight of 24779 kg.

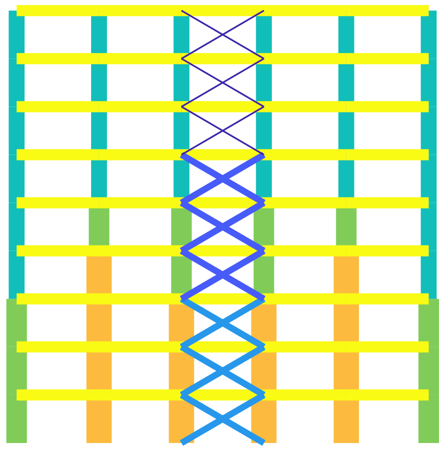


Figure 89 – Result grouped optimisation iteration 3

Is converged solution worse than previous? Iteration 3

No

The second convergence criterium is checked with the result of the optimisation. The weight of the optimum solution in the second iteration is 24779 kg, which is lighter than 25162 kg of the first iteration. Therefore, the analysis has not diverged and can continue.

The third iteration has finished, now all steps are repeated for the fourth iteration

Fully stressed design
Iteration 4

The result of the grouped optimisation from the third iteration is taken as reference design for the fully stressed design. The result is shown in Figure 90.

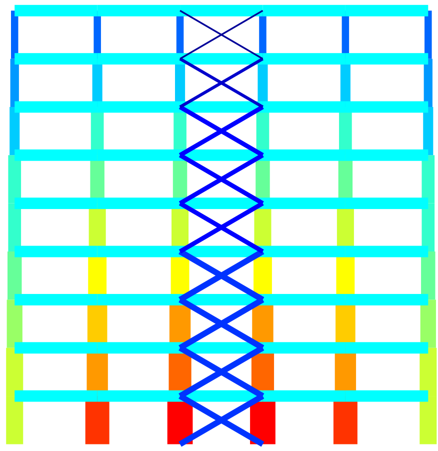


Figure 90 – Fully stressed design iteration 4

Combinatorial search Iteration 4

Again, the result of the fully stressed design is combined in a combinatorial search, as shown in Figure 91.

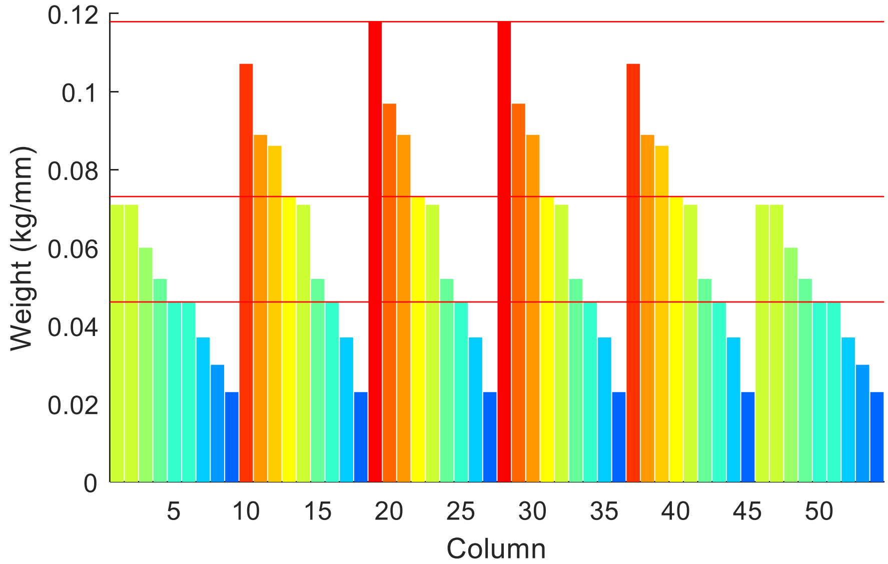


Figure 91– Combinatorial search iteration 4

The resulting groups of the combinatorial search are shown in Figure 92.

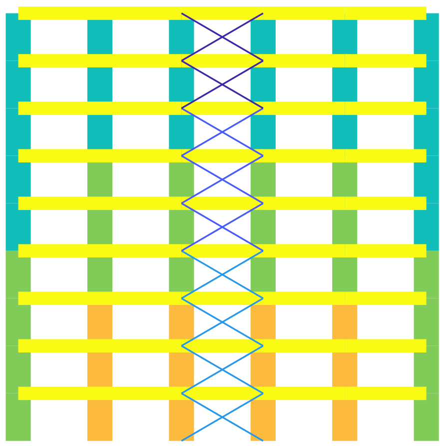
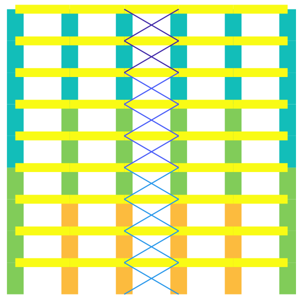


Figure 92 – Result grouping combinatorial search iteration 4

Now, the first convergence criterium is checked. Therefore, the grouping of this iteration is compared to the previous grouping. This comparison is shown in Figure 48. The grouping is the same for both iterations, so the analysis is finished.

Is grouping unchanged? Iteration 4

Yes



(a) - Iteration 3 (b) - Iteration 4

Figure 93 – Comparison grouping iteration 3 and 4

Stop

The result of the third iteration is the optimum result as shown in Figure 89.

Appendix C Animations

If this file is opened in Acrobat and Flash is installed, the following animations can be opened by clicking on the figures. Otherwise, the caption shows a link to the same animation online.

C.1 Optimisation method

The animation of Figure 94 shows how the design becomes more optimal during the optimisation of the 18-bar cantilever truss. The thicknesses represent the weight per unit length of the profiles. When multiple members have the same profile, these members are given the same colour.

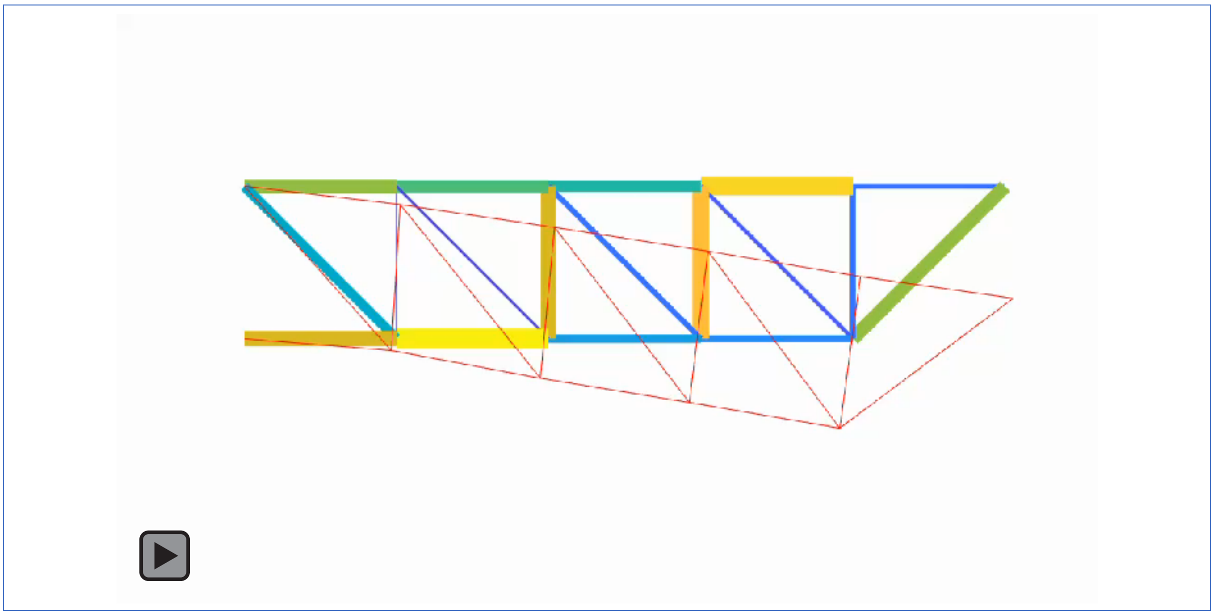


Figure 94 – Animation optimisation without grouping
This animation is also available at: https://s7.gifyu.com/images/Optimisation.gif

C.2 Optimisation with cardinality constraints

The animation of Figure 95 shows the optimisation process with the cardinality constraints method. The thicknesses represent the weight per unit length of the profiles. When multiple members have the same profile, these members are given the same colour. For the cardinality constraints method, the maximum number of groups is fixed to four in this case, but the distribution of these groups is variable. In some steps, multiple groups have the same profile, thus in that case, the effective number of groups is less than the desired number of four groups.

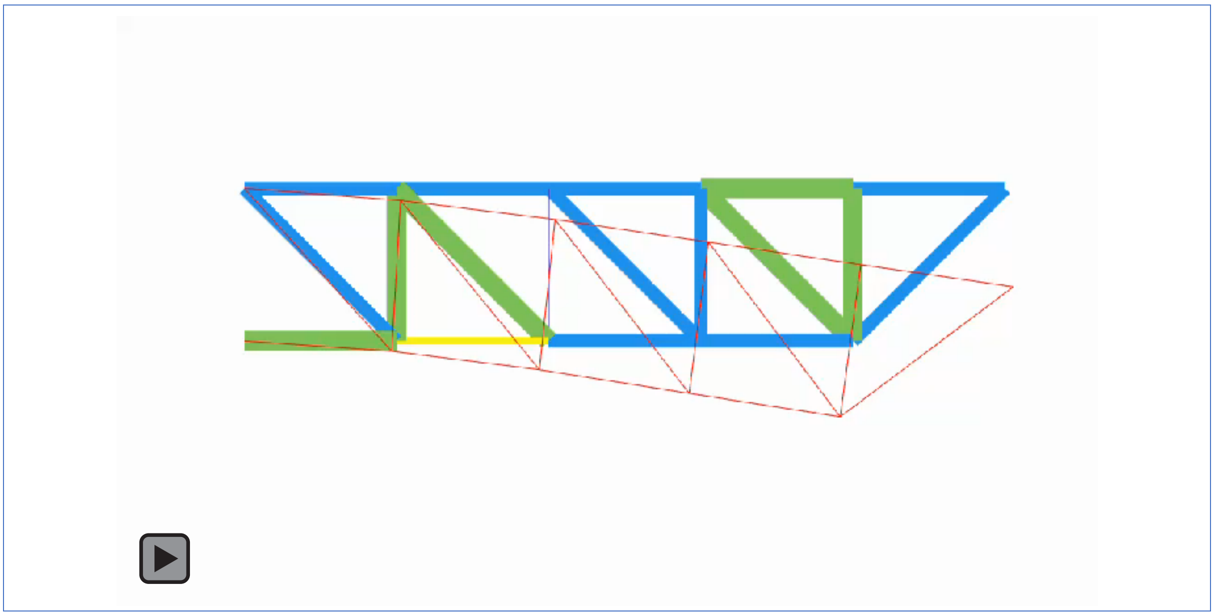


Figure 95 – Animation optimisation with cardinality constraints

This animation is also available at: https://s7.gifyu.com/images/CC.gif

C.3 Fully stressed design

The animation of Figure 96 shows the creation of the fully stressed design for the 18-bar cantilever truss. The top figure shows how for each member, the profile is altered from light to heavy. When a feasible design is found, it is saved in the bottom figure and the analysis continues with the next member. The red structure in the top figure shows the deformed structure.

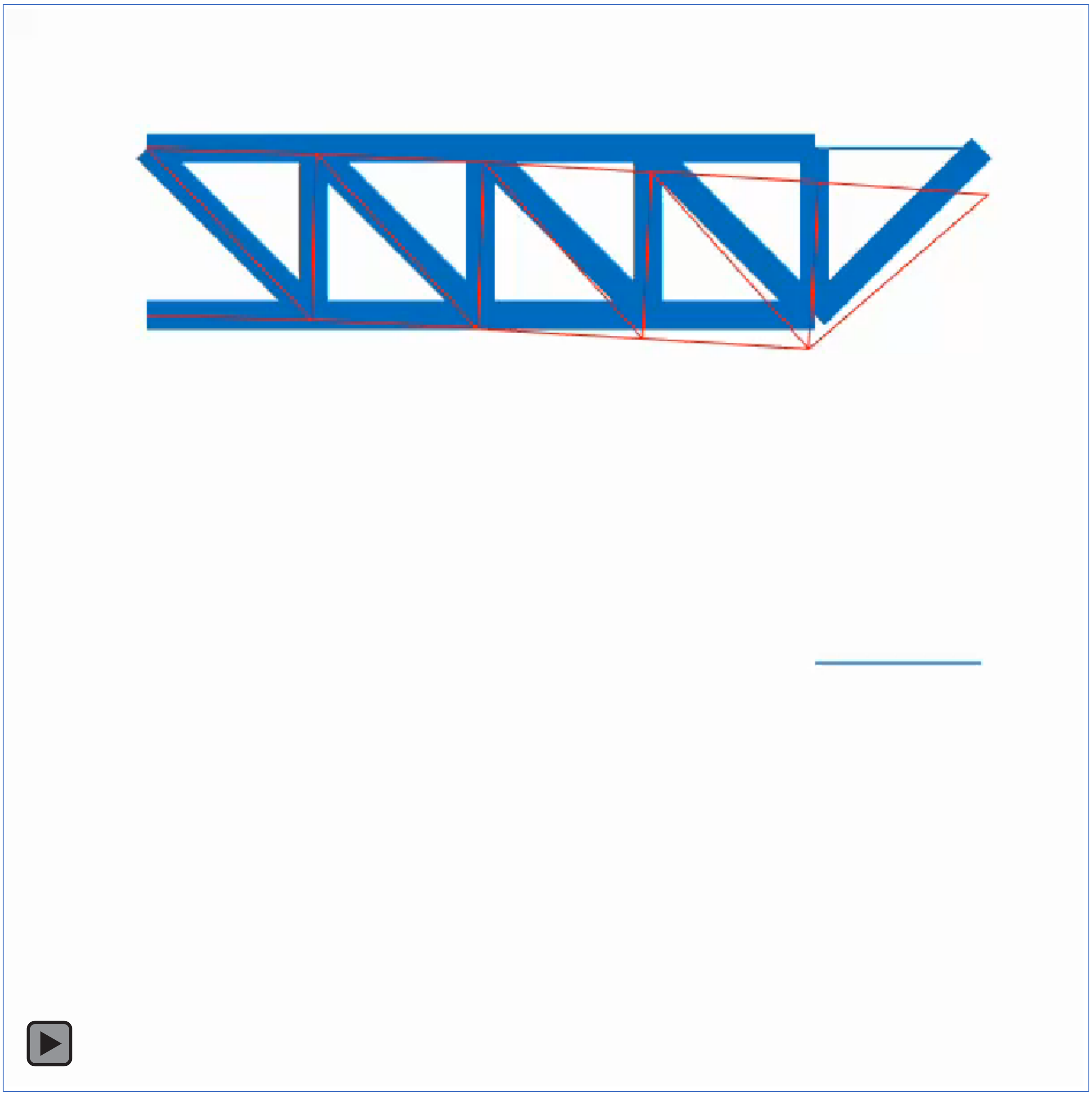


Figure 96 – Animation fully stressed design
Animation also available at https://s7.gifyu.com/images/FullyStressedDesign.gif

C.4 Combinatorial search

The animation of Figure 97 shows the procedure of the combinatorial search algorithm for the 18-bar cantilever truss. The top figure shows the combinations; the colours represent the groups, and the thicknesses represent the weight per unit length of the profiles. For each combination, the total weight is evaluated, which is shown in the bottom figure. The optimal combination is the combination with the minimum total weight, thus the minimum of the bottom figure.

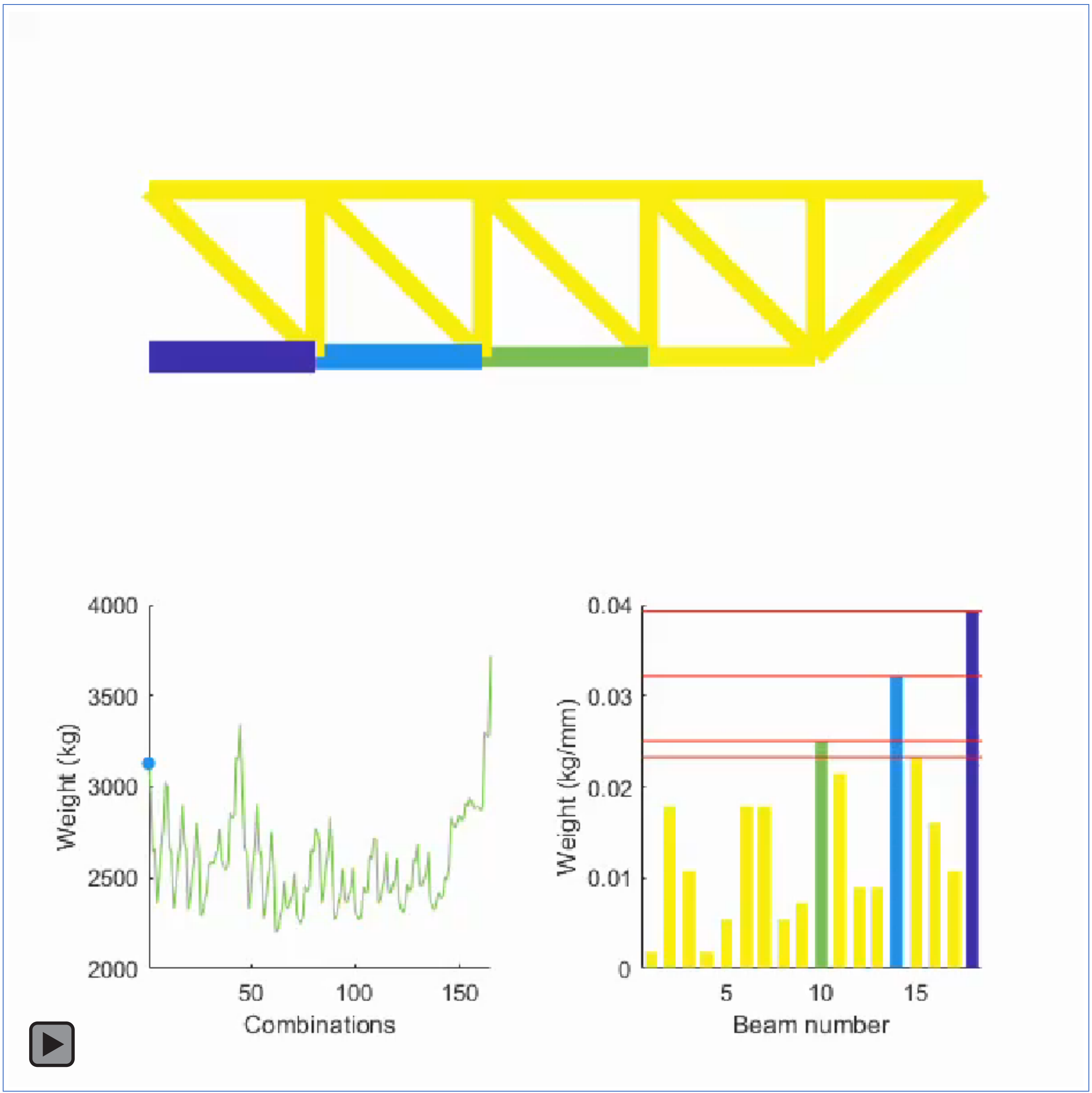


Figure 97 – Animation combinatorial search

Animation also available at: https://s7.gifyu.com/images/Combinatorial_search.gif

Appendix D Mathematical description genetic algorithm

This chapter includes an in-depth description of the mathematical implementation of the genetic algorithm. It covers characteristics of this implementation separately

Real-valued encoding

Opposed to the original implementation of binary values in the genetic algorithm by Holland (1975), variables were coded in real values in this study. Real-valued encoding shows increased efficiency, increased precision and more freedom in crossover and mutation functions (Wright 1991).

Constraint handling in selection operator

The applied genetic algorithm used a constraint handling technique developed by Kalyanmoy Deb (2000). This technique does not make use of (adaptive) penalty parameters, which are hard to define, are problem-dependent and influence the objective function.

The constraint handling technique proposed by Kalyanmoy Deb (2000) compares feasible and infeasible solutions when selection individuals from a population. By doing this, the algorithm converges to solutions in the feasible domain, while only influencing the objective in the infeasible domain. The comparison of solutions is called the tournament selection operator and is based on the following criteria, evaluated in this fixed order:

1. Feasible solutions are preferred above infeasible solutions.

2. A smaller fitness function is preferred for two feasible solutions.

3. A smaller fitness function is preferred for two infeasible solutions.

In these criteria, the fitness function $F(x_i)$ is defined as:

$$F(x_i) = \begin{cases} W(x_i) & \text{if all } g_j(x_i) \leq 0 \\ W_{\text{max}} + \sum \left| g_j(x_i) \right| & \text{otherwise} \end{cases} \tag{D.1}$$

In which x_i is the design expressed in i design variables, W is the weight of the structure and $g_i(x_i)$ is the value of constraint function j. W_{max} is the weight of the worst feasible solution, if it is not known, this value is 0.

The tournament selection can be played between more than two solutions. A tournament size of 3 was adopted as shown to be effective in literature (Deep et al. 2009).

An example of this constraint handling technique is given in Figure 98. These contour plots show the objective function *W* and fitness function *F* for a two-dimensional problem. The feasible domain is indicated with the crescent shape. It shows that the fitness values increase in the infeasible region, and the fitness value of the feasible region is equal to the objective function (Deb 2000).

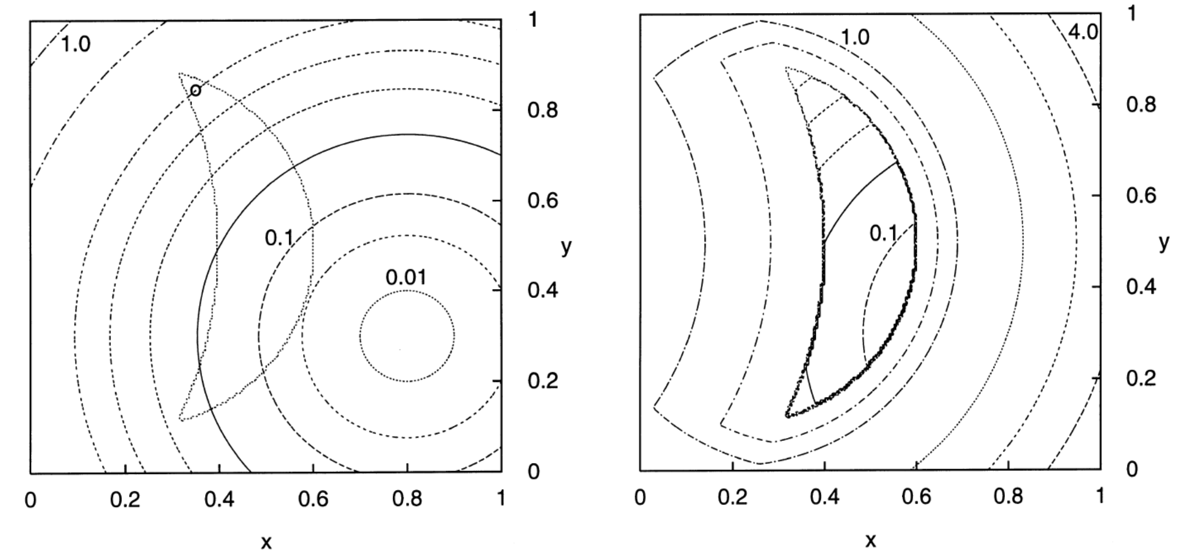


Figure 98 - Comparison of contour plots objective and fitness function
Figures taken from literature (Deb 2000)

Crossover operator

Crossover was applied by using the technique of Laplace crossover (Deep and Thakur 2007a). It generates a new generation with similar properties as the current generation, of which the spread is dependent on the diversity of the current generation. Random numbers β, satisfying the Laplace distribution are calculated by the following rule:

$$\beta_i = \begin{cases} a - b\ln(u_i) & r_i \leq 0.5 \\ a + b\ln(u_i) & r_i > 0.5 \end{cases} \tag{D.2}$$

In which *a* and *b* are location and scaling parameters. *a* = 0 and *b* = 0.35 are taken from literature (Deep et al. 2009). *u* and *r* are two random numbers from a uniform distribution between 0 and 1

A new generation $y_i^{y1} = \left(s_1^{y1}, s_2^{y1}, ..., s_n^{y1}\right)$ and $y_i^{y2} = \left(s_1^{y2}, s_2^{y2}, ..., s_n^{y2}\right)$ is generated from the current generation $x_i^{x1} = \left(s_1^{x1}, s_2^{x1}, ..., s_n^{x1}\right)$ and $x_i^{x2} = \left(s_1^{x2}, s_2^{x2}, ..., s_n^{x2}\right)$ as follows:

$$\begin{aligned} y_i^{y1} &= x_i^{x1} + \beta_i \left|x_i^{x1} - x_i^{x2}\right| \\ y_i^{y2} &= x_i^{x2} + \beta_i \left|x_i^{x1} - x_i^{x2}\right| \end{aligned} \tag{D.3}$$

In which the β_i follows from equation (D.2). As these operations can give non-integer results, the results are rounded up or down with a probability of 0.5 for both.

It should be noted that this crossover operator uses a notion of distance, which is not adequately defined in discrete steel profiles as the order of steel profiles is different for surface area, moment of inertia and torsional moment.

An example of the influence of the notion of distance is shown in Figure 99. This graph presents the properties of the steel profiles of the American W-section database. This full database was used in the 15-bar 3-storey frame problem. The order of profiles on the x-axis is as provided by the American Institute of Steel Construction (American Institute of Steel Construction 2017). The vertical axis shows the properties of area, second moment of inertia, elastic section modulus, radius of gyration around z and y-axis and weight per unit length. These properties are normalised with respect to the properties of the first profile. As clearly visible, ratios between different properties are not constant. Therefore, crossover of two similar solutions might result in a very different solution in the new generation.

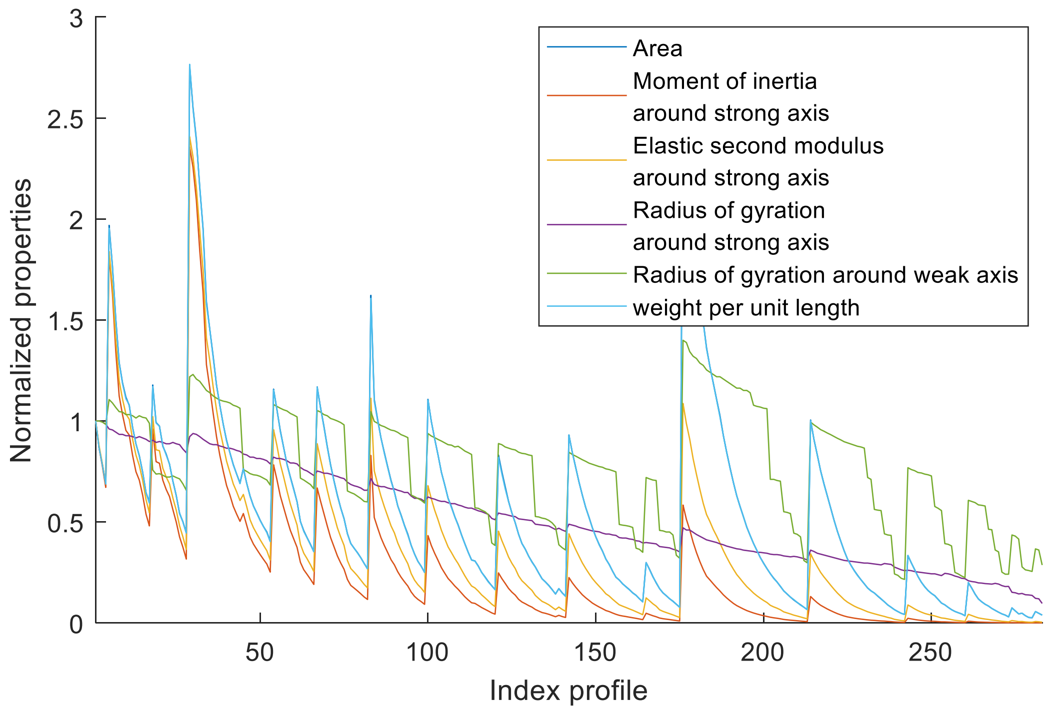


Figure 99 – Order of properties of steel profiles
Data taken from AISC database (American Institute of Steel Construction 2017)

The poorly defined order is an issue frequently encountered in optimisation methods for steel size optimisation (Alberdi and Khandelwal 2015). The influence of this is mostly implicitly neglected in literature (Murren 2011). In this study, only a small portion of the total available profiles was taken, or the rest of the problem was kept simple. This limits the number of local optima.

The crossover ratio was set to 0.8, as this value is used primarily in literature and is the default option in MATLAB (Deep et al. 2009). This crossover ratio controls the part of the generation on which the crossover operator works. A sensitivity analysis on this crossover ratio was performed to ensure convergence.

Mutation operator

For mutation of the individuals in a population, power mutation was adopted (Deep and Thakur 2007b). In this method, a random number *s* is created following the power distribution:

$$s = (s_1)^p \tag{D.4}$$

In which with s_1 is taken randomly from a uniform distribution between 0 and 1. *p is the* index of mutation. $p = 4$ is adopted from literature (Deep and Thakur 2007b).

A mutation in a solution $x_i^1 = \left(s_1^1, s_2^1, .., s_n^1\right)$ and $x_i^2 = \left(s_1^2, s_2^2, .., s_n^2\right)$ takes places as follows

$$x_i^2 = \begin{cases} x_i^1 - s\left(x_i^1 - x_i^{lb}\right) & t < r \\ x_i^1 + s\left(x_i^{ub} - x_i^1\right) & t \geq r \end{cases} \tag{D.5}$$

In which a lower bound x_i^{lb} and upper bound x_i^{ub} are defined for each design variable. *s* is taken from equation (D.4), *r* is taken randomly from a uniform distribution between 0 and 1 and $t = \dfrac{x_i^1 - x_i^{lb}}{x_i^{ub} - x_i^1}$. As with the crossover operator, the mutation operator can give non-integer results. These non-integer results are rounded up or down with a probability of 0.5 for both.

Again, it should be noted that this crossover operator uses a notion of distance, which is not adequately defined in discrete steel profiles as the order of steel profiles is different for surface area, moment of inertia and torsional moment. Again, this influence is neglected by taken only a small portion of the section database.

The mutation ratio, which defines the probability of a solution to mutate, is set indirectly by the crossover-ratio. If the cross-over ratio is increased, the mutation ratio decreases.

Elite operator

The elite operator assures that the individuals in the current generation with the best fitness value are transferred to the next generation, without being influenced by crossover or mutation operator. This makes sure the best solutions are kept in the optimisation process, while the rest of the generation explores the search space to find a better solution.

The elitism ratio, which defines the number of generations to be regarded as elite solutions, is problem dependent. As the number of elite solutions must be an integer value, the number is rounded up to a whole number of individuals. A sensitivity analysis on the elite operator was performed to ensure convergence in a preliminary phase of this study.

Population size

The population size is the number of individuals in one generation. A bigger values allows to evaluate more diverse solutions, but might prevent the algorithm from converging. As for the crossover and elite operators, a sensitivity analysis on the population size was performed to ensure convergence for the problems in this study.

Convergence criteria

As all other heuristic search methods, the genetical algorithm has no convergence check which guarantees that a local or global optimum is found. Therefore, a popular convergence criterion was used in study: to stop the optimisation process after a specified number of successive generations in which the best solution is unchanged. A sensitivity analysis on the number of successive generations was performed to ensure convergence in this study.

Parallel computing

MATLAB allows for parallel computing, which can accelerate the analysis. However, it was observed parallel computing extended computation time for simple problems. Therefore, it was not adopted. For the problems in which the structural analysis evaluation has a higher impact, parallel computing might speed up the process.

Appendix E Example genetic algorithm

The procedure of the genetic algorithm is shown for the 18-bar truss problem. For completeness, all steps of the optimisation design process are shown as presented in Chapter 1.1.2. All steps in this example are part of the block optimisation of the flow-chart shown in Chapter 5.1, Figure 44.

Collect data to describe system

The 18-bar truss problem is described by a fixed geometry, load situation and profile database. This data can be found in Appendix F

Formulate problem as optimisation problem

The problem is formulated in the standard design optimisation model. The objective function is the weight of the structure:

$$f(x_i) = \sum_{i=1}^{18} A_i L_i \rho_i \tag{E.1}$$

With A the cross-sectional area, L the length and ρ the density of a member.

Each of the 18 members x_i can be chosen from the profile database S:

$$x_i = (x_1, x_2, \ldots, x_{18}) \qquad x_i \in S \tag{E.2}$$

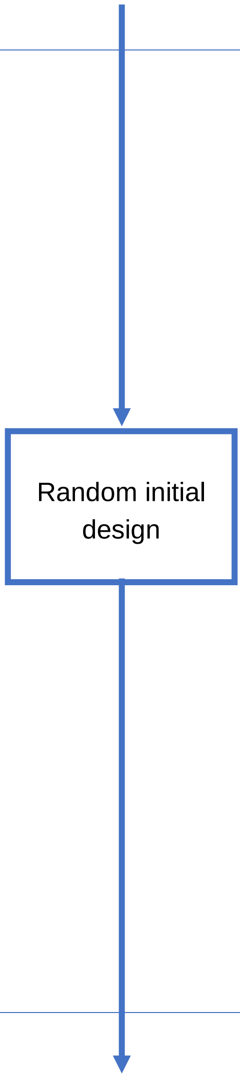
In which the profile set S contains 25 profiles, as prescribed in Appendix F.

For this example, the constraint function is defined as a maximal vertical displacement of 500 mm:

$$g(x_i) = \frac{\max(u_y)}{500} - 1 \leq 0 \tag{E.3}$$

In which u_y is the vertical displacement of all nodes. It should be noted that this constraint is only taken as an illustrative example. In the numerical experiments the constraint is a stress limit.

A stopping criterion is set on 50 successive generations with the same best solution.



Now the optimisation starts with creating 5 designs with for each element a random profile. Figure 100 shows the result.

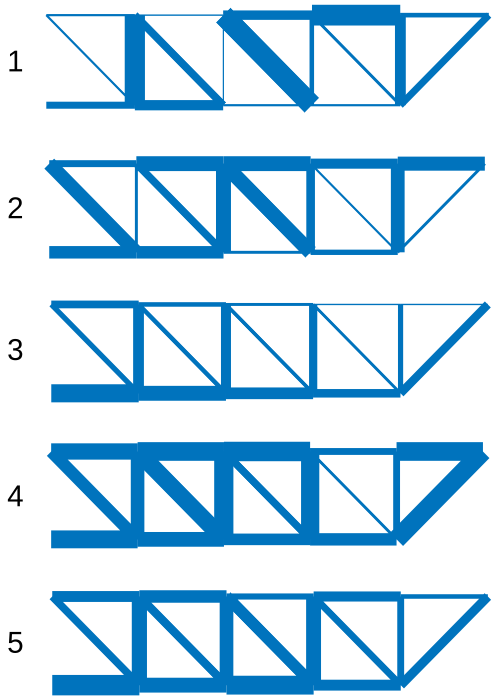
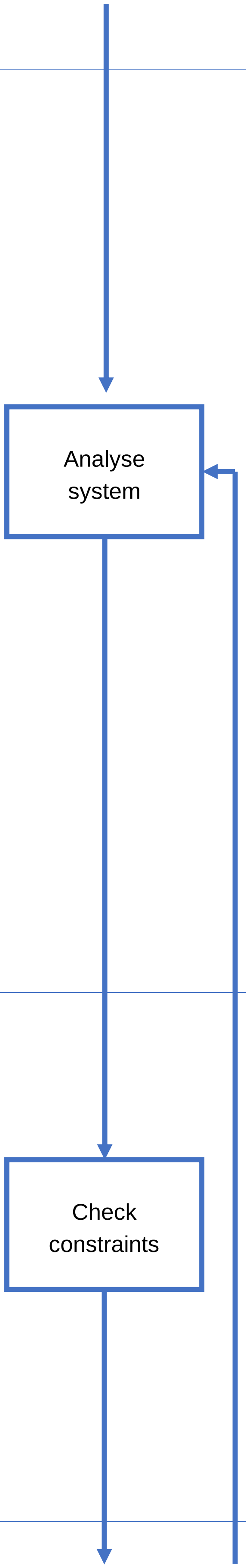


Figure 100 – Creation of 5 random structures



The 5 random structures are evaluated by a structural analysis; all internal forces and displacement of the structure are calculated. Figure 101 shows the results, the red structures shows the displacement ×8. The bar plot shows the internal forces.

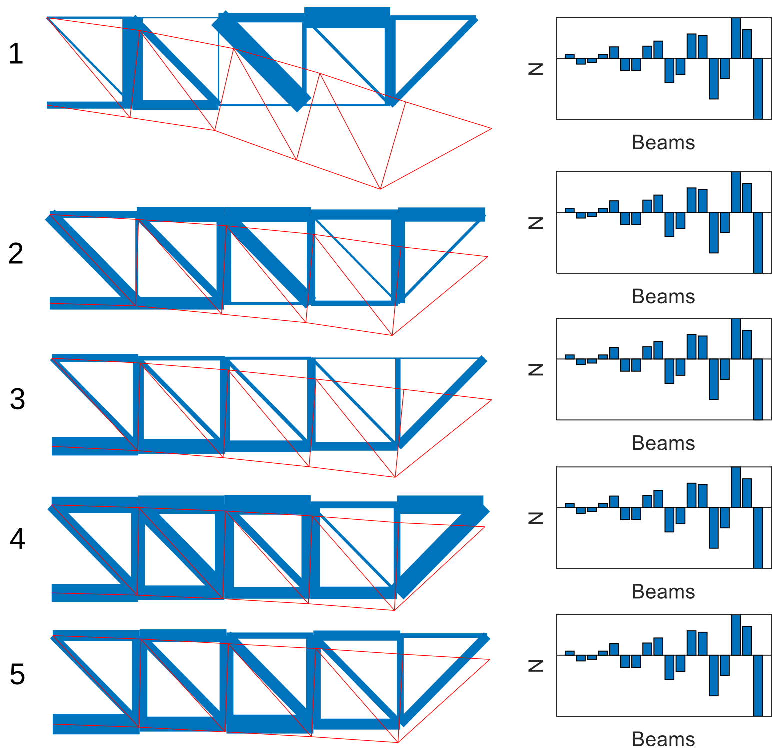
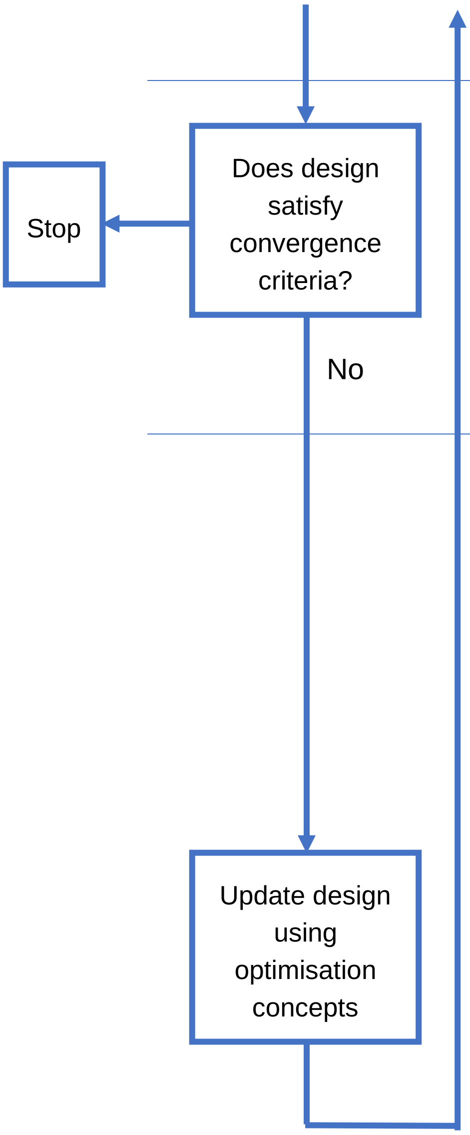


Figure 101 – Evaluate displacement and internal forces

Now the design options are checked on the constraint and objective function, Table 21 shows the results. Design 1 turns out to be infeasible as the displacement is too high; the constraint function is positive. Design 3 is the best design in this generation, as the weight is the lowest and it is feasible.

Table 21 – Constraint and objective function evaluation

Design	Weight (kg)	Maximum Displacement (mm)	Constraint function
1	2024	1006	1.012
2	2600	377	-0.246
3	1856	375	-0.250
4	3849	197	-0.606
5	3023	214	-0.572



During each iteration, the convergence criterion is checked. As this is the first generation, this criterion is not met.

Now the process of reproduction starts. By random choice, the 2nd design is mutated, while the 4th and 5th design are mixed. The 1st design is dropped because it is infeasible and the 3rd design is regarded as elite design, so is not affected by mutation or crossover. Figure 102 shows the reproduction operators mutating the green beams in the 2nd design and swapping the yellow and green beams from the 4th and 5th design.

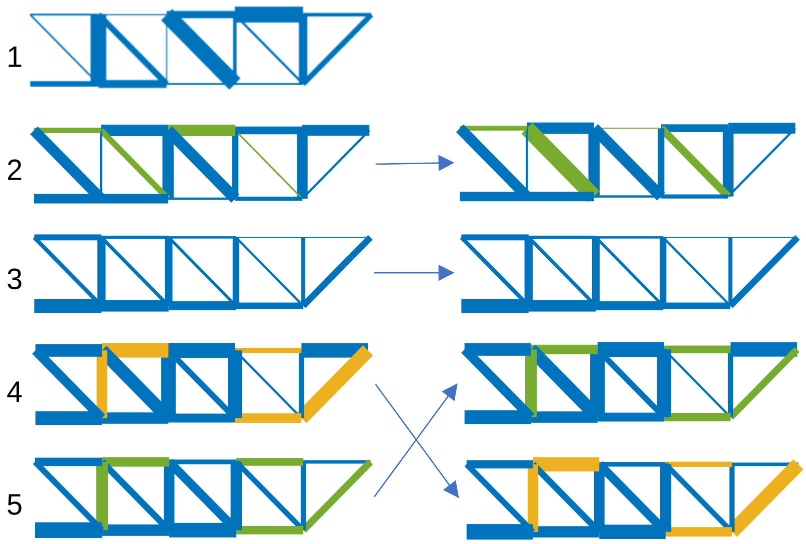


Figure 102 – Reproduction

The designs show on the right of Figure 102 are taken as a new generation and the loop continues until the convergence criterium is met.

In Appendix C.1, the result of the genetic algorithm is shown in an animation.

Appendix F Description benchmark problems

This chapter describes the benchmark problems which were used in this study. All input data is available at: http://doi.org/10.4121/uuid:4e32b29f-6647-4a36-9ea1-8931c88f8864

F.1 18-bar cantilever truss

The 18-bar cantilever truss is a statically determinate structure, which has been analysed in several papers: Salajegheh and Vanderplaats (1993) and Imai and Schmit (1981) considered both size and shape variations with ROT. Lee and Geem (2004) used ROT with only size variations. Kaveh and Zolghdar (2011) used CC. All researchers used four groups. The geometry consists of a cantilever truss beam, loaded on its top nodes.

Figure 103 shows the geometry, member and nodal numbering, dimensions and support conditions of the problem. Table 22 shows the values of the load, density and modulus of elasticity of the problem. Table 23 shows the optimisation properties. The buckling stress limit represents the Euler buckling stress limit in which the radius of gyration is approximated as $r = \sqrt{kA}$.

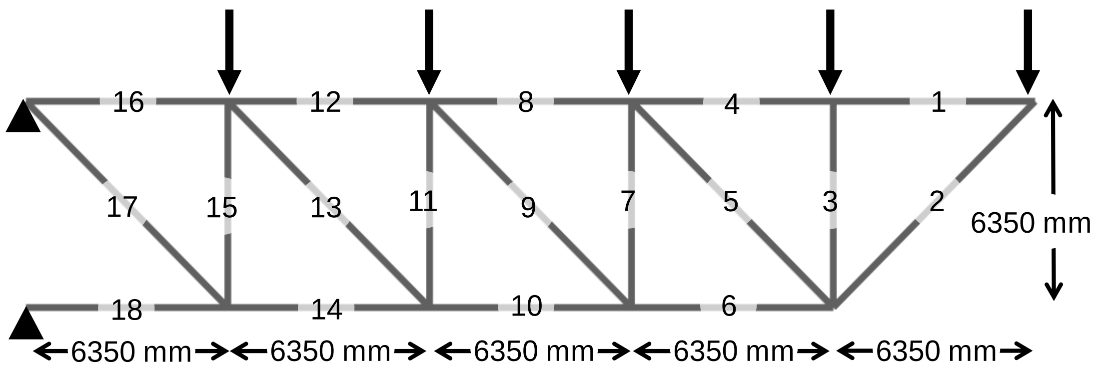


Figure 103 – Geometry 18-bar cantilever truss

Table 22 - Properties problem 18-bar cantilever truss

Load	89 kN
Density	2768 kg/m^3
Modulus of elasticity	6.894·10^4 MPa

Table 23 - Properties optimisation 18-bar cantilever truss

<table>
<tr><td>Options for members</td><td colspan="2">25 cross-sectional areas:
645, 1290, 1935, 2581, 3226, 3871, 4516, 5161, 5806, 6452, 7097, 7742, 8387, 9032, 9677, 10323, 10968, 11613, 12258, 12903, 13548, 14194, 14839, 15484, 16129 mm^2</td></tr>
<tr><td>Allowable stress limit</td><td colspan="2">Yield stress limit: ±172 MPa
And buckling stress limit for members in compression:
$\sigma_i = \frac{kA_iE}{L_i^2}$
with k = 4, buckling constant</td></tr>
<tr><td>Displacement limit</td><td colspan="2">n/a</td></tr>
<tr><td>Number of groups</td><td colspan="2">4</td></tr>
<tr><td rowspan="5">Grouping ROT
(Salajegheh and Vanderplaats 1993; Lee and Geem 2004; Imai and Schmit 2010)</td><td>Group</td><td>Members</td></tr>
<tr><td>1</td><td>1, 4, 8, 12, 16</td></tr>
<tr><td>2</td><td>2, 6, 10, 14, 18</td></tr>
<tr><td>3</td><td>3, 7, 11, 15</td></tr>
<tr><td>4</td><td>5, 9, 13, 17</td></tr>
</table>

Table 24 - Properties cost-optimisation 18-bar cantilever truss

Type of connection	All nodes: Truss connection angles Web spice plate in case of beam splice Web side plate at supports
Number of marking drawings	1
Crane	Small crane
Access equipment	Mobile scaffolding

F.2 65-bar truss beam

The 65-bar truss beam was analysed by Walls and Elvin (2010a) using ROT and UCS. It consists of a statically determinate truss beam supported at both ends.

Figure 104 shows the geometry, dimensions and support conditions of the problem. Table 25 shows the values of the load and modulus of elasticity of the problem. Table 26 shows the optimisation properties.

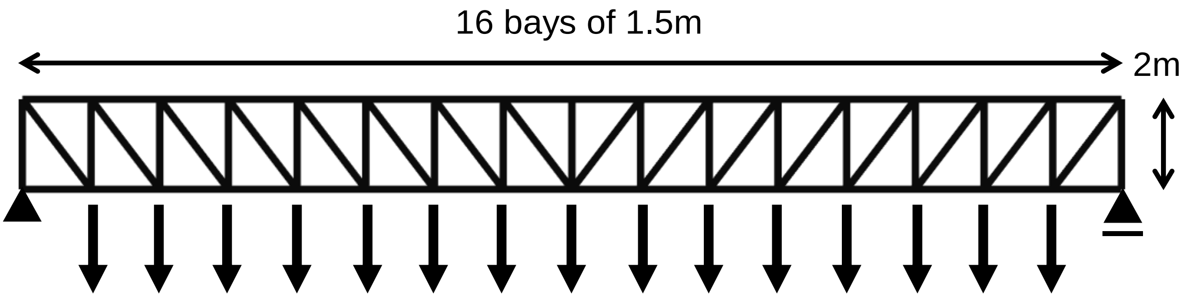


Figure 104 – Geometry 65-bar truss beam

Table 25 – Properties problem 65-bar truss beam

Load case ULS	23.2 kN
Load case SLS	16 kN
Modulus of elasticity	$200 \cdot 10^3$ MPa

Table 26 – Properties optimisation 65-bar truss beam

Options for members	42 standardized equal leg angles (Steel Construction Institute 2020)	
Allowable stress limit	For the ULS: Yield stress limit: ±350 MPa And buckling stress limit for members in compression: $\sigma_i = \frac{r_i^2 \pi^2 E}{L_i^2}$	
Displacement limit	Span/400 = 60 mm at midspan for SLS	
Number of groups	4	
Elementary symmetry grouping	33 symmetric groups	
Grouping ROT (Walls and Elvin 2010a)	Group	Members
	1	Top chords
	2	Bottom chords
	3	Verticals
	4	Diagonals

F.3 72-bar truss tower

The 72-bar truss structure has been a popular benchmark problem in literature used by many different studies. It represents a statically indeterminate five-storey structure.

Figure 105 shows the geometry, dimensions and support conditions of the problem. Table 27 shows the values of the load and modulus of elasticity of the problem. Table 28 shows the optimisation properties. The figure on page F-6 shows the structure in 3D. To enable 3D-viewing, open this document in Adobe Acrobat, enable 3D content in the taskbar and click on the figure.

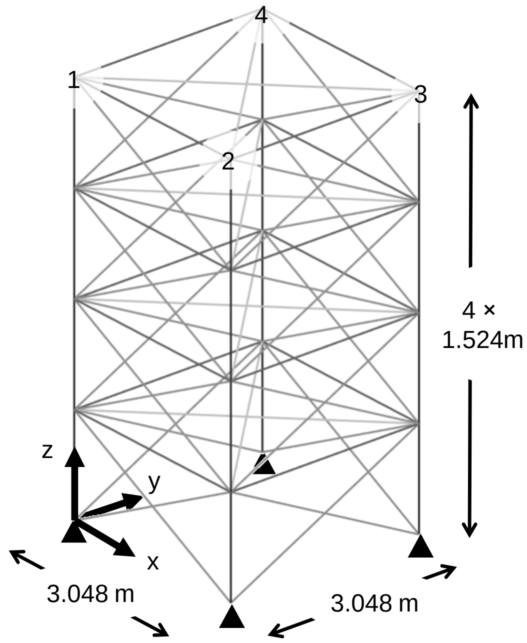


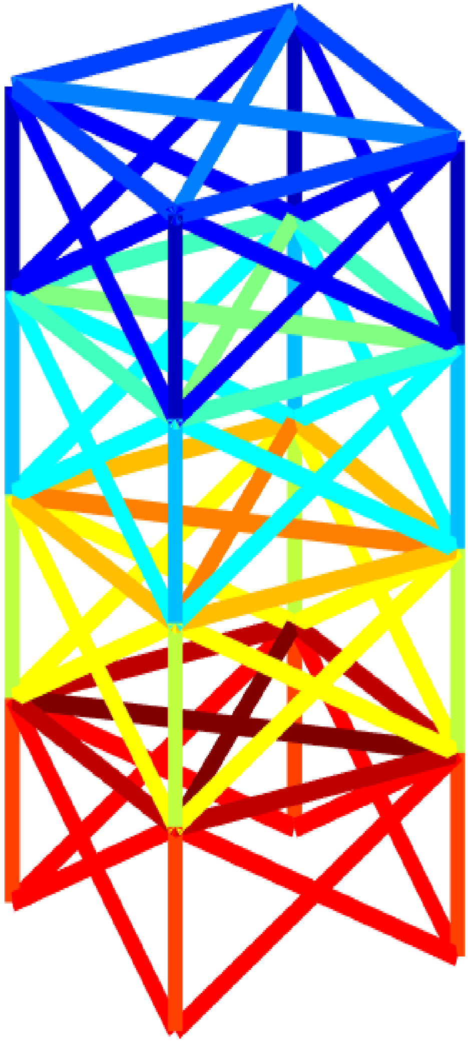
Figure 105 – Geometry 72-bar truss tower

Table 27 - Properties problem 72-bar truss tower

Load case 1 (kN)	Node	F_x	F_y	F_z
	1	22.4	22.4	-22.4
Load case 2 (kN)	Node	F_x	F_y	F_z
	1, 2, 3, 4	0	0	-22.4
Density	2768 kg/m^3			
Modulus of elasticity	6.894·10^4 MPa			

Table 28 - Properties optimisation 72-bar truss tower

Options for member	25 cross-sectional areas: 645, 1290, 1935, 2581, 3226, 3871, 4516, 5161, 5806, 6452, 7097, 7742, 8387, 9032, 9677, 10323, 10968, 11613, 12258, 12903, 13548, 14194, 14839, 15484, 16129 mm^2
Allowable stress limit	Yield stress limit: ±172 MPa And buckling stress limit for members in compression: $\sigma_i = \frac{kA_iE}{L_i^2}$ with k = 4, buckling constant
Displacement limit	6.35 mm at top nodes in all directions
Number of groups	4
Elementary symmetric grouping	16 grouping, 4 groups per storey: All columns, beams, horizontal and vertical bracings per storey (Venkaya et al. 1969) The elementary grouping is shown in the 3D-figure on the next page as well
Grouping ROT	All columns All vertical bracings All horizontal bracings All beams



F.4 112-bar truss dome

Steel dome structures have appeared frequently in literature, of which this statically indeterminate 112-bar truss dome is an example. The complete specification of the dimensions can be found in the paper of Saka (1990).

Figure 107 shows the geometry, dimensions and support conditions of the problem. Table 29 shows the values of the load and modulus of elasticity of the problem. Table 30 shows the optimisation properties. The figure on page F-8 shows the structure in 3D.

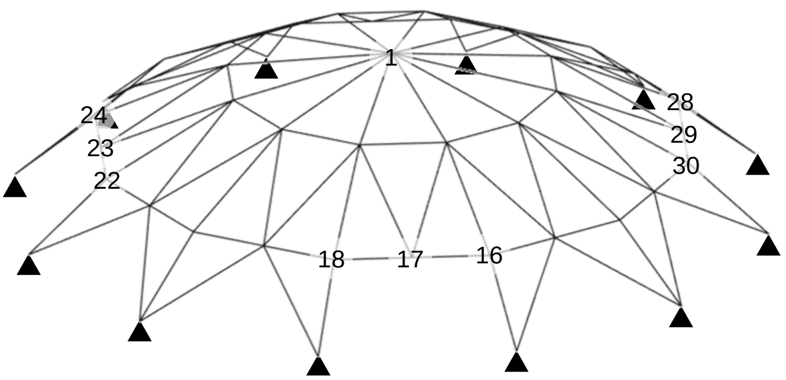


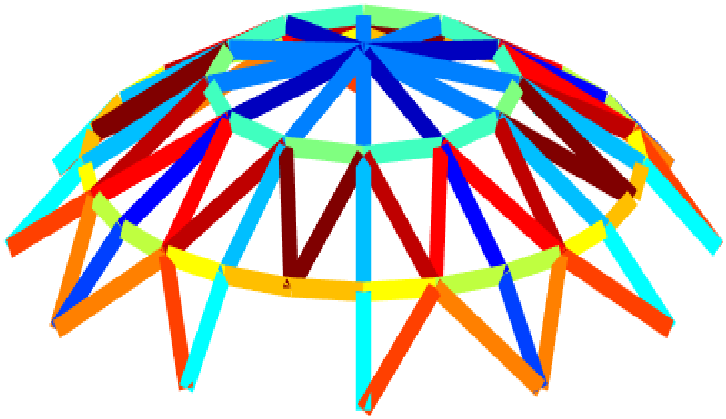
Figure 107 – Geometry 112-bar truss dome

Table 29 - Properties problem 112-bar truss dome

Load (kN)	Node	Downward load
	1	-5
	17, 23, 29, 35	-0.4
	16, 18, 22, 24, 28, 30, 34, 36	-1.2
	Other nodes	-2
Density	Not given	
Modulus of elasticity	$210 \cdot 10^3$ MPa	

Table 30 - Properties optimisation 112-bar truss dome

Options for members	43 pipe sections (Steel Construction Institute 2020)
Allowable stress limit (N/mm^2)	Yield stress limit: ±150 Mpa And buckling stress limit for members in compression: $\sigma_i = \frac{r_i^2 \pi^2 E}{L_i^2}$
Displacement limit (mm)	20 mm in node 1, 17, 23, 29, 35
Number of groups	3
Elementary symmetry grouping	The dome consists of 4 symmetric segments, this allows elementary grouping to 16 groups. The elementary grouping is shown in the 3D-figure on the next page as well
Grouping ROT	3 groups: All beams which run directly from a support to node 1 All horizontal beams All diagonal bracings



F.5 160-bar truss tower

The 160-bar truss transmission tower has appeared frequently in literature. This example has 8 load cases, considering selfweight, wind load, snapping of different cables and end tower conditions. The complete specification of the dimensions can be found in the paper of Groenwold and Sander (1997).

Figure 109 shows the geometry, nodal numbering of the relevant nodes and support conditions of the problem. Table 31 shows the values of the load and modulus of elasticity of the problem. Table 32 shows the optimisation properties. The figure on page F-12 shows the structure in 3D.

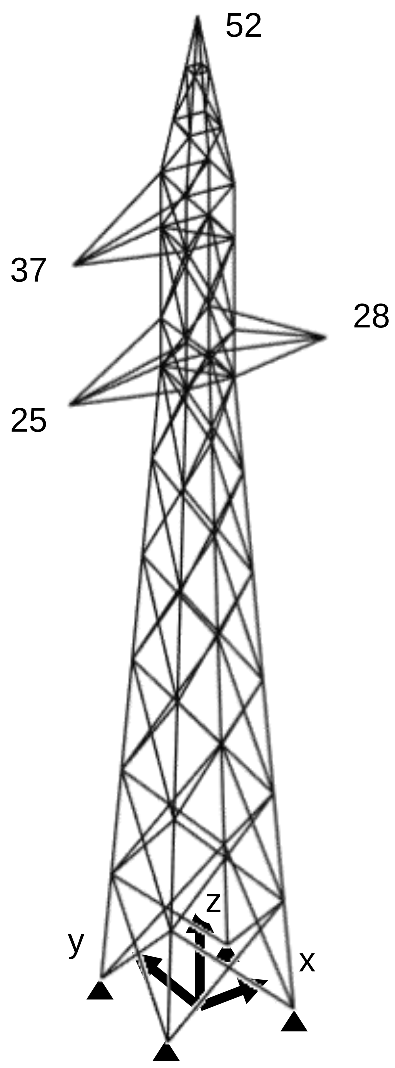


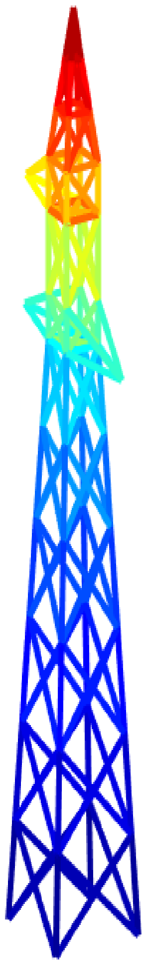
Figure 109 – Geometry 160-bar truss tower

Table 31 - Properties problem 160-bar truss tower

	Node	F_x	F_y	F_z
Load case 1 (N)	52	-8515	0	-4817
	37	-9771	0	-5356
	25	-10703	0	-5356
	28	-10703	0	-5356
Load case 2 (N)	52	-4836	12213	-3561
	37	-9771	0	-5356
	25	-10703	0	-5356
	28	-10703	0	-5356
Load case 3 (N)	52	-8996	0	-4817
	37	-9329	0	-5356
	25	-9957	0	-5356
	28	-9957	0	-5356
Load case 4 (N)	52	-8996	0	-5356
	37	-5611	12351	-4199
	25	-9957	0	-5356
	28	-9957	0	-5356
Load case 5 (N)	52	-8996	0	-4817
	37	-9329	0	-5356
	25	-9957	0	-5356
	28	-6239	12351	-4199
Load case 6 (N)	52	-8996	0	-4817
	37	-5611	12782	-4199
	25	-9957	0	-5356
	28	-9957	0	-5356
Load case 7 (N)	52	-8996	0	-4817
	37	-9329	0	-5356
	25	-9957	0	-5356
	28	-6239	12782	-4199
Load case 8 (N)	52	-4885	14323	-3561
	37	-9329	0	-5356
	25	-9957	0	-5356
	28	-9957	0	-5356
Density	7850 kg/m^3			
Modulus of elasticity	$201 \cdot 10^3$ MPa			

Table 32 - Properties optimisation 160-bar truss tower

Number of members	160
Options for design variables	42 equal leg angles (Steel Construction Institute 2020)
Allowable stress limit	Yield stress limit: ±147.15 Mpa And buckling stress limit for members in compression: $\sigma_i = \frac{r_i^2 \pi^2 E}{L_i^2}$
Displacement limit (mm)	80 mm at node 25, 26, 37, 52
Number of groups	6
Elementary symmetry grouping	Symmetric grouping into 38 elementary groups: All vertical beams per height All bracings per height, from 5th to 8th floor separate for x and y direction All horizontal bracings per height Bottom beam per outrigger Top beams per outrigger The elementary grouping is shown in the 3D-figure on the next page as well
Grouping ROT	All vertical beams up to 6th floor All vertical & horizontal beams from 7th to 9th floor All vertical & horizontal beams from 10th to 12th floor All bracings up to 6th floor All vertical and horizontal bracings from 7th to 12th floor All beams of the outriggers



F.6 15-bar 3-storey frame

The 15-bar 3-storey frame is a simple unbraced frame structure, used many times in literature.

Figure 111 shows the geometry, member numbering, dimensions and support conditions of the problem. Table 33 shows the loading conditions and modulus of elasticity, while Table 34 shows the other optimisation properties of the problem.

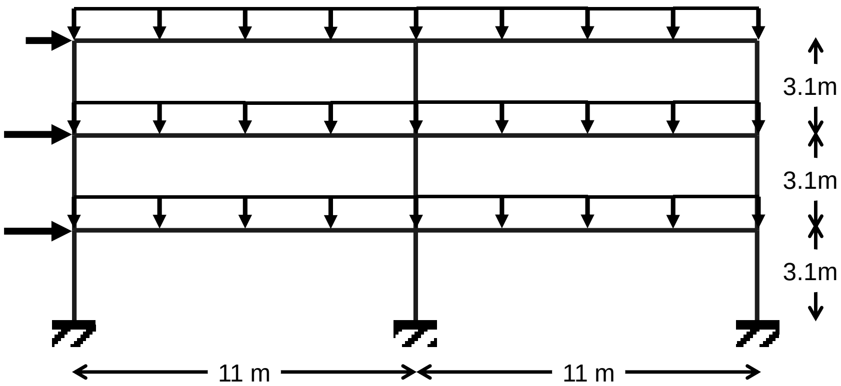


Figure 111 – Geometry 15-bar 3-storey frame

Table 33 - Properties problem 15-bar 3-storey frame

Load	Horizontal load on storey 1 & 2: 22.24 kN Horizontal load on storey 3 11.12 kN Vertical distributed load on beams: 40.86 kN/m
Modulus of elasticity	$200 \cdot 10^3$ MPa

Table 34 - Properties optimisation 15-bar 3-storey frame

Options for design variables	283 W sections for the beams 18 W10 section for columns Properties of W-sections follow from AISC Shapes Database v15.0 (American Institute of Steel Construction 2017)
Number of groups	2 groups for columns, 1 for beams
Elementary symmetry grouping	Symmetric around middle column in 9 groups
Grouping ROT	Outer columns, inner columns and beams

Allowable stress limit

Interaction ratio from AISC-LFRD requirements:

$$\frac{P_{uk}}{\varphi P_{nk}}+\frac{8}{9}\left(\frac{M_{uxk}}{\varphi_b M_{nxk}}+\frac{M_{uyk}}{\varphi_b M_{nyk}}\right)\le 1 \qquad \text{for} \qquad \frac{P_{uk}}{\varphi P_{nk}}\ge 0.2$$

$$\frac{P_{uk}}{2\varphi P_{nk}}+\frac{8}{9}\left(\frac{M_{uxk}}{\varphi_b M_{nxk}}+\frac{M_{uyk}}{\varphi_b M_{nyk}}\right)\le 1 \qquad \text{for} \qquad \frac{P_{uk}}{\varphi P_{nk}}< 0.2$$

For each member k. M_{uyk} = 0 for this 2D-frame

With tensile force in member:

P_{uk} = required axial tensile strength

$P_{nk}=f_y A_{gk}$ = nominal tensile strength

f_y = 248 MPa = specified yield stress

A_{gk} = gross area of member

$\varphi=\varphi_t=0.9$ = tensile strength reduction factor for yielding

With compressive force in member:

P_{uk} = required axial compressive strength

$P_{nk}=f_{cr} A_{gk}$ = nominal compressive strength

$$f_{cr}=\left(0.658^{\lambda_c^2}\right)f_y \qquad \text{for} \qquad \lambda_c\le 1.5$$

$$f_{cr}=\left(\frac{0.877}{\lambda_c^2}\right)f_y \quad \text{for} \qquad \lambda_c>1.5$$

$$\lambda_c=\frac{Kl}{r\pi}\sqrt{\frac{f_y}{E}}$$

K = effective length factor, approximated by (Dumonteil 1992):

$$K_z=\sqrt{\frac{1.6G_A G_B+4\left(G_A+G_B\right)+7.5}{G_A+G_B+7.5}}$$

$G_A=\dfrac{\sum EI_c/L_c}{\sum EI_b/L_b}$ = stiffness ratios of column and girders at two end joints A and B. G=1 for fixed supports

$$K_{y,beam}=\frac{1}{6}$$

$$K_{y,column}=1$$

With z rotation in plane and y rotation out of plane.

l = laterally unbraced length

r = radius of gyration?

$\varphi=\varphi_c=0.85$ = compressive strength reduction factor for yielding

M_{uxk},M_{uyk} = required flexural strength

$\varphi_b=0.90$ = flexural resistance reduction factor

$M_{nxk},M_{nyk}=f_y W_e$ = nominal flexural strength

W_e = elastic section modulus

F.7 117-bar 9-storey frame

The 117-bar 9-storey frame is a 2D, framed structure. It is an adaptation from the 195-bar structure used by Walls and Elvin (2010a).

Figure 112 shows the geometry, dimensions and support conditions of the problem. The distributed load which is shown on one storey, is applied on all storeys. Table 35 shows the loading conditions and modulus of elasticity, while Table 36 shows the other optimisation properties of the problem.

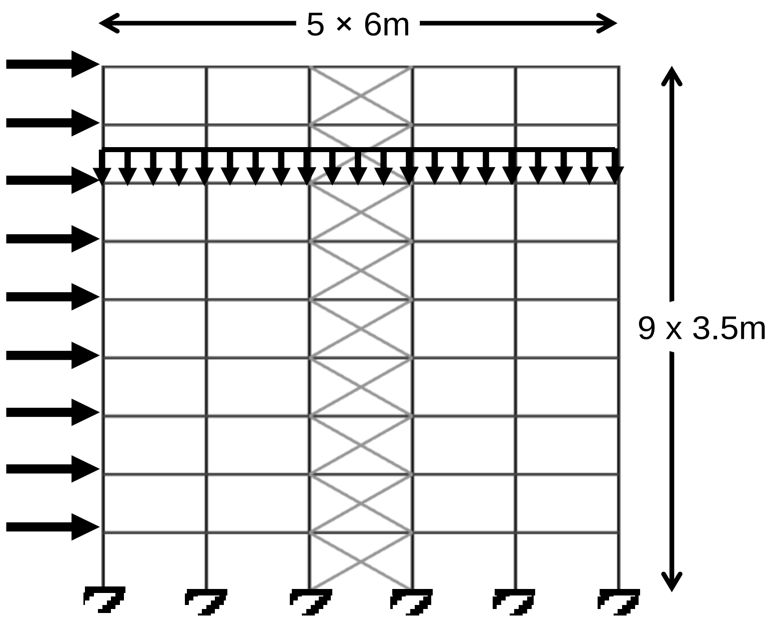


Figure 112 – Geometry 117-bar 9-storey frame

Table 35 - Properties problem 117-bar 9-storey frame

Load	40 kN horizontal load per storey 32 kN/m on every floor
Modulus of elasticity	$210 \cdot 10^3$ MPa

Table 36 – Properties optimisation 117-bar 9-storey frame

Options for members	45 beams: 20 universal beam sections 54 columns: 15 universal column sections 18 bracings: 10 equal leg angle sections (Steel Construction Institute 2020)
Allowable stress limit	Interaction ratio from AISC-LFRD requirements, see constraint function of 15-bar 3-storey frame in Appendix F.6 Yield stress limit of 350 MPa With effective length factor $K_{z,beam} = K_{z,column} = \sqrt{\frac{1.6G_A G_B + 4(G_A + G_B) + 7.5}{G_A + G_B + 7.5}}$ In which for the stiffness ratios bracings are not taken into account $K_{z,bracing} = \frac{1}{6}$ $K_{y,beam} = K_{y,bracing} = \frac{1}{6}$ $K_{y,column} = 1$
Displacement limit	Interstorey drift of 9 mm
Numbers of groups	Beams: 1 Columns: 3 Bracings: 3
Elementary symmetry grouping	Symmetric elementary grouping into 36 groups. 27 columns groups and 9 bracing groups All beams are grouped in 1 additional groups, as the load on each floor is the same.
Grouping ROT	Groups for three consecutive stories for bracings, columns.

F.8 147-bar 3-storey frame

The 147-bary 3-storey frame is a realistic 3D medium sized structure, modelled by Kazemzadeh Azad and Hassançebi (2015). It is braced in one direction, but unbraced in the other direction.

Figure 113 shows the geometry, dimensions and support conditions of the problem. Table 37 shows the loading conditions and modulus of elasticity while Table 38 shows the other optimisation properties of the problem. The figure on page F-19 shows the structure in 3D.

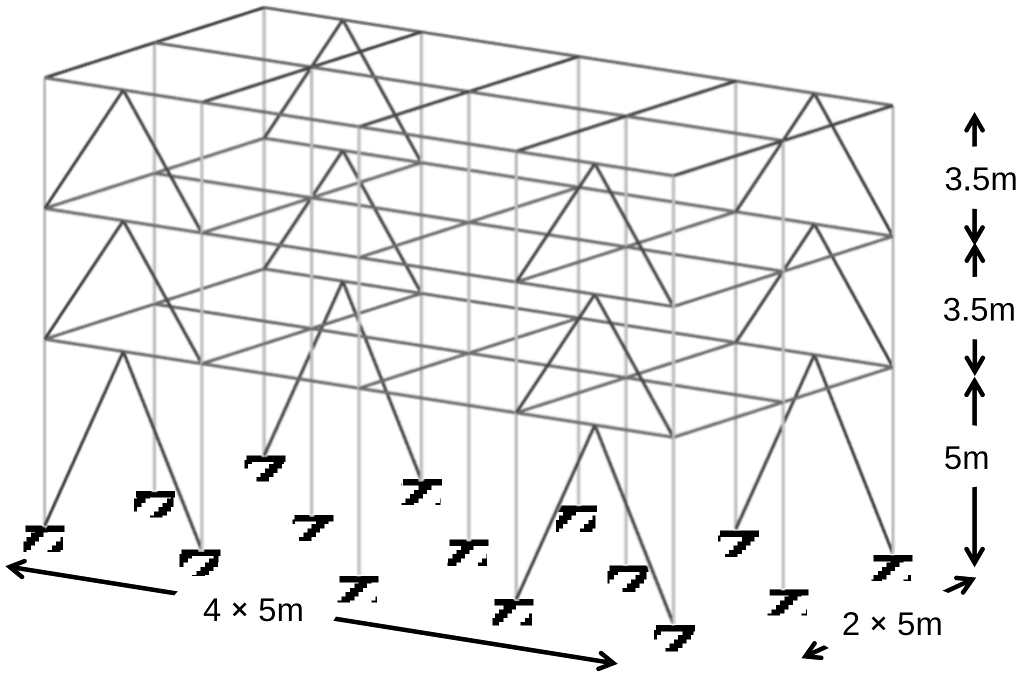


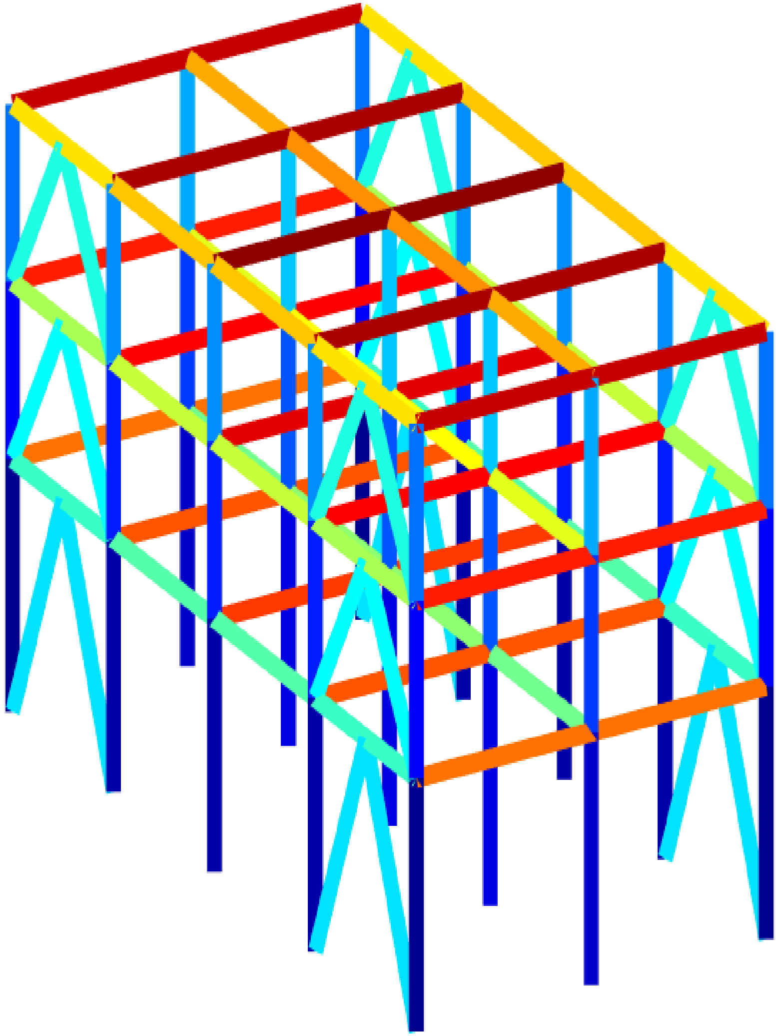
Figure 113 – Geometry 147-bar 3-storey frame

Table 37 - Properties problem 147-bar 3-storey frame

Beam orientation	Columns have strong axis in unbraced direction. Beams have strong axis in horizontal direction. Bracings have strong axis in vertical direction
Load	Load combination: 1.2D + 1.6L + 1W D = dead load = 12 kN/m on floor and 7 kN/m on roof L = live load = 20 kN/m on floor and 15 kN/m on roof W = Wind load = 15 kN on all nodes of short and long facade
Modulus of elasticity	$200 \cdot 10^3$ MPa

Table 38 - Properties optimisation 147-bar 3-storey frame

Options for members	Selection of 25 AISC W profiles Properties of W-sections follow from AISC Shapes Database v15.0 (American Institute of Steel Construction 2017)
Allowable stress limit	Interaction ratio from AISC-LFRD requirements, see constraint function of 15-bar 3-storey frame in Appendix F.6 Yield stress limit of 248.2 MPa With effective length factor $K_{z,column} = \sqrt{\frac{1.6G_A G_B + 4(G_A + G_B) + 7.5}{G_A + G_B + 7.5}}$ In which for the stiffness ratios bracings are not taken into account $K_{z,bracing} = K_{z,beam} = 1$ $K_{y,column} = 1$
Displacement limit	30 mm at top storey in two directions Storey height / 400 for inter-storey drift in two directions
Number of groups	6
Elementary symmetry grouping	Symmetric grouping into 36 groups, 12 column groups, 21 beam groups and 3 bracing groups. The elementary grouping is shown in the 3D-figure on the next page as well
Grouping ROT	Corner and both sides columns are grouped in first storey and upper two storeys. Inner columns are grouped are grouped in first storey and upper two storeys. All beams and bracings are grouped



F.9 Feyenoord stadium

This 3D-truss beam is a statically determinate, used in the design for the roof of the Feyenoord stadium (Kraaijenbrink et al. 2019)

Figure 115 shows the geometry and dimensions of the problem. Table 39 shows the loading conditions and modulus of elasticity while Table 40 shows the optimisation properties of the problem. The figure on page F-22 shows the structure in 3D.

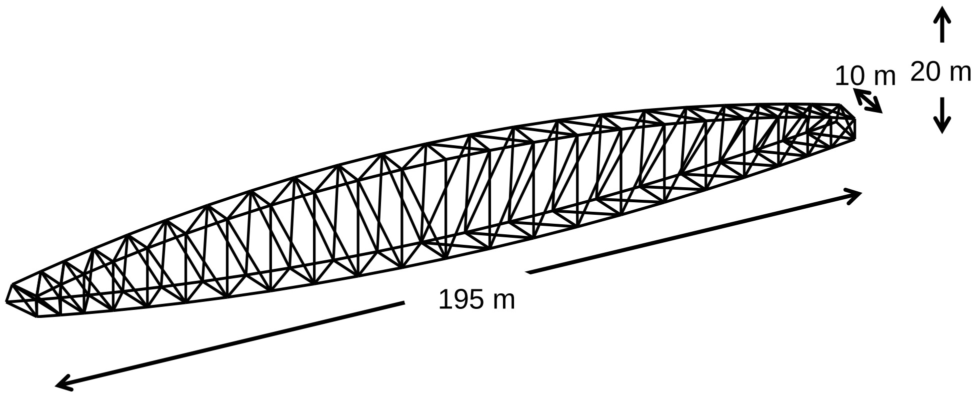


Figure 115 – Geometry roof beam Feyenoord stadium

Table 39 - Properties problem roof beam Feyenoord stadium

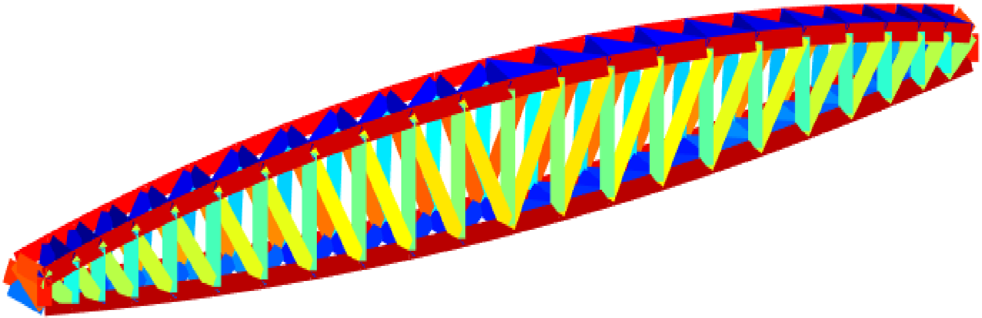
Load	The loads are described in detail in the report from Royal HaskoningDHV (2019). In all load cases the movable roof is closed. 1. Self-weight according to actual design 2. Secondary steel load 3. Architectural finishing load 4. Solar panels load 5. Event load of situation 1 from DO September 2019 6. Up- and downward wind load 7. Up- and downward internal pressure 8. Wind suction and pressure in longitudinal direction	
Load cases	ULS-1	Permanent downwards load × 1.65 Variable downwards load × 1.32 Horizontal wind in positive longitudinal direction
	ULS-2	Permanent downwards load × 1.65 Variable downwards load × 1.32 Horizontal wind in negative longitudinal direction
	ULS-3	Permanent downwards load × 0.9 Variable downwards load × 1.65 Horizontal wind in positive longitudinal direction
	ULS-4	Permanent downwards load × 0.9 Variable downwards load × 1.65 Horizontal wind in negative longitudinal direction
	SLS	Upwards wind and pressure on movable roof ×1
Modulus of elasticity	$210 \cdot 10^3$ MPa	
Support	Support of 4 outer nodes in the bottom trusses. The nodes on the outside bottom truss are supported in z-direction. One of the nodes on the inside bottom truss is supported in all directions; the other one is supported in z- and x-direction.	

Table 40 – Properties optimisation roof beam Feyenoord stadium

Options for members	A selection of 50 German RO-profiles
Allowable stress limit	Yield stress limit: ±460 MPa And buckling stress limit for members in compression: $\sigma_i = \frac{r_i^2 \pi^2 E}{L_i^2}$
Displacement limit	650 mm
Number of groups	21
Elementary symmetry grouping	Symmetric grouping and bundling of members in adjacent groups has been applied. Furthermore, the curved continuous bars are treated as one bar. The corresponding linking is shown in the figure on the next page.
Grouping ROT	1. Horizontal bars in the top plane, except for the outer one 2. Middle 9 horizontal bars and 8 intermediate diagonals in bottom plane and all diagonals in top plane 3. Middle 7 vertical bars in outside plane 4. 6 diagonal bars besides middle 8 diagonals in outside plane 5. 6 vertical bars besides middle 7 verticals in outside plane 6. Middle 7 verticals bars in inside plane 7. Middle 8 diagonals in inside plane 8. 6 vertical bars besides middle 7 verticals in inside plane 9. 6 diagonal bars besides middle 8 diagonals in inside plane 10. Both outer diagonals in outside plane 11. All outer verticals 12. Upper curved continuous bar in outside plane 13. Lower curved continuous bar in inside plane 14. Upper curved continuous bar in inside plane 15. All horizontal and diagonals in bottom plane except for middle 9 horizontal bars and 8 intermediate diagonals. 16. Diagonals and upper horizontals in end planes 17. Both second outer verticals in outside plane 18. Both second outside verticals in inside plane 19. Outer diagonals in outside plane 20. Bottom curved continuous bar in outside plane 21. Middle 8 diagonals in outside plane

Table 41 - Properties cost-optimisation 18-bar cantilever truss

Type of connection	All nodes: double truss connection CHS
Number of marking drawings	4
Crane	Big crane
Access equipment	Booms



Appendix G Cost model steel structures

Because incorporating all possible costs factors in a cost function is not possible, I chose to use the models of Watson et al. (1996). In this cost function, four categories are proposed: supply costs, fabrication costs, surface treatment costs and erections costs. The costs provided by Watson are in dollars, which are converted to euros which a 1-1 exchange rate. Furthermore, a manhour rate of €40,- per hour is adopted, including overhead costs and margins. According to the original research, this cost function is valid for medium-sized steel project in which the costs exceed €150.000,-. In this chapter, the costs in each of the four categories are described.

G.1 Steel supply costs,

Costs are calculated by a unit price per weight. The unit price is taken from the database of Watson et al. (2009).

G.2 Fabrication costs

The fabrication costs are separated in connection, detailing and transporting costs.

Connection costs

The connection costs are dependent on the geometric type of connection. Furthermore, the costs are expressed in manhours. These costs include cutting costs of the beams. Both end connections and connections along a beam are considered, as shown in Table 42a and b.

Table 42a – Manhours per connection
Taken from literature (Watson et al. 1996)

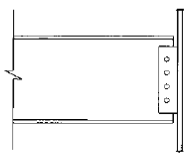
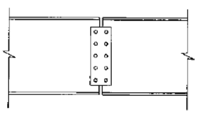
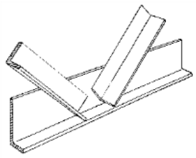
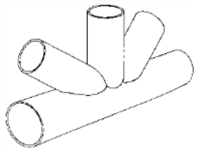
Section mass (kg/m)	Web side plate end connection	Web splice plate
< 60.5	0.8	1.1
60.5 tot 160	1.6	2.4
> 160	3.0	3.5
Diagram		

Table 42b – Manhours per connection
Taken from literature (Watson et al. 1996)

Section mass (kg/m)	Truss connection angles	Truss connection CHS
< 30	0.4	1.6
30 to 60.5	0.7	2.9
60.5 to 120	-	3.3
Diagram		

Detailing costs

Detailing costs include drawing of marking plan and drawings for each unique member. Nowadays, drawing can be exported from engineering software. Nonetheless, these costs are taken into account as this cost function is verified as a whole and is less valid when altered.

The marking plans include a fixed number of drawing per structure. For one marking plan, 20 manhours are needed.

The drawing for unique members includes drawing costs of connections. It is assumed every four hours of fabrication take one hour of drawing.

Transport cost

The transport costs are expressed as cost per member, as shown in Table 43. These costs are based on a typical travel time of 9 hours per load.

Table 43 – Transport costs
Taken from literature (Watson et al. 1996)

Section mass (kg/m)	Costs per member (€)
< 60.5	15
60.5 to 160	56
> 160	225

G.3 Surface treatments costs

In this study, only the application of an alkyd primer was adopted. The costs per unit area are shown in Table 44.

Table 44 – Surface treatment costs
Taken from literature (Watson et al. 1996)

Section mass (kg/m)	Costs per square meter (€)
< 60.5	6
60.5 to 160	5
> 160	4

G.4 Erection costs

Erection costs are calculated by time of unloading and erecting, hire costs of cranes, lifting costs per member, hire costs of access equipment and labour costs, as shown in Table 45 and Table 46. It is assumed that three men are working at all times. Heavy members have increased costs because of the complexity of connecting these elements to other elements.

Table 45 – Time for unloading and erecting per member
Taken from literature (Watson et al. 1996)

Section mass (kg/m)	**Time unload & erect per member (minutes)**
< 60.5	20
60.5 tot 160	20
> 160	24

Table 46 – Costs cranes and access equipment
Taken from literature (Watson et al. 1996)

	Cranes		**Access equipment**		
	Small crane	Big crane	2 × mobile scaffolding	2 × scissor lifts	2 × booms
Capacity	16 t	23 t	6 m	12 m	18 m
Hire costs per hour (€)	85	110	4	14	50
Section mass (kg/m)					
< 60.5 (€)	28	37	3	20	33
60.5 tot 160 (€)	28	37	3	20	33
> 160(€)	34	44	3	24	40