

Master's Thesis on

**Methods for Decreasing the Total
Solution Time of Linear Buckling Finite
Element Analyses**

by
Allan Leth Frank
&
Kasper Skindhøj Larsen

Department of Mechanical and Manufacturing Engineering



Title:

Methods for Decreasing the Total Solution
Time of Linear Buckling Finite Element
Analyses

Theme:

Master's Thesis

Project period:

1st of February 2012
1st of June 2012

Project group:

Group 13.078

Members:

Allan Leth Frank
Kasper Skindhøj Larsen

Supervisor:

Esben Lindgaard

No. printed Copies: 4

No. of Pages: 75

No. of Appendix Pages: 2

Total no. of pages: 92

Completed: 1st of June 2012

Preface

This master thesis is written by the group 13.078, at Aalborg university, Department of Mechanical & Manufacturing Engineering. The thesis at hand has been carried out from February 2012 to June 2012. The aim of this thesis is to develop a faster method for linear buckling analysis of structures, subjected to a large amount of load combinations

The thesis consists of a main report, with additional information given in appendices. A CD is included, where a electronic copy of the report can be found, along with test data. The MATLAB file used for the tests, can also be found on the CD.

The programs used for this thesis are:

- MATLAB R2010b and R2012a
- Inkscape
- ANSYS 13.0
- L^AT_EX
- Pedal x86

Citations are done in accordance to the Harvard style. In this style the last name of the author(s) and year of publication is written in square brackets. Figures, tabels and equations ect. are labeled according to the chapter in which they are presented.

In this thesis, when the word "load configuration" is used, it means a way that a force is applied to the structure. When the word "load combination" is used, it means a combination of the magnitudes of the load configurations.

Abstract

This master thesis is devoted to developing methods for decreasing the total solution time of linear buckling analyses, with a large number of load combinations. Four methods for decreasing the total solution time is presented, along with performance studies of the methods, highlighting the methods pros and cons along with limitations.

The thesis starts by giving an introduction to the method that is presently used for performing linear buckling analyses of structures with a high amount of load combinations. The introduction includes a flowchart of the solution procedure when calculating the buckling loads by the finite element method. After the presentation of the solution procedure, initial studies are conducted to determine the time consumption of some of the steps in the solution procedure. These studies are conducted in order to determine which steps in the solution procedure has the best potential for decreasing the solution time.

Based on the initial studies, methods for decreasing the total solution time are proposed. The possible methods are shortly explained, to give an understanding of the main idea in each proposed method. Next the methods for further study are selected, based on their potential to decrease the solution time and if it is possible to implement the method in a stand alone program, or as add-on to existing commercial software. After the selection of methods, the geometry, used in the performance studies is presented, and the modeling of the structure is described.

Theory governing the linear buckling is given. The distinction between bifurcation buckling and limit point buckling is outlined, and the stability criterion is derived from the total potential energy of a system. Furthermore, the calculation of buckling loads by the finite element method is explained. Ending the theory chapter is a presentation of the limitations concerning the linear buckling analysis.

The solution methods chosen earlier is presented, and explained further. The solution procedure of each method is explained in a flowchart and the theory governing the features used in the method is explained. The performance studies is explained and the results are commented and evaluated. One solution method reduces the total solution time up to 78%, according to the presently used method. Another solution method reduces the number of load combinations by 74%.

Abstrakt

Dette speciale er dedikeret til at udvikle metoder til at mindske den samlede løsnings tid for lineær bulings analyser, med et højt antal lastkombinationer. Fire metoder til at mindske den samlede løsnings tid præsenteres, sammen med studier af ydeevnen for metoderne, som fremhæver metodernes fordele og ulemper, samt begrænsninger.

Specialet starter med at give en introduktion til den metode, der i øjeblikket bruges til at udføre lineær bulings analyser af strukturer med et højt antal lastkombinationer. Indledningen indeholder et flowchart af løsnings proceduren, som beskriver beregningen af bulings laster med elementmetoden. Efter præsentationen af løsnings proceduren, er indledende undersøgelser udført for at bestemme tidsforbruget af nogle af trinene i løsnings proceduren. Disse undersøgelser udføres for at bestemme hvilke trin i løsnings proceduren, som har det bedste potentiale til at reducere løsnings tiden.

Baseret på de indledende undersøgelser, er fremgangsmåder til reducere af løsnings tiden foreslået. De mulige metoder er kort forklaret, for at give en forståelse af den grundlæggende idé i hver af de foreslåede metoder. Efter dette er metoderne, som ønskes yderligere undersøgt, valgt baseret på deres potentiale til at reducere løsnings tiden, og om det er muligt at implementere metoden i et stand alone program, eller som add-on til eksisterende kommercielle software. Efter udvælgelsen af metoderne, er strukturen, der anvendes ved ydeevne undersøgelser præsenteret, og modellering af strukturen er beskrevet.

Teori vedrørende den lineære bulings analyse er givet. Forskellen mellem bifurkations buling og grænse punkts buling er skitseret, og stabilitets kriteriet er udledt fra den samlede potentielle energi af et system. Endvidere er beregning af buling belastninger ved hjælp af elementmetoden forklaret. Sidst i teori kapitlet er en præsentation af begrænsningerne vedrørende den lineære bulings analyse.

Løsnings metoden valgt tidligere er præsenteret, og forklaret yderligere. Løsnings proceduren for hver metode er vist i et flowchart, og teorien der anvendes i metoderne er forklaret. Ydeevne undersøgelserne er forklaret og resultaterne er kommenteret og vurderet. En løsnings metode reducere den samlede løsnings tid med 78%, i forhold til den nuværende anvendte metode. En anden løsnings metode reducere antallet af last kombinationer med 74%.

Table Of Contents

| | |
|---|-----------|
| Table Of Contents | ix |
| Chapter 1 Introduction | 1 |
| 1.1 Solution Procedure for the Presently used Method | 2 |
| 1.2 Initial Studies of Presently used Solution Procedure | 5 |
| 1.3 Objective and Contents of Master Thesis | 7 |
| Chapter 2 Possible Solution Procedures and Geometry | 9 |
| 2.1 Different Methods for Decreasing the Solution Time of Buckling Analysis . | 9 |
| 2.2 Selection of Methods | 12 |
| 2.3 Geometry used in the Performance Studies | 14 |
| Chapter 3 Theory of Elastic Stability | 17 |
| 3.1 Energy Method | 18 |
| 3.2 Calculation of Buckling Loads by the Finite Element Method | 22 |
| 3.3 Limitations of Linear Buckling Analysis | 23 |
| 3.4 Summary | 24 |
| Chapter 4 Methods of Changing the Solution Procedure | 25 |
| 4.1 Loading Data | 25 |
| 4.2 Benchmark, Standard Solution Procedure. | 27 |
| 4.3 Method 1, Solution Procedure using Superposition Principle on Stress Stiffness Matrix $[K_\sigma]$ | 30 |
| 4.4 Method 2, Solution Procedure using Superposition Principle and Cholesky Factorization. | 35 |
| 4.5 Summary | 37 |
| Chapter 5 Performance Studies of Solution Procedures | 39 |
| 5.1 Setup 1 - Influence from Number of Elements and Number of Layers on Total Solution Time | 40 |
| 5.2 Setup 2 - Effects of Number of Load Combinations and Layers | 44 |
| 5.3 Summary | 47 |
| Chapter 6 Data Fitting of a Stability Boundary | 49 |
| 6.1 Selection of Sampling Points | 51 |

| | | |
|---|--|-----------|
| 6.2 | Data Fitting | 54 |
| 6.3 | Implementation | 56 |
| 6.4 | Performance Study | 57 |
| 6.5 | Summary | 68 |
| Chapter 7 Discussion | | 71 |
| Chapter 8 Conclusion | | 73 |
| Bibliography | | 75 |
| A Results from performance studies | | 77 |
| A.1 | Data from Benchmark, Method 1 and Method 2 | 77 |

Notation

MATHEMATICAL SYMBOLS

| | |
|--------------|------------------------|
| $[\]$ | Rectangular matrix |
| $[\]^T$ | Matrix transpose |
| δ | Variation |
| \mathbb{C} | Complex numbers |
| \mathbb{R} | Real numbers |
| Π | Total potential energy |
| $\{ \}$ | Vector |
| $\{ \}^T$ | Vector tranpose |

LATIN SYMBOLS

| | |
|----------------|--|
| $[B]$ | Strain displacement matrix |
| $[E]$ | Matrix of elastic stiffnesses |
| $[G]$ | Matrix containing derivatives of shape functions |
| $[K]$ | Stiffness matrix |
| $[k^e]$ | Element stiffness matrix |
| $[K_\sigma]$ | Stress stiffness matrix |
| $[k_\sigma^e]$ | Element stress stiffness matrix |
| $[K_L]$ | Initial displacement matrix |
| $[S]$ | Initial stress matrix |
| $\{\Phi\}$ | Nodal surface traction forces |
| $\{D\}$ | Global displacement vector |
| $\{F\}$ | Body forces |

| | |
|-----------|-------------------------------|
| $\{R\}$ | Total load on nodes |
| $\{u\}$ | Nodal displacement vector |
| $\{u_0\}$ | Energy density |
| m | number of load configurations |
| n | number of load combinations |

GREEK SYMBOLS

| | |
|------------|-------------------------|
| ϵ | Strains |
| Λ | Rayleigh quotient |
| λ | Load factor, Eigenvalue |
| σ | Stresses |

Introduction 1

Calculating the stability of structures has always been an important engineering discipline. Especially the calculation of the critical buckling load of a structure has been a subject for study since Leonard Euler in 1744 calculated the critical buckling load for a simply supported column.

Buckling is a phenomenon, where a structure suddenly changes from one equilibrium configuration to another equilibrium configuration.

The calculation of buckling loads of a structure is of great importance, due to the possibility of sudden failure of the structure, if the critical buckling load is reached. Some structures might lose all stability, when the buckling load is reached, which could put people at risk, if a roof or other similar structures lose all stability.

Buckling analysis with multiple load combinations is the problem at focus in this thesis. Performing a buckling analysis for a high number of load combinations for a structure, i.e. wind turbine blades, is a very time consuming task. [Laustsen 2011] establishes a need for a faster method for performing these buckling analysis of wind turbine blades, by stating that it is crucial to find an effective method to obtain a reliable solution based on structural response. A high number of load combinations has to be calculated for the wind turbine blade in order to meet certification requirements given by i.e. [GmbH 2010], which states that: The stability of a structure or parts under tensile or compressive loading has to be verified, at any location on the structure. Furthermore the combination that could cause buckling or a conservative envelope shall be determined and used for the verification of the structure.

It is thus clear that it is important to develop a method that is able to calculate the buckling loads of a wind turbine blade faster, because a high number of load combinations have to be analyzed in order to determine the load combinations that causes buckling. The faster method can however also be beneficial for other structures, the wind turbine blade is only used as an example.

In this thesis, three different approaches to solving the problem concerning the large time needed for calculating buckling analyses for a high number of load combination. First it is shown, for two methods, that by changing the solution procedure for buckling finite

element analysis, the time for calculating a specific number of load combinations, can be reduced significantly according to the presently used solution procedure. The idea of changing the solution procedure was proposed by the supervisor of this thesis, Assistant professor Esben Lindgaard. Secondly we show that a buckling boundary can be created, based on a few number of load combinations, to evaluate the entire range of load combination for buckling. The idea of this stems from one of the group members previous projects, where a similar method was developed.

It should be noted that this thesis does not develop a method specifically for use in the wind turbine industry, but a method that can be used for all structures, where a high number of load combinations exist. The reference to the wind turbine industry is only a mean to establish the context and need for a faster solution method.

The wind turbine industry will here be used to give an example of the difference from a presently used method to the methods developed in this thesis.

According to [Laustsen 2011] the problem of having a high number of load combinations, for which a buckling analysis has to be performed, is overcome by picking the load combinations, with the highest load in total, and by picking assumed critical load combinations based on experience. After the picking of load combinations, the buckling loads are calculated by the finite element method, with the assumption of linearity. With this method of picking there is a risk that the engineer picks the wrong load combinations, or fail to pick load combinations which has an unforeseen effect on the structure.

The methods developed in this thesis are able to calculate a high number of load combinations faster. Consequently a higher number of load combinations can be picked, hence reducing the risk of not calculating the critical load combinations is reduced.

1.1 Solution Procedure for the Presently used Method

The presently used solution procedure (picking of load combinations) will here be explained further, along with the solution procedure when using the finite element method for the buckling analysis.

According to [Laustsen 2011] a wind turbine blade is subjected to several thousand load combinations during its life time and for each new load combination a buckling analysis has to be performed. It quickly becomes evident that this is an enormous task. Especially if the very time consuming nonlinear analysis has to be performed for each new load combination, the task will become extremely time consuming. The present way of performing these buckling calculations, as described above, is to examine the peak values of the loads, and examine assumed critical load combinations. With the loading over time shown in Figure 1.1, the critical load combinations are picked based on engineering experience.

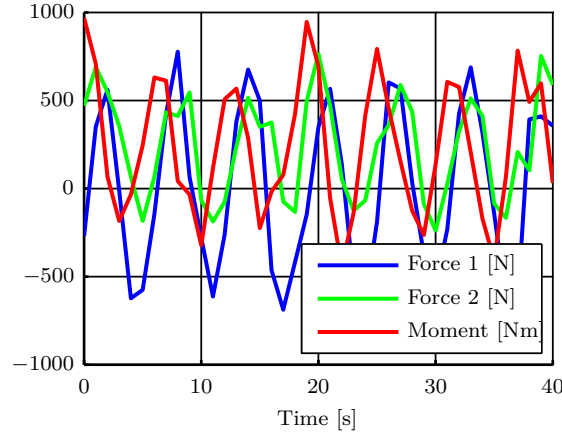


Figure 1.1: Example of arbitrary loading data.

The approach of picking critical load combinations always involves a risk, due to the needed neglect of load combinations, which might have an unforeseen structural response. One of the loads or moments, applied to the structure, might have a critical unforeseen affect on the structure, which can not be predicted by engineering experience. Thus, there is a risk that the picked load combinations are not the most critical ones. Predicting the structural response, based on engineering experience, becomes very hard if the structure that is analyzed is complex. This will thus increasing the risk of picking the wrong load combinations.

1.1.1 Finite Element Method Solution Procedure for Performing Buckling Analysis

Due to the complexity of the structures, buckling is usually calculated by numerical methods such as the finite element method. This is hence the main solution method used throughout this thesis. It is informational to look at the solution procedure for the buckling analysis using finite element methods, to get at better understanding of how and where in the solution procedure improvements could be made. The solution procedure of a commercial finite element program is shown in Figure 1.2 below and is based upon [Felippa 2008].

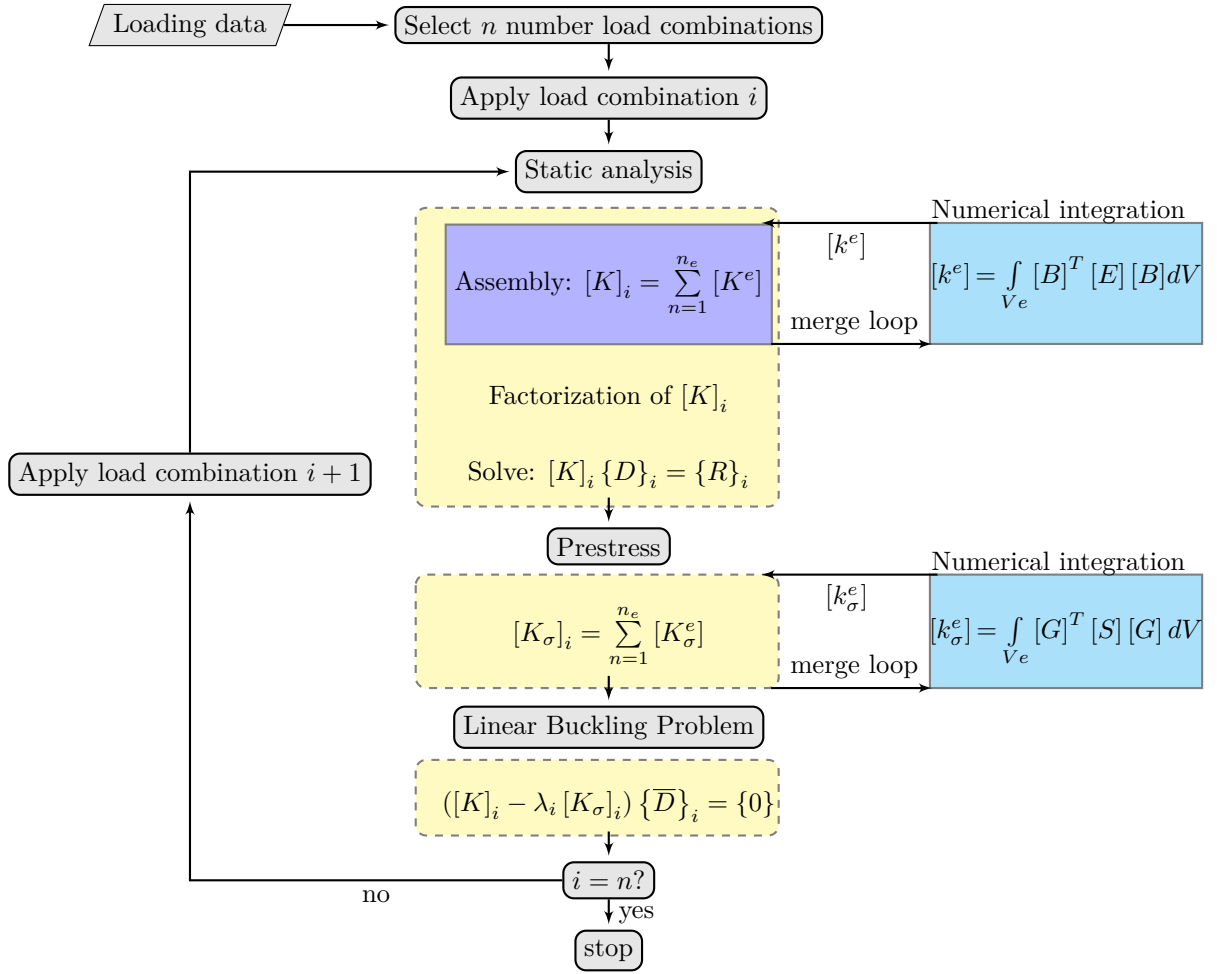


Figure 1.2: Flowchart of a standard procedure for linear buckling analysis with multiple load combinations n , when using the finite element method.

First the number of load combinations n is chosen (picked) from the loading data, then the load combination i is applied to the structure. This concludes the initial phase, and the solution loop can begin.

First thing in the solution loop is to solve the static analysis $[K]_i \{D\}_i = \{R\}_i$. In order to do this, the structure is meshed into a set of finite elements, having finite degrees of freedom. After the meshing, the element stiffness matrix $[k^e]$ is formulated for one element. The formulation of the element stiffness is shown by the equation $[k^e] = \int_{V_e} [B]^T [E] [B] dV^1$, where $[B]$ is the linear strain-displacement matrix, which relates the strains in the element to the nodal displacements. $[E]$ is the constitutive matrix, that defines how the stress-strain relation of the material is formulated. The integration of the element stiffness matrix is performed numerically with i.e. Gauss integration or Simpson integrations, which are the most commonly used in commercial FE programs. The integration effects the ac-

¹This formulation applies to isoparametric elements

curacy and the computational time.

Next the stiffness for each element is loaded into the global stiffness matrix $[K]$. This is done in a loop for all elements, where the elements are expanded and summed to create the global stiffness matrix.

When the stiffness matrix $[K]$ is obtained, it is factorized and finally the equation $[K][D] = \{R\}$ can be solved, giving the nodal displacements, and thus the strains and stresses in the structure.

The next step in the solution loop is to compute the stress stiffness matrix $[K_\sigma]$, which is given by the equation $[k_\sigma^e] = \int_{V_e} [G]^T [S] [G] dV^2$, where the terms in $[G]$ are obtained from shape functions by appropriate differentiation, and $[S]$ contains the initial stresses, obtained from the static analysis. The global stress stiffness matrix is assembled as the global stiffness matrix.

After the stress stiffness matrix is determined, the linear buckling problem:

$$([K]_i - \lambda [K_\sigma]_i) \{\bar{D}\}_i = \{0\} \quad (1.1)$$

is solved, and λ_i and $\{\bar{D}\}_i$ is obtained, where λ is the eigenvalue(load factor) and $\{\bar{D}\}_i$ is the corresponding eigenvector. As seen, this is an eigenvalue problem. The eigenvalue problem is only solved for the lowest eigenvalue, because this value is the only value of interest in a buckling analysis perspective. The calculation of buckling loads from the eigenvalues is explained in Chapter 3. This ends the solutions process.

The steps in the solution procedure continues until the eigenvalues λ_i are calculated for the desired number of load combinations n .

As seen in Figure 1.2 and the description of the solution loop, there is a number of steps in the solution procedure which could be modified. In order to gain further understanding, the time for performing the steps: Static analysis, which is the calculation of the equation $[K]\{D\} = \{R\}$, assembly of $[K]$ and the linear buckling problem, is evaluated in Section 1.2.1. Based on the knowledge of the time it takes to perform the steps, it is evaluated, which step will decrease the total solution time the most.

1.2 Initial Studies of Presently used Solution Procedure

This section describes a convergence study, along with a study of the time spend on the steps mentioned above. The studies are conducted to gain knowledge about the solution procedure and the possibilities for decreasing the total solution time. The studies performed using ANSYS.

²This formulation applies to isoparametric elements

1.2.1 Time Consumption and Convergence

A convergence study of the critical buckling load for a different number of elements is performed on the structure later described in Section 2.3. The convergence plot is shown in Figure 1.3.

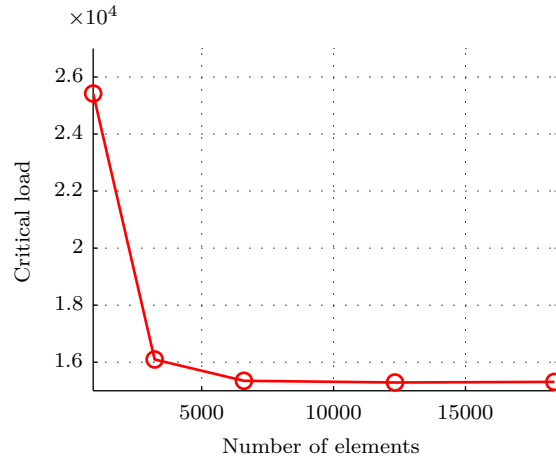


Figure 1.3: Convergence of the critical load, for a number of elements.

The number of elements are going from 884 elements to 18370 elements. With a number of elements of 884 (the first point) the result has an error of 60% according to the converged result. At 3216 elements, the critical load is within approximately 5% of the converged result, and with 6600 elements it is within 0.2% of the converged result.

However, whether or not the buckling load has converged is in the context of this thesis unimportant. The reason for this is that this thesis is concerned with development of a faster method for solution of the buckling, and does only use the buckling load to verify that the different methods predict the same buckling load and for the same load combinations. If the buckling load has converged or not is thus unimportant, as long as the error is the same for all methods.

As stated in Section 1.1 the time it takes ANSYS to calculate different steps in the solution process is beneficial to know in order to effectively decrease the total solution time of the buckling analysis. The cost of the static analysis, assembly of $[K]$, and solution of the eigenvalue problem, is estimated for a different number of elements in Figure 1.4. The results of the estimations are shown graphically in Figure 1.4a.

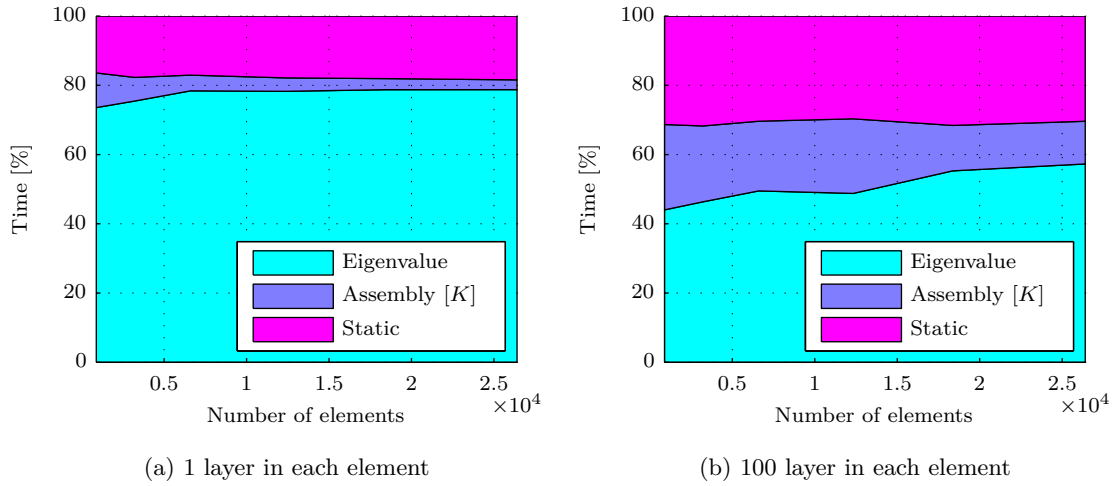


Figure 1.4: Time consumption in percentage for different processes in the buckling analysis given in Section 1.1, for 1 and 100 layers, respectively.

It is noticed that the static analysis and estimation of the eigenvalues, are the most time consuming, where the eigenvalue problem takes up 70-75%, for calculating one eigenvalue, of the total solution time. The static analysis takes up to 30 % of the total solution time, for 100 layers, as seen in Figure 1.4a. It is furthermore seen that the assembly of the stiffness matrix $[K]$ is very time consuming for 100 layers, the reason for this will be clear in Section 2.3. The absence of the time consumption for assembling the stress stiffness matrix $[K_\sigma]$ is caused by a limitation in ANSYS, that prevents for calculating the time spend on this step. It is however suspected that the assembly of $[K_\sigma]$ is included in the time consumption of the eigenvalue problem. This is because if $[K_\sigma]$ shall be exported from ANSYS the eigenvalue problem has to be calculated, in order to do so.

1.3 Objective and Contents of Master Thesis

From the above it is clear that linear calculation of buckling loads of structures subjected to a high amount of load combinations, is a time consuming and very costly affair for the companies designing the structures. In other words, there is a very clear need for a faster method of predicting these buckling loads. From this need a problem statement has been formulated as:

Development of methods for decreasing the total solution time of linear buckling analyses of structures with multiple load combinations.

The aim of this project is to develop one or more methods that can decrease the total solution time of the buckling analysis.

The thesis is limited to the linear buckling analysis, hence the nonlinear buckling analysis

will not be studied, for the purpose of decreasing the total solution time.

The contents of the remainder of this master thesis is outlined here:

Chapter 2 will describe different suggestions on how the solution procedure is improved, along with a selection of the suggested methods for further study. At the end of Chapter 2 the geometry, that will be used in the performance studies of the methods, is presented.

Chapter 3 explains the theory used for calculation of buckling loads by the finite element method. Furthermore the buckling eigenvalue problem used in linear buckling analysis is derived from the total potential energy of a system. Finally the limitations concerning the linear buckling analysis are stated.

Chapter 4 gives a description of the solution procedures selected in Chapter 2. Two methods for improving the total solution time of the buckling analysis is presented. Flowcharts are constructed to give an easy view of the different steps in the solution procedures. Furthermore the theory governing the principles used in the solution procedures are presented for each solution procedure.

Chapter 5 is devoted to the performance studies of the solution procedures. The performance of each solution procedure will be evaluated, based on a number of test setups, where the input variables are varied. Pros and cons of the different methods will be stated, along with possible limitations. Finally a summary of the obtained results is presented.

Chapter 6 describes the method of data fitting a stability boundary. The method is tested for different fitting points. As summary of the results obtained, concludes the chapter.

Chapter 7 discusses the results obtained in the thesis.

Chapter 8 is the conclusion, where it is concluded if the problem stated, has been solved.

Possible Solution Procedures and Geometry 2

2.1 Different Methods for Decreasing the Solution Time of Buckling Analysis

With focus on the solution loop shown in Figure 1.2, different methods for decreasing the solution time of the buckling analysis are suggested. The suggested methods for possible decrease of the solution time are:

- Stability boundary
- Solvers
- Use of $[K_T]'$'s definiteness
- Reusing stiffness matrix $[K]$
- superpositioning of $[K_\sigma]$
- Reducing DOFs

The methods listed above is in the following explained further. After the brief explanation of the methods, the methods are evaluated, and commented on.

2.1.1 Data Fitting of a Stability Boundary

This method is based on using the eigenvalue, which is calculated for a load combination, as a point in a load space. Consequently the result of different load combinations yields different points in the load space. With enough points calculated, an interpolation or a data fitting could be made, which leads to a stability boundary in the load space. The stability boundary divides the load space into a stable region and an unstable region, as illustrated in Figure 2.1 for 2 load configurations. By plotting the load combinations, from the loading data, in the load space, it is possible to determine if the load combinations are outside or inside the stability boundary. The points outside the boundary, are the load combinations to which buckling occurs.

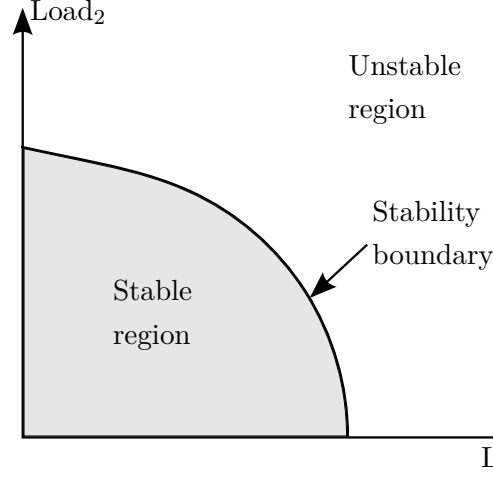


Figure 2.1: Illustration of a stability boundary for Load₁ and Load₂.

If a number of m load configurations is acting on the structure, then the boundary would be produced in a m -dimensional load space. If m is a large number it would require a large amount of calculations to establish the buckling boundary.

2.1.2 Solvers

When dealing with buckling in FE framework it is a matrix eigenvalue problem often for large sparse matrices. The matrix eigenvalue problem is solved in ANSYS by a block Lanczos algorithm. There exist a lot of other numerical methods for solving eigenvalue problems, and thus some might be more efficient than the algorithm used in ANSYS.

2.1.3 Positive Definite / Matrix Analysis

According to [Bažant & Cedoliin 2010] one is able to tell if a structure, subjected to a certain force, will buckle based on the tangent stiffness matrix being positive definite or negative definite. The tangent stiffness matrix is given in Eq. (2.1)

$$[K_T] = [K] + [K_L] + [K_\sigma] \quad (2.1)$$

Where:

- $[K]$ is the global stiffness matrix that defines the stiffness of the structure.
- $[K_L]$ is the initial displacement matrix that accounts for the change in stiffness that comes from the displacements.
- $[K_\sigma]$ is the stress stiffness matrix that accounts for the membrane forces effect on the stiffness.

The structure will not buckle if the tangent stiffness matrix is positive definite, but the structure will buckle if the tangent stiffness is negative definite. However, the only

information obtained is whether or not the structure will buckle, not to which critical force the structure will buckle.

If a Cholesky factorization is used, the matrix condition is determined without calculating the eigenvalue problem in Eq. (1.1). The eigenvalue problem takes up a large amount of the time spend in a buckling analysis, which is showed in Figure 1.4a for 1 and 100 layers, respectively. As mentioned earlier, it is not possible to determine to which load the structure will buckle, but the method can be used to sweep the load space, to find the load combinations that causes buckling. These load combinations can then be studied further, either by linear or nonlinear buckling analysis.

2.1.4 Reusing Stiffness Matrix $[K]$

As described in Section 1.1 the entire solution procedure is run for each new load combination. However, not all the steps in the solution loop needs to be evaluated for each load combination. The stiffness matrix $[K]$, is independent of the load, and does not have to be evaluated for each new load combination. The idea is hence to extract $[K]$ from the solution loop, and store it, and then feed it to the solution loop, when needed. This will save the time it takes to assemble the stiffness matrix $[K]$ for each load case.

2.1.5 Superpositioning of $[K_\sigma]$

Another way to decrease the total solution time, is to use the superposition principle. $[K_\sigma]$ may be superpositioned, so that the different $[K_\sigma]$'s for the load combinations are calculated with a unit load for each load configuration, and then multiplied with a scalar P , and superpositioned with the other load configurations. Take a structure that is subjected to three loads, Load 1, Load 2, and Load 3. The solution loop shown in Figure 1.2 is calculated with unit loads at Load 1, 2, and 3, and then $[K_{\sigma,1}]$, $[K_{\sigma,2}]$, and $[K_{\sigma,3}]$ is obtained. Then a scalar P is multiplied to represent the current magnitude of each load, finally the $[K_\sigma]$'s are multiplied with the scalar P and summed together, i.e. $[K_{\sigma,total}] = P_1 \cdot [K_{\sigma,1}] + P_2 \cdot [K_{\sigma,2}] + P_3 \cdot [K_{\sigma,3}]$.

$[K_{\sigma,total}]$ is hence the stress stiffness matrix that is inputed in the eigenvalue problem.

2.1.6 Reducing Degree of Freedom in the Structure

Model specific methods for improving the efficiency of the buckling analysis is listed below. The reducing the degree of freedom is divided into 2 types.

- Reducing degrees of freedom, using super elements
- Reducing degrees of freedom

Calculation time of the solution procedure, when performing buckling analysis, can be reduced by reducing the degree of freedom in the system. Reduction in d.o.f can be obtained by substructuring.

In substructuring the area of interest, in the structure, is meshed with a fine mesh, and the surrounding structural parts are collapsed into super elements. This will reduce the degree of freedom in the system. A figure that shown the use of super elements is shown in Figure 2.2.

As seen the structure goes from having a fine mesh over the entire structure to having a fine mesh in the region of interest and two super elements in the surrounding geometry.

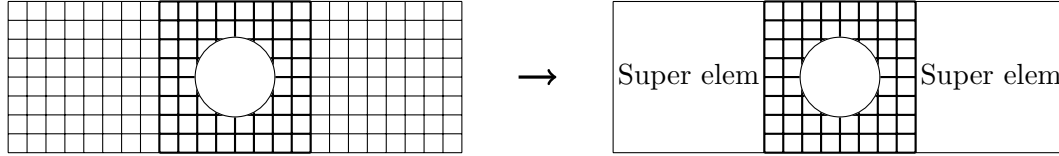


Figure 2.2: From ordinary mesh to structure with super elements.

Another way of reducing the degree of freedom of the system is to mesh the area of interest with a fine mesh and the surrounding with a coarse mesh. Doing this involves a few complications, because the mesh from the two areas are not compatible, the meshes needs to be connected with i.e MPC's. This methods, is much like the method of substructuring using super elements. However the effects from this methods depends on the coarseness of the surrounding mesh. This methods is shown in Figure 2.3

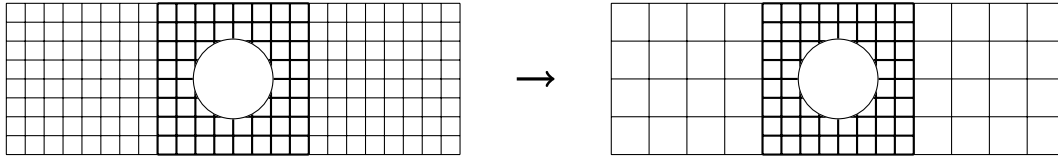


Figure 2.3: From ordinary mesh to a coarse mesh.

A third way of reducing the degree of freedom of the system is simply to cut some of the structure away and the apply cross sectional forces and moments. This is shown in figure Figure 2.4 where the critical areas of the structure is chosen, and buckling is only calculated for these parts of the structure.

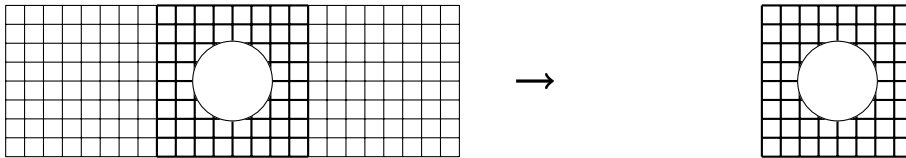


Figure 2.4: Removing parts of the structure to decrease DOFs.

2.2 Selection of Methods

The methods outlined in Section 2.1 is evaluated in this section, and methods for further study is chosen.

At this point, the exact performance of the different methods, with respect to the total

solution time, are unknown, hence the selection is based on the potential for decreasing the total solution time.

In addition to the criteria written in Chapter 1, that states that a faster method for calculation the buckling load is needed, the methods developed have to be generic, meaning that the methods should be applicable in collaboration with commercial finite element software, i.e, ANSYS, Nastran, Abacus, ect. for use in the industry. It could be as a stand alone program or as a add-on to the commercial FE software.

The methods mentioned in Section 2.1.6 Substructuring - Reducing DOFs, will not be studied further. The methods are all very model specific, and given that the aim of this thesis is to develop generic methods for buckling analysis, these methods are not chosen. The reason for this is the methods are not general methods that applies to all sorts of structures and is therefore not easily applied to the industry. Furthermore, using these methods requires a high knowledge to the structure analyzed.

The method of reusing the stiffness matrix $[K]$ is suitable as a generic method, and could be chosen for further study. For structures consisting of many layers i.e. composite structures, the assembly of the stiffness matrix $[K]$ plays a significant role in the solution of the buckling problem, which is seen in Figure 1.4b. This method is not chosen for further study in the thesis, because it is not possible in ANSYS to feed $[K]$ in the software for use in the solution loop. This might however be possible with other software.

A method with potential for decreasing the total solution time is the method of superpositioning $[K_\sigma]$ described in Section 2.1.3.

This method can reduce the number of times the solution procedure, described in Section 1.1, has to be calculated. In this project the superpositioning of $[K_\sigma]$ is performed using MATLAB, and can therefore with advantage be used in combination with the reusing stiffness matrix $[K]$ method, so that the eigenvalue problem in Eq. (1.1) is calculated in MATLAB, instead of in ANSYS. The combination of the two methods is therefore chosen for further study.

The method of determining if the matrix is positive definite or not, could decrease the total solution time. The method will, when using cholesky factorization, result in a faster determination whether or not the structure will buckle, without having to calculate the time consuming eigenvalue problem. As seen from Figure 1.4a and Figure 1.4b the calculation of the eigenvalue problem is the most computational expensive part of the solution procedure. This method is therefore chosen for further study.

The method of data fitting a stability boundary have a potential of reducing the number of buckling analyses needed, for determining the stable and unstable load combinations. This method only requires buckling analyses of the points used to fit or interpolate the boundary. The points needed for a reasonable fit, increases quadratic with the number of load configurations. A high number of load configurations, thus requires a lot of points, which could make this method slower than the presently used method. However, this

method becomes highly efficient, when having a low number of load configurations, and a large number of load combinations.

The method of changing the solver, is not studied further. This is because it is assumed that the solver used by ANSYS is the fastest for these kind of problems, hence the decrease in solution time of this method, will be very small or non at all. In other words, we do not expect to be able to develop a solver that is significantly faster than the existing solvers.

The methods has been evaluated, and based on their potential for decreasing the total solution time of the buckling analysis, the methods are either chosen for further study or disregarded in this thesis.

The methods that is chosen for further study is:

- The method that reuses the stiffness matrix in combination with the superposition of the stress stiffness matrix $[K_\sigma]$
- The method that determined the matrix definiteness
- The method of data fitting a stability boundary

In Chapter 4, the two first methods listed above will be explained in detail, along with the theory governing the methods. Furthermore, the solution procedure of each method is outlined with use of a flowchart for easy comparison of the different methods solution procedures. The last of the listed methods, will be explained in detail in Chapter 6.

2.3 Geometry used in the Performance Studies

The geometry selected for the performance studies in this project is shown in Figure 2.5.

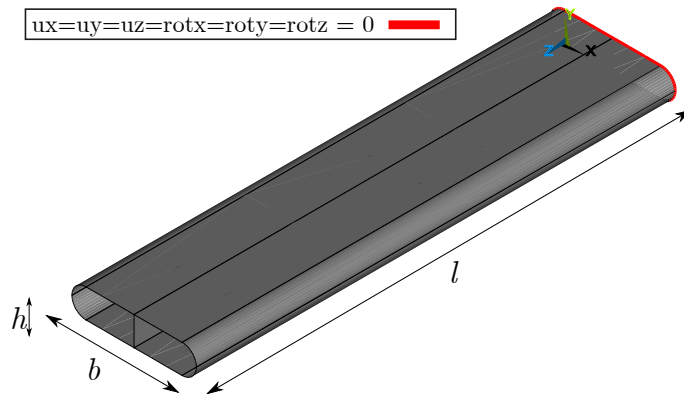


Figure 2.5: Geometry used in the performance studies.

The geometry is a hollow tube with a bulkhead placed in the middle to add stability. The walls of the tube is 1 [mm] thick, and the bulkhead is 1 [mm] thick. The height of the structure is $h = 300$ [mm], the width is $b = 1300$ [mm] and the length is $l = 5000$ [mm]. The structure is fixed at one end and free to move in the other end. Furthermore the free end is rigid, meaning that at $l = 5000$ [mm] the structure is unable to deform in the

xy-plane. The material is selected to be isotropic, to reduce the complexity of the model and the material properties is chosen to be similar to that of structural steel, having a Young's moduli of 200 [GPa], and Poisson's ratio of 0.33. The rigid region insures that local buckling in the vicinity of the point of force application, where the forces will be applied, is avoided. Local buckling is unwanted since the critical buckling value requires a very fine mesh in order to converge. This is because the lack of elements i.e degree of freedom will prevent the structure to represent the modes associated with local buckling.

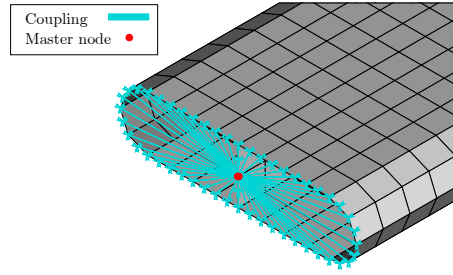


Figure 2.6: Rigid region created with master node.

2.3.1 Modeling of Benchmark Structure

The structure is modeled using ANSYS APDL 13.0.

The structure is meshed with shell elements of the type SHELL281, which is a 8-node element. The element allows for different settings according to the degree of freedom for each node in the elements, which determines the size of the stiffness matrix. For the calculations in this project each node has 6 degrees of freedom, namely: translation in x, y, z-axis, and rotation about the x, y, z-axis.

Another setting that can be controlled for the SHELL281 is the number of integration points through the thickness of each layer. The number of points can be chosen to 1, 3, 5, 7, or 9¹. The number of integration points will have large effect on the computational time, because the integration is performed for each layer in the elements. Going from 1 to 3 integration points will have as large influence hence it should be remembered that there is also integration point in the plan. if we have 4 integration points in the plane, and 1 in the thickness, the total number of integration points equals 4, see Figure 2.7a. But if we on the other hand has 3 integration points through the thickness, the total number of integration points equals 12, see Figure 2.7b. From this it is clear that the number of integration points through the thickness has a large effect on the total solution time.

¹This is for Simpson integration

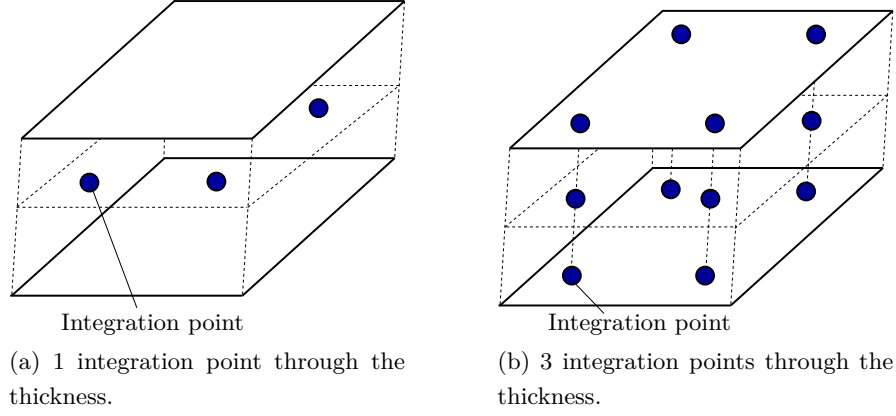


Figure 2.7: Illustration of number of integrations points, when going from 1 to 3 points through the thickness of each layer.

A third parameter is the layer data storage. The data that is stored, are the stresses in the layers and the options are:

- Storage of the data for bottom of the bottom layer and top of the top layer
- Storage of the data for top and bottom for all layers
- Storage of the top, bottom and middle for all layers

The storage of the stresses will have a influence on the computational time, but not as distinct as for the number of integration points.

The settings used in the calculations for data generation for use in this thesis is the standard settings for the element, which is: 3 integration points through the thickness of each layer and storage of the data for the top of the top layer, and bottom of the bottom layer. The data storage is actually not needed for the work in this thesis, but the element option did not allow for no data storage, hence the lowest amount of data is stored.

Theory of Elastic Stability 3

A structure can experience failure in a variety of ways, depending on the material, geometry, load and support conditions. Often two abilities of a structure are examined, namely the strength of the structure i.e. its ability to support a given load without too large stress, or the stiffness of the structure, i.e. its ability to support a given load without too large deformation. However, in this thesis the stability of the structure is examined i.e. its ability to support a given compressive load without experiencing a sudden change in its configuration or shape.

A column under axial compressive loading that is sufficiently slender will fail due to deflection, rather than failure of the material. This phenomenon is called buckling, and describes change of a structures equilibrium configuration to another equilibrium configuration, due to instability. In other words, buckling can be described as the loss of stability of an equilibrium configuration when a load reaches a critical value. Buckling is divided into two categories, namely bifurcation point buckling and limit point buckling, as shown in Figure 3.1 and Figure 3.2.

According to [Wang et al. 2004] the deflection for bifurcation buckling, under compressive loading, changes from one direction to another direction. This could be from axial displacement to lateral deflection, as shown in Figure 3.1. The bifurcation buckling has two deflection paths, one prior to the bifurcation and one after bifurcation. The path prior to the bifurcation is called the primary path, and deflection path after bifurcation is called the secondary path. The secondary path can be either symmetrical or asymmetrical, depending on the structure and loading situation i.e a structure that has a symmetric secondary path could be a column that is axially loaded in compression, as seen in Figure 3.1. Furthermore the structures with a symmetric secondary path are equally likely to buckle in one of the directions. The secondary path may rise or fall. If the path rises, the structure has post buckling strength, but if the path falls, the structure has no post buckling strength.

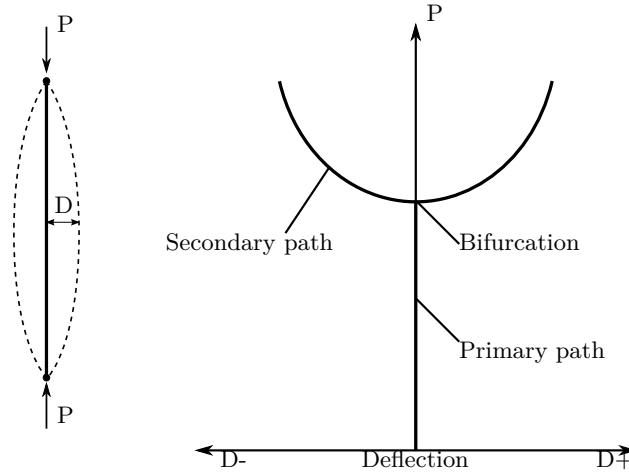


Figure 3.1: Load-deflection curve for an axially loaded column (Bifurcation point buckling).

When having limit point buckling, the buckling load attains the maximum load without any bifurcation, i.e. the structure does only have deflection in one direction. An example of limit point buckling is the snap-through, shown in Figure 3.2. The structure is loaded until it suddenly snaps through and reaches a new stability configuration.

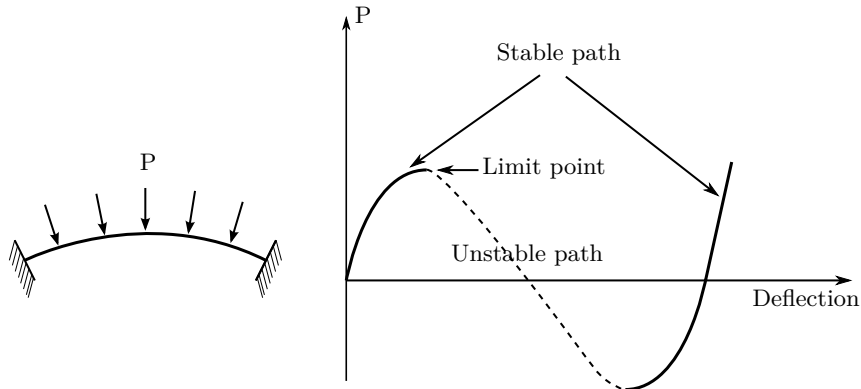


Figure 3.2: Load-deflection curve for a curved column fixed at both ends (Limit point buckling).

The bifurcation point stability problem was first solved by Leonhard Euler in 1744, for a simple supported column, and since the analytical solutions have been derived for plates, shells and cylindrical shells. However, for more complex structures no analytical buckling solution exist. The buckling analyses of complex structures is therefore often performed using the Finite Element Method.

3.1 Energy Method

In this section the stability criterion is found by an energy method, which leads to the eigenvalue problem used to find the critical buckling loads.

Lyapunov's first theorem states, according to [Bažant & Cedoliin 2010] that:

Theorem 1. *"If the potential energy in the equilibrium state is not a minimum and if the absence of a minimum is caused by the second-order terms in the Taylor series expansion of the total potential energy then the system, is unstable."*

First the total potential energy of a conservative system is described. The total potential energy consists of 2 terms, the potential of the loads applied to the system Ω and the stored strain energy of the system U . When neglecting initial stresses and strains, the total potential energy of a system is, according to [Lund & Lindgaard 2011], given in finite element form as

$$\Pi = U - \Omega = \underbrace{\int_V u_0 dV}_U - \underbrace{\int_V \{u\}^T \{F\} dV - \int_S \{u\}^T \{\Phi\} dS - \{D\}^T \{P\}}_{\Omega} \quad (3.1)$$

where $\{u_0\}$ is the energy density, $\{D\}$ is the global displacement vector, $\{u\}$ is the displacement vector, $\{F\}$ body forces, $\{\Phi\}$ is the surface tractions, $\{P\}$ is the global vector with concentrated nodal loads, V is the volume and S is the surface.

The strain energy U can also be written as $\int_V u_0 dV = \int_V \{\epsilon\} \{\sigma\} dV$, where $\{\epsilon\}$ is Green's strain vector and $\{\sigma\}$ is second order Piola-Kirchhoff's stress vector.

It is known from the calculus of variations that a Taylor expansion is used to examine a local extrema. A Taylor expansion of a function f about the position (x_0) is given on the following form

$$f(x_0 + \Delta x) = f(x_0) + \frac{(\Delta x)}{1!} \frac{\partial f}{\partial x_0} + \frac{(\Delta x)^2}{2!} \frac{\partial^2 f}{\partial x_0^2} + \dots \quad (3.2)$$

The Taylor expansion of the total potential energy is expanded around the displacement $\{D\}$, by taking small variations in the displacement vector denoted by $\delta\{D\}$.

$$\begin{aligned} \Pi(\{D\} + \delta\{D\}) &= \Pi(\{D\}) + \frac{\partial \Pi(\{D\})}{\partial \{D\}} (\{D\} + \delta\{D\} - \{D\}) \\ &\quad + \frac{1}{2!} (\{D\} + \delta\{D\} - \{D\})^T \left(\frac{\partial^2 \Pi(\{D\})}{\partial \{D\}^2} \right) (\{D\} + \delta\{D\} - \{D\}) + \dots \end{aligned} \quad (3.3)$$

The change in the total potential energy is found by subtracting the first term $\Pi(\{D\})$ in Eq. (3.3) from the term on the left hand side, and the higher order terms are disregarded because they are very small, for small variations of Π . Thus Eq. (3.3) can be written as

$$\Pi(\delta\{D\}) = \underbrace{\frac{\partial \Pi(\{D\})}{\partial \{D\}} \delta\{D\}}_{\delta^1} + \underbrace{\frac{1}{2!} \delta\{D\}^T \left(\frac{\partial^2 \Pi(\{D\})}{\partial \{D\}^2} \right) \delta\{D\}}_{\delta^2} \quad (3.4)$$

The first statement in Lyapunov's theorem stated earlier, is based on the fact that $\Pi(\delta\{D\})$ has to yield a positive number, for a minimum to exist. In other words a variation in the displacement has to yield a positive energy. Since the derivative of the potential energy is equal to the internal and external forces $\frac{\partial \Pi(\{D\})}{\partial \{D\}} = (\{R^{int}\} - \{R^{ext}\})$, then the first

variation δ^1 can be written as

$$\delta^1 \Pi = \left(\int_V [\bar{B}(\{D\})]^T \{\sigma\} dV - \{R^{ext}\} \right) \delta \{D\} \quad (3.5)$$

where $[\bar{B}(\{D\})]$ is the strain displacement matrix and $\{R^{ext}\}$ is the external load vector.

If the forces are in an equilibrium, the residual forces are equal to zero ($\{R^{int}\} - \{R^{ext}\} = Residual = 0$), then first variation δ^1 (first order term in Eq. (3.4)) is equal to zero.

Since the second derivative is equal to the tangent stiffness $\frac{\partial^2 \Pi(\{D\})}{\partial \{D\}^2} = [K_T]$, then the second variation δ^2 (second order terms) of the Taylor expansion of Π becomes

$$\delta^2 \Pi = \frac{1}{2} \delta \{D\}^T [K_T] \delta \{D\} \quad (3.6)$$

The second statement in Lyapunov's theorem states that if the absent of a minimum is caused only by the second variation the system is unstable, which is due to the fact that the first variation is zero. If the second variation is positive then a minimum exist and the system can be considered stable, but if the second variation is negative then no minimum exist and the system is unstable. This is visualized in Figure 3.3 and Figure 3.4.

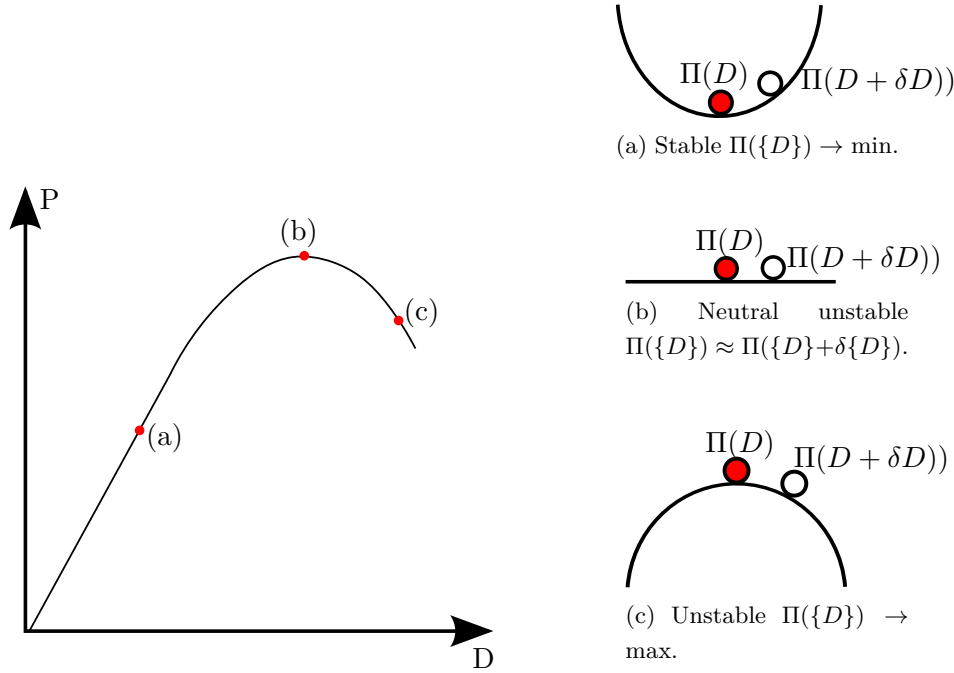


Figure 3.3: Load - displacement path.

Figure 3.4: State of equilibrium.

A change in the the potential energy for 3 different states of equilibrium is seen in Figure 3.4a, Figure 3.4b and Figure 3.4c, and their corresponding local extrema. The result of any variation ($\delta \{D\}$) in the displacement is depending on $[K_T]$, if the resulting potential energy has a strict minimum, $[K_T]$ is said to be positive definite, and negative definite if no minimum is present. In other words, when K_T is positive definite, then any

set of displacements will result in a positive total potential energy as shown in Eq. (3.7)

$$\delta^2\Pi(\{D\}) = \frac{1}{2}\delta\{D\}^T [K_T] \delta\{D\} > 0 \quad (3.7)$$

and negative definite when any set of displacements will result in a negative total potential energy as shown in Eq. (3.8).

$$\delta^2\Pi(\{D\}) < 0 \quad (3.8)$$

If the second variation is equal to zero, the stability is at a limit. To find this limit Eq. (3.6) is set to be equal to zero, and the tangent stiffness is expanded to the stiffness $[K]$ and stress stiffness $[K_\sigma]$, assuming the initial displacement stiffness K_L to be zero, because of small displacements. The stability limit is thus given as

$$\delta\{D\}^T [K] \delta\{D\} + \delta\{D\}^T [K_\sigma] \delta\{D\} = 0 \quad (3.9)$$

According to [Bažant & Cedoliin 2010], the second variation of the total potential energy, ceases to be positive definite, when at a stability limit, hence the first variation of the second variation of the total potential energy is equal to zero. This is written as:

$$\delta(\delta^2\Pi) = 0^1 \quad \text{or} \quad \frac{\partial(\delta^2\Pi)}{\partial(\delta\{D\})} = 0 \quad (3.10)$$

Taking the first variation of Eq. (3.9) and writing it on Rayleigh quotient form gives:

$$\frac{[K]\delta\{D\}}{[K_\sigma]\delta\{D\}} = -\Lambda \quad (3.11)$$

For Eq. (3.11) we have that:

$$\Lambda \begin{cases} > 1, & \text{stable} \\ = 1, & \text{stability limit} \\ < 1, & \text{unstable} \end{cases} \quad (3.12)$$

It is here noticed the Λ in Eq. (3.11) is the Rayleigh quotient. According to Bažant & Cedoliin [2010] we know that: The limit of stability is the first critical load factor λ , thus:

$$\lambda = \min \Lambda \quad (3.13)$$

This means that structure is unstable if the load factor λ is below 1, with a certain reference level of external load $\{R\}_{ref}$ applied.

From Eq. (3.11) we obtain the eigenvalue problem stated in Eq. (1.1), where the smallest root λ defines the smallest level of load where bifurcation occurs:

$$([K] + \lambda [K_\sigma]) \delta\{D\} = 0 \quad (3.14)$$

¹Note that this is not the third variation

3.2 Calculation of Buckling Loads by the Finite Element Method

This section does only explain the theory for linearized buckling analysis, the nonlinear solution of the buckling problem will thus not be discussed. The section is based on [Cook 2001].

When calculating the buckling loads with the finite element method, the eigenvalue problem in Eq. (3.14) is solved, and the smallest eigenvalue is found. The two matrices in the equation is the stiffness and stress stiffness matrix $[K]$, and $[K_\sigma]$, that contains the membrane forces, and the stiffness of the structure, respectively.

The matrix $[K_\sigma]$ in Eq. (3.14) is the matrix that takes the membrane forces and associated membrane stresses into account. The stress stiffness matrix augments the ordinary stiffness matrix $[K]$, and determines if the tangent stiffness matrix $[K_T]$ is positive definite or not. The reason that $[K_\sigma]$ determines if the tangent stiffness matrix is positive definite or not, is because $[K_\sigma]$ is negative definite and when reaching a certain value, the tangent stiffness matrix will become negative definite. In a physical perspective, the stress stiffness matrix reduces the resistance to bending deformation, when the membrane forces are compressive, but increases the resistance to bending deformation, when the forces are tensile, hence the structure will buckle if the forces are compressive, but become more stable if the forces are tensile.

3.2.1 Calculation of Buckling Loads

When the buckling load is calculated the structure is loaded with an arbitrary reference level of external load configuration, $\{R\}_{ref}$ and the linear static analysis is solved to determine the displacements and hence stresses. For stresses associated with the load configuration $\{R\}_{ref}$, the stress stiffness matrix is $\{K_\sigma\}_{ref}$. Because we have assumed linearity, other levels of the reference load configurations can be found by a scalar multiplier γ , i.e.

$$[K_\sigma] = \gamma[K_\sigma]_{ref} \quad (3.15)$$

when

$$\{R\} = \gamma\{R\}_{ref} \quad (3.16)$$

This means that if $[K_\sigma]$ is calculated with a load configuration with the magnitude of 1 [N], $[K_\sigma]$ for a load configuration of the magnitude of 50 [N], can be obtained, simply by multiplying $[K_\sigma]$ for 1 [N] with 50. It should be noticed that this multiplication does only change the magnitude of $[K_\sigma]$, not the distribution of stresses.

When the eigenvalue problem in Eq. (3.14) is solved, the factor λ obtained will be the factor which $[K_\sigma]$ has to be multiplied with in order to get zero.

We have in Eq. (3.15) and Eq. (3.16) established that if the stress stiffness matrix is multiplied by a scalar, the reference load must be multiplied by the same scalar. This means that we can obtain the critical load by multiplying the reference load configuration with λ , as shown below:

$$\{R\}_{cr} = \lambda \{R\}_{ref} \quad (3.17)$$

3.3 Limitations of Linear Buckling Analysis

Effects of imperfections and the limitations regarding linear buckling analysis will be stated here.

As mentioned earlier the method used for the prediction of buckling loads is the linear buckling. By using this method, it is avoiding to perform a full geometrically nonlinear analysis, which can be expensive and time consuming. It is however important to be aware of the limitations, concerning the use of linear buckling analysis. The method can in some cases seriously overestimate the buckling load, hence the limitations of linear buckling will be discussed in the following. The section is based on [Felippa 2007].

There are some main requirements, that has to be fulfilled for correct use of the linear buckling, which will be listed below. After the listening, the different limitations is explained further.

Main requirements:

- Symmetric bifurcation
- Conservative loading
- Linear relation between force and displacement
- Small pre-buckling deformations
- Negligible effects from imperfections

The symmetric bifurcation means that when the structure losses stability, it has to be at a symmetric bifurcation point as shown in Figure 3.1, meaning, that the structure is equally likely to buckle in two directions. This is the case for a bar fixed at on end, and axially loaded at the other end.

Conservative loading means that the loads applied to the structure have to be static, and not dynamic.

The linear relation of force and displacements, basically says the the equation $[K]\{D\} = R$ has to be fulfilled. This means that the displacements have to scale in a linear manner according to the force applied to the structure.

Small buckling deformations, means that the deformations of the structure prior to bifurcation has to be small. Negligible effects from imperfections, means that the structures calculated must not be imperfection sensitive. With imperfection sensitive structures, such as a cylindrical shell, the linear buckling analysis overestimates the buckling load, because a small geometrical imperfection will have a large effect on the buckling load.

The above examples exemplifies that it is important to know if the linear buckling analysis is applicable for the structure that is calculated in order to get good results. The linear buckling analysis may however be used even though the structure does not fulfill all the criteria. However, this requires that the engineer is highly experienced in calculation buckling of the specific structure, so that he is capable of determine whether or not the linear buckling analysis has overestimated the buckling load.

Given that the aim for this thesis is method development, whether the linear buckling analysis is fully applicable to the structure that is calculated in this thesis, or not, is of no concern.

This is because the specific buckling loads for the structure are not used in design or other applications, where it is important that the results are correct. The results are in this thesis only used to compare the different methods, prediction of to which load configurations and load combination the structure will buckle.

3.4 Summary

This chapter has given a explanation of limit point and bifurcation point buckling. Furthermore, the eigenvalue problem used in linear buckling analysis is derived from the total potential energy. Calculation of buckling loads by the finite element method is described, and finally the limitations of the linear buckling analysis are stated. Following this chapter is a chapter that describes the different solution procedures, along with the theory governing the features, used in the solution procedures.

Methods of Changing the Solution Procedure 4

In this chapter, the different methods of changing the solution procedures are described. The different solution procedures, will be presented graphically to illustrate in an easy manner, the difference between the procedures. The theory governing the methods used is described, as well. Furthermore the loading data, used for the calculations is described to give an understanding of why the loading data is constructed as it is.

4.1 Loading Data

In this section the loading data, used in this thesis, is presented along with a figure that shows where the different loads in the loading data are acting on the structure, see Figure 4.1. Furthermore, the loads is divided by their individual buckling loads, and plotted in Figure 4.3, to show that non of the loads, will cause buckling individually.

As seen in Figure 4.1, there are 5 loads acting on the structure. Four at the end, and one located at $(900, 300, 1600 \text{ [mm]})$. The magnitude of the different loads are determined from the loading data, presented in Figure 4.2. From the loading data, load combinations are obtained.

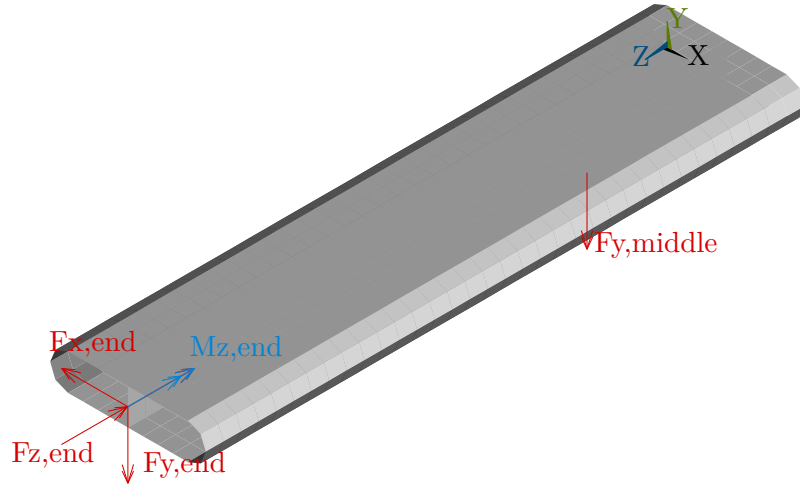


Figure 4.1: Location of the different forces acting on the structure, which magnitude corresponds to the load data given in Figure 4.2.

The load data from which the load combinations are obtained is shown in Figure 4.2.

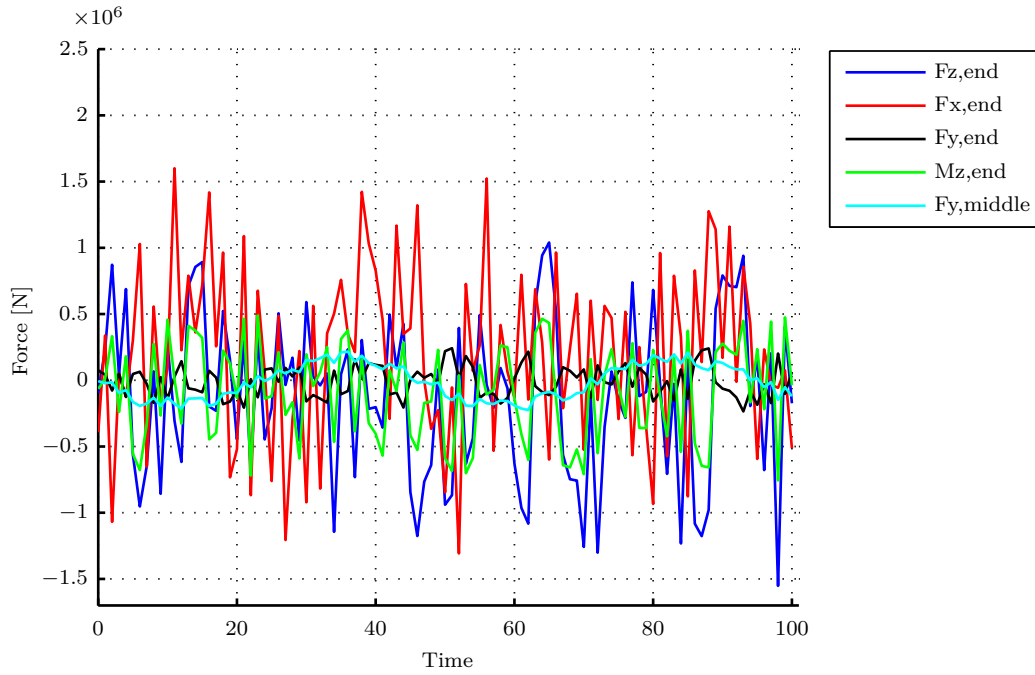


Figure 4.2: Load data, generated by MATLAB for use in the different performance studies.

The load spectrum in Figure 4.2 goes from $(-1550[kN] \text{ to } 1600[kN])$, and the timespan, from 0 - 100. The data is generated in MATLAB, and is meant to represent some fluctuating load scenario. The load combinations that are used in the performance studies in Chapter 5 are extracted for time step of the time span from 1-100, so the maximum number of load combinations is 100. The data is constructed such that 17, 19 and 19 of the 100 load combinations will buckle, for 884, 3216 and 6600 elements, respectively.

The load data is constructed such that there is no single force that exceeds its individual buckling load. This ensures that if the structure buckles, it will be the effect of the different forces influence on each other, rather than a single large force. In Figure 4.3 the forces in the load data is divided by their individual buckling load, to show that non of the individual loads, at any point are larger than their individual buckling load.

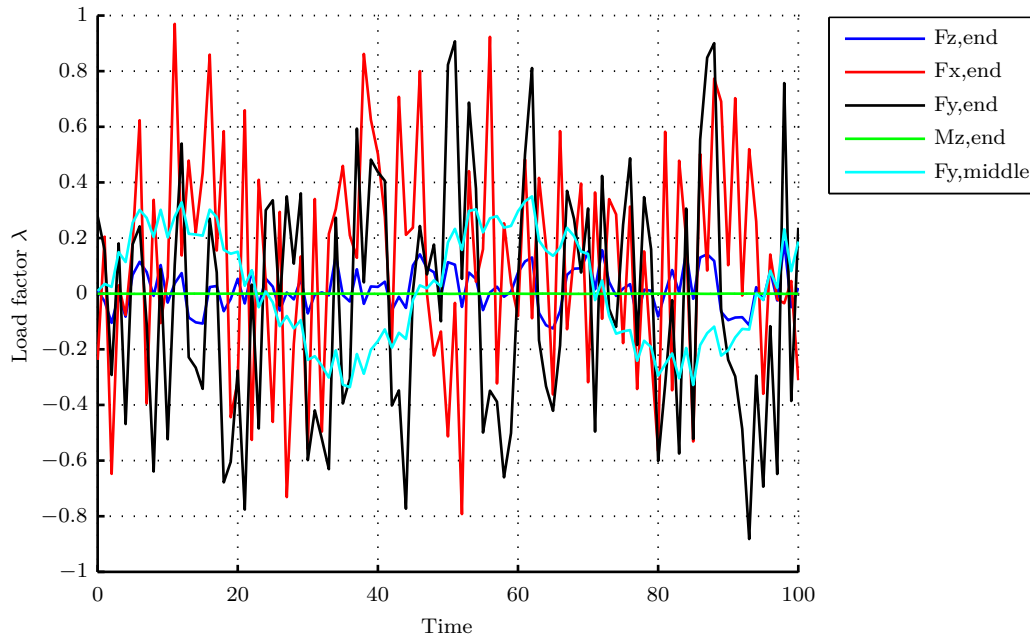


Figure 4.3: Load factor λ plottet over time, to show that the structure will not buckle to a single force at any time.

If the loads is within the factor limit -1 to 1 the force will not cause buckling. As seen the loads are within this limit.

4.2 Benchmark, Standard Solution Procedure.

In order to evaluate a method, one must have something to compare it with. Several benchmarks is performed, to find the relative performance for the different solutions.

Benchmark is an established point of reference against which methods can be measured in tests comparing their performance, reliability, and etc. [Dictionary.com 2012]

The benchmark is defined as the solution procedure described, and shown by the flowchart in Section 1.1. The reason for this is that the described solution procedure is as mentioned earlier, how the buckling analysis is presently conducted, and is therefore an obvious choice as a point of reference, for which the developed methods can be compared.

For convenience the flowchart is displayed below in Figure 4.4.

The benchmark is performed using MATLAB, which controls the selection of load com-

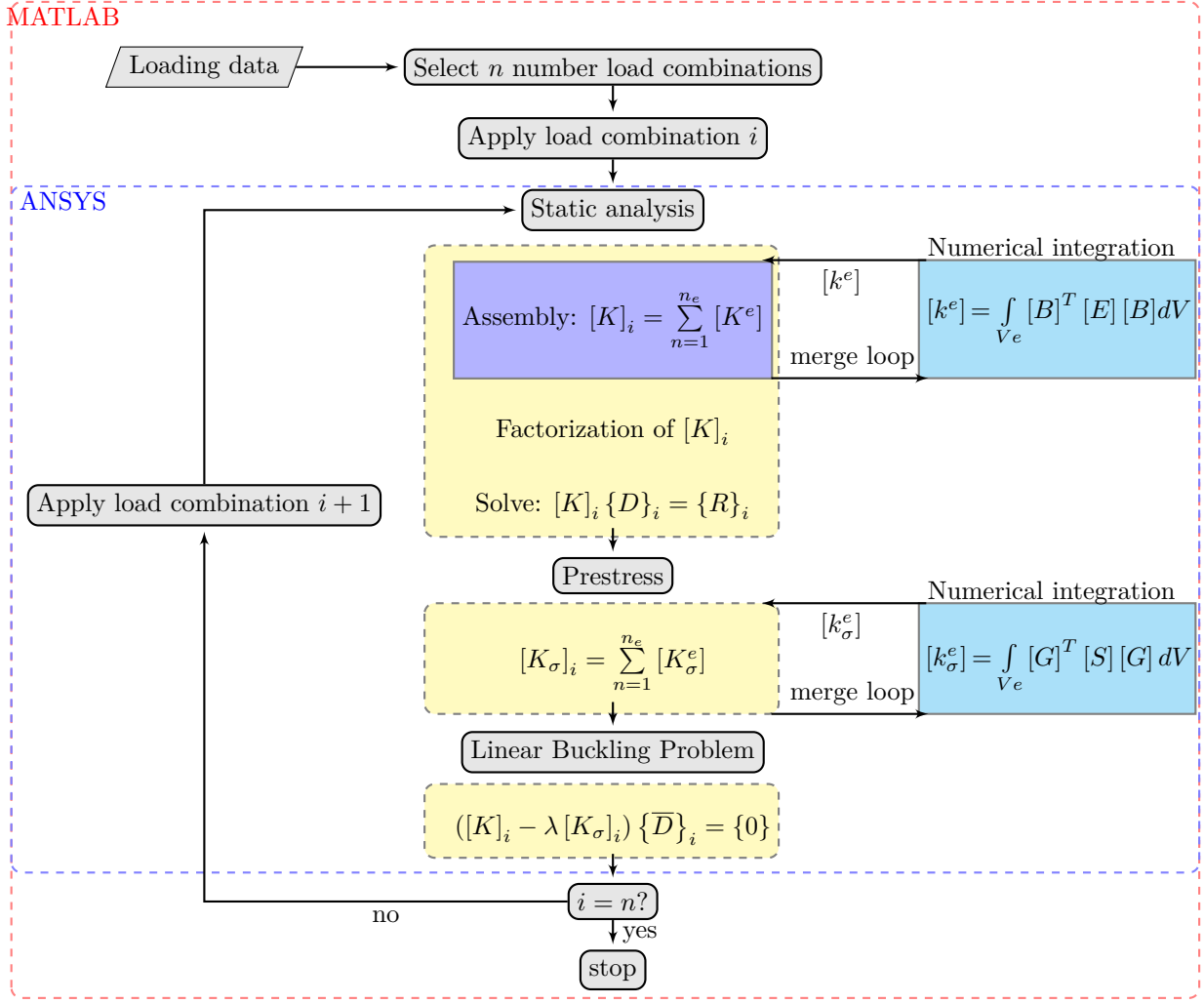


Figure 4.4: Flowchart of the Benchmark solution procedure, used for the performance studies.

bination from the load data. If load combinations $n = 10$ are chosen, the solution loop, will run 10 times, and the load factors is computed for each iteration, along with the total solution time for the analysis.

4.2.1 Implementation of the Benchmark in MATLAB

As mentioned, the Benchmark is implemented in MATLAB. A MATLAB script is written to run the specified load combinations, along with extracting data from the generated load data. The overall steps in the MATLAB script are shown in Figure 4.5.

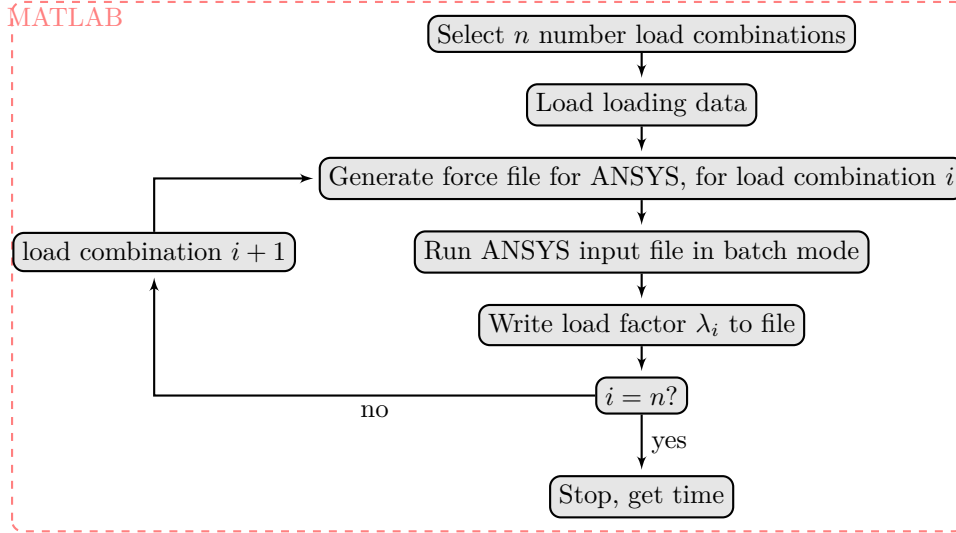


Figure 4.5: Flowchart of the Benchmark implemented in MATLAB.

As seen, the first step in the MATLAB script is that the number of load combinations that shall be calculated is chosen. Hereafter the generated load data is loaded into MATLAB. Third step is to generate a force file, for the load combination i , that is used by the ANSYS input file. The ANSYS input file is described in subsection 4.2.2.

In the fourth step ANSYS is run in batch mode, from MATLAB. When ANSYS have solved the buckling problem, the load factor is written to a file. Finally it is evaluated, if the specified number of load combination is calculated, if yes the program ends, and a total solution time is computed, if no the process is repeated, until all load combinations are calculated.

4.2.2 ANSYS Input File

The input file for ANSYS, is the file that contains all the information needed for a buckling analysis in ANSYS. The input file contains and specifies:

- The geometry of the structure being analyzed
- The boundary conditions
- Material information
- Meshing of the structure with specific elements
- Force application
- Type of analysis
- Wanted output information

For the input file used in the performance studies, conducted in this report, all information needed for an analysis is specified, with the exception of the forces applied to the structure, which is generated by MATLAB.

4.3 Method 1, Solution Procedure using Superposition

Principle on Stress Stiffness Matrix $[K_\sigma]$

This method takes advantage of the possibility for superpositioning the stress stiffness matrix, which is explained. This makes it possible to only compute the stiffness, and stress stiffness matrix once for each load configuration.

An initial study that confirms that the superpositioning is possible have been conducted and is described.

4.3.1 Superpostion

The principle of superposition states that if we look at a linear elastic structure, where two or more forces are applied, then the sum of the individual response from the forces, added together is equal to the response from the forces summed.

The response in a mechanical engineering context are, strain, stress, deflection, reaction forces and moments ect. In this the response is the stress stiffness matrix described in Chapter 3.

Since the superposition principle applies to the stress, then the principle also applies to the stress stiffness, since it is a function of stress. This means that the stress stiffness can be summed up of each individual stress stiffnesses.

4.3.2 Initial Study of Superposition Principle

For the superposition principle to work in this case, where the stress stiffness matrices are obtained with finite element analysis, the mapping, which relates the node number to a location in the finite element mesh has to be the same for every new load configuration. This means that the number of a node in a specific location must not chance from one load configuration to another.

A MATLAB script is programmed, to check the mapping of each load configuration and return an error if there is a difference. The script is run, and showed no difference in the mapping, thus making it possible to use the superposition principle.

If the superposition principle applies to $[K_\sigma]$, then the solution procedure does not require a static analysis for each load combination. This is explained by Eq. (4.1)

$$[K_{\sigma,total}] = F_1 \cdot [K_{\sigma,1}] + F_2 \cdot [K_{\sigma,2}] + \dots + F_n \cdot [K_{\sigma,m}] \quad (4.1)$$

where m is the number of load configurations. The number of static and prestress analysis that is performed, is always equal to the number of load configurations m , no matter the number of load combinations calculated.

This has been tested in ANSYS, by calculating two stress stiffness matrices for two different load configurations and summing: $([K_{\sigma,1}] + [K_{\sigma,2}])$. The summation of these is compared to an ANSYS calculation with all load configurations applied ($K_{\sigma,all}$). If the following

equation is equal to zero, then the superposition principle works for the stress stiffness:

$$([K_{\sigma,1}] + [K_{\sigma,2}]) - [K_{\sigma,all}] \approx 0 \quad (4.2)$$

Eq. (4.2) is equal to zero, meaning that the superposition principle applies to the stress stiffness matrix $[K_{\sigma}]$.

4.3.3 Solution Procedure

The solution procedure of Method 1, that uses the superpositioning principle explained above, is shown in Figure 4.6.

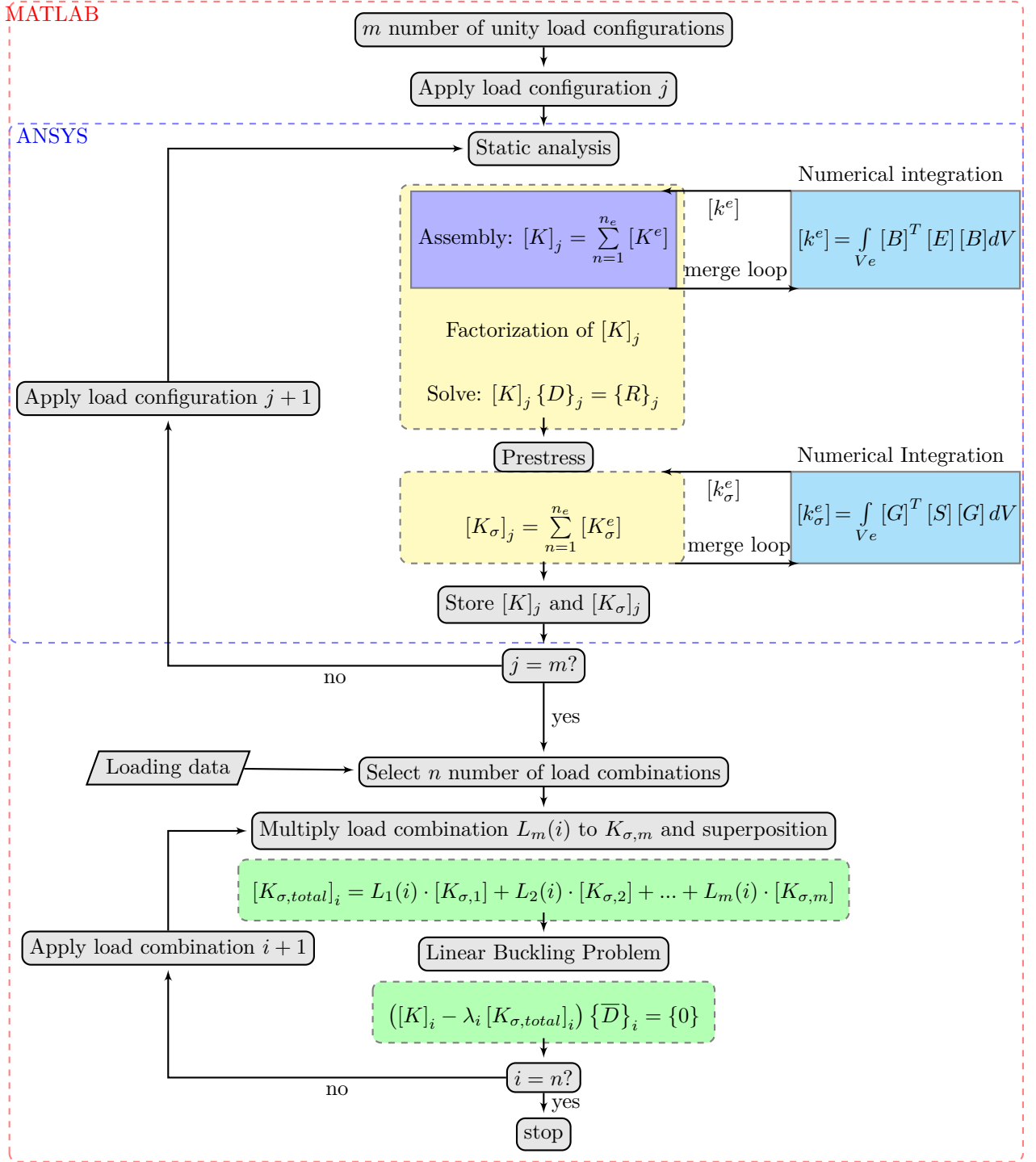


Figure 4.6: Flowchart of Method 1 for buckling analysis with multiple load combinations, when using the finite element method.

As seen from Figure 4.6, the eigenvalue problem is removed from the first loop, and is first calculated in the second loop. The first loop now contains the static analysis and calculation of prestress $[K_\sigma]$, that is calculated with different unity load configurations m . It is furthermore seen that $[K_\sigma]$ is stored for each load configuration, for use in the second

solution loop. The first loop runs until $j = m$. In the second loop, the stored $[K_\sigma]'$'s is multiplied with the load configurations for load combination i , and superpositioned. Next the eigenvalue problem is solved, and the load factor λ_i is determined. This loop continues until the load factors λ_i is calculated for the desired number of load combinations n . As seen from the flowchart, the first loop is calculated using ANSYS, and the second loop is calculated using MATLAB.

The eigenvalue problem is solved in MATLAB and the load factors λ_i are found for the specified load combinations.

However, Solving the eigenvalue problem in MATLAB invoke complications according to the comparison of Method 1 with the Benchmark method. This is because the solution time of the eigenvalue problem might be different from ANSYS to MATLAB. Given that this thesis seeks to develop a method that can decrease the total solution time of the buckling analysis, the methods can still be compared, based on their total solution time. A check of the displacement field and eigenvalues, shows no difference between solving the static problem and buckling problem in ANSYS and in MATLAB.

4.3.4 Implementation of Method 1 in MATLAB

The implementation of Method 1 in MATLAB is described in this section. As for the Benchmark, a flowchart is constructed to show the main steps in the MATLAB script for Method 1. The flowchart is shown in Figure 4.7

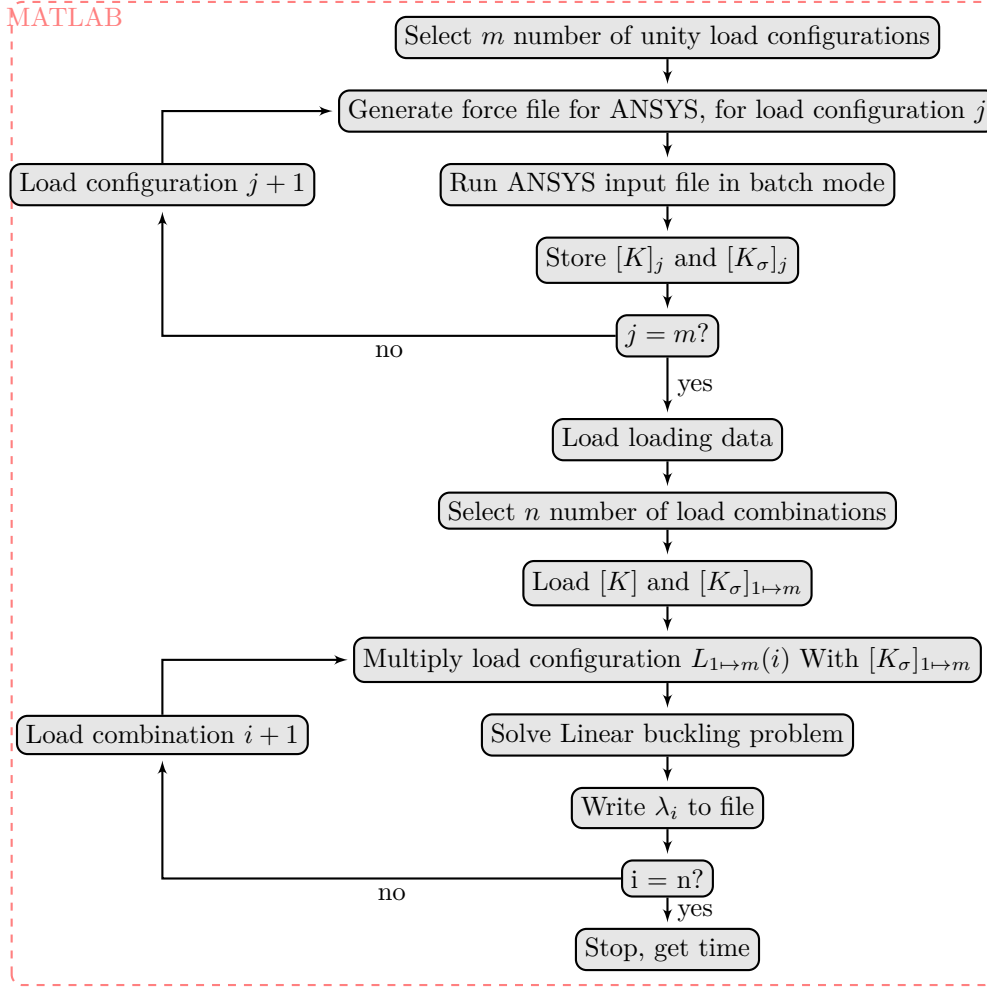


Figure 4.7: Flowchart of Method 1 implemented in MATLAB.

For Method 1 the first step is to select the number of unity load configurations. After the selection, a force file is generated, and the input file for ANSYS is run in batch mode. Next the stiffness, and stress stiffness matrices are stored, for use later in the solution process.

In the fifth step, it is evaluated, if all load configurations are calculated, if yes, MATLAB proceeds to Loading the load data.

When the data is loaded, the number of load combinations n is selected, and $[K]_j$ and $[K_\sigma]_{1 \rightarrow m}$ is loaded into MATLAB. Next the load configurations for $L_{1 \rightarrow m}(i)$ is multiplied with $[K_\sigma]_{1 \rightarrow m}$.

The eigenvalue problem is now calculated in MATLAB, and the load factor λ_i is written to a file. Finally it is evaluated, if all load combinations is calculated, if yes, the procedure stops, and the total solution time is computed.

4.4 Method 2, Solution Procedure using Superposition Principle and Cholesky Factorization.

In this section, a solution procedure using information concerning the definiteness of the tangent stiffness matrix is presented. The method, used to determine if the tangent stiffness matrix is positive definite, is the Cholesky factorization.

The mathematical conditions that has to be fulfilled for a matrix to be positive definite are stated, and the Cholesky factorization algorithm is explained, and how the algorithm determines if the matrix is positive definite or not. Finally the solution procedure is shown in a flowchart, and the procedure is explained.

4.4.1 Positive Definite

In Section 3.1 the physical meaning of having a positive definite stiffness matrix where stated. In this the mathematical conditions that have to be fulfilled for a positive definite matrix are stated. A matrix is said to be positive definite when the following equivalent conditions are met:

- All eigenvalues are positive
- All determinants are positive
- All pivots are positive

Since the conditions are equivalent they fulfill each other. In other words, all conditions are met or none at all. In the following section a factorization method that can determine if a matrix's definiteness, is presented.

4.4.2 Cholesky Factorization

The Cholesky factorization, normally used for numerical solution of a system of linear equations, is here used to evaluate if a matrix is positive definite or not. The Cholesky factorization will be explained in this section along with the algorithm for performing the factorization.

According to [Cheney & Kincaid 2007], André-Louis Cholesky found that if the matrix A is real, symmetric, and positive definite, then it has a unique factorization given as:

$$[A] = [L][L]^T \quad (4.3)$$

Where A is a $n \times n$ symmetric positive definite matrix and L is a $n \times n$ lower triangular matrix. Eq. (4.3) states that a symmetric positive definite matrix can be decomposed

into a lower matrix L multiplied by the lower matrix transposed. The algorithm is given in Algorithm 4.1 below:

```

integer i,k,n,s;
for  $k = 1$  to  $n$  do
     $l_{kk} \leftarrow \left( a_{kk} - \sum_{s=1}^{k-1} l_{ks}^2 \right)^{1/2}$ 
    for  $i = k+1$  to  $n$  do
         $l_{ik} \leftarrow \left( a_{ik} - \sum_{s=1}^{k-1} l_{is} l_{ks} \right) / l_{kk}$ 
    end
end

```

Algorithm 4.1: Cholesky factorization algorithm.

The Cholesky factorization determines if the matrix being factorized is positive definite, when the diagonal entries in the matrix are computed. This can be explained by looking at the computation of diagonal entries in Algorithm 4.1. If the diagonal entry l_{44} is computed, the formula for this would be:

$$l_{44} \leftarrow \sqrt{a_{44} - (l_{41}^2 + l_{42}^2 + l_{43}^2)} \quad (4.4)$$

Where a_{ij} is the matrix being factorized, and l_{ij} is the new lower matrix.

As can be seen the algorithm takes the square root of the diagonal entries, if the product of the values within the squareroot of Eq. (4.4) are positive, the algorithm will be able to compute a real number (\mathbb{R}), but if the product of the values are negative, the calculation will lead to a complex number (\mathbb{C}). The reason that we are able to tell if the matrix is positive definite based on the diagonal entry, is because we make use of the statement above that states that a matrix is positive definite if all pivots are positive. When the Cholesky factorization decomposes the matrix $[A]$ into a lower matrix $[L]$, it uses a form of pivoting, making the diagonal in $[L]$ to Pivots, and thus it can be assessed if the matrix is positive or not, based on the computation of the diagonal term in $[L]$. In this way one obtains information of the matrix without having to calculate the eigenvalues to determine if the matrix is positive or not.

4.4.3 Solution Procedure

In the following the solution procedure, that uses the Cholesky factorization, is presented and explained. This solution procedure follows the same solution procedure as the superpositioning, which is shown and explained in Section 4.3, with one exception. This is that the eigenvalue problem is not solved, instead a Cholesky factorization is performed, on the tangent stiffness matrix which is given as:

$$[K_T] = [K] + [K_{\sigma, total}] \quad (4.5)$$

$[K_T]$ is here reduced to consist of the terms $[K]$, and $[K_{\sigma}]$, assuming the initial displacement stiffness matrix $[K_L]$ to be zero, because of small displacements. The change in the last iteration loop is shown in Figure 4.8.

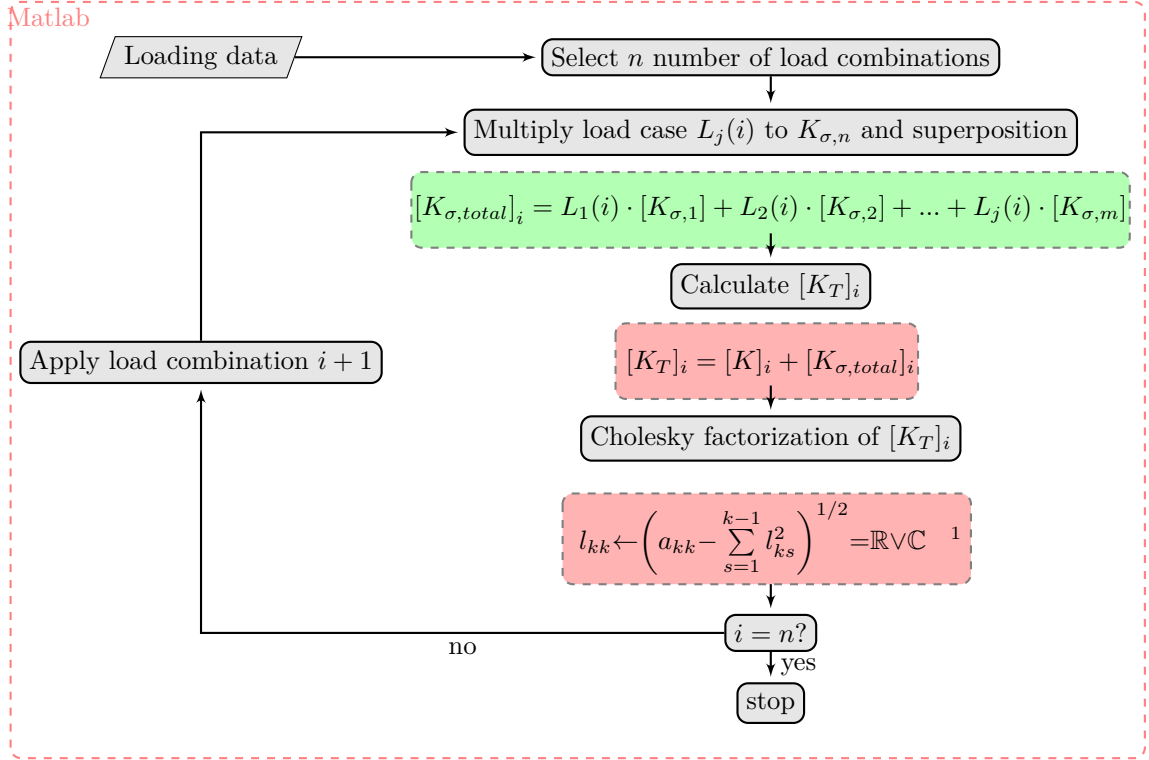


Figure 4.8: Flowchart of the last iteration loop in Method 2, using Cholesky factorization and superposition of K_σ .

By avoiding the computation of the eigenvalue problem the computational time is greatly reduced as will be shown in Chapter 5. However, this reduction in computational time comes with a cost, namely that the only information obtained is whether or not the structure will buckle to the specific load combination. No information regarding the magnitude of the buckling force is obtained with this method.

4.4.4 Implementation of Method 2 in MATLAB

The implementation of method 2, follows the same procedure as for Method 1, given in Figure 4.7, with a small change in the last loop. In the last loop, after the load configurations is multiplied with the stress stiffnesses, $[K_T]$ is calculated, and a Cholesky factorization is performed on $[K_T]$ to determine if the matrix is positive definite or not. When all load combinations is calculated, the procedure stops, and the total solution time is computed.

4.5 Summary

In this chapter the load data used in the performance studies of the different methods where presented. Furthermore, two different methods that does only have to calculate the

¹ \mathbb{R} is real numbers, and \mathbb{C} is complex numbers

time consuming standard solution procedure for each load configurations, was presented along with the principle of superpositioning and the Cholesky factorization, used in the two methods. In the next chapter the performance studies are explained, and the results of the studies are presented and discussed.

Performance Studies of Solution Procedures

5

In this chapter the total solution time, for the different solution procedures for different mesh sizes, layers, and load combinations are studied. The total solution time is beneficial to know in order to compare the different solution procedures performance, according to calculation of the buckling analysis, and possible limitations. Different setups are constructed, where the number of elements, number of layers, and number of load combinations i are varied, respectively.

Based on the results from the different setups, the methods is compared and the performance of each procedure will be evaluated in given situations. At the end of this chapter, it will be clear what procedure is the fastest, in different situations, and the pros and cons for the different solution procedures along with possible limitations.

The variables have the ranges as shown below:

- Number of elements = 6600, 3216, and 884
- Number of load combinations i = 1-101
- Number of Layers = 1-100

The number of elements, load combinations and layers are limited to 6600, 101 and 100 respectively, in order to keep the studies in a manageable size, so that the solution of the setups will not take too long, and matrices written by ANSYS for calculation in MATLAB will not get too large.

To insure that all possible effects are studied, all possible ways of combining the parameters is calculated. This means that we get $3^3 = 27$ different combinations. These combinations are calculated for each of the solution methods mentioned in Chapter 4. By doing this we get a total of $3 \times 27 = 81$ analysis.

Setup

The different setups that is calculated, are done on the same PC, to be able to compare the different analysis and setups. The PC used has the performance specifications:

- CPU: AMD 4.3 GHZ PHENOMIIx4-965
- RAM: 8GB DDR3 1333MHZ
- HDD: 7200 RPM, 1 TB

The different setups in this section is constructed from the 81 analyses conducted. The setups is explained and the explanation will include where in the solutions procedures, stated in Chapter 4, the setup will have an effect.

After each explanation the results for the setup are presented and discussed.

5.1 Setup 1 - Influence from Number of Elements and Number of Layers on Total Solution Time

In this setup, the parameters that are varied are the number of elements and number of layers. The increase in the number of elements and layers will have a different influence in the different solution procedures.

For the Benchmark, the number of elements have an effect in all of the steps in the solution procedure, that is in:

- Assembly of the stiffness matrix $[K]$
- Factorization of the stiffness matrix $[K]$
- Solution of the static problem
- Assembly of the stress stiffness matrix $[K_\sigma]$
- Solution of the eigenvalue problem

This is because the increase in number of elements will increase the degree of freedom in the system, and hence the size of the matrices.

Varying the number of layers in the element, will in the Benchmark only effect:

- Assembly of stress stiffness- and stiffness matrix, $[K_\sigma]$ and $[K]$

This is because a numerical integration has to be performed for each layer in the element. This is thus very computational demanding if many layers are used, as in laminated structures ect.

In Method 1, described in Section 4.3, the increase in number of elements, will have a different effect than that of the Benchmark. This is due to the difference in the solution procedure. I.e in Method 1 the stiffness- and stress stiffness matrices are only assembled once for each unity load configuration, where as the Benchmark is running this step for each of the load combinations.

The number of layers will have an effect in the first loop of the solution procedure, but

not in the second loop, because the size of the stiffness- and stress stiffness matrices will not change with the change in layers in the element.

In Method 2, the increase in elements will effect:

- Assembly of stress stiffness- and stiffness matrix, $[K_\sigma]$ and $[K]$, in loop 1.
- Cholesky factorization time

As for Method 1 the number of layers does only effect the first loop, where the stiffness- and stress stiffness matrix are obtained.

The variables for this setup are:

- Number of elements = 884, 3216 and 6600
- Number of load combinations $i = 101$
- Number of layers = 1, 50 and 100

5.1.1 Result (setup 1)

In this the results for setup 1 are presented. The results, which shows the increase in elements with respect to the total solution time in seconds are plotted in Figure 5.1. The results from the Benchmark, Method 1 and Method 2 are plotted for 3 different number of layer in the elements, which is 1, 50 and 100 layers.

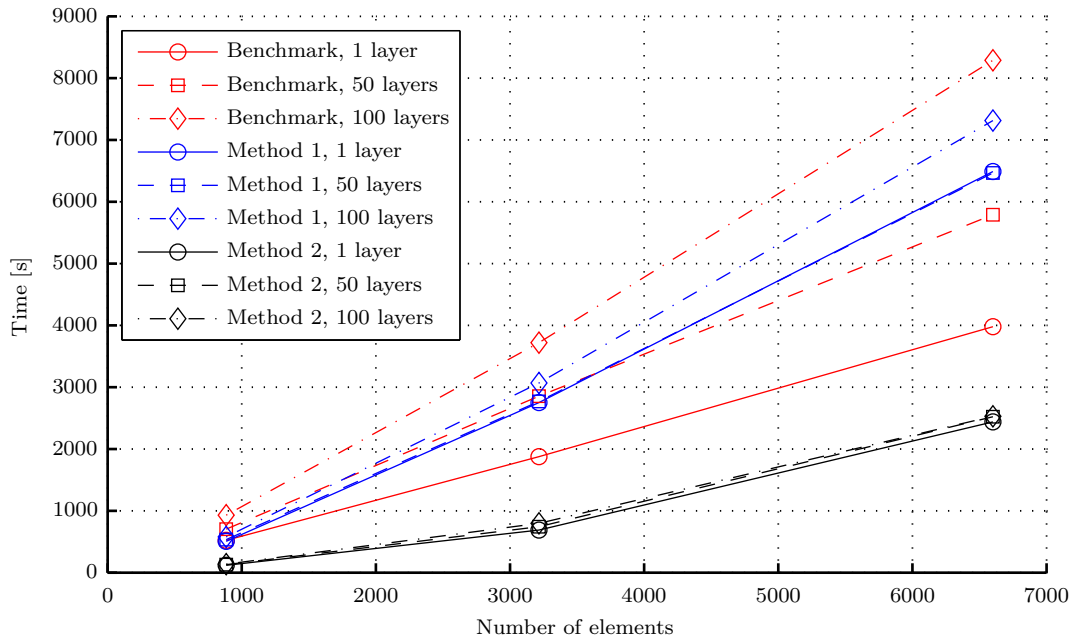


Figure 5.1: Results from setup 1 with 101 load combinations, where the increase in elements is plotted with respect to the total solution time in second. The number of layers are varied as well.

In Figure 5.1, there are three important things to notice.

The first thing that is noticed is that Method 2, decreases the total solution time the most, no matter the number of layers or number of elements.

Furthermore, Method 2 has reduced the total solution time, according to the presently used method which is Benchmark in the figure, with 69.5% for 6600 elements and 100 layers.

It is furthermore concluded that the efficiency according to the total solution time decreases with the number of elements. This can be seen if the reduction in the total solution time, with Method 2 at 3216 elements, according to the Benchmark, is compared to Method 2 at 6600 elements. For 3216 elements the reduction is 78.5%, which is higher than the reduction of solution time obtained with 6600 elements. However, the slope of the curves for the Benchmark and Method 2 are diverging from each other. From this we can conclude that Method 2 will always be faster than the Benchmark, no matter the number of elements, but the efficiency according to the total solution time is decreasing as the number of elements increases.

There are two main reasons for the decrease in efficiency according to the total solution time, for the increase in number of elements. The two reasons are, free memory and/or loading of $[K]$ and $[K_\sigma]$ into MATLAB. When performing these studies on the performance, Method 1 and 2 has to load the stiffness matrix into MATLAB for calculation, as described in Section 4.3.4, which ANSYS do not. Depending on the time usage for loading the matrices into MATLAB, this might be the factor that causes this decrease. The other factor that can have an influence, is the amount of free memory available when doing the calculations in MATLAB.

If the percentage reduction in total solution time, according to the benchmark, is calculated for Method 2, we get 38%, for the 1 layer and 6600 elements. Comparing this to the results for 100 layers and 6600 elements, which is 69.5%, it becomes clear that Method 2 becomes more effective when the number layers is increased.

The reason for the increase in efficiency, according to the total solution time, when increasing the layers, can be seen from the flowcharts, for Benchmark and Method 2, in Chapter 4.

When increasing the number of layers in the element the assembly of the stress stiffness- and stiffness matrix, $[K_\sigma]$ and $[K]$ is increased. This is because a numerical integration has to be performed for each layer in the element. This is thus very computational demanding if many layers are used. The assembly of $[K]$ and $[K_\sigma]$ is for Method 2 performed for each load configuration, but for the Benchmark the assembly are performed for each load combinations, thus the relative difference between the two methods must increase for an increasing number of layers.

The second important behavior that is noticed is that Method 1 with 50 layers, is

intersecting the Benchmark at an element number of 3600, and thus becomes slower with respect to the total solution time. Furthermore for 1 layer, Method 1 is slower, for all number of elements, than the Benchmark.

This is an unexpected behavior of the superpositioning solution procedure, that was expected to decrease the total solution time.

The reason for this is that the eigenvalues, in Method 1, are calculated in MATLAB.

MATLAB allows for negative eigenvalues, which has no physical meaning. In order to get the positive lowest eigenvalue, a shift is made, and the eigenvalue problem is solved again. Doing this means that the eigenvalue problem has to be calculated twice for each time a negative eigenvalue is obtained. Given that the computation of the eigenvalue problem is the most time consuming part of the solution process, it is clear that having to calculate the eigenvalues twice, for some of the load combinations, will have a large effect on the total solution time. The eigenvalues calculated by Method 1 is compared to the eigenvalues obtained by ANSYS. The comparison showed that Method 1 calculated the eigenvalues correctly for all load combinations.

The third behavior that is noticed is that the result obtained with Method 1 for 1 and 50 layers, and with Method 2 for all layers, are almost identical.

The reason for this is that the number of layers only have an affect on the first solution loop for Method 1 and Method 2, shown in the flowcharts in Figure 4.6 and Figure 4.8, and the difference in the solution time of the first loop from 1 to 100 layers are minor, compared to the solution time of the second loop.

The solution time of the first loop, for 1, 50 , and 100 layers, is 262, 357, and 474 seconds, respectively and the solution time of the second loop for Method 2 for all layer setups, is approx. 2120 seconds. From this, the reason for the almost identical results should be clear.

The result from Method 1, for 100 layers, should display the same behavior as the result for 1 and 50 layers, but as seen in Figure 5.1 it does not. The exact reason for this divergence from the expected behavior can be hard to determine but a guess would be that it is the effect from not having the same start conditions of the computer for each analysis i.e the CPU of the computer could be warmer, and thus the solution time will increase.

From the above discussion we can conclude that:

- Method 2 has the lowest total solution time, no matter the number of elements and layers.
- The efficiency according to the total solution time decreases with the number of elements.
- Method 1 is slower than the Benchmark for 1 and 50 layers.

In the next the intersection between Method 1 and the Benchmark is studied further.

5.2 Setup 2 - Effects of Number of Load Combinations and Layers

This setup has the purpose of determining to which number of layers, Method 1 gets a lower total solution time.

The variables for this setup are:

- Number of elements = 6600
- Number of load combinations $i = 21; 61; 101$
- Number of Layers = 1; 50; 100

Changing the number of load combinations, will have a large effect on the Benchmark, because it essentially determines the number of times the solution procedure given in Figure 4.4, is run. Hence if 100 combinations is calculated, the computational time is 5 times as much as if 20 load combinations are calculated.

For Method 1 and Method 2, the increase in number of load combinations will only have an effect in the second loop of the solution procedures. This is because the first loop in the solution procedure for Method 1 and Method 2 is independent of the load combinations, and is only calculated for the number of load configurations.

5.2.1 Results (Setup 2)

The results for setup 2 is presented in Figure 5.2. The figure shows the increase in layers with respect to the total solution time in seconds. The study is run for three different number of load combinations for all of the solution procedures.

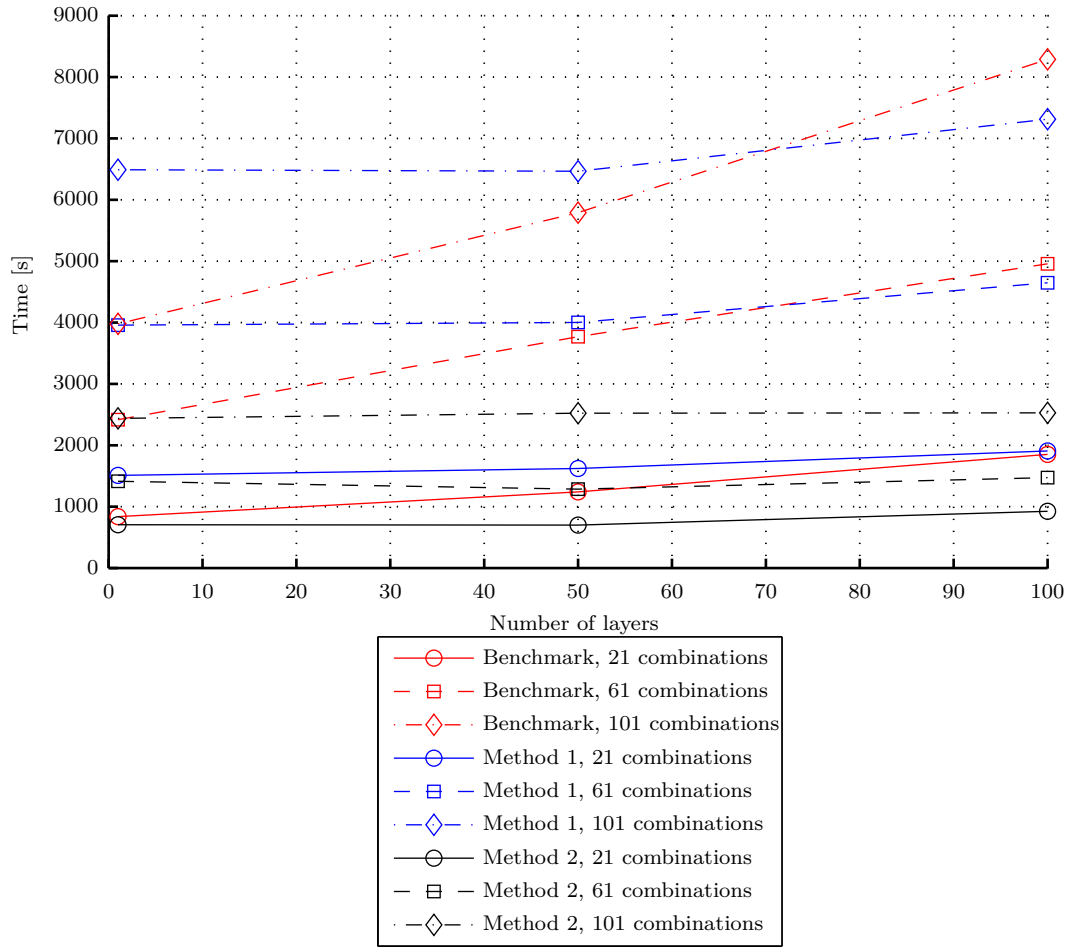


Figure 5.2: Results from setup 2 with 6600 elements, where the number of layers is plotted with respect to the total solution time in seconds. Furthermore, the load combinations are varied.

From Figure 5.2 it becomes clear to which number of layers, for a certain number of load combinations, Method 1 intersects with the standard solution procedure and thus becomes faster.

For Method 1 with 101 load combinations, intersects the Benchmark at approx. 71 layers, and for 61 load combinations at 72 layers. For 21 load combinations the Method 1 will intersect the Benchmark shortly after 100 layers.

Two things can be concluded, namely that Method 1 becomes more efficient with the increase in layers, but also by increase in load combinations, because it requires a lower number of layers for Method 1 with 101 load combinations to intersect the Benchmark.

Another noticeable behavior is that for 1 layer, Method 1 is slower than the Benchmark, for the studied number of load combinations. This could as mentioned in Setup 1 indicate

that this is due to the shift that has to be made in MATLAB.

It is further more noticed that the slopes of Method 2 is very close to zero for the entire range of layers, which corresponds with the results in Setup 1, and superposition procedure for 1 to 50 layers. The reason for this is that the number of layers has no effect on Method 1 and Method 2 (second loop in Figure 4.6), but only for the calculation of the stiffness and stress stiffness. The reason for the increase in slope for Method 1 could be the effect from the CPU, which gets warmer as solution time increases, thus the solution time becomes slower.

5.2.1.1 Behavior when Having a Few Number of Load Combinations

As seen from Figure 5.2, the lowest studied number of load combinations is 21. However, it is important to study the behavior when having a low number of load combinations (1-21). For a low number of load combinations the Benchmark will be faster than Method 1 and Method 2, because the first loop of the solution procedure of Method 1 and Method 2 basically is the Benchmark.

To which number of load combination the Benchmark intersects, with Method 1 and 2, and becomes slower is plotted in Figure 5.3, that contains 3 plots, with 1, 50 and 100 layers, respectively.

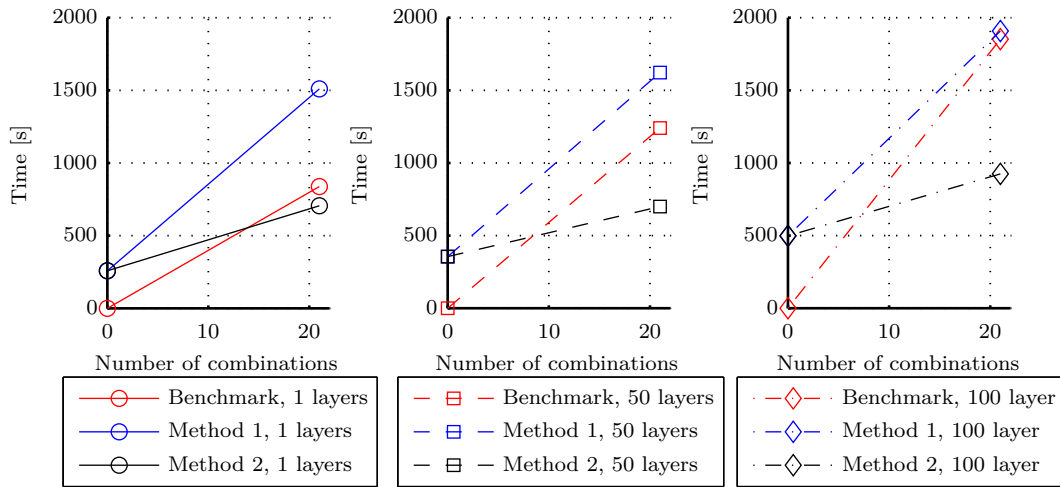


Figure 5.3: Behavior of solution procedures with 6600 elements, when having a small number of load combinations for 1, 50, and 100 layers, respectively.

The first thing that is observed in Figure 5.3 is that the Benchmark is the fastest, for a low number of load combinations. The reason for this is that the first loop in Method 1 and Method 2, has to be run for all the load configurations, before the actual solution can begin, hence the start value greater than zero, in the three plots. The Benchmark implements the five load configurations at once, and is therefore able to calculate the buckling loads faster than the other procedures for a low number of load combinations.

The second thing that is observed is that the point of intersection happens faster for

increase in layers. For 1, 50, and 100 layers, Method 2 intersects the benchmark at approx. 14, 9, and 8 load combinations, respectively.

5.3 Summary

The performance of two of the solution procedures chosen in Section 2.2, have been evaluated in the above two sections.

The performance studies have outlined the procedures advantages, disadvantages and limitations. It has been shown that Method 2 is by far the most effective procedure with respect to the total solution time. Method 2 remains faster than the Benchmark, no matter the number of elements, layers or load combinations.

Method 1 has a limited effect because of the shift that has to be done in MATLAB. Furthermore, the procedure is slower than the standard procedure when having a low number of layers. It is also shown that a minimum number of load combinations exist to which Method 1 and Method 2 are faster than the Benchmark. Finally it has been shown that the number of layers has none to little effect on the total solution time of Method 2 and partly Method 1.

In the next chapter the last of the chosen solution methods for further study in Section 2.2 is described, along with performance studies of the method.

Data Fitting of a Stability Boundary

Boundary 6

In this chapter the concept of the method is explained. After that a number methods for sampling the stability boundary is described along with a number of different fitting and interpolation methods. In [Huseyin 1975] it is described how a stability surface boundary exist, if the structure is loaded with more than one load. An example of a stability boundary, with two loads present, which are independent of each other, is seen in Figure 6.1. The stability boundary is given by one or more critical surfaces, each with their own equilibrium equation. A point on the stability boundary is in this thesis denoted as a "critical point", since it is the point where a certain load ratio intersects the stability boundary.

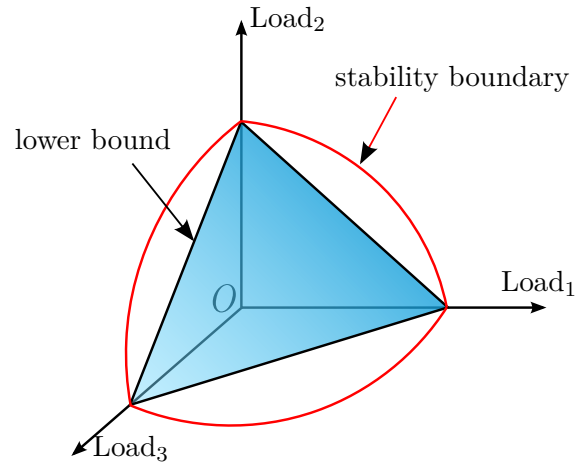
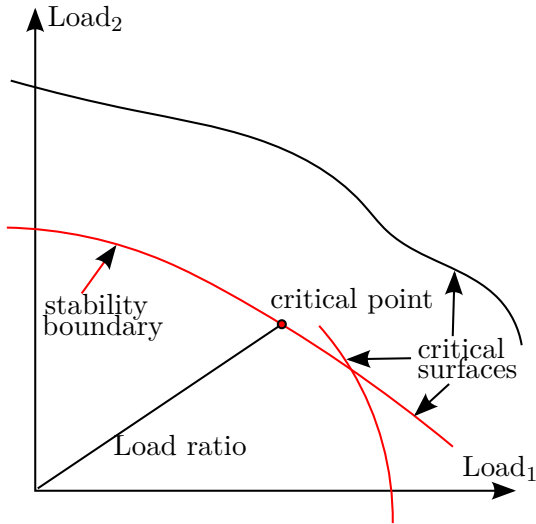


Figure 6.1: Concept of a stability boundary for two independent loads.

Figure 6.2: Stability boundary and lower bound in 3 dimensional load space.

This stability boundary consequently divides the load space into a stable and unstable region. In [Huseyin 1975] and [de Faria & Hansen 2001], a theorem and proof is given about the geometric properties of this stability boundary in linear buckling.

Theorem 2. *The stability boundary cannot have convexity towards the region of stability.*
[Huseyin 1975, page 170]

According to the theorem the boundary can never be concave, which means that a lower bound can be established, as illustrated in Figure 6.2.

For complex structures, the stability boundary cannot be found analytically. However it is possible to find discrete points of the stability boundary using the finite element method. When a linear buckling analysis for a load combination is calculated, the result is given as a load factor λ , also called eigenvalue. The eigenvalue is a scalar which multiplied by the load combination, yields a critical point in a load space for a specific load configuration (number of loads) and ratio between the loads. If two reference loads are applied then their critical point's coordinates is given by multiplying the reference loads with λ , as seen in Figure 6.3. The critical point for a load combination is a point on the stability boundary, as described in Figure 6.1. Different ratios of two loads will give different critical points in the load space. These critical points are discrete sampled points of a continuous stability boundary in the load space. If a curve fit between the points are made an approximation of the stability boundary is made. The reference points, i.e. $P_{1,ref}, P_{2,ref}$ will now be denoted as sampling points, since they are used to sample the stability boundary.

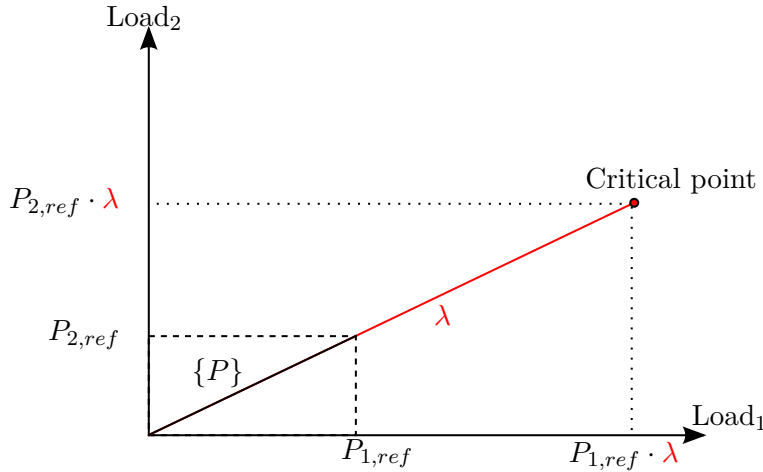


Figure 6.3: Critical point and reference/sampling point.

The critical point in 2 dimensional load space is thus given by Eq. (6.1)

$$\begin{Bmatrix} P_{1,ref} \\ P_{2,ref} \end{Bmatrix} \lambda = \begin{Bmatrix} P_{1,critical\ point} \\ P_{2,critical\ point} \end{Bmatrix} \quad (6.1)$$

The same sampling load ratios, yields the same critical point, regardless of the magnitude of the sampling loads. In example $\{P_{1,ref} = 10 [N], P_{2,ref} = 5 [N]\}$, yields the same critical point as $\{P_{1,ref} = 20 [N], P_{2,ref} = 10 [N]\}$, but the eigenvalue would be half.

The loading data of a structure, which consist of a number of load combinations, are considered points in the load space. In other words, each load combination is a point.

The load combinations can be regarded as stable if they are inside the stability boundary, and unstable if they are outside. This is illustrated in Figure 6.4, where the black points are stable load combinations, the blue points are unstable load combinations, and the white points are the critical points on the stability boundary.

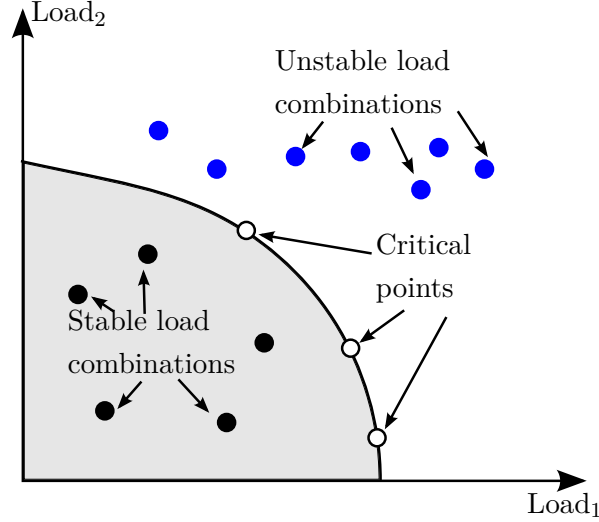


Figure 6.4: Load combinations and a stability boundary with critical points.

The precision of the stability boundary is dependant on the position of the critical points used to establish the boundary and the fitting method used. The critical points should either be equally spaced or precise in the region where most load combinations are present. The dimension of the load space is the number of load configurations m . The points used to create the stability boundary will be denoted "sampling points", because they are a discrete sample of a continuous m dimensional surface. In this chapter the position and number of sampling points, used to create the boundary, is examined, along with the methods used for fitting an approximate stability boundary.

6.1 Selection of Sampling Points

As explained previously a stability region exist in a m dimensional load space. This region is described by a discrete set of critical points. The following section is based on the selection of the sampling points, used for calculating the critical points on the stability boundary. The number and position of the sampling points have a big effect on the precision of the stability boundary. Different methods of sampling are tested and evaluated based on precision, with respect to number of sampling points. Since loading data is often very complex, the negative values of each load configuration cannot be neglected, as in previous examples of the stability boundary. The sampling methods have to include all regions, i.e. both tensile and compressive forces.

6.1.1 Central Composite Design

In Central composite design first, a set of points are created by a full factorial design, which makes 2^m points, where m is the dimensions. Full factorial design is a method which creates minimum and maximum combinations of the parameters. Next a set of $2 \cdot m$ axial points are created with the same magnitude as the previous points. This sampling method has a fixed set of points for each dimension, and the method gives a systematically estimation of combined and individual effects, as seen in Figure 6.5.

An example of 2D central composite design with no center points is shown in Figure 6.5, thus creating $(2^2) + (2 \cdot 2) = 8$ points. The points are equally space in polar coordinates.

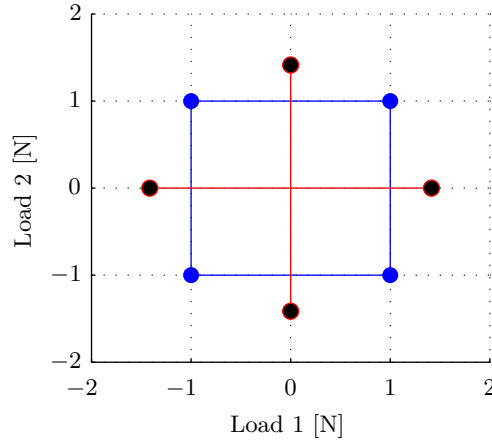


Figure 6.5: central composite design sampling of 2 load configurations.

The blue points on the corners of the square are made from the full factorial design, and the red points are the axial points.

6.1.2 Equally Spaced Points

This method of sampling points is made in order to get the best overall precision of stability boundary, with as few points as possible. First points are made by creating a n dimensional grid with a step size of z , as shown in Figure 6.6 in 2D. Then all the points, which does not have a magnitude of $1 \pm \frac{z}{2}$, are removed (filtered out), as shown in 2D in Figure 6.7. The points are removed since they will coincide on the stability boundary, with points still remaining, which is undesirable.

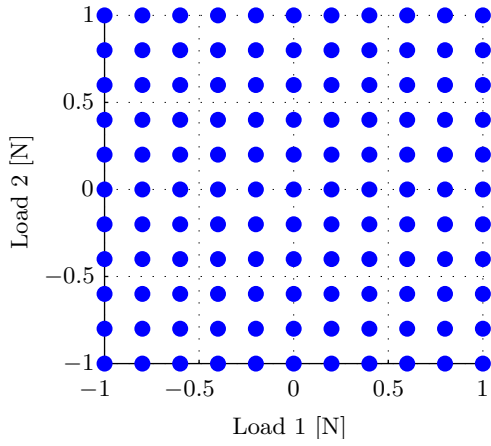


Figure 6.6: 2D grid with $z = 0.25$.

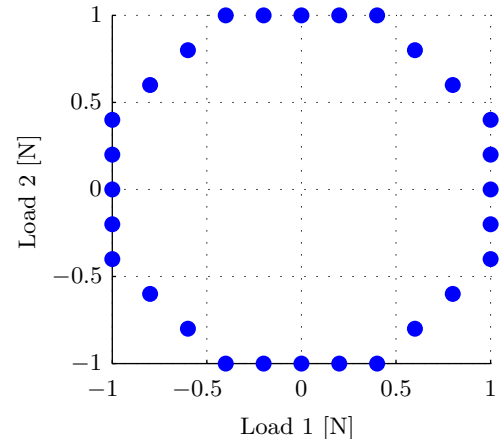


Figure 6.7: Filtering applied.

6.1.3 Cluster Analysis

This method is based on representing the load combinations from the load data by a set of clusters. Cluster analysis divides n number of load combinations into p numbers of clusters. This means the number of sampling points can be chosen as desired. A centroid point is made in each cluster, which is the mean of the points in that cluster. The cluster analysis is a statistical way, to reduce the number of load combinations, into fewer sampling points. An example of 100 load combinations for 2 load configurations, is shown in Figure 6.8, and a cluster analysis of the load combinations into 12 clusters, is shown in Figure 6.9

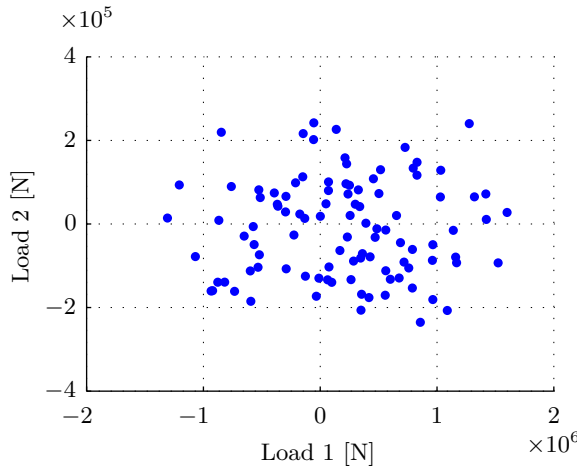


Figure 6.8: Load combinations in load space.

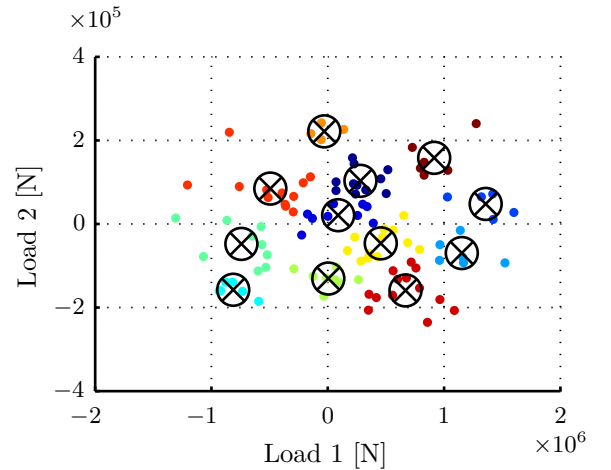


Figure 6.9: 12 clusters of the load combinations.

6.1.4 Random

To examine the effect of the position of the sampling points, random sampling points are made. The random sampling points are uniform random generated numbers, between a lower and upper bound. 20 random sampling points, with a lower and upper bound on $[-1, 1]$ for 2D is shown in Figure 6.10

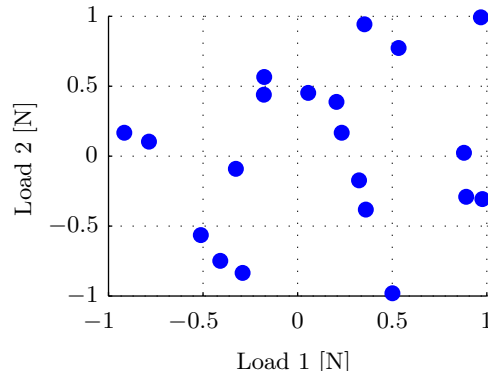


Figure 6.10: 20 random generated sampling points of the load data.

Since the generation is random, the method will produce different results with same settings, if run twice. The random sampling will help examine the sampling points positions importance. If the random sampling points performance is equal to the other methods, then the position has little effect.

6.2 Data Fitting

Data fitting is finding an approximate function f for set of values. In order to create a stability boundary from a discrete set of points, data fitting of the points is performed. Different methods is chosen, so that their precision can be compared. The methods will be compared in Section 6.4.

6.2.1 Convex Hull

According to De Berg et al. [2008], a set S of points is convex if all lines between the points are within S . Convex hull can be thought of as an elastic band stretched over a set of points, and then released so it contracts to the minimum length, which contains all points. This concept is illustrated in Figure 6.11, for 2 dimensional points.

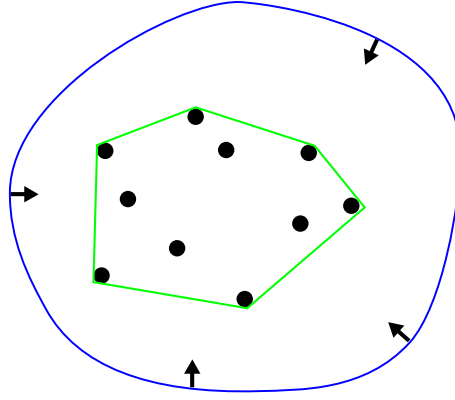


Figure 6.11: Convex hull of a set of points.

Convex hull is expected to be a conservative fitting method, based on theorem 2. In other words, it will never overestimate the stability boundary. Since convex hull is a conservative approach, the approximated stability boundary with convex hull is expected to predict more unstable load combinations, than the true stability boundary.

6.2.2 Interpolation Methods

In interpolation the fitted function is made from many pieces, defined by piecewise polynomials. Interpolation goes from data point to data point, which means the data points have to be reliable. Since the points used for interpolation in this report are exact, interpolation can be used.

Linear

Piecewise linear interpolation creates a straight (linear) line between each point in 2 dimensions, or a straight surface between 3 points in 3 dimensions. Requires at least n number of data points, where n is the dimensions.

Cubic interpolation

Piecewise cubic interpolation uses a cubic (third-degree) polynomial, and creates a cubic polynomial between each point. Requires at least n number of data points.

Biharmonic

Interpolates the minimum curvature between the data points. This method is suited for irregular spaced points, and requires at least n number of data points.

6.2.3 Curve Fitting

Creates a function that includes the entire range of the data.

Polynomial fit

Fits a quadratic polynomial curve between all the points. This method does not necessarily pass through all fitting points used. Requires at least 2^n data points, in order to fit.

Local quadratic regression

Local quadratic regression, operates by finding which points that are close to the response

being calculated, are weighted more than points far away, from the response. This method does not necessarily pass through all fitting points. Requires at least 2^n data points, in order to fit.

6.3 Implementation

First a method of sampling the points is selected, then n number of sampling points are created in MATLAB. The sampling points are written in a force file along with their load configuration. ANSYS is called in batch mode from MATLAB and reads the force file with sampling point i . ANSYS writes out the lowest eigenvalue. The applied sampling point is then multiplied with the eigenvalue, in order to get the critical point. When n critical points are calculated, the fitted stability boundary is made. The fitting method is chosen and a function of fitted stability boundary is obtained. Different values of the function is then calculated for each load combination, in order to determine if the load combination is outside or inside, the fitted stability boundary.

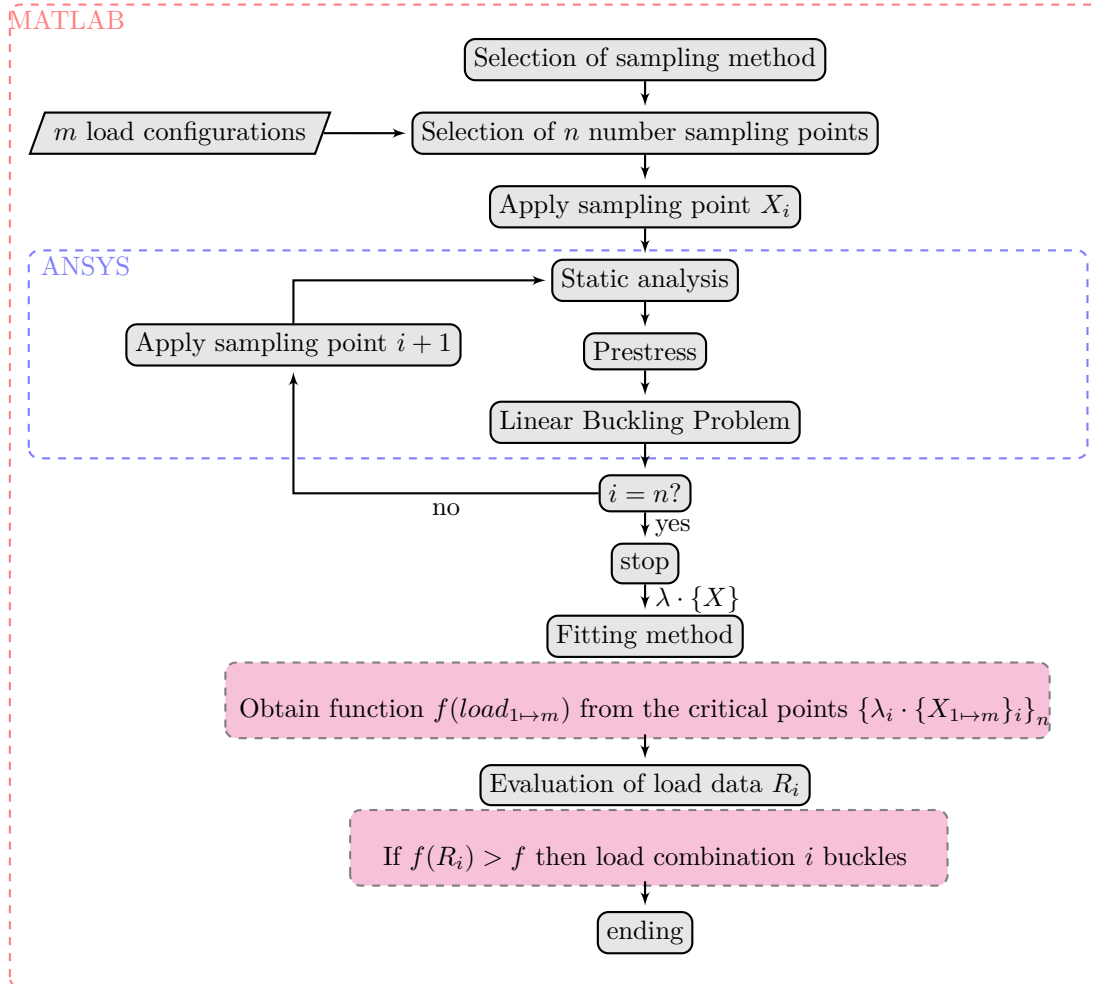


Figure 6.12: Flowchart of the data fitting of a stability boundary created with sampling points and tested for multiple load combinations n .

When the function is fitted, it calculates the function's value $f(R_i)$ at the load combinations. The values are compared to the function, and it can be decided if the load combinations are outside the fitted stability boundary.

6.4 Performance Study

In this section the different sampling methods are tested along with the different fitting methods. The performance of the fitting methods are dependant of the number of sampling points and the sampling methods, however some general features can be found.

The sampling methods and fitting methods are first studied for 3 load configurations and 101 load combinations, and then implemented for all 5 load configurations afterwards. The study is initially done for 3 load configurations, because of the possibility to visually inspect the goodness of the fit, which is troublesome in higher dimensions than 3. In the performance study of the sampling methods and fitting methods, one of each method is selected. The 3 load configurations are $F_{x,end}$, $F_{y,end}$, and $F_{y,middle}$, which are shown in Figure 4.1.

Only the convex hull algorithm can fit a volume in 3 dimensions, while the other fitting methods can only fit a surface. Consequently only a part of the stability boundary is initially analyzed.

All the forces are assumed to be positive, which means that only one quadrant of the 3 dimensional stability boundary is examined. The values of the loading data in Figure 4.1 is used, except the three load configurations always stay positive (absolute values).

The fitting methods are evaluated on their prediction of buckled load combinations (load combinations outside the stability boundary). If the fitted boundary predicts a higher number of buckled load combinations, than the true stability boundary predicts, then the fitted boundary is considered conservative.

If the fitted stability boundary shows less buckled load combinations than the true boundary shows, then the fitted boundary is considered erroneous and risky to use.

The eigenvalue is calculated for 101 load combinations in ANSYS. These results gives knowledge about what load combinations are inside ($\lambda > 1$) and outside ($\lambda < 1$) the stability boundary. The load combination inside and outside the fitted stability boundary are compared to the results from ANSYS, in order to asses the precision of the fitted boundary. An example of the results obtained by fitting a function to certain critical points is shown in Figure 6.13

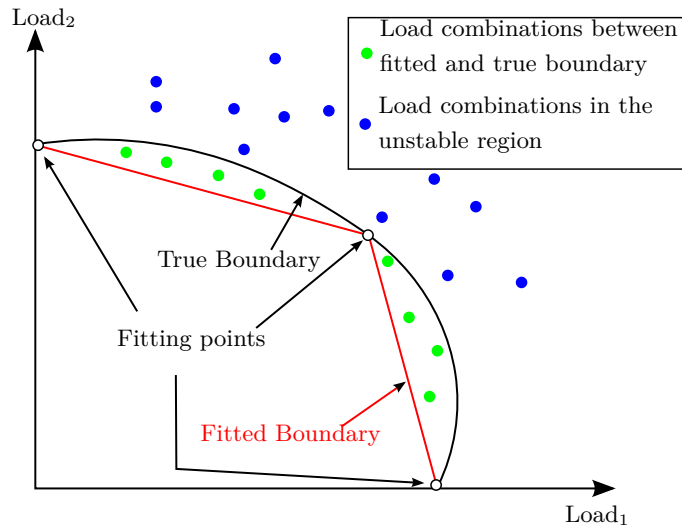


Figure 6.13: Example of a fitted stability boundary and true stability boundary for 2 loads.

In Figure 6.13 the blue points are load combinations with a $\lambda < 1$ calculated in ANSYS, and the green points are load combinations outside the fitted boundary, but with an $\lambda > 1$. The green are still inside the true boundary. It is noticed that the fitted boundary predicts more buckled load combinations, since the fitted stable region has a smaller area than the true stable region. In this example the fitted boundary is conservative, because it predicts buckling at load combinations, which actually does not buckle in ANSYS. However, overfitting of the fitted stability boundary yields different properties, as shown in Figure 6.14

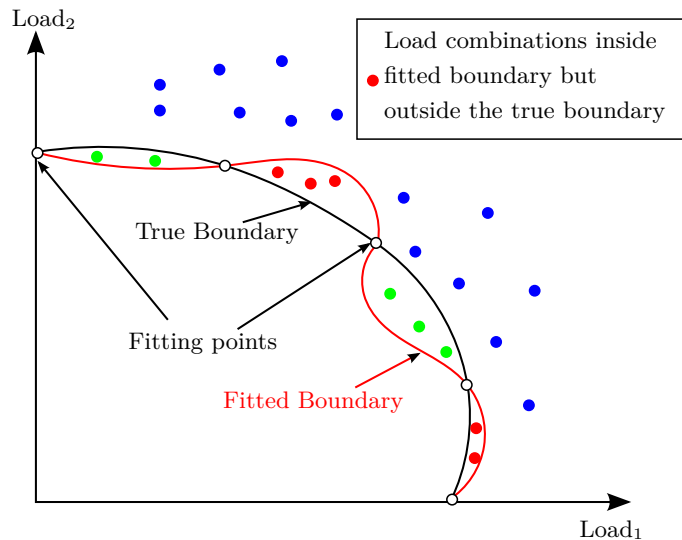


Figure 6.14: Example of a badly fitted stability boundary and true stability boundary for 2 loads.

The fitted curve now predicts stability for some load combinations, which buckles in ANSYS. This error is critical and it means that the fitted boundary is wrong and can not

be used as a tool to estimate the stability of a structure. It is checked that the same load combinations buckles, otherwise the fitted boundary, must be considered erroneous.

It is noticed that if both the fitted stability boundary and true stability boundary predicts the same amount of buckled combinations, it is not necessarily a measurer of precision. If for example, the true boundary predicts 15 buckled load combinations (\bullet), the fitted boundary might also predict 15 ($\bullet + \bullet - \bullet$). This will happen if the amount of green points \bullet is the same, as the amount of red points \bullet . These two predictions of 30 buckled load combinations are thus not the same since the fitted boundary is at some points conservative and nonconservative at others, which means the final sum of buckled load combinations can not be used a result by it self. A comparison between what load combinations R_i buckling occurs, in the fitted and true stability boundary, is needed. For the fitting method to work it needs to be able to capture all buckled load combinations, and then the precision of the fit can be measured by the green points. Fewer green points means a more precise fit. Red points means the fit is erroneous.

6.4.1 Position of the points

In this section the effect of the position of the sampling points are examined. For these load configurations the true number of unstable load combinations were found in ANSYS to be 31. To understand the effect, two sampling methods are compared. First the results for 8 sampling points created with the equally spaced method is calculated for the different fitting types, which is shown in Table 6.1.

| | \bullet | \bullet | \bullet |
|------------------------------------|-----------|-----------|-----------|
| Convex Hull | 3 | 31 | 0 |
| Linear Interpolation | 3 | 31 | 0 |
| Cubic Interpolation | 3 | 31 | 0 |
| Polynomial | 0 | 21 | 10 |
| The Local Regression Method | 0 | 24 | 7 |
| Biharmonic | 1 | 2 | 29 |

Table 6.1: 8 Equally spaced sampling points.

In the first 3 fitting methods (convex, linear, and cubic), 8 equally spaced sampling points predicts 34 buckled load combinations, where only 3 critical combinations are conservative according to the true boundary found in ANSYS. These results are regarded as a good approximation. The cubic interpolation of the stability surface for the 8 equally spaced sampling points is shown in Figure 6.15, where it is noticed that fit seems reasonable, since it yields a surface covering the load space.

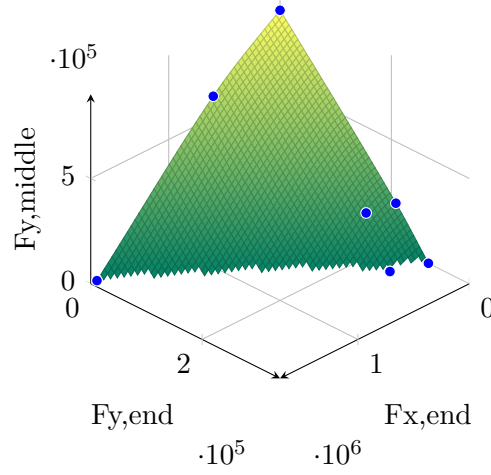


Figure 6.15: Cubic interpolation with 8 equally spaced sampling points.

The last 3 fitting methods (Polynomial, Local Regression and Biharmonic) in Table 6.1 are erroneous since they have red points. If the random sampling method with 8 points is used, then different results are obtained, compared to the equally spaced method, as seen in Table 6.2.

| | ● | ● | ● |
|------------------------------------|----|----|----|
| Convex Hull | 52 | 31 | 0 |
| Linear Interpolation | 31 | 31 | 0 |
| Cubic Interpolation | 31 | 31 | 0 |
| Polynomial | 0 | 26 | 5 |
| The Local Regression Method | 0 | 25 | 6 |
| Biharmonic | 3 | 6 | 25 |

Table 6.2: Random 8 points.

The first 3 fitting methods now predicts more unstable load combinations, according the true boundary. This means that the position of the sampling points have a significant impact on the precision of the fitted stability boundary. However, for the last 3 fitting methods, the results have become more precise than for the equally space sampling with 8 points.

A visual inspection of the fitting, seen in Figure 6.16a, shows that reason for this imprecision (green points) of the first 3 fitting methods, is because the fit only covers a part of the stability boundary. The individual buckling loads have to be included since they are the exterior points, for the fitting to cover the whole quadrant. The last 3 methods are able to cover more of the stability boundary, without exterior points, as seen for the polynomial fit in Figure 6.16b.

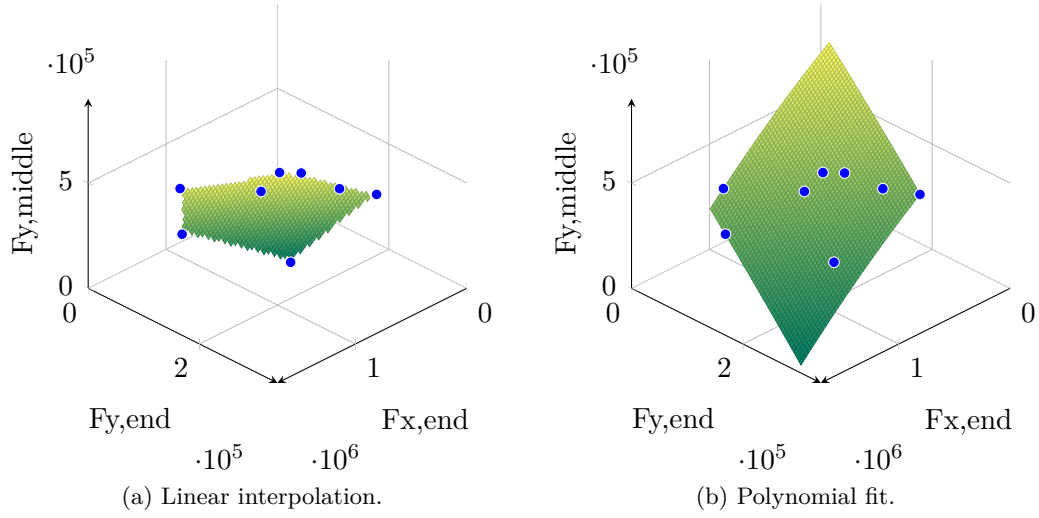


Figure 6.16: 8 random sampling points.

The same tendency is seen for the first 3 fitting methods with a cluster sampling method. However, the local regression fit and polynomial fit are converged with the true boundary as seen in Table 6.3, since they predict no false buckling.

| | ● | ● | ● |
|------------------------------------|----|----|----|
| Convex Hull | 47 | 31 | 0 |
| Linear Interpolation | 66 | 31 | 0 |
| Cubic Interpolation | 66 | 31 | 0 |
| Polynomial | 0 | 31 | 0 |
| The Local Regression Method | 0 | 31 | 0 |
| Biharmonic | 42 | 21 | 10 |

Table 6.3: 8 cluster sampling points without corrections factors.

6.4.2 Correction Algorithm

The equally spaced and central composite design sampling points methods are based on the idea that equally spaced points will yield the best fitting result. However a problem arises when the critical points are calculated. The problem is that the critical points are not equally spaced as the sampled points used to calculate the critical points. This is because of the difference in the individual buckling values of each load. The problem is exemplified in Figure 6.17, where an equally spaced sampling point is shown, along with an equally spaced critical point.

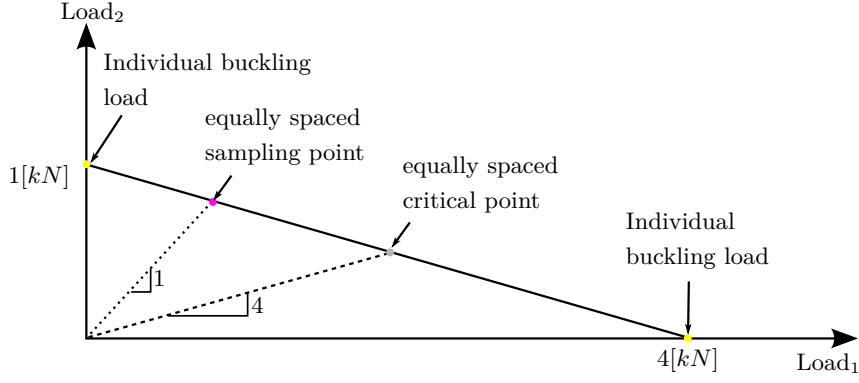


Figure 6.17: Example of equally spaced sampling point and equally spaced critical point.

In order for the critical points to be equally spaced, the axis needs to be normalized according to their buckling values. An algorithm is constructed which calculates the individual buckling loads and finds the smallest, then it calculates the ratio of the individual buckling loads with the smallest. The ratio is thus the correction factor. The result of the algorithm is shown in Figure 6.16, where Figure 6.18a is the initial equally spread sampling points, and Figure 6.18b is the equally spread sampling points with a correction factor.

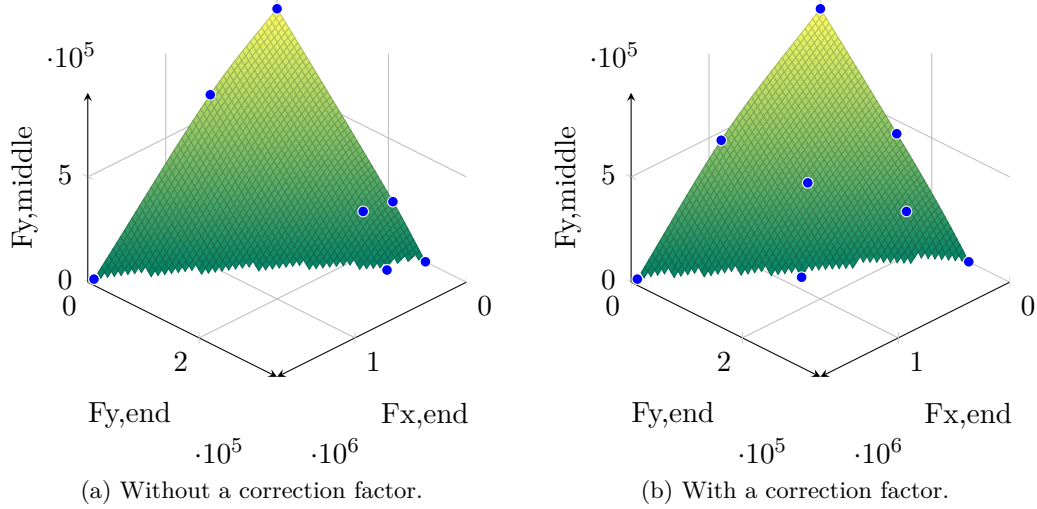


Figure 6.18: CCD 4 points with a correction factor.

This implementation improves the precision of several fitting methods, which can be seen in Table 6.5 and Table 6.4.

| | ● | ● | ● |
|------------------------------------|---|----|----|
| Convex Hull | 3 | 31 | 0 |
| Linear Interpolation | 3 | 31 | 0 |
| Cubic Interpolation | 3 | 31 | 0 |
| Polynomial | 0 | 21 | 10 |
| The Local Regression Method | 0 | 24 | 7 |
| Biharmonic | 1 | 2 | 29 |

Table 6.4: 8 equally spaced sampling points without corrections factors.

| | ● | ● | ● |
|------------------------------------|---|----|----|
| Convex Hull | 2 | 31 | 0 |
| Linear Interpolation | 2 | 31 | 0 |
| Cubic Interpolation | 1 | 31 | 0 |
| Polynomial | 0 | 31 | 0 |
| The Local Regression Method | 2 | 31 | 0 |
| Biharmonic | 2 | 12 | 19 |

Table 6.5: 8 equally spaced sampling points with corrections factors.

For cubic interpolation, it is noticed how the false number of buckled combinations (green points), decreases from 3 in Table 6.4, to 1 in Table 6.5. As seen in the tables the number of green points is reduced, which means the approximated boundary is closer to the true boundary.

The cluster sampling point method is a statistical analysis of the loading data. Thus a correction factor is not needed for the cluster method, since the difference in the load ratio is considered with statistical analysis.

| | ● | ● | ● |
|------------------------------------|----|----|----|
| Convex Hull | 60 | 31 | 0 |
| Linear Interpolation | 67 | 31 | 0 |
| Cubic Interpolation | 67 | 31 | 0 |
| Polynomial | 0 | 0 | 0 |
| The Local Regression Method | 0 | 0 | 0 |
| Biharmonic | 30 | 12 | 19 |

Table 6.6: 4 cluster sampling points without corrections factors.

| | ● | ● | ● |
|------------------------------------|---|----|----|
| Convex Hull | 4 | 31 | 0 |
| Linear Interpolation | 4 | 31 | 0 |
| Cubic Interpolation | 4 | 31 | 0 |
| Polynomial | 0 | 0 | 0 |
| The Local Regression Method | 0 | 0 | 0 |
| Biharmonic | 0 | 4 | 27 |

Table 6.7: ccd 4 points with correction factor.

However the cluster method is not as precise with few points (see Table 6.6), as the central composite design with a correction factor (see Table 6.7).

6.4.3 Performance of the Fitting Methods

The different fitting methods explained in Section 6.2, are compared according to precision of the boundary surface. Only the relevant results are shown. The fitting methods yields a different precision for each sampling method. The first method to converge for the smallest amount of sampling points is the polynomial fit.

| | ● | ● | ● |
|------------------------------------|---|----|----|
| Convex Hull | 2 | 31 | 0 |
| Linear Interpolation | 2 | 31 | 0 |
| Cubic Interpolation | 1 | 31 | 0 |
| Polynomial | 0 | 31 | 0 |
| The Local Regression Method | 2 | 31 | 0 |
| Biharmonic | 2 | 12 | 19 |

Table 6.8: 8 equally spaced sampling points with correction factor.

As seen in Table 6.8 the polynomial fitting is converged with the true boundary, with only 8 sampling points, both for the equally spaced sampling points and cluster sampling points (Table 6.3). However, because polynomial fit and local regression fitting methods requires at least 8 number of points, and hence they does not yield any result for central composite design with 4 points, which is seen Table 6.7. For a small number of points, convex hull, linear interpolation and cubic interpolation are more precise than the biharmonic method. In Table 6.7 it is seen that the biharmonic interpolation is erroneous, because of 27 red points. The first 3 fitting methods in Table 6.7 gives an reasonable approximation of the stability boundary, considering only 4 points are used to created the boundary. If the sampling method used is random then last three fitting methods; polynomial, local regression and biharmonic, converges faster than then first 3 fitting methods. This is seen from the result, in Table 6.9.

| | ● | ● | ● |
|------------------------------------|----|----|---|
| Convex Hull | 8 | 31 | 0 |
| Linear Interpolation | 21 | 31 | 0 |
| Cubic Interpolation | 21 | 31 | 0 |
| Polynomial | 0 | 31 | 0 |
| The Local Regression Method | 0 | 31 | 0 |
| Biharmonic | 0 | 31 | 0 |

Table 6.9: Random sampling of 64 points with correction factor.

The last 3 fitting methods are not as dependant on the position of the points, as convex hull, linear interpolation and cubic interpolation. Thus have converge faster with a random sampling method. All the fitting methods have converged with the true boundary found in ANSYS, with 64 equally spaced sampling points with the correction algorithm, as seen in Table 6.10

| | ● | ● | ● |
|------------------------------------|---|----|---|
| Convex Hull | 0 | 31 | 0 |
| Linear Interpolation | 0 | 31 | 0 |
| Cubic Interpolation | 0 | 31 | 0 |
| Polynomial | 0 | 31 | 0 |
| The Local Regression Method | 0 | 31 | 0 |
| Biharmonic | 0 | 31 | 0 |

Table 6.10: combinations 64 points.

There are no green or red points or red points present for any of the fitting methods which means they are approximately close to the true boundary.

The polynomial fitting is the first to converge with the true boundary. The result of the polynomial interpolation is shown Figure 6.19.

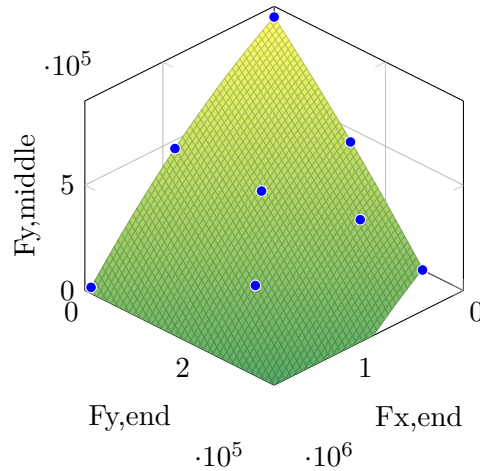


Figure 6.19: Polynomial fit of 8 equally spaced sampling points.

For the load configuration of positive $F_{x,end}$, $F_{y,end}$ and $F_{y,middle}$, 64 equally spaced sampling points are able to represent the stability boundary for 100 load combinations for all fitting methods, but the polynomial fit required only 8 points. This means the number of ANSYS calculations needed in order to analyse the loading data w.r.t. buckling has been reduced from 100 analyses to 8.

If the number of 100 load combinations are increased, then 64 sampling points might not be enough, for the fitting methods which contain local linear polynomials (convex hull and linear interpolation).

Even though the polynomial fit converges with fewer points, it does in some cases produce erroneous results with respect to the true stability boundary, as seen in Table 6.11.

| | ● | ● | ● |
|------------------------------------|----|----|----|
| Convex Hull | 38 | 31 | 0 |
| Linear Interpolation | 57 | 31 | 0 |
| Cubic Interpolation | 56 | 31 | 0 |
| Polynomial | 1 | 30 | 1 |
| The Local Regression Method | 0 | 30 | 1 |
| Biharmonic | 15 | 7 | 24 |

Table 6.11: Random sampling of 8 points with correction factor.

Polynomial fit is allowed to have residuals at the fitting points, i.e. it does not necessarily pass through all fitting points. Without the results from ANSYS, it is hard to estimate if polynomial fit produces erroneous results or conservative results. The polynomial fit involves a risk.

The fitting methods have also been tested for the other seven quadrants of the 3 dimensional stability boundary. The most crucial result is found when the load configuration is changed to $-F_{y,middle}$, $F_{y,end}$ and $F_{x,end}$. In Table 6.12, the results from different fitting methods is shown for the new load configuration.

| | ● | ● | ● |
|------------------------------------|----|----|---|
| Convex Hull | 1 | 10 | 0 |
| Linear Interpolation | 6 | 10 | 0 |
| Cubic Interpolation | 6 | 10 | 0 |
| Polynomial | 23 | 26 | 7 |
| The Local Regression Method | 5 | 10 | 0 |
| Biharmonic | 3 | 2 | 8 |

Table 6.12: 37 equally spaced sampling points.

Even though it is only last 3 fitting methods, which produces erroneous results (red points), a visual inspection of the fitting methods reveals an error in all fitting methods, except convex hull. In Figure 6.20 it is seen that the linear interpolation fails to fit the stability

boundary compared to convex hull.

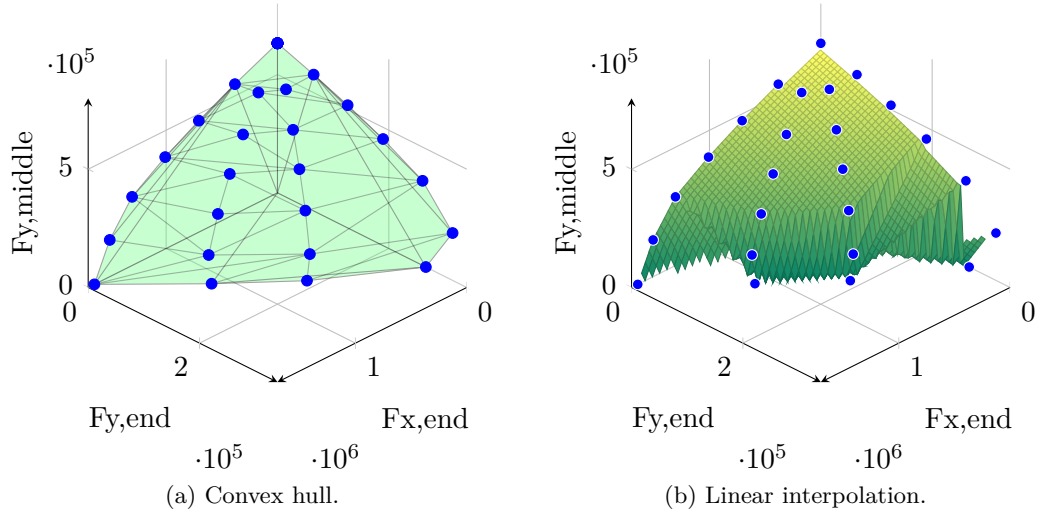


Figure 6.20: 37 equally spaced sampling points.

As seen in Figure 6.20b the fitting methods makes a wrong boundary when $Fx, end = 0$, which is due to an inflection point on the boundary. The fitting points are no longer unique to the axes, because of the inflection points. Increasing the number of points or using a higher order polynomial does not make the fit correct. This is shown in 2D for the stability line at $Fx, end = 0$ in Figure 6.21, where 2 polynomials fail to represent the stability boundary.

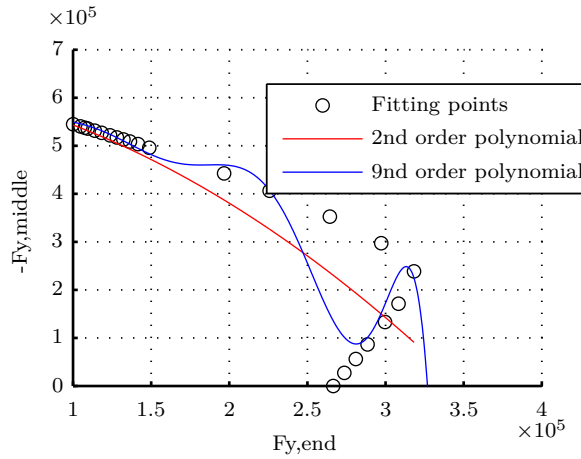


Figure 6.21: Curve fitting failure due to an inflection point.

The interpolation methods (linear, cubic and biharmonic) and the local regression cannot be performed, since the fitting points are not unique to Fy, end . This means that convex hull is the only fitting method that will converge with the true stability boundary. In Table 6.13 it seen how convex hull, have converged but all other fitting methods fail to.

| | ● | ● | ● |
|------------------------------------|---|----|---|
| Convex Hull | 0 | 10 | 0 |
| Linear Interpolation | 4 | 10 | 0 |
| Cubic Interpolation | 6 | 10 | 0 |
| Polynomial | 5 | 11 | 4 |
| The Local Regression Method | 5 | 10 | 0 |
| Biharmonic | 7 | 6 | 4 |

Table 6.13: 109 equally spaced sampling points with a correction factor.

6.5 Summary

The polynomial fit was the fastest converging with the true stability boundary when no inflection points exist on the boundary. The polynomial fit performs better, as seen in Table 6.1, but the fitting method also produces critical errors, as seen in Table 6.2

Without a correction factor the cluster sampling method, was the most precise, with a polynomial and a local regions fit.

The worst fitting method was the biharmonic interpolation, since it produced erroneous results (red points) and converges slower then the polynomial fit and the local regression method.

With one or more inflection points on the boundary, all the fitting methods fails except convex hull.

The correction factor should be used for all sampling methods, except the cluster sampling method. When using central composite design as sampling method along with the correction factor, a lot of load combinations are disregarded, compared to a few sampling points.

The individual buckling loads should be added when fitting since they often are the exterior points of the stability boundary. The random sampling method failed to cover the stability boundary properly, because of the lack of exterior points. The individual loads are calculated first in order to find the correction factors. The sampling points are multiplied with the correction factors.

All fitting methods converged when no inflection points are present, with 64 equally spaced sampling points and a correction factor. However when inflection points are present only the convex hull algorithm converged with 109 equally spaced sampling points and a correction factor. Convex hull is the safest of the fitting methods, since it never overestimates the stability boundary. 109 is a lot of sampling points for 3 load configurations. The most beneficial way to use the method of data fitting of a stability boundary is concluded to be with the following settings:

- Central composite design

- Correction factor
- Convex hull

By using this setup a coarse approximation can be made, which in this case reduces the number of load combinations from 100 to 35, with only 4 sampling points. The eigenvalue or a Cholesky factorization shall be performed to find the remaining load combinations.

These settings for the method is used on the 5 load configurations and loading data, which is described in Section 4.1. The central composit design produces 26 sampling points for 5 load configurations. By use of convex hull and the correction factor, the number of load combinations was reduced from 100 to 22.

Discussion 7

This thesis has presented 4 different suggestions to how the total solution time of the linear buckling analysis can be decreased. The method that has proven the most effective, is Method 2 which uses the principle of superposition in collaboration with the Cholesky factorization. However, all method has proven to decrease the solution time, in one or more situations. It is furthermore important to note that the methods, calculated the same buckling loads.

Generally the results displayed the expected behaviors, which could be seen from the flowcharts of the different methods. Minor divergence between the results was however experienced. The causes of this could be many, and is very hard to determine with certainty. Some of the divergence properly comes from the computer calculating the buckling analysis. If the temperature of the CPU rises or falls, the solution time might become slower and faster. It was not possible to strictly control the environment in which the calculations was performed, and a bit of divergence in the results was hence expected. Some results was unexpected. of of the unexpected results is for Method 1, which uses the superposition principle. This method displayed a slower calculation time, than the benchmark method in some situations. It was discovered that the reason for the unexpected behavior, comes from the calculation of the eigenvalue problem in MATLAB. the method should therefore significantly faster, if this problem can be overcome. This could be by implementing a new solver in MATLAB, or using another program similar to MATLAB, with the same features.

It should be noted that all the results obtained in this thesis are only valid for the current settings used. The behavior of having more elements, a higher number of layers, different material properties in the layers ect. cannot be accounted for by this thesis.

The fitted stability boundary comes very close to the true stability boundary when a lot of sampling points are used. The data fitting of stability boundary could be applied to nonlinear. However, the fact that the convex hull method is an conservative approach, is based on the theorem, which states that the stability boundary cannot be concave, is only valid for linear buckling analyses. This means that the convex hull method should be used with care, in nonlinear buckling analyses.

A combination of Method 2 and data fitting of a stability boundary, would be of interest in future work, since it has the potential of reducing the total solution time even more than Method 2.

Buckling of the model used in this thesis

The term buckling, does for most people imply total failure of a structure, i.e the structure loses all stability, but this is not always the case. Buckling can have different effects on different structure. Local buckling in a composite structure might lead to delamination and cause serious damage, where the same local buckling in a structure constructed with steel, does not create damage to the structure.

In this thesis only the linear buckling analysis, which is an eigenvalue problem, is calculated. It would however have been informational to run a couple of nonlinear analysis in the beginning of the work. This could give informations that is not obtained with the linear buckling analysis, such as if the structure will lose all stability when reaching the buckling load, or the buckling is local and does not display any risk to the total stability of the structure. The behavior of the linear vs. the nonlinear, when multiple loads are applied to the structure might also be beneficial to know, to see if the nonlinear and the linear predicts the same behavior for the structure with multiple loads applied.

Stand alone program / Add on program

The methods in this thesis is developed with the possibility to use them in a stand alone program or as a add on to an existing commercial software. The step towards creating this stand alone program in MATLAB, is small. Since the methods already are implemented in MATLAB, only small modifications such as defining how the loads and number of loads is applied to all kind of structures.

Conclusion 8

In the introduction in Chapter 1 the need for a faster method was stated, based on the project proposal from [Laustsen 2011]. This need is met, by the development of 3 different methods, that each has decreased the total solution time of the linear buckling finite element analysis. Hence it is now possible to calculate the linear buckling analysis faster.

Method 1

In Method 1 the possibility of superpositioning the stress stiffness matrix $[K_\sigma]$ is used, to avoid calculating the entire solution procedure of the buckling finite element analysis for all load combinations. Method 1 does only need to calculate the buckling finite element analysis solution procedure for each of the load configurations. This should in theory save a huge amount of calculation time, but due to negative eigenvalues obtained in MATLAB, when solving the eigenvalue problem, a shift had to be made. Hence the eigenvalue problem had to be calculated twice for some of the load combinations. This limited the decrease in solution time for Method 1. However, the method has decreased the solution time, if the structure is constructed of 100 layers, and 6600 elements, and 101 load combinations are calculated.

Method 2

In Method 2 the total solution time is decreased by using the superpositioning of the stress stiffness matrix $[K_\sigma]$ and a Cholesky factorization of the tangent stiffness matrix $[K_T]$. By doing this, the calculation of the entire solution procedure for the finite element buckling analysis, for all load combinations are avoided, as in Method 1. However, instead of calculating the eigenvalue, a Cholesky factorization is used to determine if the load combinations buckle, based on the definiteness of $[K_T]$. This saves a large amount of time, because the eigenvalue problem does not have to be solved. A decrease in the total solution time with respect to the presently used method of up to 78 % is obtained with Method 2. However, the Cholesky factorization does not give any information about the magnitude of instability or stability. The only information obtained is whether or not the structure will buckle.

Data Fitting of Stability Boundary

The number of load combinations was reduced from 100 to 22. This was achieved using a convex hull to create the stability boundary, from 26 sampling points made with the central composite design method. The other fitting methods failed when an inflection point was present on the stability boundary. Convex hull is an conservative approach, but converges with enough points. Central composite design sampling with correction factors gave the best approximation per sampling points.

The statement concerning the methods should be able to implementations by an add-on or stand alone program, is also fulfilled. However, this is not done in this thesis, but minor changes has to be made to the MATLAB-script in order to make a stand alone program that uses the ANSYS for the calculation of stiffness matrices.

These methods gives the engineer the opportunity to calculated more load combinations, and reducing the risk of not selection a possible critical load combination. The methods can be used to find the load combinations, that shall undergo a full geometric nonlinear buckling analysis.

Bibliography

- Z. Bažant & L. Cedoliin (2010). Stability of Structures: Elastic, Inelastic, Fracture and Damage Theories. World Scientific.
- E. Cheney & D. Kincaid (2007). Numerical mathematics and computing. Brooks/Cole.
- R. Cook (2001). Concepts and applications of finite element analysis. Wiley.
- M. De Berg, et al. (2008). Computational Geometry: Algorithms and Applications. Springer.
- A. de Faria & J. Hansen (2001). ‘On buckling optimization under uncertain loading combinations’.
- Dictionary.com (2012). ‘Benchmark’. <http://dictionary.reference.com/browse/benchmark>.
- C. Felippa (2007). ‘Nonlinear Finite Element Methods’. <http://www.colorado.edu/engineering/cas/courses.d/NFEM.d/>.
- C. Felippa (2008). ‘Introduction to Finite Element Methods’. <http://www.colorado.edu/engineering/cas/courses.d/IFEM.d/>.
- G. L. I. S. GmbH (2010). Guideline for the Certification of Wind Turbines. Germanischer Lloyd, Hamburg.
- K. Huseyin (1975). Nonlinear theory of elastic stability. NOORDHOFF.
- S. Laustsen (2011). Efficient Buckling Analysis of Wind Turbine Blades with Multiple Load Cases. Project Proposal - SUZLON.
- E. Lund & E. Lindgaard (2011). ‘Lecture notes: Course on Finite Element Methods’.
- C. Wang, et al. (2004). Exact Solutions for Buckling of Structural Members. CRC Series in Computational Mechanics and Applied Analysis. CRC Press.

Results from performance studies



A.1 Data from Benchmark, Method 1 and Method 2

| | 21 combinations | 61 combinations | 101 combinations |
|--------------------------|-----------------|-----------------|------------------|
| 1 Layer, 884 elements | 112.033 | 167.961 | 198.627 |
| 1 Layer, 3216 elements | 393.442 | 549.047 | 787.855 |
| 1 Layer, 6600 elements | 837.869 | 1240.35 | 1852.17 |
| 50 Layer, 884 elements | 318.251 | 478.976 | 568.262 |
| 50 Layer, 3216 elements | 1137.58 | 1572.81 | 2406.39 |
| 50 Layer, 6600 elements | 2417.43 | 3770.31 | 4957.33 |
| 100 Layer, 884 elements | 522.729 | 704.001 | 929.404 |
| 100 Layer, 3216 elements | 1876.8 | 2855.34 | 3721.04 |
| 100 Layer, 6600 elements | 3979.32 | 5790.13 | 8288.84 |

Table A.1: Benchmark

| | 21 combinations | 61 combinations | 101 combinations |
|--------------------------|-----------------|-----------------|------------------|
| 1 Layer, 884 elements | 128.1972 | 145.5164 | 158.5157 |
| 1 Layer, 3216 elements | 651.822 | 686.656 | 798.334 |
| 1 Layer, 6600 elements | 1509.746 | 1622.109 | 1907.97 |
| 50 Layer, 884 elements | 316.1173 | 337.1345 | 363.404 |
| 50 Layer, 3216 elements | 1701.045 | 1707.15 | 1947.825 |
| 50 Layer, 6600 elements | 3957.288 | 4001.636 | 4648.572 |
| 100 Layer, 884 elements | 506.6277 | 534.9525 | 578.5104 |
| 100 Layer, 3216 elements | 2751.779 | 2769.841 | 3068.334 |
| 100 Layer, 6600 elements | 6489.749 | 6465.665 | 7313.264 |

Table A.2: Method 1

| | 21 combinations | 61 combinations | 101 combinations |
|---------------------------------|------------------------|------------------------|-------------------------|
| 1 Layer, 884 elements | 49.62 | 81.0535 | 118.884 |
| 1 Layer, 3216 elements | 233.506 | 454.27 | 689.159 |
| 1 Layer, 6600 elements | 705.641 | 1413.488 | 2438.619 |
| 50 Layer, 884 elements | 64.6192 | 94.8612 | 129.7137 |
| 50 Layer, 3216 elements | 272.451 | 481.999 | 743.271 |
| 50 Layer, 6600 elements | 699.684 | 1286.056 | 2522.515 |
| 100 Layer, 884 elements | 70.2363 | 101.901 | 137.494 |
| 100 Layer, 3216 elements | 333.853 | 558.629 | 797.549 |
| 100 Layer, 6600 elements | 924.656 | 1472.942 | 2529.024 |

Table A.3: Method 2