Frequency Response Analysis using Adaptive Component Mode Synthesis

Tor Troeng

November 7, 2010

Abstract

Solutions to physical problems described by Differential Equations on complex domains are in except for special cases almost impossible to find. This turns our interest toward numerical approaches. Since the size of the numerical models tends to be very large when handling complex problems, the area of model reduction is always a hot topic. In this report we look into a model reduction method called Component Mode Synthesis. This can be described as dividing a large and complex domain into smaller and more manageable ones. On each of these subdomains, we solve an eigenvalue problem and use the eigenvectors as a reduced basis. Depending on the required accuracy we might want to use many or few modes in each subdomain, this opens for an adaptive selection of which subdomains that affects the solution most. We cover two numerical examples where we solve Helmholtz equation in a linear elastic problem. The first example is a truss and the other a gear wheel. In both examples we use an adaptive algorithm to refine the reduced basis and compare the results with a uniform refinement and with a classic model reduction method called Modal Analysis. We also introduce a new approach when computing the coupling modes only on the adjacent subdomains.
1 Acknowledgements

First of all I would like to thank my Supervisor Professor Mats G. Larson for all the input and bright ideas before and during the project. I also like to express my greatest gratitudes to Håkan Jakobsson for taking time answering my endless amount of questions and make me understand at least some of the mysteries in mathematics. Thanks to Dr. Fredrik Bengzon for all the contribution in starting the project and the inspiration through the way. I would also like to thank all my other colleagues at the Department of Mathematics and Mathematical Statistics. Dr. August Johansson, Robert Söderhund, Karl Larsson and Per Westerlund for helping me and always keeping a good and friendly atmosphere, it is an honor to work with you.
Contents

1 Acknowledgements 2

2 Introduction 4
  2.1 Background 4
  2.2 Structural analysis 5
  2.3 Frequency response analysis 5
  2.4 Finite Element Method 6
  2.5 Model reduction 8

3 Problem and method description 10
  3.1 Helmholtz equation 10
  3.2 Component Mode Synthesis 11
    3.2.1 1-dimensional case 11
    3.2.2 General Setting 14
    3.2.3 Sparsity pattern of the mass and stiffness matrix 15
  3.3 Adaptivity 17
    3.3.1 Refinement strategy 19
  3.4 Beam Theory 21
  3.5 Local vs global interface 23

4 Application 26
  4.1 Linear Elasticity 26
  4.2 The equations to be considered 26
  4.3 Numerical examples 28
    4.3.1 The Truss 28
    4.3.2 The gear wheel 32

5 Conclusions and discussion 36
  5.1 Local vs global interface 36
  5.2 Numerical examples 36
  5.3 Future and on going work 37
2 Introduction

2.1 Background

Mathematics has been a topic of interest since the beginning of mankind. The fundamentals of today’s rigorous theories has been developed through thousands of years. An important application, and also a motivation for development, has been to understand and describe the fundamental laws of nature, such as Newton’s famous second law. Differential equations has been around since the 17th century and both Newton and Leibniz made great contributions to the development of the theory. Working ourself through history we end up in today’s era of numerical calculations, which is based upon the groundwork built up through time. Since it is often impossible to find analytical solutions to real world differential equations, such as those involved in a weather forecast, finding the stress exerted on a bridge, Schrödinger equation and so on so forth, we are bound to use numerical methods to search for a solution in an approximative way.

Among the numerical methods the Finite Element Method (FEM) is common for a range of different applications such as structural engineering, heat conduction, convection-diffusion problems and many more. FEM is based upon dividing a domain into small elements, and stating conditions for how they should interact. This is especially useful when solving partial differential equations on complex geometries. Since time and money are crucial factors when solving this problems, a lot of efforts are made to make numerical simulations and models as precise as possible before making a real model. A numerical problem in the industry tends to be very large, sometimes in order of 10 million degrees of freedom, which is hard to simulate even for a super computer. This opens for model reduction. Model reduction is basically a technique for reducing complex problems and still preserve the fundamental dynamics of the system. One way of doing this reduction is to use Component Mode Synthesis (CMS). CMS is a classical method for model reduction of FEM models in linear elasticity and was first introduced in the 60’s by Hurty[1]. The method is based on dividing the computational domain into subdomains and an interface, solve an eigenvalue problem on the subdomains and on the interface, then use the eigenmodes as a basis in Galerkin’s method. The amount of eigenmodes to use is a balance between precision and computer efficiency and since both are important factors we use an adaptive choice of eigenmodes. The adaptive refinement is based on the theory presented in [2]. To sum up the type of problem we have considered in this report is a physical problem which is described by a partial differential equation. Since we are interested to solve the equation on a complex domain it can not be solved analytically and we therefor search for a numerical solution using FEM. Since FEM tends to produce large system of equations we will use CMS as a model reduction method to decrease the
degrees of freedom.

2.2 Structural analysis

Structural analysis is a way to study and predict the behaviour of structures due to a predefined load of various kinds, this can for example be a uniformly distributed load such as snow on a roof. There are three main types of analysis to be made, analytical, numerical and experimental. Analytical studies are restricted to simple models which can often be solved by hand. Experimental frequency analysis are based on finding the natural modes with various types of vibration testing. None of those methods will be further investigated in this report, instead we will focus on the numerical analysis.

Numerical analysis is based on the same mathematical equations as the analytical. Among the numerical methods FEM, further described in section 2.4, is the most common and is widely used. This approach is very effective on complex geometries and combines physical equations with mathematical theory. Structural analysis in the industry is used on a wide range of geometries such as aircrafts, buildings, bridges, cars and so on. This puts high demand on the resolution of the model and makes the numerical models large and complex, often in the order of $10^6$ degrees of freedom.

Modal analysis is one branch of structural analysis and was first used in the 40’s as a tool for understanding the dynamics of an aircraft[3]. The goal of the method is to find the natural modes of the structures and then use those modes for further calculations or for finding critical frequencies. This can be done be actual measurements (experimental approach) or it can be done numerically. An advantage in using numerical testing instead of experimental is that you only need a theoretical model, assumptions of boundary conditions and material parameters. This makes it easier to set up and, if necessary, change an in-accurate model, which saves both time and money.

2.3 Frequency response analysis

Frequency response analysis (FRA) is a method for finding crucial behaviours of a structure in a given frequency spectrum. For example the famous Tacoma Narrows Bridge which were exposed by a wind making it vibrate near its natural frequency and suddenly collapse. To avoid this kind of disasters FRA is an important tool, making it possible to test the structure for a spectrum of interest either with a numerical approach or by actual measurements. In frequency response for structural analysis we consider the Helmholtz equation, further described and derived in section 3.1. This equation can vary depending of which system to be considered, by either adding or removing a force or a damping the model can be formed to best fit the real
problem. Often one wants to cover a wide spectrum of frequencies and if we want to use a numerical approach this puts high demands on the efficiency of the implemented algorithm and also often requires a model reduction method. In our case we use Component Mode Synthesis. The numerical section of this report are focused on making FRA on different geometries on both a spectrum and for specific frequencies.

2.4 Finite Element Method

The main idea of all numerical methods is to find an approximative discrete solution to a continuous problem. The Finite Element Method is a numerical method for solving partial differential equations, (PDE:s). For simplicity we look at the Poisson equation

\[ -\Delta u(x) = f(x) \quad x \in \Omega \]  
\[ u(x) = 0 \quad x \in \Gamma \]  

Figure 1: An arbitrary domain \( \Omega \) with the boundary \( \Gamma \)

We are seeking a solution to the function \( u \) on the domain \( \Omega \)

Now we introduce the linear space

\( V = \{ v: v \text{ is continuous on } \Omega, \nabla v \text{ is piecewise continuous and bounded on } \Omega \text{ and } v = 0 \text{ on } \Gamma \} \)

Reformulate the differential equation on variational form.

Find \( u(x) \in V \) such that

\[ \int_{\Omega} \nabla v \cdot \nabla u = \int_{\Omega} vf, \quad \forall v \in V \]  

(2.3)

Next step is to discretize the problem.

Let \( \{ \varphi_i \}_{i=1}^N \) be the piecewise continuous polynomial functions on a mesh of \( \Omega \) with \( N \) nodes. A node is a point on the domain where the elements are jointed and also there we calculate our coefficients.

The finite element approximation now reads:

Find \( u_h(x) \in V_h \) such that

Find \( u_h(x) \in V_h \) such that
\[ \int_{\Omega} \nabla v \cdot \nabla u_h = \int_{\Omega} v f \quad \forall v \in V_h \quad (2.4) \]

where \( V_h \subset V \)

Now let \( u_h = \xi_j \varphi_j \), where \( \xi_j \) are \( N \) unknown coefficients. And \( v = \sum_{i=1}^{N} \varphi_i \) are basis functions such that.

Substitute into (2.4) we get

\[ \sum_{j=1}^{N} \int_{\Omega} \nabla \varphi_j \cdot \nabla \varphi_i \xi_j = \int_{\Omega} \varphi_i f \quad i = 1, ..., N \quad (2.5) \]

In matrix form the linear system (2.5) can be expressed as

\[ A\xi = b \]

where

\[ A_{ij} = \int_{\Omega} \nabla \varphi_j \cdot \nabla \varphi_i \quad (2.6) \]

\[ b_j = \int_{\Omega} \varphi_j f \quad (2.7) \]

Now we have a linear system of equations to solve. The matrix \( A \) is symmetric and positive definite which means that we always can find a unique solution to the problem. Since the basis functions \( \varphi_i \) only have support in the close surroundings of the node \( i \) the matrix \( A \) is sparse. This is a very important feature of FEM since it makes it possible to handle large problems.

An important question for the solution \( u_h \) is how much it differs from the continuous solution \( u \). Lets put \( e = u - u_h \), the difference between the analytical solution and the finite element solution. The following theorem states that the solution \( u_h \) is the best approximation to \( u \) in \( V_h \). [4]

Let \( \| \cdot \| \) denote the norm

\[ \| w \| = \left( \int \omega^2 \right)^{1/2} \quad (2.8) \]
Theorem 1. For any $v \in V_h$ we have
\[
\|e\| = \|(u - u_h)\| \leq \|(u - v)\| \tag{2.9}
\]

2.5 Model reduction

Model reduction is basically a technique for reducing complex problems, i.e. reduce the DOFs, and still preserve the fundamental dynamics of the system. A common technique to perform a model reduction is to project the complex model onto a lower order subspace, provided that we know this subspace. In figure 3 this method is explained. In this approach we use a subspace $R$ and projects the solution $X$ onto the subspace. And for example in CMS the subspace $R$ is built up with eigenmodes. In the full problem we have to solve a $n \times n$ system of equations and in the reduced problem we have reduced the size of the problem in to $m \times m$, where $m < n$. A requirement for this to be efficient is of course that the computational cost to make the extra matrix operations is cheaper than computing the full problem.

\[
\begin{align*}
A & \quad \xi = F \\
n \times n & \quad n \times 1 & \quad n \times 1
\end{align*}
\]

The full problem

\[
\begin{align*}
\xi & = A^{-1} - F \\
n \times 1 & \quad n \times n & \quad n \times 1
\end{align*}
\]

Reduced problem

\[
\begin{align*}
R^T & \quad A \quad R = A_R \\
m \times n & \quad n \times n & \quad m \times m
\end{align*}
\]

\[
\begin{align*}
F & = \xi R^{-1} \\
m \times 1 & \quad m \times 1
\end{align*}
\]

Figure 3: A schematic figure of how model reduction works.
The size of the subspace may vary depending on accuracy and computational cost. Three parameters to always keep in mind when working with reduced models are. Accuracy, computational cost and implementation time. They are of course in conflict, and a compromise is often necessary [5]
3 Problem and method description

The problem I have focused on is to implement Component Mode Synthesis as a method to solve Frequency response analysis in linear elasticity. This is interesting since a lot of the problems solving this type of equations ends up in time consuming algorithms and very large problems. Using model reduction, in this case CMS, enables us to faster and with less memory consuming find reduced solutions to the problem.

3.1 Helmholtz equation

Helmholtz equation is an elliptic partial differential equation and often occurs in physical problems involving time and space. We can derive the Helmholtz equation from the wave equation by assuming that $U$ is time-harmonic.

The wave equation:

$$\nabla^2 U(r, t) - \frac{1}{c^2} \frac{\partial^2}{\partial t^2} U(r, t) = 0$$  \hspace{1cm} (3.1)

We start by assuming that $U$ is time-harmonic.

$$U(r, t) = U(r) e^{i\omega t}$$  \hspace{1cm} (3.2)

Which gives that:

$$\frac{\partial^2}{\partial t^2} U(r, t) = -\omega^2 U(r) e^{i\omega t}$$  \hspace{1cm} (3.3)

$$\nabla^2 U(r, t) = \nabla^2 U(r) e^{i\omega t}$$  \hspace{1cm} (3.4)

Inserted in 3.5 and dividing with $e^{i\omega t}$ results in what we call Helmholtz equation.

$$\nabla^2 U(r) + k^2 U(r) = 0 \hspace{1cm} k = \frac{\omega}{c}$$  \hspace{1cm} (3.5)

This homogeneous equation can be modified with an added force.

$$\nabla^2 U(r) + k^2 U(r) = f$$  \hspace{1cm} (3.6)

In frequency response analysis we solve Helmholtz equation for different $k$ spanning a range of frequencies.
3.2 Component Mode Synthesis

Component Mode Synthesis (CMS) is actually a family of techniques, all with the same main idea but branched out in different approaches and applications. The main idea is to partition the original computational domain into subdomains. For each subdomain an associated subspace is constructed and on each subspace a linear elastic eigenvalue problem is solved. The calculated eigenmodes are used as a basis to reduce the original problem, the number of eigenmodes used in each subspace may vary and is a subject to adaption. The method commonly used for this is a cut off frequency deciding which modes to use. The method used in this report are developed by Jakobsson, Bengzon and Larson in [2]. This is explained further in section 3.3.

3.2.1 1-dimensional case

To best explain the idea behind CMS we consider the 1-dimensional case. Think of a domain Ω=[0,1]. Partition the domain into two subdomains and an interface as in figure 4.

\[ x = 0 \quad \Gamma \quad x = 1 \]

\[ \Omega_1 \quad \Omega_2 \]

Figure 4: Partitioning of the domain.

Solving the eigenvalue problem, \(-u_{xx} = \lambda x\) on the subdomains gives us a set of eigenfunctions.

\[ z_n = \sin(2n\pi x) \quad x \in \Omega_i, n = 1...N \quad (3.7) \]

Where N is the degrees of freedom of the subdomain i.

To find the interface functions, we solve the equation,

\[ -u_{xx} = 0 \quad x \in \Omega_1 \cup \Omega_2 \quad (3.8) \]

\[ u(\Gamma) = 1 \quad (3.9) \]

\[ u(1) = u(0) = 0 \quad (3.10) \]

which gives us a global behaviour on Ω. Figure 5 shows us some of the basis functions.

11
Expanding this theory to a partition with more subdomains, we can interpret the interface functions as ”hat functions” on a coarse mesh.

\[ x = 0 \quad \Gamma_1 \quad \Gamma_2 \quad \Gamma_3 \quad x = 1 \]

Figure 6: Ω divided into four subdomains and with three interface nodes

To get further understanding of the idea behind CMS we make an example were we compare the analytical solution with the reduced on a 1 dimensional domain as in figure 6. We start by defining the problem, we will make it as easy as possible. Consider the stationary ODE.

\[ u_{xx} - \omega^2 u = f \tag{3.11} \]

Lets put \( f = -x^4(10x - 10) + 12x^2(10x - 10) + 80x^3 \).

It is easy to solve this ODE analytically and the result is \( u = 10(x - 1)x^4 \).

The same way as in FEM we now want to rewrite the problem on variational form, finding the base functions, and then project the problem onto the subspace.

Find \( u_r \in \mathcal{V} \) such that

\[ -\int_0^1 v_x u_{r,x} - \omega^2 \int_0^1 v u_r = \int_0^1 v f, \quad \forall v \in \mathcal{V} \tag{3.12} \]
The reduced space $\mathcal{V}$ contains eigenmodes defined on the subdomains and of coupling modes defined between the subspaces as in figure 6.

This gives us the following system of equations to solve.

$$K\xi - \omega^2 M\xi = F$$  (3.13)

where similar to FEM, $M$ is the mass matrix, $K$ the stiffness matrix and $F$ is the load vector with entries $F_i = \int f v \, dx$.

The reduced basis can be split into two parts one coming from the eigenmodes and one coming from the coupling modes. $u_r = u_h + u_m$. A figure of how the different modes affect on the reduced solution is displayed in figure 7 below.

Figure 7: Red represents the analytical solution and blue the reduced solution of problem 3.11. (a) A basis with only coupling modes, $u_h$. (b) A basis with only eigenmodes, $u_m$. (c) A basis with coupling modes and the first eigenmode in each subdomain, $u_h + u_m$. (d) A basis with coupling modes and the three first eigenmodes in each subdomain, $u_h + u_m$.

In figure 7(a) and 7(b) one can clearly see that to keep the error low we need to include both subdomain modes and coupling modes. Only using one of them will produce larger errors. The error between the reduced solution
and the analytical solution measured in $L^2$-norm are displayed in the table below.

| Coupling modes | Subdomain modes | $||E|| = u - u_r$ |
|----------------|-----------------|------------------|
| 0              | 1 in each subdomain | 3.77e-1          |
| all            | 0               | 1.26e-1          |
| all            | 1 in each subdomain | 1.30e-2          |
| all            | 3 in each subdomain | 2.00e-3          |
| all            | 10 in each subdomain | 1.55e-04         |

This is just an example describing the principle behind CMS, note that the reduced modes are still analytical and no numerical errors have been introduced yet. In the next chapter we will introduce the theory behind CMS.

### 3.2.2 General Setting

Now we have seen how the idea behind CMS works in a 1-dimensional setting using Helmholtz equation. To expand this idea into a more general setting we will need a little more sophisticated mathematics. But still for simplicity, let’s go back to considering only the Laplacian operator. We start by partition the domain.

Consider a domain $\Omega$ partitioned into $n$ disjoint subdomains $\Omega = \sum_{i=1}^{n} \Omega_i$. The interface connecting the subdomains is denoted $\Gamma$. For every subdomain we construct a subspace $V_i \subset V$.

$$V_i = \{ v \in H^1(\Omega) : v_{|_{\Omega \setminus \Omega_i}} = 0 \}.$$  \hspace{1cm} (3.14)

To connect the information between the subdomains we construct a subspace to $V$ as the trace space of $V$ associated with $\Gamma$, denoted $V_\Gamma$. Note that now we use a global interface in difference of the one used in section 3.2.1.

Let $E \nu \in V$ denote the harmonic extension of a function $\nu \in V_\Gamma$ to $\Omega$, given by the solution to the problem.

Find $E \nu \in V$ such that

$$\int_{\Omega} \nabla (E \nu) \cdot \nabla v = 0, \forall v \in V$$ \hspace{1cm} (3.15)

$$E \nu = \nu, \forall v \in V_\Gamma$$ \hspace{1cm} (3.16)

To construct a basis for each subdomain, $\Omega_i$ we associate an eigenvalue problem to each of them. Find $(\lambda_i, z_i) \in \mathcal{R} \times V_i$ such that

$$\int_{\Omega} \nabla z_i \cdot \nabla v = \lambda_i \int_{\Omega} z_i \cdot v \hspace{0.5cm}, \forall v \in V_i, i = 0, ..., n$$ \hspace{1cm} (3.17)
Each set of eigenfunctions, $\lambda_{i,1},...,\lambda_{i,N}$, now represent an orthonormal basis on the associated subspace $\Omega_i$. The basis for the domain, $\mathcal{V}$ can be formulated by combining all the subspaces.

$$\bigoplus_{i=1}^{n}\{z_{i,j}\}_{j=1}^{\infty}\bigoplus\{z_{\Gamma,j}\}_{j=1}^{\infty}$$  \hspace{1cm} (3.18)

Next step is to construct a discrete basis in the same manner. This gives us a finite number of eigenfunctions in each subspace. The corresponding problems to equation 3.15 and 3.17 then becomes.

Find $\mathcal{E}\nu \in \mathcal{V}^h$ such that

$$\int_{\Omega} \nabla (\mathcal{E}\nu) \cdot \nabla v = 0 \hspace{1cm} \forall v \in \mathcal{V}^h$$  \hspace{1cm} (3.19)

Find $(\lambda_i, z_i) \in \mathcal{R} \times \mathcal{V}^h_i$ such that

$$\int_{\Omega} \nabla z_i \cdot \nabla v = \lambda_i \int_{\Omega} z_i \cdot v \hspace{1cm} \forall v \in \mathcal{V}^h_i, i = 0,...,n$$  \hspace{1cm} (3.20)

The discrete basis, $\mathcal{V}^h$ for the whole domain, can be expressed as.

$$\bigoplus_{i=1}^{n}\{z_{i,j}\}_{j=1}^{N}\bigoplus\{z_{\Gamma,j}\}_{j=1}^{N}$$  \hspace{1cm} (3.21)

where $N$ are the degrees of freedom for the subspace $i$. [2]

This far we haven’t made any model reductions at all, the only thing we have done is that we have changed the basis in to a linear combination of eigenfunctions. Depending on the needed accuracy and on the application we can reduce this system and still preserve the dynamic behaviour, which is the main core of this type of reduction. An important feature of this basis in comparison with for example modal analysis is that the sparsity pattern of the reduced mass and stiffness matrices have dominant entries on the diagonal, which reduces the computational cost for large calculations.

Looking back at figure 3 we can now use parts of $\mathcal{V}^h$ as our reduced basis $R$. Project the full problem onto $R$ and solve the reduced problem. Note that if we use the whole space $\mathcal{V}^h$ as our reduced basis, the reduced problem will have the same size as the full problem, $m = n$.

### 3.2.3 Sparsity pattern of the mass and stiffness matrix

As we talked about earlier an important feature of CMS is that the density of the matrices used for calculations are low. This simplifies computations and the storage of the matrices does not demand as much space. In figure 8, one can observe the sparsity pattern of the mass and stiffness matrix for the system in the 1-dimensional example of section 3.2.1 using 10 eigenmodes
in each subdomain and three coupling modes. Since the eigenmodes are linear independent they only appear as non-zero entries on the diagonal i.e \( \int \varphi_i \varphi_j \neq 0 \) if and only if \( i = j \). The coupling modes bridges between the subdomains and are represented as the inverted L-shape in the mass matrix and as the gather of 7 dots in the lower right corner in the stiffness matrix.

![Sparsity pattern of the mass and stiffness matrix](image)

**Figure 8:** Sparsity pattern of the mass and stiffness matrix. The amount of non-zero entries is denoted \( nz \). (a) Sparsity pattern of the mass matrix. (b) Sparsity pattern of the stiffness matrix.
3.3 Adaptness

Since the accuracy of the reduced model depends on the degrees of freedom, i.e. the number of eigenfunctions in each subspace, we use an adaptive algorithm that controls the eigenfunctions used in each subspace. To be able to capture a local high frequency behavior a large number of functions must be included, while a low frequency behaviour demands less. So the needed eigenfunctions may vary greatly between subspaces. This makes the adaptive algorithm an important feature to keep the problem small enough to save time and accurate enough to capture the necessary frequencies. In the problems covered in this report high frequency behavior represent large deformations and small frequency behavior represent small deformations.

To perform the adaptive refinement we use an indicator function, $\eta_i$, for each subspace. The indicator is based on how close the reduced solution is the reference solution and the subspace with largest $\eta_i$ should be enriched. Let $R^h_i(w) \in V^h_i$ be the discrete residual in the corresponding subspace $i$. For the Helmholtz equation, we denote the subspace residual defined for $w \in V^h_i$ as

$$\int_{\Omega_i} R^h_i v = \int_{\Omega_i} f \cdot v + \int_{\Gamma_N} g_N \cdot v - \int_{\Omega_i} \nabla w \cdot \nabla v + \omega^2 \int_{\Omega_i} w \cdot v \quad \forall v \in V^h_i. \quad (3.22)$$

The indicator are defined as

$$\eta_i = \frac{1}{\Lambda_{i,m_i+1}} \|R^h_i(U^m)\|_i^2, \quad i = 1, \ldots, n \quad (3.23)$$

Where $\Lambda_{i,m_i+1}$ is the smallest eigenvalue not included in the reduced basis. A large value of the indicator indicates a large error in the subspace. Since we are both considering calculations of a spectrum of frequencies and of a single frequency, we are using different adaptive algorithms. Both are based on the indicator, $\eta$. Further reading on the indicator and a posteriori error estimates can be found in [2].
Algorithm 1 Adaption for a single $\omega$

1: start with $m_i = k_i$ eigenfunctions in the basis for $V_{i}^{h,m_i}$, $i = 1, ..., k$
2: Solve the reduced system
3: Compute the indicators for each subdomain $\eta_i$, $i = 1, ..., k$
4: if $\eta = \sum_{i=1}^{k} \eta_i \leq TOL$ then
5: Stop
6: else
7: if $\eta_i$ is large $i = 1, ..., k$ then
8: Use $m_i = m_i + l_i$ eigenfunctions in the basis for $V_{i}^{h,m_i}$
9: end if
10: Go to 2.
11: end if

Algorithm 2 Adaption for a spectrum of $\omega$

1: Start with an initial $\omega^2$
2: while $\omega^2 < \omega_{end}^2$ do
3: start with $m_i = k_i$ eigenfunctions in the basis for $V_{i}^{h,m_i}$, $i = 1, ..., n$
4: Solve the reduced system
5: Compute the indicators for each subdomain and add them to $\eta_i = \eta_i + \eta_i^{\omega^2}$, $i = 1, ..., n$
6: $\omega^2 = \omega^2 + \text{step}$
7: end while
8: if $\eta = \sum_{i=1}^{n} \eta_i \leq TOL$ then
9: Stop
10: else
11: if $\eta_i$ is large $i = 1, ..., n$ then
12: Use $m_i = m_i + l_i$ eigenfunctions in the basis for $V_{i}^{h,m_i}$
13: end if
14: Go to 2.
15: end if
Algorithm 3 Adaption for a spectrum of $\omega$

1: Start with a initial $\omega_2^2$
2: while $\omega_2^2 < \omega_{\text{end}}^2$ do
3: start with $m_i = n_i$ eigenfunctions in the basis for $V^{h,m_i}_i$, $i = 1,\ldots,k$
4: Solve the reduced system
5: Compute the indicators for each subdomain $\eta_i$, $i = 1,\ldots,n$
6: if $\eta = \sum_{i=1}^{k} \eta_i \leq TOL$ then
7: Go to 14.
8: else
9: if $\eta_i$ is large $i = 1,\ldots,n$ then
10: Use $m_i = m_i + l_i$ eigenfunctions in the basis for $V^{h,m_i}_i$
11: end if
12: Go to 4.
13: end if
14: $\omega_2^2 = \omega_2^2 + \text{step}$
15: end while

Algorithm 2 uses adaption based on the whole frequency spectrum, in the subspace with largest $\eta$ through out the whole spectrum we add modes. In algorithm 3 we use an individual adaption for each $\omega$.

3.3.1 Refinement strategy

We use two different approaches to how we refine each subspace, given the indicator. The first one is based on finding the largest indicator, $\eta_{\text{max}}$ and then refine in all subspaces that have larger error than a given share of $\eta_{\text{max}}$.

Algorithm 4 Refinement strategy 1

1: Find $\eta_{\text{max}}$
2: if $\eta_i > c \cdot \eta_{\text{max}}$, $i = 1,\ldots,n$ then
3: Add a fixed amount of modes in subspace $i$
4: end if

The other one is slightly more sophisticated but still simple. Its based on the normalized distribution of the indicator in all subspaces. You calculate how large the indicator is relative to the sum of all and then you multiply this with a fixed rate of modes. In each adaption step, you will get equal amount of added modes but spread accordingly over the subspaces.

Algorithm 5 Refinement strategy 2

1: Find the normalized distribution, $\gamma$ of the indicator $\eta$
2: Add nodes accordingly to: $F \cdot \gamma$, (where $F$ is a fixed rate of modes.)
Using refinement strategy 2 gives us a good control over how many adaptive steps we want to take. Each step is of course costly, but if it is important that we use as few modes as possible while keeping the error low, a small fixed rate is to prefer. In the other hand if the amount of adaption steps is a crucial factor and the amount of modes is not we can use a coarser refinement. So depending on the demands we put on the reduced basis, we can adjust the strategy.

Figure 9: Plot of three different fixed rates for a the truss problem described in section 14. On the x-axis we have the degrees of freedom, M. On the Y-axis we have $||E|| = (U - U^h)K(U - U^h)^T$. To reach a basis with 9000 DOFs, (approximately 50 % of the total DOFs,) we needed 84 adaption steps for a fixed rate of 100 modes, 9 steps for 1000 and 2 steps for 5000 modes. Note that the difference between the strategies are as most around 2000 DOFs. Which is approximately 10 % of the the total amount and a usual size of a reduced basis.
3.4 Beam Theory

CMS is a general method and can be applied on several areas. One interesting application is to compare beam theory with CMS using only the coupling modes as a basis. This is of interest since beam theory is a 1-dimensional approximation of a 2-dimensional problem based on the three first eigen-modes of the 2-dimensional beam.

Beam theory is a way of describing the deflections of a beam due to loads, stresses etc. This can also be applied on whole structures. The solid base of beam theory were established in the beginning of the twentieth century. Actually it all started way earlier but it took some time for the theories to be accepted as an art of engineering in stead of a mathematical theory. One of the first breakthrough is the building of the Eiffel Tower in Paris, which was at that time to be considered as an engineering master piece. There are many different branches of beam theory, Euler-Bernoulli, Rayleigh, Shear, Timoshenko beam theory and more. The two we are going to consider here is Euler-Bernoulli and Timoshenko beam theory, both based on linear elasticity but uses different assumptions regarding how to consider shear deformation and rotation of inertia. The following basic assumptions are commonly used on both theories [6].

1. One dimension is considerably larger than the other two.
2. The material is linear elastic, i.e. small deformations.
3. The Poisson effect is neglected. Poisson effect is the phenomenon that occurs when a material is compressed in one direction and due to that expands in its other directions.
4. The cross-sectional area is symmetric, so that the neutral and centroidal axes coincide.
5. Planes perpendicular to the neutral axis remain perpendicular after deformation.
6. The angle of rotation is small, so that the small angle assumption can be used.

Euler-Bernoulli equation.

\[
\frac{d^2}{dx^2} (EI \frac{d^2 w}{dx^2}) + cfw = q(x)
\] (3.24)

Where \( E \) is the modulus of elasticity, \( I \) the second moment of area about the y-axis of the beam, \( cf \) the elastic foundation modulus, \( q(x) \) is the transverse load. All given as input data to the model.
To assume Timoshenko theory we add conditions on how rotary inertia and shear deformation affects the beam. This gives us two coupled equations to determine the deflection $w$ and the rotation $\Psi$

$$\frac{d}{dx} (GAK_s (\Psi + \frac{dw}{dx}) + c_f w = q(x) \quad (3.25)$$

$$\frac{d}{dx} (EI\frac{d\Psi}{dx} + (GAK_s(\Psi + \frac{dw}{dx}) = 0 \quad (3.26)$$

where $G$ is the shear modulus and $K_s$ is a shape factor of the beam. All other variables are the same as for Bernoulli theory. Note that when the shear strain $(\Psi + \frac{dw}{dx})$ is zero and we insert the second equation into the first we attain Bernoulli theory. Timoshenko theory is commonly used on beams with length to height ratio smaller than 10 and Euler-Bernoulli is used for more slender beams.

![Figure 10: A segment of a Timoshenko Beam subject to shear deformation with length $dx$](image)

Figure 10 shows on the differences between Euler-Bernoulli and Timoshenko theory. When the shear angle is zero, the center line will coincide with the line perpendicular to the face and Euler-Bernoulli theory is pliable [7]
3.5 Local vs global interface

A problem we encounter when using classical CMS is that no matter how much we partition into fine handleable subdomains when it comes to the interface, we are still stuck with a large eigenvalue problem to solve. To avoid this problem we let the harmonic extension only be non-zero on the adjacent subdomains. Of course this demands that an interface is only adjacent to two subdomains. If not we will encounter a problem with interface-junctions and this will not be treated here. This approach will make us end up with smaller eigenvalue problems to solve, a kind of CMS on the interface. A consequence of this is that the reduced basis will be sparser. Further in the text this different methods will be referred to as local interface and global interface methods.

![Figure 11: A schematic instruction of a local and a global interface problem.](image)

A problem using local interfaces is that the error estimation theories described in [2] does not hold. This is of course a theoretical drawback but...
does not necessarily affect the numerical usability. In the following part of this section we use a qualitative numerical example to investigate what happens when we have a beam with a given length, L, and a given height, h. We divide it into different amount of partitions using local and global interfaces. The beam is clamped to the wall at $x = 0$ and has a uniformly distributed force acting on it. In figure 12 and 13 we compare the deflection of the beam using three different approaches. The first is beam theory, based on the three first eigenmodes, which gives us a 1-dimensional analytical expression for the beam. The second is a FEM approximation for a 2-dimensional beam, to be able to compare the results we plot the deflection of the beam’s centerline. The third is also the centerline of 2-dimensional beam but now with a CMS approximation using only the coupling modes as a basis. We use 3 coupling modes for each interface for the local setting. For the global setting we add 3 modes for every partition so the total amount of coupling modes are equal for both local and global interface. For example, when we have 2 subdomains, we use 3 coupling modes and when we have 3 subdomains, we use 6 coupling modes etc.

Figure 12: Comparison of deflection of a beam using three different methods. Euler Bernoulli Beam theory, FEM and CMS using global interface. (a) two subdomains, (b) three subdomains, (c) four subdomains, (d) five subdomains.
Figure 13: Comparison of deflection of a beam using three different methods. Euler Bernoulli Beam theory, FEM and CMS using local interface. (a) Two subdomains (b) Three subdomains (c) Four subdomains (d) Five Subdomains
4 Application

Since CMS is a general method it can be applied on a range of different areas. The principle will be the same but the equations will differ. In the applications in this section we will use CMS as a method to solve problems in linear elasticity.

4.1 Linear Elasticity

Linear elasticity is based upon the more sophisticated theory of non linear elasticity but with the assumption that we only have small deformations, typically in order of 0.1% or less. Linear elasticity are only valid for stresses that do not produce yielding. One more important property is that linear elasticity obeys Hooke’s law which means that the displacement of the structure is in linear proportion with the added force. In mathematical terms the equations involved in linear elasticity are

\[ \epsilon(u) = \frac{1}{2} [\nabla u + (\nabla u)^T], \]
\[ \sigma(u) = \lambda (\nabla \cdot u) + 2\mu \epsilon(u), \]

where \( \epsilon \) is the infinitesimal strain tensor, \( \sigma \) is the Cauchy stress tensor, \( \lambda \) is the Lamé modulus and \( \mu \) is the shear modulus represented by

\[ \lambda = \frac{E\nu}{(1+\nu)(1-2\nu)}, \]
\[ \mu = \frac{E}{2(1+\nu)}, \]

both \( E \) and \( \nu \) are material parameters. Note that this formulation is only valid for isotropic materials.

4.2 The equations to be considered

Now we will apply the linear elastic equations on Helmholtz equation which gives us a set of partial differential equations describing an elastostatic situation.

Let \( \Omega \subset R^2 \) be a domain with boundary \( \partial\Omega \). Let \( \partial\Omega = \Gamma_D \cup \Gamma_N \) such that \( \Gamma_D \cap \Gamma_N = \emptyset \). The outward unit normal to \( \partial\Omega \) is denoted \( n \). Then the Helmholtz equation can be described as: Find the vector displacements \( u(x) \) and the symmetric stress tensor, \( \sigma(u) \), such that
\[-\nabla \cdot \sigma(u) - \omega^2 u = f \quad x \in \Omega \quad (4.5)\]

\[\sigma(u) = 2\mu \epsilon(u) + \lambda (\nabla \cdot u) I \quad x \in \Omega \quad (4.6)\]

\[\sigma(u) = 0 \quad x \in \Gamma_D \quad (4.7)\]

\[n \cdot \sigma(u) = g_N \quad x \in \Gamma_N \quad (4.8)\]

Where \( f \) is a body force, \( \epsilon(u) = 1/2(\nabla u + \nabla u^T) \) is the strain tensor, \( I \) is the 2\( \times \)2 identity matrix and \( g_N \) is a boundary load.

Representing the system of equations above on weak form. Find \( u \in \mathcal{V} \) such that

\[
\int_{\Omega} \sigma(u) : \epsilon(v) - \omega^2 \int_{\Omega} u \cdot v - \int_{\Gamma_N} g_N \cdot v = \int_{\Omega} f \cdot v, \quad \forall v \in \mathcal{V} \quad (4.9)
\]

where \( A : B = \sum_{i,j=1}^{2} A_{ij} B_{ij} \).
4.3 Numerical examples

In this section we will consider two numerical examples to demonstrate the efficiency of the CMS method and also show on the relevance of adaptivity. The structures and problem that we use are meant to resemble a real situation but are not standardized in any sense. In the first example we use local coupling modes while in the second we use global coupling modes.

All the code are written in MATLAB and the inbuilt eigenvalue solver are used to compute the eigenvalues. MATLAB’s eigenvalue solver are based upon fortran ARPACK routines. The package are deigned to solve large eigenvalues problems using a Arnoldi–Lanzcos process called Implicitly Restarted Arnoldi Method (IRAM). Since many of the matrices we are considering in the problem are relative large, sparse, fairly diagonalized and we are only interested in a few eigenvalues the ARPACK package seems to be a good choice.

4.3.1 The Truss

![Truss](image)

Figure 14: The truss clamped at the wall and with an added rand load $f$.

The numerical example we are considering is a truss clamped to the wall at the boundary $\Gamma_D$ as in figure 14. We exert the structure with a traction force on the boundary $\Gamma_N$ and is denoted $f$ this force is Gaussian shaped and centered at the arrow in figure 14. For simplicity the material constants are chosen as: Young’s modulus, $E = 1$, Possion’s ratio $\nu = 0.3$
and density $\rho = 1$. Note that for steel those constants are approximately $E = 2 \cdot 10^{11} \text{N/m}^2$, $\nu = 0.3$ and $\rho = 8 \cdot 10^3 \text{kg/m}^3$.

We divide the truss into a unstructured mesh with approximately 18000 triangular elements. The subdomains are divided into segments and are predefined as in figure 14. In this example we use local interface as explained in section 3.5.

In figure 15 we see plots of different $\omega$ with convergence of the indicator $\eta$ and the energy norm, $|||E||| = (U - U^h)K(U - U^h)^T$. Here we use the finite element solution as the reference, $U$, so when we talk about convergence we mean the convergence towards the finite element solution not the analytical. On the x-axis we have the degrees of freedom of the system, $M$.

Figure 15: Plots of the error and indicator for three different $\omega$ and for a spectra. The dashed line represent a uniform refinement and the solid line represent the adaptive refinement. (a) $\omega^2 = 0.005$, corresponding to a frequency between the first and the second eigenvalue; (b) $\omega^2 = 0.05$, corresponding to a frequency between the ninth and the tenth eigenvalue; (c) $\omega^2 = 0.15$, corresponding to a frequency between the 19:th and the 20:th eigenvalue (d) $\omega^2 = 0.005 - 0.15$, note that now $\eta$ is scaled with a factor $1 + \frac{\omega^2}{|\lambda - \omega^2|}$ where $\lambda$ is the closest eigenvalue to $\omega^2$.
Table 1: Table of how many degrees of freedom (DOFs) needed to reduce the relative error $||E||_{rel}$ to 1 % or less.

<table>
<thead>
<tr>
<th>$\omega^2$</th>
<th>Uniform refinement</th>
<th>Adaptive refinement</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.005</td>
<td>11152</td>
<td>1930</td>
</tr>
<tr>
<td>0.05</td>
<td>8552</td>
<td>1668</td>
</tr>
<tr>
<td>0.15</td>
<td>11152</td>
<td>2343</td>
</tr>
</tbody>
</table>

Table 2: Table of the distribution of the DOFs to achieve a relative error $||E||_{rel}$ to 1 % or less. Two important subspaces has been chosen. The loaded subdomain (the domain where the distribution of the force is the most) and the coupling modes.

<table>
<thead>
<tr>
<th>Distribution of the DOFs at a relative error of 1 %</th>
<th>The loaded subdomain</th>
<th>Coupling Modes</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\omega^2$</td>
<td>Uniform refinement</td>
<td>Adaptive refinement</td>
</tr>
<tr>
<td>0.005</td>
<td>2 %</td>
<td>24 %</td>
</tr>
<tr>
<td>0.05</td>
<td>2 %</td>
<td>21 %</td>
</tr>
<tr>
<td>0.15</td>
<td>2 %</td>
<td>38 %</td>
</tr>
</tbody>
</table>
Figure 16: A comparison between CMS and modal analysis.

Figure 17: The reduced solution to the problem specified in the problem above
4.3.2 The gear wheel

The numerical example we are considering is a gear wheel clamped at the inner ring, $\Gamma_D$ as in figure 18a. We exert the gear wheel with a local Gaussian shaped traction force, $f$ on the boundary of a cog. For simplicity the material constants are chosen as: Young’s modulus, $E = 1$, Possion’s ratio $\nu = 0.3$ and density $\rho = 1$. Note that for steel those constants are approximately $E = 2 \cdot 10^{11} \text{N/m}^2$, $\nu = 0.3$ and $\rho = 8 \cdot 10^3 \text{kg/m}^3$.

We divide the gear wheel into a unstructured mesh with approximately 17,000 triangular elements. The subdomains are divided into segments and are predefined as in figure 18b. In this example, we use global coupling modes which means that the modes are defined on the whole domain.

In figure 19, we see the convergence of the indicator $\eta$ and the energy norm, $||| E |||= (U - U^h)K(U - U^h)^T$. As in the example with the lattice we use the finite element solution as the reference.
Figure 19: Plots of the error and indicator for three different $\omega$ and for a spectrum ranging from the first to the twentieth eigenvalue. The dashed line represent a uniform refinement and the solid line represent the adaptive refinement. (a) $\omega^2 = 15$, corresponding to a frequency between the first and the second eigenvalue; (b) $\omega^2 = 135$, corresponding to a frequency between the tenth and the eleventh eigenvalue; (c) $\omega^2 = 152$, corresponding to a frequency between the 21th and the 22th eigenvalue (d) $\omega^2 = 15 - 152$, note that now $\eta$ is scaled with a factor $1 + \frac{\omega^2}{|\lambda - \omega^2|}$ where $\lambda$ is the closest eigenvalue to $\omega^2$ calculated with FEM.
Table 3: Table of how many degrees of freedom (DOFs) needed to reduce the relative error $||E||_{rel}$ to 1 % or less.

<table>
<thead>
<tr>
<th>$\omega^2$</th>
<th>Uniform refinement</th>
<th>Adaptive refinement</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>2528</td>
<td>1193</td>
</tr>
<tr>
<td>135</td>
<td>3328</td>
<td>793</td>
</tr>
<tr>
<td>152</td>
<td>2728</td>
<td>798</td>
</tr>
</tbody>
</table>

Table 4: Table of the distribution of the DOFs to achieve a relative error $||E||_{rel}$ to 1 % or less. Two important subspaces has been chosen. The loaded subdomain (the domain where the distribution of the force is the most) and the Coupling modes.

<table>
<thead>
<tr>
<th>Distribution of the DOFs at a relative error of 1 %</th>
</tr>
</thead>
<tbody>
<tr>
<td>The loaded subdomain</td>
</tr>
<tr>
<td>$\omega^2$</td>
</tr>
<tr>
<td>------------</td>
</tr>
<tr>
<td>15</td>
</tr>
<tr>
<td>135</td>
</tr>
<tr>
<td>152</td>
</tr>
<tr>
<td>Coupling Modes</td>
</tr>
<tr>
<td>$\omega^2$</td>
</tr>
<tr>
<td>------------</td>
</tr>
<tr>
<td>15</td>
</tr>
<tr>
<td>135</td>
</tr>
<tr>
<td>152</td>
</tr>
</tbody>
</table>
Figure 20: A comparison between CMS and modal analysis.

Figure 21: The reduced solution to the problem specified in the problem above.
5 Conclusions and discussion

In this report I have focused on covering the implementation of CMS to perform Frequency Response Analysis on industrial geometries using an adaptive refinement. There are as mentioned before many variations how to use CMS and the one covered here is for linear elastic problems but can of course be further developed into other areas. The studies that have been made are focused on evaluating the adaptive refinement compared to a uniform one. The comparison is focused on the size of the reduced model compared to the deviation against the FE solution. Note also that the examples uses different approaches for computing the coupling modes. In the truss we use local interface while in the gear wheel we use global interface.

5.1 Local vs global interface

Comparing the different methods shows on different advantages. Section 3.5 shows on the problem using local coupling modes. Even though we do not need to solve a large eigenvalue problem, using local coupling produce less accurate approximations than for global, given a fixed amount of modes. This means that we need to use a larger basis using local interface to get as good approximation as for global. To draw conclusions from this is not that straight forward since we have a lot of computational advantage finding the local interfaces at the same time as we have drawbacks in using them as a basis. Another advantage in using local interface is that the reduction matrix $V^h$ is sparser than for global interface. This saves us both time and memory. It should be interesting to see further investigations in this subject and test on larger problems.

5.2 Numerical examples

This section covers some thoughts on the numerical section. Analyzing figure 15 and 19, we see that we have faster convergence for the adaptive refinement than for the uniform refinement for all cases studied. We can also notice a strong correlation between $\eta$ and $|||E|||$. Note that $\eta$ only represents an indicator and if we want to use it as an error control estimator we need to add a scaling factor, different for each $\omega$. The convergence of the uniform refinement seems to be more stable between different geometries and frequencies, while the convergence of the adaptive loop, although its always faster than the uniform, seems to depend more of which $\omega$ we choose. In table 2 and 4 one can see how the distribution of modes differs between the adaptive and the uniform refinement. I have selected to display two interesting type of modes. The modes on the loaded subdomain and the coupling modes. Here it is clear that the adaptive refinement chooses to add a lot of modes in the loaded subdomain where the stresses and strains
are high. Also the coupling modes are important to have a good global behaviour.

This is only one way of studying the functionality of the adaption and several others such as time-efficiency and memory allocation are of interest. If we compare the results of CMS with modal analysis one can clearly see that modal analysis has faster convergence towards the FE-solution in the beginning. This is quite natural since modal analysis only uses global modes which often gives a good approximation with few modes. However if more DOFs are added in the reduced method the better CMS works as a basis in comparison to modal analysis. This can be explained by thinking of a high order, 1000th or so, eigenmode. In the modal analysis case we lose a lot of accuracy when we introduce higher order modes, this is due to primary two reasons. First we introduce errors due to that the wavelength of the vibration are reaching the element size of the mesh, this problem is also encountered in the FE-solution. The second source of error is the round-of errors in the iterative eigenvalue solver (IRAM) propagates for each eigenmode. In CMS we encounter the problem with vibrations reaching the mesh size but we reduce the iterative errors due to that we use smaller subdomains. Another advantage in CMS is that the computational cost for finding local modes is cheaper than the global ones used in modal analysis. CMS also uses sparse mass and stiffness matrices which reduces the computational cost and the memory allocation. We must also remember that CMS is meant for finding eigenvalues and solving frequency response problems where we have trouble solving the global problem at all or when it is to time demanding.

Another thing that we need to have in mind is that all the code is written in MATLAB and only in 2 dimensions. This limits us to fully investigate CMS as a method. Problems in the order of $10^6$ DOFs is simply impossible to manage. The numerical examples are restricted to models in the region of $10^4$ DOFs, especially when we want to make comparison with modal analysis and full finite element models. This makes it quiet unnecessary to look into questions of memory allocation and time-efficiency. To really access this problem, need to implement the problem in a more basic programming language such as C++ and maybe also consider parallel computation.

5.3 Future and on going work

There are a lot of loose ends in this project and a lot of things that need to be further investigated. Here is a list of some of the areas where it should be interesting to see further development.

1. Investigate the adaptivity for more geometries and load cases.

2. Develop the methods to be able to handle acoustic problems

3. Further develop the refinement strategy so it better can handle large
frequency domains. The one used now only add modes when stepping in frequency space but it would be interesting to see a refinement method that both can add and remove modes.

4. Make further investigations on the efficiency of the implementation. Interesting questions are the ideal size of the subdomains? How much time can we save on putting an effort to chose an optimal reduced basis?

5. Implement the code in C++ and test the adaptivity on larger problems. Also a 3 dimensional implementation is in the to do list.

6. Make an implementation of an adaptive refinement for Automated Multi Level Substructuring system, AMLS is an extension of CMS but in several levels see REF for further reading, this work is on going. For further reading about AMLS we refer the reader to [8]

7. Investigate further on the subject of local vs global interfaces. Implement in C++ and test on larger geometries.
References


