Analysis of a system of elliptic partial differential equations and its possible boundary conditions when discretized with Hermite and Lagrange elements

av

Adrian Roaldssønn Hope

MASTERTHESIS
for the degree of
Master i Anvendt matematikk og mekanikk

(Master of Science)

Det matematisk- naturvitenskapelige fakultet
Universitetet i Oslo

January 2013

Faculty of Mathematics and Natural Sciences
University of Oslo

Acknowledgments

This thesis has been written as a part of a Master of applied mathematics degree at the University of Oslo, Department of Mathematics.

My supervisors have been Mikael Mortensen and Kent-Andre Mardal. I am thankful to Kent-Andre Mardal for all the help and advice he has given throughout the thesis. I would also like to thank Simula Research Laboratory for providing computing resources and a good work environment at their facilities.

I would like to thank Steffen Sjursen, Tuva Hope, Øyvind Aardal and Unni Hope for proofreading.

My friends and family have been invaluable to me during my time as a student at the University. Their support and friendship have been very important to me through the years.

Contents

Chapter 1

Introduction

This thesis examines a system of elliptic partial differential equations. The system itself is not elliptic. It will be called "the simplified k-ϵ model" and is defined as:

$$\begin{cases} \epsilon - \Delta k = & f_1(x) \quad on\,\Omega \\ -\Delta\epsilon + k = & f_2(x) \quad on\,\Omega. \end{cases} \tag{1.1}$$

There are two main goals in this thesis. The first goal is to determine if a previously unused set of boundary conditions is numerically stable. The second goal is to make a stable implementation (1.1) using a mixed finite element method. If these goals are reached, the lessons learned from examining the simplified model can be valuable when implementing the full k-ϵ-model.

The new set of boundary conditions (1.3) will be compared to a set of boundary conditions (1.2) from the literature [1]

$$BC_1 = \begin{cases} \epsilon(x) = & g_1(x) \quad on\,\partial\Omega \\ k(x) = & g_2(x) \quad on\,\partial\Omega \end{cases} \tag{1.2}$$

$$BC_2 = \begin{cases} \epsilon(x) = & g_1(x) \quad on\,\partial\Omega \\ \frac{\partial\epsilon}{\partial n} = & g_2(x) \quad on\,\partial\Omega. \end{cases} \tag{1.3}$$

Previous work on the k-ϵ-model by implementing the finite element method discovered issues with convergence of the numerical scheme [2, 3, Smith, R.M]. Smith used a continuous Lagrange approach to the problem, which only allows for one boundary condition for each variable.

The problem (1.1) is a simplification of the more complex turbulence problem called the k-ϵ- model (1.4):

$$\begin{aligned} \partial_t k + \bar{u}\nabla k - \tfrac{1}{2}c_\mu \tfrac{k^2}{\epsilon}|\nabla\bar{u} + \nabla\bar{u}^T|^2 - \nabla\cdot\left(c_\mu\tfrac{k^2}{\epsilon}\nabla k\right) + \epsilon = &\ 0 \\ \partial_t\epsilon + \bar{u}\nabla\epsilon - \tfrac{1}{2}c_1\tfrac{k^2}{\epsilon}|\nabla\bar{u} + \nabla\bar{u}^T|^2 - \nabla\cdot\left(c_\epsilon\tfrac{k^2}{\epsilon}\nabla\epsilon\right) + c_2\tfrac{\epsilon^2}{k} = &\ 0. \end{aligned} \tag{1.4}$$

The k-ϵ- model is one of the most commonly used turbulence models. It consists of two transport equations and models the transport of the variables k,ϵ. The variable k represents the turbulent kinetic energy, while ϵ represents the turbulent dissipation. A new set of possible boundary conditions should be useful for using the model in engineering problems.

In the study of the simplified k-ϵ- model, the finite element method (FEM) is utilized. Both regular and mixed finite element methods are used. The work on mixed finite elements largely depends on two publications [4, 5].

The element types Lagrange-2, Lagrange-3 and Hermite elements are used. The Lagrange-elements are chosen because they are $H^1(\Omega)$ conforming and are among the simplest and most used elements. The Hermite elements are $H^2(\Omega)$ conforming and among the simplest elements to conform with $H^2(\Omega)$. Hermite elements are crucial because they allow for two boundary conditions to be set. They are therefore needed to examine what boundary conditions can be set. These two element types will be used when examining the stability of the simplified k-ϵ- model.

Hermite elements are not commonly used. One of the reasons is rounding error issues pertaining to mesh refinement. These issues are solved in this thesis in chapter 6 by scaling the basis functions. Since they are not commonly used, there is not a lot of literature on the matter, thus most of chapter 7 through 9 is original research.

In chapter two, the thesis enumerates a list of boundary conditions known to work. The use of a different type of boundary conditions is potentially useful when applying the k-ϵ model to real life problems where the experimental information about the boundary conditions is limited.

Time is excluded from the simplified models. To implement time, it is necessary to first calculate the current state of the system, and then use the time information to calculate the next time step. If the calculation of the current state is inaccurate, it is not possible to calculate the correct states for later times. If the approach proves useful, then adding time to the models can be done.

The analysis of condition numbers and the preconditioning of the systems is based on the papers [6, 7]. The preconditioner is constructed so that when it is combined with the differential operator becomes an automorphi. Analyzing the preconditioned form will say something about how well defined the differential form is and how stable the solutions are.

For the Finite Element Method the books [8, 9] are used.

Chapter 2

Arriving at the problem

For a sense of completeness we explain the assumptions, reasoning and ideas that leads to the k-ϵ- model. We follow [1, p. 40-41].

The problem (1.1) is a simplification of the more complex turbulence problem called the k-ϵ- model (2.5). This is derived from the general Navier-Stokes equations for incompressible flows (2.1)

$$\begin{aligned} \nabla \cdot u = & \ 0 \\ \partial_t u + u \nabla u + \nabla p - \nu \Delta u = & \ \tfrac{f}{\rho}. \end{aligned} \tag{2.1}$$

with ν as the kinematic viscosity, $p = P/\rho$ is the reduced pressure, ρ is a density field, u is a velocity vector field, P is a pressure field and f represents external forces. Consider (2.1) with random initial data $u^0 = \bar{u}^0 + u'^0$, where $\bar{u}$ represents the expected value and u'^0 is the random element. Taking the expected value of the entire Navier-Stokes then leads to

$$\begin{aligned} \nabla \cdot \bar{u} = & \ 0 \\ \partial_t \bar{u} + \nabla \cdot \overline{(\bar{u} + u') \otimes (\bar{u} + u')} + \nabla \bar{p} - \nu \Delta \bar{u} = & \ \bar{f} \end{aligned}$$

which is the same as

$$\begin{aligned} \nabla \cdot \bar{u} = 0, \qquad & R = -\overline{u' \otimes u'} \\ \partial_t \bar{u} + \nabla \cdot (\bar{u} \otimes \bar{u}) + \quad & \nabla \bar{p} - \nu \Delta \bar{u} = \bar{f} + \nabla \cdot R. \end{aligned} \tag{2.2}$$

With the following assumptions, it is possible to derive the k-ϵ equations:

- Frame invariance and 2D mean flow, ν_T a polynomial function of k, ϵ.
- u'^2 and $|\nabla \times u'|^2$ are passive scalars when convected by $\bar{u} + u'$.
- Ergodicity allows statistical averages to be replaced by space averages.
- Local isotropy of the turbulence at level of small scales.

- A Reynold hypothesis for $\overline{\nabla \times u' \otimes \nabla \times u'}$.
- A closure hypothesis $|\nabla \times \nabla \times u'|^2 = c_2 \frac{\epsilon^2}{k}$.

The set of equations used to find R, the turbulence in the flow, is

$$k = \frac{1}{2}\overline{|u'|^2} \qquad \epsilon = \frac{\nu}{2}\overline{|\nabla u' + \nabla u'^T|^2} \tag{2.3}$$

with k as the turbulent kinetic energy or small scales, and ϵ is the rate of viscous energy dissipation. Reynolds hypothesis is that the turbulence (R) in flows and is a local function of $\nabla u' + \nabla u'^T$. In two dimensional mean flows we have,

$$R = \nu_T(\nabla \bar{u} + \nabla \bar{u}^T) + \alpha I \qquad \nu_T = c_\mu \frac{k^2}{\epsilon}. \tag{2.4}$$

Combining the assumptions , (2.3), (2.4) with (2.2) we arrive at (2.5)

$$\begin{aligned} \partial_t k + \bar{u}\nabla k - \tfrac{1}{2} c_\mu \tfrac{k^2}{\epsilon}|\nabla \bar{u} + \nabla \bar{u}^T|^2 - \nabla \cdot \left(c_\mu \tfrac{k^2}{\epsilon}\nabla k\right) + \epsilon = &\ 0 \\ \partial_t \epsilon + \bar{u}\nabla \epsilon - \tfrac{1}{2} c_1 \tfrac{k^2}{\epsilon}|\nabla \bar{u} + \nabla \bar{u}^T|^2 - \nabla \cdot \left(c_\epsilon \tfrac{k^2}{\epsilon}\nabla \epsilon\right) + c_2 \tfrac{\epsilon^2}{k} = &\ 0. \end{aligned} \tag{2.5}$$

With the constants $c_\mu, c_\epsilon, c_1, c_2$ chosen so that the model is accurate for:

- The decay in time of homogeneous turbulence.
- The measurements in shear layers in local equilibrium.
- The log-wall law in boundary layers.

The suggested set of boundary conditions according to [1] is:

- $\bar{u}, k, \epsilon$ given initially everywhere.
- $\bar{u}, k, \epsilon$ given on the inflow boundaries at all t
- $\nu_t \partial_n \bar{u}, \nu_t \partial_n k, \nu_t \partial_n \epsilon$ given on the outflow boundaries for all t.
- $u \cdot n = 0, \frac{\bar{u}\cdot s}{\sqrt{\nu|\partial_n \bar{u}|}} - \frac{1}{\chi} log \left(\delta \sqrt{\frac{1}{\nu}|\partial_n \bar{u}|}\right) + \beta = 0$ on $\Gamma + \delta$.
- $k|_{\Gamma+\delta} = |\mu \partial_n(\bar{u} \cdot s)| c_\mu^{-\frac{1}{2}}, \epsilon|_{\Gamma+\delta} = \frac{1}{\chi\delta}|\mu \partial_n(\bar{u} \cdot s)|^{\frac{3}{2}}$.

Where Γ represents a solid wall, and δ is an adjustable artificial boundary parallel to the wall with $\delta(t, x) \in [10, 100]\nu/u_\tau$. s is the tangent to the boundary, and n is the normal to the boundary. In order to study other possible sets of boundary conditions for 2.5, it is practical to do some simplifications. The main simplification is to assume that $\frac{k^2}{\epsilon}$ is linear, and to not consider changes in time. Thus (2.5) reduces to (2.6)

$$\begin{aligned} \bar{u}\nabla k - \tfrac{1}{2}c_\mu\tfrac{k^2}{\epsilon}|\nabla\bar{u} + \nabla\bar{u}^T|^2 - c_\mu\tfrac{k^2}{\epsilon}\Delta k + \epsilon = & \; 0 \\ \bar{u}\nabla\epsilon - \tfrac{1}{2}c_1\tfrac{k^2}{\epsilon}|\nabla\bar{u} + \nabla\bar{u}^T|^2 - c_\epsilon\tfrac{k^2}{\epsilon}\Delta\epsilon + c_2\tfrac{\epsilon^2}{k} = & \; 0 \end{aligned} \tag{2.6}$$

To study a problem similar to (2.6), yields valuable knowledge about possible boundary conditions for (2.5). Thus we study an as simple set of equations possible, the candidate set of equations (2.7) is chosen

$$\begin{cases} \epsilon(x) - \Delta k(x) = & f_1(x) \quad on\, \Omega \\ -\Delta\epsilon(x) + k(x) = & f_2(x) \quad on\, \Omega. \end{cases} \tag{2.7}$$

2.1 A short description of FEM

The finite element method is a method of numerically approximating a function by describing it in terms of local polynomials. One usually starts out with a problem, lets say Poisson problem (2.8).

$$\begin{cases} -\Delta u = & f \quad on\, \Omega \\ u = & 0 \quad on\, \partial\Omega \end{cases} \tag{2.8}$$

where Ω is an open domain, and $\partial\Omega$ is its boundary. Then a weak form is created by multiplying by a test function and integrating by parts creating (2.9)

$$\int_\Omega \nabla u \cdot \nabla v \,\mathrm{d}x = \int_\Omega f v \,\mathrm{d}x. \tag{2.9}$$

We say; Let $u \in H_0^1(\Omega)$, u is a weak solution of (2.8) if (2.9) holds for all $v \in H_0^1(\Omega)$. $H_0^1(\Omega) = C_c^0(\overline{\Omega}) \cap H^1(\Omega)$, where $H^1(\Omega)$ is a Sobolev space and $C_c^0(\overline{\Omega})$ [1] is the space of all continuous functions with compact support.

For easier notation, we let $W = \{w \in H_0^1(\Omega)\}$. To discretize u, v we restrict our numerical solution and test functions to a subspace $v_h, u_h \in W_h \subset W$. The space W_h is spanned by a set of basis functions $\{\phi_i\}_{i=1}^N$. The selection of these basis functions is determined by which element is chosen. The domain is subdivided into a set of elements. These elements is usually triangular or square in 2D, and line segments in 1D. The element type defines local basis functions for each sub domain. Each elements have a set of evaluation points. The basis functions form a basis with respect to the evaluation points. They are usually polynomials. If two basis functions $\phi_{i,L_1}, \phi_{j,L_2}$ has the same evaluation point, then the sum of them is the global basis function.

[1] Notation $\bar{\Omega} = \Omega \cup \partial\Omega$ is used.

We make an assumption that our numerical solution can be described in terms of the basis functions $\tilde{u} = \sum U_k \phi(x)_k$. We then have;

$$\int_\Omega \sum U_k \phi(x)'_k \cdot \sum V_i \phi(x)'_i \,\mathrm{d}x = \int_\Omega f \sum V_i \phi(x)_i \,\mathrm{d}x \tag{2.10}$$

$$\sum \sum U_k V_i \int_\Omega \phi(x)'_k \phi(x)'_i \,\mathrm{d}x = \sum V_i \int_\Omega f \phi(x)_i \,\mathrm{d}x \tag{2.11}$$

Since this has to hold for any test function v, we can choose to use a set test functions;

$$v_i = \sum_{j=1}^{n} V_{i,j} \phi_j \tag{2.12}$$

$$\begin{cases} V_{i,i} = 1 \\ V_{i,j} = 0 \quad \text{for } i \neq j \end{cases} \tag{2.13}$$

Meaning that $(V_{i,i})$ is equal to 1, while all of the other coefficients are 0. (2.11) then becomes a set of n equations with $\{U_i\}$ as the only unknown values. By setting $A_{i,j} = \int_\Omega \phi(x)'_k \phi(x)'_i \,\mathrm{d}x$, and $b_j = \sum \int_\Omega f \phi(x)_i \,\mathrm{d}x$, the system (2.11) is the same as the matrix equations (2.14).

$$A\tilde{u} = b \tag{2.14}$$

where $\tilde{u} = [U_1, U_2, \ldots, U_n]^T$.

By solving (2.14) for $\tilde{u}$ we obtain our finite element approximation

For this thesis this approach to the finite element method will be used, but in a system of equations.

Chapter 3

Condition Numbers

This chapter describes condition numbers and their importance in PDEs. As we know a linear PDE system $Lu = f$ can be approximated by using FEM. Such that approximating the solution can be done by instead solving the linear system $Ax = b$. In general one would like to solve this by taking the inverse of the matrix A. A natural question to ask oneself is: "Is the taking the inverse numerically stable?" To answer this question we introduce condition numbers. We let e represent the error in the vector b coming from the representation of b in the computer. So we will have $Ay = b + e$. Then $\|A^{-1}(e)\|/\|A^{-1}b\|$ will represent the relative error in the solution, and$\|e\|/\|b\|$ will be the relative error in the data. We then figure out the error in the solution relative the error in the data by dividing one by the other.

$$\begin{aligned}\frac{\|A^{-1}e\|/\|A^{-1}b\|}{\|e\|/\|b\|} &= \frac{\|A^{-1}e\|\|b\|}{\|e\|\|A^{-1}b\|} \\ = \frac{\|A^{-1}e\|}{\|e\|}\frac{\|b\|}{\|A^{-1}b\|} &\leq \|A^{-1}\| \cdot \|A\|\end{aligned}$$

on the other hand we have

$$\begin{aligned}\frac{\|A^{-1}e\|/\|A^{-1}b\|}{\|e\|/\|b\|} &= \frac{\|A^{-1}e\|\|b\|}{\|e\|\|A^{-1}b\|} \\ = \frac{\|A^{-1}e\|}{\|e\|}\frac{\|b\|}{\|A^{-1}b\|} &\geq \frac{1}{\|A\| \cdot \|A^{-1}\|}\end{aligned}$$

by defining $\kappa(A) := \|A\| \cdot \|A^{-1}\|$ we get

$$\begin{aligned}
\frac{1}{\|A\| \cdot \|A^{-1}\|} &\leq \frac{\|A^{-1}e\|/\|A^{-1}b\|}{\|e\|/\|b\|} \leq \|A\| \cdot \|A^{-1}\| \\
\frac{\|e\|}{\|b\|}\frac{1}{\kappa(A)} &\leq \frac{\|A^{-1}(e+b) - A^{-1}b\|}{\|A^{-1}b\|} \leq \frac{\|e\|}{\|b\|}\kappa(A) \\
\frac{\|e\|}{\|b\|}\frac{1}{\kappa(A)} &\leq \frac{\|y-x\|}{\|x\|} \leq \frac{\|e\|}{\|b\|}\kappa(A).
\end{aligned}$$

Therefore we see that the condition number impose a bound on the error of the solution of the linear problem.

We can furthermore compute the condition number. The norm of a matrix is the following: $\|A\| := sup\{\|Ax\| : \|x\| \leq 1\}$ which in an l_2 -matrix norm can be re written as $\|A\| = \sup_{x \in C^n} \frac{|x^T Ax|}{|x|^2}$ which is the magnitude of the largest eigenvalue of A called μ^A_{max}. Similarly $\|A^{-1}\| = \sup_{x \in C^n} \frac{|x^T A^{-1}x|}{|x|^2} = \frac{1}{\inf_{x \in C^n} \frac{|x^T Ax|}{|x|^2}}$. And $\inf_{x \in C^n} \frac{|x^T Ax|}{|x|^2}$ is the smallest possible eigen value of A called μ^A_{min}. By combining all this we get that $\kappa(A) := \|A\| \cdot \|A^{-1}\| = \frac{|\mu^A_{max}|}{|\mu^A_{min}|} \geq 1$. The smallest possible condition number A can have is 1.

3.1 Condition number relating to the k-ϵ model

It is useful to examine simpler versions of the k-ϵ-model and solve any issues there before extrapolating the the full model. This thesis will look at the properties of three different weak formulations of the simplified problem in one and two dimensions. One of these formulations is the linear formulation, the two others are systems of dependent equations. The main differences between the two systems are the selection of function spaces for the trail and test functions in FEM. This will be described in detail in chapter 4. The matrices produced by FEM is then preconditioned, and the condition number of the preconditioned matrix is analyzed. The idea is that if the condition number of the preconditioned matrix remain within $O(1)$, then the unconditioned matrix makes an isomorphism between the function space of the solutions $((k, \epsilon))$ and the function space of the input data $(f_1, f_2))$ [6, 7].

3.1.1 An example

Let $(k, \epsilon) \in H^1(\Omega) \times H^1(\Omega)$ and the finite element method creates the pairing $< A(k, \epsilon), (v_1.v_2) >=< (f_1, f_2), (v_1.v_2) > \in H^{-1}(\Omega) \times H^{-1}(\Omega)$. We use the

Riesz mapping[1] to construct a preconditioner B. The preconditioner should be constructed so that $< BA(k,\epsilon),(v_1.v_2) >=< B(f_1,f_2),(v_1.v_2) > \in H^1(\Omega) \times H^1(\Omega)$. One may then analyze the condition number $\kappa(BA)$ in order to figure out if the mapping $(k,\epsilon) \rightarrow < A(k,\epsilon),(v_1.v_2) >$ is an isomorphism.

If $c_0 \leq \kappa(BA) \leq c_1$ when the mesh is refined, then $\frac{c_0}{\|(BA)^{-1}\|} \leq \|BA\| \leq \frac{c_1}{\|(BA)^{-1}\|}$. Thus the inverse operator of BA is well defined. Therefor the mapping $BA : H^1(\Omega) \times H^1(\Omega) \rightarrow H^1(\Omega) \times H^1(\Omega)$ is an isomorphism. Since B is known to be an isomorphism, it also follows that A itself is isomorphic.

[1] see appendix

Chapter 4

Defining a weak formulation of the simplified k -ϵ- model

For this section we assume Ω is open and bounded, with a C^1 boundary. As described earlier, a simplification of the k-ϵ problem can be defined as following.

Find ϵ and k solving the equation

$$\begin{aligned} \epsilon(x) - \Delta k(x) =& f_1(x) \quad on\,\Omega \\ -\Delta\epsilon(x) + k(x) =& f_2(x) \quad on\,\Omega. \end{aligned}$$

Two possible sets of boundary conditions are

$$BC_1 = \begin{cases} \epsilon(x) = & g_1(x) \quad on\,\partial\Omega \\ k(x) = & g_2(x) \quad on\,\partial\Omega \end{cases} \tag{4.1}$$

$$BC_2 = \begin{cases} \epsilon(x) = & g_1(x) \quad on\,\partial\Omega \\ \frac{\partial\epsilon}{\partial n} = & g_2(x) \quad on\,\partial\Omega. \end{cases} \tag{4.2}$$

Writing the problem in matrix form looks like:

$$\begin{bmatrix} I & -\Delta \\ -\Delta & I \end{bmatrix} \cdot \begin{bmatrix} \epsilon \\ k \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix}. \tag{4.3}$$

We can then multiply it with a test function, integrate, and integrate by parts to obtain a weak formulation

$$\begin{bmatrix} v_1 & v_2 \end{bmatrix} \cdot \begin{bmatrix} I & -\Delta \\ -\Delta & I \end{bmatrix} \cdot \begin{bmatrix} \epsilon \\ k \end{bmatrix} = \begin{bmatrix} v_1 & v_2 \end{bmatrix} \cdot \begin{bmatrix} f_1 \\ f_2 \end{bmatrix}$$

$$\int_\Omega v_1\epsilon - v_1\Delta k - v_2\Delta\epsilon + v_2 k\, d\,\mathrm{x} = \int_\Omega v_1 f_1 + v_2 f_2\, d\,\mathrm{x}.$$

Integration by parts can yield two different weak formulations.

$$\int_\Omega v_1\epsilon + \nabla k \cdot \nabla v_1 + \nabla v_2 \cdot \nabla \epsilon + v_2 k \, d\,\mathrm{x} = \int_\Omega v_1 f_1 + v_2 f_2 \, d\,\mathrm{x} \tag{4.4}$$

$$\int_\Omega v_1\epsilon - k\Delta v_1 - v_2\Delta\epsilon + v_2 k \, d\,\mathrm{x} = \int_\Omega v_1 f_1 + v_2 f_2 \, d\,\mathrm{x}. \tag{4.5}$$

4.1 The Trace Theorem

The trace theorem [10, p. 258] states that:

Trace Theorem 4.1 *Assume Ω is bounded and $\partial\Omega$ is C^1.*
Then there exists a bounded linear operator

$$T : W^{1,p}(\Omega) \to L^p(\partial\Omega)$$

such that;
$Tu = u|_{\partial\Omega}$ if $u \in W^{1,p}(\Omega) \cap C(\bar{\Omega})$
and
$\|Tu\|_{L^p(\partial\Omega)} \leq C\|u\|_{W^{1,p}(\Omega)}$ for each $u \in W^{1,p}(\Omega)$, with the constant C depending only on p and Ω.

4.2 $H^2 \times L^2$ formulation

A weak formulation reads as follows:

Weak formulation 4.2.1 *Find $\epsilon \in H^2(\Omega)$ and $k \in L^2(\Omega)$*
solving (4.5) $\forall(v_1, v_2) \in H^2(\Omega) \times L^2(\Omega)$.

For this formulation, we use the boundary condition (4.2). The advantage of using (4.2) is that it bounded by the trace theorem. For all functions $u \in H^2(\Omega)$, there exists a trace Tu with $\|\frac{\partial u}{\partial n}\|_{L^2(\partial\Omega)} \leq C_1\|u\|_{H^2(\Omega)}$, and $\|u\|_{L^2(\partial\Omega)} \leq C_2\|u\|_{H^2(\Omega)}$. Thus it makes sense to talk about the trace. It is not however obvious that it makes sense to talk about the trace of $L^2(\Omega)$, and it is not obvious that it is bounded for every bounded function $w \in L^2(\Omega)$.

4.3 $H^1 \times H^1$ formulation

A weak formulation reads as follows:

Weak formulation 4.3.1 *Find $\epsilon \in H^1(\Omega)$ and $k \in H^1(\Omega)$*
solving (4.4) $\forall(v_1, v_2) \in H^1(\Omega) \times H^1(\Omega)$.

For this formulation, we use the boundary condition (4.1). The advantage of using (4.1) is that it bounded by the trace theorem. For all functions $u \in H^1(\Omega)$, there exists a trace Tu where we have $\|u\|_{L^2(\partial\Omega)} \leq C_2\|u\|_{H^1(\Omega)}$. Thus we know it makes sense to talk about the trace. We can then construct a trace function $T(\epsilon, k)_{H^1(\Omega)\times H^1(\Omega)} \to (\epsilon, k)_{L^2(\partial\Omega)\times L^2(\partial\Omega)}$ which is bounded.

4.4 Argument about choice in boundary conditions

The choices of the boundary conditions 4.2 and 4.3 are made to work on as general problems as possible. Allowing $(\epsilon, k) \in H^1(\overline{\Omega}) \times H^1(\overline{\Omega})$, or $(\epsilon, k) \in H^2(\overline{\Omega}) \times L^2(\overline{\Omega})$ permits the usage of 12 different boundary conditions, some of them listed here. The full table is listed in the appendix as 11.3. Most of these possible boundary conditions have the disadvantage that they have higher requirements for the function space of the solution. These requirements are usually of the type: Find $(\epsilon, k) \in H^2(\overline{\Omega}) \times L^2(\Omega)$. They require the closure to be included in the solution space. The Trace Theorem is not enough to prove the existence of the boundary condition for all functions in the solution space. The Trace theorem is used to determine the exact requirements.

The boundary conditions (4.2) and (4.1) exists for all functions in the function spaces $H^2(\Omega) \times L^2(\Omega)$ and $H^1(\Omega) \times H^1(\Omega)$ respectively.

List of boundary conditions requiring $(\epsilon, k) \in H^2(\overline{\Omega}) \times L^2(\Omega)$

$$\begin{cases} \frac{\partial\epsilon(x)}{\partial n} = & g_1(x) \quad on\, \partial\Omega \\ \frac{\partial^2\epsilon(x)}{(\partial n)^2} = & g_2(x) \quad on\, \partial\Omega \end{cases} \tag{4.6}$$

$$\begin{cases} \epsilon(x) = & g_1(x) \quad on\, \partial\Omega \\ \frac{\partial^2\epsilon(x)}{(\partial n)^2} = & g_2(x) \quad on\, \partial\Omega \end{cases} \tag{4.7}$$

If we have $\epsilon \in H^2(\Omega)$ the Trace theorem says we know that the trace Tu exists and is bounded by $\|\epsilon\|_{H^2(\Omega)}$. $\epsilon \in H^2(\Omega)$ means $\|\epsilon\|_{H^2(\Omega)} < \infty$. So $\|Tu\|_{H^1(\partial\Omega)}$ is well defined and bounded. The boundary condition

$$\frac{\partial^2\epsilon(x)}{(\partial n)^2} = g_2(x) \quad on\, \partial\Omega \tag{4.8}$$

is not bounded by $\|\epsilon\|_{H^2(\Omega)}$. So the problem has to read as follows to be well defined:

Weak formulation 4.4.1 *Find $\epsilon \in H^2(\overline{\Omega})$ and $k \in L^2(\Omega)$ solving* (4.5) $\forall (v_1, v_2) \in H^2(\overline{\Omega}) \times L^2(\Omega)$.

Chapter 5

Uniqueness of Solutions of Bilinear forms

This section contains proofs of uniqueness of solutions of the bilinear forms. In 1969 Ivo Babuška submitted to Springer-Verlag the theorem later called "Babuška-Lax-Milgram Theorem" [11, 5]. The "Babuška-Lax-Milgram Theorem" is a generalization of the "Lax-Milgram Theorem" [10]. It will be used to prove uniqueness of the $H^1(\Omega) \times H^1(\Omega)$ formulation in this thesis.

Babuška-Lax-Milgram Theorem 5.1 *Let H_1 and H_2 be two Hilbert (complex and complete) spaces with scalar product $(\cdot,\cdot)_{H_1}$ and $(\cdot,\cdot)_{H_2}$. Let $B[u,v]$ be a bilinear form on $H_1 \times H_2$ for $u \in H_1, v \in H_2$ such that;*

$$|B[u,v]| \leq C_1 \|u\|_{H_1} \|v\|_{H_2} \tag{5.1}$$

$$\sup_{\substack{u \in H_1 \\ \|u\| \leq 1}} |B[u,v]| \geq C_2 \|v\|_{H_2} \tag{5.2}$$

$$\sup_{\substack{v \in H_2 \\ \|v\| \leq 1}} |B[u,v]| \geq C_3 \|u\|_{H_1} \tag{5.3}$$

with $C_2 > 0, C_3 > 0, C_1 < \infty$.

Let f be a linear functional on H_2 ($f \in H_2^*$). Then there exists exactly one element $u_f \in H_1$ such that

$$B[u_f, v] = \overline{f(v)}^{1} \quad \forall v \in H_2$$

and

$$\|u_f\|_{H_1} \leq \frac{\|f\|_{H_2^*}}{C_3}$$

[1] $\overline{f(v)}$ is the complex conjugate of $f(v)$

5.1 System $H^1 \times H^1$

5.1.1 Preliminaries

Let $\epsilon, k \in H^1(\Omega)$

$$\begin{cases} \epsilon(x) - \Delta k(x) = & f_1(x) & \text{on } \Omega \\ -\Delta\epsilon(x) + k(x) = & f_2(x) & \text{on } \Omega \\ \epsilon = & g_1 & \text{on } \partial\Omega \\ k = & g_2 & \text{on } \partial\Omega. \end{cases} \tag{5.4}$$

To examine (5.4) we look at a similar system (5.5)

$$\begin{cases} u_1(x) - \Delta u_1(x) = & f_1(x) & \text{on } \Omega \\ -\Delta u_1(x) + u_2(x) = & f_2(x) & \text{on } \Omega \\ u_1 = & 0 & \text{on } \partial\Omega \\ u_2 = & 0 & \text{on } \partial\Omega. \end{cases} \tag{5.5}$$

We multiply it with a test function $(v_1, v_2) \in H^1(\Omega) \times H^1(\Omega)$ and integrate by parts to get

$$\begin{bmatrix} v_1 & v_2 \end{bmatrix} \cdot \begin{bmatrix} I & -\Delta \\ -\Delta & I \end{bmatrix} \cdot \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \cdot \begin{bmatrix} f_1 \\ f_2 \end{bmatrix}$$

$$\int_\Omega v_1 u_1 - v_1 \Delta u_2 - v_2 \Delta u_1 + v_2 u_1 \,\mathrm{d}x = \int_\Omega v_1 f_1 + v_2 f_2 \,\mathrm{d}x \tag{5.6}$$

$$\int_\Omega v_1 \epsilon + \nabla v_1 \cdot \nabla k + \nabla v_2 \cdot \nabla \epsilon + v_2 k \,\mathrm{d}x = \int_\Omega v_1 f_1 + v_2 f_2 \,\mathrm{d}x. \tag{5.7}$$

If we show that (5.7) holds for $(u_1, u_2) \in H^1_0(\Omega) \times H^1_0(\Omega)$, there must exists a pair $(w_1, w_2) \in H^1_{g_1}(\Omega) \times H^1_{g_2}(\Omega)$ such that $(\epsilon - w_1, k - w_2) = (u_1, u_2) \in H^1_0(\Omega) \times H^1_0(\Omega)$ and

$$\begin{aligned} \tilde{f}_1 =& f_1 - w_1 + \Delta w_2 \quad \in H^{-1}(\Omega) \\ \tilde{f}_2 =& f_2 + \Delta w_2 - w_1 \quad \in H^{-1}(\Omega). \end{aligned}$$

Thus with these substitutions (5.4) turns into this:

$$\begin{aligned} u_1(x) - \Delta u_2(x) =& \tilde{f}_1(x) \quad \text{on } \Omega \\ -\Delta u_1(x) + u_2(x) =& \tilde{f}_2(x) \quad \text{on } \Omega \\ u_1 =& 0 \quad \text{on } \partial\Omega \\ u_2 =& 0 \quad \text{on } \partial\Omega \end{aligned}$$

5.1.2 Using Babuška-Lax-Milgram

The proof in this section was created by the student.

If we assume that $(u_1, u_2) \in H_0^1(\Omega) \times H_0^1(\Omega)$

$$\begin{aligned}
|B[u,v]| &= \sqrt{\left|\int_\Omega u_1 v_1 + \nabla u_2 \cdot \nabla v_1 \,\mathrm{d}x\right|^2 + \left|\int_\Omega u_2 v_2 + \nabla u_1 \cdot \nabla v_2 \,\mathrm{d}x\right|^2} \\
&= \sqrt{\left(\left|\int_\Omega u_1 v_1 + \nabla u_2 \cdot \nabla v_1 \,\mathrm{d}x\right|\right)^2 + \left(\left|\int_\Omega u_2 v_2 + \nabla u_1 \cdot \nabla v_2 \,\mathrm{d}x\right|\right)^2} \\
&\leq \sqrt{\left(\int_\Omega |u_1 v_1 + \nabla u_2 \cdot \nabla v_1| \,\mathrm{d}x\right)^2 + \left(\int_\Omega |u_2 v_2 + \nabla u_1 \cdot \nabla v_2| \,\mathrm{d}x\right)^2} \\
&= \sqrt{\left(\|u_1 v_1 + \nabla u_2 \cdot \nabla v_1\|_{L^1(\Omega)}\right)^2 + \left(\|u_2 v_2 + \nabla u_1 \cdot \nabla v_2\|_{L^1(\Omega)}\right)^2}
\end{aligned}$$

applying Minkowskis inequality

$$\begin{aligned}
&\leq \sqrt{\left(\|u_1 v_1\|_{L^1(\Omega)} + \|\nabla u_2 \cdot \nabla v_1\|_{L^1(\Omega)}\right)^2 + \left(\|u_2 v_2\|_{L^1(\Omega)} + \|\nabla u_1 \cdot \nabla v_2\|_{L^1(\Omega)}\right)^2} \\
&\leq \Big(\left(\|u_1\|_{L^2(\Omega)}\|v_1\|_{L^2(\Omega)} + \|\nabla u_2\|_{L^2(\Omega)}\|\nabla v_1\|_{L^2(\Omega)}\right)^2 \\
&+ \left(\|u_2\|_{L^2(\Omega)}\|v_2\|_{L^2(\Omega)} + \|\nabla u_1\|_{L^2(\Omega)}\|\nabla v_2\|_{L^2(\Omega)}\right)^2 \Big)^{\frac{1}{2}}
\end{aligned}$$

$$\begin{aligned}
&= \Big(\|u_1\|^2_{L^2(\Omega)}\|v_1\|^2_{L^2(\Omega)} + \|\nabla u_2\|^2_{L^2(\Omega)}\|\nabla v_1\|^2_{L^2(\Omega)} \\
&+ 2\|u_1\|_{L^2(\Omega)}\|v_1\|_{L^2(\Omega)}\|\nabla u_2\|_{L^2(\Omega)}\|\nabla v_1\|_{L^2(\Omega)} \\
&+ \|u_2\|^2_{L^2(\Omega)}\|v_2\|^2_{L^2(\Omega)} + \|\nabla u_1\|^2_{L^2(\Omega)}\|\nabla v_2\|^2_{L^2(\Omega)} \\
&+ 2\|u_2\|_{L^2(\Omega)}\|v_2\|_{L^2(\Omega)}\|\nabla u_1\|_{L^2(\Omega)}\|\nabla v_2\|_{L^2(\Omega)}\Big)^{\frac{1}{2}} \\
&\leq \Big(\|u_1\|^2_{H^1(\Omega)}\|v_1\|^2_{H^1(\Omega)} + \|u_2\|^2_{H^1(\Omega)}\|v_1\|^2_{H^1(\Omega)} \\
&+ 2\|u_1\|_{H^1(\Omega)}\|v_1\|_{H^1(\Omega)}\|u_2\|_{H^1(\Omega)}\|v_1\|_{H^1(\Omega)} \\
&+ \|u_2\|^2_{H^1(\Omega)}\|v_2\|^2_{H^1(\Omega)} + \|u_1\|^2_{H^1(\Omega)}\|v_2\|^2_{H^1(\Omega)} \\
&+ 2\|u_2\|_{H^1(\Omega)}\|v_2\|_{H^1(\Omega)}\|u_1\|_{H^1(\Omega)}\|v_2\|_{H^1(\Omega)}\Big)^{\frac{1}{2}}
\end{aligned}$$

$$
\begin{aligned}
&= \Big(\Big(\|u_1\|^2_{H^1(\Omega)}\|v_1\|^2_{H^1(\Omega)} + \|u_2\|^2_{H^1(\Omega)}\|v_1\|^2_{H^1(\Omega)} \\
&+ \|u_2\|^2_{H^1(\Omega)}\|v_2\|^2_{H^1(\Omega)} + \|u_1\|^2_{H^1(\Omega)}\|v_2\|^2_{H^1(\Omega)}\Big) \\
&+ 2\|u_1\|_{H^1(\Omega)}\|v_1\|_{H^1(\Omega)}\|u_2\|_{H^1(\Omega)}\|v_1\|_{H^1(\Omega)} \\
&+ 2\|u_2\|_{H^1(\Omega)}\|v_2\|_{H^1(\Omega)}\|u_1\|_{H^1(\Omega)}\|v_2\|_{H^1(\Omega)}\Big)^{\frac{1}{2}}
\end{aligned}
$$

$$
\begin{aligned}
&= \Big(\Big(\|u_1\|^2_{H^1(\Omega)} + \|u_2\|^2_{H^1(\Omega)}\Big)\Big(\|v_1\|^2_{H^1(\Omega)} + \|v_2\|^2_{H^1(\Omega)}\Big) \\
&+ 2\|u_1\|_{H^1(\Omega)}\|u_2\|_{H^1(\Omega)}\Big(\|v_1\|^2_{H^1(\Omega)} + \|v_2\|^2_{H^1(\Omega)}\Big)\Big)^{\frac{1}{2}} \\
&= \Big(\ \|u\|^2_{H^1(\Omega)\times H^1(\Omega)}\|v\|^2_{H^1(\Omega)\times H^1(\Omega)} \\
&+ 2\|u_1\|_{H^1(\Omega)}\|u_2\|_{H^1(\Omega)}\|v\|^2_{H^1(\Omega)\times H^1(\Omega)}\Big)^{\frac{1}{2}} \\
&= \|v\|_{H^1(\Omega)\times H^1(\Omega)}\Big(\|u\|^2_{H^1(\Omega)\times H^1(\Omega)} + 2\|u_1\|_{H^1(\Omega)}\|u_2\|_{H^1(\Omega)}\Big)^{\frac{1}{2}}
\end{aligned}
$$

by Youngs Inequality

$$
\begin{aligned}
&\le \|v\|_{H^1(\Omega)\times H^1(\Omega)}\Big(\|u\|^2_{H^1(\Omega)\times H^1(\Omega)} + \|u_1\|^2_{H^1(\Omega)} + \|u_2\|^2_{H^1(\Omega)}\Big)^{\frac{1}{2}} \\
&= \|v\|_{H^1(\Omega)\times H^1(\Omega)}\Big(\|u\|^2_{H^1(\Omega)\times H^1(\Omega)} + \|u\|^2_{H^1(\Omega)\times H^1(\Omega)}\Big)^{\frac{1}{2}} \\
&= \sqrt{2}\|v\|_{H^1(\Omega)\times H^1(\Omega)}\|u\|_{H^1(\Omega)\times H^1(\Omega)}.
\end{aligned}
$$

Thus the requirement (5.1) is met by:

$$
|B[u,v]| \le \sqrt{2}\|v\|_{H^1(\Omega)\times H^1(\Omega)}\|u\|_{H^1(\Omega)\times H^1(\Omega)}.
$$

Next step is to prove the coercivity requirements (5.2) and (5.3) .

$$
\begin{aligned}
B[u,w] =& \int_\Omega u_1 w_1 + \nabla u_2 \cdot \nabla w_1 \,\mathrm{d}x + \int_\Omega u_2 w_2 + \nabla u_1 \cdot \nabla w_2 \,\mathrm{d}x \\
=&(u_1, w_1)_{L^2(\Omega)} + (u_2, w_1)_{H^1_0(\Omega)} + (u_2, w_2)_{L^2(\Omega)} + (u_1, w_2)_{H^1_0(\Omega)}.
\end{aligned}
$$

We prove the coercivity requirements by finding a function that fulfills it. This can be done because the coercivity requirements is a sup requirement. By careful construction of a candidate function the sup requirement is met. This is done with

$$w = \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} = K \begin{bmatrix} u_2 - u_1 + u_1 \\ u_1 + u_2 - u_2 \end{bmatrix} \quad \left(= K \begin{bmatrix} u_2 \\ u_1 \end{bmatrix} \right) \tag{5.8}$$

$$K = \frac{1}{\sqrt{\|u_1\|^2_{H^1(\Omega)} + \|u_2\|^2_{H^1(\Omega)}}}. \tag{5.9}$$

Thus $\|w\|_{H^1(\Omega)\times H^1(\Omega)} = 1$, so the requirements is met.

$$\begin{aligned}
&= K(u_1, u_2 - u_1 + u_1)_{L^2(\Omega)} + K(u_2, u_2 - u_1 + u_1)_{H^1_0(\Omega)} \\
&+ K(u_2, u_1 + u_2 - u_2)_{L^2(\Omega)} + K(u_1, u_1 + u_2 - u_2)_{H^1_0(\Omega)} \\
&= K(u_1, u_2)_{L^2(\Omega)} - K(u_1, u_1)_{L^2(\Omega)} + K(u_1, u_1)_{L^2(\Omega)} \\
&+ K(u_2, u_2)_{H^1_0(\Omega)} - K(u_2, u_1)_{H^1_0(\Omega)} + K(u_2, u_1)_{H^1_0(\Omega)} \\
&+ K(u_2, u_1)_{L^2(\Omega)} - K(u_2, u_2)_{L^2(\Omega)} + K(u_2, u_2)_{L^2(\Omega)} \\
&+ K(u_1, u_1)_{H^1_0(\Omega)} + K(u_1, u_2)_{H^1_0(\Omega)} - K(u_1, u_2)_{H^1_0(\Omega)} \\
&= K\|u_1\|^2_{H^1_0(\Omega)} + K\|u_2\|^2_{H^1_0(\Omega)} + K\|u_1\|^2_{L^2_(\Omega)} + K\|u_2\|^2_{L^2_(\Omega)} \\
&- K\|u_2\|^2_{L^2(\Omega)} - K\|u_1\|^2_{L^2(\Omega)} + 2K(u_2, u_1)_{L^2(\Omega)} \\
&= K\|u_1\|^2_{H^1_0(\Omega)} + K\|u_2\|^2_{H^1_0(\Omega)} + K\|u_1\|^2_{L^2_(\Omega)} + K\|u_2\|^2_{L^2_(\Omega)} - K\|u_1 - u_2\|^2_{L^2(\Omega)} \\
&= K\|u_1\|^2_{H^1_0(\Omega)} + K\|u_2\|^2_{H^1_0(\Omega)} + K\|u_1\|^2_{L^2_(\Omega)} + K\| - u_2\|^2_{L^2_(\Omega)} - K\|u_1 - u_2\|^2_{L^2(\Omega)} \\
&\quad \text{minowskis inequality} \\
&\geq K\|u_1\|^2_{H^1_0(\Omega)} + K\|u_2\|^2_{H^1_0(\Omega)} + K\|u_1 - u_2\|^2_{L^2_(\Omega)} - K\|u_1 - u_2\|^2_{L^2(\Omega)} \\
&= K\|u_1\|^2_{H^1_0(\Omega)} + K\|u_2\|^2_{H^1_0(\Omega)} \\
&= \frac{K}{2}\|u_1\|^2_{H^1_0(\Omega)} + \frac{K}{2}\|u_2\|^2_{H^1_0(\Omega)} + \frac{K}{2}\|u_1\|^2_{H^1_0(\Omega)} + \frac{K}{2}\|u_2\|^2_{H^1_0(\Omega)} \\
&\quad \text{Poincares inequality} \\
&\geq \frac{K}{2}\|u_1\|^2_{H^1_0(\Omega)} + \frac{K}{2}\|u_2\|^2_{H^1_0(\Omega)} + \frac{C_1}{2}\|u_1\|^2_{L^2(\Omega)} + \frac{C_2}{2}\|u_2\|^2_{L^2(\Omega)} \\
&\geq D\Big(\|u_1\|^2_{H^1(\Omega)} + \|u_2\|^2_{H^1(\Omega)}\Big) \\
&= D\|u\|^2_{H^1(\Omega)\times H^1(\Omega)}
\end{aligned}$$

So the requirement (5.3). To show (5.2), note that $B[\cdot,\cdot]$ is symmetric so:

$$\begin{aligned}
|B[u,w]| &= \left| B\left[\left[\begin{array}{c} u_1 \\ u_2 \end{array} \right], K \left[\begin{array}{c} u_2 \\ u_1 \end{array} \right] \right] \right| \\
&= \left| KB\left[\left[\begin{array}{c} u_1 \\ u_2 \end{array} \right], \left[\begin{array}{c} u_2 \\ u_1 \end{array} \right] \right] \right| \\
&= \left| KB\left[\left[\begin{array}{c} u_2 \\ u_1 \end{array} \right], \left[\begin{array}{c} u_1 \\ u_2 \end{array} \right] \right] \right| \\
&= \left| B\left[\left[\begin{array}{c} u_2 \\ u_1 \end{array} \right], K \left[\begin{array}{c} u_1 \\ u_2 \end{array} \right] \right] \right| \\
&= |B[w,u]|
\end{aligned}$$

Which fulfills the requirement (5.2).

5.2 System $H^2 \times L^2$

The argument in this section was created by the student. For simplicity we write H^{-2} for the dual space of H^2, and $H^{-2} \times L^2$ will be identified as the dual space of $H^2 \times L^2$,we also assume that $\Omega = (0,1)^n$

5.2.1 Preliminaries

Before looking at the problem, we can examine a candidate for preconditioning for the system. The preconditioner will be the matrix arising from the problem (5.10)

$$\left\{ \begin{array}{rcll} \Delta\Delta u_1(x) + u_1 = & f_1(x) & on\, \Omega \\ u_2 = & f_2(x) & on\, \Omega \\ u_1 = & 0 & on\, \partial\Omega \\ \frac{\partial u_1}{\partial n} = & 0 & on\, \partial\Omega \end{array} \right. \tag{5.10}$$

The bilinear from associated with (5.10) is (5.11).

$$P[u,v] = \int_\Omega \Delta u_1 \Delta v_1 + u_1 v_1 + u_2 v_2 \,\mathrm{d}x \tag{5.11}$$

$P[u,v]$ is a positive definite and symmetric, so its an inner product on $H^2(\Omega) \times L^2(\Omega)$. The system (5.10) has a unique solution for every pair f_1, f_2 in the dual space of $H^2(\Omega) \times L^2(\Omega)$ by Riesz representation theorem. It is obvious that $P \in \mathcal{L}(H^2 \times L^2, H^{-2} \times L^2)$ (the space of linear functionals from $H^2 \times L^2$ to $H^{-2} \times L^2$). Since P^{-1} exists, it is a member of $\mathcal{L}(H^{-2} \times L^2, H^2 \times L^2)$. P has a unique solution, we know that P has an inverse so that that $P^{-1} \circ P = I_{H^2 \times L^2}$ (the identity functional on $H^2 \times L^2$).

5.2.2 The bilinear form

A similar argument as the one made in 5.1.1 can be made to show that we only need to prove uniqueness for (5.12) with $\{u_1, v_1\} \in H_0^2(\Omega), \{u_2, v_2\} \in L^2(\Omega)$

The bilinear form of the system looks like:

$$B[u,v] = \int_\Omega -\Delta v_1 u_2 - \Delta u_1 v_2 + u_1 v_1 + u_2 v_2 \, \mathrm{d}x. \tag{5.12}$$

With the matrix

$$B = \begin{bmatrix} I & -\Delta \\ -\Delta & I \end{bmatrix}. \tag{5.13}$$

If we let the operator matrix B work on a pair of functions $u = (u_1, u_2) \in H^2 \times L^2$, then

$$\begin{aligned} B \cdot u &= \begin{bmatrix} I & -\Delta \\ -\Delta & I \end{bmatrix} \cdot \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \\ &= \begin{bmatrix} u_1 - \Delta u_2 \\ -\Delta u_1 + u_2 \end{bmatrix} \quad \in \begin{bmatrix} H^{-2} \\ L^2 \end{bmatrix} \end{aligned}$$

Thus $B \in \mathcal{L}(H^2 \times L^2, H^{-2} \times L^2)$ as well. We conclude that $P^{-1} \circ B \in \mathcal{L}(H^2 \times L^2, H^2 \times L^2)$.

We know $(P^{-1} \circ P)$ is bounded and 0 is not part of its spectrum If the same holds true for $(P^{-1} \circ B)$, we can use Lax-Milgram to prove uniqueness of solutions for $(P^{-1} \circ B)[u,v]$.

First we need to show that the maximal and the minimal eigenvalue of $P^{-1}B$ is limited by constants c_1, c_2.

$$\begin{cases} P = \begin{bmatrix} \Delta\Delta + I & 0 \\ 0 & I \end{bmatrix} \\ B = \begin{bmatrix} I & -\Delta \\ -\Delta & I \end{bmatrix} \end{cases} \tag{5.14}$$

The corresponding eigenvalue problem of (5.14) is (5.15)

$$\begin{bmatrix} I & -\Delta \\ -\Delta & I \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \lambda \begin{bmatrix} \Delta\Delta + I & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix}. \tag{5.15}$$

If we are working are working on $C_c^4([0,1]^n)$ the operator $-\Delta$ has eigenvalues $\lambda_D = \{k^2\pi^2\}_{k=1}^N \in (\pi^2, \infty)$ [12, p.67]. When applying FEM to 5.15, it turns into a matrix problem; 5.16.

$$\begin{bmatrix} I & C \\ C & I \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \lambda \begin{bmatrix} D & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix}. \tag{5.16}$$

C is rectangular because $u, v \in H^2 \times L^2$, so instead of squaring, we use D.

Since the function space is $C_c^4([0,1]^n)$ the eigenvalues take the from $\lambda_k = (k\pi L(\Omega))^2$ [12], and have single multiplicity. Let e_i be the eigenvector associated with λ_i as an eigenvalue for $-\Delta$. Then we have that $u_i = [ae_i, be_i]$ is an eigenvalue for (5.15).

$$\begin{aligned} \begin{bmatrix} I & -\Delta \\ -\Delta & I \end{bmatrix} \begin{bmatrix} ae_i \\ be_i \end{bmatrix} &= \rho \begin{bmatrix} \Delta\Delta + I & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} ae_i \\ be_i \end{bmatrix} \\ \begin{bmatrix} I & \lambda_i I \\ \lambda_i I & I \end{bmatrix} \begin{bmatrix} ae_i \\ be_i \end{bmatrix} &= \rho \begin{bmatrix} \lambda_i^2 I + I & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} ae_i \\ be_i \end{bmatrix}. \end{aligned}$$

Further reductions is possible:

$$\begin{aligned} \begin{bmatrix} I & \lambda_i I \\ \lambda_i I & I \end{bmatrix} \begin{bmatrix} ae_i \\ be_i \end{bmatrix} &= \rho \begin{bmatrix} \lambda_i^2 I + I & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} ae_i \\ be_i \end{bmatrix} \\ \left(\begin{bmatrix} 1 & \lambda_i \\ \lambda_i & 1 \end{bmatrix} \otimes I \right) \left(\begin{bmatrix} a \\ b \end{bmatrix} \otimes e_i \right) &= \rho \left(\begin{bmatrix} \lambda_i^2 + 1 & 0 \\ 0 & 1 \end{bmatrix} \otimes I \right) \left(\begin{bmatrix} a \\ b \end{bmatrix} \otimes e_i \right) \\ \left(\begin{bmatrix} 1 & \lambda_i \\ \lambda_i & 1 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} \right) \otimes (Ie_i) &= \rho \left(\begin{bmatrix} \lambda_i^2 + 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} \right) \otimes (Ie_i). \end{aligned}$$

Where $\otimes$ is the Kronecker product. Then the eigenvalues are the eigenvalues of (5.17).

$$\left(\begin{bmatrix} 1 & \lambda_i \\ \lambda_i & 1 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} \right) = \rho \left(\begin{bmatrix} \lambda_i^2 + 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} \right). \tag{5.17}$$

The eigenvalues of (5.17) takes the form

$$\rho_i = \frac{\lambda_i^2 + 2 + |\lambda_i|\sqrt{5\lambda_i^2 + 4}}{2\lambda_i^2 + 2} \tag{5.18}$$

$$\rho_i' = \frac{\lambda_i^2 + 2 - |\lambda_i|\sqrt{5\lambda_i^2 + 4}}{2\lambda_i^2 + 2}. \tag{5.19}$$

Where $\{\rho_i\}, \{\rho'\}$ is two sets of eigenvalues. Zero is not an eigenvalue;

$$\begin{aligned}
\rho_i =& \frac{\lambda_i^2 + 2 + |\lambda_i|\sqrt{5\lambda_i^2 + 4}}{2\lambda_i^2 + 2} \\
0 =& \frac{\lambda_i^2 + 2 + |\lambda_i|\sqrt{5\lambda_i^2 + 4}}{2\lambda_i^2 + 2} \\
& \text{no solution for } \lambda \geq 1 \\
\rho_i' =& \frac{\lambda_i^2 + 2 - \lambda_i\sqrt{5\lambda_i^2 + 4}}{2\lambda_i^2 + 2} \\
0 =& \frac{\lambda_i^2 + 2 + \lambda_i\sqrt{5\lambda_i^2 + 4}}{2\lambda_i^2 + 2} \\
& \begin{cases} \lambda_{i,1} = 1 \\ \lambda_{i,2} = -1 \end{cases}
\end{aligned}$$

Zero is not an eigenvalue of the system since$\lambda \in (\pi^2, \infty)$. We see that, when discretizing of the operator $-\Delta$, the set $(\pi k)^2{}_{k=1}^{N}$ will be all the eigenvalues of the $N \times N$ matrix representing the discretization. When the system (5.15) is discretized, it is easy to confirm that there will be $2N$ eigenvalues in the set $\{\rho(\lambda_i)\} \cup \{\rho'(\lambda_i)\}$. Since there is exactly as many eigenvalues as there is rows, we conclude that all of the eigenvalues are found. The eigenvalues has single multiplicity. All the eigenfunctions are linearly independent and span $\mathbb{R}^{2N}$ (see 11.2). Therefore we conclude that we have found all the eigenvectors (discretized eigenfunctions). We find all of the eigenfunctions first by solving the systems (5.20) and (5.21).

$$\begin{bmatrix} 1 & \lambda_i \\ \lambda_i & 1 \end{bmatrix} \begin{bmatrix} a_i \\ b_i \end{bmatrix} = \begin{bmatrix} \rho_i\lambda_i^2 + \rho_i & 0 \\ 0 & \rho_i \end{bmatrix} \begin{bmatrix} a_i \\ b_i \end{bmatrix} \tag{5.20}$$

$$\begin{bmatrix} 1 & \lambda_i \\ \lambda_i & 1 \end{bmatrix} \begin{bmatrix} c_i \\ d_i \end{bmatrix} = \begin{bmatrix} \rho_i'\lambda_i^2 + \rho_i' & 0 \\ 0 & \rho_i' \end{bmatrix} \begin{bmatrix} c_i \\ d_i \end{bmatrix}. \tag{5.21}$$

The eigenfunctions will be with their respective eigenvalues will be

$$\begin{bmatrix} a_i e_i \\ b_i e_i \end{bmatrix}, \rho_i \tag{5.22}$$

$$\begin{bmatrix} c_i e_i \\ d_i e_i \end{bmatrix}, \rho_i'. \tag{5.23}$$

By letting λ tend to infinity, the eigenvalues converge towards

$$\lim_{i\to\infty} \rho_i = \frac{1+\sqrt{5}}{2} \tag{5.24}$$

$$\lim_{i\to\infty} \rho'_i = \frac{1-\sqrt{5}}{2} \tag{5.25}$$

$$\tag{5.26}$$

These are the golden ratio (ϕ), and the negative golden ratio conjugate. Since both $|\rho'_i|, |\rho_i|$ are strictly increasing when i increases, the condition number of $P^{-1}B$ is be limited by,

$$\lim \kappa(P^{-1}B) \geq \left| \frac{\frac{1+\sqrt{5}}{2}}{\frac{\lambda_0^2+2-\lambda_0\sqrt{5\lambda_0^2+4}}{2\lambda_0^2+2}} \right| \tag{5.27}$$

$$= \left| \frac{\frac{1+\sqrt{5}}{2}}{\frac{\pi^4+2-\sqrt{5\pi^4+4\pi^2}}{2\pi^4+2}} \right| \approx \frac{1.6180}{0.606127} \approx 2.66946 \tag{5.28}$$

$$\tag{5.29}$$

So we have that $c_1 \geq \|P^{-1}B\|$ and $c_2 \leq \|(P^{-1}B)^{-1}\|$, then $|(P^{-1}Bu, v)| \leq c_1|(u,v)| \leq c_1\|u\|\|v\|$. And also, $\inf_u \sup_v \frac{(P^{-1}Bu,v)}{\|u\|\|v\|} \geq c_2^{-1}$. Since the bilinear form is symmetric, $\inf_v \sup_u \frac{(P^{-1}Bu,v)}{\|u\|\|v\|} \geq c_2^{-1}$ is also true. All of the requirements of the Babuška-Lax-Milgram Theorem are meet, so the bilinear form $P^{-1} \circ B[u,v] = L_p(v)$ has an unique solution. This can only be the case if $B[u,v] = L_b(v)$ has an unique solution.

Chapter 6

Technical issues and implementation

6.1 Condition numbers

There are three different definitions of condition numbers used in this thesis.

Condition Number Definition 1 *Let A be $n \times n$ matrix with real eigenvalues* $0 \leq \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$. *Then the condition number is*

$$\kappa_1(A) = \frac{\lambda_n}{\lambda_1}.$$

Condition Number Definition 2 *Let A be $n \times n$ matrix with real eigenvalues $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$. Then the condition number is*

$$\kappa_2(A) = \frac{\max_i |\lambda_i|}{\min_j |\lambda_j|}.$$

Condition Number Definition 3 *Let A be a real $m \times n$ matrix with real and complex eigenvalues $\{\lambda_i\}_{i=1}^{n}$. Let $B = A^T A$, with real eigenvalues* $0 \leq \gamma_1 \leq \gamma_2 \leq \cdots \leq \gamma_n$

$$\kappa_3(A) = \frac{\sqrt{\gamma_1}}{\sqrt{\gamma_n}}.$$

6.2 Python issues

The simplified k-ϵ model is implemented in python. The libraries numpy [13], scipy [13] and syFi [14] are used. The inverting algorithm in scipy is to unstable for the purposes of this thesis.

```
scipy.linalg.inv(A)
```

Instead the matrixes are generated in python, and written in ".m" format. Then they where implemented in octave [15], and the "qz" algorithm is used. This approach proved more stable. A sample code for doing this looks like:

```
nMax= 8


 def run(N) :
    #calculating the matrixes
    .
    .
    .
    .


    fi_Str = "␣"

    for key in AA.keys():
        i,j = key
        d[i][j] = float(AA[key])
        fi_Str += "A%d(%d,␣%d)=%s␣;␣\n␣" % (N,i+1,j+1,d[i][j])

    prCo = zeros([m,m],float)
    for key in PC.keys():
        i,j = key

        prCo[i][j] = float(PC[key])
        fi_Str += "Pc%d(%d,␣%d)=%s;\n" % (N,i+1,j+1,prCo[i][j])
    #eigenvalues of system
     fi_Str += "v%d␣=␣␣eig(A%d)␣;␣␣\n" % (N,N)
    fi_Str += "C%d␣=␣␣max(abs(v%d))/min(abs(v%d))␣␣␣\n" % (N,N,N)
    #qz for finding the list of eigenvalues
    fi_Str += "␣eg%d␣=␣sort(qz(Pc%d,␣A%d))␣;␣\n" % (N,N,N)
    fi_Str += "␣␣eg%d(1)␣␣;\n" % (N)
    fi_Str += "␣eg%d(size(eg%d)(1)␣)␣;\n␣" % (N,N)
    #printing out the absolute values, they are close to 1
    fi_Str += "␣␣maxEgenverdi%d␣=␣max(abs(eg%d))␣␣\n" % (N,N)
    fi_Str += "␣␣minEgenverdi%d␣=␣min(abs(eg%d))␣␣\n" % (N,N)
    fi_Str + = "␣Cond%d=maxEgenverdi%d/minEgenverdi%d\n"%(N,N,N)

    return fi_Str

 if __name__== "__main__" :

    f = open("testPrec.m",’w’)
    fi_str = ""
    for i in range (0, nMax):
    #print "␣dette␣er␣’i’␣" ,i
        print 2**i
    # N = 125 = breakdown
        a = run(2**i)
        fi_str += a

    f.write(fi_str)
```

6.3 SyFi

SyFi finite element package SyFi it is a C++ library built on top of the symbolic math library GiNaC[16]. In this thesis it is used to generate the finite element matrixes. This is a sample code used to generate a Hermite and a discontinuous Lagrange reference matrix.

```
import SyFi as SF

SF.initSyFi(2)

l = SF.Line([0.0, 0.0], [1.0/N,0.0])
fe = SF.Hermite(l)

lagOrder = 2
feLa = SF.DiscontinuousLagrange(l,lagOrder-1)

#hermite mass matrix
A = {}
#hermite biharmonic matrix
B = {}
#hermite mass matrix
M = {}
#biharmoic matrix with mass matrix
PaC = {}
weig = N
for i in range(0, 4):
    if i%2 == 1 :
        p1 = weig
    else :
        p1 = 1

    for j in range(0, 4):
        if j%2 ==1 :
            p2 = weig
        else :
            p2 = 1
        M[(i,j)] = l.integrate(fe.N(i)*fe.N(j))*p1*p2
        B[(i,j)] =  l.integrate(swig.diff(swig.diff(fe.N(i),x),x) *
            swig.diff(swig.diff(fe.N(j),x),x))*p1*p2
        A[(i,j)] = M[(i,j)]
        PaC[(i,j)] = B[(i,j)]+ M[(i,j)]

#Lagrange mass function
M2 = {}
```

```
for i in range(0, lagOrder):
    for j in range(0, lagOrder):
        M2[(i,j)] = l.integrate(feLa.N(i)*feLa.N(j))

#mixed herm-lag matrix
AD = {}
for i in range(0, 4):
    if i%2 == 1 :
        p1 = weig
    else :
        p1 = 1
    for j in range(0,lagOrder):
        AD[(i,j)] = l.integrate(-feLa.N(j)*swig.diff(swig.diff(fe.N(i),x),x))*p1
```

Chapter 7

Hermite elements

7.1 Hermite elements in 1D

The basis functions on the real line, where the element spans $[0, 1]$ has the following basis functions

$$\begin{aligned}\phi_0 =&2x^3 - 3x^2 + 1\\ \phi_1 =&-2x^3 + 3x^2\\ \phi_2 =&x^3 - 2x^2 + x\\ \phi_3 =&x^3 - x^2.\end{aligned}$$

For a visual representation of the Hermite basis functions see figures: 7.1, 7.2, 7.3, 7.4.

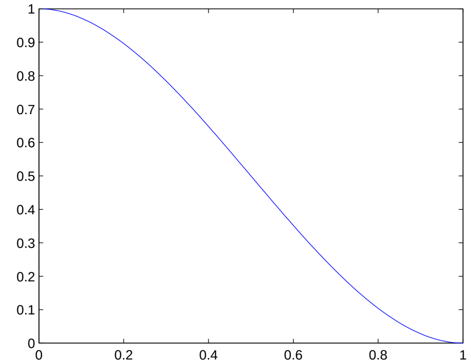


Figure 7.1: ϕ_0, the basis function evaluation value at 0.

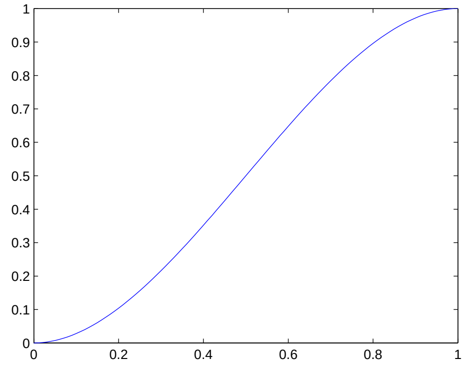


Figure 7.2: ϕ_1, the basis function evaluation value at 1.

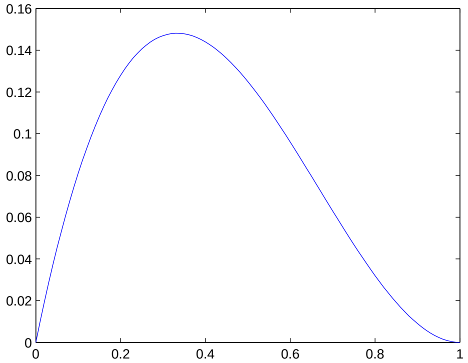


Figure 7.3: ϕ_2, the basis function evaluation derivative at 0.

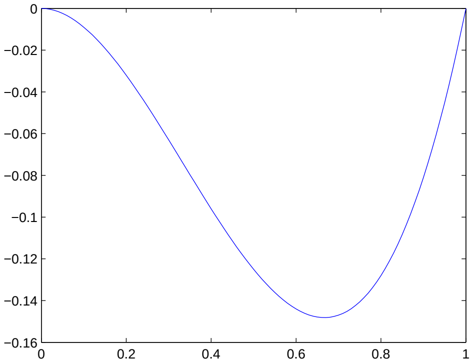


Figure 7.4: ϕ_3, the basis function evaluation derivative at 1.

7.2 Scaling Hermite elements 1D

There are a few important issues to notice when working with implementing FEM on this problem. The first issue is that has to be worked out an issue relating to scaling. When the mesh grid becomes more refined, the norm of different basis functions scales differently for Hermite elements. This issue caused a great deal of problems in the simulations before it was identified. If left unresolved, the matrices quickly becomes very unstable when the mesh is refined. Half of the diagonal values on the Hermite FEM matrix decreases N^2 times faster than the other diagonal values (as shown later in this chapter). This makes the FEM approximation unstable when confined to double precision.

The first attempt to solve this issue was to increase the mesh size, instead of refining it. The normal procedure for refinement is :

$$\begin{aligned}[0,1] \to \{[0,\frac{1}{2}],[\frac{1}{2},1]\} \to \{[0,\frac{1}{4}],[\frac{1}{4},\frac{2}{4}],[\frac{2}{4},\frac{3}{4}],[\frac{3}{4},1]\} \\ \to \cdots \to \{[0,\frac{1}{N}],\ldots,[\frac{N-1}{N},1]\}.\end{aligned}$$

This was changed to

$$[0,1] \to \{[0,1],[1,2]\} \to \cdots \to \{[0,1],\ldots,[N-1,N]\}.$$

This refinement mapping was implemented and tested. It solved the issue of the diagonal values but, was discarded for two reasons:

- The mass matrix (matrix representing I) dominated the eigenvalue analysis.
- The eigenvalue analysis in the literature is dependent on a fixed grid.

To mimic the positive results of the domain scaling idea, scaling of the basis functions is introduced. The basis functions associated with the rapidly decreasing diagonal values are the basis functions representing the derivative. These basis functions are multiplied with a number dependent on the mesh. This solves the issue. Note that after solving a scaled system, the values have to be rescaled to obtain the actual solution.

Next we calculate how to scale the basis functions. We compare how the diagonal values will scale when the mesh is refined. The first integrals are

$$\int_0^1 \phi_0 \,\mathrm{d}x = \int_0^1 \phi_1 \,\mathrm{d}x = \frac{1}{2}$$
$$\int_0^1 \phi_2 \,\mathrm{d}x = -\int_0^1 \phi_3 \,\mathrm{d}x = \frac{1}{12}.$$

By calculating the basis functions for the Hermite element on the domain $[0,\frac{1}{N}]$

and sets $N = 16$ we get:

$$\begin{aligned}\phi_0^N =& 8192x^3 - 768x^2 + 1\\ \phi_1^N =& -8192x^3 + 768x^2\\ \phi_2^N =& 256x^3 - 32x^2 + x\\ \phi_3^N =& 256x^3 - 16x^2.\end{aligned}$$

The integrals are

$$\int_0^{\frac{1}{16}} \phi_0^N \,\mathrm{d}x = \int_0^{\frac{1}{16}} \phi_1^N \,\mathrm{d}x = \frac{1}{32}$$
$$\int_0^{\frac{1}{16}} \phi_2^N \,\mathrm{d}x = -\int_0^{\frac{1}{16}} \phi_3^N \,\mathrm{d}x = \frac{1}{3072}.$$

Divide ϕ_0 on ϕ_0^N and ϕ_1 on ϕ_1^N:

$$\begin{aligned}\frac{\int_0^1 \phi_0 \,\mathrm{d}x}{\int_0^{\frac{1}{16}} \phi_0^N \,\mathrm{d}x} =& \frac{\frac{1}{2}}{\frac{1}{32}} = 16 = N\\ \frac{\int_0^1 \phi_2 \,\mathrm{d}x}{\int_0^{\frac{1}{16}} \phi_2^N \,\mathrm{d}x} =& \frac{\frac{1}{12}}{\frac{1}{3072}} = 256 = N^2.\end{aligned}$$

Computed directly the condition number of a matrix produced with unscaled basis functions is bound to run into rounding error. This is illustrated by the table 7.2 (figures: 7.10, 7.11, 7.12, 7.13, 7.14). To avoid this problem, new scaled basis functions are produced. The increased stability of the basis functions of equation (7.2) over the natural reference basis functions (7.1), is easily demonstrated by the table 7.1 (figures: 7.5, 7.6, 7.7, 7.8 and 7.9) as compared to table 7.2. As we can see in the table, the M^H changes as $O(1)$. The mass matrix represents equations of the type $u = f$ and should therefore have stable condition numbers. A^H changes as $O(2^N)$, this is as good as it should be, and a big improvement over the table 7.2. A^D, scales as $O(2^N)$, but it starts as a very large number. B scales as $O(4^N)$, as it should.

$$\begin{cases}\phi_0^N(0) & = 1\\ \frac{\partial \phi_1^N(0)}{x} & = 1\\ \phi_2^N(1) & = 1\\ \frac{\partial \phi_3^N(1)}{\partial x} & = 1\end{cases} \tag{7.1}$$

$$\begin{cases}\phi_0^N(0) = N\\ \frac{\partial \phi_1^N(0)}{\partial x} = N\\ \phi_2^N(1) = N\\ \frac{\partial \phi_3^N(1)}{\partial x} = N.\end{cases} \tag{7.2}$$

Table 7.1: Condition numbers (κ_1) for scaled matrices from a one dimensional grid.

number of	M^H	$A^H + M^H$	$A^D + M^H$	$B + M^H$
elements	cond number	cond number	cond number	cond number
$1 = 2^0$	1056. 67	31. 99	$7.\ 1703 \cdot 10^4$	$9.\ 1266 \cdot 10^2$
$2 = 2^1$	878. 62	37. 05	$1.\ 9623 \cdot 10^5$	$4.\ 7009 \cdot 10^3$
$4 = 2^2$	797. 01	88. 31	$1.\ 7188 \cdot 10^6$	$3.\ 2143 \cdot 10^4$
$8 = 2^3$	807. 56	336. 34	$2.\ 8839 \cdot 10^7$	$3.\ 1096 \cdot 10^5$
$16 = 2^4$	817. 33	1295. 55	$6.\ 3369 \cdot 10^8$	$3.\ 8925 \cdot 10^6$
$32 = 2^5$	820. 07	5058. 34	$1.\ 4050 \cdot 10^{10}$	$5.\ 5665 \cdot 10^7$
$64 = 2^6$	820. 79	19957. 34	$3.\ 1416 \cdot 10^{11}$	$8.\ 4549 \cdot 10^8$
$128 = 2^7$	821. 97	79246. 83	$7.\ 0636 \cdot 10^{12}$	$1.\ 3197 \cdot 10^{10}$
$256 = 2^8$	821. 01	315790. 78	$1.\ 5919 \cdot 10^{14}$	$2.\ 0861 \cdot 10^{11}$
$512 = 2^9$	821. 02	1260737. 96	$3.\ 5711 \cdot 10^{15}$	$3.\ \ 3179 \cdot 10^{12}$
$1024 = 2^{10}$	821. 03	5038069. 14	$7.\ 0389 \cdot 10^{16}$	$5.\ 2948 \cdot 10^{13}$

$$
\begin{aligned}
B_{i,j} &= \int \Delta H_i \Delta H_j \,\mathrm{d}x \\
A^H_{i,j} &= \int DH_i \cdot DH_j \,\mathrm{d}x \\
A^D_{i,j} &= \int -H_i \Delta H_j \,\mathrm{d}x \\
S^D_{i,j} &= \int -L_i \Delta H_j \,\mathrm{d}x \\
S^S_{i,j} &= \int \nabla L_i \cdot \nabla H_j \,\mathrm{d}x \\
A^L_{i,j} &= \int \nabla L_i \cdot \nabla L_j \,\mathrm{d}x \\
M^S_{i,j} &= \int H_i L_j \,\mathrm{d}x \\
M^L_{i,j} &= \int L_i L_j \,\mathrm{d}x \\
M^H_{i,j} &= \int H_i H_j \,\mathrm{d}x
\end{aligned}
$$

Figure 7.5: Plot of the $2 - \log$ condition number of scaled matrices in 1D as described in the table 7.1.

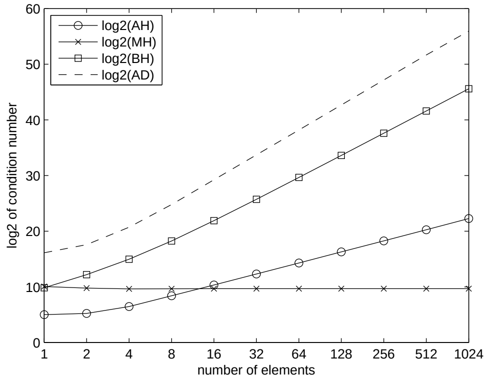


Figure 7.6: Plot of the condition number of Scaled mass matrix (M^H) in 1D as described in the table 7.1.

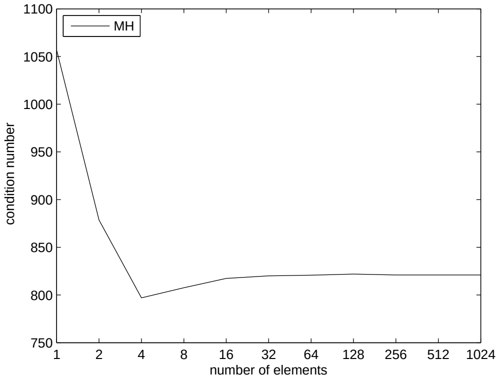


Figure 7.7: Plot of the condition number of Scaled A^H matrix in 1D as described in the table 7.1.

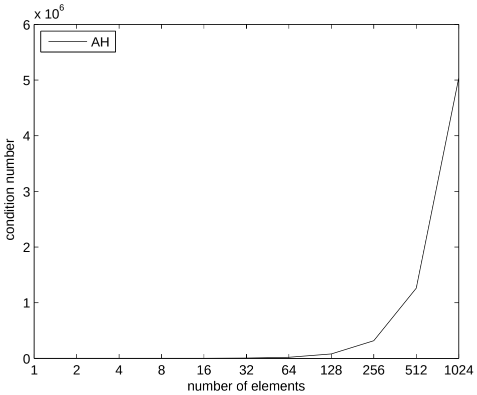


Figure 7.8: Plot of the condition number of Scaled A^D matrix in 1D as described in the table 7.1.

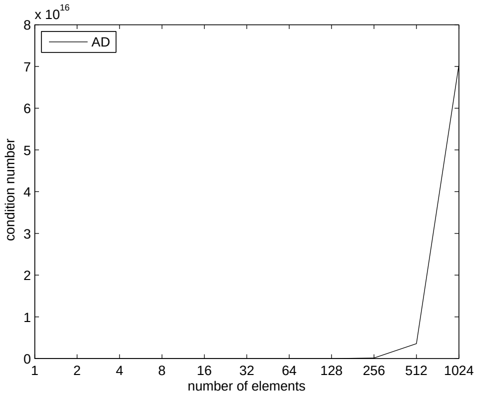


Figure 7.9: Plot of the condition number of Scaled B^H matrix in 1D as described in the table 7.1.

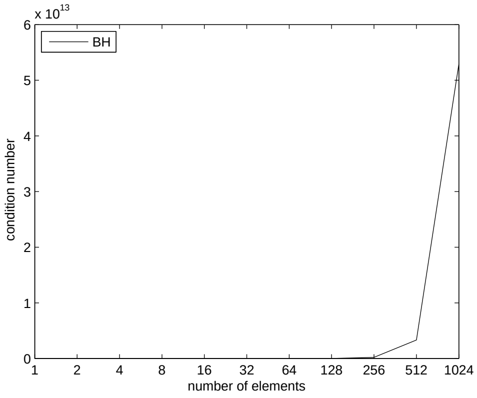


Table 7.2: Condition numbers (κ_1) for unscaled matrices from a one dimensional grid.

number of elements	M^H cond number	$A^H + M^H$ cond number	$A^D + M^H$ cond number	$B + M^H$ cond number
$1 = 2^0$	$1.\ 0567 \cdot 10^{3}$	$3.\ 1985 \cdot 10^{1}$	$7.\ 1703 \cdot 10^{4}$	$9.\ 1266 \cdot 10^{2}$
$2 = 2^1$	$3.\ 4735 \cdot 10^{3}$	$1.\ 4626 \cdot 10^{2}$	$6.\ 5010 \cdot 10^{5}$	$1.\ 2395 \cdot 10^{4}$
$4 = 2^2$	$1.\ 2588 \cdot 10^{4}$	$6.\ 3619 \cdot 10^{2}$	$1.\ 6921 \cdot 10^{7}$	$1.\ 88138 \cdot 10^{5}$
$8 = 2^3$	$5.\ 1021 \cdot 10^{4}$	$2.\ 6859 \cdot 10^{3}$	$6.\ 0728 \cdot 10^{8}$	$2.\ 8449 \cdot 10^{6}$
$16 = 2^4$	$2.\ 0660 \cdot 10^{5}$	$1.\ 0969 \cdot 10^{4}$	$2.\ 4487 \cdot 10^{10}$	$4.\ 3518 \cdot 10^{7}$
$32 = 2^5$	$8.\ 2922 \cdot 10^{5}$	$4.\ 4140 \cdot 10^{4}$	$1.\ 0434 \cdot 10^{12}$	$6.\ 7675 \cdot 10^{8}$
$64 = 2^6$	$3.\ 3198 \cdot 10^{7}$	$1.\ 7684 \cdot 10^{5}$	$4.\ 5805 \cdot 10^{13}$	$1.\ 0656 \cdot 10^{10}$
$128 = 2^7$	$1.\ 3282 \cdot 10^{8}$	$7.\ 0769 \cdot 10^{5}$	$2.\ 0494 \cdot 10^{15}$	$1.\ 6905 \cdot 10^{11}$
$256 = 2^8$	$5.\ 3132 \cdot 10^{8}$	$2.\ 8311 \cdot 10^{6}$	$1.\ 0096 \cdot 10^{17}$	$2.\ 6930 \cdot 10^{12}$
$512 = 2^9$	$2.\ 1253 \cdot 10^{9}$	$1.\ 1325 \cdot 10^{7}$	$3.\ 1641 \cdot 10^{18}$	$4.\ 2990 \cdot 10^{13}$
$1024 = 2^{10}$	$8.\ 5013 \cdot 10^{9}$	$4.\ 5298 \cdot 10^{7}$	$5.\ 9802 \cdot 10^{20}$	$6.\ 8696 \cdot 10^{14}$

Figure 7.10: Plot of the $2-\log$ condition number of unscaled matrix in 1D as described in the table 7.2.

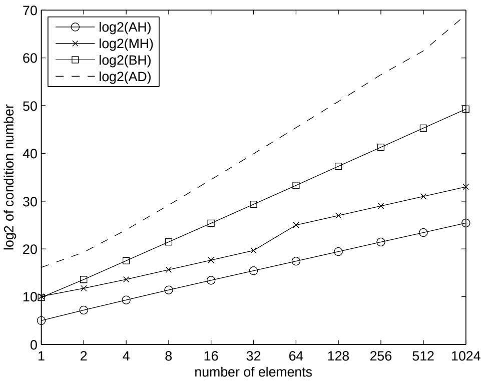
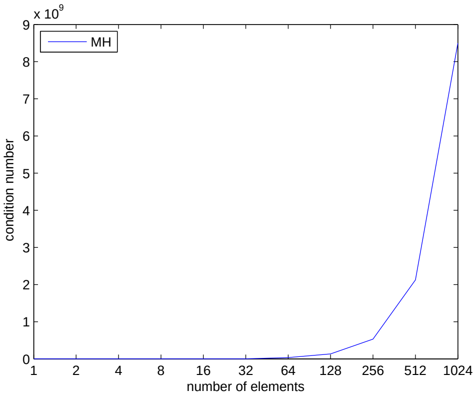


Figure 7.11: Plot of the condition number of unscaled mass matrix in 1D as described in the table 7.2.



7.3 Hermite elements 2D

For a two dimensional finite element mesh, the questions how to translate the scaling are not trivial. In this section, two different scalings are tested.

The set of basis functions for a two-dimensional Hermite element is a tensor product between the basis function sets of two 1D elements. These one dimensional elements are the Hermite element running along one of the edges, and of a Hermite element running along one of the other edges normal to the first edge, I will call them L_1, L_2. Each basis functions of a Hermite-2D element on a rectangle is a multiplication between one of basis functions L_1, and one of the basis functions of L_2. The set $\{\phi_i(x,y)\}_{2D} = \{\phi_i(x)\phi_j(y)\}_{i,j=1,2,3,4}$. There is a total of 16 basis functions for the 2-D Hermite element.

Figure 7.12: Plot of the condition number of unscaled A^H matrix in 1D as described in the table 7.2.

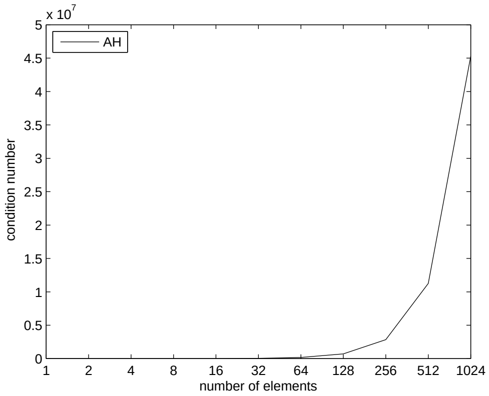


Figure 7.13: Plot of the condition number of unscaled A^D matrix in 1D as described in the table 7.2.

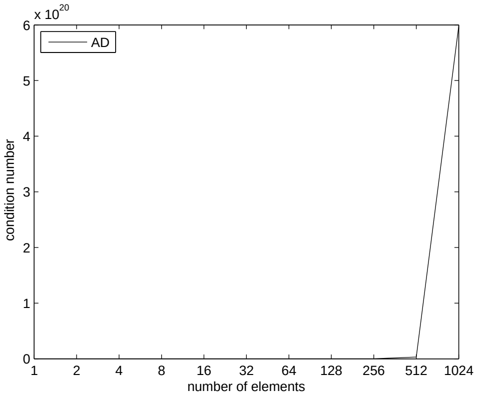
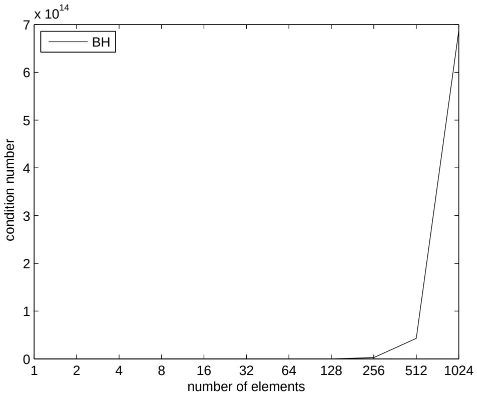


Figure 7.14: Plot of the condition number of unscaled B^H matrix in 1D as described in the table 7.2.



In 2D it is therefore natural to assume that the scaling also should be multiplied. For simplicity only the four basis functions evaluating the reference square in the point $(0,0)$ will be provided when describing the scaling of the basis functions. The unscaled basis functions that evaluate the reference element are calculated so that they are zero in every evaluation point away from $(0,0)$. In the point $(0,0)$ they will conform to the following conditions;

$$\begin{aligned}
\frac{\partial\phi_0(0,0)}{\partial x} &= \frac{\partial\phi_0(0,0)}{\partial y} = \frac{\partial^2\phi_0(0,0)}{\partial x\partial y} = 0 \\
\phi_1(0,0) &= \frac{\partial\phi_1(0,0)}{\partial y} = \frac{\partial^2\phi_1(0,0)}{\partial x\partial y} = 0 \\
\phi_2(0,0) &= \frac{\partial\phi_2(0,0)}{\partial x} = \frac{\partial^2\phi_2(0,0)}{\partial x\partial y} = 0 \\
\phi_3(0,0) &= \frac{\partial\phi_3(0,0)}{\partial x} = \frac{\partial\phi_3(0,0)}{\partial y} = 0 \\
\phi_0(0,0) &= \frac{\partial\phi_1(0,0)}{\partial x} = \frac{\partial\phi_2(0,0)}{\partial y} = \frac{\partial^2\phi_3(0,0)}{\partial x\partial y} = 1.
\end{aligned}$$

A first suggestion for scaling of these basis functions will look like;

$$\begin{aligned}
\phi_0(0,0) &= 1 \\
\frac{\partial\phi_1^N(0,0)}{\partial x} &= N \\
\frac{\partial\phi_2^N(0,0)}{\partial y} &= N \\
\frac{\partial^2\phi_3^N(0,0)}{\partial x\partial y} &= N^2.
\end{aligned}$$

This is the most obvious choice, since this is the direct tensor product of the scaled one dimensional elements from previous sections. Running the simulation again but with these modifications gave the condition numbers of tables 7.3 and 7.4 (figures: 7.15, 7.16, 7.17, 7.18, 7.19 and 7.20). M^H behaves good. A^H does not develop as it should, ideally it should scale as $O(4^N)$. A^D scales as it should, but it starts out with a fairly big condition number. B_1 scales as it should, but it starts out with a fairly big condition number. B_2 scales as it should, and seems to be the best choice for a biharmonic matrix.

Table 7.3: Condition numbers (κ_1) for matrices representing a 2- dimensional grid with fully scaled basis functions.

number of elements	M^H cond number	$A^H + M^H$ cond number	$A^D + M^H$ cond number
1 = $(2^0)^2$	1. 1165 $\cdot 10^{6}$	1. 6961 $\cdot 10^{4}$	1. 7368 $\cdot 10^{9}$
4 = $(2^1)^2$	7. 7197 $\cdot 10^{5}$	2. 1925 $\cdot 10^{4}$	5. 3687 $\cdot 10^{9}$
16 = $(2^2)^2$	6. 3522 $\cdot 10^{5}$	1. 7823 $\cdot 10^{4}$	8. 8709 $\cdot 10^{10}$
64 = $(2^3)^2$	6. 5215 $\cdot 10^{5}$	1. 7296 $\cdot 10^{4}$	5. 7546 $\cdot 10^{12}$
256 = $(2^4)^2$	6. 6803 $\cdot 10^{5}$	1. 7616 $\cdot 10^{4}$	6. 9134 $\cdot 10^{14}$
1024 = $(2^5)^2$	6. 7252 $\cdot 10^{5}$	1. 7722 $\cdot 10^{4}$	9. 5156 $\cdot 10^{15}$

Table 7.4: Condition numbers (κ_1) for the two types biharmonic matrices representing a 2- dimensional grid with fully scaled basis functions.

number of elements	$B_2 + M^H$ cond number	$B_1 + M^H$ cond number
1 = $(2^0)^2$	2. 4324 $\cdot 10^{3}$	2. 7417 $\cdot 10^{5}$
4 = $(2^1)^2$	1. 1152 $\cdot 10^{4}$	1. 2095 $\cdot 10^{6}$
16 = $(2^2)^2$	7. 4394 $\cdot 10^{4}$	1. 6625 $\cdot 10^{6}$
64 = $(2^3)^2$	6. 6903 $\cdot 10^{5}$	3. 8788 $\cdot 10^{6}$
256 = $(2^4)^2$	7. 9697 $\cdot 10^{6}$	2. 0004 $\cdot 10^{7}$
1024 = $(2^5)^2$	1. 1081 $\cdot 10^{7}$	1. 9406 $\cdot 10^{8}$

Figure 7.15: Plot of the $2 - \log$ condition number of scaled matrices in 2D as described in the tables 7.3 and 7.4.

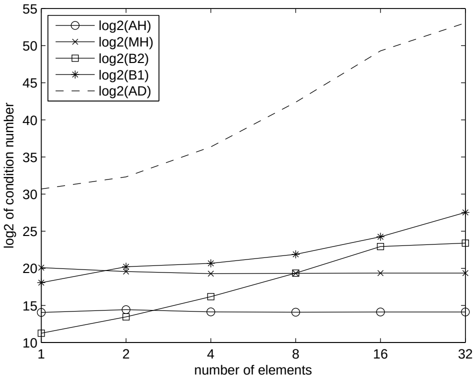


Figure 7.16: Plot of the condition number of Scaled mass matrix in 2D as described in the table 7.3.

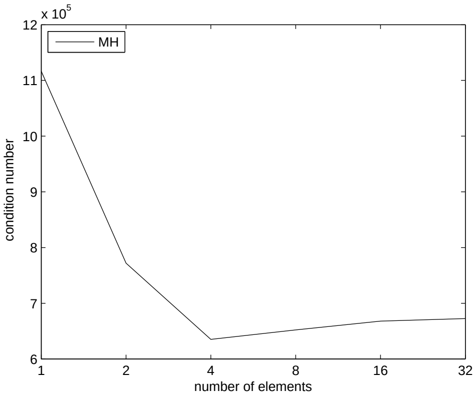


Figure 7.17: Plot of the condition number of Scaled A^H matrix in 2D as described in the table 7.3.

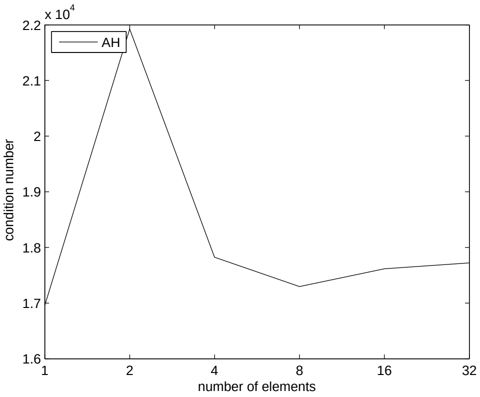


Figure 7.18: Plot of the condition number of Scaled A^D matrix in 2D as described in the table 7.3.

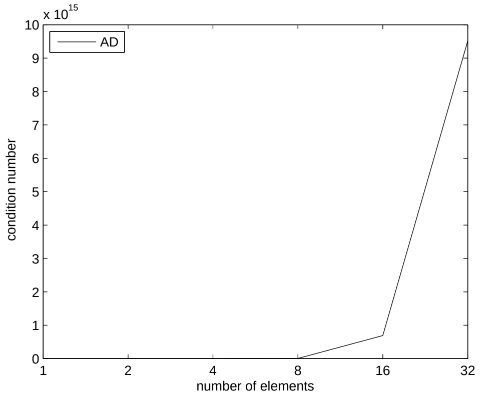


Figure 7.19: Plot of the condition number of Scaled B_2^H matrix in 2D as described in the table 7.4.

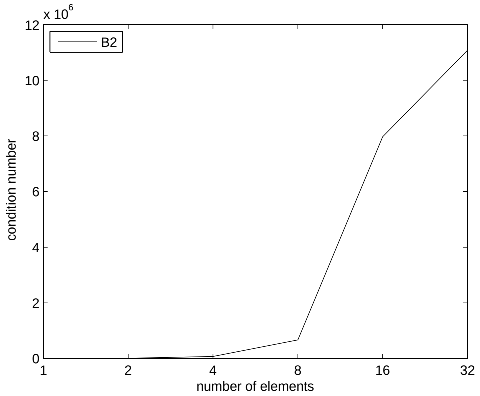
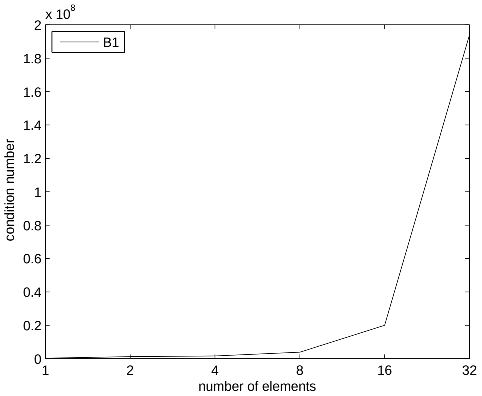


Figure 7.20: Plot of the condition number of Scaled B_1^H matrix in 2D as described in the table 7.4.



This scaling is not the only scaling possible. To test the hypothesis that the full scaling model is the best option, the "Reduced scaling" will be introduced. The Reduced scaling, is a weaker form of scaling. It is different than the full scaling in how it treats the basis functions ϕ_{4i}. These are the basis functions that conforms the suggested solution with both the derivative in x and y direction. The reduced scaling is defined by the following conditions:

$$\begin{aligned}
\phi_0(0,0) &= 1 \\
\frac{\partial \phi_1^N(0,0)}{\partial x} &= N \\
\frac{\partial \phi_2^N(0,0)}{\partial y} &= N \\
\frac{\partial^2 \phi_3^N(0,0)}{\partial x \partial y} &= N.
\end{aligned}$$

Running simulations with the reduced scaling gives the condition numbers of the tables 7.5 and 7.6 (figures: 7.21, 7.22, 7.23, 7.24, 7.25 and 7.26). The A^H is the only condition number that benefits from the reduced scaling over the full scaling.

Table 7.5: Condition numbers (κ_1) for matrices representing a 2- dimensional grid with reduced scaled basis functions.

number of elements	M^H cond number	$A^H + M^H$ cond number	$A^D + M^H$ cond number
1 = $(2^0)^2$	1. 1165 $\cdot 10^{6}$	1. 6961 $\cdot 10^{4}$	1. 7368 $\cdot 10^{9}$
4 = $(2^1)^2$	3. 0413 $\cdot 10^{6}$	8. 6383 $\cdot 10^{4}$	1. 5488 $\cdot 10^{10}$
16 = $(2^2)^2$	9. 9365 $\cdot 10^{6}$	2. 7881 $\cdot 10^{5}$	5. 6817 $\cdot 10^{11}$
64 = $(2^3)^2$	4. 0717 $\cdot 10^{7}$	1. 0799 $\cdot 10^{6}$	4. 7056 $\cdot 10^{13}$
256 = $(2^4)^2$	1. 6678 $\cdot 10^{8}$	4. 3982 $\cdot 10^{6}$	9. 6000 $\cdot 10^{15}$
1024 = $(2^5)^2$	6. 7154 $\cdot 10^{8}$	1. 7698 $\cdot 10^{7}$	3. 5710 $\cdot 10^{16}$

Table 7.6: Condition numbers (κ_1) for the two types biharmonic matrices representing a 2- dimensional grid with reduced scaled basis functions.

number of elements	$B_2 + M^H$ cond number	$B_1 + M^H$ cond number
1 = $(2^0)^2$	$2.\ 4324 \cdot 10^3$	$2.\ 7417 \cdot 10^5$
4 = $(2^1)^2$	$1.\ 3862 \cdot 10^4$	$3.\ 0276 \cdot 10^6$
16 = $(2^2)^2$	$7.\ 4437 \cdot 10^4$	$8.\ 1370 \cdot 10^6$
64 = $(2^3)^2$	$6.\ 6903 \cdot 10^5$	$2.\ 6019 \cdot 10^7$
256 = $(2^4)^2$	$7.\ 9697 \cdot 10^6$	$2.\ 8929 \cdot 10^8$
1024 = $(2^5)^2$	$1.\ 1081 \cdot 10^7$	$3.\ 4510 \cdot 10^9$

Figure 7.21: Plot of the $2 - \log$ condition number of reduced scaled matrices in 2D.

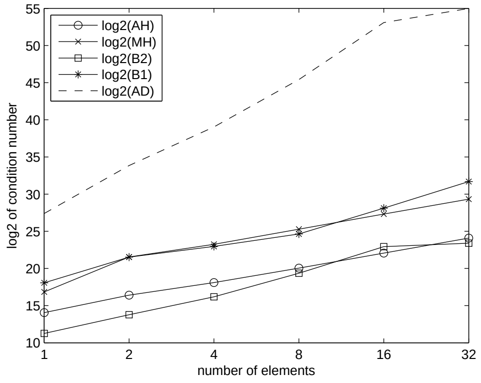


Figure 7.22: Plot of the condition number of reduced scaled mass matrix in 2D as described in the tables 7.5 and 7.6.

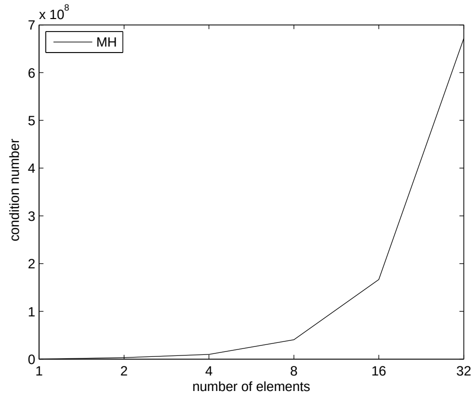


Figure 7.23: Plot of the condition number of reduced scaled A^H matrix in 2D as described in the table 7.5.

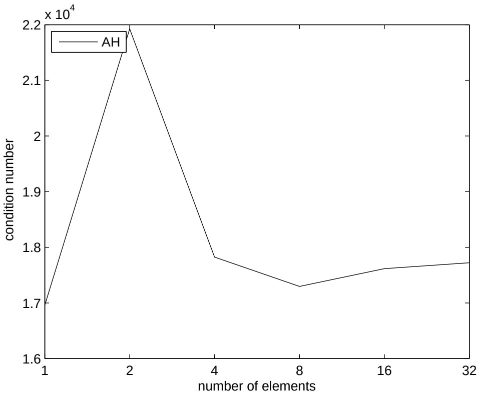


Figure 7.24: Plot of the condition number of reduced scaled A^D matrix in 2D as described in the table 7.5.

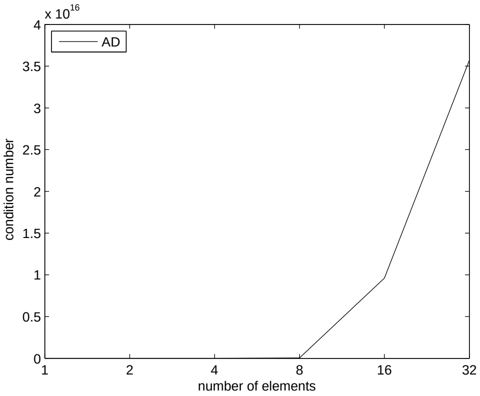


Figure 7.25: Plot of the condition number of Scaled B_2^H matrix in 2D as described in the table 7.6.

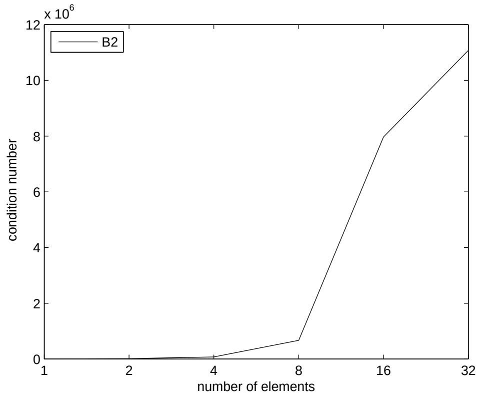
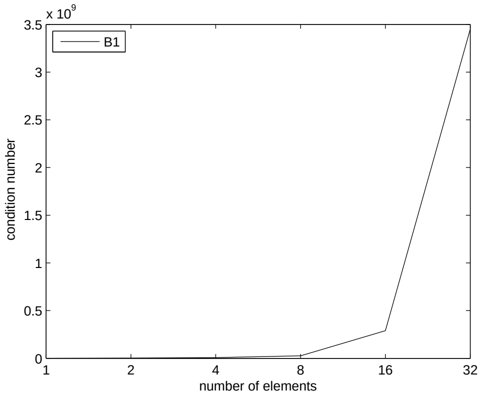


Figure 7.26: Plot of the condition number of reduced scaled B_1^H matrix in 2D as described in the table 7.6.



Chapter 8

Solving in 1D

8.1 Solving as a linear PDE system

8.1.1 Representation from $H^2(\Omega) \times L^2(\Omega)$.

As described, there are two competing weak representation (4.4) and (4.5). This section will look at the $H^2 \times L^2$ representation. We now select $v_1, \epsilon \in H^2(\Omega)$ and $v_2, k \in L^2(\Omega)$. With this choice we use the equation (4.5) as our weak form (bilinear). The finite element to use will be the Hermite elements for v_1, ϵ , and the Lagrange elements for v_2, k. In order to increase the accuracy and to keep the evaluations stable the choice to use the Lagrange-2 elements was made. The reasoning behind this choice is that if a line is divided into N different elements, then Hermite elements will give $2 \cdot (N+1)$ evaluations, while a Lagrange-2 will give $2 \cdot N + 1$ evaluations. So combining this knowledge, the matrices that will be tested are:

$$Co_D = \begin{bmatrix} M^H & S^D \\ (S^D)^T & M^L \end{bmatrix}$$

$$PrCo = \begin{bmatrix} B + M^H & 0 \\ 0 & M^L \end{bmatrix}^{-1} \begin{bmatrix} M^H & S^D \\ (S^D)^T & M^L \end{bmatrix}.$$

The matrix $PrCo$ is not symmetric and will have negative eigenvalues. The definition of the condition number used is $\kappa(PrCo) = \frac{\max_i\{|\lambda_i|\}}{\min_i\{|\lambda_i|\}}$. With this in mind, the key matrices of the system is in table 8.1. The preconditioned system in 8.1 does not scale as well as it should, and therefore the implementation does not work properly. There is some minor issues with the $S^D + M^S$ for small meshes, but it seems to work out.

Table 8.1: Condition numbers for scaled matrices from a one dimensional grid

number of	$\kappa_3(S^D + M^S)$	$\kappa_1(Co_D)$	$\kappa(PrCo)$
elements	condition number	condition number	condition number
1 = 2^0	371. 19	7. 2404 $\cdot 10^1$	1. 000 $\cdot 10^0$
2 = 2^1	17. 117	1. 1163 $\cdot 10^2$	1. 3417 $\cdot 10^0$
4 = 2^2	17. 662	4. 4115 $\cdot 10^2$	3. 2533 $\cdot 10^0$
8 = 2^3	30. 475	1. 7138 $\cdot 10^3$	1. 5295 $\cdot 10^1$
16 = 2^4	57. 266	6. 6693 $\cdot 10^3$	6. 2036 $\cdot 10^1$
32 = 2^5	106. 13	2. 6186 $\cdot 10^4$	2. 4880 $\cdot 10^2$
64 = 2^6	199. 96	1. 0362 $\cdot 10^5$	9. 959 $\cdot 10^2$
128 = 2^7	385. 74	4. 1209 $\cdot 10^5$	no data
256 = 2^8	756. 62	1. 6434 $\cdot 10^6$	no data
512 = 2^9	1498. 2	6. 5635 $\cdot 10^6$	no data
1024= 2^{10}	2981. 2	2. 6233 $\cdot 10^7$	no data

8.1.2 Representations from $H^1(\Omega) \times H^1(\Omega)$.

As described, there are two competing weak representation (4.4), (4.5). This section will look at the $H^1 \times H^1$ representation.

$$\int_\Omega \begin{bmatrix} v_1\epsilon & \nabla k \cdot \nabla v_1 \\ \nabla v_2 \cdot \nabla\epsilon & v_2 k \end{bmatrix} d\,\mathrm{x} = \int_\Omega \begin{bmatrix} v_1 f_1 \\ v_2 f_2 \end{bmatrix} d\,\mathrm{x}$$

In this representation we choose $v_1, v_2, \epsilon, k \in H^1(\Omega)$, and solve the systems with the following system matrices.

$$Co_N = \begin{bmatrix} M^L & A^L \\ (A^L)^T & M^L \end{bmatrix}$$

$$PrCo_N = \begin{bmatrix} S^L + M^L & 0 \\ 0 & S^L + M^L \end{bmatrix}^{-1} \begin{bmatrix} M^L & S^L \\ (S^L)^T & M^L \end{bmatrix}$$

where N is the degrees of freedom allowed in the Lagrange FEM approximation. The eigenvalues of this system is given in the tables 8.2 and 8.3 (figures: 8.1, 8.2, 8.3, 8.4). All the results in the tables 8.2 and 8.3 does behave as expected and we can conclude that Co_2 and Co_3 are successful implementations.

Table 8.2: Condition numbers (κ_1) for Lagrange-type matrices from a one dimensional grid

number of elements	Co_2 condition number	$PrCo_2$ condition number
$1 = 2^0$	4.3333	1.1818
$2 = 2^1$	18.640	2.0026
$4 = 2^2$	73.041	1.8309
$8 = 2^3$	279.99	1.5931
$16 = 2^4$	1079.4	1.4500
$32 = 2^5$	4215	1.3845
$64 = 2^6$	16631	1.3374
$128 = 2^7$	66039	1.3041
$256 = 2^8$	263158	1.2807
$512 = 2^9$	1050614	1.2644
$1024= 2^{10}$	4198390	1.2529

Table 8.3: Condition numbers (κ_1) for Lagrange-type matrices from a one dimensional grid

number of elements	Co_3 condition number	$PrCo_3$ condition number
$1 = 2^0$	25.453	3.7338
$2 = 2^1$	96.610	2.9384
$4 = 2^2$	371.52	2.4709
$8 = 2^3$	1436.78	2.2112
$16 = 2^4$	5617.36	2.0722
$32 = 2^5$	22172.	1.9997
$64 = 2^6$	88048	1.9627
$128 = 2^7$	350875	1.9439
$256 = 2^8$	1400816	1.9344
$512 = 2^9$	5597851	1.9297
$1024= 2^{10}$	22380528	1.9273

Figure 8.1: Plot of the 2 − log condition number of Lagrange-2, Lagrange-3 and the preconditioned systems matrices in 1D as described in the tables 8.2 and 8.3

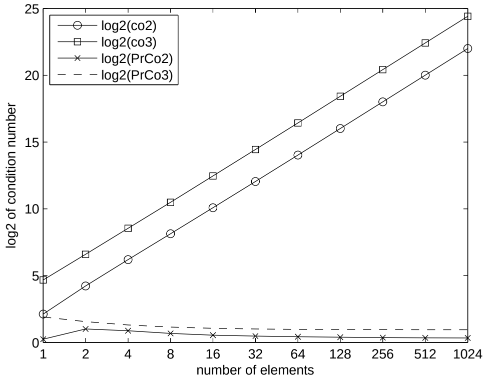


Figure 8.2: Plot of the condition number of $Co2$ Lagrange-2 system 1D as described in the table 8.2

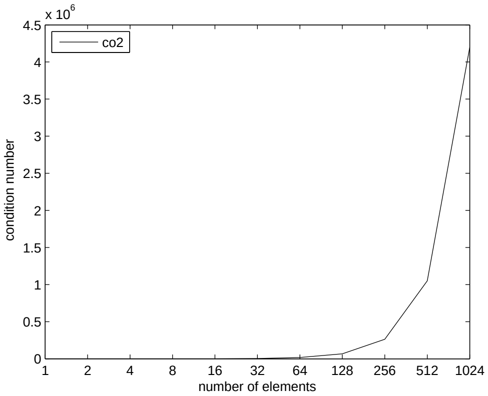


Figure 8.3: Plot of the condition number of $Co3$ Lagrange-3 system 1D as described in the table 8.3

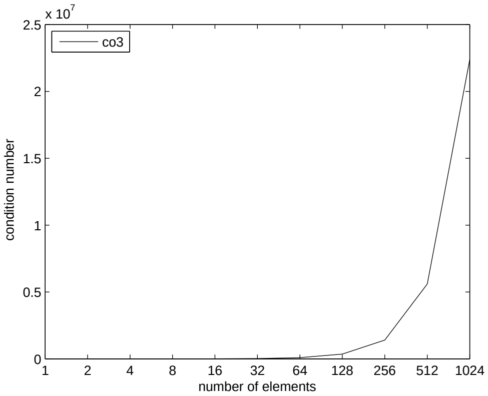
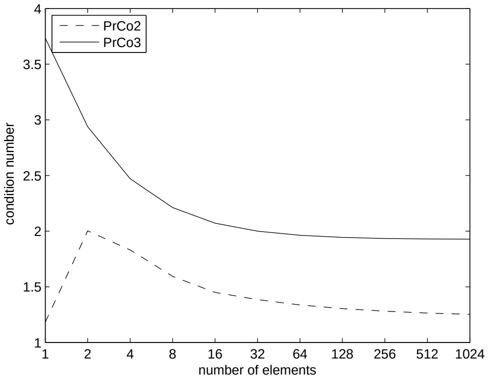


Figure 8.4: Plot of the condition number the preconditioned Lagrange-2 and the preconditioned Lagrange-3 systems matrices in 1D as described in the tables 8.2 and 8.3



8.1.3 Representations from $H^2(\Omega) \times H^1(\Omega)$.

Another possible choice for the function spaces of the tests and trail functions is to chose $v_2, k \in H^1(\Omega)$,$v_1, \epsilon \in H^2(\Omega)$. With this representation the following system is produced:

$$Co_S = \begin{bmatrix} M^H & S^S + M^S \\ (S^S + M^S)^T & M^L \end{bmatrix}$$
$$PrCo = \begin{bmatrix} B & 0 \\ 0 & M^L \end{bmatrix}^{-1} \begin{bmatrix} M^H & S^S + M^S \\ (S^S + M^S)^T & M^L \end{bmatrix}.$$

Before assembling the matrix Co_S it is important to note that choosing to use the Lagrange-1 representation will produce issues with the stiffness matrix A^S. Lets assume that we have the ordinary Hermite element. Then on a linear element $[a, b]$ the basis functions ϕ_0^τ, ϕ_1^τ, ϕ_2^τ, ϕ_3^τ are as previously defined. The Lagrange functions will be $L_0 = \frac{x}{a-b} + \frac{b}{b-a}$ and $L_1 = \frac{x}{b-a} + \frac{a}{a-b}$. We then get

$$\begin{aligned}
\int_b^a \phi_0^\tau(x)' L_0(x)' \,\mathrm{d}x &= \int_b^a \phi_0^\tau(x)' \frac{1}{a-b} \,\mathrm{d}x \\
&= \frac{1}{a-b}\left(\phi_0^\tau(a) - \phi_0^\tau(b)\right) = \frac{1}{a-b}(1-0) = \frac{1}{a-b} \\
\int_b^a \phi_0^\tau(x)' L_1(x)' \,\mathrm{d}x &= \int_b^a \phi_0^\tau(x)' \frac{1}{b-a} \,\mathrm{d}x \\
&= \frac{1}{b-a}\left(\phi_0^\tau(a) - \phi_0^\tau(b)\right) = \frac{1}{b-a}(1-0) = -\frac{1}{a-b} \\
\int_b^a \phi_1^\tau(x)' L_0(x)' \,\mathrm{d}x &= \int_b^a \phi_1^\tau(x)' \frac{1}{a-b} \,\mathrm{d}x \\
&= \frac{1}{a-b}\left(\phi_1^\tau(a) - \phi_1^\tau(b)\right) = \frac{1}{a-b}(0-0) = 0 \\
\int_b^a \phi_1^\tau(x)' L_1(x)' \,\mathrm{d}x &= \int_b^a \phi_1^\tau(x)' \frac{1}{b-a} \,\mathrm{d}x \\
&= \frac{1}{b-a}\left(\phi_1^\tau(a) - \phi_1^\tau(b)\right) = \frac{1}{b-a}(0-0) = 0.
\end{aligned}$$

With these kinds of calculations one can create the table 8.4 As showed in chapter 7 scaling the Hermite elements basis functions ϕ^τ with a factor $C(\tau) = \frac{1}{|a-b|} = \frac{1}{N}$ has proven to be beneficial with regards to the condition number of the FEM-matrix. The variant of 8.4 for scaled elements is the table 8.5 To avoid this problem the Lagrange-2 scheme is used and it creates the table 8.6. The matrix $PrCo_s$ in table 8.6 is very bad, we conclude that this implementation does not work.

Table 8.4: Values of inner products between Lagrange-1 and Hermite basis functions on a single linear element

$(\cdot,\cdot)_{L^2(\tau)}$	$\phi_0^\tau(x)'$	$\phi_1^\tau(x)'$	$\phi_2^\tau(x)'$	$\phi_3^\tau(x)'$
L_0'	$\frac{1}{a-b}$	0	$-\frac{1}{a-b}$	0
L_1'	$-\frac{1}{a-b}$	0	$\frac{1}{a-b}$	0

Table 8.5: Values of inner products between Lagrange-1 and scaled Hermite basis functions on a single linear element

$(\cdot,\cdot)_{L^2(\tau)}$	$\phi_0^\tau(x)'$	$\phi_1^\tau(x)'$	$\phi_2^\tau(x)'$	$\phi_3^\tau(x)'$
L_0'	1	0	-1	0
L_1'	-1	0	1	0

Table 8.6: Condition numbers (κ_3) for combined matrices using Lagrange-2

number of elements	$S^S + M^S$ condition number	Co_S condition number	$PrCo_S$ condition number
$1 = 2^0$	5. 8156	9. 1885 $\cdot 10^{2}$	1. 5119 $\cdot 10^{17}$
$2 = 2^1$	11. 321	4. 8547 $\cdot 10^{3}$	2. 9019 $\cdot 10^{19}$
$4 = 2^2$	13. 790	2. 0227 $\cdot 10^{4}$	2. 5807 $\cdot 10^{17}$
$8 = 2^3$	22. 521	7. 3769 $\cdot 10^{4}$	9. 8466 $\cdot 10^{16}$
$16 = 2^4$	42. 717	2. 7072 $\cdot 10^{5}$	2. 5269 $\cdot 10^{17}$
$32 = 2^5$	83. 158	1. 0307 $\cdot 10^{6}$	1. 2922 $\cdot 10^{17}$
$64 = 2^6$	163. 92	4. 0236 $\cdot 10^{6}$	1. 6143 $\cdot 10^{17}$
$128 = 2^7$	325. 38	1. 5911 $\cdot 10^{7}$	8. 4882 $\cdot 10^{16}$
$256 = 2^8$	648. 27	6. 3276 $\cdot 10^{7}$	8. 6149 $\cdot 10^{16}$
$512 = 2^9$	1294. 0	2. 5238 $\cdot 10^{8}$	9. 8993 $\cdot 10^{16}$
$1024 = 2^{10}$	2585. 5	1. 0081 $\cdot 10^{9}$	2. 6523 $\cdot 10^{19}$

Table 8.7: Condition numbers (κ_3) for combined matrices using Lagrange-2 without mass matrix

number of elements	S^S condition number	Co_{SH} condition number
1 = 2^0	2. 9565 $\cdot 10^{16}$	9. 1885 $\cdot 10^2$
2 = 2^1	17. 220	3. 8944 $\cdot 10^3$
4 = 2^2	14. 997	1. 5337 $\cdot 10^4$
8 = 2^3	22. 813	5. 8823 $\cdot 10^4$
16 = 2^4	42. 840	2. 2672 $\cdot 10^5$
32 = 2^5	83. 217	8. 8523 $\cdot 10^5$
64 = 2^6	163. 95	3. 4926 $\cdot 10^6$
128 = 2^7	325. 39	1. 3868 $\cdot 10^7$
256 = 2^8	648. 27	5. 5263 $\cdot 10^7$
512 = 2^9	1294. 0	2. 2063 $\cdot 10^8$
1024= 2^{10}	2585. 5	8 8166 $\cdot 10^8$

As one can clearly see this representation of the problem creates bad results, and should not be used. A modification of this representation can be:

$$\begin{aligned} Co_{SH} &= \begin{bmatrix} M^H & S^S \\ (S^S)^T & M^L \end{bmatrix} \\ PrCo_{SH} &= \begin{bmatrix} B + M^H & 0 \\ 0 & M^L \end{bmatrix}^{-1} \begin{bmatrix} M^H & S^S \\ (S^S)^T & M^L \end{bmatrix} \\ PrCo_{AH} &= \begin{bmatrix} S^H + M^H & 0 \\ 0 & S^L + M^L \end{bmatrix}^{-1} \begin{bmatrix} M^H & S^S \\ (S^S)^T & M^L \end{bmatrix}, \end{aligned}$$

where

$$S^S_{i,j} = \int \nabla L_i \cdot \nabla H_j \, \mathrm{d}x.$$

It produces the tables 8.7 and 8.8 (figures: 8.5, 8.6, 8.7). The condition number of S^S is very high, this can be expected. Derivatives without a mass matrix is usually badly conditioned when there is only one element in the mesh grid. The condition numbers of Co_{SH} are good, but none of the two tested preconditioner does not work. This might be because we used a H^2 conforming element to approximate a H^1 function.

Table 8.8: Condition numbers (κ_3) for preconditioned combined matrices using Lagrange-2 without mass matrix

number of elements	$PrCo_{SH}$ condition number	$PrCo_{AH}$ condition number
$1 = 2^0$	1. 5206 $\cdot 10^{5}$	7. 1572 $\cdot 10^{2}$
$2 = 2^1$	1. 3918 $\cdot 10^{6}$	1. 9253 $\cdot 10^{3}$
$4 = 2^2$	2. 6644 $\cdot 10^{7}$	5. 6273 $\cdot 10^{3}$
$8 = 2^3$	5. 5366 $\cdot 10^{8}$	1. 9746 $\cdot 10^{4}$
$16 = 2^4$	1. 1995 $\cdot 10^{10}$	7. 3565 $\cdot 10^{4}$
$32 = 2^5$	2. 6545 $\cdot 10^{11}$	2. 8346 $\cdot 10^{5}$
$64 = 2^6$	5. 9378 $\cdot 10^{12}$	1. 1122 $\cdot 10^{6}$
$128 = 2^7$	1. 3383 $\cdot 10^{14}$	4. 4058 $\cdot 10^{6}$
$256 = 2^8$	3. 0841 $\cdot 10^{15}$	1. 7537 $\cdot 10^{7}$
$512 = 2^9$	5. 1202 $\cdot 10^{16}$	6. 9974 $\cdot 10^{7}$
$1024= 2^{10}$	2. 3051 $\cdot 10^{18}$	2. 7955 $\cdot 10^{8}$

Figure 8.5: Plot of the $2-\log$ condition number of the preconditioned Lagrange-3-Hermite systems matrices in 1D as described in table 8.8

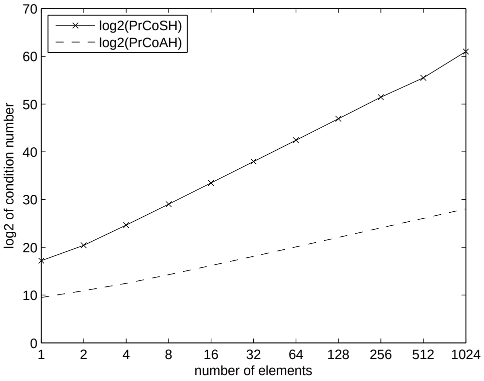


Figure 8.6: Plot of the condition number of the preconditioned Lagrange-3-Hermite systems matrices in 1D as described in table of $PrCo_{SH}$ in 8.8

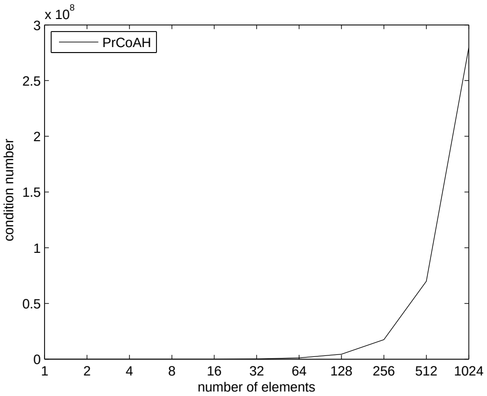
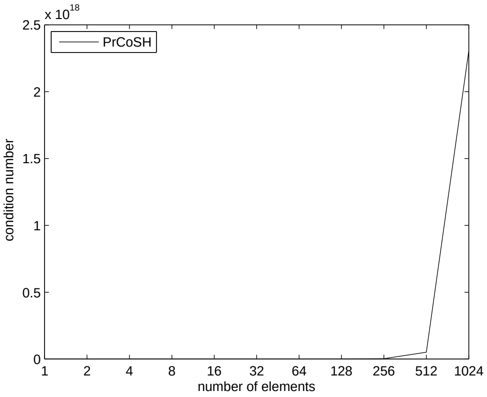


Figure 8.7: Plot of the condition number of the preconditioned Lagrange-3-Hermite systems matrices in 1D as described in table of $PrCo_{SH}$ in 8.8



8.2 Solving as a Biharmonic equation

Assuming $(\epsilon, k) \in H^2(\Omega) \times L^2(\Omega)$

$$\begin{cases} -\Delta\epsilon + k = f_1 & |\Delta \\ \epsilon - \Delta k = f_2 \end{cases}$$

$$\begin{cases} -\Delta\Delta\epsilon + \Delta k = \Delta f_1 & | + \text{(II)} \\ \epsilon - \Delta k = f_2 \end{cases}$$

$$\begin{aligned} -\Delta\Delta\epsilon + \Delta k + \epsilon - \Delta k =& \Delta f_1 + f_2 \\ -\Delta\Delta\epsilon + \epsilon =& \Delta f_1 + f_2 \\ \int_\Omega -v\Delta\Delta\epsilon + v\epsilon \, \mathrm{d}x =& \int_\Omega v\Delta f_1 + v f_2 \, \mathrm{d}x \\ \int_\Omega -\Delta v\Delta\epsilon + v\epsilon \, \mathrm{d}x =& \int_\Omega v\Delta f_1 + v f_2 \, \mathrm{d}x + \text{boundery terms} \end{aligned}$$

the discrete representation of this problem is

$$[-B^H + M^H].$$

The table for the condition numbers was described in the table 7.1. It is not preconditioned, but it scales as it should.

8.2.1 Representation from $H^2(\Omega) \times L^2(\Omega)$. Discontinuous Lagrange

In this section the Problem is implemented with discontinuous Galerkin. This section will look at the $H^2 \times L^2$ representation. We now select $v_1, \epsilon \in H^2(\Omega)$ and $v_2, k \in L^2(\Omega)$. With this choice we use the equation (4.5) as our weak form (bilinear). The finite element to use will for v_1, ϵ be the Hermite elements, while for v_2, k will be the Discontinuous Lagrange elements.

$$\begin{aligned} B_{i,j} &= \int \Delta H_i \Delta H_j \, \mathrm{d}x \\ S^{dD}_{i,j} &= \int -dL_i \Delta H_j \, \mathrm{d}x \\ M^{dL}_{i,j} &= \int dL_i dL_j \, \mathrm{d}x \\ M^H_{i,j} &= \int H_i H_j \, \mathrm{d}x \end{aligned}$$

Table 8.9: Condition numbers (κ_2) for preconditioned scaled combined matrices using discontinuous Lagrange-2

number of elements	$PrCo_D$ condition number	Co_{DH} condition number
1 = 2^0	2.6249	72.404
2 = 2^1	2.6279	78.838
4 = 2^2	2.6279	123.82
8 = 2^3	2.6280	290.23
16 = 2^4	2.6280	899.37
32 = 2^5	2.6280	3206.4
64 = 2^6	2.6280	1.2166e+04
128 = 2^7	2.6280	4.7465e+04

$$
\begin{aligned}
Co_{DH} &= \begin{bmatrix} M^H & S^{dD} \\ (S^{dD})^T & M^L \end{bmatrix} \\
PrCo_D &= \begin{bmatrix} B + M^H & 0 \\ 0 & M^L \end{bmatrix}^{-1} \begin{bmatrix} M^H & S^{dD} \\ (S^{dD})^T & M^L \end{bmatrix}.
\end{aligned}
$$

The matrix $PrCo_D$ is not symmetric and will have negative eigenvalues. The definition of the condition number used is $\kappa(PrCo_D) = \frac{\max_i\{|\lambda_i|\}}{\min_i\{|\lambda_i|\}}$. The "qz" algorithm in octave is used.

Table 8.10 and table 8.9 shows very good results. The condition numbers unpreconditioned system shows the importance of the scaling for numerical stability. The condition numbers preconditioned system shows condition numbers that conform with the expectations from chapter 5 ($\lim(\kappa(P^{-1}B)) \rightarrow$ 2.66946). It appears we have found a good implementation for the simplified k-ϵ model with a $H^2 \times L^2$ representation.

Table 8.10: Condition numbers (κ_2) for preconditioned unscaled combined matrices using discontinuous Lagrange-2

number of elements	$PrCo_D$ condition number	Co_{DH} condition number
$1 = 2^0$	2.6249	72.404
$2 = 2^1$	2.6279	213.85
$4 = 2^2$	2.6279	731.68
$8 = 2^3$	2.6280	2660.7
$16 = 2^4$	2.6280	1.0055e+04
$32 = 2^5$	2.6280	3.8962e+04
$64 = 2^6$	2.6280	1.5324e+05
$128 = 2^7$	2.6280	6.0764e+05

Chapter 9

System in 2 Dimensions

9.1 As a single equation in 2D

$$\begin{aligned} \epsilon - \Delta k =& f_1(x) \quad on\, \Omega \qquad (9.1) \\ -\Delta\epsilon + k =& f_2(x) \quad on\, \Omega. \qquad (9.2) \end{aligned}$$

By derivating one of the equations, we reduce this system into a system of one unknown.

$$\begin{aligned} -\Delta\epsilon(x) + k(x) =& f_2(x) \quad on\, \Omega \\ -\sum_{i=1}^{n} \frac{\partial^2 \epsilon(x)}{\partial x_i} + k(x) =& f_2(x) \quad on\, \Omega \\ \sum_{j=1}^{n} \frac{\partial^2}{\partial x_j^2}\Big(\sum_{i=1}^{n} \frac{\partial^2 \epsilon(x)}{\partial x_i^2} + k(x)\Big) =& \sum_{j=1}^{n} \frac{\partial^2}{\partial x_j^2} f_2(x) \quad on\, \Omega \\ \sum_{j=1}^{n}\sum_{i=1}^{n} \frac{\partial^4 \epsilon(x)}{\partial x_i^2 \partial x_j^2} + \sum_{i=1}^{n} \frac{\partial^2}{\partial x_i^2} k(x) =& \sum_{i=1}^{n} \frac{\partial^2}{\partial x_i^2} f_2(x) \quad on\, \Omega. \end{aligned}$$

This can be added to (9.2)

$$\begin{aligned} \sum_{j=1}^{n}\sum_{i=1}^{n} \frac{\partial^4 \epsilon(x)}{\partial x_i^2 \partial x_j^2} + \sum_{i=1}^{n} \frac{\partial^2}{\partial x_i^2} k(x) - \sum_{i=1}^{n} \frac{\partial^2}{\partial x_i^2} k(x) + \epsilon(x) =& \sum_{i=1}^{n} \frac{\partial^2}{\partial x_i^2} f_2(x) + f_1(x) \quad on\, \Omega \\ \sum_{j=1}^{n}\sum_{i=1}^{n} \frac{\partial^4 \epsilon(x)}{\partial x_i^2 \partial x_j^2} + \epsilon(x) =& \sum_{i=1}^{n} \frac{\partial^2}{\partial x_i^2} f_2(x) + f_1(x) \quad on\, \Omega. \end{aligned}$$

We multiply with a test function

$$\sum_{j=1}^{n}\sum_{i=1}^{n}\frac{\partial^4\epsilon(x)}{\partial x_i^2\partial x_j^2}v+\epsilon(x)v=\sum_{i=1}^{n}\frac{\partial^2}{\partial x_i^2}f_2(x)v+f_1(x)v \quad on\,\Omega$$

$$\int_\Omega\sum_{j=1}^{n}\sum_{i=1}^{n}\frac{\partial^4\epsilon(x)}{\partial x_i^2\partial x_j^2}v+\epsilon(x)v\,\mathrm{d}x=\int_\Omega\sum_{i=1}^{n}\frac{\partial^2}{\partial x_i^2}f_2(x)v+f_1(x)v\,\mathrm{d}x.$$

Integrate by parts to obtain a weak solution,

$$\int_{\partial\Omega}\sum_{j=1}^{n}\sum_{i=1}^{n}\frac{\partial^3\epsilon(x)}{\partial x_i^2\partial x_j}vn_j-\sum_{j=1}^{n}\sum_{i=1}^{n}\frac{\partial^2\epsilon(x)}{\partial x_i^2}\frac{\partial v}{\partial x_j}n_j\,\mathrm{d}S(x)-$$

$$\int_\Omega\sum_{j=1}^{n}\sum_{i=1}^{n}\frac{\partial^2\epsilon(x)}{\partial x_i^2}\frac{\partial^2 v}{\partial x_j^2}+\epsilon(x)v\,\mathrm{d}x=\int_\Omega\sum_{i=1}^{n}\frac{\partial^2}{\partial x_i^2}f_2(x)v+f_1(x)v\,\mathrm{d}x.$$

This can be written as

$$\int_\Omega\Delta\epsilon(x)\Delta v+\epsilon(x)v\,\mathrm{d}x=\int_\Omega\Delta f_2(x)v+f_1(x)v\,\mathrm{d}x+\text{boundery term.}$$

A different representation is

$$\int_\Omega\sum_{j=1}^{n}\sum_{i=1}^{n}\frac{\partial^4\epsilon(x)}{\partial x_i^2\partial x_j^2}v+\epsilon(x)v\,\mathrm{d}x=\int_\Omega\sum_{i=1}^{n}\frac{\partial^2}{\partial x_i^2}f_2(x)v+f_1(x)v\,\mathrm{d}x$$

$$\int_{\partial\Omega}\sum_{j=1}^{n}\sum_{i=1}^{n}\frac{\partial^3\epsilon(x)}{\partial x_i^2\partial x_j}vn_j-\sum_{j=1}^{n}\sum_{i=1}^{n}\frac{\partial^2\epsilon(x)}{\partial x_i\partial x_j}\frac{\partial v}{\partial x_j}n_i\,\mathrm{d}S(x)-$$

$$\int_\Omega\sum_{j=1}^{n}\sum_{i=1}^{n}\frac{\partial^2\epsilon(x)}{\partial x_i\partial x_j}\frac{\partial^2 v}{\partial x_j\partial x_i}+\epsilon(x)v\,\mathrm{d}x=\int_\Omega\sum_{i=1}^{n}\frac{\partial^2}{\partial x_i^2}f_2(x)v+f_1(x)v\,\mathrm{d}x.$$

This can be written as

$$\int_\Omega\mathbf{D}^2(\epsilon):\mathbf{D}^2(v)+\epsilon v\,\mathrm{d}x=\int_\Omega\Delta f_2v+f_1v\,\mathrm{d}x+\text{boundery term.}$$

$\mathbf{D}^2(u)$ is the Hessian matrix. These weak formulations are equal in one dimension. By setting $n=2$ the left hand side of the equations looks like this:

$$\int_\Omega u_{xx}v_{xx}+u_{xx}v_{yy}+u_{yy}v_{xx}+u_{yy}v_{yy}+uv\,\mathrm{d}x=<u,v>_{(B)} \tag{9.3}$$

$$\int_\Omega u_{xx}v_{xx}+2u_{xy}v_{xy}+u_{yy}v_{yy}+uv\,\mathrm{d}x=<u,v>_{(H)}\,. \tag{9.4}$$

uv is removed to obtain a H_0^2 inner product ((9.6), (9.5))

$$\int_\Omega u_{xx}v_{xx}+u_{xx}v_{yy}+u_{yy}v_{xx}+u_{yy}v_{yy}\,\mathrm{d}x=<u,v>_{(1)} \tag{9.5}$$

$$\int_\Omega u_{xx}v_{xx}+2u_{xy}v_{xy}+u_{yy}v_{yy}\,\mathrm{d}x=<u,v>_{(2)}\,. \tag{9.6}$$

Table 9.1: functions in the null space of the functionals$< \cdot,\cdot >_{(1)}$,$< \cdot,\cdot >_{(2)}$

	$< u,u >_{(1)}$	$< u,u >_{(2)}$
$u = C$	0	0
$u = Cx$	0	0
$u = Cy$	0	0
$u = Cxy$	0	$2C^2\|\Omega\|$

We identify which functions in H^2 that are part of the null space of the inner product. They are in table 9.1. The null spaces of these inner products are different. To simplify further investigation of the stability of these representations, it is prudent to remove the zero eigenvalues when determining the condition number.

9.1.1 Definition Point Spectrum

Let T be a non-invertible bounded linear operator. Then the point spectrum $\sigma_p(T)$ consists of all the nonzero eigenvalues of T.

9.1.2 Definition non-zero discrete eigenvalues

Let $A \in M_{n\times n}$ be a symmetric matrix, with eigenvalues $0 < \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$. Then an eigenvalue λ_i is said to be non-zero if and only if

$$\lambda_{i+1} \quad \text{is non-zero} \tag{9.7}$$

$$\frac{\lambda_i}{\lambda_{i+1}} > 10^{-7} \tag{9.8}$$

$$\lambda_i > 10^{-4}h^4. \tag{9.9}$$

9.1.3 Definition non-zero discrete condition number

$\kappa_{pd}(A) = \frac{\max|\sigma_p(A)|}{\min|\sigma_p(A)|}$ with the discrete definition of non-zero eigenvalues.

For the two representations, this works out to removing the 3 and 4 smallest eigenvalues ((9.4) and (9.3) respectively) before computing the condition numbers. Eight different scaling of the basis functions is selected. The $\sqrt{3}$ scaling is added to make all the diagonal elements of a B be exactly the same.

Table 9.2: Condition numbers (κ_{pd}) of matrices arising from types of scaling for Hermite elements

number of elements	$B^1_{1,1,1}$ cond number	$B^1_{1,\sqrt{3},1}$ cond number	$B^1_{1,\sqrt{3},2}$ cond number
$1 = (2^0)^2$	2584.6	1116.5	500.35
$4 = (2^1)^2$	382880	130028	49000
$16 = (2^2)^2$	51853049	17331457	6468032
$64 = (2^3)^2$	8252398263	2755125937	1016105092
$256 = (2^4)^2$	1.5743e+12	525437544994	192216629691
$1024 = (2^5)^2$	3.4127e+14	no result	no result

9.1.4 The condition number of a Biharmonic matrix

$$\begin{aligned}
(B^N_{2,w,p})_{i,j} &= \int \mathbf{D}^2(H^N_i) : \mathbf{D}^2(H^N_j)\,\mathrm{d}x \\
(B^N_{1,w,p})_{i,j} &= \int \Delta(H^N_i)\Delta(H^N_j)\,\mathrm{d}x \\
\phi_0(0,0) &= 1 \\
\frac{\partial \phi^N_1(0,0)}{\partial x} &= wN \\
\frac{\partial \phi^N_2(0,0)}{\partial y} &= wN \\
\frac{\partial^2 \phi^N_3(0,0)}{\partial x \partial y} &= (wN)^p
\end{aligned}$$

With this, the tables 9.2, 9.3, 9.4 and 9.5. As we can see in these tables full scaling with N is better than reduced scaling with N. Reduced Scaling with N is better than no scaling. Adding the $\sqrt{3}$ scaling improves the condition numbers further. The Hessian inner product ((9.4)) is better than the biharmonic inner product ((9.3)).

9.1.5 The condition number of the system as one equation

The ϵ-k equations is expressed in two different variational forms:

$$\int_\Omega \Delta\epsilon(x)\Delta v + \epsilon(x)v\,\mathrm{d}x = \int_\Omega \Delta f_2(x)v + f_1(x)v\,\mathrm{d}x + \text{boundery term} \quad (9.10)$$

$$\int_\Omega \mathbf{D}^2(\epsilon) : \mathbf{D}^2(v) + \epsilon v\,\mathrm{d}x = \int_\Omega \Delta f_2 v + f_1 v\,\mathrm{d}x + \text{boundery term.} \quad (9.11)$$

Table 9.3: Condition numbers (κ_{pd}) of matrices arising from types of scaling for Hermite elements

number of	$B^1_{2,1,1}$	$B^1_{2,\sqrt{3},1}$	$B^1_{2,\sqrt{3},2}$
elements	cond number	cond number	cond number
$1 = (2^0)^2$	2427.2	999.36	512.07
$4 = (2^1)^2$	54294	18351	9257.0
$16 = (2^2)^2$	975975	325526	163189
$64 = (2^3)^2$	16833179	5611233	2807417
$256 = (2^4)^2$	275872937	91957807	45986184
$1024 = (2^5)^2$	4442364060	1480788243	740423344

Table 9.4: Condition numbers (κ_{pd}) of matrices arising from types of scaling for Hermite elements

number of	$B^N_{1,1,1}$	$B^N_{1,\sqrt{3},1}$	$B^N_{1,1,2}$	$B^N_{1,\sqrt{3},2}$
elements	cond number	cond number	cond number	cond number
$1 = (2^0)^2$	2584.6	1116.53	2584.6	500.35
$4 = (2^1)^2$	98437	35381	29571	5620.4
$16 = (2^2)^2$	3307999	1153197	433374	141023
$64 = (2^3)^2$	135349903	49499784	14939717	8865144
$256 = (2^4)^2$	7154662894	3059553590	1324295802	1112003947
$1024 = (2^5)^2$	no result	308279719496	212530478242	202342715830

Table 9.5: Condition numbers (κ_{pd}) of matrices arising from types of scaling for Hermite elements

number of	$B^N_{2,1,1}$	$B^N_{2,\sqrt{3},1}$	$B^N_{2,1,2}$	$B^N_{2,\sqrt{3},2}$
elements	cond number	cond number	cond number	cond number
$1 = (2^0)^2$	2427.2	999.36	2427.2	347.73
$4 = (2^1)^2$	13860	4890.1	3547.3	451.39
$16 = (2^2)^2$	61292	20670	3982.1	749.48
$64 = (2^3)^2$	263290	87986	6953.1	5801.09
$256 = (2^4)^2$	1077887	359511	68489	65081
$1024 = (2^5)^2$	4338501	1446384	883276	871224

Table 9.6: Condition numbers (κ_1) of matrices arising from types of scaling for Hermite elements

number of	$H^1_{1,1,1}$	$H^1_{1,\sqrt{3},1}$	$H^1_{1,\sqrt{3},2}$
elements	cond number	cond number	cond number
$1 = (2^0)^2$	274174	119207	99570
$4 = (2^1)^2$	11824630	4004690	1866975
$16 = (2^2)^2$	128049428	42776856	18323535
$64 = (2^3)^2$	1631605949	544222077	198267421
$256 = (2^4)^2$	72057123133	24024240804	8315282872
$1024 = (2^5)^2$	3.4379e+12	1.1460e+12	389233623024

Table 9.7: Condition numbers (κ_1) of matrices arising from types of scaling for Hermite elements

number of	$H^1_{2,1,1}$	$H^1_{2,\sqrt{3},1}$	$H^1_{2,\sqrt{3},2}$
elements	cond number	cond number	cond number
$1 = (2^0)^2$	2432.38	1001.0	921.30
$4 = (2^1)^2$	54306	18355	13779
$16 = (2^2)^2$	975967	325523	181955
$64 = (2^3)^2$	16833145	5611222	2807938
$256 = (2^4)^2$	275872874	91957787	45986398
$1024 = (2^5)^2$	4442364155	1480788307	740423427

The tables 9.6, 9.7 ,9.8 and 9.9 provides the condition numbers for the left hand matrices as provided by FEM on these equations. $H^{\cdot}_{1,\cdot,\cdot} = B^{\cdot}_{1,\cdot,\cdot} + M^{\cdot}_{\cdot,\cdot}$ represents the equation (9.10) and $H^{\cdot}_{2,\cdot,\cdot} = B^{\cdot}_{2,\cdot,\cdot} + M^{\cdot}_{\cdot,\cdot}$ represents the equation (9.11). They are the sums of the biharmonic and the mass matrix, with the same scaling of the basis functions.

As we can see in these tables full scaling with N is better than reduced scaling with N. Reduced Scaling with N is better than no scaling. Adding the $\sqrt{3}$ scaling improves the condition numbers further. The Hessian inner product ((9.4)) is better than the biharmonic inner product ((9.3)).

9.2 Representations as a Linear system in 2D

In this section the k-ϵ model is implemented with Lagrange-3 and Hermite elements. The system is an $H^2 \times L^2$ representation and is defined in (9.12).

Table 9.8: Condition numbers (κ_1) of matrices arising from types of scaling for Hermite elements

number of elements	$H^N_{1,1,1}$ cond number	$H^N_{1,\sqrt{3},1}$ cond number	$H^N_{1,\sqrt{3},2}$ cond number
$1 = (2^0)^2$	274174	119207	99570
$4 = (2^1)^2$	3027583	1076377	286564
$16 = (2^2)^2$	8136984	2813752	530594
$64 = (2^3)^2$	26019491	9038317	1856347
$256 = (2^4)^2$	289291368	101728884	13539678
$1024 = (2^5)^2$	3451034726	1214028856	157858076

Table 9.9: Condition numbers (κ_1) of matrices arising from types of scaling for Hermite elements

number of elements	$H^N_{2,1,1}$ cond number	$H^N_{2,\sqrt{3},1}$ cond number	$H^N_{2,\sqrt{3},2}$ cond number
$1 = (2^0)^2$	2432.4	1001.0	922.27
$4 = (2^1)^2$	13862	7303.4	7295.6
$16 = (2^2)^2$	74437	58516	58522
$64 = (2^3)^2$	669028	612352	612364
$256 = (2^4)^2$	7969676	7761649	7761663
$1024 = (2^5)^2$	110812171	110021915	110021930

Table 9.10: Condition numbers (κ_1) of matrices arising from types of scaling for Hermite elements when solving the equations in 2-D

number of elements	$Co^1_{1,1}$ cond number	$Co^1_{3,1}$ cond number	$Co^1_{3,2}$ cond number
$1 = (2^0)^2$	5.3102e+06	2.1878e+06	7.6073e+05
$4 = (2^1)^2$	9.5665e+05	3.2232e+05	1.1365e+05
$16 = (2^2)^2$	5.2881e+07	1.7635e+07	5.9285e+06
$64 = (2^3)^2$	1.3235e+09	4.4118e+08	1.4781e+08

Table 9.11: Condition numbers (κ_1) of matrices arising from types of scaling for Hermite elements when solving the equations in 2-D

number of elements	$Co^N_{1,1}$ cond number	$Co^N_{3,1}$ cond number	$Co^N_{1,2}$ cond number	$Co^N_{3,2}$ cond number
$1 = (2^0)^2$	5.3102e+06	2.1878e+06	5.3102e+06	7.6073e+05
$4 = (2^1)^2$	2.4307e+05	8.4899e+04	6.6080e+04	9516.7
$16 = (2^2)^2$	3.3166e+06	1.1155e+06	2.2289e+05	3.4470e+04
$64 = (2^3)^2$	2.0696e+07	6.9126e+06	3.8493e+05	6.8329e+04

The preconditioned system is (9.13).

$$Co_{k,n} = \begin{bmatrix} M^H & S^D \\ (S^D)^T & M^L \end{bmatrix} \tag{9.12}$$

$$PrCo_{v,k,n} = \begin{bmatrix} B_{v,k,n} + M^H & 0 \\ 0 & M^L \end{bmatrix}^{-1} \begin{bmatrix} M^H & S^D \\ (S^D)^T & M^L \end{bmatrix} \tag{9.13}$$

where v represents the type of biharmonic matrix used (as in the previous section), k is the weighting of the Hermite basis functions and n is the power of the weighting of the double derivative evaluation. If $k = 3$ and $n = 2$ all the values along the diagonal of the Hermite stiffness matrix (without the mass matrix) will be of the same size. The tables for the scaled matrices of (9.12) are given in the tables 9.10 and 9.11. They confirm the importance of scaling the Hermite elements.

The tables for the scaled matrices of (9.13) are given in the tables 9.12 and 9.13. The condition numbers in the preconditioned tables increases. This implies that this implementation does not work. As is evident in 9.13 and 9.12, the scaling does not change the condition number, this is to be expected. Let T be a scaling matrix. T will be a diagonal matrix, with diagonal values corre-

Table 9.12: Condition numbers (κ_2) of matrices arising from types of scaling for Hermite elements when solving the equations in 2-D

number of elements	$PrCo^N_{1,1,1}$ cond number	$PrCo^N_{1,3,1}$ cond number	$PrCo^N_{1,1,2}$ cond number	$PrCo^N_{1,3,2}$ cond number
$1 = (2^0)^2$	2.7184e+04	2.7184e+04	2.7184e+04	2.7184e+04
$4 = (2^1)^2$	4513.0	4513.0	4513.0	4513.0
$16 = (2^2)^2$	2.5914e+05	2.5914e+05	2.5914e+05	2.5914e+05
$64 = (2^3)^2$	1.3422e+06	1.3422e+06	1.3422e+06	1.3422e+06

Table 9.13: Condition numbers (κ_2) of matrices arising from types of scaling for Hermite elements when solving the equations in 2-D

number of elements	$PrCo^N_{2,1,1}$ cond number	$PrCo^N_{2,3,1}$ cond number	$PrCo^N_{2,1,2}$ cond number	$PrCo^N_{2,3,2}$ cond number
$1 = (2^0)^2$	1.4648e+05	1.4648e+05	1.4648e+05	1.4648e+05
$4 = (2^1)^2$	9.2140e+04	9.2140e+04	9.2140e+04	9.2140e+04
$16 = (2^2)^2$	1.0307e+06	1.0307e+06	1.0307e+06	1.0307e+06
$64 = (2^3)^2$	1.0034e+07	1.0034e+07	3.8493e+05	1.0034e+07

sponding to the square root of the scaling of the basis function corresponding with that row and column. The elements on T^{-1} will be one divided by the corresponding element on T. Thus the preconditioned problem will be;

$$\begin{aligned}(TPrT)^{-1}(TCoT) =&T^{-1}Pr^{-1}T^{-1}TCoT\\ =&T^{-1}Pr^{-1}CoT.\end{aligned}$$

This gives no benefit when it comes to the condition number. The diagonal elements of the unscaled and the scaled preconditioned problems are the same, the only thing that changes is the off diagonal elements. Since the matrices are diagonally dominant, the change between different scalings is dramatic. However including the scaling increases the stability when calculating the preconditioner.

Chapter 10

Conclusions

In this thesis a system of elliptical partial differential equations called the simplified k-ϵ model has been examined. Its weak form is well defined analytically. The boundary conditions we can apply are dependent on which solution space we choose. For all $(\epsilon, k) \in H^1(\Omega) \times H^1(\Omega)$ the trace $\|(\epsilon, k)\|_{L^2(\partial\Omega)\times L^2(\partial\Omega)}$ is bounded by $\|(\epsilon, k)\|_{H^1(\Omega)\times H^1(\Omega)}$. For all $(\epsilon, k) \in H^2(\Omega) \times L^2(\Omega)$ the trace of ϵ ($\|\frac{\partial\epsilon}{\partial n}\|_{L^2(\partial\Omega)} + \|\epsilon\|_{L^2(\partial\Omega)}$) is bounded by $\|\epsilon\|_{H^2(\Omega)}$. Therefore these formulations were used with their respective boundary conditions. $(\epsilon, k) \in H^1(\Omega) \times H^1(\Omega)$ is used with one boundary condition on each variable, while $(\epsilon, k) \in H^2(\Omega) \times L^2(\Omega)$ is used for two boundary conditions on ϵ.

The formulations are well defined analytically and Riesz mappings has been used as preconditioners for each formulation.

In this master thesis the H^2 conforming Hermite elements has been successfully implemented. The identified issues pertaining to mesh refinement has been fixed by scaling. To scale a Hermite element we multiply the basis functions representative of the derivates based on the size of the element and how many derivatives it includes. The scaling of the Hermite elements allowed for numerical experiments. Various combinations of elements, scaling and weak formulations were tested. Many of the experiments were implemented both in one and two dimensions.

The results of the $H^2 \times L^2$ formulations, when continuous Lagrange (CG) is used, are not stable. The continuous Lagrange paired with Hermite elements is $H^2 \times H^1$ conforming. Because $H^1 \subset L^2$ the combination of Hermite and CG is also $H^2 \times L^2$ conforming. The idea was therefore that Hermite-CG formulation would work. This thesis have showed by numerical experiments that this idea is flawed, and the Hermite-CG formulation is not stable.

The results of the $H^2 \times L^2$ formulations, when discontinuous Lagrange (dCG) is used, are very uplifting. The condition numbers are very close to the condition numbers of the analytical calculations in (5.28). This implies that the preconditioned simplified k-ϵ system is successfully implemented and that

$$B \sim S^{dD} \cdot (S^{dD})^T. \tag{10.1}$$

The scaling of the basis functions for Hermite with dCG simulations shows its usefulness when comparing unpreconditioned tables. It is reasonable to suspect that scaling allows higher degree of mesh refinement for a fixed accuracy demand, even when a system is preconditioned.

As expected from [1] the $H^1 \times H^1$ formulation works. It gave quite good results as the mathematical formulation suggest. The condition numbers are stable as the tables of section 8.1.2 shows. However boundary conditions for both ϵ and k are required. It does not work very well if Hermite combined with CG elements are used.

In 2-dimensions with Hermite and CG elements the results is similar to the 1-dimensional case. Here, scaling shows its usefulness. We also learned that (9.6) was more stable than (9.5).

It is reasonable to suspect that the lessons learned about CG vs dCG in the 1-dimensional case also applies to the 2-dimensional cases. We can suspect this because the Hermite-CG system gave similar types of results in 1-dimensions as it did in 2-dimensions, and because the systems that did not work in 1 dimensions used elements that conformed with higher order Sobolev spaces than was strictly needed (see section 8.1.1 and 8.1.3).

Since the implementation of the preconditioned simplified k-ϵ model was successful the next logical step is to implement the full k-ϵ model with this new set of boundary conditions.

It is important to note that the $H^2 \times L^2$ formulations are symmetric. Therefore two new possible sets of boundary conditions can be used

$$BA_k = \begin{cases} k = g_1 & \text{on } \partial\Omega \\ \frac{\partial k}{\partial n} = g_2 & \text{on } \partial\Omega \end{cases} \tag{10.2}$$

$$BA_\epsilon = \begin{cases} \epsilon = g_1 & \text{on } \partial\Omega \\ \frac{\partial \epsilon}{\partial n} = g_2 & \text{on } \partial\Omega \end{cases}. \tag{10.3}$$

It is useful if the full k-ϵ model can be successfully implemented with the new set of boundary conditions. The results of this thesis open up for numerical approximations of the k-ϵ model in situations where it is possible to obtain accurate information about one of the variables, but not the other. This can be very useful in engineering where the k-ϵ models are among of the most commonly used turbulence models.

So from this thesis we should be able to conclude that the $H^2 \times L^2$ formulations work. It should be implemented with Hermite and dCG elements. The $H^1 \times H^1$ formulation worked, but only if two CG elements are used. It is important to not chose elements that conform with Sobolev spaces of higher order than the particular one we are interested in. The Hermite elements should be scaled to produce a more stable approximation.

Chapter 11

Appendix

11.1 Definitions

Definition of $L^2(\Omega)$ space [10] 1 *$L^2(\Omega)$ is the functions space of square integrable functions on Ω. u is said to be in $L^2(\Omega)$ if*

$$\|u\|_{L^2(\Omega)} = \left(\int_\Omega |u|^2 \, \mathrm{d}x \right)^{\frac{1}{2}} < \infty. \tag{11.1}$$

Definition of $H^n(\Omega)$ space [10] 2 *$H^n(\Omega)$ is the Sobolev space containing derivatives of power n and and L^2 type norm. The norm is defined as*

$$\|u\|_{H^n(\Omega)} = \left(\int_\Omega \sum_{|\alpha| \leq n} |D^\alpha u|^2 \, \mathrm{d}x \right)^{\frac{1}{2}} < \infty. \tag{11.2}$$

It has the following inner product:

$$(u, v)_{H^n(\Omega)} = \int_\Omega \sum_{|\alpha| \leq n} D^\alpha u D^\alpha v \, \mathrm{d}x. \tag{11.3}$$

Where $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_d)$ is a multi index, and $|\alpha| = \alpha_1 + \alpha_2 + \cdots + \alpha_d$. While d is the number of dimensions and

$$D^\alpha u = \frac{\partial^{\alpha_1} u}{\partial x_i^{alpha_1}} + \frac{\partial^{\alpha_2} u}{\partial x_2^{alpha_2}} + \cdots + \frac{\partial^{\alpha_d} u}{\partial x_d^{alpha_d}}$$

$u \in H^n \leftrightarrow \|u\|_{H^n} < \infty$.

Definition of $H_0^1(\Omega)$ space [10] 3 *$H_0^1(\Omega)$ is the Sobolev space containing derivatives of power n and and L^2 type norm. the norm is defined as:*

$$\|u\|_{H_0^1(\Omega)} = \left(\int_\Omega |\nabla u|^2 \,\mathrm{d}x\right)^{\frac{1}{2}} < \infty. \tag{11.4}$$

It has the following inner product:

$$(u, v)_{H_0^1(\Omega)} = \int_\Omega \nabla u \cdot \nabla v \,\mathrm{d}x. \tag{11.5}$$

Definition of $H^1(\Omega)$ space [10] 4 *$H^1(\Omega)$ is the Sobolev space containing derivatives of power n and and $L^2(\Omega)$ type norm. The norm is defined as:*

$$\|u\|_{H^1(\Omega)} = \left(\int_\Omega |\nabla u|^2 + |u|^2 \,\mathrm{d}x\right)^{\frac{1}{2}} < \infty. \tag{11.6}$$

It has the following inner product:

$$(u, v)_{H^1(\Omega)} = \int_\Omega \nabla u \cdot \nabla v + uv \,\mathrm{d}x \tag{11.7}$$

Note that $L^2(\Omega) = H^0(\Omega)$.

$C_c^\infty(\Omega)$ is the space of all infinitely continuous differentiable functions with compact support.
$H_0^n(\Omega) = H^n(\Omega) \cap C_c^{n-1}(\bar{\Omega})$.

11.2 Theorems and Functional analysis

Youngs inequality [10] 5

$$ab \leq \frac{a^2}{2} + \frac{b^2}{2}. \tag{11.8}$$

Minkowskis inequality [10] 6

$$\|f + g\|_{L^2(\Omega)} \leq \|f\|_{L^2(\Omega)} + \|g\|_{L^2(\Omega)}. \tag{11.9}$$

Definition of a Bounded Linear Functional [10] 7 *Let X be a Banach Space. A bounded linear operator $u^* : X \to \mathbb{R}$ is called a bounded linear functional on X.*

We write X^* to denote the collection of all bounded linear functionals on X. X^* is the dual space of X.

Let H be a Hilbert space with inner product $(\cdot,\cdot)$ and $u \in H$, $u^* \in H^*$ (so $u^*(u) \in \mathbb{R}$).

Riesz Representation Theorem [10] 8 *H^* can canonically be identified with H. For each $u^* \in H^*$ there exists a unique element $u \in H$ such that $u^*(v) = (u, v)$ for all $v \in H$. $u^* \mapsto u$ is a linear isomorphism from H^* onto H.*

Let $\mathcal{L}(H^*, H)$ be the set of all linear functionals from H^* to H

Definition of Riesz map 9 *$R \in \mathcal{L}(H^*, H)$ is the Riesz map if: Given $u^* \in H^*$ then $u^*(v) = (R(u^*), v)_H$.*

Independent Eigenvector Theorem 1 *If A is an real $N \times N$ matrix with $\{\lambda\}_{i=1}^N$ distinct real nonzero eigenvalues. Then any set of $\{e_i \neq 0\}_{i=1}^N$ eigenvectors with $Ae_i = \lambda_i e_i$ form a basis for $\mathbb{R}^N$*

Proof

It is sufficient to prove that $\{e_i\}_{i=1}^N$ is a linearly independent set. If $\{e_i\}_{i=1}^N$ is dependent, then there must exist at least one minimal subset of $2 \leq j \leq N$ linearly dependent vectors $\{v_i\}_{i=1}^j$, with the eigenvalues $\{\rho_i\}_{i=1}^j$. Since the eigenvectors are dependent there must exist as set $\{a_i\}_{i=1}^j$ with at least two non-zero constants such that,

$$a_1 v_1 + a_2 v_2 + \cdots + a_j v_j = 0.$$

We can assume that $a_j \neq 0$ and define $b_k = -a_k/a_j$ and get

$$v_j = b_1 v_1 + v_2 v_2 + \cdots + b_{j-1} v_{j-1}. \tag{11.10}$$

Multiply (11.10) with A and get

$$\rho_j v_j = \rho_1 b_1 v_1 + \rho_2 v_2 v_2 + \cdots + \rho_{j-1} b_{j-1} v_{j-1}. \tag{11.11}$$

Multiply (11.10) with ρ_j and get

$$\rho_j v_j = \rho_j b_1 v_1 + \rho_j v_2 v_2 + \cdots + \rho_j b_{j-1} v_{j-1}. \tag{11.12}$$

Subtract (11.11) from (11.12) and get

$$0 = (\rho_j - \rho_1) b_1 v_1 + (\rho_j - \rho_2) v_2 v_2 + \cdots + (\rho_j - \rho_{j-1} b_{j-1} v_{j-1}. \tag{11.13}$$

Since at least one of the b_k is non-zero, the set $\{v_i\}_{i=1}^{j-1}$ is linearly dependent, and we have a contradiction.

Matrixes for 1D

$$
\begin{aligned}
B_{i,j} &= \int \Delta H_i \Delta H_j \,\mathrm{d}x \\
A^H_{i,j} &= \int DH_i \cdot DH_j \,\mathrm{d}x \\
A^D_{i,j} &= \int -H_i \Delta H_j \,\mathrm{d}x \\
S^D_{i,j} &= \int -L_i \Delta H_j \,\mathrm{d}x \\
S^S_{i,j} &= \int \nabla L_i \cdot \nabla H_j \,\mathrm{d}x \\
A^L_{i,j} &= \int \nabla L_i \cdot \nabla L_j \,\mathrm{d}x \\
M^S_{i,j} &= \int H_i L_j \,\mathrm{d}x \\
M^L_{i,j} &= \int L_i L_j \,\mathrm{d}x \\
M^H_{i,j} &= \int H_i H_j \,\mathrm{d}x
\end{aligned}
$$

Matrixes for 2D

$$
\begin{aligned}
(B_2^N)_{i,j} &= \int \mathbf{D}^2(H_i^N) : \mathbf{D}^2(H_j^N)\,\mathrm{d}x \\
(B_1^N)_{i,j} &= \int \Delta(H_i^N)\Delta(H_j^N)\,\mathrm{d}x \\
A_{i,j}^H &= \int DH_i \cdot DH_j\,\mathrm{d}x \\
A_{i,j}^D &= \int -H_i \Delta H_j\,\mathrm{d}x \\
S_{i,j}^D &= \int -L_i \Delta H_j\,\mathrm{d}x \\
S_{i,j}^S &= \int \nabla L_i \cdot \nabla H_j\,\mathrm{d}x \\
A_{i,j}^L &= \int \nabla L_i \cdot \nabla L_j\,\mathrm{d}x \\
M_{i,j}^S &= \int H_i L_j\,\mathrm{d}x \\
M_{i,j}^L &= \int L_i L_j\,\mathrm{d}x \\
M_{i,j}^H &= \int H_i H_j\,\mathrm{d}x
\end{aligned}
$$

Where

$$
\begin{aligned}
\mathbf{D}^2 u &= \sum_{\substack{i=1 \\ j=i}}^{d} \frac{\partial^2 u}{\partial x_i \partial x_j} \\
\Delta u &= \sum_{i=1}^{d} \frac{\partial^2 u}{(\partial x_i)^2}
\end{aligned}
$$

and d is the number of dimensions.

11.3 Boundary condition table

Table 11.1: Table of different boundary conditions

$(\epsilon, k) \in H^2(\Omega) \times L^2(\Omega)$		$(\epsilon, k) \in H^1(\Omega) \times H^1(\Omega)$	
$g_1 =$	$g_2 =$	$g_1 =$	$g_2 =$
$\frac{\partial \epsilon}{\partial n}$	ϵ	ϵ	k
$(\epsilon, k) \in H^2(\overline{\Omega}) \times L^2(\Omega)$		$(\epsilon, k) \in H^1(\overline{\Omega}) \times H^1(\Omega)$	
$\frac{\partial^2 \epsilon}{(\partial n)^2}$	ϵ	$\frac{\partial \epsilon}{\partial n}$	ϵ
$\frac{\partial^2 \epsilon}{(\partial n)^2}$	$\frac{\partial \epsilon}{\partial n}$	$\frac{\partial \epsilon}{\partial n}$	k
$(\epsilon, k) \in H^2(\Omega) \times L^2(\overline{\Omega})$		$(\epsilon, k) \in H^1(\Omega) \times H^1(\overline{\Omega})$	
ϵ	k	$\frac{\partial k}{\partial n}$	k
$\frac{\partial \epsilon}{\partial n}$	k	ϵ	$\frac{\partial k}{\partial n}$
$(\epsilon, k) \in H^2(\overline{\Omega}) \times L^2(\overline{\Omega})$		$(\epsilon, k) \in H^1(\overline{\Omega}) \times H^1(\overline{\Omega})$	
$\frac{\partial^2 \epsilon}{(\partial n)^2}$	k	$\frac{\partial \epsilon}{\partial n}$	$\frac{\partial k}{\partial n}$

Bibliography

[1] B. Mohammadi and O. Pironneau, *Applied Shape Optimization for Fluids*. Oxford: Clarendon press, 2001.

[2] R. M. Smith, "On the finite-element calculation of turbulent flow using the k-ϵ model," *International Journal for Numerical Methods in Fluids*, vol. 4, pp. 303–319, 1984.

[3] R. M. Smith, "A practical method of two-equation turbulence modelling using finite elements," *International Journal for Numerical Methods in Fluids*, vol. 4, pp. 321–336, 1984.

[4] F. Brezzi and M. Fortin, *Mixed and Hybrid Finite Element Methods*. Springer, 1991.

[5] I. Babuška, "Error-bounds for finite element method," *Numerische Mathematik*, vol. 16, pp. 322–333, 1971.

[6] K. A. Mardal and R. Winther, *Efficient Preconditioned Solution Methods for Elliptic Partial Differential Equations*, ch. 4, Construction of Preconditioners by Mapping Properties for Systems of Partial Differential Equations, pp. 66–83. Bentham Science Publishers, 2011.

[7] K. A. Mardal and R. Winther, "Preconditioning discretizations of systems of partial differential equations," *Numerical Linear Algebra with Applications*, vol. 18, pp. 1–40, 2011.

[8] K. Eriksson, D. Estep, P. Hansbo, and C. Johnson, *Computational Differential Equations*. Lund: Studentlitteratur, 1996.

[9] A. Logg, K. A. Mardal, and G. N. Wells, *Automated Solution of Differential Equations by the Finite Element Method - The FEniCs Book*. Springer, 2012.

[10] L. C. Evans, *Partial Differential Equations*. American Mathematical Society, 2 ed., 2010.

[11] Encyclopedia of Mathematics, "Babuška-Lax-Milgram Theorem." http://www.encyclopediaofmath.org/, 2012. Retrieved 4. jan 2013.

[12] A. Tveito and R. Winther, *Introduction to Partial Differential Equations - A Computational Approach*. Springer, 1998.

[13] E. Jones, T. Oliphant, P. Peterson, *et al.*, "SciPy: Open source scientific tools for Python." http://www.scipy.org/, 2001–.

[14] M. Alnæs and K.-A. Mardal, "Syfi and sfc: symbolic finite elements and form compilation," in *Automated Solution of Differential Equations by the Finite Element Method* (A. Logg, K.-A. Mardal, and G. Wells, eds.), vol. 84 of *Lecture Notes in Computational Science and Engineering*, pp. 273–282, Springer Berlin Heidelberg, 2012.

[15] Octave community, "GNU/Octave." www.gnu.org/software/octave/, 2012.

[16] C. Bauer, A. Frink, R. B. Kreckel, *et al.*, "Ginac is not a cas." http://www.ginac.de, 2012.