

Preconditioners for Karush–Kuhn–Tucker Systems arising in Optimal Control

Astrid Battermann

Thesis submitted to the Faculty of the Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE
IN
MATHEMATICS

APPROVED:
Matthias Heinkenschloss, Chair
Christopher Beattie
John A. Burns

June 14, 1996
Blacksburg, Virginia

Keywords: Preconditioning, Karush–Kuhn–Tucker Systems, Indefinite Systems,
Quadratic Programming, Optimal Control

Copyright 1996, Astrid Battermann

PRECONDITIONERS FOR KARUSH–KUHN–TUCKER SYSTEMS ARISING IN OPTIMAL CONTROL

Astrid Battermann

Committee Chairman: Dr. Matthias Heinkenschloss

Mathematics

(ABSTRACT)

This work is concerned with the construction of preconditioners for indefinite linear systems. The systems under investigation arise in the numerical solution of quadratic programming problems, for example in the form of Karush–Kuhn–Tucker (KKT) optimality conditions or in interior–point methods. Therefore, the system matrix is referred to as a KKT matrix. It is not the purpose of this thesis to investigate systems arising from general quadratic programming problems, but to study systems arising in linear quadratic control problems governed by partial differential equations.

The KKT matrix is symmetric, nonsingular, and indefinite. For the solution of the linear systems generalizations of the conjugate gradient method, MINRES and SYMMLQ, are used. The performance of these iterative solution methods depends on the eigenvalue distribution of the matrix and of the cost of the multiplication of the system matrix with a vector. To increase the performance of these methods, one tries to transform the system to favorably change its eigenvalue distribution. This is called preconditioning and the nonsingular transformation matrices are called preconditioners. Since the overall performance of the iterative methods also depends on the cost of matrix–vector multiplications, the preconditioner has to be constructed so that it can be applied efficiently.

The preconditioners designed in this thesis are positive definite and they maintain the symmetry of the system. For the construction of the preconditioners we strongly exploit the structure of the underlying system. The preconditioners are composed of preconditioners for the submatrices in the KKT system. Therefore, known efficient preconditioners can be readily adapted to this context. The derivation of the preconditioners is motivated by the properties of the KKT matrices arising in optimal control problems. An analysis of the preconditioners is given and various cases which are important for interior point methods are treated separately. The preconditioners are tested on a typical problem, a Neumann boundary control for an elliptic equation. In many important situations the preconditioners substantially reduce the number of iterations needed by the solvers. In some cases, it can even be shown that the number of iterations for the preconditioned system is independent of the refinement of the discretization of the partial differential equation.

Acknowledgments

I would like to express my sincere gratitude to my advisor and committee chairman, Dr. Matthias Heinkenschloss, for his support and guidance which made this work possible.

I want to thank Dr. Burns and Dr. Beattie for serving on my committee. They were always encouraging and supportive.

Special thanks to all people in ICAM. I am glad that I got to know them.

I owe a lot to my parents. Their support was very important during this year.

Contents

1	Introduction	1
1.1	Motivation	1
1.2	Outline of the Thesis	3
2	The Quadratic Programming Problem and the KKT Matrix	4
2.1	The Quadratic Programming Problem	4
2.2	Interior-Point Methods for the Solution of the Quadratic Programming Problem	6
2.3	Three Cases	8
2.3.1	No Bound Constraints	8
2.3.2	Bound Constraints for u	8
2.3.3	Bound Constraints for u and y	9
3	Eigenvalue Estimates	12
3.1	The Eigenvalue Distribution	12
3.2	A Theorem by Rusten and Winther	13
4	SYMMLQ AND MINRES	16
4.1	Introduction to SYMMLQ and MINRES	16
4.2	Derivation of SYMMLQ and MINRES	17
4.3	Convergence Analysis	20
4.3.1	Convergence Results for MINRES	24
4.3.2	Convergence Results for SYMMLQ	34
4.4	Implementation of SYMMLQ and MINRES	34
4.4.1	Orthogonal Bases for the Krylov Subspaces	37
4.4.2	SYMMLQ	41
4.4.3	MINRES	48
5	Preconditioning	52
5.1	The Issue of Preconditioning	52
5.2	The Preconditioned Algorithms	53

6	The Preconditioners	61
6.1	Introduction	61
6.2	The First Preconditioner	63
6.2.1	Derivation of the First Preconditioner	63
6.2.2	Expected Performance of the First Preconditioner	67
6.2.3	Comparison with Gill, Murray, Ponceleón and Saunders	69
6.2.4	Application of the First Preconditioner	70
6.3	The Second Preconditioner	70
6.3.1	Derivation of the Ideal Second Preconditioner	71
6.3.2	Derivation of the General Second Preconditioner	72
6.3.3	Expected Performance of the Second Preconditioner	74
6.3.4	Application of the Second Preconditioner	75
6.3.5	Quality of the Solution	76
6.4	The Third Preconditioner	79
6.4.1	Derivation of the Third Preconditioner	79
6.4.2	Expected Performance of the Third Preconditioner	80
6.4.3	Application of the Third Preconditioner	81
6.4.4	Quality of the Solution	82
7	Applications	85
7.1	Neumann Control for an Elliptic Equation	85
7.2	The Problem Discretization	85
7.3	Eigenvalues of FEM Matrices	88
7.4	Condition Number of the KKT–System	90
7.5	Numerical Results without a Preconditioner	92
7.6	Numerical Results with the First Preconditioner	100
7.7	Numerical Results with the Second Preconditioner	109
7.8	Numerical Results with the Third Preconditioner	117
8	Conclusion and Future Work	125
8.1	Conclusion	125
8.2	Future Work	126

List of Figures

7.1	The grid for $n_x = n_y = 5$	86
7.2	The eigenvalues and singular values of the submatrices in K for $n_x = n_y = 20$ and $\alpha = 1, D_y = 0, D_u = 0$	94
7.3	The eigenvalues of the KKT-system for $n_x = n_y = 20$ and $\alpha = 1, D_y = 0, D_u = 0$	95
7.5	The eigenvalues and singular values of the submatrices before preconditioning for a grid $n_x = n_y = 20$ with $D_u = 10^4 \cdot I, D_y = 0, \alpha = 1$	97
7.6	The eigenvalues of the KKT-system before preconditioning for $n_x = n_y = 20$ with $D_u = 10^4 \cdot I, D_y = 0, \alpha = 1$	97
7.7	The eigenvalues of the KKT-system before preconditioning for $n_x = n_y = 20$ with $D_y = 10^4 \cdot I, D_u = 0, \alpha = 1$	98
7.4	The residuals, the absolute and the relative error of MINRES- and SYMMLQ-iterates on the system K for $n_x = n_y = 5$ with $D_y = 0, D_u = 0, \alpha = 1$	99
7.8	The eigenvalues and singular values of the preconditioned submatrices in $P_1^{-1} K P_1^{-T}$ with $\alpha = 1, D_y = 0, D_u = 0$ for $n_x = n_y = 20$	102
7.9	The eigenvalues of the preconditioned KKT-matrix $P_1^{-1} K P_1^{-T}$ with $\alpha = 1, D_y = 0, D_u = 0$ for $n_x = n_y = 20$	103
7.10	The eigenvalues and singular values of the submatrices in $P_1^{-1} K P_1^{-T}$ with $D_u = 10^4 \cdot I, D_y = 0, \alpha = 1$ for $n_x = n_y = 20$	105
7.11	The eigenvalues of $P_1^{-1} K P_1^{-T}$ with $D_u = 10^4 \cdot I, D_y = 0, \alpha = 1$ for $n_x = n_y = 20$	105
7.13	The eigenvalues of $P_1^{-1} K P_1^{-T}$ with $D_y = 10^4 \cdot I, D_u = 0, \alpha = 1$ for $n_x = n_y = 20$	107
7.12	The residuals, the absolute and the relative error of MINRES- and SYMMLQ-iterates on the system $P_1^{-1} K P_1^{-T}$ for $n_x = n_y = 5$ with $\alpha = 1, D_y = 0, D_u = 0$	108
7.14	The eigenvalues and singular values of the preconditioned submatrices in $P_2^{-1} K P_2^{-T}$ for $n_x = n_y = 20, \alpha = 1, D_y = 0, D_u = 0$	110
7.15	The eigenvalues of the preconditioned KKT-matrix $P_2^{-1} K P_2^{-T}$ for $n_x = n_y = 20, \alpha = 1, D_y = 0, D_u = 0$	111
7.16	The eigenvalues of the KKT matrix $P_2^{-1} K P_2^{-T}$ with $D_y = 10^4 \cdot I, \alpha = 1, D_u = 0$ for $n_x = n_y = 20$	114

7.17	The residuals, the absolute and the relative error of MINRES- and SYMMLQ-iterates on the system $P_2^{-1}KP_2^{-T}$ for $n_x = n_y = 10$ with $D_y = 0, D_u = 0, \alpha = 1$	115
7.18	The residuals, the absolute and the relative error of MINRES- and SYMMLQ-iterates on the system $P_2^{-1}KP_2^{-T}$ for $n_x = n_y = 10$ with $D_y = 0, D_u = 0, \alpha = 10^{-5}$	116
7.19	The eigenvalues of the submatrices in $P_3^{-1}KP_3^{-T}$ for $n_x = n_y = 20, \alpha = 1, D_y = 0, D_u = 0$	118
7.20	The eigenvalues of the preconditioned KKT-matrix $P_3^{-1}KP_3^{-T}$ for $n_x = n_y = 20, \alpha = 1, D_y = 0, D_u = 0$	119
7.21	The eigenvalues of the submatrices in $P_3^{-1}KP_3^{-T}$ for $n_x = n_y = 20, \alpha = 10^{-5}, D_y = 0, D_u = 0$	120
7.22	The eigenvalues of the preconditioned KKT-matrix $P_3^{-1}KP_3^{-T}$ for $n_x = n_y = 20, \alpha = 10^{-5}, D_y = 0, D_u = 0$	121
7.23	The residuals, the absolute and the relative error of MINRES- and SYMMLQ-iterates on the system $P_3^{-1}KP_3^{-T}$ for $n_x = n_y = 10$ with $D_y = 0, D_u = 0, \alpha = 1$	123
7.24	The residuals, the absolute and the relative error of MINRES- and SYMMLQ-iterates on the system $P_3^{-1}KP_3^{-T}$ for $n_x = n_y = 10$ with $D_y = 0, D_u = 0, \alpha = 10^{-7}$	124

List of Tables

7.1	Computed and estimated spectrum of K with $\alpha = 1$, $D_y = 0$, $D_u = 0$	94
7.2	Iterations of MINRES and SYMMLQ on K with $\alpha = 1$, $D_y = 0$, $D_u = 0$. . .	95
7.3	Condition numbers of the system K and the submatrices for different grid sizes.	95
7.4	Largest value of α for that MINRES and SYMMLQ can no longer compute a solution to the system with K within the required accuracy in less than $2m + n$ steps.	96
7.5	Iterations of MINRES and SYMMLQ for K with $\alpha = 1$ and $D_u = 10^4 \cdot I$, $D_y = 0$	98
7.6	Iterations of MINRES and SYMMLQ for K with $\alpha = 1$ and $D_y = 10^4 \cdot I$, $D_u = 0$	98
7.7	Computed and estimated spectrum of $P_1^{-1}KP_1^{-T}$ with $\alpha = 1$, $D_y = 0$, $D_u = 0$.	102
7.8	Condition numbers of the preconditioned system $P_1^{-1}KP_1^{-T}$ with $\alpha = 1$, $D_y = 0$, $D_u = 0$ and the submatrices for different grid sizes.	103
7.9	Iterations of MINRES and SYMMLQ for $P_1^{-1}KP_1^{-T}$ with $\alpha = 1$, $D_y = 0$, $D_u = 0$	103
7.10	Computed and estimated spectrum of $P_1^{-1}KP_1^{-T}$ with $\alpha = 10^{-5}$, $D_y = 0$, $D_u = 0$	104
7.11	Iterations of MINRES and SYMMLQ for $P_1^{-1}KP_1^{-T}$ with $\alpha = 10^{-5}$, $D_y = 0$, $D_u = 0$	104
7.12	Largest value of α for that MINRES and SYMMLQ can no longer compute a solution for $P_1^{-1}KP_1^{-1}$ with $D_y = 0$, $D_u = 0$ within the required accuracy in less than $2m + n$ steps.	104
7.13	Iterations of MINRES and SYMMLQ for $P_1^{-1}KP_1^{-T}$ with $D_u = 10^4 \cdot I$, $D_y = 0$, $\alpha = 1$	106
7.14	Iterations of MINRES and SYMMLQ for $P_1^{-1}KP_1^{-T}$ with $\alpha = 1$ and $D_y = 10^4$, $D_u = 0$	107
7.15	Computed spectrum of $P_2^{-1}KP_2^{-T}$ with $\alpha = 1$, $D_y = 0$, $D_u = 0$	110
7.16	Condition numbers of the preconditioned system $P_2^{-1}KP_2^{-T}$ and the submatrices for different gridsizes; $\alpha = 1$, $D_y = 0$, $D_u = 0$	111
7.17	Iterations of MINRES and SYMMLQ for $P_2^{-1}KP_2^{-T}$ with $\alpha = 1$, $D_y = 0$, $D_u = 0$	112
7.18	Computed spectrum of $P_2^{-1}KP_2^{-T}$ with $\alpha = 10^{-5}$, $D_y = 0$, $D_u = 0$	113

7.19	Largest value of α for that MINRES and SYMMLQ can no longer compute a solution to the system with $P_2^{-1}KP_2^{-1}$ ($D_y = 0, D_u = 0$) within the required accuracy in less than the maximal number of steps.	113
7.20	Iterations of MINRES and SYMMLQ for $P_2^{-1}KP_2^{-T}$ with $D_u = 10^4 \cdot I, \alpha = 1, D_y = 0$	113
7.21	Iterations of MINRES and SYMMLQ for $P_2^{-1}KP_2^{-T}$ with $\alpha = 1$ and $D_y = 10^4 \cdot I, D_u = 0$	114
7.22	Computed spectrum of $P_3^{-1}KP_3^{-T}$ with $\alpha = 1, D_y = 0, D_u = 0$	118
7.23	Iterations of MINRES and SYMMLQ on $P_3^{-1}KP_3^{-T}$ with $\alpha = 1, D_y = 0, D_u = 0$	119
7.24	Condition numbers of $P_3^{-1}KP_3^{-T}$ and $W^T H W$ with $\alpha = 1, D_y = 0, D_u = 0$	119
7.25	Computed spectrum of $P_3^{-1}KP_3^{-T}$ with $\alpha = 10^{-5}, D_y = 0, D_u = 0$	120
7.26	Iterations of MINRES on $P_3^{-1}KP_3^{-T}$ for decreasing values of α with $D_y = 0, D_u = 0$. The values of α are given on the top line.	120
7.27	Iterations of MINRES and SYMMLQ on $P_3^{-1}KP_3^{-T}$ with $D_u = 10^4 \cdot I, \alpha = 1, D_y = 0$	122
7.28	Iterations of MINRES and SYMMLQ for $P_3^{-1}KP_3^{-T}$ with $D_y = 10^4 \cdot I, \alpha = 1, D_u = 0$	122

Chapter 1

Introduction

1.1 Motivation

In this work we are concerned with the construction of preconditioners for the indefinite linear system

$$\begin{pmatrix} H_y & 0 & A^T \\ 0 & H_u & B^T \\ A & B & 0 \end{pmatrix} \begin{pmatrix} y \\ u \\ p \end{pmatrix} = \begin{pmatrix} -c \\ -d \\ b \end{pmatrix}, \quad (1.1)$$

where

$$\begin{aligned} y \in \mathbb{R}^m, u \in \mathbb{R}^n, p \in \mathbb{R}^m, c \in \mathbb{R}^m, d \in \mathbb{R}^n, b \in \mathbb{R}^m, \\ H_y \in \mathbb{R}^{m \times m}, H_u \in \mathbb{R}^{n \times n}, A \in \mathbb{R}^{m \times m}, B \in \mathbb{R}^{m \times n}. \end{aligned} \quad (1.2)$$

The systems we are interested in arise in the numerical solution of quadratic programming problems

$$\text{Minimize } \frac{1}{2} y^T M_y y + \frac{\alpha}{2} u^T M_u u + c^T y + d^T u \quad (1.3)$$

subject to

$$Ay + Bu = b \quad (1.4)$$

and

$$\begin{aligned} y_{\text{low}} \leq y \leq y_{\text{upp}}, \\ u_{\text{low}} \leq u \leq u_{\text{upp}} \end{aligned} \quad (1.5)$$

by interior–point methods. In this case, the matrices H_y and H_u are of the form

$$H_y = M_y + D_y \quad \text{and} \quad H_u = \alpha \cdot M_u + D_u$$

with nonnegative diagonal matrices $D_y \in \mathbb{R}^{m \times m}$, $D_u \in \mathbb{R}^{n \times n}$, and $\alpha \in \mathbb{R}$.

Since the system (1.1) is related to the Karush–Kuhn–Tucker optimality conditions, we refer to it as a Karush–Kuhn–Tucker system. The system matrix in (1.1) will be called a Karush–Kuhn–Tucker (KKT) matrix and denoted by K .

We do not investigate systems (1.1) arising from general quadratic programming problems, but study systems arising in linear quadratic control problems governed by partial

differential equations. A typical example is the Neumann boundary control for an elliptic equation, given as follows:

$$\text{Minimize } \frac{1}{2} \int_{\Omega} (y(x) - y_d(x))^2 dx + \frac{\alpha}{2} \int_{\partial\Omega} u^2(x) ds \quad (1.6)$$

over all (y, u) satisfying the state equation

$$\begin{aligned} -\Delta y(x) + y(x) &= f(x) & x \in \Omega, \\ \frac{\partial}{\partial n} y(x) &= u(x) & x \in \partial\Omega. \end{aligned} \quad (1.7)$$

After discretization with finite elements, this leads to a quadratic programming problem of the form (1.3) to (1.5). In this situation it can be assumed that $A \in \mathbb{R}^{m \times m}$ is nonsingular, which corresponds to the unique solvability of the discretized differential equation. Moreover, we assume that H_y and H_u are positive definite. In our applications the matrices M_y and M_u are positive definite. Since D_y and D_u are nonnegative diagonal matrices the assumption of positive definiteness of H_y and H_u is satisfied if $\alpha > 0$. So the system matrix is nonsingular. Since the submatrices H_y and H_u are symmetric, the KKT-matrix is symmetric. It can be shown that the KKT-matrix is indefinite.

For the solution of linear systems of the form (1.1) we use iterative solution methods. Systems arising from the discretization of differential equations tend to be very large. Considering this, iterative solvers are a suitable approach. A well known iterative solution method, frequently used for large and sparse systems, is the conjugate gradient method. However, the conjugate gradient method can not be used for systems of the form (1.1), because the Karush–Kuhn–Tucker matrix is indefinite. Instead, we employ the Krylov subspace methods MINRES and SYMMLQ, derived by Paige and Saunders [13], to solve the linear system. These are generalizations of the conjugate gradient method, applicable for symmetric indefinite systems. What makes these iterative solution methods particularly attractive is the fact that they only require products with the system matrix. Thus it is not necessary to actually assemble the entire system. Moreover, the sparsity structure of the system can be exploited. Matrices arising from the discretization of differential equations usually have few nonzero entries. Taking advantage of the sparsity structure of the matrix in the implementation is often easily realized with a special routine to compute the matrix–vector product. However, MINRES and SYMMLQ may require a large number of iterations to compute a solution to the system (1.1). The convergence of MINRES and SYMMLQ depend on the distribution of the eigenvalues of K . Large spreads and little clustering in the spectrum of K leads to slow convergence of the iterative methods. Therefore, one tries to find a linear system \tilde{K} equivalent to the original one, but with 'better' eigenvalue distribution. Transforming the original system into an equivalent system by similarity transformations to improve the performance of solution methods on the system is called preconditioning. If one wants to maintain symmetry, one tries to find similarity transformations $\tilde{K} = P^{-1} K P^{-T}$ so that \tilde{K} has a 'better' eigenvalue distribution than K . What is meant by a 'better' eigenvalue distribution will be made precise later in this thesis.

The construction of preconditioners for the system (1.1) in the situation outlined earlier is the main purpose of this work. In the design of preconditioners one has to outbalance

between the efficiency of preconditioners regarding their application and the degree to which they improve the eigenvalue distribution. Assuming nonsingularity, the condition number of any matrix can be reduced to one. But the costs are often judged too high.

The preconditioners $M = PP^T$ we suggest are positive definite. Furthermore, they maintain the symmetry of the system. For the construction of our preconditioners we strongly exploit the structure of the underlying system (1.1). A factorization of the system is not attempted. Our preconditioners are composed of preconditioners for the submatrices H_y , H_u and A .

1.2 Outline of the Thesis

The thesis is organized as follows.

In § 2 we investigate the quadratic programming problem (1.3) to (1.5). We discuss the Necessary and Sufficient Optimality Conditions, the Karush–Kuhn–Tucker Conditions, and briefly describe interior–point methods for the solution of the system arising from the KKT–conditions. This will give us some insight into the structure of the systems considered in the remainder of this work.

Results on the eigenvalue distribution of this system are reviewed in § 3.

The iterative solution methods MINRES and SYMMLQ are presented in § 4. Since their convergence behavior is a point of main interest to us, a large part of that chapter is dedicated to convergence analysis.

The general notion of preconditioning is reviewed in § 5. Subsequently, we derive the changes that are necessary in the implementation of MINRES and SYMMLQ to incorporate preconditioners.

In § 6 we present the preconditioners designed in this work. These are derived and analyzed based on the properties of the system matrix K established earlier. We investigate the expected gain in the solution process due to changes in the spectrum is of interest as well as the costs of applying the preconditioners.

We test the preconditioners on a typical problem in § 7.1. This application is the Neumann boundary control for an elliptic equation mentioned earlier. It gives rise to a system matrix of the form (1.1). Our analysis covers the original linear system and the numerical results for the preconditioned systems. We will see that in some important situations we can substantially lower the number of iterations needed by MINRES and SYMMLQ. In some cases the number of iterations is independent of the mesh used in the discretization of the problem.

A conclusion is drawn in § 8.1, and we look ahead on future work in § 8.2.

Most of the results in § 2 through § 5 can be found in the literature. However, these results are adapted and presented in a form suitable to motivate the design and allow the analysis of the preconditioners. Most of the material in § 6 and § 7 is original work.

Chapter 2

The Quadratic Programming Problem and the KKT Matrix

The systems (1.1) for which we want to construct preconditioners arise in methods for the solution of quadratic programming problems. To construct efficient preconditioners we first have to examine the structure of the system matrix K arising in these applications. This is the purpose of this section. First we will review some results concerning the quadratic programming problem and then we will sketch interior–point methods for its solution. It is not the purpose of this section to give a comprehensive overview, but rather to focus on the aspects important for the design of preconditioners. In this section we will also introduce some notation and motivate some of the assumptions that will be made.

2.1 The Quadratic Programming Problem

We consider the quadratic programming problem in standard form:

$$\text{Minimize } \frac{1}{2} \begin{pmatrix} y \\ u \end{pmatrix}^T \begin{pmatrix} M_{yy} & M_{yu} \\ M_{uy} & M_{uu} \end{pmatrix} \begin{pmatrix} y \\ u \end{pmatrix} + \begin{pmatrix} c \\ d \end{pmatrix}^T \begin{pmatrix} y \\ u \end{pmatrix} \quad (2.1)$$

subject to

$$Ay + Bu = b \quad (2.2)$$

and

$$y \geq 0, u \geq 0. \quad (2.3)$$

Using straightforward extensions, bound constraints of the form

$$y_{\text{low}} \leq y \leq y_{\text{upp}}, \quad (2.4)$$

$$u_{\text{low}} \leq u \leq u_{\text{upp}}$$

can be handled as well. However, to reduce the complexity of notation, we restrict our attention to the standard form (2.1) to (2.3). The problem (2.1) to (2.3) is called quadratic

programming problem. In the following we will often use QP to denote the quadratic programming problem.

Throughout this section we often use the notation

$$M = \begin{pmatrix} M_{yy} & M_{yu} \\ M_{uy} & M_{uu} \end{pmatrix}, \quad g = \begin{pmatrix} c \\ d \end{pmatrix}, \quad C = (A \mid B), \quad W = \begin{pmatrix} -A^{-1}B \\ I \end{pmatrix}$$

and

$$x = \begin{pmatrix} y \\ u \end{pmatrix}, \quad q = \begin{pmatrix} q_y \\ q_u \end{pmatrix}.$$

The existence of solutions of the QP (2.1) to (2.3) is guaranteed if the objective function is bounded from below on the set of feasible points.

Theorem 2.1.1 (Existence of Solutions) *Let M be positive semidefinite. If*

$$q(x) = \frac{1}{2}x^T Mx + g^T x$$

is bounded from below on the set of feasible points $\{(y, u) \mid Ay + Bu = b, y \geq 0, u \geq 0\}$, then the QP (2.1) to (2.3) admits a solution.

Proof: See e.g. [3, Appendix 2]. □

Necessary and sufficient optimality conditions are given by the Karush–Kuhn–Tucker conditions (2.5). We will in the following often use the short form KKT–conditions to refer to these conditions.

Theorem 2.1.2 (Necessary and Sufficient Optimality Conditions) *Let M be positive semidefinite. The vector (y, u) is a solution of (2.1) to (2.3) if and only if there exist $p \in \mathbb{R}^m$, $q_y \in \mathbb{R}^m$, and $q_u \in \mathbb{R}^n$ such that*

$$\begin{aligned} M_{yy}y + M_{yu}u + A^T p - q_y &= -d, \\ M_{uy}y + M_{uu}u + B^T p - q_u &= -c, \\ Ay + Bu &= b, \\ y^T q_y + u^T q_u &= 0, \\ q_y, q_u &\geq 0, \\ y, u &\geq 0. \end{aligned} \tag{2.5}$$

Proof: See e.g. [3, § 12.3], [10, pp. 192, 193]. □

2.2 Interior–Point Methods for the Solution of the Quadratic Programming Problem

In the presence of bound constraints (2.3) interior–point methods, in particular primal–dual Newton interior–point algorithms are very attractive methods for the solution of large–scale QPs. Unlike active set methods, which usually move along the boundary of the set $\{(y, u) \mid y \geq 0, u \geq 0\}$, interior–point methods, as suggested by the name, generate iterates (y, u) that are in the interior, i.e. satisfy $y > 0, u > 0$. This property allows interior–point methods to generate iterates that cut through the feasible set and move towards an optimum rather than exchanging one active index at a time and marching along the boundary towards an optimum. In many cases, this property allows one to prove polynomial complexity of interior–point methods. See e.g. [17] for an overview of interior–point methods. We will sketch primal–dual Newton interior–point methods and barrier methods for the solution of the QP (2.1) to (2.3).

We will employ the notation usual in interior–point methods: For a given vector x , the diagonal matrix with diagonal entries equal to the entries of x is denoted by X . Throughout this section, e denotes the vector of ones: $e = (1, \dots, 1)$.

The construction of primal–dual Newton interior–point is based on the so–called perturbed KKT conditions corresponding to (2.5), which are given by

$$\begin{aligned} Mx + C^T p - q &= -g, \\ Cx &= b, \\ XQe &= \theta e, \end{aligned} \tag{2.6}$$

and $x, q > 0$, where $\theta > 0$. To move from a current iterate (x, p, q) with $x, q > 0$ to the next iterate (x_+, p_+, q_+) , primal–dual Newton interior–point methods compute the Newton step $(\Delta x, \Delta p, \Delta q)$ for the perturbed KKT conditions (2.6) and set

$$(x_+, p_+, q_+) = (x + \alpha_x \Delta x, p + \alpha_p \Delta p, q + \alpha_q \Delta q),$$

where the step sizes $\alpha_x, \alpha_p, \alpha_q \in (0, 1]$ are chosen so that $x_+, q_+ > 0$. Then the perturbation parameter θ is updated based on $x_+^T q_+$ and the previous step is repeated. We refer to the literature, e.g. [4], [17], [18], for details. We will focus on the relation of the Newton system with the system (1.1) under consideration.

The Newton system for the perturbed KKT–conditions (2.6) is given by

$$\begin{pmatrix} M & C^T & -I \\ C & 0 & 0 \\ Q & 0 & X \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta p \\ \Delta q \end{pmatrix} = - \begin{pmatrix} Mx + C^T p - q + g \\ Cx - b \\ XQe - \theta e \end{pmatrix}. \tag{2.7}$$

The nonsymmetric system (2.7) can be reduced to a symmetric system. If we use the last equation in (2.7) to eliminate Δq ,

$$\Delta q = -X^{-1}Q\Delta x - Qe + \theta X^{-1}e \tag{2.8}$$

then we arrive at the system

$$\begin{pmatrix} M + X^{-1}Q & C^T \\ C & 0 \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta p \end{pmatrix} = - \begin{pmatrix} Mx + C^T p + g - \theta X^{-1}e \\ Cx - b \end{pmatrix} \quad (2.9)$$

or, using the original notation,

$$\begin{pmatrix} M_{yy} + Y^{-1}Q_y & M_{yu} & A^T \\ M_{uy} & M_{uu} + U^{-1}Q_u & B^T \\ A & B & 0 \end{pmatrix} \begin{pmatrix} \Delta y \\ \Delta u \\ \Delta p \end{pmatrix} = - \begin{pmatrix} M_{yy}y + M_{yu}u + A^T p + c - \theta Y^{-1}e \\ M_{uy}y + M_{uu}u + B^T p + d - \theta U^{-1}e \\ Ay + Bu - b \end{pmatrix}. \quad (2.10)$$

If $M_{yu} = 0, M_{uy} = 0$, the system (2.10) is of the form (1.1). As variables y_j or u_i approach the bound, i.e. approach zero, large quantities are added to the diagonals (j, j) or (i, i) , respectively. In actual computations more care must be taken during the reduction of the system (2.7) to avoid cancellation in the reduction process due to very large elements in X^{-1} , see e.g. [5]. A stable reduction of the system (2.7) is discussed in [5]. The unknowns and the right hand side in that reduced system differ from those in (2.10). However, the system matrix in the stable reduction is equal to the system matrix in (2.10). For our purposes it is therefore not necessary to present the lengthier stable reduction and we refer to [5] for details.

For completeness we also mention barrier methods for the solution of the QP (2.1) to (2.3). In a barrier method with logarithmic barrier function, one generates a sequence of iterates (y, u, p) with $y > 0, u > 0$ by approximately minimizing

$$\text{Minimize } \frac{1}{2} \begin{pmatrix} y \\ u \end{pmatrix}^T \begin{pmatrix} M_{yy} & M_{yu} \\ M_{uy} & M_{uu} \end{pmatrix} \begin{pmatrix} y \\ u \end{pmatrix} + \begin{pmatrix} c \\ d \end{pmatrix}^T \begin{pmatrix} y \\ u \end{pmatrix} - \mu \sum_{i=1}^m \ln(y_i) - \mu \sum_{i=1}^n \ln(u_i) \quad (2.11)$$

subject to

$$Ay + Bu = b \quad (2.12)$$

and $y, u > 0$. During the iteration, the positive barrier parameter μ is adjusted so that $\mu \rightarrow 0$. The minimization is performed by Newton's method. The KKT conditions for the problem (2.11), (2.12) are given by

$$\begin{aligned} Mx - \mu X^{-1}e + C^T p &= -g, \\ Cx &= b. \end{aligned} \quad (2.13)$$

The Newton system for (2.13) is given by

$$\begin{pmatrix} M + \mu X^{-2} & C^T \\ C & 0 \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta p \end{pmatrix} = - \begin{pmatrix} Mx - \mu X^{-1}e + C^T p + g, \\ Cx - b. \end{pmatrix}. \quad (2.14)$$

If $M_{yu} = 0, M_{uy} = 0$, the system (2.14) is of the form (1.1). As before, large quantities are added to the diagonals (j, j) or (i, i) , as variables y_j or u_i approach the bound, i.e. approach zero, respectively. For more details on the barrier method we refer to [17]. For a discussion of the relation, in particular the differences, between barrier methods and primal-dual Newton interior-point methods see [4].

2.3 Three Cases

To learn more about the QP and the reduced system (2.10) it will be helpful to distinguish among three cases. This discussion will also help to relate the results in this paper to the results on the solution of KKT–systems in barrier methods for linear programming that can be found in the literature, see e.g. [6], [7]. Obviously, the QP (2.1) to (2.3) reduces to a linear program if $M = 0$.

Throughout this subsection we assume that A is nonsingular. As a consequence, the matrix

$$C = (A \mid B)$$

has full row rank. We will also assume that the QP has a solution, i.e. that the KKT–system (2.5) has a solution.

2.3.1 No Bound Constraints

If the bound constraints are not active, then the Lagrange multipliers q_y and q_u are zero at the solution and the KKT–conditions (2.5) are equivalent to

$$\begin{pmatrix} M_{yy} & M_{yu} & A^T \\ M_{uy} & M_{uu} & B^T \\ A & B & 0 \end{pmatrix} \begin{pmatrix} y \\ u \\ p \end{pmatrix} = \begin{pmatrix} -c \\ -d \\ b \end{pmatrix}. \quad (2.15)$$

If the primal–dual Newton interior–point method is applied, then the matrices $Y^{-1}Q_y$ and $U^{-1}Q_u$ will converge towards zero and the system matrix in (2.10) will eventually be almost equal to the one in (2.5). If the matrix M is positive definite on the null–space of C , the system (2.15) has a unique solution.

2.3.2 Bound Constraints for u

Let y_*, u_* be a solution of the QP and suppose that the bound constraints for y_* are not active. Let $\{l_1, \dots, l_k\}$ denote the set of active indices for u_* ,

$$\{l_1, \dots, l_k\} = \{i \mid (u_*)_i = 0\}.$$

In this case the Lagrange multipliers at the solution satisfy

$$q_y = 0, \quad \text{and} \quad (q_u)_i = 0, \quad i \notin \{l_1, \dots, l_k\}.$$

If we define the matrix $I(u_*) \in \mathbb{R}^{k \times n}$ by

$$(I(u_*))_{ij} = \begin{cases} 1 & \text{if } j = l_i, \\ 0 & \text{otherwise,} \end{cases}$$

then the KKT conditions (2.5) are equivalent to

$$\begin{pmatrix} M_{yy} & M_{yu} & A^T & 0 \\ M_{uy} & M_{uu} & B^T & I(u_*)^T \\ A & B & 0 & 0 \\ 0 & I(u_*) & 0 & 0 \end{pmatrix} \begin{pmatrix} y \\ u \\ p \\ q_u^a \end{pmatrix} = \begin{pmatrix} -c \\ -d \\ b \\ 0 \end{pmatrix}, \quad (2.16)$$

where q_u^a denotes the Lagrange multipliers corresponding to the active indices. Since A is nonsingular, the matrix

$$\bar{C} = \begin{pmatrix} A & B \\ 0 & I(u_*) \end{pmatrix}$$

has full row rank. Therefore, the system (2.16) is uniquely solvable, provided the matrix M is positive definite on the null-space of \bar{C} .

If $M = 0$, then the QP reduces to an LP. In this case the solution of the optimization problem can be found in a vertex. The columns of $C = (A \mid B)$ corresponding to the positive components of the vertex (y_*, u_*) are linearly independent, see e.g. [3, § 2]. Since, by assumption, $y_* > 0$ and A is nonsingular, we can conclude that $u_* = 0$. Consequently, $I(u_*) \in \mathbb{R}^{n \times n}$ is the identity matrix. In the language of linear programming, y_* are the basis variables and u_* are the nonbasis variables. Thus, we have exactly m positive basis variables. This case is called the *nondegenerate case* in linear programming.

It has been observed, e.g. [6], [7], that in the nondegenerate case the KKT systems in barrier methods for linear programming can be preconditioned effectively. This will also be true in our case. If only bounds on u are active, efficient preconditioners can be constructed for the problems investigated in this paper. However, in our applications, ill-conditioning also arises from the matrices A . Although proven to be nonsingular, the matrices A arising in our applications have a wide spectrum which causes a large spread in the spectrum of the KKT matrix K . This will be investigated in more detail in Section 6.

2.3.3 Bound Constraints for u and y

Let y_*, u_* be a solution of the QP. Furthermore, let $\{l_1^u, \dots, l_{k_u}^u\}$ denote the set of active indices for u_* ,

$$\{l_1^u, \dots, l_{k_u}^u\} = \{i \mid (u_*)_i = 0\}$$

and let $\{l_1^y, \dots, l_{k_y}^y\}$ denote the set of active indices for y_* ,

$$\{l_1^y, \dots, l_{k_y}^y\} = \{i \mid (y_*)_i = 0\}.$$

In this case the Lagrange multipliers at the solution satisfy

$$(q_y)_i = 0, \quad i \notin \{l_1^y, \dots, l_{k_y}^y\} \quad \text{and} \quad (q_u)_i = 0, \quad i \notin \{l_1^u, \dots, l_{k_u}^u\}.$$

If we define the matrices $I(y_*) \in \mathbb{R}^{k_y \times m}$, $I(u_*) \in \mathbb{R}^{k_u \times n}$ by

$$(I(y_*))_{ij} = \begin{cases} 1 & \text{if } j = l_i^y, \\ 0 & \text{otherwise,} \end{cases} \quad \text{and} \quad (I(u_*))_{ij} = \begin{cases} 1 & \text{if } j = l_i^u, \\ 0 & \text{otherwise,} \end{cases}$$

then the KKT conditions (2.5) are equivalent to

$$\begin{pmatrix} M_{yy} & M_{yu} & A^T & I(y_*)^T & 0 \\ M_{uy} & M_{uu} & B^T & 0 & I(u_*)^T \\ A & B & 0 & 0 & 0 \\ I(y_*) & 0 & 0 & 0 & 0 \\ 0 & I(u_*) & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} y \\ u \\ p \\ q_y^a \\ q_u^a \end{pmatrix} = \begin{pmatrix} -c \\ -d \\ b \\ 0 \\ 0 \end{pmatrix}, \quad (2.17)$$

where q_y^a, q_u^a denote the Lagrange multipliers corresponding to the active indices.

The assumption that A is nonsingular is not sufficient to guarantee that the matrix

$$\widehat{C} = \begin{pmatrix} A & B \\ I(y_*) & 0 \\ 0 & I(u_*) \end{pmatrix} \quad (2.18)$$

has full row rank. If \widehat{C} does not have full rank, then the system (2.17) does not have a unique solution. In fact, in this case there exists $(\delta p, \delta q_y^a, \delta q_u^a) \neq (0, 0, 0)$ with

$$\begin{pmatrix} A^T & I(y_*)^T & 0 \\ B^T & 0 & I(u_*)^T \end{pmatrix} \begin{pmatrix} \delta p \\ \delta q_y^a \\ \delta q_u^a \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix},$$

and, thus,

$$\begin{pmatrix} M_{yy} & M_{yu} & A^T & I(y_*)^T & 0 \\ M_{uy} & M_{uu} & B^T & 0 & I(u_*)^T \\ A & B & 0 & 0 & 0 \\ I(y_*) & 0 & 0 & 0 & 0 \\ 0 & I(u_*) & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} y \\ u \\ p + t\delta p \\ q_y^a + t\delta q_y^a \\ q_u^a + t\delta q_u^a \end{pmatrix} = \begin{pmatrix} -c \\ -d \\ b \\ 0 \\ 0 \end{pmatrix} \quad \forall t \in \mathbb{R}.$$

Thus, in this case the Lagrange multipliers (p, q_y^a, q_u^a) are not uniquely determined.

If $M = 0$, then the QP reduces to an LP and the solution of the optimization problem can be found in a vertex (y_*, u_*) . In this case the columns of $C = (A \mid B)$ corresponding to the positive components of the vertex (y_*, u_*) are linearly independent. Thus, at most m components of (y_*, u_*) can be positive. If less than m components of (y_*, u_*) are positive the vertex is called *degenerate*, see e.g. [3, § 2].

In the nondegenerate case, i.e. if m components of (y_*, u_*) are positive, then we can find a column permutation Π for the matrix \widehat{C} defined in (2.18) such that

$$\widehat{C} \Pi = \begin{pmatrix} A & B \\ I(y_*) & 0 \\ 0 & I(u_*) \end{pmatrix} \Pi = \begin{pmatrix} C_B & C_N \\ 0 & I \end{pmatrix},$$

where $C_B \in \mathbb{R}^{m \times m}$ is the nonsingular basis matrix and $I \in \mathbb{R}^{n \times n}$ denotes the identity. This shows that in the nondegenerate case the matrix \widehat{C} has full row rank. In the degenerate

case, however, $l < m$ components of (y_*, u_*) are positive. Thus,

$$\widehat{C} = \begin{pmatrix} A & B \\ I(y_*) & 0 \\ 0 & I(u_*) \end{pmatrix} \in \mathbb{R}^{(m+(m+n-l)) \times (m+n)}$$

and $2m + n - l > m + n$. Hence, the matrix \widehat{C} cannot have full row rank.

This shows that the degenerate case occurs if and only if \widehat{C} does not have full row rank.

For the construction of preconditioners in barrier methods for linear programming the degenerate case is the difficult one. For example, the preconditioners discussed in [6], are far less effective in reducing the condition number of the KKT matrix in the degenerate case than they are in the nondegenerate case, cf. Tables 1 and 2 in [6]. Unfortunately, but not surprisingly this will also be the case in our situation. If bounds are only imposed on the controls u , efficient and rather general preconditioners can be derived. However, if state constraints, i.e. bounds on y , are present and active, then the design of preconditioners is much more difficult. We will investigate this in detail in Section 6.

Chapter 3

Eigenvalue Estimates

The eigenvalue distribution of the Karush–Kuhn–Tucker system is of great importance for the iterative solution methods we use. The convergence of MINRES and SYMMLQ depends mainly on the eigenvalue distribution of the system.

First we will study the numbers of positive, negative and zero eigenvalues of

$$K = \begin{pmatrix} H_y & 0 & A^T \\ 0 & H_u & B^T \\ A & B & 0 \end{pmatrix}, \quad (3.1)$$

then we estimate the eigenvalues of the entire system by the eigenvalues and singular values of the constituting submatrices.

We assume that A is invertible, and we use the notation

$$H = \begin{pmatrix} H_y & 0 \\ 0 & H_u \end{pmatrix}, \quad C = (A \mid B), \quad W = \begin{pmatrix} -A^{-1}B \\ I \end{pmatrix}.$$

3.1 The Eigenvalue Distribution

To find out about the eigenvalue distribution of K , we apply congruence transformations. From the decomposition

$$\begin{aligned} & \begin{pmatrix} I & 0 & 0 \\ -(A^{-1}B)^T & I & 0 \\ 0 & 0 & A^{-1} \end{pmatrix} \begin{pmatrix} H_y & 0 & A^T \\ 0 & H_u & B^T \\ A & B & 0 \end{pmatrix} \begin{pmatrix} I & -A^{-1}B & 0 \\ 0 & I & 0 \\ 0 & 0 & A^{-T} \end{pmatrix} \\ &= \begin{pmatrix} H_y & (H_y \mid 0) W & I \\ W^T \begin{pmatrix} H_y \\ 0 \end{pmatrix} & W^T H W & 0 \\ I & 0 & 0 \end{pmatrix} \end{aligned} \quad (3.2)$$

one can see immediately that K is invertible if and only if $W^T H W$ is invertible. Next we form

$$\begin{aligned}
& \begin{pmatrix} I & 0 & 0 \\ 0 & I & -W^T \begin{pmatrix} H_y \\ 0 \end{pmatrix} \\ 0 & 0 & I \end{pmatrix} \begin{pmatrix} H_y & (H_y|0) W & I \\ W^T (H_y|0) & W^T H W & 0 \\ I & 0 & 0 \end{pmatrix} \begin{pmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & -(H_y|0) W & I \end{pmatrix} \\
&= \begin{pmatrix} H_y & 0 & I \\ 0 & W^T H W & 0 \\ I & 0 & 0 \end{pmatrix} \tag{3.3}
\end{aligned}$$

and

$$\begin{aligned}
& \begin{pmatrix} I & 0 & -\frac{1}{2}H_y \\ 0 & 0 & I \\ 0 & I & 0 \end{pmatrix} \begin{pmatrix} H_y & 0 & I \\ 0 & W^T H W & 0 \\ I & 0 & 0 \end{pmatrix} \begin{pmatrix} I & 0 & 0 \\ 0 & 0 & I \\ -\frac{1}{2}H_y & I & 0 \end{pmatrix} \\
&= \begin{pmatrix} 0 & I & 0 \\ I & 0 & 0 \\ 0 & 0 & W^T H W \end{pmatrix}. \tag{3.4}
\end{aligned}$$

The $2m + n$ eigenvalues of the matrix on the right hand side of (3.4) are equal to the n eigenvalues of $W^T H W$ and to $+1$ and -1 , each with multiplicity m . Hence, by Sylvester's law of inertia, the matrix K has $m + n_+$ positive eigenvalues, $m + n_-$ negative eigenvalues and n_0 eigenvalues equal to zero, where n_+ , n_- , n_0 are the numbers of positive, negative and zero eigenvalues of $W^T H W$, respectively.

3.2 A Theorem by Rusten and Winther

If the matrix H is positive definite, then $W^T H W$ is positive definite and the system matrix K has $m + n$ positive eigenvalues and m negative eigenvalues. The theorem introduced in this section gives a more detailed description of the eigenvalue distribution of K .

Let the matrix H symmetric and positive definite and let $\mu_1 \geq \dots \geq \mu_{m+n} > 0$ be the eigenvalues of H . Then

$$\mu_{m+n} \|x\|^2 \leq \langle Hx, x \rangle \leq \mu_1 \|x\|^2 \quad \forall x \in \mathbb{R}^{m+n}. \tag{3.5}$$

Here and in the remainder of this section we use $\|\cdot\|$ to denote the 2-norm $\|x\|^2 = x^T x$.

We recall that for a matrix $C \in \mathbb{R}^{m \times (m+n)}$ the singular values $\sigma_1 \geq \dots \geq \sigma_m$ are the square roots of the eigenvalues of $C^T C$. We call the smallest singular value σ_m and the largest σ_1 . If C is of full rank, it has singular values $\sigma_1 \geq \dots \geq \sigma_m > 0$. It holds that

$$\sigma_m \|y\| \leq \|C^T y\| \leq \sigma_1 \|y\| \quad \forall y \in \mathbb{R}^m. \tag{3.6}$$

and

$$\sigma_m \|x\| \leq \|Cx\| \leq \sigma_1 \|x\| \quad \forall x \in N(C)^\perp \quad (3.7)$$

Note that this implies that the right inequality holds for all $x \in \mathbb{R}^{m+n}$.

For a system matrix of this structure, the following result holds which is taken from [14]:

Theorem 3.2.1 *Let $\mu_1 \geq \mu_2 \geq \dots \geq \mu_{m+n} > 0$ be the eigenvalues of H , let $\sigma_1 \geq \dots \geq \sigma_m > 0$ be the singular values of C^T . The eigenvalues $\lambda_1 \geq \dots \geq \lambda_{m+n} > 0 > \lambda_{m+n+1} \geq \dots \geq \lambda_{2m+n}$ of K satisfy*

$$\lambda_{2m+n} \geq \frac{1}{2}(\mu_{m+n} - \sqrt{\mu_{m+n}^2 + 4\sigma_1^2}), \quad (3.8)$$

$$\lambda_{m+n+1} \leq \frac{1}{2}(\mu_1 - \sqrt{\mu_1^2 + 4\sigma_m^2}), \quad (3.9)$$

$$\lambda_{m+n} \geq \mu_{m+n}, \quad (3.10)$$

$$\lambda_1 \leq \frac{1}{2}(\mu_1 + \sqrt{\mu_1^2 + 4\sigma_1^2}). \quad (3.11)$$

Proof: Let $\lambda \in \Lambda(K)$ and let (x, y) be the corresponding eigenvector, i.e.

$$Hx + C^T y = \lambda x, \quad (3.12)$$

$$Cx = \lambda y. \quad (3.13)$$

Note that if $x = 0$, then $y = 0$ by (3.13). This is not admissible for an eigenvector (x, y) . Hence $x \neq 0$.

1. First we bound the positive eigenvalues. Let λ be a positive eigenvalue of K . Combining the inner product of (3.12) with x and the inner product of (3.13) with y yields

$$x^T Hx + \lambda \|y\|^2 = \lambda \|x\|^2.$$

Since by (3.5)

$$\mu_{m+n} \|x\|^2 \leq x^T Hx = \lambda \|x\|^2 - \lambda \|y\|^2,$$

we have

$$(\mu_{m+n} - \lambda) \|x\|^2 \leq -\lambda \|y\|^2 \leq 0,$$

and thus

$$\lambda \geq \mu_{m+n}.$$

To derive an upper bound, use (3.13) in the form $y = \frac{1}{\lambda} Cx$ and the inner product of (3.12) with x to obtain

$$x^T Hx + \frac{1}{\lambda} x^T C^T Cx = \lambda \|x\|^2.$$

Using (3.5) and (3.7) we derive the inequality

$$(\lambda^2 - \mu_1\lambda - \sigma_1^2)\|x\|^2 \leq 0.$$

The roots of $\lambda^2 - \mu_1\lambda - \sigma_1 = 0$ are $\frac{1}{2}(\mu_1 \pm \sqrt{\mu_1^2 + 4\sigma_1^2})$.

Since $\|x\|^2$ is positive, we conclude

$$\lambda \leq \frac{1}{2}(\mu_1 + \sqrt{\mu_1^2 + 4\sigma_1^2}).$$

2. Now consider a negative eigenvalue λ of K .

The derivation of a lower bound for the negative eigenvalues is similar to the derivation of the upper bound for the positive eigenvalues.

To derive an upper bound for the negative eigenvalues, let $x = v + w$, where $v \in N(C)^\perp$ and $w \in N(C)$. Taking the inner product of (3.12) with v and substituting y from (3.13) into this expression we get, using orthogonality,

$$v^T H w = -v^T H v - \frac{1}{\lambda} \|Cv\|^2 + \lambda \|v\|^2.$$

Using the bounds (3.5) and (3.7) we obtain

$$v^T H w \geq (\lambda - \mu_1 - \frac{1}{\lambda} \sigma_m^2) \|v\|^2. \quad (3.14)$$

To proceed, we must find a bound for $v^T H w$. This is achieved by taking the inner product of (3.12) with w and using (3.5). Since $w \in N(C)^\perp$ we get

$$w^T H x + w^T C^T y = w^T H v + w^T H w = \lambda w^T x = \lambda w^T w.$$

Thus with $\lambda - \mu_{m+n} < 0$ this implies

$$w^T H v \leq (\lambda - \mu_{m+n}) \|w\|^2 \leq 0.$$

Together with (3.14) and the symmetry of H we obtain

$$0 \geq (\lambda - \mu_1 - \frac{\sigma_m^2}{\lambda}) \|v\|^2, \quad \text{and thus} \quad 0 \leq (\lambda^2 - \mu_1\lambda - \sigma_m^2) \|v\|^2.$$

If $v = 0$, (3.13) implies $y = 0$ and (3.12) reduces to $Hw = \lambda w$. Since λ is negative and H positive definite, this is a contradiction. It follows that $(\lambda^2 - \mu_1\lambda - \sigma_m^2) \geq 0$, leading to the final estimate

$$\lambda \leq \frac{1}{2}(\mu_1 - \sqrt{\mu_1^2 + 4\sigma_m^2}).$$

□

Chapter 4

SYMMLQ AND MINRES

4.1 Introduction to SYMMLQ and MINRES

The Karush–Kuhn–Tucker systems we want to solve are of the form

$$\begin{pmatrix} H_y & 0 & A^T \\ 0 & H_u & B^T \\ A & B & 0 \end{pmatrix} \begin{pmatrix} y \\ u \\ p \end{pmatrix} = \begin{pmatrix} c \\ d \\ b \end{pmatrix}, \quad (4.1)$$

where H_y and H_u are symmetric.

In our applications, the system matrix is very large and sparse. Usually we only compute the blocks and do not really assemble them into an entire system matrix. Therefore we want to use iterative solvers that only require matrix–vector products. Moreover, the matrices in our applications are not always explicitly known. Only their action on vectors can be computed, so that an iterative approach may even be the only appropriate way to handle the arising systems.

An effective and popular method for symmetric positive definite systems is the conjugate gradient method. However, since the system matrix in (4.1) is symmetric indefinite, the conjugate gradient method is not applicable. For symmetric indefinite systems Paige and Saunders [13] have derived two iterative methods, MINRES and SYMMLQ, which can be viewed as generalizations of the conjugate gradient method for the solution of indefinite systems. These methods will be introduced and analyzed in this chapter.

In this chapter we do not use the notation (4.1), but the notation generally used for linear systems. Instead of (4.1) we consider

$$Ax = b, \quad (4.2)$$

where $A \in \mathbb{R}^{n \times n}$ is symmetric indefinite.

We use the notation

$$\|x\| = \|x\|_2 = \sqrt{x^T x} \quad \text{and} \quad \langle x, y \rangle = x^T y.$$

The presentation in this chapter closely follows [9].

4.2 Derivation of SYMMLQ and MINRES

An iterative method for the solution of symmetric positive definite linear systems, suitable for large and sparse problems

$$Ax = b \tag{4.3}$$

is the conjugate gradient method. MINRES and SYMMLQ are generalizations of the conjugate gradient method for the symmetric indefinite case. Because SYMMLQ and MINRES are closely related to the conjugate gradient method, we give a brief introduction to the conjugate gradient method.

The derivation of the conjugate gradient method can be based on the fact that for positive definite matrices A the problem (4.3) is equivalent to the minimization problem

$$\min_{x \in \mathbb{R}^n} F(x) = \frac{1}{2} \langle x, Ax \rangle - \langle x, b \rangle. \tag{4.4}$$

In the positive definite case, the minimization problem has a unique solution. The minimizer is $x = A^{-1}b$. This makes (4.4) and (4.3) interchangeable.

By construction of the method, the conjugate gradient method really solves

$$Ax = r_0, \tag{4.5}$$

where r_0 is the initial residual $r_0 = b - Ax_0$. We do not want to assume that our starting vector is $x_0 = 0$. If $x_0 \neq 0$ is a more appropriate initial guess, we apply the conjugate gradient method to (4.3) with b replaced by $r_0 = b - Ax_0$. Equally, we write (4.4) in the form

$$\min_{x \in \mathbb{R}^n} F(x) = \frac{1}{2} \langle x, Ax \rangle - \langle x, r_0 \rangle. \tag{4.6}$$

The starting point in the derivation of the conjugate gradient method is to consider how one might go about minimizing the functional F in (4.6). A classical approach is the steepest descent or gradient method. Because of symmetry of A , the gradient of F is given by

$$\nabla F(x) = Ax - r_0.$$

If the gradient is nonzero there exists a positive scalar α such that

$$F(x - \alpha \nabla F(x)) < F(x).$$

We now adopt an iterative point of view. Given the iterate x_j in step j we take the direction of the negative gradient to get to the next iterate

$$x_{j+1} = x_j - \alpha_j \nabla F(x_j) = x_j - \alpha_j (Ax_j - r_0) = x_j - \alpha_j r_j,$$

where $r_j = Ax_j - r_0$ denotes the residual in iteration j . We choose the step size α such that F is minimized along the step. The solution is

$$\alpha_j = \frac{\langle r_j, Ax_j - r_0 \rangle}{\langle r_j, Ar_j \rangle}.$$

With this choice, the 'exact step size', a drawback of the gradient method becomes apparent. It holds that successive gradients and steps are orthogonal. This usually leads to unsatisfactory convergence behavior of the gradient method. The reason is that we compute a new iterate that is optimal with respect to the search direction r_j , but not with respect to the previously used search directions.

Nevertheless, we do not totally discard this choice of a search direction. Our goal now is to construct a method which preserves the optimality with respect to previously used directions and that effects improvement in the minimization problem (4.6).

We say that $x_j \in \text{span}\{p_0, \dots, p_j\}$ is *optimal* with respect to $\{p_0, \dots, p_j\}$ if

$$\langle Ax_j - r_0, v \rangle = 0 \quad \forall v \in \text{span}\{p_0, \dots, p_j\}. \quad (4.7)$$

If p_0, \dots, p_{j-1} are search directions from the previous iterations and if x_j is the current iterate, optimal with respect to $\text{span}\{p_0, \dots, p_{j-1}\}$, then we want to construct a new iterate x_{j+1} with the help of a new search direction p_j such that if we take the exact step $s_j = \alpha_j p_j$, the new iterate is optimal with respect to p_j and to $\text{span}\{p_0, \dots, p_{j-1}\}$, i.e. we want p_j, x_{j+1} such that

$$F(x_{j+1}) = \min_{x \in \text{span}\{p_0, \dots, p_{j-1}, p_j\}} F(x).$$

Using the optimality of x_j with respect to $\text{span}\{p_0, \dots, p_{j-1}\}$, we find that for $i = 0, \dots, j-1$

$$\begin{aligned} 0 &= \langle Ax_{j+1} - r_0, p_i \rangle \\ &= \langle A(x_j + \alpha_j p_j) - r_0, p_i \rangle \\ &= \langle Ax_j - r_0, p_i \rangle + \alpha_j \langle Ap_j, p_i \rangle \\ &= \alpha_j \langle Ap_j, p_i \rangle. \end{aligned} \quad (4.8)$$

The property $\langle Ap_j, p_i \rangle = 0, i \neq j$, is the A -orthogonality of p_j with respect to the directions p_0, \dots, p_{j-1} . So we have to search for a vector p_j that is A -orthogonal to p_0, \dots, p_{j-1} in order to get a new search direction.

If we are given A -orthogonal $p_i, i = 0, \dots, j-1$, and a current iterate x_j that is optimal with respect to the span of these search directions, then we can take the negative gradient as a descent direction and modify it in order to suffice the additional requirement. Using the Gram-Schmidt Orthogonalization we construct p_j from r_j in subtracting those components of r_j that are not A -orthogonal to the previous directions p_i .

Suppose we have $p_i, i = 0, \dots, j-1$, such that $\langle p_i, Ap_i \rangle \neq 0, i = 0, \dots, j-1$, and $\langle p_i, Ap_k \rangle = 0$ for $k \neq i, i, k = 0, \dots, j-1$, then

$$p_j = r_j - \sum_{i=0}^{j-1} \frac{\langle r_j, Ap_i \rangle}{\langle p_i, Ap_i \rangle} p_i$$

is A -orthogonal to p_0, \dots, p_{j-1} . One can show that

$$\text{span}\{p_0, \dots, p_i\} = \text{span}\{r_0, \dots, r_i\} = \text{span}\{r_0, Ar_0, \dots, A^i r_0\} = \mathcal{K}_{i+1}(A, r_0).$$

We have then $Ap_i \in \text{span}\{r_0, \dots, A^{i+1}r_0\} = \text{span}\{p_0, \dots, p_{i+1}\}$. Since x_j is optimal with respect to $\text{span}\{p_0, \dots, p_{i-1}\}$ we get $\langle r_j, Ap_i \rangle = 0$ for $i = 0, 1, \dots, i-2$. Hence

$$p_j = r_j - \frac{\langle r_j, Ap_{j-1} \rangle}{\langle p_{j-1}, Ap_{j-1} \rangle} p_{j-1}.$$

The requirement that x_{j+1} is optimal with respect to p_j yields

$$\alpha_j = \frac{\langle r_j, p_j \rangle}{\langle Ap_j, p_j \rangle} = \frac{\|r_j\|^2}{\langle Ap_j, p_j \rangle},$$

see (4.8) with $i = j$. The identity $\langle r_j, p_j \rangle = \|r_j\|^2$ follows from the construction of p_j .

In each step the conjugate gradient method computes the iterate x_k in the Krylov subspace

$$\mathcal{K}_k(A, r_0) = \text{span}\{r_0, Ar_0, \dots, A^{k-1}r_0\}$$

which minimizes F over $\mathcal{K}_k(A, r_0)$, i.e. the iterate x_k solves (4.6). Problem (4.6) is equivalent to (4.7), i.e. to solving

$$\langle Ax_k - r_0, v \rangle = 0 \quad \forall v \in \mathcal{K}_k(A, r_0). \quad (4.9)$$

The conjugate gradient method minimizes the error in the A -norm $\|\cdot\|_A$. This norm is for symmetric positive definite matrices A defined by $\|x\|_A = x^T Ax$. The minimization of $\|e\|_A$ follows from the identity

$$\begin{aligned} F(x_j) &= \min_{x \in \mathcal{K}_j(A, r_0)} F(x) \\ &= \min_{x \in \mathcal{K}_j(A, r_0)} \left\{ \frac{1}{2} \langle x, Ax \rangle - \langle x, r_0 \rangle \right\} \\ &= \min_{x \in \mathcal{K}_j(A, r_0)} \frac{1}{2} \left\{ \langle x, Ax \rangle - 2 \langle x, Ax_* \rangle + \langle x_*, Ax_* \rangle \right\} \\ &= \min_{x \in \mathcal{K}_j(A, r_0)} \frac{1}{2} \langle x_* - x, A(x_* - x) \rangle \\ &= \min_{x \in \mathcal{K}_j(A, r_0)} \frac{1}{2} \|e\|_A. \end{aligned} \quad (4.10)$$

If A is not positive definite, then (4.5) and (4.6) are not equivalent. In fact, (4.6) does not have a solution if A has negative eigenvalues, and it may not have a solution if A is only positive semidefinite. Even though in this case the foregoing derivation is not applicable, one can try to extend the conjugate gradients method by trying to compute the iterates x_k as a solution of (4.9). This leads to SYMMLQ.

SYMMLQ tries to compute the iterate $x_k \in \mathcal{K}_k(A, r_0)$ such that x_k solves

$$\langle r_0 - Ax_k, v \rangle = 0 \quad \forall v \in \mathcal{K}_k(A, r_0). \quad (4.11)$$

The vector $x_k \in \mathcal{K}_k(A, r_0)$ is called a *Galerkin approximation* to the solution x_* of $Ax = b$ over the Krylov subspace $\mathcal{K}_k(A, r_0)$.

Unfortunately, (4.11) need not have a solution. Consider the following example:

If

$$r_0 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad A = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix},$$

then

$$\mathcal{K}_1(A, r_0) = \text{span}\{r_0\} = \left\{ \begin{pmatrix} \alpha \\ 0 \end{pmatrix} : \alpha \in \mathbb{R} \right\}.$$

We have then with $x = (x_1, 0)^T, v = (v_1, 0)^T \in \mathcal{K}_1(A, r_0)$

$$\langle r_0 - Ax, r_0 \rangle = (1, -x_1)^T (v_1, 0) = v_1 \neq 0$$

in general, so that the Galerkin approximation problem does not have a solution.

This is the reason why SYMMLQ uses a slightly different iterate which is derived from the implementation of this method and will be discussed in detail in Section 4.4. If A is positive definite, then (4.9) has a unique solution, and SYMMLQ is equivalent to the method of conjugate gradients. In this case, SYMMLQ minimizes the error $\|e_j\|_A$ in each step.

An alternative is MINRES. It is based on another approximation to the exact solution. In iteration $k, k = 0, 1, \dots$, MINRES computes $x_k \in \mathcal{K}_k(A, r_0)$ such that x_k solves

$$\min_{x \in \mathcal{K}_k(A, r_0)} \|r_0 - Ax\|. \quad (4.12)$$

This definition of an approximation is motivated by the use of the residual $Ax_k - r_0$ as a measure for the closeness of the current iterate and the exact solution. The vector x_k is called a *minimum residual approximation* to the solution x_* of $Ax = r_0$. The least squares problem (4.12) always has a unique solution. This will be shown in Theorem 4.4.5.

The implementation of SYMMLQ and MINRES will be discussed in Section 4.4.

4.3 Convergence Analysis

MINRES and SYMMLQ are, like the conjugate gradient method, n -step procedures. Their finite termination will be established here. However, rounding errors may lead to a loss of orthogonality among theoretically orthogonal vectors and finite termination is not mathematically guaranteed. Moreover, when these iterative solvers are applied, n is usually so big that $O(n)$ iterations represent an unacceptable amount of work. As a consequence, it is customary to regard the methods as genuinely iterative techniques with termination based upon an iteration maximum and the residual norm. With this point of view, the rate of convergence becomes important.

The convergence of Krylov subspace methods is related to properties of uniform best approximating polynomials. This relation is based on the fact that vectors in Krylov subspaces have a special representation. This representation will now be derived. We need the following notation.

Let Π_k denote the space of all polynomials of degree k or less, and Π_k^1 the space of all polynomials of degree k or less that are one at the origin, i.e.

$$\Pi_k^1 = \{p \in \Pi_k \mid p(0) = 1\}.$$

Recall that we use the 2-norm, i.e. $\|\cdot\|$ always means $\|\cdot\|_2$.

Theorem 4.3.1 *Let $A \in \mathbb{R}^{n \times n}$ and let $v \in \mathbb{R}^n$. Let Π_k denote the space of polynomials of degree less or equal to k , then*

$$\mathcal{K}_k(A, v) = \{p(A)v \mid p \in \Pi_{k-1}\}.$$

Proof: Let $x \in \mathcal{K}_k(A, v)$. Then x is a linear combination of $\{v, Av, \dots, A^{k-1}v\}$, i.e. there are scalars $\alpha_i \in \mathbb{R}, i = 0, \dots, k-1$, such that

$$x = \alpha_0 v + \alpha_1 Av + \dots + \alpha_{k-1} A^{k-1} v = \sum_{i=0}^{k-1} \alpha_i A^i v = p(A)v$$

for the polynomial p defined by these coefficients. Clearly $p \in \Pi_{k-1}$. Conversely, if $x = p(A)v$ for a polynomial $p \in \Pi_{k-1}$, then x is an element of the Krylov subspace as a linear combination of the basis vectors $v, Av, \dots, A^{k-1}v$. \square

We consider Krylov subspace methods solving the problem

$$Ax = r_0$$

with $r_0 = Ax_0 - b$. However, we are interested in solutions x_* of the problems $Ax = b$. This corresponds to a variable transformation $x_k + x_0 \rightarrow x_k$. If x_k are the iterates generated by the Krylov subspace methods satisfying $x_k \in \mathcal{K}_k(A, r_0)$, then $x_k + x_0 \in x_0 + \mathcal{K}_k(A, r_0)$. We consider vectors x satisfying $x + x_0 \in x_0 + \mathcal{K}_k(A, r_0)$.

Theorem 4.3.1 establishes that vectors in the Krylov subspace have a special representation, and due to this representation we have that $x + x_0 = x_0 + p_{k-1}(A)r_0$ for some polynomial $p_{k-1} \in \Pi_{k-1}$ of degree less or equal to $k-1$. With this representation for x we can write the residual $r(x) = b - A(x + x_0)$ in the form

$$r(x) = b - A(x + x_0) = b - Ax_0 - Ax = r_0 - Ax = (I - Ap_{k-1}(A))r_0 = p_k^1(A)r_0. \quad (4.13)$$

The polynomial $p_k^1 = (1 - p_{k-1}(\cdot))$ is of degree less or equal to k and satisfies $p_k^1(0) = 1$. So we write $p_k^1 \in \Pi_k^1$. Similarly, the error $e(x) = x_* - x_0 - x$ can be written as

$$e(x) = x_* - x_0 - x = x_* - x_0 - p_{k-1}(A)r_0 = (I - p_{k-1}(A))r_0 = p_k^1(A)r_0. \quad (4.14)$$

The polynomial $p_k^1 = (1 - p_{k-1}(\cdot))$ in the representation of the residual is the same as in the representation of the error.

If we consider the iterates x_k , we write

$$r_k = r(x_k) = b - A(x_k + x_0)$$

and

$$e_k = e(x_k) = x_* - (x_k + x_0).$$

The conjugate gradient method and its generalizations, MINRES and SYMMLQ, iterate on Krylov subspaces of increasing dimension that eventually are invariant subspaces of the system matrix A . One of the features these methods have in common is the finite convergence. This feature, obvious by construction of the subspace and the linear independence of the basis vectors, will now be formally shown.

Theorem 4.3.2 *Let $A \in \mathbb{R}^{n \times n}$ be nonsingular. Then there exists a polynomial $p \in \Pi_{n-1}$ such that*

$$A^{-1} = p(A).$$

Proof: The Hamilton–Cayley Theorem says that a matrix annihilates its own characteristic polynomial, i.e. if $A \in \mathbb{R}^{n \times n}$ and if $p_A(\lambda) = \det(A - \lambda I)$ denotes the characteristic polynomial of A , then

$$p_A(A) = 0.$$

From this we can conclude the following: If $p_A(\lambda) = \sum_{i=0}^n a_i \lambda^i$, then $a_0 \neq 0$ if and only if A is nonsingular, i.e. if $\lambda = 0$ is not an eigenvalue of A . In this case, we find that

$$I = A(-a_0^{-1} \sum_{i=1}^n a_i A^{i-1}).$$

Hence,

$$A^{-1} = -a_0^{-1} \sum_{i=1}^n a_i A^{i-1}$$

and we know that there exists a polynomial p_{n-1} of degree less or equal to $n - 1$ such that the inverse of A can be written as a polynomial in A : $A^{-1} = p_{n-1}(A)$. In particular we obtain

$$A^{-1}r_0 = p_{n-1}(A)r_0.$$

□

If $A^k r_0 \in \mathcal{K}_k(A, r_0)$ for some k , then $A^l r_0 \in \mathcal{K}(A, r_0)$ for all $l \geq k$. This means that we have encountered an invariant subspace for A . This implication will be shown in Lemma 4.3.4. The solution to $Ax = r_0$ can be found in this subspace which is in the worst case encountered for $k = n$, in more favorable circumstances for $k \ll n$. From this we can conclude that there exists a polynomial p_{k-1} of degree less or equal to $k - 1$ such that

$$A^{-1}r_0 = p_{k-1}(A)r_0. \tag{4.15}$$

Before investigating more specialized results concerning the convergence of the minimum residual approximations, we state the result on the finite termination of Krylov minimum residual methods.

Theorem 4.3.3 *Let $A \in \mathbb{R}^{n \times n}$ be a nonsingular matrix. If x_k are minimum residual approximations of x_* on $\mathcal{K}_k(A, r_0)$, then there exists $k_* \leq n$ such that the residual $r_{k_*} = b - Ax_*$ satisfies*

$$\|r_{k_*}\| = 0.$$

Proof: From Theorem 4.3.2 we know that there exists a polynomial p_{k_*-1} of degree less or equal to $k_* - 1 \leq n - 1$, such that $x_* - x_0 = A^{-1}r_0 = p_{k_*-1}(A)r_0$ and so $r_0 - Ap_{k_*-1}(A)r_0 = 0$. Hence,

$$\|r_{k_*}\| = \min_{p \in \Pi_{k_*}^1} \|p(A)r_0\| \leq \|(I - Ap_{k_*-1}(A))r_0\| = 0.$$

□

Likewise, we can show finite convergence for the Galerkin approximations. The previous proof relies on the minimization property of the MINRES iterates. Because SYMMLQ does not minimize the residual, we have to use another approach.

In the following lemma it will be shown that Krylov subspaces of maximal dimension are invariant subspaces for the generating matrix. This means that if $A^k \in \mathcal{K}_k(A, v)$, then

$$A(\mathcal{K}_l(A, v)) \subset \mathcal{K}_k(A, v) \quad \forall l \geq k.$$

Lemma 4.3.4 *If $A^k v \in \mathcal{K}_k(A, v)$, then $A^l v \in \mathcal{K}_k v$ for all $l \geq k$.*

Proof: The proof can be done by induction. Here only the actual induction step

$$A^k v \in \mathcal{K}_k(A, v) \implies A^{k+1} v \in \mathcal{K}_k(A, v)$$

will be done. Since we know from Theorem 4.3.1 that $A^k v \in \mathcal{K}_k(A, v)$ if and only if it has a representation

$$A^k v = \sum_{j=0}^{k-1} \alpha_j A^j v,$$

for some scalars $\alpha_j \in \mathbb{R}$, we find by applying such a representation twice that

$$A^{k+1} v = AA^k v = \sum_{j=0}^{k-1} \alpha_j AA^j v = \sum_{j=0}^{k-2} \alpha_j A^{j+1} v + \alpha_{k-1} \sum_{j=0}^{k-1} \alpha_j A^j v \in \mathcal{K}_k(A, v).$$

□

Theorem 4.3.5 *Let $A \in \mathbb{R}^{n \times n}$ be a nonsingular matrix. Suppose that the Galerkin approximations x_k of x_* on $\mathcal{K}_k(A, r_0)$ exist. Then there exists $k_* \leq n$ such that the residual $r_{k_*} = r_0 - Ax_*$ satisfies*

$$r_{k_*} = 0.$$

Proof: For some $k \in \mathbb{N}$ the Krylov subspace $\mathcal{K}_k(A, r_0)$ is an invariant subspace for A , i.e.

$$\exists k \in \mathbb{N} : \quad A\mathcal{K}_k(A, r_0) \subset \mathcal{K}_k(A, r_0).$$

This is at least true for $k = n$, since $\mathcal{K}_i(A, r_0) \subset \mathbb{R}^n$ for all i . Suppose that $\mathcal{K}_{k_*}(A, r_0)$ is an invariant subspace for A and that $x_{k_*} \in \mathcal{K}_{k_*}(A, r_0)$ is the Galerkin approximation to the solution x_* of $Ax = r_0$. Then by optimality we have

$$\langle Ax_{k_*} - r_0, v \rangle = 0 \quad \forall v \in \mathcal{K}_{k_*}(A, r_0). \quad (4.16)$$

Because of the invariance of $\mathcal{K}_{k_*}(A, r_0)$ it holds

$$Ax_{k_*} - r_0 \in \mathcal{K}_{k_*}(A, r_0).$$

Using $v = Ax_{k_*} - r_0$ in (4.16) gives

$$\|Ax_{k_*} - r_0\|^2 = \|r_{k_*}\|^2 = 0.$$

□

4.3.1 Convergence Results for MINRES

The MINRES iterate x_k satisfies $\|r_0 - Ax_k\| = \min_{x \in \mathcal{K}_k(A, r_0)} \|r_0 - Ax\|$ and hence

$$\|r_k\| = \|r_0 - Ax_k\| = \|r_0 - Ap_{k-1}(A)r_0\|$$

for a polynomial p_{k-1} satisfying

$$\|r_k\| = \|r_0 - Ap_{k-1}(A)r_0\| = \min_{p \in \Pi_{k-1}} \|r_0 - Ap(A)r_0\|, \quad (4.17)$$

or, equivalently,

$$\|r_k\| = \|r_0 - Ax_k\| = \|r_0 - Ap_{k-1}(A)r_0\| = \|p_k^1(A)r_0\|$$

for a polynomial $p_k^1 \in \Pi_k^1$ satisfying

$$\|p_k^1(A)r_0\| = \min_{p \in \Pi_k^1} \|p(A)r_0\|.$$

Therefore we can use the norm of the residual $r_k = b - Ax_k$ to monitor the convergence of MINRES.

Our main interest are symmetric indefinite system matrices. In this situation we will from now on use the following notation. If $A \in \mathbb{R}^{n \times n}$ is nonsingular and symmetric indefinite, then all eigenvalues of A are contained in two intervals on the real line, one on the positive, one on the negative part. The spectrum is denoted by $\Lambda(A) = \Lambda$, and

$$\Lambda = [a, b] \cup [c, d] \quad \text{for } b < 0 < c$$

A set $E \subset \mathbb{R}$ with the property $E \supset \Lambda$ is called an inclusion set for the spectrum. Setting

$$\bar{\lambda} = \max_{\lambda \in \Lambda} |\lambda| \quad \text{and} \quad \underline{\lambda} = \min_{\lambda \in \Lambda} |\lambda|$$

we have $[a, b] \subset [-\bar{\lambda}, -\underline{\lambda}]$, $[c, d] \subset [\underline{\lambda}, \bar{\lambda}]$. So, for example, $E = [-\bar{\lambda}, -\underline{\lambda}] \cup [\underline{\lambda}, \bar{\lambda}]$ is an inclusion set. In addition to this, λ_i denotes the i -th largest eigenvalue, i.e.

$$\lambda_1 \geq \dots \geq \lambda_l > 0 > \lambda_{l+1} \geq \dots \geq \lambda_n.$$

Relation (4.17) and the minimization properties of MINRES imply the following result, see e. g. [15], [9], and for similar results for the conjugate gradient method see [1].

Theorem 4.3.6 *Let $A \in \mathbb{R}^{n \times n}$ be symmetric and $\Lambda = \{\lambda_1, \dots, \lambda_n\}$ denote its spectrum. If x_k are minimum residual approximations to the solution of $Ax = r_0$ on a Krylov sequence, then the following estimates hold for the corresponding residuals:*

$$\|r_k\| \leq \min_{p \in \Pi_k^1} \max_{i=1, \dots, n} |p(\lambda_i)| \|r_0\|, \quad (4.18)$$

$$\|r_k\| \leq \min_{p \in \Pi_2^k} \max_{i=1, \dots, n} |p(\lambda_i)| \|r_{k-2}\|. \quad (4.19)$$

Proof: For symmetric matrices A there exists a similarity transformation such that $A = V\Lambda V^T$ where V is orthonormal and Λ is a diagonal matrix that contains the eigenvalues of A .

Since V is orthonormal,

$$\begin{aligned} A^j &= AA \dots A \\ &= V\Lambda V^T V\Lambda V^T \dots V\Lambda V^T \\ &= V\Lambda\Lambda \dots \Lambda V^T \\ &= V\Lambda^j V^T \end{aligned}$$

for all $j \geq 0$. Thus $p(A) = Vp(\Lambda)V^T$ holds for every polynomial p .

Using a similarity transformation $A = V\Lambda V^T$ and the fact $p(A) = Vp(\Lambda)V^T$ yield the following estimates:

$$\begin{aligned}
\|r_k\| &= \|r_0 - Ax_k\| \\
&= \|p_k^1(A)r_0\| \\
&= \min_{p \in \Pi_k^1} \|p(A)r_0\| \\
&= \min_{p \in \Pi_k^1} \|Vp(\Lambda)V^T r_0\| \\
&= \min_{p \in \Pi_k^1} \|p(\Lambda)V^T r_0\| \\
&= \min_{p \in \Pi_k^1} \left(\sum_{i=1}^n (p(\lambda_i)v_i^T r_0)^2 \right)^{1/2} \\
&\leq \min_{p \in \Pi_k^1} \max_{i=1, \dots, n} |p(\lambda_i)| \left(\sum_{i=1}^n (v_i^T r_0)^2 \right)^{1/2} \\
&= \min_{p \in \Pi_k^1} \max_{i=1, \dots, n} |p(\lambda_i)| \|r_0\|.
\end{aligned}$$

The second assertion can be shown by considering

$$\begin{aligned}
\|r_k\| &= \|Ax_k - r_0\| \\
&= \|A(x_k - x_{k-2}) - r_{k-2}\| \\
&= \min_{x \in \mathcal{K}_k(A, r_0)} \|A(x - x_{k-2}) - r_{k-2}\| \\
&\leq \min_{x \in \mathcal{K}_2(A, r_{k-2})} \|Ax - r_{k-2}\|.
\end{aligned}$$

The inequality holds true because $r_{k-2} = r_0 - Ax_{k-2} \in \mathcal{K}_{k-1}(A, r_0)$ and so

$$x_k - x_{k-2} \in \mathcal{K}_k(A, r_0) \supset \mathcal{K}_2(A, r_{k-2}).$$

Using the same arguments as in the last part we then find

$$\begin{aligned}
\|r_k\| &\leq \min_{x \in \mathcal{K}_2(A, r_{k-2})} \|r_{k-2} - Ax\| \\
&\leq \min_{p \in \Pi_2^1} \max_{i=1, \dots, n} \|r_{k-2}\|.
\end{aligned}$$

□

As a direct implication of this theorem we will show that if A has only l distinct eigenvalues, then MINRES will terminate in l steps.

Theorem 4.3.7 *Let $A \in \mathbb{R}^{n \times n}$ be a nonsingular symmetric indefinite matrix with l distinct eigenvalues. If $x_k \in \mathcal{K}_k(A, r_0)$ are minimum residual approximations of x_* , then $\|r_l\| = 0$.*

Proof: Let $\Lambda = \{\lambda_1, \dots, \lambda_l\}$ be the set of eigenvalues of A . The eigenvalues of A are the roots of the polynomial

$$\hat{p}(x) = \prod_{j=1}^l (1 - x/\lambda_j),$$

which is of degree l . Since by (4.18) it holds

$$\|r_l\| \leq \min_{p \in \Pi_l^1} \max_{i=1, \dots, n} |p(\lambda_i)| \|r_0\| \leq \max_{i=1, \dots, n} |\hat{p}(\lambda_i)| \|r_0\| = \max_{i=1, \dots, n} \left| \prod_{j=1}^l (1 - \lambda_i/\lambda_j) \right| \|r_0\| = 0,$$

we have the desired result. \square

Theorem 4.3.7 shows that the iterative process will stop after l steps if the system matrix has l distinct eigenvalues. If this number l is small compared to the dimension of the system, we have a large computational gain. This result on its own already motivates preconditioning, which affects the eigenvalue distribution of the system matrix. Preconditioning will be introduced in Section 5.

The convergence analysis of minimum residual approximations is closely related to the Chebyshev approximation problem. We will therefore introduce briefly the Chebyshev approximation problem and Chebyshev Polynomials. This presentation relies on [9] and [1].

Chebyshev Polynomials can be written in different forms. Consider first the function

$$T_k(\cos \theta) = \cos(k\theta), \quad -\pi \leq \theta \leq \pi.$$

Using the variable transformation $x = \cos(\theta)$ we define for $k \in \mathbb{N}_0$ the k th Chebyshev Polynomial T_k by

$$T_k = \cos(k \arccos(x)), \quad x \in [-1, 1].$$

By the trigonometric identity

$$\cos((k+1)\theta) = 2 \cos(\theta) \cos(k\theta) - \cos((k-1)\theta)$$

we find that the Chebyshev Polynomials obey the three term recursion

$$T_0(x) = 1, \quad T_1(x) = x, \quad T_{k+1}(x) = 2xT_k(x) - T_{k-1}(x), \quad k = 2, 3, \dots \quad (4.20)$$

This representation justifies the notion 'polynomial'. Moreover, this recursion can be used to extend the Chebyshev polynomials onto the whole real line. For every fixed x , the recursion in (4.20) has a characteristic equation $\lambda^2 = 2x\lambda - 1$ whose roots are $\lambda = x \pm \sqrt{x^2 - 1}$. Using these and the initial values $T_0(x) = 1, T_1(x) = x$, one finds that the Chebyshev Polynomials are given by

$$T_k(x) = \frac{1}{2} \left((x + \sqrt{x^2 - 1})^k + (x - \sqrt{x^2 - 1})^k \right), \quad k = 0, 1, \dots \quad (4.21)$$

The problem

$$\min_{q \in \Pi_k^1} \max_{x \in I} |q(x)| \quad (4.22)$$

for some closed and bounded interval I on the positive real line is a Chebyshev approximation problem. For $b > a > 0$ the following result holds:

$$\max_{x \in [a, b]} |q_k^*(x)| = \min_{q \in \Pi_k^1} \max_{x \in [a, b]} |q(x)|,$$

where

$$q_k^*(x) = T_k \left(\frac{b+a-2x}{b-a} \right) / T_k \left(\frac{b+a}{b-a} \right). \quad (4.23)$$

The maximum is given by

$$\max_{x \in [a, b]} |q_k^*(x)| = \left(T_k \left(\frac{b+a}{b-a} \right) \right)^{-1}. \quad (4.24)$$

Here we require $b > a > 0$ because then we ascertain with $\frac{b+a}{b-a} > 1$ that the denominator in the definition of q_k^* , $T_k\left(\frac{b+a}{b-a}\right)$, is not zero. This follows because all roots of the Chebyshev Polynomial of order k , given by

$$x_i^0 = \cos \left(\frac{2i-1}{k} \frac{\pi}{2} \right), i = 1, \dots, k,$$

lie in $[-1, 1]$. Note that division by $T_k\left(\frac{b+a}{b-a}\right)$ in the definition of q_k^* normalizes it such that $q_k^* \in \Pi_k^1$.

Additionally, the following estimate of $T_k\left(\frac{b+a}{b-a}\right)$ which will be used in Theorems 4.3.8, 4.3.9 and 4.3.10 holds. The estimate follows directly from the formulation (4.21). For $k = 1$ it holds trivially

$$T_k \left(\frac{b+a}{b-a} \right) = \frac{b+a}{b-a}. \quad (4.25)$$

For $k > 1$ we have from (4.21)

$$\begin{aligned} T_k \left(\frac{b+a}{b-a} \right) &= \frac{1}{2} \left(\left(\frac{b+a}{b-a} + \frac{2\sqrt{ab}}{b-a} \right)^k + \left(\frac{b+a}{b-a} - \frac{2\sqrt{ab}}{b-a} \right)^k \right) \\ &= \frac{1}{2} \left(\left(\frac{(\sqrt{a} + \sqrt{b})^2}{b-a} \right)^k + \left(\frac{(\sqrt{a} - \sqrt{b})^2}{b-a} \right)^k \right) \\ &= \frac{1}{2} \left(\left(\frac{\sqrt{b/a} + 1}{\sqrt{b/a} - 1} \right)^k + \left(\frac{\sqrt{b/a} - 1}{\sqrt{b/a} + 1} \right)^k \right) \\ &\geq \frac{1}{2} \left(\frac{\sqrt{b/a} + 1}{\sqrt{b/a} - 1} \right)^k, \end{aligned} \quad (4.26)$$

where the last inequality comes from the estimate $c_1 + c_2 \geq \max\{c_1, c_2\}$ for positive real numbers c_1, c_2 .

With these tools we can now investigate convergence behavior of MINRES.

The following standard convergence estimate can be found for example in [15].

Theorem 4.3.8 *Let $A \in \mathbb{R}^{n \times n}$ be a nonsingular, symmetric indefinite matrix. If $x_k \in \mathcal{K}_k(A, r_0)$ are minimum residual approximations of x_* , then the residuals $r_k = r_0 - Ax_k$ obey*

$$\|r_k\| \leq 2 \left(\frac{\kappa - 1}{\kappa + 1} \right)^{\lfloor k/2 \rfloor} \|r_0\|,$$

where κ is the condition number of A given by $\kappa = \bar{\lambda}/\underline{\lambda}$. Here, $\underline{\lambda} = \min_{\lambda \in \Lambda} |\lambda|$, $\bar{\lambda} = \max_{\lambda \in \Lambda} |\lambda|$, and $\lfloor k/2 \rfloor$ denotes the largest integer less or equal to $k/2$.

Proof: Knowing the result for the Chebyshev approximation problem we want to apply it to the recursion (4.18) already derived. To do this, we map the set $[-\bar{\lambda}, -\underline{\lambda}] \cup [\underline{\lambda}, \bar{\lambda}]$, located on both sides of the origin, onto the interval $[\underline{\lambda}^2, \bar{\lambda}^2]$ on the positive part of the real line. This is admissible since for $p \in \Pi_{\lfloor k/2 \rfloor}^1$ the polynomial $p(\lambda^2)$ satisfies $p(\lambda^2) \in \Pi_k^1$.

This established we find

$$\begin{aligned} \|r_k\|/\|r_0\| &\leq \min_{p \in \Pi_k^1} \max_{\underline{\lambda} \leq |\lambda| \leq \bar{\lambda}} |p(\lambda)| \\ &\leq \min_{p \in \Pi_{\lfloor k/2 \rfloor}^1} \max_{\underline{\lambda} \leq |\lambda| \leq \bar{\lambda}} |p(\lambda^2)| \\ &\leq \min_{p \in \Pi_{\lfloor k/2 \rfloor}^1} \max_{\underline{\lambda}^2 \leq \lambda \leq \bar{\lambda}^2} |p(\lambda)| \\ &= \left(T_{\lfloor k/2 \rfloor} \left(\frac{\kappa^2 + 1}{\kappa^2 - 1} \right) \right)^{-1} \\ &\leq 2 \left(\frac{\kappa - 1}{\kappa + 1} \right)^{\lfloor k/2 \rfloor}. \end{aligned}$$

The estimate follows from (4.26). □

Considering the power $\lfloor k/2 \rfloor$ it is obvious that a decrease need not occur in every iteration. But it can be shown that a reduction in the residual is achieved at least after two iterations.

Theorem 4.3.9 *Let $A \in \mathbb{R}^{n \times n}$ be a nonsingular, symmetric indefinite matrix. If x_k are minimum residual approximations on $\mathcal{K}_k(A, r_0)$, then the residuals $r_k - r_0 = Ax_k$ obey*

$$\|r_k\| \leq \left(\frac{\kappa^2 - 1}{\kappa^2 + 1} \right) \|r_{k-2}\|,$$

where $\kappa = \bar{\lambda}/\underline{\lambda}$.

Proof: Analogously to the proof of Theorem 4.3.8 we find that, using (4.19), (4.24), and (4.25),

$$\begin{aligned}
\|r_k\|/\|r_{k-2}\| &\leq \min_{p \in \Pi_2^1} \max_{\underline{\lambda} \leq \lambda \leq \bar{\lambda}} |p(\lambda_i)| \\
&\leq \min_{p \in \Pi_1^1} \max_{\underline{\lambda} \leq \lambda \leq \bar{\lambda}} |p(\lambda_i^2)| \\
&\leq \min_{p \in \Pi_1^1} \max_{\underline{\lambda}^2 \leq \lambda \leq \bar{\lambda}^2} |p(\lambda_i)| \\
&= \left(T_1 \left(\frac{\kappa^2 + 1}{\kappa^2 - 1} \right) \right)^{-1} \\
&\leq \frac{\kappa^2 - 1}{\kappa^2 + 1}.
\end{aligned}$$

□

An assumption implicitly underlying Theorem 4.3.8 is that the intervals containing the eigenvalues of A are of equal size and that they have the same distance from the origin:

$$[a, b] \subset [-\bar{\lambda}, -\underline{\lambda}], [c, d] \subset [\underline{\lambda}, \bar{\lambda}].$$

If this is the case and if the eigenvalues are equally distributed, then the theorem gives a good description of the convergence behavior of MINRES. However, the distribution and the clustering of the eigenvalues will be important for the convergence of the method. If there are few well separated clusters of eigenvalues, then the prediction will be pessimistic, and sharper results can actually be derived.

If there are few negative eigenvalues, then the following result is of interest:

Theorem 4.3.10 *Let $A \in \mathbb{R}^{n \times n}$ be a nonsingular, symmetric indefinite matrix with eigenvalues*

$$\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_l > 0 > \lambda_{l+1} \geq \dots \geq \lambda_n.$$

If x_k are minimum residual approximations of x_ on $\mathcal{K}_k(A, r_0)$, then*

$$\|r_{k+n-l}\| \leq 2 \left(\prod_{i=l+1}^n \frac{\lambda_1 - \lambda_i}{|\lambda_i|} \right) \left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1} \right)^k \|r_0\|$$

for $k \geq 0$, where $\kappa = \frac{\lambda_1}{\lambda_l}$.

Proof: Consider the polynomial

$$q_{k+n-l}(x) = \prod_{i=l+1}^n (1 - x/\lambda_i) q_k^*(x),$$

where $q_k^*(x)$ is defined as in (4.23). Then $q_{k+n-l} \in \Pi_{k+n-l}^1$, and for $i \in \{l+1, \dots, n\}$ it holds that $q_{k+n-l}(\lambda_i) = 0$. Moreover, for all $i \in \{l+1, \dots, n\}$ and $j \in \{1, \dots, l\}$ we have the inequality

$$|1 - \lambda_j/\lambda_i| = |\lambda_i - \lambda_j|/|\lambda_i| \leq (\lambda_1 - \lambda_i)/|\lambda_i|.$$

Hence,

$$\begin{aligned} \|r_{k+n-l}\| &\leq \min_{p \in \Pi_{k+n-l}^1} \|p(A)\| \|r_0\| \\ &\leq \max_{j=1, \dots, n} |q_{k+n-l}(\lambda_j)| \|r_0\| \\ &= \max_{j=1, \dots, l} |q_{k+n-l}(\lambda_j)| \|r_0\| \\ &\leq \prod_{i=l+1}^n \frac{\lambda_1 - \lambda_i}{|\lambda_i|} \max_{j=1, \dots, l} |q_k^*(\lambda_j)| \|r_0\|. \end{aligned}$$

The estimate is a simple consequence of the construction of q_{n+k-l} . From the last expression we get immediately the assertion using (4.24) and (4.26). \square

This result is of interest, if, for one, there are only few negative eigenvalues, so that the estimate can be established after a small number of iterations, and if secondly the negative eigenvalues are not too small, because otherwise the factor $\prod(\lambda_1 - \lambda_i)/|\lambda_i|$ will be large.

These situations do in general not occur in our applications. Nevertheless, we have included this result because it gives the idea how one might go about isolating some eigenvalues in order to establish more refined convergence results than that in Theorem 4.3.8.

One situation we have found useful to look at is the case where there is one cluster of large eigenvalues and another cluster of eigenvalues of moderate size on the positive real line, and essentially the same distribution on the negative side of the origin. In this case the following two results hold. They are generalizations of Theorem 4.3.10.

Theorem 4.3.11 *Let $A \in \mathbb{R}^{n \times n}$ be a nonsingular, symmetric indefinite matrix with eigenvalues*

$$\begin{aligned} \lambda_1 \geq \dots \geq \lambda_{l_1} \gg \lambda_{l_1+1} \geq \dots \geq \lambda_{l_1+l_2} > 0, \\ 0 > \lambda_{l_1+l_2+1} \geq \dots \geq \lambda_{l_1+l_2+l_3} \gg \lambda_{l_1+l_2+l_3+1} \geq \dots \geq \lambda_n. \end{aligned}$$

Let

$$\begin{aligned} I_1 &= \{1, \dots, l_1\}, \\ I_2 &= \{l_1 + 1, \dots, l_1 + l_2\}, \\ I_3 &= \{l_1 + l_2 + 1, \dots, l_1 + l_2 + l_3\}, \\ I_4 &= \{l_1 + l_2 + l_3 + 1, \dots, l_1 + l_2 + l_3 + l_4\} \\ &= \{l_1 + l_2 + l_3 + 1, \dots, n\} \quad \text{and} \\ I &= I_1 \cup I_2 \cup I_3 \cup I_4. \end{aligned}$$

If x_k are minimum residual approximations of x_* on $\mathcal{K}_k(A, r_0)$, then

$$\|r_{k+l_1+l_4}\| \leq 2 \left(\prod_{i \in I_1} \frac{\lambda_i - \lambda_{l_1+l_2+l_3}}{\lambda_i} \right) \left(\prod_{i \in I_4} \frac{\lambda_{l_1+1} - \lambda_i}{-\lambda_i} \right) \left(\frac{\kappa - 1}{\kappa + 1} \right)^{\lfloor k/2 \rfloor} \|r_0\|$$

for $k \geq 0$, where $\kappa = \frac{\max_{j \in I_2 \cup I_3} |\lambda_j|}{\min_{j \in I_2 \cup I_3} |\lambda_j|}$.

Analogously, it holds

$$\|r_{k+l_2+l_3}\| \leq 2 \left(\prod_{i \in I_2} \frac{\lambda_i - \lambda_n}{\lambda_i} \right) \left(\prod_{i \in I_3} \frac{\lambda_1 - \lambda_i}{-\lambda_i} \right) \left(\frac{\kappa - 1}{\kappa + 1} \right)^{\lfloor k/2 \rfloor} \|r_0\|$$

for $k \geq 0$, where $\kappa = \frac{\max_{j \in I_1 \cup I_4} |\lambda_j|}{\min_{j \in I_1 \cup I_4} |\lambda_j|}$.

Proof: Consider the polynomial

$$q_{l_1+l_4}(x) = \prod_{i \in I_1 \cup I_4} (1 - x/\lambda_i) q_{\lfloor k/2 \rfloor}^*(x),$$

where $q_{\lfloor k/2 \rfloor}^*(x)$ is defined as in (4.23) with k replaced by $\lfloor k/2 \rfloor$. Then

$$q_{l_1+l_4} \in \Pi_{2\lfloor k/2 \rfloor + l_1 + l_4}^1,$$

and for $i \in I_1 \cup I_4$ it holds that $q_{l_1+l_4}(\lambda_i) = 0$. Moreover, for all $i \in I_1$ and $j \in I_2 \cup I_3$ we have the inequality

$$\left| 1 - \frac{\lambda_j}{\lambda_i} \right| = \left| \frac{\lambda_i - \lambda_j}{\lambda_i} \right| \leq \frac{\lambda_i - \lambda_{l_1+l_2+l_3}}{\lambda_i}, \quad (4.27)$$

and for $i \in I_4$, $j \in I_2 \cup I_3$ it holds

$$\left| 1 - \frac{\lambda_j}{\lambda_i} \right| \leq \frac{-\lambda_i + \lambda_{l_1+1}}{-\lambda_i}. \quad (4.28)$$

Hence,

$$\begin{aligned} \|r_{k+l_1+l_4}\| &\leq \min_{p \in \Pi_{k+l_1+l_4}^1} \|p(A)\| \|r_0\| \\ &\leq \min_{p \in \Pi_{k+l_1+l_4}^1} \max_{j \in I} |p(\lambda_j)| \|r_0\| \\ &\leq \min_{p \in \Pi_{2\lfloor k/2 \rfloor + l_1 + l_4}^1} \max_{j \in I} |p(\lambda_j)| \|r_0\| \\ &\leq \max_{j \in I} |q_{l_1+l_4}(\lambda_j)| \|r_0\| \\ &\leq \max_{j \in I_2 \cup I_3} |q_{l_1+l_4}(\lambda_j)| \|r_0\| \\ &\leq \left(\prod_{i \in I_1} \frac{\lambda_i - \lambda_{l_1+l_2+l_3}}{\lambda_i} \right) \left(\prod_{i \in I_4} \frac{-\lambda_i + \lambda_{l_1+1}}{-\lambda_i} \right) 2 \left(\frac{\kappa - 1}{\kappa + 1} \right)^{\lfloor k/2 \rfloor} \|r_0\|. \end{aligned}$$

The estimate is a consequence of the construction of $q_{l_1+l_4}$, following from (4.27), (4.28) and the estimates (4.24) and (4.26) for the Chebyshev approximation problem. The second assertion can be shown by essentially the same arguments. \square

A special case of the situation analyzed in the previous theorem occurs in our applications. We encountered a distribution of eigenvalues where eigenvalues of moderate size were situated in two clusters around the origin, and another cluster of large eigenvalues lay on the positive side of the origin. This situation can be analyzed as a special case of the situation described above with $l_4 = 0$.

Inclusion sets for the matrices we are interested in are often of the form

$$E = [-d, -ch^2] \cup [ch^2, d]. \quad (4.29)$$

Typically, h denotes a mesh parameter of increasingly small size. In this case

$$\kappa = \frac{d}{ch^2} = O(h^{-2}).$$

Rewriting the convergence governing factor in the form

$$\frac{\kappa - 1}{\kappa + 1} = 1 - 2\frac{1}{\kappa + 1} = 1 - 2\left(\frac{1}{\kappa} - \frac{1}{\kappa^2 + \kappa}\right) = 1 - 2\frac{1}{\kappa} + O(h^4)$$

shows that convergence is determined by a factor

$$\gamma \leq 1 - 2h^2c/d + O(h^4)$$

for an inclusion set E of this form. It follows from the foregoing presentation (see (4.18 in particular) that

$$\frac{\|r_k\|}{\|r_0\|} \leq \min_{p \in \Pi_k^1} \max_{i=1, \dots, n} |p(\lambda_i)| := \gamma_k. \quad (4.30)$$

The factor

$$\gamma = \lim_{k \rightarrow \infty} \gamma_k^{1/k}$$

is called the asymptotic convergence rate.

If the eigenvalues of the indefinite matrix are not symmetric about the origin, but do depend on a mesh size parameter, then the following result by Wathen, Fischer and Silvester [16] is of interest:

Theorem 4.3.12 *Let $A \in \mathbb{R}^{n \times n}$ be a nonsingular, symmetric indefinite matrix with eigenvalues in the inclusion set*

$$E = E(h) := [-a, -bh] \cup [ch^2, d], \quad a, b, c, d, h > 0. \quad (4.31)$$

Then the asymptotic convergence rate γ can be estimated as follows:

$$\gamma \leq 1 - h^{3/2} \sqrt{bc/ad} + O(h^{5/2}).$$

This tells us that, although an asymmetric distribution of the eigenspectrum must in general be judged unfavorably, we still profit from having a dependence of the spectrum bounds on a lower power of the small parameter h than in the symmetric case in (4.29).

4.3.2 Convergence Results for SYMMLQ

As we have seen in Section 4.2, if A is symmetric positive definite, one can use the function value of F to measure convergence of the Galerkin approximation. This is a point common to both SYMMLQ and the conjugate gradient method. However, since the conjugate gradient method can be applied only for positive definite matrices, whereas SYMMLQ works for indefinite matrices, too, where (4.4) has no solution, it is less clear how to measure progress in the indefinite case.

In the case of a positive definite system matrix, we can define the norm $\|\cdot\|_A$ by $\|x\|_A = \sqrt{x^T A x}$ and the corresponding scalar product $\langle x, x \rangle_A = x^T A x$. In Section 4.2 we have derived the conjugate gradient method, and we have seen that the conjugate gradient method minimizes the error $\|e\|_A$ in every iteration, cf. (4.10). Since the conjugate gradient iterates are the Galerkin approximations, we obtain for symmetric positive definite matrices A the estimate

$$\|e_k\|_A \leq \min_{p \in \Pi_k^1} \max_{i=1, \dots, n} |p(\lambda_i)| \|r_0\|_A,$$

corresponding to the result established for MINRES in (4.18). Since the estimates are exactly of the same type, the convergence results derived above immediately carry over.

However, if A is indefinite, it defines no norm and corresponding scalar product, and thus the initial estimate cannot be derived. Thus similar convergence results do not hold. This reflects the fact that the Galerkin approximation not necessarily exists in the indefinite case.

As for the minimum residual approximations, we still have finite convergence for the Galerkin approximations. This was already established in Theorem 4.3.5.

4.4 Implementation of SYMMLQ and MINRES

In Section 4.2 we have seen how the conjugate gradient method is derived and how this approach is motivated, namely by the minimization of the functional F given in (4.4). This approach is no longer appropriate for the extension on the indefinite case. However, one can still try to compute the approximation we relied on in the positive definite case in (4.9) or use the approximation (4.12). In addition to this, another point of view motivates the choice of Krylov subspaces.

Let \mathcal{X} be an invariant subspace of A . \mathcal{X} is an invariant subspace for A if and only if $AX = XB$ for some $B \in \mathbb{R}^{m \times m}$, where the m columns of $X \in \mathbb{R}^{n \times m}$ span \mathcal{X} . This means that the action of A on the m -dimensional subspace \mathcal{X} is completely determined by B . If $r_0 \in \mathcal{X}$, then $Ax = r_0$ can be solved in the following way: $r_0 = Xc$ holds for some $c \in \mathbb{R}^m$, so solve $By = c$ for $y \in \mathbb{R}^m$, and the solution is $x = Xy$.

Thus the problem of dimension $n \times n$ is reduced to an $m \times m$ system. This can result in quite a computational advantage. So the plan is to find the smallest invariant subspace containing r_0 .

As we have seen in Section 4.2, the conjugate gradient method constructs a new search direction in every step of the iteration. The search directions p_0, \dots, p_j span the Krylov subspace of order j . Similarly, MINRES and SYMMLQ iterate on Krylov subspaces of increasing dimension. To proceed, we need to express the vectors in the Krylov subspace of order j in terms of a basis for the subspace of order $j + 1$, i.e. to perform a change of basis.

We have the following transition from the basis for the j - dimensional subspace to the basis of the $j + 1$ -dimensional:

$$\begin{aligned}
AK_j &= [Ar_0, A^2r_0, \dots, A^j r_0] \\
&= [r_0, Ar_0, A^2r_0, \dots, A^j r_0] \begin{pmatrix} 0 & \dots & \dots & 0 \\ 1 & \ddots & & \vdots \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & 1 \end{pmatrix} \\
&= K_{j+1} \begin{pmatrix} 0 & \dots & \dots & 0 \\ 1 & \ddots & & \vdots \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & 1 \end{pmatrix} \tag{4.32}
\end{aligned}$$

$$(4.33)$$

Here we simply employed the natural basis, given by the columns of $K_j = [r_0, \dots, A^{j-1}r_0]$ for the Krylov subspace $\mathcal{K}_j(A, r_0)$ of order j . If we have orthogonal decompositions of the basis matrices K_j and K_{j+1} , i.e. $K_j = Q_j R_j$, $K_{j+1} = Q_{j+1} R_{j+1}$, where Q_j, Q_{j+1} are orthogonal and R_j, R_{j+1} upper triangular, we see the following.

$$\begin{aligned}
AQ_j R_j &= Q_{j+1} R_{j+1} \begin{pmatrix} 0^T \\ I_k \end{pmatrix} \\
AQ_j &= Q_{j+1} R_{j+1} \begin{pmatrix} 0^T \\ I_k \end{pmatrix} R_j^{-1} \\
&= Q_{j+1} \bar{H}_{j+1}, \tag{4.34}
\end{aligned}$$

where \bar{H}_{k+1} can be shown to be upper Hessenberg. A matrix $H \in \mathbb{R}^{n \times m}$ is called an upper Hessenberg matrix if $h_{ij} = 0$ for all (i, j) with $i > j + 1$. This means that H is upper diagonal with possibly additional entries in the lower subdiagonal.

The previous presentation in (4.34) indicates the implementation of another basis of the Krylov subspace $\mathcal{K}_i(A, r_0)$ of order i than the natural basis. In fact these vectors are

computationally near to linear dependence. Additionally, orthogonal bases often are of great computational advantage. Their usage will be introduced in Section 4.4.1.

Upper Hessenberg matrices play an important role in the successive construction of an invariant subspace. This becomes obvious in the following theorem.

Theorem 4.4.1 *Let $A \in \mathbb{R}^{n \times n}$ be symmetric and let v_1, \dots, v_{m+1} be linearly independent such that*

$$\text{span}\{v_1, \dots, v_i\} = \mathcal{K}_i(A, r_0), \quad i = 1, \dots, m+1.$$

There exists an upper Hessenberg matrix $\bar{T}_m \in \mathbb{R}^{(m+1) \times m}$ such that

$$AV_m = V_{m+1}\bar{T}_m, \tag{4.35}$$

where V_{m+1} denotes the matrix with columns v_1, \dots, v_{m+1} . The upper Hessenberg matrix \bar{T}_m is uniquely determined by A and v_1, \dots, v_{m+1} . If V_{m+1} is orthonormal, then

$$V_m^*AV_m = T_m, \tag{4.36}$$

where $T_m \in \mathbb{R}^{m \times m}$ is the matrix obtained from \bar{T}_m by deleting the last row. In particular, T_m and \bar{T}_m are tridiagonal.

Proof: Since $\text{span}\{v_1, \dots, v_i\} = \mathcal{K}_i(A, r_0)$, we obtain $Av_i \in \mathcal{K}_{i+1}(A, r_0)$. Moreover, since $\{v_1, \dots, v_{i+1}\}$ is a basis of $\mathcal{K}_{i+1}(A, r_0)$ there exist scalars t_{ij} , $j = 1, \dots, i+1$, such that

$$Av_j = \sum_{j=1}^{i+1} t_{ij}v_j.$$

The scalars t_{ij} , $j = 1, \dots, i+1$, are uniquely determined by Av_i and v_1, \dots, v_{i+1} . Setting

$$t_{ij} = 0, \quad \text{if } i > j+1,$$

and defining \bar{T}_m to be the matrix with entries t_{ij} we obtain (4.35). We can see from the construction that \bar{T}_m is an upper Hessenberg matrix. The equation (4.36) is an immediate result of (4.35) and the orthogonality of V_{m+1} . If A is symmetric, then $V_m^*AV_m$ is symmetric. Therefore, the upper Hessenberg matrix T_m in (4.36) has to be symmetric. This implies that \bar{T}_m has to be a tridiagonal matrix. \square

Theorem 4.4.2 *Let $A \in \mathbb{R}^{n \times n}$ and let $v \in \mathbb{R}^n$.*

1. $\mathcal{K}_i(QAQ^*, Qv) = Q\mathcal{K}_i(A, v)$ for unitary Q .
2. Let vectors v_1, \dots, v_m be given such that

$$\text{span}\{v_1, \dots, v_i\} = \mathcal{K}_i(A, v_1) \quad \forall i = 1, \dots, m,$$

then

$$\text{span}\{v_1, \dots, v_i, Av_i\} = \mathcal{K}_{i+1}(A, v_1) \quad \forall i = 1, \dots, m.$$

Proof:

1. This follows by the orthogonality of Q .
2. Since

$$\text{span}\{v_1, \dots, v_i\} = \text{span}\{v_1, Av_1, \dots, A^{i-1}v_1\},$$

we find that

$$\text{span}\{v_1, \dots, v_i, Av_i\} \subset \mathcal{K}_{i+1}(A, v)$$

On the other hand we have that

$$\mathcal{K}_{i+1}(A, v_1) = \mathcal{K}_i(A, v_1) \cup (A^i v_1)$$

and $\mathcal{K}_i(A, v_1) = \text{span}\{v_1, \dots, v_i\}$. Moreover,

$$A^i v_1 = AA^{i-1}v_1 \in A\mathcal{K}_i(A, v_1) = \text{span}\{Av_1, \dots, Av_i\},$$

and, since $\text{span}\{v_1, \dots, v_{i-1}\} = \mathcal{K}_{i-1}(A, v_1)$,

$$\begin{aligned} \text{span}\{Av_1, \dots, Av_{i-1}\} &= A\mathcal{K}_{i-1}(A, v_1) = \text{span}(Av_1, \dots, AA^{i-2}v_1) \\ &\subset \mathcal{K}_i(A, v_1) = \text{span}\{v_1, \dots, v_i\}. \end{aligned}$$

Hence $\mathcal{K}_{i-1}(A, v_1) \subset \text{span}\{v_1, \dots, v_i, Av_i\}$.

□

Part 2 of the preceding theorem is important for the numerical computation of a solution for the linear system $Ax = r_0$. MINRES and SYMMLQ successively construct a basis $\{v_1, \dots, v_i\}$ of $\mathcal{K}_i(A, r_0)$. Instead of effectively computing powers A^i and then taking the matrix vector product $A^i r_0$, a basis for $\mathcal{K}_{i+1}(A, r_0)$ can then be computed for the expense of one matrix - vector multiplication, namely Av_i , because $\mathcal{K}_{i+1}(A, v_1) = \{v_1, \dots, v_i, Av_i\}$. In Part 1 we see how a unitary transformation of A can be expressed in terms of the Krylov subspace generated by A .

We now turn to the issue of computing orthogonal bases for the subspace underlying the iterative process.

4.4.1 Orthogonal Bases for the Krylov Subspaces

Using $\text{span}\{r_0, \dots, A^{m-1}r_0\}$ as a representation for $\mathcal{K}_m(A, r_0)$, the Gram-Schmidt algorithm successively orthogonalizes the vectors $A^j r_0$ against the previously obtained orthogonal vectors v_i , $i = 1, \dots, j$. This is not done by computing $A^j r_0$. Instead of using $A^j r_0$ as the new column and orthogonalizing it against the orthonormal vectors v_1, \dots, v_j already obtained, Av_j is used for this process. This gives $K_{j+1} = (K_j, Av_j)$.

Since the classical Gram–Schmidt is numerically unstable (see e.g. [8], p.218), one often takes refuge to the modified Gram–Schmidt process, which is in this context known as the Arnoldi process.

Algorithm 4.4.3 (Arnoldi Process)

1. given r_0 and m
2. set $v_1 = r_0/\|r_0\|$
3. for $j = 1, \dots, m - 1$
 - 3.1. $\hat{v}_{j+1} = Av_j$
 - 3.2. for $i = 1, \dots, j$
 - 3.2.1. $t_{ij} = \langle \hat{v}_{j+1}, v_i \rangle$
 - 3.2.2. $\hat{v}_{j+1} = \hat{v}_{j+1} - t_{ij}v_i$
 - 3.3. $t_{j+1,j} = \|\hat{v}_{j+1}\|$
 - 3.4. if $t_{j+1,j} = 0$ stop
 - 3.5. $v_{j+1} = \hat{v}_{j+1}/t_{j+1,j}$

The Arnoldi process computes the entries t_{ij} , $j = 1, \dots, i + 1$, of a matrix that represents the change of basis as given in (4.32). If A is symmetric, and v_1, \dots, v_m are the vectors generated by Algorithm (4.4.3), then we obtain from Theorem 4.4.1 that

$$V_m^*AV_m = T_m$$

for a symmetric tridiagonal matrix $T_m \in \mathbb{R}^{m \times m}$. In particular we have that $t_{ij} = 0$ for $i < j - 1$. Therefore the j -th step of Algorithm (4.4.3) reduces to

$$\hat{v}_{j+1} = Av_j - \langle Av_j, v_{j-1} \rangle v_{j-1} - \langle Av_j, v_j \rangle v_j \tag{4.37}$$

$$v_{j+1} = \hat{v}_{j+1}/\|\hat{v}_{j+1}\|, \tag{4.38}$$

where we formally set $v_0 = 0$. This simplification shows that the new vector only has to be orthogonalized against the preceding two orthogonal basis vectors.

With $\delta_j = \|\hat{v}_j\|$, the orthogonality of v_i , $i = 1, \dots, j$, and the symmetry of A we obtain

$$\begin{aligned} \langle Av_j, v_{j-1} \rangle &= \langle v_j, Av_{j-1} \rangle \\ &= \langle v_j, Av_{j-1} - \langle Av_{j-1}, v_{j-2} \rangle v_{j-2} - \langle Av_{j-1}, v_{j-1} \rangle v_{j-1} \rangle \\ &= \langle v_j, \hat{v}_j \rangle = \frac{1}{\delta_j} \|\hat{v}_j\|^2 = \delta_j. \end{aligned}$$

Furthermore,

$$\langle Av_j, v_j \rangle = \langle Av_j - \delta_j v_{j-1}, v_j \rangle.$$

Using this representation, we obtain the so-called Lanczos Tridiagonalization:

Algorithm 4.4.4 (Lanczos Tridiagonalization)

1. given r_0 and m
2. set $\hat{v}_1 = r_0$, $v_0 = 0$ and $\delta_1 = \|\hat{v}_1\|$
3. for $j = 1, \dots, m - 1$
 - 3.1. if $\delta_j = 0$ stop
 - 3.2. $v_j = \hat{v}_j / \delta_j$
 - 3.3. $\hat{v}_{j+1} = Av_j - \delta_j v_{j-1}$
 - 3.4. $\gamma_j = \langle \hat{v}_{j+1}, v_j \rangle$
 - 3.5. $\hat{v}_{j+1} = \hat{v}_{j+1} - \gamma_j v_j$
 - 3.6. $\delta_{j+1} = \|\hat{v}_{j+1}\|$

Note that the process stops if $\delta_{j+1} = 0$. This means that the potential new basis vector $Av_j = \hat{v}_{j+1}$ is linearly dependent of the preceding basis vectors. It lies entirely in the direction of the preceding basis vectors, and so the orthogonalization process only leaves over the zero vector. In our context this means that an invariant subspace is encountered.

From Algorithm 4.4.4 one can see that the vectors v_1, \dots, v_{j+1} satisfy

$$V_j^* AV_j = T_j, \quad (4.39)$$

$$AV_j = V_{j+1} \bar{T}_j = V_j T_j + \delta_{j+1} v_{j+1} e_j^T, \quad (4.40)$$

where $V_j = [v_1, \dots, v_j]$, $V_{j+1} = [v_1, \dots, v_{j+1}]$ and T_j, \bar{T}_j are tridiagonal matrices as given in Theorem 4.4.1.

The matrix $T_j \in \mathbb{R}^{j \times j}$ is of the following form for all $j = 1, \dots, m$:

$$T_j = \text{tridiag}(\delta_j, \gamma_j, \delta_{j+1}) = \begin{pmatrix} \gamma_1 & \delta_2 & & & \\ \delta_2 & \gamma_2 & \delta_3 & & \\ & \ddots & \ddots & \ddots & \\ & & \delta_{j-1} & \gamma_{j-1} & \delta_j \\ & & & \delta_j & \gamma_j \end{pmatrix}, \quad (4.41)$$

and \bar{T}_j is obtained by deleting the last row of $\bar{T}_j \in \mathbb{R}^{(j+1) \times j}$, where

$$\bar{T}_j = \text{tridiag}(\delta_j, \gamma_j, \delta_{j+1}) = \begin{pmatrix} \gamma_1 & \delta_2 & & & \\ \delta_2 & \gamma_2 & \delta_3 & & \\ & \ddots & \ddots & \ddots & \\ & & \delta_{j-1} & \gamma_{j-1} & \delta_j \\ & & & \delta_j & \gamma_j \\ & & & & \delta_{j+1} \end{pmatrix}. \quad (4.42)$$

With the help of these matrices we now derive an alternative formulation for the problems we try to solve, i.e. for the approximations to the exact solution we seek.

Since $\text{span}\{v_1\} = \text{span}\{r_0\} = \mathcal{K}_1(A, r_0)$ we have

$$\beta v_1 = \beta V_m e_1 = r_0, \quad (4.43)$$

with $\beta = \|r_0\| \in \mathbb{R}$. If v_1, \dots, v_j are orthonormal for $j = 1, \dots, m$, then Theorem 4.4.1 and (4.43) yield that

$$\begin{aligned} 0 &= \langle Ax_j - r_0, v \rangle && \forall v \in \mathcal{K}_j(A, r_0) \\ &= \langle AV_j y_j - r_0, V_j y \rangle && \forall y \in \mathbb{R}^j \\ &= \langle AV_j y_j - V_j \beta e_1, V_j y \rangle && \forall y \in \mathbb{R}^j. \end{aligned}$$

This holds if and only if

$$\begin{aligned} 0 &= \langle V_j^* AV_j y_j - \beta e_1, y \rangle && \forall y \in \mathbb{R}^j \\ &= \langle T_j y_j - \beta e_1, y \rangle && \forall y \in \mathbb{R}^j. \end{aligned}$$

Thus the problem (4.11) is equivalent to solving

$$T_j y = \beta e_1 = \|r_0\| e_1. \quad (4.44)$$

Similarly, the problem formulation (4.12) for MINRES can be written as

$$\min_{y \in \mathbb{R}^j} \frac{1}{2} \|AV_j y - r_0\|^2 = \min_{y \in \mathbb{R}^j} \frac{1}{2} \|V_{j+1} \bar{T}_j y - r_0\|^2. \quad (4.45)$$

Since $r_0 = \beta V_{j+1} e_1$ and, if v_1, \dots, v_j for all $j = 1, \dots, m$ are orthonormal, $\|V_{j+1} y\| = \|y\|$ for all $y \in \mathbb{R}^{j+1}$, this leads to the following problem equivalent to (4.12):

$$\min_{y \in \mathbb{R}^j} \frac{1}{2} \|\bar{T}_j y - \beta e_1\|^2, \quad (4.46)$$

where $\beta = \|r_0\|$.

In Section 4.2 we have seen that a solution z_j to (4.44) may not exist if A is not positive definite. The least squares problem (4.46), however, is always uniquely solvable.

Theorem 4.4.5 *Suppose that A is nonsingular and that $\delta_2, \dots, \delta_j$ are nonzero. If $\delta_{j+1} \neq 0$, then*

$$\min_{y \in \mathbb{R}^j} \|\bar{T}_j y - \beta e_1\|^2 \quad (4.47)$$

has a unique solution $y_j \in \mathbb{R}^j$. If $\delta_{j+1} = 0$, then $x = V_j y_j$ solves $Ax = r_0$, where y_j solves (4.44) and, as a consequence, solves (4.47) with zero residual.

the 'natural' basis $\{r_0, Ar_0, \dots, A^{j-1}r_0\}$ to an orthonormal basis $\{v_1, \dots, v_j\}$ of $\mathcal{K}_j(A, r_0)$. Now we consider as the underlying basis the columns of $\bar{W}_j = V_j Q_j^T$. Storage of the full matrix \bar{W}_j can be prohibitive for large dimensional problems. Fortunately, it is not necessary to store all previous basis vectors and explicitly form $\bar{W}_j \bar{z}_j$. This will become obvious when a recursion for the iterates is derived.

The iterates $x_j = \bar{W}_j \bar{z}_j$ are the Galerkin approximations for the solution of $Ax = r_0$. Since we are interested in a solution of $Ax = b$ and since $r_0 = b - Ax_0$, the approximation for the solution x_* is given by $x_0 + x_j = \bar{W}_j \bar{z}_j$. The iterates $x_0 + x_j$ will be denoted by x_j , i.e. $x_j = x_0 + \bar{W}_j \bar{z}_j$.

We have seen that the matrix \bar{L}_j is singular if and only if $\bar{d}_j = 0$. In this case the Galerkin approximation may not exist, and it is certainly not unique. To overcome this problem, we change to slightly different iterates as we already indicated. In addition to \bar{L}_j we define $L_j \in \mathbb{R}^{j \times j}$ to be the matrix obtained from \bar{L}_j by replacing \bar{d}_j with d_j . The matrix L_j is the upper $j \times j$ - submatrix of \bar{L}_{j+1} . Moreover, we set

$$W_j = (w_1, \dots, w_{j-1}, w_j)$$

(W_j is obtained from \bar{W}_{j+1} by deleting the last column), and

$$z_j = (\zeta_1, \dots, \zeta_{j-1}, \zeta_j),$$

where z_j solves

$$L_j z_j = \|r_0\| e_1.$$

W_j is obtained from \bar{W}_{j+1} by deleting the last column. As it is suggested by the notation, the solution z_j of $L_j z_j = \|r_0\| e_1$ effectively differs from the solution \bar{z}_j of $\bar{L}_j \bar{z}_j = \|r_0\| e_1$ only in one component. The first $j-1$ components of \bar{z}_j and z_j are identical, and for the j -th component we obtain that $\zeta_j = c_{j+1} \bar{\zeta}_j$. This follows from the relation (4.48) between d_j and \bar{d}_j .

Setting

$$x_j^L = x_0 + W_j z_j,$$

the vectors x_j^L obey the recursion

$$x_0^L = x_0, \quad x_j^L = x_0 + W_{j-1} z_{j-1} + \zeta_j w_j = x_{j-1}^L + \zeta_j w_j, \quad j \geq 1.$$

This recursion shows that it is not necessary to store all the vectors w_j . Instead, the current iterate is obtained as a linear combination of the previous vector x_{j-1}^L and the latest basis vector. The vectors v_j, w_j can be formed, used and discarded one by one.

For the Galerkin approximation x_j we have

$$x_j = x_0 + \bar{W}_j \bar{z}_j = x_0 + W_{j-1} z_{j-1} + \bar{\zeta}_j \bar{w}_j = x_{j-1}^L + \bar{\zeta}_j \bar{w}_j.$$

This can be written as

$$x_j = x_j^L + \zeta_j (\bar{w}_j / c_{j+1} - w_j)$$

if we use the recursion for x_j^L and the relation $\zeta_j = c_{j+1}\bar{\zeta}_j$. The recursion (4.49) for \bar{w}_j , w_j yields

$$c_{j+1}w_j = c_{j+1}^2\bar{w}_j + c_{j+1}s_{j+1}v_{j+1} = c_{j+1}^2\bar{w}_j - s_{j+1}(\bar{w}_{j+1} - s_{j+1}\bar{w}_j) = \bar{w}_j - s_{j+1}\bar{w}_{j+1}$$

and thus we have the transition formula

$$x_j = x_j^L + (\zeta_j s_{j+1}/c_{j+1})\bar{w}_{j+1}. \quad (4.51)$$

Thus, one can use the recursion for x_j^L throughout the iteration and in the final step one can use this formula to compute x_j from x_j^L .

The iteration is terminated if the residual is small. The residual can be monitored during the iteration without knowing the current Galerkin approximation x_j because the following formula for the residual holds:

$$Ax_j - b = AV_j y_j - r_0 = V_j(T_j y_j - \|r_0\|e_1) + \delta_{j+1}v_{j+1}e_j^T y_j = \delta_{j+1}v_{j+1}y_j^{(j)},$$

where the vector y_j is the solution of (4.44) and $y_j^{(j)}$ denotes the j -th component of y_j . The vector y_j is not directly available, but its last component can be computed cheaply. From $T_j = T_j^T = Q_j^T \bar{L}_j^T$ we obtain that

$$\bar{L}_j^T y_j = \|r_0\|Q_j e_1.$$

Since $Q_j = G_{jj}^T \cdots G_{2j}^T$, the last row of the equation is given by

$$\bar{d}_j y_j^{(j)} = \|r_0\|s_2 \cdots s_j.$$

thus

$$\begin{aligned} -r_j &= Ax_j - r_0 \\ &= \delta_{j+1}v_{j+1}\|r_0\|s_2 \cdots s_j / \bar{d}_j \\ &= v_{j+1}\|r_0\|s_2 \cdots s_j s_{j+1}/c_{j+1}, \end{aligned} \quad (4.52)$$

and so

$$\|r_j\| = \|r_0\| |s_2 \cdots s_j s_{j+1}/c_{j+1}| = \|r_{j-1}\| |s_{j+1}c_j/c_{j+1}|. \quad (4.53)$$

If $\delta_{j_0} = 0$ for some j_0 , then the Krylov space is invariant, i.e. $\mathcal{K}_j(A, r_0) = \mathcal{K}_{j_0-1}(A, r_0)$ for all $j \geq j_0 - 1$, and the solution is found. We are then in the situation $AV_m = V_m T_m$. This is also shown by the formula for the residual: If $\delta_{j_0} = 0$, then $r_{j_0-1} = 0$, and x_{j_0-1} solves the system.

The formulas (4.51), (4.52) and (4.53) are valid if and only if the Galerkin approximations exist, i.e. if $\bar{d}_j \neq 0$. Because of the relation (4.48) this is equivalent to $c_{j+1} \neq 0$.

Although the Galerkin approximation may not exist, the approximations x_j^L always exist. Therefore we compute these instead of the vectors x_j throughout the iteration and use (4.51) to compute the Galerkin approximation at the end. To show that the method introduced above is well-defined we need to show that L_j is nonsingular.

Lemma 4.4.6 *Let $A \in \mathbb{R}^{n \times n}$ be nonsingular, and let j_0 be such that $\delta_{j_0+1} = 0$, $\delta_1, \dots, \delta_{j_0} \neq 0$. Then L_j is nonsingular for $j = 1, \dots, j_0$.*

Proof: Let $j \leq j_0$ be the first index such that L_j is singular. Then $d_1, \dots, d_{j-1} \neq 0$, and $d_j = 0$. This means that $\bar{d}_j = \delta_{j+1} = 0$. With (4.39) this yields

$$AV_j = V_j T_j.$$

Thus, since A is nonsingular and V_j is orthogonal, T_j is nonsingular. If $\delta_{j+1} = 0$, then $L_j = \bar{L}_j = T_j Q_j^T$. This shows that L_j cannot be singular. \square

If $\delta_{j_0} = 0$, then the exact solution of $Ax = b$ is found. In this case $L_{j_0} = \bar{L}_{j_0}$ and $x_{j_0} = x_{j_0}^L$. In theory, the iteration stops with $\delta_{j+1} = 0$, and then the iterates that are actually computed, the Galerkin approximation and the exact solution coincide. However, the stopping criterion in practice is a small residual.

The foregoing presentation leads to the following implementation of the algorithm.

Algorithm 4.4.7 (SYMMLQ)

1. given $A \in \mathbb{R}^{n \times n}$ symmetric, $b \in \mathbb{R}^n$, $x_0 \in \mathbb{R}^n$
2. compute $r_0 = b - Ax_0$, set
 - 2.1. $\hat{v}_1 = r_0$
 - 2.2. $\delta_1 = \|r_0\|$
3. if $\delta_1 \neq 0$, then $v_1 = \hat{v}_1/\delta_1$;
4. else $v_1 = \hat{v}_1 = 0$;
5. endif
6. $\bar{w}_1 = v_1$, $v_0 = 0$, $x_0^L = x_0$
7. while $\|r_j\| \geq \epsilon$
 - 7.1. $\hat{v}_{j+1} = Av_j - \delta_j v_{j-1}$
 - 7.2. $\gamma_j = \langle \hat{v}_{j+1}, v_j \rangle$
 - 7.3. $\hat{v}_{j+1} = \hat{v}_{j+1} - \gamma_j v_j$
 - 7.4. $\delta_{j+1} = \|\hat{v}_{j+1}\|$
 - 7.5. if $\delta_{j+1} \neq 0$, then $v_{j+1} = \hat{v}_{j+1}/\delta_{j+1}$
 - 7.6. else $v_{j+1} = \hat{v}_{j+1} = 0$;

7.7. *endif*

7.8. *if* $j = 1$, *then*

7.8.1. $\bar{d}_j = \gamma_j$

7.8.2. $\tilde{e}_{j+1} = \delta_{j+1}$

7.9. *elseif* $j > 1$, *then*

7.9.1. *Apply Givens rotation* G_j *to row* j :

7.9.2. $\bar{d}_j = s_j \tilde{e}_j - c_j \gamma_j$

7.9.3. $e_j = c_j \tilde{e}_j + s_j \gamma_j$

7.9.4. *Apply Givens rotation* G_j *to row* $j + 1$:

7.9.5. $f_{j+1} = s_j \delta_{j+1}$

7.9.6. $\tilde{e}_{j+1} = -c_j \delta_{j+1}$

7.10. *endif*

7.11. *determine Givens rotation* G_{j+1}

7.11.1. $d_j = \sqrt{\bar{d}_j^2 + \delta_{j+1}^2}$

7.11.2. $c_{j+1} = \bar{d}_j / d_j$

7.11.3. $s_{j+1} = \delta_{j+1} / d_j$

7.12. *if* $j = 1$, *then* $\zeta_1 = \delta_1 / d_1$;

7.13. *elseif* $j = 2$ *then* $\zeta_2 = -\zeta_1 e_2 / d_2$;

7.14. *elseif* $j > 2$, *then* $\zeta_j = (-\zeta_{j-1} e_j - \zeta_{j-2} f_j) / d_j$;

7.15. *endif*

7.16. $w_j = c_{j+1} \bar{w}_j + s_{j+1} v_{j+1}$

7.17. $\bar{w}_{j+1} = s_{j+1} \bar{w}_j - c_{j+1} v_{j+1}$

7.18. $x_j^L = x_{j-1}^L + \zeta_j w_j$

7.19. *if* $j = 1$, *then* $\text{res} = \|r_0\| \cdot |s_2|$;

7.20. *if* $j > 1$, *then* $\text{res} = \text{res} \cdot |s_{j+1}|$;

7.21. *endif*

7.22. *if* $c_{j+1} \neq 0$, *then* $\|r_j\| = \text{res} / c_{j+1}$

7.23. *else set* $\|r_j\| = \infty$

7.24. *endif*

end

8. $x_j = x_j^L + (\zeta_j s_{j+1} / c_{j+1}) \bar{w}_{j+1}$

by construction of the $(j + 1)$ st Givens rotation. The first j columns of $\tilde{L}_j G_{j+1}$ are equal to the matrix L_j obtained from \bar{L}_j by replacing \bar{d}_j with d_j . Hence,

$$T_j^2 + \delta_{j+1}^2 e_j e_j^T = \bar{L}_j \bar{L}_j^T + \delta_{j+1}^2 e_j e_j^T = \tilde{L}_j \tilde{L}_j^T = \tilde{L}_j G_{j+1} G_{j+1}^T \tilde{L}_j^T = L_j L_j^T.$$

Using this, we can rewrite the alternative formulation (4.55) as

$$L_j L_j^T y = \|r_0\| \bar{L}_j Q_j e_1. \quad (4.56)$$

By construction of L_j , \bar{L}_j and the relation $c_{j+1} = \bar{d}_j/d_j$ (cf. (4.48)) we find that

$$\bar{L}_j = L_j D_j, \quad D_j = \text{diag}(1, \dots, 1, c_{j+1}). \quad (4.57)$$

Since L_j is nonsingular we obtain the following linear equation which is equivalent to (4.56)

$$L_j^T y = \|r_0\| D_j Q_j e_1 \quad (4.58)$$

The definition of $Q_j = G_{jj}^T \dots G_{2j}^T$ and the structure of the Givens rotations yield that the components of

$$\|r_0\| D_j Q_j e_1 = t_j = (\tau_1, \dots, \tau_j)^T$$

obey

$$\tau_1 = \|r_0\| c_2, \quad \tau_i = \|r_0\| s_2 s_3 \dots s_i c_{i+1}, \quad i = 2, \dots, j. \quad (4.59)$$

Since L_j^T is an upper triangular matrix, the solution y_i of (4.58) has to be computed by solving the system backwards. Hence y_j can only be computed if L_j^T is completely known. Since y_j is computed by backward substitution, there is no obvious connection between y_j and $y_{j+1}, j - 1, \dots, m - 1$. To obtain a recursion for the minimum residual iterates, we define

$$M_j = (m_1, \dots, m_j) = V_j L^{-T}. \quad (4.60)$$

Then the solution of (4.54) is given by

$$x_j^M = V_j y_j = V_j L_j^{-T} L_j^T y_j = M_j t_j = x_{j-1}^M + m_j \tau_j.$$

From (4.60) and from the definition of L_j we obtain that the columns of M_j satisfy

$$m_{j-2} f_j + m_{j-1} e_j + m_j d_j = v_j,$$

or

$$m_j = (v_j - m_{j-2} f_j - m_{j-1} e_j) / d_j.$$

As in the previous cases, the minimum residual iterates recursively defined by (4.54) are approximations of the solution of $Ax = r_0$. Since we are interested in a solution of $Ax = b$ and since $r_0 = b - Ax_0$, the approximation of the solution $x_* = A^{-1}b$ is given as $x_0 + x_j^M$. Therefore we use a recursion for $x_0 + x_j^M$. We find that (denoting $x_0 + x_j^M$ by x_j^M) x_j^M obeys

$$x_0^M = x_0, \quad x_j^M = x_{j-1}^M + m_j \tau_j,$$

where x_0 is the given initial approximation. The corresponding residual can be written as

$$r_j = b - Ax_j^M = r_0 - AV_j y_j = V_j \|r_0\| e_1 - V_j T_j y_j - v_{j+1} \delta_{j+1} e_j^T y_j, \quad (4.61)$$

where y_j is the solution of (4.58). Using (4.59) and the definition of y_j we find that the last component of y_j is given by

$$e_j^T y_j = y_j^{(j)} = \|r_0\| s_2 s_3 \cdots s_j c_{j+1} / d_j.$$

Since $s_j = \delta_{j+1} / d_j$, this implies that

$$\delta_{j+1} e_j^T y_j = \|r_0\| s_2 s_3 \cdots s_j s_{j+1} c_{j+1}. \quad (4.62)$$

With $T_j^T = T_j = Q_j^T \bar{L}_j^T$, (4.57) and (4.58) we obtain the equation

$$\begin{aligned} \|r_0\| e_1 - T_j y_j &= Q_j^T (\|r_0\| Q_j e_1 - D_j L_j^T y_j) \\ &= Q_j^T (\|r_0\| Q_j e_1 - \|r_0\| D_j^2 Q_j e_1) \\ &= Q_j^T (\|r_0\| (I - D_j^2) Q_j e_1). \end{aligned}$$

Due to the structure of D_j , (4.57) and the structure of the Givens rotations we obtain that $Q_j e_1 = s_2 s_3 \cdots s_j$ and

$$\begin{aligned} \|r_0\| e_1 - T_j y_j &= Q_j^T (\|r_0\| (I - D_j^2) Q_j e_1) \\ &= \|r_0\| s_2 s_3 \cdots s_j (1 - c_{j+1}^2) Q_j^T e_j \\ &= \|r_0\| s_2 s_3 \cdots s_j s_{j+1}^2. \end{aligned} \quad (4.63)$$

Combining (4.61), (4.62), (4.63) and the fact that the vectors v_i are orthonormal we can deduce that

$$\|r_j\|^2 = \|r_0\|^2 (s_2 s_3 \cdots s_j)^2 s_{j+1}^4 + \|r_0\|^2 (s_2 s_3 \cdots s_j)^2 s_{j+1}^2 c_{j+1}^2 = \|r_0\|^2 (s_2 s_3 \cdots s_j)^2 s_{j+1}^2$$

This gives as formula for the residual norm :

$$\|r_j\| = \|r_0\| |s_2 s_3 \cdots s_j s_{j+1}| = \|r_{j-1}\| |s_{j+1}|.$$

The above presentation leads to the final form of the MINRES algorithm.

Algorithm 4.4.8 (MINRES)

1. given $A \in \mathbb{R}^{n \times n}$ symmetric, $b \in \mathbb{R}^n$, $x_0 \in \mathbb{R}^n$.
2. compute $r_0 = b - Ax_0$, set
 - 2.1. $\hat{v}_1 = r_0$
 - 2.2. $\delta_1 = \|r_0\|$

2.3. $v_0 = 0, m_0 = m_{-1} = 0$

3. *while* $\|r_j\| > \epsilon$

3.1. *if* $\delta_j \neq 0$, *then* $v_j = \hat{v}_j/\delta_j$;

3.2. *else* $v_j = \hat{v}_j = 0$;

3.3. *endif*

3.4. $\hat{v}_{j+1} = Av_j - \delta_j v_{j-1}$

3.5. $\gamma_j = \langle \hat{v}_{j+1}, v_j \rangle$,

3.6. $\hat{v}_{j+1} = \hat{v}_{j+1} - \gamma_j v_j$

3.7. $\delta_{j+1} = \|\hat{v}_{j+1}\|$

3.8. *if* $j = 1$, *then*

3.8.1. $\bar{d}_j = \gamma_j$

3.8.2. $\tilde{e}_{j+1} = \delta_{j+1}$

3.9. *elseif* $j > 1$, *then*

3.9.1. *Apply Givens rotation* G_j *to row* j :

3.9.2. $\bar{d}_j = s_j \tilde{e}_j - c_j \gamma_j$

3.9.3. $e_j = c_j \tilde{e}_j + s_j \gamma_j$

3.9.4. *Apply Givens rotation* G_j *to row* $j + 1$:

3.9.5. $f_{j+1} = s_j \delta_{j+1}$

3.9.6. $\tilde{e}_{j+1} = -c_j \delta_{j+1}$

3.10. *endif*

3.11. *Determine Givens rotation* G_{j+1}

3.11.1. $d_j = \sqrt{\bar{d}_j^2 + \delta_{j+1}^2}$,

3.11.2. $c_{j+1} = \bar{d}_j/d_j$

3.11.3. $s_{j+1} = \delta_{j+1}/d_j$

3.12. *if* $j = 1$, *then* $\tau_1 = \|r_0\|c_2$

3.13. *elseif* $j > 1$, *then* $\tau_j = \|r_0\|s_2s_3 \dots s_j c_{j+1}/c_j$

3.14. *endif*

3.15. $m_j = (v_j - m_{j-1}e_j - m_{j-2}f_j)/d_j$

3.16. $x_j = x_{j-1} + \tau_j m_j$

3.17. $\|r_j\| = |s_{j+1}| \|r_{j-1}\|$

end

Chapter 5

Preconditioning

5.1 The Issue of Preconditioning

We have seen that the convergence of MINRES and SYMMLQ is mainly determined by the distribution of the eigenvalues of the system matrix. Roughly one can say that the convergence is better when the eigenvalues are clustered. More detailed results have been discussed in Section 4.3. This is the reason why we try to precondition the matrices under consideration.

The general aim of preconditioning for a symmetric matrix A is to find a nonsingular matrix P such that

$$P^{-1}AP^{-T}$$

has a better distribution of eigenvalues than the original system matrix A . One often tries to find P such that the condition number of the preconditioned matrix is much smaller than the condition number of A . This corresponds to a shrinkage of the spectrum. Often, the outer bounds of the eigenspectrum of a matrix arising from a finite element discretization are bounded by a constant, while the inner bounds depend on the mesh constant and move towards the origin with increasing fineness of the mesh. In this case one tries to move small eigenvalues away from zero while leaving large eigenvalues essentially unchanged. But this is not the only issue. By preconditioning, the distribution of the eigenvalues in the spectrum can be favorably altered, not only the size of the spectrum. Moreover, preconditioning is only useful if the gain due to better distribution of the eigenvalues and smaller condition number is not destroyed by computationally expensive matrix operations. Therefore the solution of linear systems $Py = z$ should be cheap.

Instead of $Ax = b$ we consider the preconditioned system

$$\tilde{A}\tilde{x} = \tilde{b},$$

with

$$\tilde{A} = P^{-1}AP^{-T}, \quad \tilde{x} = P^T x, \quad \tilde{b} = P^{-1}b,$$

where $P \in \mathbb{R}^{n \times n}$ is a nonsingular matrix.

For a nonsingular and symmetric matrix $A \in \mathbb{R}^{n \times n}$, the spectral condition number, denoted by $\kappa_2(A)$, is given by

$$\kappa_2(A) = \frac{\bar{\lambda}}{\underline{\lambda}},$$

where $\bar{\lambda}$ denotes the eigenvalue of A with largest absolute value, and $\underline{\lambda}$ the eigenvalue smallest in absolute value.

A point that is important about the 2–norm condition number – which is the condition number we constantly consider – for the construction of preconditioners is the fact that the eigenvalues of $P^{-1}AP^{-T}$, $A(PP^T)^{-1}$ and of $(PP^T)^{-1}A$ are identical. Thus it is sufficient to consider $(PP^T)^{-1}A$ or $A(PP^T)^{-1}$ instead of $P^{-1}AP^{-T}$. For the construction of preconditioners for symmetric systems we have to find a symmetric positive definite matrix M playing the role of PP^T such that M is a good and computationally cheap approximation for A^{-1} . Since M is symmetric positive definite, there exists a Cholesky decomposition $M = PP^T$, and the preconditioner can be chosen from this decomposition. However, in many cases this decomposition is only used formally. The expenses are often judged to high. In the analysis it is sufficient to consider $(PP^T)^{-1}A = M^{-1}A$. Why this is sufficient will be shown in this following Section 5.2.

5.2 The Preconditioned Algorithms

First we turn to the changes introduced in the implementation of the iterative solvers by preconditioning. Of course, it would be possible to simply apply the methods to the transformed system and proceed as before. But then we have to pay a lot of unnecessary expenses. First of all, applying MINRES and SYMMLQ to the transformed system means that P^{-1} and its transpose can be computed and can, hopefully, be effectively applied. This is not always the case. Often enough, a preconditioner M is known, and a decomposition $M = PP^T$ is known to exist, but one does not want to pay the expenses of effectively computing this decomposition. Secondly, this approach requires one matrix vector product $\tilde{A} \cdot x$ in each iteration, which means in effect solving a linear system with P and P^T in each iteration and applying A . At the end of the process, the solution provided by the iteration has to be transformed to a solution of the unpreconditioned system. These expenses can be reduced by the changes we present in this section.

If we apply the MINRES and SYMMLQ to the transformed system

$$\tilde{A}\tilde{x} = \tilde{b}, \tag{5.1}$$

then we iterate on Krylov subspaces $\mathcal{K}_j(P^{-1}AP^{-T}, P^{-1}r_0)$.

In each step j , MINRES minimizes

$$\begin{aligned} \|\tilde{A}\tilde{x} - \tilde{r}_0\|_2^2 &= \|P^{-1}AP^{-T}P^T x - P^{-1}r_0\|_2^2 \\ &= \|P^{-1}(AP^{-T}P^T x - r_0)\|_2^2 \end{aligned}$$

$$\begin{aligned}
&= \|P^{-1}(Ax - r_0)\|_2^2 \\
&= \langle P^{-T}P^{-1}(Ax - r_0), Ax - r_0 \rangle \\
&= \|Ax - r_0\|_{M^{-1}}^2.
\end{aligned} \tag{5.2}$$

SYMMLQ computes iterates \tilde{x}_j such that

$$\langle \tilde{A}\tilde{x}_j - \tilde{r}_0, v \rangle = 0 \quad \forall v \in \mathcal{K}_j(P^{-1}AP^{-T}, P^{-1}r_0). \tag{5.3}$$

This is equivalent to searching for vectors x_j such that

$$\langle P^{-1}AP^{-T}P^T x_j - P^{-1}r_0, v \rangle = \langle P^{-1}(Ax_j - r_0), v \rangle = 0 \quad \forall v \in \mathcal{K}_j(P^{-1}AP^{-T}, P^{-1}r_0).$$

If we apply Algorithm 4.4.7 to the equation

$$\tilde{A}\tilde{x} = \tilde{b}, \tag{5.4}$$

with

$$\tilde{A} = P^{-1}AP^{-T}, \quad \tilde{x} = P^T x, \quad \tilde{b} = P^{-1}b,$$

then we basically have to replace the matrix A in the algorithm by

$$\tilde{A} = P^{-1}AP^{-T},$$

the vectors v_j, \hat{v}_j by

$$\begin{aligned}
\tilde{v}_j &= P^T v_j, \\
\tilde{\hat{v}}_j &= P^T \hat{v}_j,
\end{aligned}$$

and the residual r_j by the transformed residual

$$\tilde{r}_j = P^{-1}r_j.$$

The Lanczos process for the transformed problem $\tilde{A}\tilde{x} = \tilde{b}$ takes the vectors $P^{-1}r_0, (P^{-1}AP^{-T})P^{-1}r_0, \dots, (P^{-1}AP^{-T})^{j-1}P^{-1}r_0$, or, equivalently, the vectors $P^{-1}r_0, P^{-1}AM^{-1}r_0, \dots, P^{-1}(AM^{-1})^{j-1}r_0$, and orthogonalizes them against previously computed basis vectors $\tilde{v}_1, \dots, \tilde{v}_j$, where $\tilde{v}_1 = P^{-1}r_0/\|P^{-1}r_0\|$. The result is an orthogonal basis $\tilde{v}_1, \dots, \tilde{v}_j$ for the underlying Krylov spaces $\mathcal{K}_j(P^{-1}AP^{-T}, P^{-1}r_0)$, $j = 1, \dots, m$. The process is given as follows.

Algorithm 5.2.1 (Lanczos Tridiagonalization, Version 1)

1. given \tilde{r}_0 and m
2. set $\tilde{v}_1 = \tilde{r}_0/\|\tilde{r}_0\|$, $\tilde{v}_0 = 0$ and $\delta_1 = \|\tilde{v}_1\|$

3. for $j = 1, \dots, m - 1$
 - 3.1. if $\delta_j = 0$ stop
 - 3.2. $\tilde{v}_j = \tilde{v}_j / \delta_j$
 - 3.3. $\tilde{v}_{j+1} = \tilde{A}\tilde{v}_j - \delta_j\tilde{v}_{j-1}$
 - 3.4. $\gamma_j = \langle \tilde{v}_{j+1}, \tilde{v}_j \rangle$
 - 3.5. $\tilde{v}_{j+1} = \tilde{v}_{j+1} - \gamma_j\tilde{v}_j$
 - 3.6. $\delta_{j+1} = \|\tilde{v}_{j+1}\|$
- end

This version of the process gives us iterates \tilde{x}_j in $\mathcal{K}_j(P^{-1}AP^{-T}, P^{-1}r_0)$. Since we are interested in unpreconditioned iterates

$$x_j = P^{-T}\tilde{x}_j \in P^{-T}\mathcal{K}_j(P^{-1}AP^{-T}) \quad (5.5)$$

$$= P^{-T}P^{-1}\mathcal{K}_j(AP^{-T}P^{-1}, r_0) \quad (5.6)$$

$$= M^{-1}\mathcal{K}_j(AM^{-1}, r_0) \quad (5.7)$$

$$= \mathcal{K}_j(M^{-1}A, M^{-1}r_0) \quad (5.8)$$

for the original problem, we change to vectors $v_j = P^{-T}v_j$ in the Krylov subspace $\mathcal{K}_j(M^{-1}A, M^{-1}r_0)$ by theoretically applying P^{-T} to all vectors. Moreover, we need the preconditioned matrix $\tilde{A} = P^{-1}AP^{-T}$ in this version. But the actual computation of a decomposition $M = PP^T$ is something one tries to prevent. Often enough, a preconditioner M is known, and a decomposition PP^T is known to exist. But one does not really want to pay the expenses of a factorization. Fortunately, this can be circumvented by the indicated change.

While we now consider iterates in a different Krylov subspace, our basis vectors are unchanged. The vectors $\tilde{v}_1, \dots, \tilde{v}_j \in \mathcal{K}_j(P^{-1}AP^{-T}, P^{-1}r_0)$ are still the actual basis vectors. This is the reason why the normalization is not changed. We obtain by rewriting the normalization in terms of M instead of the factors P, P^T ($M = PP^T$)

$$\begin{aligned} \gamma_j &= \langle P^T\hat{v}_{j+1}, P^T v_j \rangle = \langle PP^T\hat{v}_{j+1}, v_j \rangle = \langle M\hat{v}_{j+1}, v_j \rangle, \\ \delta_{j+1} &= \langle P\hat{v}_{j+1}, P^T v_j \rangle = \langle PP^T\hat{v}_{j+1}\hat{v}_{j+1} \rangle = \langle M\hat{v}_{j+1}, \hat{v}_{j+1} \rangle. \end{aligned}$$

Algorithm 5.2.2 (Lanczos Tridiagonalization, Version 2)

1. given $M^{-1}r_0$ and m
2. set $\hat{v}_1 = M^{-1}r_0$, $v_0 = 0$ and $\delta_1 = \langle \hat{v}_1, r_0 \rangle$
3. for $j = 1, \dots, m - 1$

- 3.1. *if* $\delta_j = 0$ *stop*
 - 3.2. $v_j = \hat{v}_j / \delta_j$
 - 3.3. $\hat{v}_{j+1} = M^{-1}Av_j - \delta_j v_{j-1}$
 - 3.4. $\gamma_j = \langle M\hat{v}_{j+1}, v_j \rangle$
 - 3.5. $\hat{v}_{j+1} = \hat{v}_{j+1} - \gamma_j v_j$
 - 3.6. $\delta_{j+1} = \langle M\hat{v}_{j+1}, v_{j+1} \rangle$
- end*

In this form it is necessary to be able to compute a matrix vector product $M \cdot x$ as well as to solve linear systems $Mx = b$ with M . This might be not feasible. In order to overcome this, we introduce new vectors $\hat{u}_j = PP^T \hat{v}_j = M\hat{v}_j$ and $v_j = Mv_j$. This has the effect of delaying the solve with M . Note that solving with M is done once in each iteration. This is why systems with M should be solvable at moderate cost.

Algorithm 5.2.3 (Lanczos Tridiagonalization, Final Version)

1. *given* r_0 *and* m
 2. *set* $\hat{u}_1 = r_0$, $u_0 = 0$
 3. *solve* $M\hat{v}_1 = \hat{u}_1$
 4. *compute* $\delta_1 = \langle \hat{u}_1, \hat{v}_1 \rangle$
 5. *for* $j = 1, \dots, m - 1$
 - 5.1. *if* $\delta_j = 0$ *stop*
 - 5.2. $v_j = \hat{v}_j / \delta_j$
 - 5.3. $u_j = \hat{u}_j / \delta_j$
 - 5.4. $\hat{u}_{j+1} = Av_j - \delta_j u_{j-1}$
 - 5.5. $\gamma_j = \langle \hat{u}_{j+1}, v_j \rangle$
 - 5.6. $\hat{u}_{j+1} = \hat{u}_{j+1} - \gamma_j u_j$
 - 5.7. *solve* $M\hat{v}_{j+1} = \hat{u}_{j+1}$
 - 5.8. $\delta_{j+1} = \langle \hat{u}_{j+1}, \hat{v}_{j+1} \rangle$
- end*

This is the preconditioned form of the Lanczos Tridiagonalization with explicit normalization. For completeness we give the process with implicit normalization as well. Implicit normalization is cheaper by n multiplications in each step than explicit normalization.

Algorithm 5.2.4 (Lanczos Tridiagonalization, Final Version with implicit normalization)

1. given r_0 and m
 2. set $\hat{u}_1 = r_0, u_0 = 0$
 3. solve $M\hat{v}_1 = \hat{u}_1$
 4. compute $\delta_1 = \langle \hat{u}_1, \hat{v}_1 \rangle$
 5. for $j = 1, \dots, m - 1$
 - 5.1. if $\delta_j = 0$ stop
 - 5.2. $v_j = \hat{v}_j / \delta_j$
 - 5.3. $\hat{u}_{j+1} = Av_j - \frac{\delta_j}{\delta_{j-1}}\hat{u}_{j-1}$
 - 5.4. $\gamma_j = \langle \hat{u}_{j+1}, v_j \rangle$
 - 5.5. $\hat{u}_{j+1} = \hat{u}_{j+1} - \frac{\gamma_j}{\delta_j}\hat{u}_j$
 - 5.6. solve $M\hat{v}_{j+1} = \hat{u}_{j+1}$
 - 5.7. $\delta_{j+1} = \langle \hat{u}_{j+1}, \hat{v}_{j+1} \rangle$
- end

We have derived the Lanczos process for the preconditioned problem in a form requiring as few expenses as possible in rewriting the process such that for one a decomposition of the preconditioner M into factors P, P^T is not necessary and secondly iterates x_j can be directly computed in iterating on the Krylov subspace $\mathcal{K}_j(M^{-1}A, M^{-1}r_0)$ instead of $\mathcal{K}_j(P^{-1}AP^{-T}, P^{-1}r_0)$.

MINRES in its preconditioned form, using the preconditioned version of the Lanczos process with implicit normalization, is given as follows.

Algorithm 5.2.5 (preconditioned MINRES)

1. given $A \in \mathbb{R}^{n \times n}$ symmetric, $b \in \mathbb{R}^n, x_0 \in \mathbb{R}^n$.
2. compute $\hat{u}_1 = b - Ax_0$
3. solve $M\hat{v}_1 = \hat{u}_1$.
4. compute $\delta_1 = \langle \hat{v}_1, \hat{u}_1 \rangle$, set
 - 4.1. $\|\tilde{r}_0\| = \sqrt{\delta_1}$
 - 4.2. $\hat{u}_0 = 0, \delta_0 = 1$

- 4.3. $m_0 = m_{-1} = 0$
5. *while* $\|\tilde{r}_j\| > \epsilon$
- 5.1. *if* $\delta_j \neq 0$, *then* $v_j = \hat{v}_j/\delta_j$;
- 5.2. *else* $v_j = \hat{v}_j = 0$;
- 5.3. *endif*
- 5.4. $\hat{u}_{j+1} = Av_j - \frac{\delta_j}{\delta_{j-1}}\hat{u}_{j-1}$
- 5.5. $\gamma_j = \langle \hat{u}_{j+1}, v_j \rangle$
- 5.6. $\hat{u}_{j+1} = \hat{u}_{j+1} - \frac{\gamma_j}{\delta_j}\hat{u}_j$
- 5.7. *solve* $M\hat{v}_{j+1} = \hat{u}_{j+1}$
- 5.8. $\delta_{j+1} = \langle \hat{v}_{j+1}, \hat{u}_{j+1} \rangle$
- 5.9. *if* $j = 1$, *then*
- 5.9.1. $\bar{d}_j = \gamma_j$
- 5.9.2. $\tilde{e}_{j+1} = \delta_{j+1}$
- 5.10. *elseif* $j > 1$, *then*
- 5.10.1. *Apply Givens rotation* G_j *to row* j ;
- 5.10.2. $\bar{d}_j = s_j\tilde{e}_j - c_j\gamma_j$
- 5.10.3. $e_j = c_j\tilde{e}_j + s_j\gamma_j$
- 5.10.4. *Apply Givens rotation* G_j *to row* $j + 1$;
- 5.10.5. $f_{j+1} = s_j\delta_{j+1}$
- 5.10.6. $\tilde{e}_{j+1} = -c_j\delta_{j+1}$
- 5.11. *endif*
- 5.12. *Determine Givens rotation* G_{j+1}
- 5.12.1. $d_j = \sqrt{\bar{d}_j^2 + \delta_{j+1}^2}$
- 5.12.2. $c_{j+1} = \bar{d}_j/d_j$
- 5.12.3. $s_{j+1} = \delta_{j+1}/d_j$
- 5.13. *if* $j = 1$, *then* $\tau_1 = \|\tilde{r}_0\|c_2$;
- 5.14. *elseif* $j > 1$, *then* $\tau_j = \|\tilde{r}_0\|s_2s_3 \dots s_j c_{j+1}/c_j$;
- 5.15. *endif*
- 5.16. $m_j = (v_j - m_{j-1}e_j - m_{j-2}f_j)/d_j$
- 5.17. $x_j = x_{j-1} + \tau_j m_j$
- 5.18. $\|\tilde{r}_j\| = |s_{j+1}| \|\tilde{r}_{j-1}\|$
- end*

Incorporating the transformed Lanczos process into SYMMLQ leads to its preconditioned form:

Algorithm 5.2.6 (preconditioned SYMMLQ)

1. *given* $A \in \mathbb{R}^{n \times n}$ symmetric, $b \in \mathbb{R}^n$, $x_0 \in \mathbb{R}^n$.
2. *compute* $\hat{u}_1 = b - Ax_0$, $\delta_0 = 1$
3. *solve* $M\hat{v}_1 = \hat{u}_1$
4. *compute* $\delta_1 = \langle \hat{v}_1, \hat{u}_1 \rangle$, *set* $\|\tilde{r}_0\| = \sqrt{\delta_1}$
5. *if* $\delta_1 \neq 0$, *then* $v_1 = r_0/\delta_1$;
6. *else* $v_1 = 0$;
7. *endif*
8. $\bar{w}_1 = v_1$, $v_0 = 0$, $x_0^L = x_0$
9. *while* $\|\tilde{r}_j\| \geq \epsilon$
 - 9.1. $\hat{u}_{j+1} = Av_j - \frac{\delta_j}{\delta_{j-1}}\hat{u}_{j-1}$
 - 9.2. $\gamma_j = \langle \hat{u}_{j+1}, v_j \rangle$
 - 9.3. $\hat{u}_{j+1} = \hat{u}_{j+1} - \frac{\gamma_j}{\delta_j}\hat{u}_j$
 - 9.4. *solve* $M\hat{v}_{j+1} = \hat{u}_{j+1}$
 - 9.5. $\delta_{j+1} = \langle \hat{v}_{j+1}, \hat{u}_{j+1} \rangle$
 - 9.6. *if* $\delta_{j+1} \neq 0$, *then* $v_{j+1} = \hat{v}_{j+1}/\delta_{j+1}$
 - 9.7. *else* $v_{j+1} = \hat{v}_{j+1} = 0$;
 - 9.8. *endif*
 - 9.9. *if* $j = 1$, *then*
 - 9.9.1. $\bar{d}_j = \gamma_j$.
 - 9.9.2. $\tilde{e}_{j+1} = \delta_{j+1}$
 - 9.10. *elseif* $j > 1$, *then*
 - 9.10.1. *Apply Givens rotation* G_j *to row* j :
 - 9.10.2. $\bar{d}_j = s_j\tilde{e}_j - c_j\gamma_j$
 - 9.10.3. $e_j = c_j\tilde{e}_j + s_j\gamma_j$
 - 9.10.4. *Apply Givens rotation* G_j *to row* $j + 1$:
 - 9.10.5. $f_{j+1} = s_j\delta_{j+1}$
 - 9.10.6. $\tilde{e}_{j+1} = -c_j\delta_{j+1}$

- 9.11. *endif*
- 9.12. *Determine Givens rotation G_{j+1}*
- 9.12.1. $d_j = \sqrt{\bar{d}_j^2 + \delta_{j+1}^2}$
- 9.12.2. $c_{j+1} = \bar{d}_j/d_j$
- 9.12.3. $s_{j+1} = \delta_{j+1}/d_j$
- 9.13. *if $j = 1$, then $\zeta_1 = \zeta_1/d_1$;*
- 9.14. *if $j = 2$, then $\zeta_2 = -\zeta_1 e_2/d_2$;*
- 9.15. *elseif $j > 2$, then $\zeta_j = (-\zeta_{j-1} e_j - \zeta_{j-2} f_j)/d_j$;*
- 9.16. *endif*
- 9.17. $x_j^L = x_{j-1}^L + \zeta_j(c_{j+1}\bar{w}_j + s_{j+1}v_{j+1})$
- 9.18. $\bar{w}_{j+1} = s_{j+1}\bar{w}_j - c_{j+1}v_{j+1}$
- 9.19. $\|\tilde{r}_j\| = |s_{j+1}| \|\tilde{r}_{j-1}\|$
10. $x_j = x_j^L + (\zeta_j s_{j+1}/c_{j+1})\bar{w}_{j+1}$

Chapter 6

The Preconditioners

6.1 Introduction

We now turn to the preconditioners for matrices of the form

$$K = \begin{pmatrix} H_y & 0 & A^T \\ 0 & H_u & B^T \\ A & B & 0 \end{pmatrix}, \quad (6.1)$$

where

$$H_y = M_y + D_y \quad \text{and} \quad H_u = \alpha \cdot M_u + D_u.$$

We assume that $H_y \in \mathbb{R}^{m \times m}$, $H_u \in \mathbb{R}^{n \times n}$ are symmetric positive definite and that $A \in \mathbb{R}^{m \times m}$ is nonsingular.

In general, the effectiveness of a preconditioner does depends on the particular system the preconditioner is used for. There are some preconditioners, for example the preconditioner constructed by an incomplete Cholesky factorization, or a truncated series approach, that are designed without taking into account the structure of the matrix. Their usage can be highly effective, but this is not necessarily the case. Matrices that arise from the discretization of partial differential equations by finite element methods are highly structured, and they have been studied by numerous authors, so that many of their features are well-known. For this reason we do not attempt to use preconditioners of such general design, but focus on preconditioners taking advantage of the special form and features of the matrices we are interested in. We want to precondition the system such that its eigenvalues are bounded independently of the mesh constant. Often, the eigenvalues that are large in absolute value are bounded by constants arising from the nature of the discretization, but the smaller eigenvalues do depend on the mesh constant and are moving towards the origin as the mesh becomes finer. This causes the condition number to grow. Whenever the goal of constant bounds for the spectrum is achieved, the performance of the iterative solvers is independent of the fineness of the discretization, and the iteration numbers is essentially the same for coarse and fine meshes, i.e. for matrices of moderate size and very large matrices.

In our derivation of the preconditioners we are motivated by different assumptions on the underlying matrices. We distinguish four different cases in general.

Case 1: $\alpha = 1$, $D_y = 0$, $D_u = 0$

In this case we can reduce the condition number of the systems under consideration considerably. By preconditioning we reduce the iterations required by MINRES and SYMMLQ to a number which appears to be independent of the grid size.

Case 2: $\alpha \ll 1$, $D_y = 0$, $D_u = 0$

In this case, the spectrum of H_u moves towards the origin, and while the conditioning of H_u itself is not changed, the condition number of K increases significantly. As α decreases, the system with K becomes hard to solve, and for sufficiently small values of α MINRES and SYMMLQ need an unacceptably large number of iterations. The performance of MINRES and SYMMLQ improves on the preconditioned systems.

Case 3: $\alpha = 1$, $D_y = 0$, $D_u \gg I$

If there are inequality constraints for u , we often have to deal with a diagonal matrix D_u with entries that are considerably larger than 1. We write $D_u \gg I$ and mean this to be understood componentwise. Large entries in D_u can be shown to affect the conditioning of the preconditioned system only to a moderate amount. In fact, they can even help to neutralize a small parameter α or large entries in D_y . Like in Case 1, we can construct efficient preconditioners. In Section 2 we mentioned the connection between the systems arising in our applications and the systems turning up in linear programming. The case with inequality constraints on u corresponds to the non-degenerate case in linear programming, and it is possible to derive efficient preconditioners. For a comparison of our first preconditioner and a preconditioner proposed by Gill, Murray, Ponceléon and Saunders [6] see Section 6.2.3.

Case 4: $\alpha = 1$, $D_y \gg I$, $D_u = 0$

This situation is less favorable for the preconditioned systems we analyze than the preceding ones. The situation where constraints are imposed on y may correspond to the degenerate case in linear programming. This was mentioned in Section 2. Inequality constraints for y can lead to a matrix $D_y \gg I$. A large diagonal in H_y unfavorably affects the performance of MINRES and SYMMLQ on the preconditioned systems we consider as well as on the original K of our application.

For the evaluation of the preconditioners we investigate the modification of the spectrum of K due to preconditioning and the cost of applying the preconditioner. These two issues are discussed for various preconditioners in this section and for a specific example in Section 7.1. In Sections 7.6, 7.7 and 7.8 we also investigate the quality of the computed solution.

In the following P_y and P_u are preconditioners of H_y and H_u , respectively, i.e. P_y and P_u are nonsingular matrices such that

$$P_y^{-1}H_yP_y^{-T} \approx I, \quad \text{and} \quad P_u^{-1}H_uP_u^{-T} \approx I. \quad (6.2)$$

By \tilde{A}^{-1} we denote an approximate inverse of A ,

$$\tilde{A}^{-1}A \approx I. \quad (6.3)$$

6.2 The First Preconditioner

6.2.1 Derivation of the First Preconditioner

To motivate the first preconditioner, we make the following assumptions on the spectra of the submatrices in

$$K = \begin{pmatrix} H_y & 0 & A^T \\ 0 & H_u & B^T \\ A & B & 0 \end{pmatrix}. \quad (6.4)$$

Qualitatively, these assumptions hold for a large class of applications, see e.g. Section 7.1. We assume that the spectra of the matrices H_u and H_y depend on the mesh constant h such that essentially

$$\Lambda(H_y) = [c_1h^l, c_2h^l], \quad \text{and} \quad \Lambda(H_u) = [c_3h^k, c_4h^k] \quad (6.5)$$

for some constants c_1, c_2, c_3, c_4 and some integers k, l . Furthermore, we assume that A is a square nonsingular matrix. Although A is nonsingular, it is ill-conditioned. We denote by μ the union of the eigenvalues of H_y and H_u and by σ the singular values of $(A|B)$. From the estimates

$$\lambda_{2m+n} \geq \frac{1}{2}(\mu_{\min} - \sqrt{\mu_{\min}^2 + 4\sigma_{\max}^2}), \quad (6.6)$$

$$\lambda_{m+n+1} \leq \frac{1}{2}(\mu_{\max} - \sqrt{\mu_{\max}^2 + 4\sigma_{\min}^2}), \quad (6.7)$$

$$\lambda_{m+n} \geq \mu_{\min}, \quad (6.8)$$

$$\lambda_1 \leq \frac{1}{2}(\mu_{\max} + \sqrt{\mu_{\max}^2 + 4\sigma_{\max}^2}) \quad (6.9)$$

derived in Theorem 3.2.1 for the eigenvalues of the system (6.4) we see that the eigenvalues of the system move towards the origin if the mesh constant h becomes smaller. This influence of the mesh constant can be neutralized by the application of

$$\begin{pmatrix} H_y^{-1/2} & 0 & 0 \\ 0 & H_u^{-1/2} & 0 \\ 0 & 0 & I_n \end{pmatrix} \quad (6.10)$$

to the system K in (6.4) from the left and from the right. Note that H_y and H_u are symmetric. This leads to the system

$$\begin{pmatrix} I_m & 0 & H_y^{-1/2} A^T \\ 0 & I_n & H_u^{-1/2} B^T \\ AH_y^{-1/2} & BH_u^{-1/2} & 0 \end{pmatrix}. \quad (6.11)$$

For this system we know that the values μ_{min}, μ_{max} in (6.6) through (6.9) are 1. But, although A is a square nonsingular matrix, we assume ill-conditioning. If H_y satisfies (6.5), then the conditioning of $H_y^{-1/2} A$ is essentially equal to the conditioning of A . If we multiply A by $H_y^{-1/2}$, this affects the singular values

$$\sigma^2(AH_y^{-1/2} | BH_u^{-1/2}),$$

which are the eigenvalues of

$$(AH_y^{-1/2} | BH_u^{-1/2}) \cdot (AH_y^{-1/2} | BH_u^{-1/2})^T = AH_y^{-1} A^T + BH_u^{-1} B^T.$$

Under the assumption (6.5) the singular values of $(AH_y^{-1/2} | BH_u^{-1/2})$ often rise in comparison to the singular values of $(A|B)$. We can reduce these by preconditioning with the matrix

$$\begin{pmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & H_y^{1/2} A \end{pmatrix}. \quad (6.12)$$

This transforms the system (6.11) into

$$\begin{pmatrix} I & 0 & I \\ 0 & I & H_u^{-1/2} B^T A^{-T} H_y^{1/2} \\ I & H_y^{1/2} A^{-1} B H_u^{-1/2} & 0 \end{pmatrix}. \quad (6.13)$$

The transformations (6.10) and (6.12) lead to the ideal preconditioner P_1^* which is given by

$$P_1^* = \begin{pmatrix} H_y^{1/2} & 0 & 0 \\ 0 & H_u^{1/2} & 0 \\ 0 & 0 & AH_y^{-1/2} \end{pmatrix}.$$

In this case,

$$(P_1^*)^{-1} K (P_1^*)^{-T} = \begin{pmatrix} I_m & 0 & I_m \\ 0 & I_n & H_u^{-1/2} B^T A^{-T} H_y^{1/2} \\ I_m & H_y^{1/2} A^{-1} B H_u^{-1/2} & 0 \end{pmatrix}. \quad (6.14)$$

Here and in the following, we use I_m, I_n to denote the dimension of the identity matrices.

Instead of $H_y^{1/2}$ and $H_u^{-1/2}$ one can use the Cholesky factors of H_y and H_u , respectively. This is a particular case of the generalization discussed in the following.

In general, $H_y^{-1/2}, H_u^{-1/2}$, and A^{-1}, A^{-T} can not be computed exactly. To derive a practicable preconditioner, we assume that preconditioners P_y, P_u of H_y, H_u are available and that an approximate inverse \tilde{A}^{-1} of A is known. This leads to the first preconditioner in a general form which is given by

$$P_1 = \begin{pmatrix} P_y & 0 & 0 \\ 0 & P_u & 0 \\ 0 & 0 & \tilde{A}P_y^{-T} \end{pmatrix},$$

or, equivalently, by its inverse,

$$P_1^{-1} = \begin{pmatrix} P_y^{-1} & 0 & 0 \\ 0 & P_u^{-1} & 0 \\ 0 & 0 & P_y^T \tilde{A}^{-1} \end{pmatrix}.$$

The preconditioned KKT matrix is

$$P_1^{-1} K P_1^{-T} = \begin{pmatrix} P_y^{-1} H_y P_y^{-T} & 0 & P_y^{-1} \tilde{A}^{-T} A^T P_y \\ 0 & P_u^{-1} H_u P_u^{-T} & P_u^{-1} B^T \tilde{A}^{-T} P_y \\ P_y^T \tilde{A}^{-1} A P_y^{-T} & P_y^T \tilde{A}^{-1} B P_u^{-T} & 0 \end{pmatrix} \quad (6.15)$$

and we expect that

$$(P_1)^{-1} K (P_1)^{-T} = \begin{pmatrix} \tilde{I}_m & 0 & \tilde{I}_m \\ 0 & \tilde{I}_n & P_u^{-1} B^T \tilde{A}^{-T} P_y \\ \tilde{I}_m & P_y^T \tilde{A}^{-1} B P_u^{-T} & 0 \end{pmatrix}. \quad (6.16)$$

with \tilde{I} an approximate identity matrix. The preconditioned system still has the structure allowing us to give estimates on its spectrum with Theorem 3.2.1. The derivation of the general form of our first preconditioner is motivated by the assumption that for preconditioners P_y, P_u of H_y, H_u and for an approximate inverse \tilde{A}^{-1} of A the singular values of $P_y^T \tilde{A}^{-1} B P_u^{-T}$ are small. This is the case for the matrices arising in our application and can be shown to hold true more generally for problems of this type.

In order to derive bounds for the eigenvalues of the preconditioned system we need to establish the following lemma.

Lemma 6.2.1 *Let $\tilde{B} \in \mathbb{R}^{m \times n}$. The singular values σ_i of $(I_m | \tilde{B})$ are given by*

$$\sigma_i = \sqrt{1 + \sigma_i^2(\tilde{B})}, \quad i = 1, \dots, m,$$

where $\sigma_i(\tilde{B})$ are the singular values of \tilde{B} . If $m \geq n$, \tilde{B} has n singular values, and we set $\sigma_i(\tilde{B}) = 0$ for $i = n + 1, \dots, m$.

Proof: The symmetry of $\tilde{B}\tilde{B}^T \in \mathbb{R}^{m \times m}$ implies that there exists an orthonormal matrix $Q \in \mathbb{R}^{m \times m}$ such that

$$Q^T \tilde{B}\tilde{B}^T Q = \text{diag}(\lambda_i(\tilde{B}\tilde{B}^T)) = \text{diag}(\sigma_i^2(\tilde{B})), \quad i = 1, \dots, m.$$

This implies

$$\begin{aligned} Q^T(I + \tilde{B}\tilde{B}^T)Q &= Q^T((I|\tilde{B})(I|\tilde{B})^T)Q \\ &= Q^T Q + Q^T(\tilde{B}\tilde{B}^T)Q \\ &= I + \text{diag}(\lambda_i(\tilde{B}\tilde{B}^T)) \\ &= I + \text{diag}(\sigma_i^2(\tilde{B})), \end{aligned}$$

since

$$\lambda_i(\tilde{B}\tilde{B}^T) = \sigma_i^2(\tilde{B}) \quad \text{for } i = 1, \dots, m, \quad \text{if } m \leq n,$$

and

$$\lambda_i(\tilde{B}\tilde{B}^T) = \begin{cases} \sigma_i^2(\tilde{B}) & \text{for } i = 1, \dots, n, \\ 0 & \text{for } i = n + 1, \dots, m, \end{cases} \quad \text{if } m \geq n.$$

This gives the assertion. □

In the following we denote the largest and smallest singular values of \tilde{B} by $\sigma_{min}, \sigma_{max}$. Note that only in the case $n \geq m$ we have a smallest singular value $\sigma_{min} = \sigma_m$ which is possibly greater than zero.

Using Theorem 3.2.1 and Lemma 6.2.1 we now obtain the following result for the preconditioner P_1 .

Let $\sigma_{max}, \sigma_{min} \geq 0$ denote the largest and smallest singular values of $P_y^T \tilde{A}^{-1} B P_u^{-T}$, respectively, and let μ_{max}, μ_{min} denote the largest and smallest eigenvalues of the upper left part \tilde{I}_{m+n} of the preconditioned system. The eigenvalues $\lambda_1 \geq \dots \geq \lambda_{m+n} > 0 > \lambda_{m+n+1} \geq \dots \geq \lambda_{2m+n}$ of the preconditioned system (6.15) obey

$$\lambda_{2m+n} \geq \frac{1}{2}(\mu_{min} - \sqrt{5 + 4\sigma_{max}^2}), \quad (6.17)$$

$$\lambda_{m+n+1} \leq \frac{1}{2}(\mu_{max} - \sqrt{5 + 4\sigma_{min}^2}), \quad (6.18)$$

$$\lambda_{m+n} \geq \mu_{min}, \quad (6.19)$$

$$\lambda_1 \leq \frac{1}{2}(\mu_{max} + \sqrt{5 + 4\sigma_{max}^2}). \quad (6.20)$$

If we assume the ideal preconditioner P_1^* , i.e. if $P_y^{-1} H_y P_y^{-T} = I_m$ and $P_u^{-1} H_u P_u^{-T} = I_n$, these expressions simplify with $\mu_{min} = \mu_{max} = 1$. For the matrices arising in our application it can be shown that

$$\|M_y^{1/2} A^{-1} B M_u^{-1/2}\| \leq c \quad (6.21)$$

for a constant c independent of h . This is formally derived in a more general framework in [2]. Thus in Case 1 ($H_y = M_y$ and $H_u = M_u$) we expect that, for preconditioners P_u, P_y and \tilde{A} neutralizing the dependency of H_y, H_u and A on the mesh constant h , we can similarly bound the singular values of $P_y^T \tilde{A}^{-1} B P_u^{-T}$ such that

$$\|P_y^T \tilde{A}^{-1} B P_u^{-T}\| \leq c_P, \quad (6.22)$$

where c_P is a constant independent of h .

The expected performance of the first preconditioner in the four cases is discussed below.

6.2.2 Expected Performance of the First Preconditioner

With the tools collected in Section 6.2.1 we now investigate the expected performance of the preconditioner in the different cases. By $\sigma_i^{(l)} = \sigma_i^{(l)}(P_y^T \tilde{A}^{-1} B P_u^{-T})$, $l = 1, 2, 3, 4$, we denote the singular values of $P_y^T \tilde{A}^{-1} B P_u^{-T}$ in Case $l = 1, 2, 3, 4$.

Case 1: $\alpha = 1$, $D_y = 0$, $D_u = 0$

If $\alpha = 1$, (6.21) shows that there exists a constant upper bound for the singular values $\sigma^{(1)}(H_y^{1/2} A^{-1} B H_u^{-1/2})$. The preconditioner P_1 can be expected to perform well if the preconditioning matrices P_y, P_u and \tilde{A} neutralize the influence of the mesh size h on the submatrices and thus on the system, and if the singular values of $P_y^T \tilde{A}^{-1} B P_u^{-T}$ are bounded by a small constant c_P . If the eigenvalues of $P_y^{-1} H_y P_y^{-1}$ and $P_u^{-T} H_u P_u^{-1}$ are close to one and if $\sigma_{min}^{(1)} \ll 1$, where $\sigma_i^{(1)}$ denote the singular values of $(P_y^T \tilde{A}^{-1} B P_u^{-T})$, we can deduce

$$\lambda_{m+n} \approx 1, \quad \lambda_{m+n+1} \approx \frac{1}{2}(1 - \sqrt{5}),$$

so that the eigenvalues of the preconditioned system are bounded away from zero. If in addition $\sigma_{max}^{(1)}$ is of moderate size, the condition number of the preconditioned system $P_1^{-1} K P_1^{-T}$ is small.

The preconditioner will perform poorly if the singular values of $P_y^T \tilde{A}^{-1} B P_u^{-T}$ are not small. This happens in two of the four cases we consider next.

Case 2: $\alpha \ll 1$, $D_y = 0$, $D_u = 0$

If a small parameter α determines the size of the eigenvalues of the matrix M_u , we must expect that bounds on the norm $\|H_y^{1/2} A^{-1} B H_u^{-1/2}\|$ grow with the reciprocal of $\sqrt{\alpha}$. Denoting by $\sigma_i^{(2)}$ the singular values of $H_y^{1/2} A^{-1} B H_u^{-1/2}$, we have the relationship

$$\sigma_i^{(2)} = \frac{1}{\sqrt{\alpha}} \sigma_i^{(1)}.$$

For decreasing values of α the spectrum of $P_y^T \tilde{A}^{-1} B P_u^{-T}$ expands and the conditioning of the preconditioned system deteriorates.

Case 3: $\alpha = 1$, $D_y = 0$, $D_u \gg I$

If interior-point methods are applied to problems with inequality constraints for u , H_u has a diagonal that is considerably larger in size than the remaining entries. This is the case $H_u = \alpha M_u + D_u$, where $D_u \gg I$, i.e. some diagonal entries may become very large. Analogously we write $P_u = \alpha P_O + P_D$, where P_D stands for the (large) diagonal entries and P_O for the off-diagonal entries that are generally of moderate size. By $\sigma_i^{(3)}$ we denote the singular values of $P_y^T \tilde{A}^{-1} B P_u^{-T}$. The estimate

$$\begin{aligned}
\sigma_i^{(3)} &= \sigma_i^{(3)}(P_y^T \tilde{A}^{-1} B P_u^{-T}) \\
&= \sigma_i^{(3)}(P_y^T \tilde{A}^{-1} B (\alpha P_O + P_D)^{-T}) \\
&= \sigma_i^{(3)}(P_y^T \tilde{A}^{-1} B P_D^{-T} (\alpha P_D^{-1} P_O + I)^{-T}) \\
&\leq \|P_y^T \tilde{A}^{-1} B\| \|P_D^{-T}\| \|\alpha (P_D^{-1} P_O + I)^{-T}\| \\
&\leq \|P_y^T \tilde{A}^{-1} B\| \|P_D^{-T}\| \cdot \frac{1}{1 - \|\alpha P_O^T P_D^{-T}\|}
\end{aligned}$$

follows from the Banach-Lemma (see for example [8], p.59).

If D_u dominates the matrix H_u , $\|\alpha P_O P_D^{-T}\|$ will be of negligible size. If additionally $\alpha \ll 1$, this contributes to reducing the factor $1/(1 - \|\alpha P_O P_D^{-1}\|)$ to a constant close to one. The norm $\|P_y^T \tilde{A}^{-1} B\|$ can be expected to be of moderate size, while $\|P_D^{-1}\|$ will be very small. The singular values $\sigma^{(3)}$ converge to zero as the entries in the diagonal D_u , and with it in P_D grow. In the case of large diagonal entries in H_u we can expect a good performance of the solvers on the preconditioned system, due to a small condition number of $P_1^{-1} K P_1^{-T}$ which is in turn induced by small singular values of $P_y^T \tilde{A}^{-1} B P_u^{-T}$.

Case 4: $\alpha = 1$, $D_y \gg I$, $D_u = 0$

The diagonal of H_y can become very large if interior-point methods are applied to problems with inequality constraints on y , and if these inequality constraints on y are active. The spectrum of the system matrix blows up under the influence of these large entries.

If we denote by P_y the preconditioner for H_y and by P_O , P_D its off-diagonal part and its diagonal part, respectively, then we see that the matrix $P_y^T \tilde{A}^{-1} B P_u^{-T}$ will have very large singular values. This is indicated by the estimates ($M = \tilde{A}^{-1} B P_u^{-T} P_u^{-1} B^T \tilde{A}^{-T}$)

$$\lambda_{\max}((P_O + P_D)^T M (P_O + P_D)) \geq \lambda_{\max}(P_D^T M P_D) + \lambda_{\min}(P_O^T M P_O + P_O^T M P_D + P_D^T M P_O)$$

and

$$\lambda_{\min}((P_O + P_D)^T M (P_O + P_D)) \leq \lambda_{\min}(P_O^T M P_D + P_D^T M P_O + P_D^T M P_D) + \lambda_{\max}(P_O^T M P_O).$$

(For the estimates see [8], p.411.)

6.2.3 Comparison with Gill, Murray, Ponceleón and Saunders

In [6], Gill, Murray, Ponceleón and Saunders are concerned with preconditioning of indefinite systems. The systems they are dealing with arise in linear programming and are generally of the form

$$K = \begin{pmatrix} H_u & 0 & B^T \\ 0 & H_y & A^T \\ B & A & 0 \end{pmatrix}. \quad (6.23)$$

Here B is a rectangular matrix corresponding to the non-basis variables, A is a square nonsingular basis matrix, and the matrices $H_y = D_y$, $H_u = D_u$ correspond to μX^{-2} in (2.6) introduced for inequality constraints. After a permutation of rows 1 and 2 and columns 1 and 2, the system (6.23) is equal to the system (1.1). We use the notation in (6.23) to be consistent with [6]. Gill et al. are concerned with the situation where, due to the application of barrier methods or interior-point methods for linear programming, the diagonal entries of H_u grow to very large values which cause the condition number of the system to rise. To cancel the influence of the large entries they suggest to precondition the system by

$$\begin{pmatrix} H_u^{-1/2} & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{pmatrix}. \quad (6.24)$$

This leads to the equivalent system

$$\begin{pmatrix} I & 0 & H_u^{-1/2} B^T \\ 0 & H_y & A^T \\ B H_u^{-1/2} & A & 0 \end{pmatrix}. \quad (6.25)$$

In the light of Theorem 3.2.1 we see that in our situation, with the spectrum of H_y depending on the square of the mesh constant h , the conditioning of the system deteriorates with increasing fineness of the mesh, due to small eigenvalues in the upper part of the system. Moreover, we have to deal with an ill-conditioned matrix A that gives rise to large singular values of $(B H_u^{-1/2} | A)$. So this preconditioner will bring only minor improvement for the systems arising in our application. Another preconditioner Gill et al. suggest leads to the system

$$\begin{pmatrix} I & 0 & H_u^{-1/2} B^T \\ 0 & A^{-1} H_y A^{-T} & A^T \\ B H_u^{-1/2} & A & 0 \end{pmatrix}. \quad (6.26)$$

The numerical results of Gill et al. indicate that these two preconditioners give good results in the nondegenerate case. However, this second preconditioner requires the application of A^{-1} and A^{-T} and is, assuming that the application of $H_u^{-1/2}$ and $H_y^{-1/2}$ is cheap, as costly as the application of our preconditioner P_1 . Still, the singular values of $(B H_u^{-1/2} | A)$, large in our applications, are unchanged, so that in our application the second preconditioner by Gill et al. will not be much better than the first one they suggest. Our preconditioner P_1 ,

however, can be shown to reduce the condition number of the system in our application to a small constant independent of the mesh size. The cost of applying P_1 is comparable to the cost of applying the preconditioner leading to (6.26).

For the applications Gill e .al. consider, their first two preconditioners do not give satisfying results if the diagonal entries of H_y grow, too. This is the degenerate case in linear programming. We have encountered similar difficulties in the corresponding case, where constraints on y cause the diagonal of H_y to grow if interior-point methods are applied. See e.g. Section 7.5.

6.2.4 Application of the First Preconditioner

Of course, it is important that the preconditioner is efficient. The application of the preconditioner P_1 can be done as follows. Let $z = (z_1, z_2, z_3)^T$ with $z_1 \in \mathbb{R}^m, z_2 \in \mathbb{R}^n, z_3 \in \mathbb{R}^m$ and let $x = (x_1, x_2, x_3)^T$ with $x_1 \in \mathbb{R}^m, x_2 \in \mathbb{R}^n, x_3 \in \mathbb{R}^m$. The transformed vector $x = P_1^{-1}z$ can be computed by solving the linear systems

$$\begin{aligned} x_1 &= P_y^{-1}z_1, \\ x_2 &= P_u^{-1}z_2, \\ x_3 &= P_y^T \tilde{A}^{-1}z_3. \end{aligned}$$

Likewise, $w = P_1^{-T}x$, where $w = (w_1, w_2, w_3)^T$ with $w_1 \in \mathbb{R}^m, w_2 \in \mathbb{R}^n, w_3 \in \mathbb{R}^m$, can be computed by solving the linear systems

$$\begin{aligned} w_1 &= P_y^{-T}x_1, \\ w_2 &= P_u^{-T}x_2, \\ w_3 &= \tilde{A}^{-T}P_y^T x_3. \end{aligned}$$

Of course, we never compute the inverses of matrices, but solve the corresponding systems.

6.3 The Second Preconditioner

We have seen that we cannot expect the preconditioned system (6.15) to be well-conditioned in all the cases we consider. The structure of the system (6.15) does not allow to apply further transformations only to the critical part $P_y^T \tilde{A}^{-1}BP_u^{-T}$ without affecting other parts of the system, too. Therefore, it is our goal to eliminate the blocks coupling the left upper part \tilde{I}_{m+n} of the matrix in (6.16) with its lower part $(I|\tilde{B})$, where \tilde{B} is $P_y^T \tilde{A}^{-1}BP_u^{-T}$. The spectrum of a block diagonal matrix is the union of the spectra of the blocks. Thus a block-diagonal form should be easier to handle than the system $P_1^{-1}KP_1^{-T}$ in (6.16), where we constantly have to employ Theorem 3.2.1 to state any estimates about the anticipated spectrum, and where the interaction of the eigenvalues of the upper part \tilde{I}_{m+n} and the singular values of the lower part is a delicate issue.

6.3.1 Derivation of the Ideal Second Preconditioner

In order to make the derivation of the second preconditioner transparent, we start by transforming the preconditioned system (6.13) that is achieved by applying the ideal preconditioner P_1^* to the original matrix K .

A first Gauss elimination step for (6.13) is the transformation

$$\begin{aligned} & \begin{pmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & -H_y^{1/2}A^{-1}B H_u^{-1/2} & I \end{pmatrix} (P_1^*)^{-1}K(P_1^*)^{-T} \begin{pmatrix} I & 0 & 0 \\ 0 & I & -H_u^{-1/2}B^T A^{-T}H_y^{1/2} \\ 0 & 0 & I \end{pmatrix} \\ = & \begin{pmatrix} I & 0 & I \\ 0 & I & 0 \\ I & 0 & -H_y^{1/2}A^{-1}B H_u^{-1}B^T A^{-T}H_y^{1/2} \end{pmatrix}. \end{aligned} \quad (6.27)$$

Block diagonal structure is then achieved in a second step by transforming (6.27) into

$$\begin{aligned} & \begin{pmatrix} I & 0 & 0 \\ 0 & I & 0 \\ -I & 0 & I \end{pmatrix} \\ & \times \begin{pmatrix} I & 0 & I \\ 0 & I & 0 \\ I & 0 & -H_y^{1/2}A^{-1}B H_u^{-1}B^T A^{-T}H_y^{1/2} \end{pmatrix} \\ & \times \begin{pmatrix} I & 0 & -I \\ 0 & I & 0 \\ 0 & 0 & I \end{pmatrix} \\ = & \begin{pmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & I & -(I + H_y^{1/2}A^{-1}B H_u^{-1}B^T A^{-T}H_y^{1/2}) \end{pmatrix}. \end{aligned} \quad (6.28)$$

Combining the transformations in (6.27) and (6.28) with the ideal preconditioner P_1^* yields the ideal preconditioner P_2^* , given by its inverse as

$$(P_2^*)^{-1} = \begin{pmatrix} H_y^{-1/2} & 0 & 0 \\ 0 & H_u^{-1/2} & 0 \\ -H_y^{-1/2} & -H_y^{1/2}A^{-1}B H_u^{-1} & H_y^{1/2}A^{-1} \end{pmatrix}. \quad (6.29)$$

The ideal preconditioned system is

$$P_2^{*-1}K P_2^{*-T} = \begin{pmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & I & -(I + \tilde{B}\tilde{B}^T) \end{pmatrix}. \quad (6.30)$$

6.3.2 Derivation of the General Second Preconditioner

Unfortunately, we cannot in general assume that systems with A , $H_y^{1/2}$ or $H_u^{1/2}$ can be solved. Moreover, the derivation above started off at the ideally preconditioned system (6.14). If the starting point is the matrix in (6.15), then the step (6.27) becomes

$$\begin{aligned}
& \begin{pmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & -P_y^T \tilde{A}^{-1} B P_u^{-T} & I \end{pmatrix} P_1^{-1} K P_1^{-T} \begin{pmatrix} I & 0 & 0 \\ 0 & I & -P_u^{-1} B^T \tilde{A}^{-T} P_y \\ 0 & 0 & I \end{pmatrix} \\
&= \begin{pmatrix} P_y^{-1} H_y P_y^{-T} & 0 & P_y^{-1} A^T \tilde{A}^{-T} P_y \\ 0 & P_u^{-1} H_u P_u^{-T} & S_{32}^T \\ P_y^T \tilde{A}^{-1} A P_y^{-T} & S_{32} & S_{33} \end{pmatrix}, \tag{6.31}
\end{aligned}$$

where

$$\begin{aligned}
S_{32} &= P_y^T \tilde{A}^{-1} B P_u^{-T} \cdot (I - P_u^{-1} H_u P_u^{-T}), \\
S_{33} &= P_y^T \tilde{A}^{-1} B P_u^{-T} (P_u^{-1} H_u P_u^{-T} - 2I) P_u^{-1} B^T \tilde{A}^{-T} P_y.
\end{aligned}$$

It can be assumed that $P_y^{-1} H_y P_y^{-T}$, $P_u^{-1} H_u P_u^{-T}$ and $\tilde{A}^{-1} A$ are approximate identities and with them $P_y^T \tilde{A}^{-1} A P_y^{-T} \approx I$, whereas $P_y^T \tilde{A}^{-1} B P_u^{-T} \cdot (I - P_u^{-1} H_u P_u^{-T}) \approx 0$. In this situation it is less clear than in (6.28), which step can be considered as the most favorable translation of (6.28) to the altered system corresponding to (6.30). In fact, we have two possibilities to proceed.

a) i) An exact elimination step for $P_y^T \tilde{A}^{-1} A P_y^{-T}$ in (6.31) would require the application of

$$\begin{aligned}
-P_y^T \tilde{A}^{-1} A P_y^{-T} \cdot (P_y^{-1} H_y P_y^{-T})^{-1} &= -P_y^T \tilde{A}^{-1} A P_y^{-T} \cdot (P_y^T H_y^{-1} P_y) \\
&= -P_y^T \tilde{A}^{-1} A H_y^{-1} P_y.
\end{aligned}$$

Here H_y^{-1} would be replaced by its preconditioner $P_y^{-T} P_y^{-1}$ because solving with H_y need not be feasible. The step (6.28) becomes

$$\begin{aligned}
& \begin{pmatrix} I & 0 & 0 \\ 0 & I & 0 \\ -P_y^T \tilde{A}^{-1} A P_y^{-T} & 0 & I \end{pmatrix} \\
& \times \begin{pmatrix} P_y^{-1} H_y P_y^{-T} & 0 & P_y^{-1} A^T \tilde{A}^{-T} P_y \\ 0 & P_u^{-1} H_u P_u^{-T} & S_{32}^T \\ P_y^T \tilde{A}^{-1} A P_y^{-T} & S_{32} & S_{33} \end{pmatrix} \\
& \times \begin{pmatrix} I & 0 & -P_y^{-1} A^T \tilde{A}^{-T} P_y \\ 0 & I & 0 \\ 0 & 0 & I \end{pmatrix} \\
&= \begin{pmatrix} P_y^{-1} H_y P_y^{-T} & 0 & S_{31}^{(a)T} \\ 0 & P_u^{-1} H_u P_u^{-T} & S_{32}^T \\ S_{31}^{(a)} & S_{32} & S_{33}^{(a)} \end{pmatrix}, \tag{6.32}
\end{aligned}$$

where

$$\begin{aligned}
S_{31}^{(a)} &= P_y^T \tilde{A}^{-1} A P_y^{-T} \cdot (I - P_y^{-1} H_y P_y^{-T}), \\
S_{32} &= P_y^T \tilde{A}^{-1} B P_u^{-T} \cdot (I - P_u^{-1} H_u P_u^{-T}), \\
S_{33}^{(a)} &= P_y^T \tilde{A}^{-1} A P_y^{-T} (P_y^{-1} H_y P_y^{-T} - 2I) P_y^{-1} A^T \tilde{A}^{-T} P_y \\
&\quad + P_y^T \tilde{A}^{-1} B P_u^{-T} (P_u^{-1} H_u P_u^{-T} - 2I) P_u^{-1} B^T \tilde{A}^{-T} P_y \\
&= S_{33} + P_y^T \tilde{A}^{-1} B P_u^{-T} (P_u^{-1} H_u P_u^{-T} - 2I) P_u^{-1} B^T \tilde{A}^{-T} P_y.
\end{aligned}$$

Combining the congruence transformations used in (6.31) and (6.32) and the preconditioner P_1 yields the second preconditioner. The second preconditioner is in this general form given by

$$\begin{aligned}
P_{2a}^{-1} &= \begin{pmatrix} I & 0 & 0 \\ 0 & I & 0 \\ -P_y^T \tilde{A}^{-1} A P_y^{-T} & 0 & I \end{pmatrix} \\
&\quad \times \begin{pmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & -P_y^T \tilde{A}^{-1} B P_u^{-T} & I \end{pmatrix} \\
&\quad \times \begin{pmatrix} P_y^{-1} & 0 & 0 \\ 0 & P_u^{-1} & 0 \\ 0 & 0 & P_y^T \tilde{A}^{-1} \end{pmatrix} \\
&= \begin{pmatrix} P_y^{-1} & 0 & 0 \\ 0 & P_u^{-1} & 0 \\ -P_y^T \tilde{A}^{-1} A P_y^{-T} & -P_y^T \tilde{A}^{-1} B P_u^{-T} P_u^{-1} & P_y^T \tilde{A}^{-1} \end{pmatrix}.
\end{aligned}$$

a) ii) Alternatively, one might have the idea to use the approximate identity $I \approx P_y^T \tilde{A}^{-1} A P_y^{-T}$ to eliminate the off-diagonal part. This yields the same transformation as in a)i).

The costs for the application of the second preconditioner in this form is discussed in Section 6.3.4.

b) Obeying the considerations in a) above, (6.31) would be transformed in an additional step requiring essentially the application of A and a solve with \tilde{A} in order to eliminate an approximate identity. It might not be necessary to pay these additional computational costs. The step below leads to a system similar to (6.32).

$$\begin{aligned}
&\begin{pmatrix} I & 0 & 0 \\ 0 & I & 0 \\ -I & 0 & I \end{pmatrix} \\
&\quad \times \begin{pmatrix} P_y^{-1} H_y P_y^{-T} & 0 & P_y^{-1} A^T \tilde{A}^{-T} P_y \\ 0 & P_u^{-1} H_u P_u^{-T} & S_{32}^T \\ P_y^T \tilde{A}^{-1} A P_y^{-T} & S_{32} & S_{33} \end{pmatrix}
\end{aligned}$$

$$\begin{aligned}
& \times \begin{pmatrix} I & 0 & -I \\ 0 & I & 0 \\ 0 & 0 & I \end{pmatrix} \\
& = \begin{pmatrix} P_y^{-1}H_yP_y^{-T} & 0 & S_{31}^{(b)T} \\ 0 & P_u^{-1}H_uP_u^{-T} & S_{32}^T \\ S_{31}^{(b)} & S_{32} & S_{33}^{(b)} \end{pmatrix}, \tag{6.33}
\end{aligned}$$

where

$$\begin{aligned}
S_{31}^{(b)} &= P_y^T \tilde{A}^{-1} A P_y^{-T} - P_y^{-1} H_y P_y^{-T}, \\
S_{32} &= P_y^T \tilde{A}^{-1} B P_u^{-T} \cdot (I - P_u^{-1} H_u P_u^{-T}), \\
S_{33}^{(b)} &= S_{33} - 2P_y^{-1} A^T \tilde{A}^{-T} P_y + P_y^{-1} H_y P_y^{-T}.
\end{aligned}$$

The second preconditioner is in this general form given by

$$\begin{aligned}
P_{2b}^{-1} &= \begin{pmatrix} I & 0 & 0 \\ 0 & I & 0 \\ -I & 0 & I \end{pmatrix} \begin{pmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & -P_y^T \tilde{A}^{-1} B P_u^{-T} & I \end{pmatrix} \begin{pmatrix} P_y^{-1} & 0 & 0 \\ 0 & P_u^{-1} & 0 \\ 0 & 0 & P_y^T \tilde{A}^{-1} \end{pmatrix} \\
&= \begin{pmatrix} P_y^{-1} & 0 & 0 \\ 0 & P_u^{-1} & 0 \\ -P_y^{-1} & -P_y^T \tilde{A}^{-1} B P_u^{-T} P_u^{-1} & P_y^T \tilde{A}^{-1} \end{pmatrix}.
\end{aligned}$$

The transformation of the system K to the system $P_{2b}^{-1} K P_{2b}^{-T}$ is essentially no costlier than the transformation to $P_1^{-1} K P_1^{-T}$, assuming that the dominant costs are those of solving with the approximation \tilde{A} to A . The application of the preconditioner will be discussed in Section 6.3.4.

6.3.3 Expected Performance of the Second Preconditioner

The matrix $I + \tilde{B}\tilde{B}^T = I + H_y^{1/2} A^{-1} B H_u^{-1} B^T A^{-T} H_y^{1/2}$ is a rank- k -modification of the identity. Here k denotes the rank of B . The matrix $B \in \mathbb{R}^{m \times n}$ is a rectangular matrix, so that its rank is $k \leq \min\{m, n\}$. This means that $B H_u^{-1} B^T \in \mathbb{R}^{m \times m}$ has k nonzero eigenvalues and $m - k$ eigenvalues equal to zero. Since $\text{rank}(A_1 \cdot A_2) \leq \text{rank}(A_1) \cdot \text{rank}(A_2)$ and because $H_y^{1/2}$ and A are nonsingular, $\text{rank}(\tilde{B}\tilde{B}^T) = k$. This means that the lower block $-(I + \tilde{B}\tilde{B}^T)$ in the preconditioned system $(P_2^*)^{-1} K (P_2^i)^{-T}$ is a matrix with $m - k$ eigenvalues equal to -1 , and k eigenvalues that possibly differ from -1 . Thus the ideal system (6.30) has at most $k + 2$ distinct eigenvalues. The system (6.30) has $m + n$ eigenvalues equal to one, $m - k$ eigenvalues that are equal to -1 , and k eigenvalues that we cannot locate exactly. Thus, the iterative solution methods we use will theoretically solve a linear system with the ideally preconditioned system $(P_2^*)^{-1} K (P_2^*)^{-T}$ in (6.30) after at most $k + 2$ iterations. This ideal situation is not encountered if approximate preconditioners P_y, P_u of H_y, H_u and \tilde{A} of A are used. However, if an exact factorization of A is used, and if the eigenvalues of $P_y^{-1} H_y P_y^{-T}$

and $P_u^{-1}H_uP_u^{-T}$ are clustered around 1, then we expect a substantial decrease of the residual after the first $k + 2$ steps.

Notice that \tilde{B} is identical to the $(3, 2)$ -block in the ideal system $(P_1^*)^{-1}K(P_1^*)^{-T}$ that was studied in the previous section.

Case 1: $\alpha = 1, D_y = 0, D_u = 0$

In the case where $\alpha \approx 1, D_y = 0, D_u = 0$, the performance of the iterative solution methods is expected to be good. The reason is that for moderately sized H_y, H_u the singular values of $\tilde{B} = P_y^T A^{-1} B P_u^{-T}$ are of moderate size (see (6.21)), and for good preconditioners P_y, P_u of H_y, H_u the spectra of the blocks in (6.33) are narrow.

Case 2: $\alpha \ll 1, D_y = 0, D_u = 0$

If the parameter α is small, the spectrum of $I + \tilde{B}\tilde{B}^T = I + P_y^T A^{-1} B P_u^{-T} P_u^{-1} B^T A^{-T} P_y$ must be expected to grow with the reciprocal of α . The performance of MINRES and SYMMLQ will deteriorate.

Case 3: $\alpha = 1, D_y = 0, D_u \gg I$

In the case, where the diagonal of H_u increases to large values, the spectrum of $\tilde{B} = P_y^T A^{-1} B P_u^{-T}$ shrinks. The iterations the iterative solvers need are likely to decrease with respect to those in Case 1. More than $m - k$ eigenvalues will be considered as -1 by the solvers, so that the number of computationally distinct eigenvalues reduces. Moreover, the matrix $I - P_u^{-1}H_uP_u^{-T}$ will be nearer to the zero matrix than in Case 1.

Case 4: $\alpha = 1, D_y \gg I, D_u = 0$

The performance will be worse than in the preceding cases in the presence of a large diagonal in H_y . The spectrum of $\tilde{B} = P_y^T A^{-1} B P_u^{-T}$ will be enlarged considerably. All eigenvalues of $P_y^T A^{-1} B P_u^{-T} P_u^{-1} B^T A^{-T} P_y$ will be large.

6.3.4 Application of the Second Preconditioner

a) The application of the preconditioner P_{2a} can be done as follows. Let $z = (z_1, z_2, z_3)^T$ with $z_1 \in \mathbb{R}^m, z_2 \in \mathbb{R}^n, z_3 \in \mathbb{R}^m$ and let $x = (x_1, x_2, x_3)^T$ with $x_1 \in \mathbb{R}^m, x_2 \in \mathbb{R}^n, x_3 \in \mathbb{R}^m$. The transformed vector $x = P_{2a}^{-1}z$ can be computed as follows.

$$\begin{aligned} x_1 &= P_y^{-1}z_1, \\ x_2 &= P_u^{-1}z_2, \\ x_3 &= P_y^T \tilde{A}^{-1} \left(-Ax_1 - B P_u^{-T} x_2 + z_3 \right). \end{aligned}$$

We can compute $w = P_{2a}^{-T}x$, where $w = (w_1, w_2, w_3)^T$ with $w_1 \in \mathbb{R}^m$, $w_2 \in \mathbb{R}^n$, $w_3 \in \mathbb{R}^m$, by solving the linear systems

$$\begin{aligned} w_3 &= \tilde{A}^{-T}P_y x_3, \\ w_2 &= P_u^{-T}(x_2 - P_u^{-1}B^T w_3), \\ w_1 &= P_y^{-T}x_1 - P_y^{-1}A^T w_3. \end{aligned}$$

Assuming that the preconditioners for H_u and H_y can be applied efficiently and that, therefore, the cost of applying \tilde{A}^{-1} and of the multiplication with A dominates the other computations, we can see that the application of P_{2a}^{-1} is essentially costlier than the application of P_1^{-1} in requiring two multiplications with A and A^T , respectively.

(b) The application of the preconditioner P_{2b} can be done as follows. Let $z = (z_1, z_2, z_3)^T$ with $z_1 \in \mathbb{R}^m$, $z_2 \in \mathbb{R}^n$, $z_3 \in \mathbb{R}^m$, let $x = (x_1, x_2, x_3)^T$ with $x_1 \in \mathbb{R}^m$, $x_2 \in \mathbb{R}^n$, $x_3 \in \mathbb{R}^m$. The transformed vector $x = P_{2b}^{-1}z$ can be computed as follows.

$$\begin{aligned} x_1 &= P_y^{-1}z_1, \\ x_2 &= P_u^{-1}z_2, \\ x_3 &= P_y^T \tilde{A}^{-1}(z_3 - B P_u^{-T}x_2) - x_1. \end{aligned}$$

We can compute $w = P_{2b}^{-T}x$, where $w = (w_1, w_2, w_3)^T$ with $w_1 \in \mathbb{R}^m$, $w_2 \in \mathbb{R}^n$, $w_3 \in \mathbb{R}^m$, by solving the linear systems

$$\begin{aligned} w_1 &= P_y^{-T}(x_1 - x_3), \\ w_3 &= \tilde{A}^{-T}P_y x_3, \\ w_2 &= P_u^{-T}(x_2 - P_u^{-1}B^T w_3). \end{aligned}$$

Assuming that the preconditioners for H_u and H_y can be applied efficiently and that, therefore, the cost of applying \tilde{A}^{-1} dominates the other computations, we can see that the application of P_{2b}^{-1} is essentially not costlier than the application of P_1^{-1} . Therefore, we chose this form of the second preconditioner in our implementation. The numerical results are presented in Section 7.7.

6.3.5 Quality of the Solution

The preconditioned system $(P_2^*)^{-1}K(P_2^*)^{-T}$ is of block-diagonal form. This enables us to analyze how the error depends on the eigenvalues of the preconditioned system. The error in some components of the solution has the potential to rise considerably in the presence of small eigenvalues in the spectrum of the preconditioned system.

In the following we denote by K_2 the preconditioned system

$$K_2 = (P_2^*)^{-1}K(P_2^*)^{-T} = \begin{pmatrix} I_m & 0 & 0 \\ 0 & I_n & 0 \\ 0 & 0 & C \end{pmatrix} \quad (6.34)$$

with a symmetric matrix $C = -(I + \tilde{B}\tilde{B}^T) \in \mathbb{R}^{m \times m}$. It has a eigenvalue decomposition

$$V^T K_2 V = \begin{pmatrix} I_m & 0 & 0 \\ 0 & I_n & 0 \\ 0 & 0 & \Lambda \end{pmatrix}, \quad (6.35)$$

where

$$\Lambda = \text{diag}(\mu_{m+n+1}, \dots, \mu_{2m+n}).$$

We denote by μ_i , $i = m+n+1, \dots, 2m+n$, the eigenvalues of K_2 associated with C . Recall that $\mu_i = 1$ for $i = 1, \dots, m+n$. Here, Λ denotes the part of the spectrum associated with C . The orthogonal matrix $V \in \mathbb{R}^{(2m+n) \times (2m+n)}$ can be partitioned into $V = (V_1|V_2|V_3)$ with $V_1 \in \mathbb{R}^{(2m+n) \times m}$, $V_2 \in \mathbb{R}^{(2m+n) \times n}$ and $V_3 \in \mathbb{R}^{(2m+n) \times m}$. We can deduce from the special structure of the preconditioned system in (6.34) that

$$V_1 = \begin{pmatrix} I_m \\ 0 \\ 0 \end{pmatrix}, V_2 = \begin{pmatrix} 0 \\ I_n \\ 0 \end{pmatrix}, \quad \text{and } V_3 = \begin{pmatrix} 0 \\ 0 \\ \tilde{V}_3 \end{pmatrix}$$

with $\tilde{V}_3 \in \mathbb{R}^{m \times m}$. MINRES in its preconditioned form iterates on vectors $\tilde{x}_k \in \mathcal{K}_k(K_2, \tilde{r}_0)$ for $k = 0, 1, \dots$, starting at the (transformed) initial residual $\tilde{r}_0 = (P_2^*)^{-1} r_0$. In each step k of the iteration, MINRES minimizes

$$\|K_2 \tilde{x}_k - \tilde{r}_0\|_2 = \|(P_2^*)^{-1} K (P_2^*)^{-T} \cdot (P_2^*)^T x_k - (P_2^*)^{-1} r_0\|_2,$$

where $x_k \in \mathcal{K}_k(P_2^{*-1} P_2^{*-T} K, P_2^{*-1} P_2^{*-T} r_0)$ is a vector in the Krylov subspace spanned by $P_2^{*-1} P_2^{*-T} r_0$ and the matrix $P_2^{*-1} P_2^{*-T} K = (P_2^{*T} P_2^*)^{-1} K$. Using the notation $\hat{x} = V^T \tilde{x}$ and $\hat{r}_0 = V^T \tilde{r}_0$, we see that the requirement on the residual

$$\|K_2 \tilde{x} - \tilde{r}_0\|_2 \leq \epsilon \quad (6.36)$$

is equivalent to

$$\left\| \begin{pmatrix} I_m & 0 & 0 \\ 0 & I_n & 0 \\ 0 & 0 & \Lambda \end{pmatrix} \hat{x} - \hat{r}_0 \right\|_2 \leq \epsilon. \quad (6.37)$$

If we denote by x^* the exact solution to the original system with K and r_0 , i.e.

$$r_0 = K x^*,$$

and analogously by \tilde{x}^* the exact solution to the linear system with K_2 and \tilde{r}_0 , so that

$$\tilde{r}_0 = K_2 \tilde{x}^* = (P_2^*)^{-1} K (P_2^*)^{-T} (P_2^*)^T x^*,$$

then we have

$$\hat{r}_0 = V^T \tilde{r}_0 = \begin{pmatrix} I_m & 0 & 0 \\ 0 & I_n & 0 \\ 0 & 0 & \Lambda \end{pmatrix} V^T \tilde{x}^* \quad \text{with } \hat{x}^* = V^T \tilde{x}^*.$$

Therefore, (6.37) can be written as

$$\left\| \begin{pmatrix} I_m & 0 & 0 \\ 0 & I_n & 0 \\ 0 & 0 & \Lambda \end{pmatrix} (\hat{x} - \hat{x}^*) \right\|_2 \leq \epsilon. \quad (6.38)$$

Since (6.38) and (6.36) are equivalent, the estimate (6.36) holds if and only if

$$\begin{aligned} |\hat{x}_i - \hat{x}_i^*| &\leq \epsilon & i = 1, \dots, m+n, \\ |\mu_i| |\hat{x}_i - \hat{x}_i^*| &\leq \epsilon & i = m+n+1, \dots, 2m+n. \end{aligned} \quad (6.39)$$

The error is bounded by

$$|\hat{x}_i - \hat{x}_i^*| \leq \begin{cases} \epsilon & i = 1, \dots, m+n, \\ \frac{\epsilon}{|\mu_i|} & i = 2m+1, \dots, 2m+n. \end{cases} \quad (6.40)$$

Introducing a partitioning similar to that of V we write for the error $\hat{e} = \hat{x} - \hat{x}^* = (\hat{e}_1, \hat{e}_2, \hat{e}_3)^T$ with $\hat{e}_1 \in \mathbb{R}^m, \hat{e}_2 \in \mathbb{R}^n, \hat{e}_3 \in \mathbb{R}^m$. The error $\tilde{e} = \hat{x} - \hat{x}^*$ is the error in coordinates transformed by V^T . For the error in the preconditioned MINRES-iterates $\tilde{x} = P_2^{*T} x$ we get

$$\tilde{e}_i = V \hat{e}_i = \begin{pmatrix} \hat{e}_1 \\ \hat{e}_2 \\ \tilde{V}_3 \hat{e}_3 \end{pmatrix}. \quad (6.41)$$

From this orthogonal transformation and (6.40) we can deduce the following about the size of the error in the components of the preconditioned iterates:

$$\|\tilde{e}_1\|_2 = \|\hat{e}_1\|_2 \leq \epsilon \sqrt{m}, \quad (6.42)$$

$$\|\tilde{e}_2\|_2 \leq \epsilon \sqrt{n}, \quad (6.43)$$

$$\|\tilde{e}_3\|_2 \leq \epsilon \sqrt{\sum_{i=m+n+1}^{2m+n} \frac{1}{\mu_i^2}}. \quad (6.44)$$

The error in the components \tilde{e}_1 and \tilde{e}_2 is of the order of the residual. The estimate (6.44) indicates that the error in the component \tilde{e}_3 is potentially much larger than the residual ϵ . The error in the coordinates for the original system is given by

$$e = x_k - x^* = (P_2^*)^{-T} \tilde{e} = \begin{pmatrix} H_y^{-1/2} & 0 & 0 \\ 0 & H_u^{-1/2} & 0 \\ -H_y^{-1/2} & -H_y^{-1/2} A^{-1} B H_u^{-1/2} & H_y^{1/2} A^{-1} \end{pmatrix} \begin{pmatrix} \tilde{e}_1 \\ \tilde{e}_2 \\ \tilde{e}_3 \end{pmatrix}. \quad (6.45)$$

Partitioning this into the components $e_1 \in \mathbb{R}^m, e_2 \in \mathbb{R}^n, e_3 \in \mathbb{R}^m$, the error in the original coordinates is

$$e_1 = H_y^{-1/2} \tilde{e}_1, \quad (6.46)$$

$$e_2 = H_u^{-1/2} \tilde{e}_3, \quad (6.47)$$

$$\begin{aligned} e_3 &= -H_y^{-1/2} \tilde{e}_1 - H_y^{-1/2} A^{-1} B H_u^{-1/2} \tilde{e}_2 + H_y^{1/2} A^{-1} \tilde{e}_3 \\ &= H_y^{1/2} \left(A^{-1} (\tilde{e}_3 - B H_u^{-1/2} \tilde{e}_2) - \tilde{e}_1 \right). \end{aligned} \quad (6.48)$$

6.4 The Third Preconditioner

6.4.1 Derivation of the Third Preconditioner

A third preconditioner can be derived from the congruence transformations we introduced in §3.1. The ideal preconditioner P_3^* , given by its inverse as

$$(P_3^*)^{-1} = \begin{pmatrix} I_m & 0 & -1/2 H_y A^{-1} \\ 0 & 0 & A^{-1} \\ -(A^{-1}B)^T & I_n & (A^{-1}B)^T H_y A^{-1} \end{pmatrix}.$$

transforms K such that we get the blockdiagonal system

$$(P_3^*)^{-1} K (P_3^*)^{-T} = \begin{pmatrix} 0 & I_m & 0 \\ I_m & 0 & 0 \\ 0 & 0 & W^T H W \end{pmatrix}. \quad (6.49)$$

As before, W denotes

$$W = \begin{pmatrix} -A^{-1}B \\ I \end{pmatrix}$$

and is a representation for the nullspace of $C = (A|B)$. Since $H_{uy} = H_{yu} = 0$, the matrix $W^T H W$ is given by

$$W^T H W = B^T A^{-T} H_y A^{-1} B + H_u.$$

Note that $W^T H W \in \mathbb{R}^{n \times n}$. The partitioning of the blocks within the system has changed. This is the reason why we here use the notation $I = I_m$ and $I = I_n$, respectively.

We see that in order to solve a system with the ideal preconditioner P_3^* , we do not have to solve with H_y . It is only necessary to apply H_y , i.e. to compute a matrix-vector product $H_y \cdot x$. Therefore, we do not replace H_y by its preconditioner $P_y P_y^T$.

In a general form, the third preconditioner is given by its inverse as

$$P_3^{-1} = \begin{pmatrix} I_m & 0 & -1/2 H_y \tilde{A}^{-1} \\ 0 & 0 & \tilde{A}^{-1} \\ -(\tilde{A}^{-1}B)^T & I_n & (\tilde{A}^{-1}B)^T H_y \tilde{A}^{-1} \end{pmatrix}.$$

The system is then

$$(P_3)^{-1} K (P_3)^{-T} = \begin{pmatrix} S_{11} & S_{21}^T & S_{31}^T \\ S_{21} & 0 & S_{32}^T \\ S_{31} & S_{32} & S_{33} \end{pmatrix},$$

where

$$\begin{aligned} S_{11} &= H_y - \frac{1}{2} H_y \tilde{A}^{-1} A - \frac{1}{2} A^T \tilde{A}^{-T} H_y, \\ S_{21} &= \tilde{A}^{-1} A, \end{aligned}$$

$$\begin{aligned}
S_{31} &= (\tilde{A}^{-1}B)^T \left(H_y \tilde{A}^{-1}A - H_y + \frac{1}{2}A^T \tilde{A}^{-T} H_y - \frac{1}{2}H_y \right), \\
S_{32} &= (\tilde{A}^{-1}B)^T (I - A^T \tilde{A}^{-T}), \\
S_{33} &= (\tilde{A}^{-1}B)^T H_y (\tilde{A}^{-1}B) + H_u + (\tilde{A}^{-1}B)^T (2H_y - A^T \tilde{A}^{-T} H_y - H_y \tilde{A}^{-1}A) (\tilde{A}^{-1}B).
\end{aligned}$$

The third preconditioner does not require H_u . The matrix H_y arises in its original form, not its inverse. Therefore, only the approximate inverse \tilde{A} of A is needed, but no preconditioners P_y or P_u .

The application of the preconditioner requires essentially twice the amount of work in comparison to the first two preconditioners since the application of P_3^{-1} involves A^{-1} and A^{-T} .

6.4.2 Expected Performance of the Third Preconditioner

If we use the ideal preconditioner, we transform (6.1) into a system with at most $n + 2$ distinct eigenvalues. The $2m$ eigenvalues of

$$\begin{pmatrix} 0 & I_m \\ I_m & 0 \end{pmatrix} \tag{6.50}$$

are 1 and -1 , both with multiplicity m , and $W^T H W$ is of dimension n , so that it has n eigenvalues. These are positive since we assume H to be positive definite on the nullspace of C . The Krylov subspace methods MINRES and SYMMLQ will require not more than $n + 2$ steps to compute the exact solution to a linear system with the ideally preconditioned matrix (6.49). This is an advantage particularly in the situation where n is relatively small. This is the case in our application. In any case, $n + 2$ distinct eigenvalues for a system of dimension $2m + n$ is a relatively small number.

Case 1: $\alpha = 1$, $D_y = 0$, $D_u = 0$

The eigenvalues of $(A^{-1}B)^T H_y (A^{-1}B)$ are small in our application, located between 10^{-6} and 10^{-1} . However, since the eigenvalues of H_u are of moderate size for $\alpha = 1$, $D_y = 0$, $D_u = 0$, the eigenvalues of $W^T H W$ are of moderate size. Thus we can expect a low number of iterations.

Case 2: $\alpha \ll 1$, $D_y = 0$, $D_u = 0$

The eigenvalues of H_u are determined by the size of the parameter α . If α becomes small, the eigenvalues of H_u become small. Since the eigenvalues of $(A^{-1}B)^T H_y (A^{-1}B)$ are small, the eigenvalues of $W^T H W$ are in this situation considerably smaller than in Case 1. We must expect a raised number of iterations compared to Case 1. However, as soon as the eigenvalues of H_u have, under the influence of a small α , become smaller than the eigenvalues of $(A^{-1}B)^T H_y (A^{-1}B)$, the eigenvalue distribution of the preconditioned system will remain essentially the same, unchanged by a still decreasing parameter α .

Case 3: $\alpha = 1$, $D_y = 0$, $D_u \gg I$

If the diagonal of H_u is increased by a considerable amount such that H_u is dominated by its diagonal, then the eigenvalues of H_u are dominated by the diagonal entries. If the increase in H_u is uniform, the preconditioned system will have essentially three different eigenvalues. The iteration numbers can be expected to be even lower than in Case 1.

Case 4: $\alpha = 1$, $D_y \gg I$, $D_u = 0$

The situation is less favorable than in the preceding cases if the diagonal of H_y is increased. Even if the increase in the diagonal of H_y is uniform and the eigenvalues of H_y are all located around one large value, the spectrum of $(A^{-1}B)^T H_y (A^{-1}B)$ will have large spreads and little clustering. This is caused by the action of $A^{-1}B$ and of its transpose. MINRES and SYMMLQ need a considerably higher number of iterations than in the preceding cases if H_y is dominated by a large diagonal.

6.4.3 Application of the Third Preconditioner

The application of the preconditioner P_3 can be done in the following way. Note that the vector x is partitioned differently from z and w . Let $z = (z_1, z_2, z_3)^T$ with $z_1 \in \mathbb{R}^m$, $z_2 \in \mathbb{R}^n$, $z_3 \in \mathbb{R}^m$, let $x = (x_1, x_2, x_3)^T$ with $x_1 \in \mathbb{R}^m$, $x_2 \in \mathbb{R}^n$, $x_3 \in \mathbb{R}^n$. The transformed vector $x = P_3^{-1}z$ can be computed as follows.

$$\begin{aligned} x_2 &= A^{-1}z_3, \\ x_1 &= z_1 - \frac{1}{2}H_y x_2, \\ x_3 &= z_2 + B^T A^{-T} (H_y x_2 - z_1). \end{aligned}$$

Using one additional array t in the implementation, we need to compute the product with H_y only once:

$$\begin{aligned} x_2 &= A^{-1}z_3, \\ t &= H_y x_2, \\ x_1 &= z_1 - \frac{1}{2}t, \\ x_3 &= z_2 + B^T A^{-T} (t - z_1). \end{aligned}$$

Since the components in z_3 are no longer needed after solving the system $z_3 = Ay_2$, an additional array t is not really needed; we can overwrite z_3 with $H_y A^{-1}z_3$.

The application of the transpose of the third preconditioner can be done in a similar way. We can compute $w = P_3^{-T}x$, where $w = (w_1, w_2, w_3)^T$ with $w_1 \in \mathbb{R}^m$, $w_2 \in \mathbb{R}^n$, $w_3 \in \mathbb{R}^m$, by solving the linear systems

$$w_2 = x_3,$$

$$\begin{aligned}
t &= A^{-1}Bx_3, \\
w_1 &= x_1 - t, \\
w_3 &= A^{-T}\left(x_2 + H_y\left(t - \frac{1}{2}x_1\right)\right).
\end{aligned}$$

Note that in this case an additional array t for the implementation is actually necessary.

We have to form the matrix–vector product with H_y once to apply the transpose of the preconditioner P_3 . This is necessary for the application of P_3 , too. Assuming that H_y can be applied efficiently and that, therefore, the cost of applying A^{-1} dominates the other computations, we can see that the costs of the application of P_3 are essentially twice the costs of applying the preconditioners P_1 and P_2 .

6.4.4 Quality of the Solution

The preconditioned system $(P_3^*)^{-1}K(P_3^*)^{-T}$ is a block–diagonal matrix like the ideally preconditioned system $(P_2^*)^{-1}K(P_2^*)^{-T}$ that we considered in Section 6.3.5. Similarly to the analysis in Section 6.3.5 we can derive estimates for the absolute error in the solution to the preconditioned system, depending on the eigenvalues of the lower block.

In the following we denote by K_3 the preconditioned system

$$K_3 = (P_3^*)^{-1}K(P_3^*)^{-T} = \begin{pmatrix} 0 & I_m & 0 \\ I_m & 0 & 0 \\ 0 & 0 & C \end{pmatrix} \quad (6.51)$$

with a symmetric matrix $C \in \mathbb{R}^{n \times n}$. It has an eigenvalue decomposition

$$V^T K_3 V = \begin{pmatrix} I_m & 0 & 0 \\ 0 & -I_m & 0 \\ 0 & 0 & \Lambda \end{pmatrix}, \quad (6.52)$$

where

$$\Lambda = \text{diag}(\mu_{2m+1}, \dots, \mu_{2m+n}).$$

We denote by μ_i , $i = 2m + 1, \dots, 2m + n$, the eigenvalues of K_3 associated with C . Recall that $|\mu_i| = 1$ for $i = 1, \dots, 2m$. Here, Λ denotes the part of the spectrum associated with C . The orthogonal matrix $V \in \mathbb{R}^{(2m+n) \times (2m+n)}$ can be partitioned into $V = (V_1|V_2|V_3)$ with $V_1 \in \mathbb{R}^{(2m+n) \times m}$, $V_2 \in \mathbb{R}^{(2m+n) \times m}$ and $V_3 \in \mathbb{R}^{(2m+n) \times n}$. We can deduce from the special structure of the preconditioned system in (6.51) that

$$V_1 = \frac{1}{\sqrt{2}} \begin{pmatrix} I_m \\ I_m \\ 0 \end{pmatrix}, V_2 = \frac{1}{\sqrt{2}} \begin{pmatrix} I_m \\ -I_m \\ 0 \end{pmatrix}, \quad \text{and } V_3 = \begin{pmatrix} 0 \\ 0 \\ \tilde{V}_3 \end{pmatrix}$$

with $\tilde{V}_3 \in \mathbb{R}^{n \times n}$. MINRES in its preconditioned form iterates on vectors $\tilde{x}_k \in \mathcal{K}_k(K_3, \tilde{r}_0)$ for $k = 0, 1, \dots$, starting at the (transformed) initial residual $\tilde{r}_0 = (P_3^*)^{-1}r_0$. In each step k of

the iteration, MINRES minimizes

$$\|K_3 \tilde{x}_k - \tilde{r}_0\|_2 = \|(P_3^*)^{-1} K (P_3^*)^{-T} \cdot (P_3^*)^T x_k - (P_3^*)^{-1} r_0\|_2,$$

where $x_k \in \mathcal{K}_k(P_3^{*-1} P_3^{*-T} K, P_3^{*-1} P_3^{*-T} r_0)$ is a vector in the Krylov subspace spanned by $P_3^{*-1} P_3^{*-T} r_0$ and the matrix $P_3^{*-1} P_3^{*-T} K = (P_3^{*T} P_3^*)^{-1} K$. Using the notation $\hat{x} = V^T \tilde{x}$ and $\hat{r}_0 = V^T \tilde{r}_0$ we see that the requirement on the residual

$$\|K_3 \tilde{x} - \tilde{r}_0\|_2 \leq \epsilon \quad (6.53)$$

is equivalent to

$$\left\| \begin{pmatrix} I_m & 0 & 0 \\ 0 & -I_m & 0 \\ 0 & 0 & \Lambda \end{pmatrix} \hat{x} - \hat{r}_0 \right\|_2 \leq \epsilon. \quad (6.54)$$

If we denote by x^* the exact solution to the original system with K and r_0 , i.e.

$$r_0 = K x^*,$$

and analogously by \tilde{x}^* the exact solution to the linear system with K_3 and \tilde{r}_0 , so that

$$\tilde{r}_0 = K_3 \tilde{x}^* = (P_3^*)^{-1} K (P_3^*)^{-T} (P_3^*)^T x^*,$$

then we have

$$\hat{r}_0 = V^T \tilde{r}_0 = \begin{pmatrix} I_m & 0 & 0 \\ 0 & -I_m & 0 \\ 0 & 0 & \Lambda \end{pmatrix} V^T \tilde{x}^* \quad \text{with } \hat{x}^* = V^T \tilde{x}^*.$$

Therefore, (6.54) can be written as

$$\left\| \begin{pmatrix} I_m & 0 & 0 \\ 0 & -I_m & 0 \\ 0 & 0 & \Lambda \end{pmatrix} (\hat{x} - \hat{x}^*) \right\|_2 \leq \epsilon. \quad (6.55)$$

Since (6.55) and (6.53) are equivalent, the estimate (6.53) holds if and only if

$$\begin{aligned} |\hat{x}_i - \hat{x}_i^*| &\leq \epsilon & i = 1, \dots, 2m, \\ |\mu_i| |\hat{x}_i - \hat{x}_i^*| &\leq \epsilon & i = 2m + 1, \dots, 2m + n. \end{aligned} \quad (6.56)$$

The error is bounded by

$$|\hat{x}_i - \hat{x}_i^*| \leq \begin{cases} \epsilon & i = 1, \dots, 2m, \\ \frac{\epsilon}{|\mu_i|} & i = 2m + 1, \dots, 2m + n. \end{cases} \quad (6.57)$$

Introducing a partitioning similar to that of V we write for the error $\hat{e} = \hat{x} - \hat{x}^* = (\hat{e}_1, \hat{e}_2, \hat{e}_3)^T$ with $\hat{e}_1 \in \mathbb{R}^m$, $\hat{e}_2 \in \mathbb{R}^m$, $\hat{e}_3 \in \mathbb{R}^n$. The error $\tilde{e} = \hat{x} - \hat{x}^*$ is the error in coordinates transformed by V^T . For the error in the preconditioned MINRES-iterates $\tilde{x} = P_3^{*T} x$ we get

$$\tilde{e}_i = V \hat{e}_i = \begin{pmatrix} \frac{1}{\sqrt{2}}(\hat{e}_1 + \hat{e}_2) \\ \frac{1}{\sqrt{2}}(\hat{e}_1 - \hat{e}_2) \\ \tilde{V}_3 \hat{e}_3 \end{pmatrix}. \quad (6.58)$$

From this orthogonal transformation and (6.57) we can deduce the following about the size of the error in the components of the preconditioned iterates:

$$\|\tilde{e}_1\|_2 \leq \frac{1}{\sqrt{2}}(\|\hat{e}_1\|_2 + \|\hat{e}_1\|_2) \leq \epsilon \sqrt{2m}, \quad (6.59)$$

$$\|\tilde{e}_2\|_2 \leq \epsilon \sqrt{2m}, \quad (6.60)$$

$$\|\tilde{e}_3\|_2 = \|\hat{e}_3\|_2 \leq \epsilon \sqrt{\sum_{i=2m+1}^{2m+n} \frac{1}{\mu_i^2}}. \quad (6.61)$$

The error in the components \tilde{e}_1 and \tilde{e}_2 is of the order of the residual. The estimate (6.61) indicates that the error in the component \tilde{e}_3 is potentially much larger than the residual ϵ . The error in the coordinates for the original system is given by

$$e = x_k - x^* = (P_3^*)^{-T} \tilde{e} = \begin{pmatrix} I_m & 0 & -(A^{-1}B) \\ 0 & 0 & I_n \\ -1/2 A^{-T} H_y & A^{-T} & A^{-T} H_y (A^{-1}B) \end{pmatrix} \begin{pmatrix} \tilde{e}_1 \\ \tilde{e}_2 \\ \tilde{e}_3 \end{pmatrix}. \quad (6.62)$$

Partitioning this into the components $e_1 \in \mathbb{R}^m$, $e_2 \in \mathbb{R}^n$, $e_3 \in \mathbb{R}^m$, the error in the original coordinates is

$$e_1 = \tilde{e}_1 - A^{-1}B \tilde{e}_3, \quad (6.63)$$

$$e_2 = \tilde{e}_2, \quad (6.64)$$

$$\begin{aligned} e_3 &= -\frac{1}{2} A^{-T} H_y \tilde{e}_1 + A^{-T} \tilde{e}_2 + A^{-T} H_y A^{-1} B \tilde{e}_3 \\ &= A^{-T} \left(\tilde{e}_2 + H_y \left(A^{-1} B \tilde{e}_3 - \frac{1}{2} \tilde{e}_1 \right) \right). \end{aligned} \quad (6.65)$$

Chapter 7

Applications

In this section we consider an optimal control problem governed by partial differential equations. For the numerical solution, the partial differential equations are discretized using finite elements.

7.1 Neumann Control for an Elliptic Equation

As an example we consider the Neumann control for an elliptic equation which is given as follows:

$$\text{Minimize } \frac{1}{2} \int_{\Omega} (y(x) - y_d(x))^2 dx + \frac{\alpha}{2} \int_{\partial\Omega} u^2(x) ds \quad (7.1)$$

over all (y, u) satisfying the state equation

$$\begin{aligned} -\Delta y(x) + y(x) &= f(x) & x \in \Omega, \\ \frac{\partial}{\partial n} y(x) &= u(x) & x \in \partial\Omega. \end{aligned} \quad (7.2)$$

This and other control problems are studied in [12, Sec. II.2.4].

7.2 The Problem Discretization

We consider the weak formulation of (7.2). Given u in the control space $L^2(\partial\Omega)$, we seek y in the state space $H^1(\Omega)$ such that

$$\int_{\Omega} \nabla y(x) \nabla \varphi(x) dx + \int_{\Omega} y(x) \varphi(x) dx - \int_{\partial\Omega} u(x) \gamma(\varphi)(x) ds = \int_{\Omega} f(x) \varphi(x) dx \quad \forall \varphi \in H^1(\Omega). \quad (7.3)$$

This is called the weak formulation of (7.2). We replace (7.2) by (7.3). In (7.3), the function γ denotes the trace operator, defining the restriction of φ on $\partial\Omega$.

For the numerical solution of the optimal control problem we apply a finite element discretization using a quasi-uniform triangulation and piecewise linear basis functions.

Let $\Omega = \cup_{i=1}^N T_i$ be a triangulation. As usual, we let h_T denote the diameter of the triangle T and we define $h = \max_{T \in \{T_i\}} h_T$.

Let m denote the (total) number of vertices in the triangulation and let n be the number of vertices on the boundary. Let $l_1, \dots, l_n \in \{1, \dots, m\}$ be the indices of boundary vertices. For example, for the grid in Figure 7.1 we have that $m = 36$, $n = 20$, and

$$l_1, \dots, l_n = 1, 2, 3, 4, 5, 6, 7, 12, 13, 18, 19, 24, 25, 30, 31, 32, 33, 34, 35, 36.$$

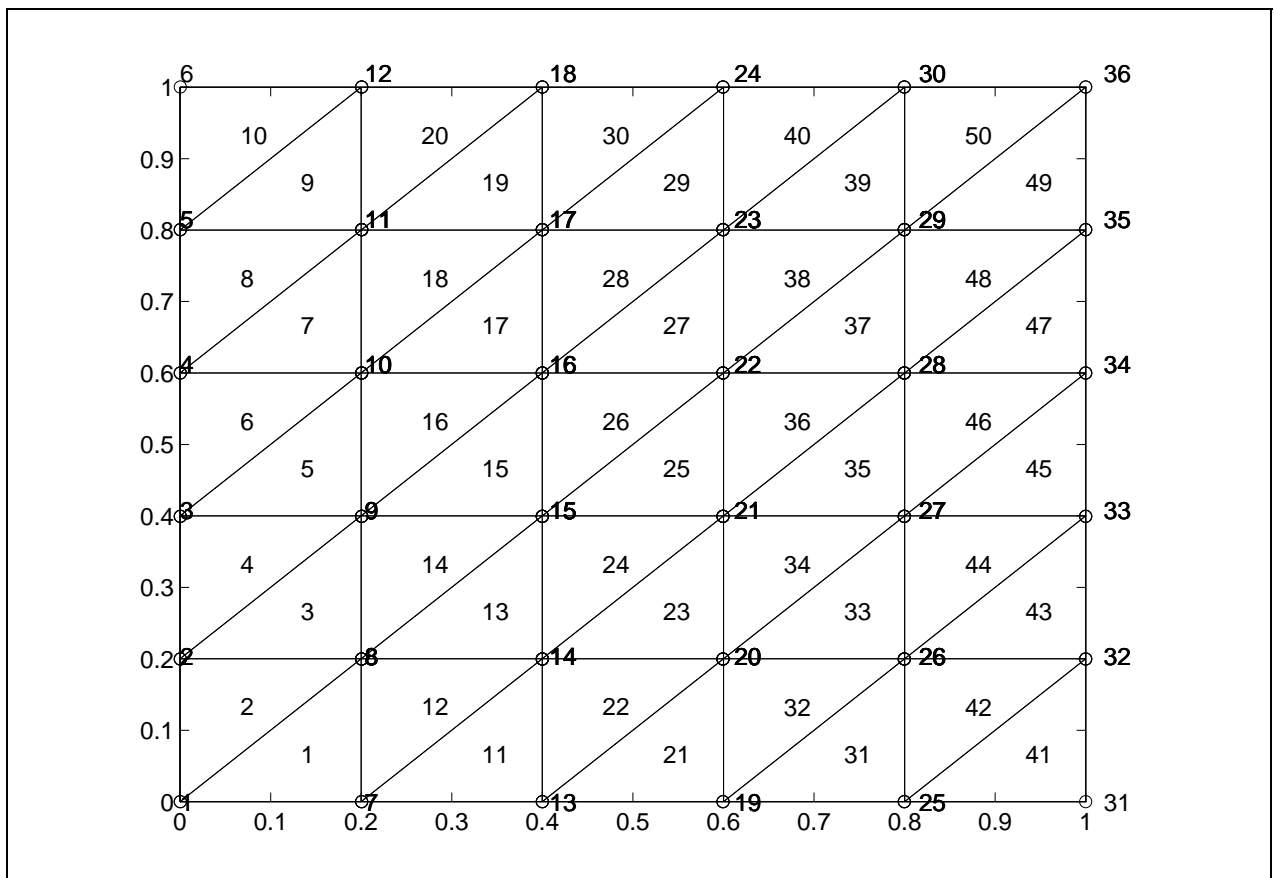


Figure 7.1: The grid for $n_x = n_y = 5$.

Moreover, let φ_i be the piecewise linear function with $\varphi_i(x_j) = \delta_{ij}$ for all vertices x_j , $j = 1, \dots, m$, and let $\hat{\varphi}_i$ be the piecewise linear function defined on the boundary $\partial\Omega$ with $\hat{\varphi}_i(x_{l_j}) = \delta_{ij}$ for all boundary vertices x_{l_j} , $j = 1, \dots, n$. Notice that if x_{l_i} is a boundary node, then $\hat{\varphi}_i = \gamma(\varphi_{l_i})$.

If φ_i , $i = 1, \dots, m$, are the basis functions we set

$$v_h = \sum_{i=1}^m v_i \varphi_i.$$

Moreover, for basis functions $\widehat{\varphi}_i$, $i = 1, \dots, n$, defined on $\partial\Omega$ we set

$$\widehat{v}_h = \sum_{i=1}^n \widehat{v}_i \widehat{\varphi}_i.$$

In our computations we use quasi-uniform triangulations of the domain $\Omega = (0, 1)^2$ which are constructed as follows. The intervals $(0, 1)$ on the x and y axis are subdivided into n_x and n_y subintervals, respectively. The resulting $n_x n_y$ subsquares are each subdivided into two triangles. Hence, the diameter of all triangles are equal and given by

$$h = \sqrt{n_x^{-2} + n_y^{-2}}.$$

The construction can be seen in Figure 7.1.

The unknown functions y and u are approximated using piecewise linear functions y_h, u_h , respectively,

$$y_h(x) = \sum_{i=1}^m y_i \varphi_i(x), \quad u_h(x) = \sum_{i=1}^n u_i \widehat{\varphi}_i(x).$$

The weak formulation (7.3) becomes

$$\begin{aligned} \int_{\Omega} \left(\sum_{j=1}^m y_j \nabla \varphi_j(x) \right) \nabla \varphi_i(x) dx + \int_{\Omega} \left(\sum_{j=1}^m y_j \varphi_j(x) \right) \varphi_i(x) dx \\ - \int_{\partial\Omega} \left(\sum_{j=1}^n u_j \widehat{\varphi}_j(x) \right) \gamma(\varphi_i)(x) ds \\ = \int_{\Omega} f(x) \varphi_i(x) dx \quad \forall i = 1, \dots, m. \end{aligned} \quad (7.4)$$

Instead of varying φ over all test functions, we now consider only the basis functions φ_i , $i = 1, \dots, m$. If we define the matrices

$$\begin{aligned} A &= \left(\int_{\Omega} \nabla \varphi_j \nabla \varphi_i dx + \int_{\Omega} \varphi_j \varphi_i dx \right)_{1 \leq i, j \leq m}, \\ B &= \left(- \int_{\partial\Omega} \widehat{\varphi}_j \varphi_i dx \right)_{1 \leq i \leq m, 1 \leq j \leq n} \end{aligned}$$

and the vector

$$b = \left(\int_{\Omega} f \varphi_i dx \right)_{1 \leq i \leq m},$$

then the weak formulation (7.3) can be written in the form

$$Ay + Bu = b.$$

The objective function in (7.1) is equal to

$$\frac{1}{2} \int_{\Omega} y(x)^2 dx - \int_{\Omega} y(x) y_d(x) dx + \frac{\alpha}{2} \int_{\partial\Omega} u^2(x) ds + \frac{1}{2} \int_{\Omega} y_d(x)^2 dx.$$

Since the minimizer is not affected by the constant term $\frac{1}{2} \int_{\Omega} y_d(x)^2 dx$, it is omitted. Using y_h and u_h instead of y, u this leads to the following discretization of the objective function.

$$\frac{1}{2} \int_{\Omega} \sum_{i,j=1}^m (y_i y_j \varphi_i(x) \varphi_j(x)) dx - \int_{\Omega} \sum_{i=1}^m (y_i \varphi_i(x)) y_d(x) dx + \frac{\alpha}{2} \int_{\Omega} \sum_{i,j=1}^n (u_i u_j \widehat{\varphi}_i(x) \widehat{\varphi}_j(x)) dx$$

If we define the matrices

$$\begin{aligned} M_y &= \left(\int_{\Omega} \varphi_i \varphi_j dx \right)_{1 \leq i, j \leq m}, \\ M_u &= \left(\alpha \int_{\partial\Omega} \widehat{\varphi}_i \widehat{\varphi}_j dx \right)_{1 \leq i, j \leq n}, \\ M_{yu} &= M_{uy}^T = 0, \end{aligned}$$

and the vectors

$$c = \left(\int_{\Omega} y_d \varphi_i dx \right)_{1 \leq i \leq m}, \quad d = 0_n,$$

then (7.4) can be written as

$$\frac{1}{2} y^T M_y y + \frac{\alpha}{2} u^T M_u u + c^T y + d^T u. \quad (7.5)$$

Combining this with the discretization of the constraints, we obtain the discretized problem

$$\text{Minimize } \frac{1}{2} y^T M_y y + \frac{\alpha}{2} u^T M_u u + c^T y + d^T u \quad (7.6)$$

subject to

$$Ay + Bu = b. \quad (7.7)$$

7.3 Eigenvalues of FEM Matrices

For matrices arising from finite element discretizations of partial differential equations, certain bounds for eigenvalues are known. See, for example, [11, Lemma 7.3].

Some results that are interest for us will be collected in this section. We do not give any proofs in this section. The results can be found – in a more general framework – in, for example, [11, Lemma 7.3]. There the proofs are provided as well.

Let $\Omega = (0, 1)^2$ be the unit square and let $\{\mathcal{T}_h\}$ be a family of triangulations of Ω , i.e. $\mathcal{T}_h = \{T_i\}$. Let h_T denote the largest edge of the triangle T and let ρ_T denote the diameter

of the largest circle contained in T . Suppose that there exist constants β_1, β_2 independent of $h = \max_{T \in \{T_i\}} h_T$ such that for all $T \in \mathcal{T}_h$ and all h ,

$$h_T \geq \beta_1 h, \quad (7.8)$$

$$\frac{\rho_T}{h_T} \geq \beta_2. \quad (7.9)$$

We can give constants β_1 and β_2 such that (7.8) and (7.9) are satisfied for our application. This is shown at the end of Section 7.3.

Moreover, let $\varphi_i, i = 1, \dots, m$, be piecewise linear functions on each T_i . We set

$$v_h = \sum_{i=1}^M v_i \varphi_i.$$

Lemma 7.3.1 *Let $\varphi_i, i = 1, \dots, m$, be functions that are piecewise linear on each T_i and set*

$$v_h = \sum_{i=1}^M v_i \varphi_i.$$

If (7.8) and (7.9) are satisfied, then there exist constants c_1, c_2, c_3 such that

$$c_1 h^2 \sum_{i=1}^M v_i^2 \leq \int_{\Omega} v_h^2(x) dx \leq c_2 h^2 \sum_{i=1}^M v_i^2, \quad (7.10)$$

and

$$\|\nabla v_h\|_{L^2(\Omega)}^2 \leq c_3 h^{-2} \|v_h\|_{L_2(\Omega)}^2. \quad (7.11)$$

Lemma 7.3.2 *Let $\hat{\varphi}_i, i = 1, \dots, n$, be functions defined on $\partial\Omega$ that are piecewise linear on each $\partial\Omega \cap T_i$ and set*

$$v_h = \sum_{i=1}^K v_i \hat{\varphi}_i.$$

If (7.8) and (7.9) are satisfied, then there exist constants c_4, c_5, c_6 such that

$$c_4 h \sum_{i=1}^K v_i^2 \leq \int_{\partial\Omega} v_h^2(x) dx \leq c_5 h \sum_{i=1}^K v_i^2. \quad (7.12)$$

In our computations we use grids of the form shown in Figure 7.1. In this case all triangles are congruent and we have that

$$h_T = \sqrt{\frac{1}{n_x^2} + \frac{1}{n_y^2}} \quad \forall T \in \mathcal{T}_h.$$

If $n_x = n_y$,

$$h_T = \frac{\sqrt{2}}{n_x} \quad \forall T \in \mathcal{T}_h.$$

Moreover, if $n_x = n_y$, the circle with center $(\frac{1}{4n_x}, \frac{1}{4n_x})$ is contained in the triangle with vertices $(0, 0)$, $(0, \frac{1}{n_x})$, and $(\frac{1}{n_x}, 0)$. Its diameter is $1/2n_x$. This yields

$$\frac{\rho_T}{h_T} \geq \frac{n_x}{2n_x\sqrt{2}} = \frac{1}{2\sqrt{2}}.$$

Hence, in our computations, the estimates (7.8) and (7.9) are satisfied with

$$\beta_1 = 1, \quad \beta_2 = \frac{1}{2\sqrt{2}}.$$

7.4 Condition Number of the KKT–System

With the help of the estimates for eigenvalues of matrices arising from the discretization of partial differential equations we collected in Section 7.3, we can now derive estimates for the eigenvalues of the matrices we are interested in.

It is well known that the matrices A , M_y , and M_u are positive definite. To see this consider for example that

$$\begin{aligned} y^T A y &= \sum_{i,j=1}^m \left(y_j \left(\int_{\Omega} \nabla \varphi_j \nabla \varphi_i dx + \int_{\Omega} \varphi_j \varphi_i dx \right) y_i \right) \\ &= \int_{\Omega} \left(\sum_{i,j=1}^m y_j \nabla \varphi_j \nabla \varphi_i y_i \right) dx + \int_{\Omega} \left(\sum_{i,j=1}^m y_j \varphi_j \varphi_i y_i \right) dx \\ &= \int_{\Omega} (\nabla y_h)^2 dx + \int_{\Omega} (y_h)^2 dx \geq 0. \end{aligned}$$

Moreover, due to the construction of the discretization, the singular values of B are equal to the eigenvalues of M_u .

Lemma 7.4.1 *The singular values σ_i^B of B are given by $\sigma_i^B = \lambda_i^u$, where λ_i^u are the eigenvalues of M_u .*

Proof: The definition of the basis functions shows that

$$\begin{aligned} \varphi_{l_i}|_{\partial\Omega} &= \widehat{\varphi}_i & i = 1, \dots, n, \\ \varphi_i|_{\partial\Omega} &= 0 & i \notin \{l_1, \dots, l_n\}. \end{aligned}$$

Hence,

$$B = \begin{pmatrix} M_u \\ 0 \end{pmatrix}.$$

Thus, if $M_u = V D V^T$ where D is the diagonal matrix of (positive) eigenvalues and V is the orthogonal matrix of eigenvectors, then

$$B = \begin{pmatrix} V & 0 \\ 0 & I \end{pmatrix} \begin{pmatrix} D \\ 0 \end{pmatrix} V^T.$$

□

Using the results (7.10) for M_y and (7.12) for M_u , we can write

$$c_1 h^2 \|y\|^2 \leq y^T M_y y \leq c_2 h^2 \|y\|^2, \quad (7.13)$$

$$c_4 h \|u\|^2 \leq u^T M_u u \leq c_5 h \|u\|^2. \quad (7.14)$$

In order to get estimates for A we use (7.10) and (7.11). With these we get the estimates

$$\begin{aligned} c_1 h^2 \|y\|^2 &\leq y^T A y \\ &= \int_{\Omega} \sum_{i=1}^m y_i \nabla \varphi_i \sum_{j=1}^m y_j \nabla \varphi_j dx + \int_{\Omega} \sum_{i=1}^m y_i \varphi_i \sum_{j=1}^m y_j \varphi_j dx \\ &= \int_{\Omega} \nabla y_h^2 + \int_{\Omega} y_h^2 \\ &\leq c_2 h^2 \|y\|^2 + \|\nabla y_h^2\|_{L_2(\Omega)} \\ &\leq c_2 h^2 \|y\|^2 + c_3 h^{-2} \|y_h\|_{L_2(\Omega)}^2 \\ &\leq c_2 h^2 \|y\|^2 + c_3 c_2 \|y\|^2. \end{aligned} \quad (7.15)$$

So, gathering (7.13), (7.14), and (7.15), we find that

$$\Lambda(M_y) \subset [c_1 h^2, c_2 h^2], \quad (7.16)$$

$$\Lambda(M_u) \subset [c_4 h, c_5 h], \quad (7.17)$$

$$\Lambda(A) \subset [c_1 h^2, c_2 h^2 + c_2 c_3]. \quad (7.18)$$

To estimate the singular values $\sigma_1 \geq \dots \geq \sigma_m > 0$ of $(A | B)$ we observe that

$$\begin{aligned} \|Ay + Bu\|^2 &\leq 2\|Ay\|^2 + 2\|Bu\|^2 \leq 2(c_2 h^2 + c_2 c_3)^2 \|y\|^2 + 2(c_5 h)^2 \|u\|^2 \\ &\leq 2 \max\{(c_2 h^2 + c_2 c_3)^2, (c_5 h)^2\} (\|y\|^2 + \|u\|^2). \end{aligned}$$

Hence,

$$\sigma_1 \leq \sqrt{2 \max\{(c_2 h^2 + c_2 c_3)^2, (c_5 h)^2\}}. \quad (7.19)$$

Moreover, from the inequality

$$\left\| \begin{pmatrix} A^T y \\ B^T y \end{pmatrix} \right\| \geq \|A^T y\| \geq c_1 h^2 \|y\|$$

we can deduce that

$$\sigma_m \geq c_1 h^2. \quad (7.20)$$

If the estimates (7.16), (7.17), (7.19), and (7.20) would be sharp, then, for small h and $c_4 h \geq c_1 h^2$, we obtain the following estimates for the bounds on the spectrum presented in Theorem 3.2.1.

If $c_4 h \geq c_1 h^2$, then

$$\begin{aligned} \lambda_{2m+n} &\geq \frac{1}{2}(\mu_{m+n} - \sqrt{\mu_{m+n}^2 + 4\sigma_1^2}) \approx -\sqrt{2}c_2c_3, \\ \lambda_{m+n+1} &\leq \frac{1}{2}(\mu_1 - \sqrt{\mu_1^2 + 4\sigma_n^2}) \approx \frac{1}{2} \left(1 - \sqrt{1 + 4\frac{c_1^2}{c_5^2} h^2}\right) c_5 h \approx -\frac{c_1^2}{c_5} h^3, \\ \lambda_{m+n} &\geq \mu_{m+n} \approx c_1 h^2, \\ \lambda_1 &\leq \frac{1}{2}(\mu_1 + \sqrt{\mu_1^2 + 4\sigma_1^2}) \approx c_2 c_3. \end{aligned}$$

Here the estimate for λ_{m+n+1} is correct because for small x we have

$$1 - \sqrt{1+x^2} = \frac{(1 - \sqrt{1+x^2})(1 + \sqrt{1+x^2})}{(1 + \sqrt{1+x^2})} \approx \frac{1 - (1+x^2)}{2} = -\frac{x^2}{2}.$$

If $c_4 h < c_1 h^2$, then

$$\begin{aligned} \lambda_{m+n+1} &\leq \frac{1}{2}(\mu_1 - \sqrt{\mu_1^2 + 4\sigma_n^2}) \approx \frac{1}{2} \left(1 - \sqrt{1 + 4\frac{c_1^2}{c_2^2} h^2}\right) c_2 h^2, \\ \lambda_{m+n} &\geq \mu_{m+n} \approx c_4 h. \end{aligned}$$

From (7.16), (7.17), (7.18) and Lemma 7.4.1 we find that

$$\|M_y^{1/2} A^{-1} B M_u^{-1/2}\| \leq \|M_y^{1/2}\| \|A^{-1}\| \|B\| \|M_u^{-1/2}\| \leq c h^{-1/2} \quad (7.21)$$

for some constant c independent of h . This inequality corresponds to (6.21). However, unlike the estimate in (6.21), here the bound depends on $h^{1/2}$ and goes to infinity if h goes to zero. This behaviour could not be observed in our numerical results, see Table 7.8. A more detailed analysis of the norm $\|M_y^{1/2} A^{-1} B M_u^{-1/2}\|$ using Sobolev space estimates, is given in [2].

We now turn to the numerical results we obtained for this problem. All computations are done using Matlab on a Sun Sparcstation.

7.5 Numerical Results without a Preconditioner

In this section we collect the numerical results in the unpreconditioned case.

Case 1: $\alpha = 1$, $D_y = 0$, $D_u = 0$

Table 7.1 shows the computed spectrum of K and the estimate of the spectrum using Theorem 3.2.1. Table 7.1 confirms that for $\alpha = 1$, $D_y = 0$ and $D_u = 0$ the outer bounds for the spectrum are constant, while the inner bounds depend on the mesh constant. This was derived in Section 7.1. The estimate according to Theorem 3.2.1 is in general good, but differs from the computed eigenvalues of the original system for the negative eigenvalues that are small in absolute value. The eigenvalues and singular values of the submatrices that build the system are shown in Figure 7.2. For a grid with $n_x = n_y = 20$, the mesh constant is $h = 7.07 \cdot 10^{-2}$. We have seen in (7.16) that the spectrum of H_u can be described as $\Lambda(H_u) = \mathcal{O}(h)$, and for H_y it holds $\Lambda(H_y) = \mathcal{O}(h^2)$. The eigenvalues for the Karush-Kuhn-Tucker matrix are plotted in Figure 7.3. The estimates for the spectrum $\Lambda(K)$ are accurate except for the bound on the small positive eigenvalues. The condition number of the system which is of order 10^3 , and those of the submatrices are given in Table 7.3. For the original system, the iterations needed by MINRES and SYMMLQ grow with the mesh size. The dependence on the mesh size is induced by H_y and H_u . We have stated that essentially $\Lambda(H_y) = \mathcal{O}(h^2)$, so that for the smallest eigenvalue μ_{min} arising in (7.4) we have $\mu_{min} \approx h^2$.

We stop the iterative process if either the residual is smaller than 10^{-5} , or if the iteration number exceeds $2m + n$, which is the dimension of the system and the maximum number of steps MINRES and SYMMLQ take until they encounter the exact solution. We give the dimensions of our systems in the tables together with the iteration count. In Figure 7.4 the residual of the MINRES- and SYMMLQ-iterates are shown.

Case 2: $\alpha \ll 1$, $D_y = 0$, $D_u = 0$

If we have a regularization parameter α that is 'small enough', MINRES and SYMMLQ can no longer solve the original system in less than $2m + n$ steps. The iterative process is in this case always stopped with the maximal number of iterations. It depends on the size of the system what 'small enough' means. Our numerical experiments show that the larger the matrices are the better MINRES and SYMMLQ can cope with regularization parameters α around $10^{-1}, 10^{-2}$. But it is obvious that the conditioning of the system deteriorates considerably under the influence of a factor $\alpha \ll 1$. The numerical experiments confirm the analysis in Section 6.1.

In our analysis we distinguish four different cases. We motivated Cases 3 and 4 where we consider large diagonal entries in H_y and H_u with the action of interior-point methods on the system. We did not apply an interior-point method, but 'simulated' the action of an interior-point method in adding large diagonal entries to the respective diagonals of H_y and H_u . We consider as 'large' entries values of order 10^4 because the entries of H_y and H_u are in general smaller than 1.

(The estimated spectrum is computed using Theorem 3.2.1.
In all computations, $n_x = n_y$.)

n_x	h	Computed Spectrum					Estimated Spectrum				
5	2.82e-1	-7.37e+0	-8.53e-2	3.17e-2	7.39e+0	-7.38e+0	-8.18e-2	2.73e-3	7.48e+0		
10	1.41e-1	-7.82e+0	-2.78e-2	9.62e-3	7.83e+0	-7.83e+0	-2.63e-2	6.82e-4	7.88e+0		
20	7.07e-2	-7.95e+0	-8.24e-3	2.67e-3	7.95e+0	-7.95e+0	-7.58e-3	1.70e-4	7.98e+0		
30	4.71e-2	-7.98e+0	-3.92e-3	1.23e-3	7.98e+0	-7.98e+0	-3.51e-3	7.58e-5	8.00e+0		

Table 7.1: Computed and estimated spectrum of K with $\alpha = 1$, $D_y = 0$, $D_u = 0$.

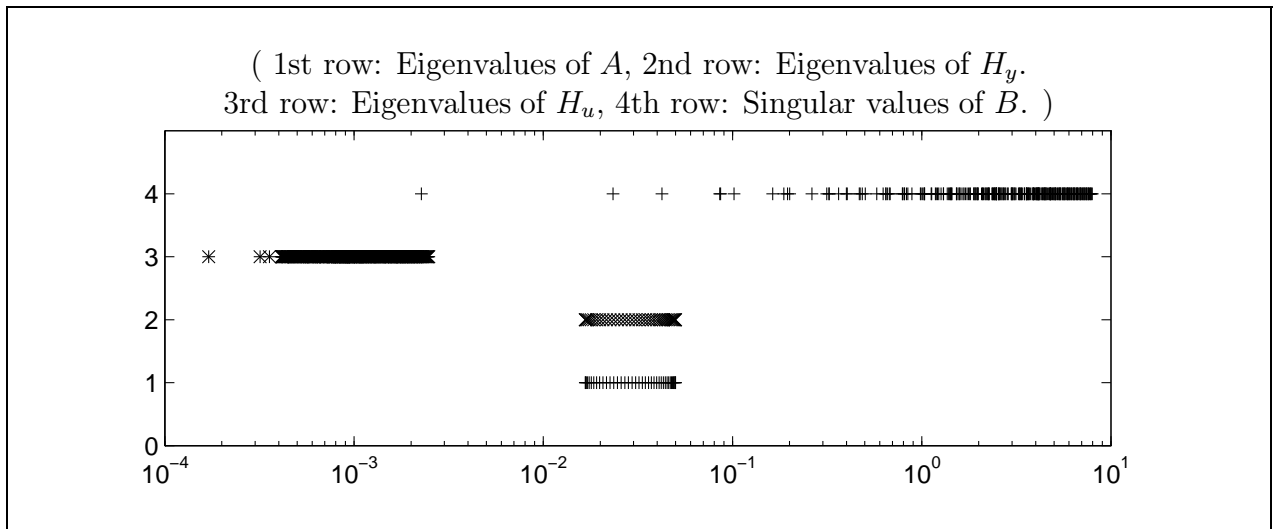


Figure 7.2: The eigenvalues and singular values of the submatrices in K for $n_x = n_y = 20$ and $\alpha = 1$, $D_y = 0$, $D_u = 0$.

(Positive eigenvalues of K are denoted by '+'.
 Negative eigenvalues of K , given in absolute value, are denoted by '*'.
 The lines denote the estimate for the positive and negative parts of the spectrum.)

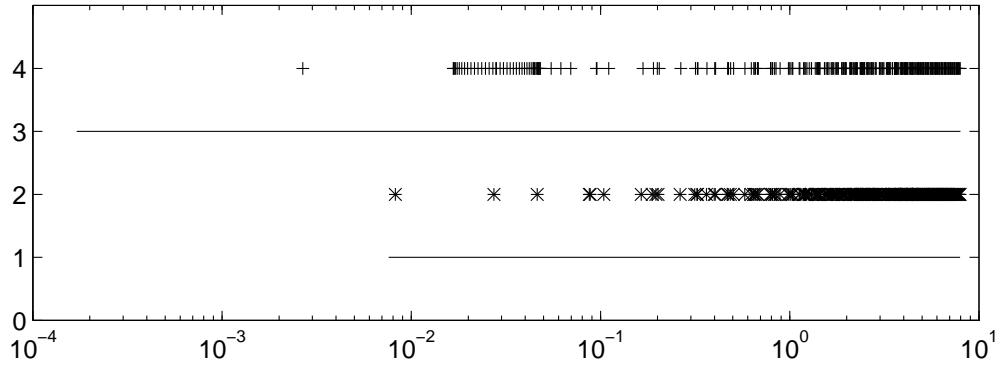


Figure 7.3: The eigenvalues of the KKT-system for $n_x = n_y = 20$ and $\alpha = 1$, $D_y = 0$, $D_u = 0$.

(In all computations, $n_x = n_y$.)

grid size	5	10	15	20	25	30
dimension	92	282	572	962	1452	2042
MINRES	47	185	431	784	1070	1483
SYMMLQ	47	179	407	647	902	1209

Table 7.2: Iterations of MINRES and SYMMLQ on K with $\alpha = 1$, $D_y = 0$, $D_u = 0$.

(In all computations, $n_x = n_y$.)

grid size	5	10	15	20	25	30
K	2.32e+2	8.13e+2	1.72e+3	2.98e+3	4.56e+3	6.49e+3
H_y	1.33e+1	1.43e+1	1.45e+0	1.46e+1	1.46e+1	1.46e+1
H_u	3.00e+0	3.00e+0	3.00e+0	3.00e+0	3.00e+0	3.00e+0
B	3.00e+0	3.00e+0	3.00e+0	3.00e+0	3.00e+0	3.00e+0
A	2.67e+2	9.48e+2	2.03e+3	3.51e+3	5.39e+3	7.67e+3

Table 7.3: Condition numbers of the system K and the submatrices for different grid sizes.

(In all computations, $n_x = n_y$.)						
grid size	5	10	15	20	25	30
dimension	92	282	572	962	1452	2042
MINRES	10^{-2}	10^{-2}	10^{-3}	10^{-3}	10^{-3}	10^{-4}
SYMMLQ	10^{-3}	10^{-3}	10^{-3}	10^{-3}	10^{-4}	10^{-4}

Table 7.4: Largest value of α for that MINRES and SYMMLQ can no longer compute a solution to the system with K within the required accuracy in less than $2m + n$ steps.

Case 3: $\alpha = 1$, $D_y = 0$, $D_u = 10^4 \cdot I$

If the diagonal of H_u is increased by 10^4 , this constitutes no problem for MINRES and SYMMLQ. In fact, even less iterations are necessary to compute a solution with the required accuracy with the required accuracy than in Case 1. The iteration numbers for Case 3 are given in Table 7.5. The changes in the spectrum of K and in the eigenvalue distribution of the submatrices are visible in Figures 7.5 and 7.6. The matrix H_u now only has the multiple eigenvalue 10^4 , and K has an additional eigenvalue at 10^4 .

Case 4: $\alpha = 1$, $D_y = 10^4 \cdot I$, $D_u = 0$

If the diagonal of H_y is increased by the same amount as the diagonal of H_u in Case 3, the situation turns out to be much worse. We have seen in Section 2 that this case can correspond to the degenerate case in linear programming, and we expect deterioration in the performance of MINRES and SYMMLQ. In fact, the iterative solvers need the maximal number of steps for all grids but the smallest. The necessary iterations are given in Table 7.6. The eigenvalue distribution in this case is shown in Figure 7.7. It is changed considerably with respect to the distribution in Case 1. In addition to the newly introduced eigenvalue located at 10^4 , the negative eigenvalues of the system move towards zero.

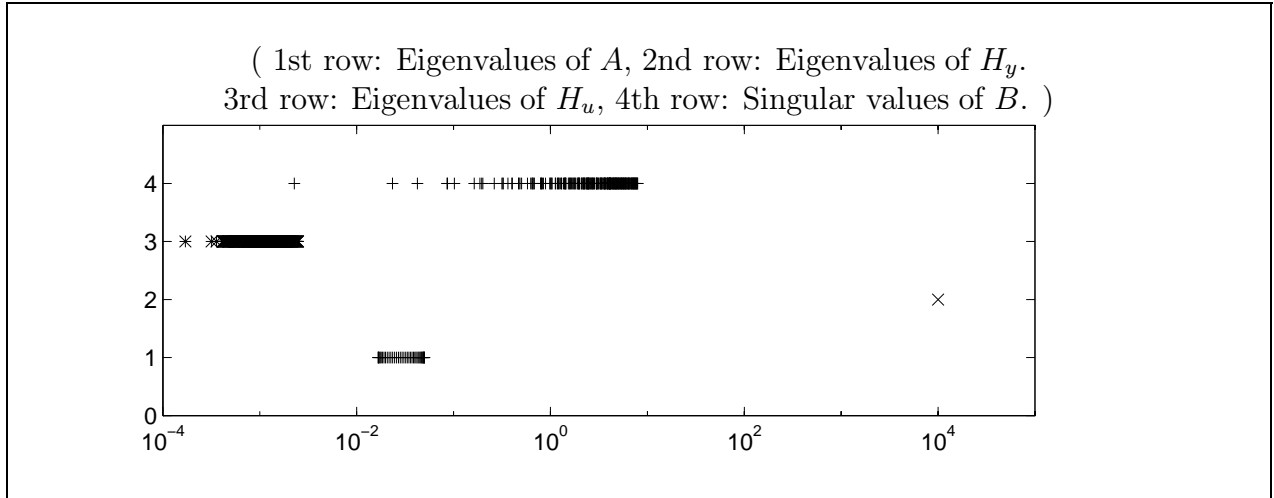


Figure 7.5: The eigenvalues and singular values of the submatrices before preconditioning for a grid $n_x = n_y = 20$ with $D_u = 10^4 \cdot I$, $D_v = 0$, $\alpha = 1$.

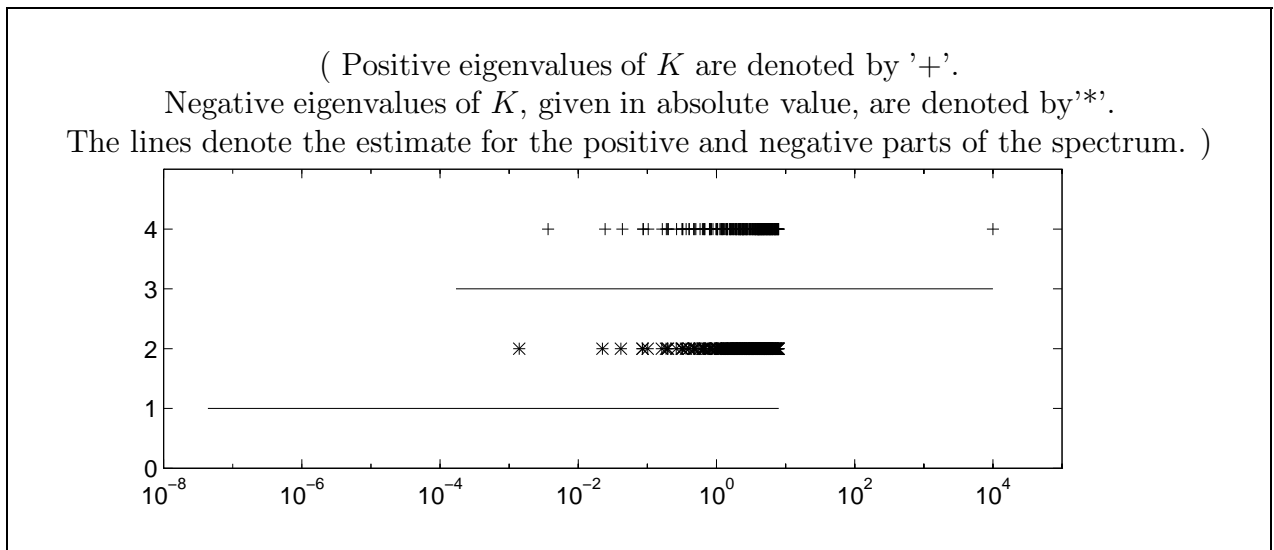


Figure 7.6: The eigenvalues of the KKT-system before preconditioning for $n_x = n_y = 20$ with $D_u = 10^4 \cdot I$, $D_v = 0$, $\alpha = 1$.

(In all computations, $n_x = n_y$.)

grid size	5	10	15	20	25	30
dimension	92	282	572	962	1452	2042
MINRES	54	173	349	589	857	1183
SYMMLQ	54	173	349	579	848	1165

Table 7.5: Iterations of MINRES and SYMMLQ for K with $\alpha = 1$ and $D_u = 10^4 \cdot I$, $D_y = 0$.

(In all computations, $n_x = n_y$.)

grid size	5	10	15	20	25	30
dimension	92	282	572	962	1452	2042
MINRES	73	282	572	962	1452	2042
SYMMLQ	73	282	572	962	1452	2042

Table 7.6: Iterations of MINRES and SYMMLQ for K with $\alpha = 1$ and $D_y = 10^4 \cdot I$, $D_u = 0$.

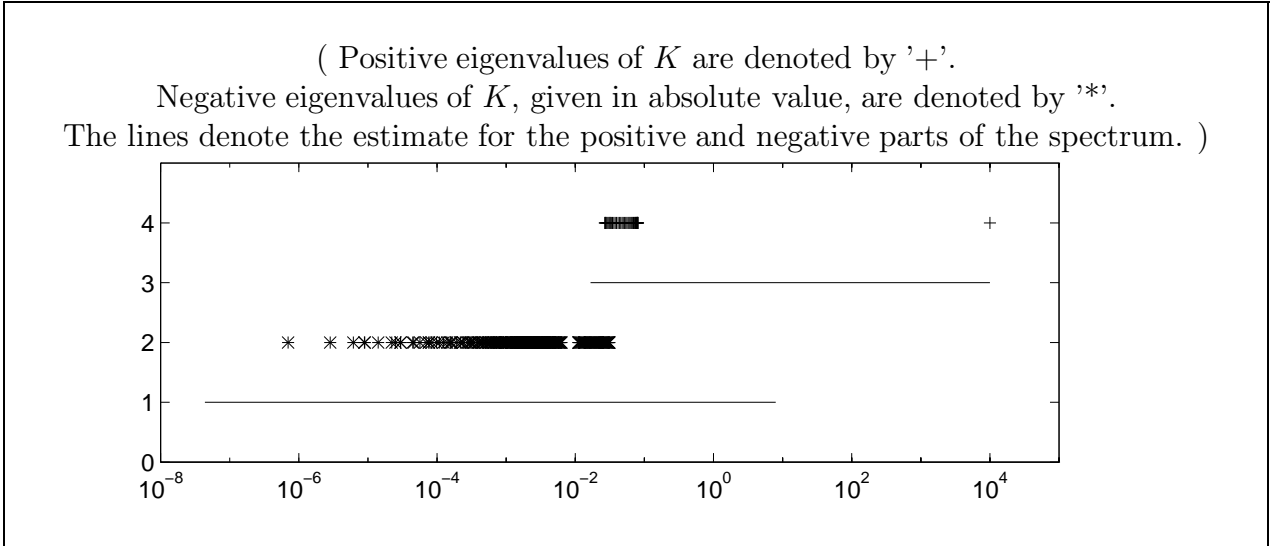


Figure 7.7: The eigenvalues of the KKT-system before preconditioning for $n_x = n_y = 20$ with $D_y = 10^4 \cdot I$, $D_u = 0$, $\alpha = 1$.

(First diagram: Residuals of the iterates.
 Second diagram: The absolute error in the components of the solution vector.
 Third diagram: The relative error in the components of the solution vector.)

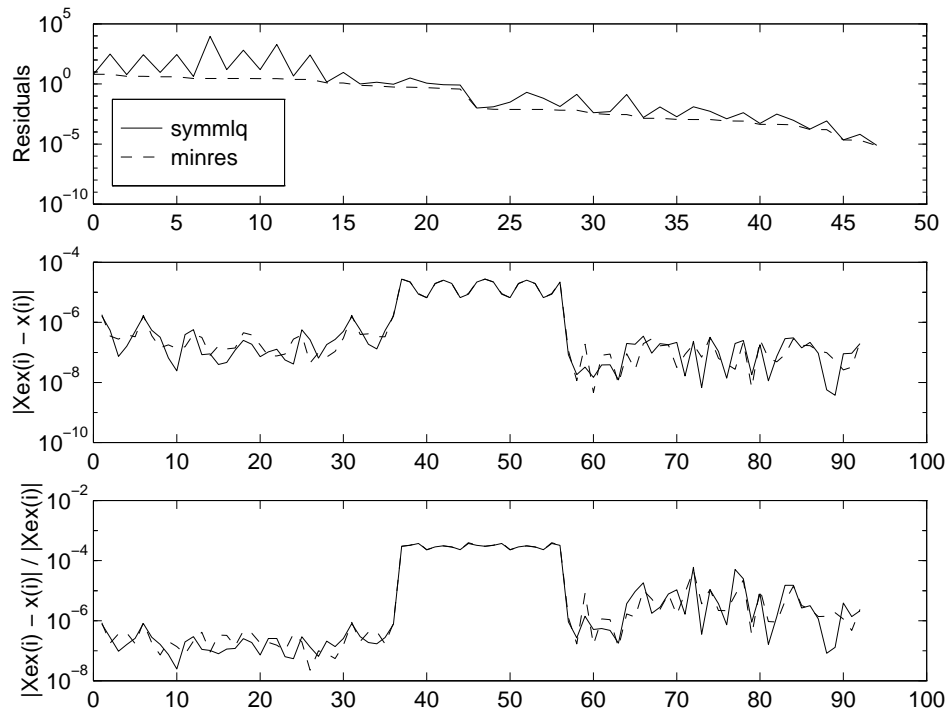


Figure 7.4: The residuals, the absolute and the relative error of MINRES- and SYMMLQ-iterates on the system K for $n_x = n_y = 5$ with $D_y = 0$, $D_u = 0$, $\alpha = 1$.

7.6 Numerical Results with the First Preconditioner

Our first preconditioner is given by

$$P_1 = \begin{pmatrix} P_y & 0 & 0 \\ 0 & P_u & 0 \\ 0 & 0 & AP_y^{-1} \end{pmatrix},$$

where $P_y = \text{diag}(H_y)^{1/2}$ and $P_u = \text{diag}(H_u)^{1/2}$ denote the square roots of the diagonals of H_y and H_u , respectively. They are a good enough approximation to the matrices, transforming the spectra of H_y and H_u such that the spectra of the preconditioned matrices $P_y^{-1}H_yP_y^{-T}$ and $P_u^{-1}H_uP_u^{-T}$ are bounded independently of the mesh constant h or the parameter α . We use the sparse LU -factorization of A to solve the systems with P_1 and P_1^T . The preconditioned KKT-matrix is

$$P_1^{-1}KP_1^{-T} = \begin{pmatrix} P_y^{-1}H_yP_y^{-T} & 0 & I \\ 0 & P_u^{-1}H_uP_u^{-T} & P_u^{-1}B^T A^{-T} P_y \\ I & P_y^T A^{-1} B P_u^{-T} & 0 \end{pmatrix}.$$

Case 1: $\alpha = 1$, $D_y = 0$, $D_u = 0$

In Section 6.2 we derived the bounds

$$\begin{aligned} \lambda_{2m+n} &\geq \frac{1}{2}(\mu_{\min} - \sqrt{5 + 4\sigma_{\max}^2}), \\ \lambda_{m+n+1} &\leq \frac{1}{2}(\mu_{\max} - \sqrt{5 + 4\sigma_{\min}^2}), \\ \lambda_{m+n} &\geq \mu_{\min}, \\ \lambda_1 &\leq \frac{1}{2}(\mu_{\max} + \sqrt{5 + 4\sigma_{\max}^2}) \end{aligned}$$

for the first preconditioner. Here $\sigma_{\min}, \sigma_{\max}$ denote the extreme singular values of $P_y^T A^{-1} B P_u^{-T}$. The values μ_{\min}, μ_{\max} are the extreme eigenvalues of the preconditioned matrices in the upper left part of the system. Since we do not use $H_y^{-1/2}$ and $H_u^{-1/2}$ as preconditioners, but the square roots of the respective diagonals of H_y, H_u , so that we get $P_y^{-1}H_yP_y^{-T} \approx I$, $P_u^{-1}H_uP_u^{-T} \approx I$ and for their eigenvalues $\mu \approx 1$, we cannot simply replace the values μ_{\min}, μ_{\max} by 1. We have computed the eigenvalues of $P_y^{-1}H_yP_y^{-T}$ and $P_u^{-1}H_uP_u^{-T}$ and we have seen that for this example they lie in the interval $[0.5, 2]$. The largest negative eigenvalue λ_{m+n+1} of K is bounded independently of h . Combining this with the bound for the smallest positive eigenvalue λ_{m+n} of K we see that the eigenvalues of the preconditioned system are bounded away from zero. Application of the preconditioner P_1 transforms the system K into an equivalent system with a condition number that is independent of the mesh constant. We can state this because the eigenvalues large in absolute value are bounded independently of h as well. This can be deduced from the existence of an upper bound on the singular values on $P_y^T A^{-1} B P_u^{-T}$ as it is given in (6.21) and (7.21).

The estimated spectrum and the computed bounds are given in Table 7.7. The eigenvalue distribution of the preconditioned submatrices is shown in Figure 7.8. We see that the diagonal matrices P_y, P_u act well as preconditioners in coalescing the spectra of H_y, H_u such that

$$\Lambda(P_u^{-1}H_uP_u^{-T}) \subset [0.5, 2] \quad \text{and} \quad \Lambda(P_y^{-1}H_yP_y^{-T}) \subset [0.5, 2].$$

The largest singular value of $P_y^T A^{-1} B P_u^{-T}$ is smaller than 2, and the small singular values move towards the origin. The eigenvalues of the preconditioned system K for a grid with $n_x = n_y = 20$ are plotted in Figure 7.9. The spectrum of $P_1^{-1} K P_1^{-T}$ is shrunk considerably with respect to that of the original system K . The action of the preconditioner on the system reduces the condition number of the system from 10^3 to a number smaller than 10. The condition numbers of the preconditioned system and of the submatrices are given in Table 7.8.

These results show that we can expect a good performance of the Krylov subspace methods MINRES and SYMMLQ on the preconditioned system. In fact the number of iterations seems to be independent of the grid size, and this number is considerably lower than the number of iterations the solvers needed in the unpreconditioned case. The iteration numbers are given in Table 7.9.

Case 2: $\alpha \ll 1, D_y = 0, D_u = 0$

We concluded that the extreme eigenvalues of the system we consider are bounded independently of the mesh size if we precondition with P_1 . However, they are not bounded independently of the regularization parameter α . This follows from the estimate in (6.21). The results are not satisfying in the case of a small regularization parameter α . While there is no change in $A^{-1}A, P_u^{-1}H_uP_u^{-T}$ and $P_y^{-1}H_yP_y^{-T}$, the singular values of $P_y^T A^{-1} B P_u^{-T}$ now are multiplied by $1/\sqrt{\alpha}$. This is mirrored in the change of the outer bounds for the spectrum. The extreme eigenvalues of the system for $\alpha = 10^{-5}$ are given in Table 7.10. The estimates for the spectrum are accurate. The condition number of the system was hardly reduced by preconditioning; it is still of order 10^3 . MINRES and SYMMLQ need a substantially larger number of iterations than in Case 1. The number of steps they need is given in Table 7.10. However, they are still able to solve the system with $P_1^{-1} K P_1^{-1}$ in a relatively small number of steps for values of α where they already needed $2m + n$ steps in the unpreconditioned case. This is shown in Table 7.12.

Case 3: $\alpha = 1, D_y = 0, D_u = 10^4 \cdot I$

If the diagonal entries in H_u are increased by a considerable amount, we do not expect a deterioration in the conditioning of the system, but even a slight improvement. This was derived in Section 6.1 and is confirmed by the numerical results. The eigenvalues and singular values of the submatrices are given in Figure 7.10. The eigenvalues of the preconditioned system are shown in Figure 7.11. We see that they are clustered more tightly than for $D_u = 0$. The eigenvalues of $P_u^{-1}H_uP_u^{-T}$ are smaller than in Case 1. MINRES and SYMMLQ

(The estimated spectrum is computed using Theorem 3.2.1.
In all computations, $n_x = n_y$.)

n_x	h	Computed Spectrum				Estimated Spectrum			
5	2.83e-1	-1.35e+0	-4.41e-1	5.00e-1	3.00e+0	-1.77e+0	-4.14e-1	5.00e-1	3.24e+0
10	1.41e-1	-1.35e+0	-4.25e-1	5.00e-1	3.00e+0	-1.77e+0	-4.14e-1	5.00e-1	3.24e+0
20	7.07e-2	-1.35e+0	-4.18e-1	5.00e-1	3.00e+0	-1.77e+0	-4.14e-1	5.00e-1	3.24e+0
30	4.71e-2	-1.35e+0	-4.16e-1	5.00e-1	3.00e+0	-1.77e+0	-4.14e-1	5.00e-1	3.24e+0

Table 7.7: Computed and estimated spectrum of $P_1^{-1}KP_1^{-T}$ with $\alpha = 1$, $D_y = 0$, $D_u = 0$.

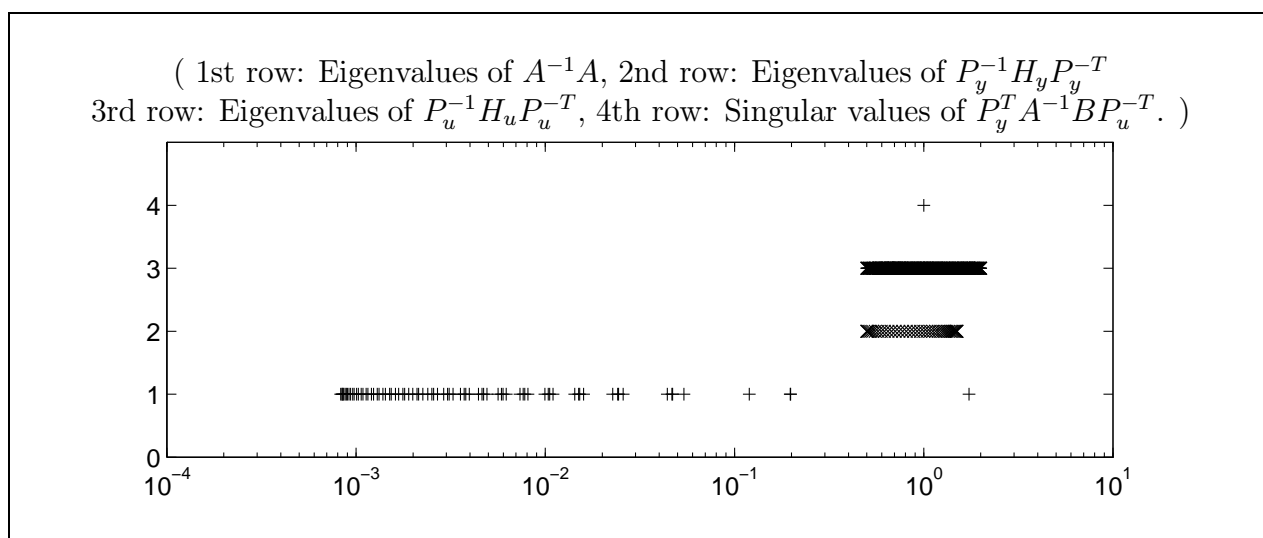


Figure 7.8: The eigenvalues and singular values of the preconditioned submatrices in $P_1^{-1}KP_1^{-T}$ with $\alpha = 1$, $D_y = 0$, $D_u = 0$ for $n_x = n_y = 20$.

(Positive eigenvalues of K are denoted by '+'.
 Negative eigenvalues of $P_1^{-1}KP_1^{-T}$, given in absolute value, are denoted by '*'.
 The lines denote the estimate for the positive and negative parts of the spectrum.)

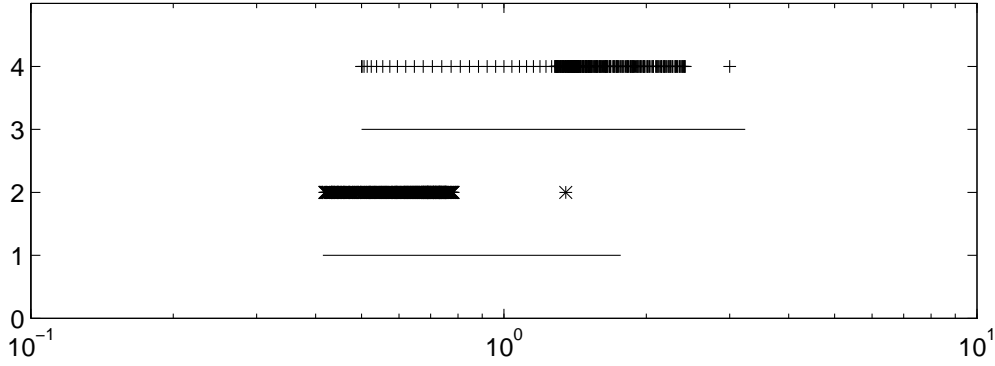


Figure 7.9: The eigenvalues of the preconditioned KKT-matrix $P_1^{-1}KP_1^{-T}$ with $\alpha = 1$, $D_y = 0$, $D_u = 0$ for $n_x = n_y = 20$.

(In all computations, $n_x = n_y$.)

grid size	5	10	15	20	25	30
$\kappa(P_1^{-1}KP_1^{-T})$	6.80e+0	7.05e+0	7.13e+0	7.17e+0	7.19e+0	7.20e+0
$\kappa(P_y^{-1}H_yP_y^{-1})$	4.00e+0	4.00e+0	4.00e+0	4.00e+0	4.00e+0	4.00e+0
$\kappa(P_u^{-1}H_uP_u^{-1})$	3.00e+0	3.00e+0	3.00e+0	3.00e+0	3.00e+0	3.00e+0
$\kappa(P_y^T A^{-1}BP_u^{-1})$	2.58e+2	7.36e+2	1.36e+3	2.09e+3	2.92e+3	3.84e+3

Table 7.8: Condition numbers of the preconditioned system $P_1^{-1}KP_1^{-T}$ with $\alpha = 1$, $D_y = 0$, $D_u = 0$ and the submatrices for different grid sizes.

(In all computations, $n_x = n_y$.)

grid size	5	10	15	20	25	30
dimension	92	282	572	962	1452	2042
MINRES	23	25	24	21	21	19
SYMMLQ	23	24	22	21	19	19

Table 7.9: Iterations of MINRES and SYMMLQ for $P_1^{-1}KP_1^{-T}$ with $\alpha = 1$, $D_y = 0$, $D_u = 0$.

(The estimated spectrum is computed using Theorem 3.2.1.
In all computations, $n_x = n_y$.)

n_x	h	Computed Spectrum				Estimated Spectrum			
5	2.83e-1	-5.47e+2	-5.16e-1	5.95e-1	5.49e+2	-5.48e+2	-4.14e-1	5.00e-1	5.49e+2
10	1.41e-1	-5.47e+2	-4.40e-1	5.57e-1	5.49e+2	-5.48e+2	-4.14e-1	5.00e-1	5.49e+2
20	7.07e-2	-5.47e+2	-4.21e-1	5.10e-1	5.49e+2	-5.48e+2	-4.14e-1	5.00e-1	5.49e+2
30	4.71e-2	-5.47e+2	-4.17e-1	5.10e-1	5.49e+2	-5.48e+2	-4.14e-1	5.00e-1	5.49e+2

Table 7.10: Computed and estimated spectrum of $P_1^{-1}KP_1^{-T}$ with $\alpha = 10^{-5}$, $D_y = 0$, $D_u = 0$.

(In all computations, $n_x = n_y$.)

grid size	5	10	15	20	25	30
dimension	92	282	572	962	1452	2042
MINRES	76	120	120	118	104	108
SYMMLQ	72	109	107	105	100	99

Table 7.11: Iterations of MINRES and SYMMLQ for $P_1^{-1}KP_1^{-T}$ with $\alpha = 10^{-5}$, $D_y = 0$, $D_u = 0$.

(In all computations, $n_x = n_y$.)

grid size	5	10	15	20	25	30
MINRES	10^{-6}	10^{-7}	10^{-8}	10^{-8}	10^{-9}	10^{-9}
SYMMLQ	10^{-6}	10^{-7}	10^{-8}	10^{-9}	10^{-9}	10^{-10}

Table 7.12: Largest value of α for that MINRES and SYMMLQ can no longer compute a solution for $P_1^{-1}KP_1^{-1}$ with $D_y = 0$, $D_u = 0$ within the required accuracy in less than $2m + n$ steps.

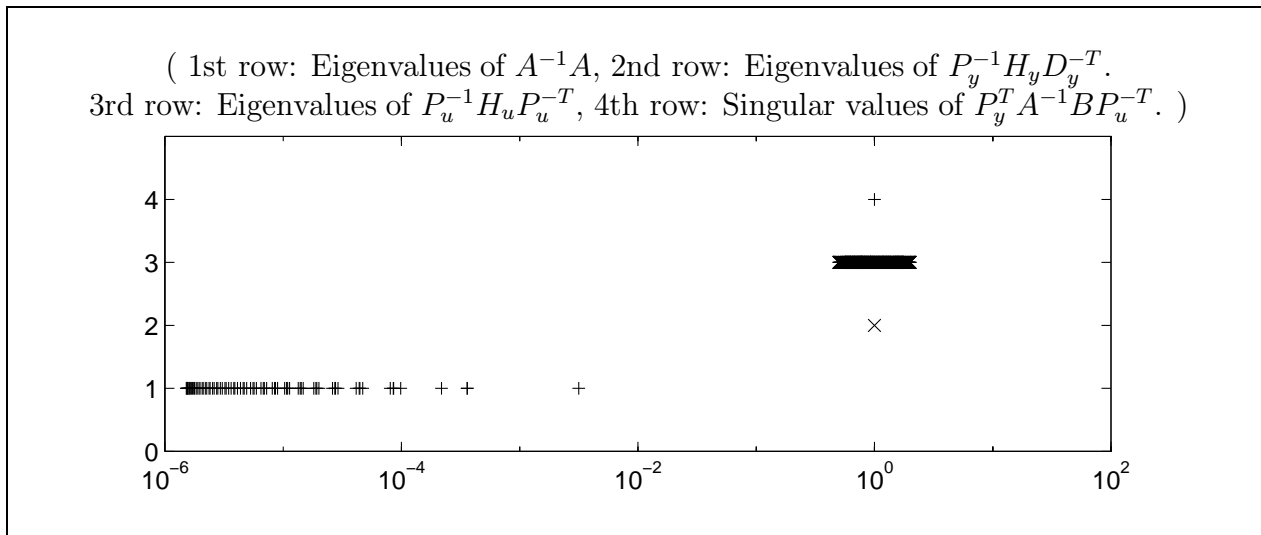


Figure 7.10: The eigenvalues and singular values of the submatrices in $P_1^{-1}KP_1^{-T}$ with $D_u = 10^4 \cdot I$, $D_y = 0$, $\alpha = 1$ for $n_x = n_y = 20$.

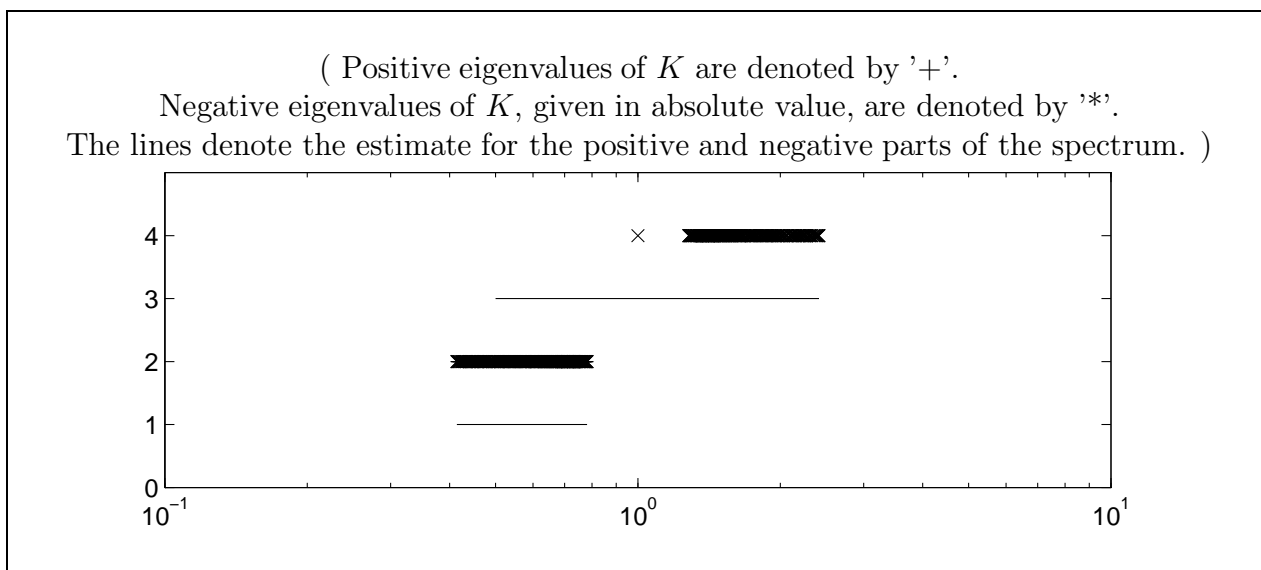


Figure 7.11: The eigenvalues of $P_1^{-1}KP_1^{-T}$ with $D_u = 10^4 \cdot I$, $D_y = 0$, $\alpha = 1$ for $n_x = n_y = 20$.

(In all computations, $n_x = n_y$.)						
grid size	5	10	15	20	25	30
dimension	92	282	572	962	1452	2042
MINRES	16	18	18	18	18	16
SYMMLQ	16	18	18	18	17	16

Table 7.13: Iterations of MINRES and SYMMLQ for $P_1^{-1}KP_1^{-T}$ with $D_u = 10^4 \cdot I$, $D_y = 0$, $\alpha = 1$.

compute a solution with less iterations than in Case 1. The number of iterations they need seems to be independent of the grid size.

Case 4: $\alpha = 1$, $D_y = 10^4 \cdot I$, $D_u = 0$

The situation is different to the preceding case if the diagonal of H_y increases. In this case, the singular values of $P_y^T A^{-1} B P_u^{-T}$ are large, and the spectrum of $P_1^{-1} K P_1^{-T}$ is barely shrunk with respect to the spectrum of the original K . The eigenvalue distribution for the preconditioned system is shown in Figure 7.13. MINRES and SYMMLQ need a considerably larger number of iterations than in Cases 1 and 3. The number of iterations is given in Table 7.14.

Quality of the Solution

The typical behavior of the residuals of MINRES and SYMMLQ iterates in the preconditioned case is shown in Figure 7.12. The residuals of the two iterative methods do resemble each other much more than in the original case. The original case was shown in Figure 7.4. The absolute error for the solution to the preconditioned system is smaller than for the solution to the original system in the u - and p -components, but larger in the y -component. The discrepancy in the accuracy of the three components in the solution to the original system is no longer present in the absolute error of this solution. However, the relative error in the u -components is still relatively high with respect to the other two components. The overall relative error for the preconditioned system is around one significant digit smaller than for the original system.

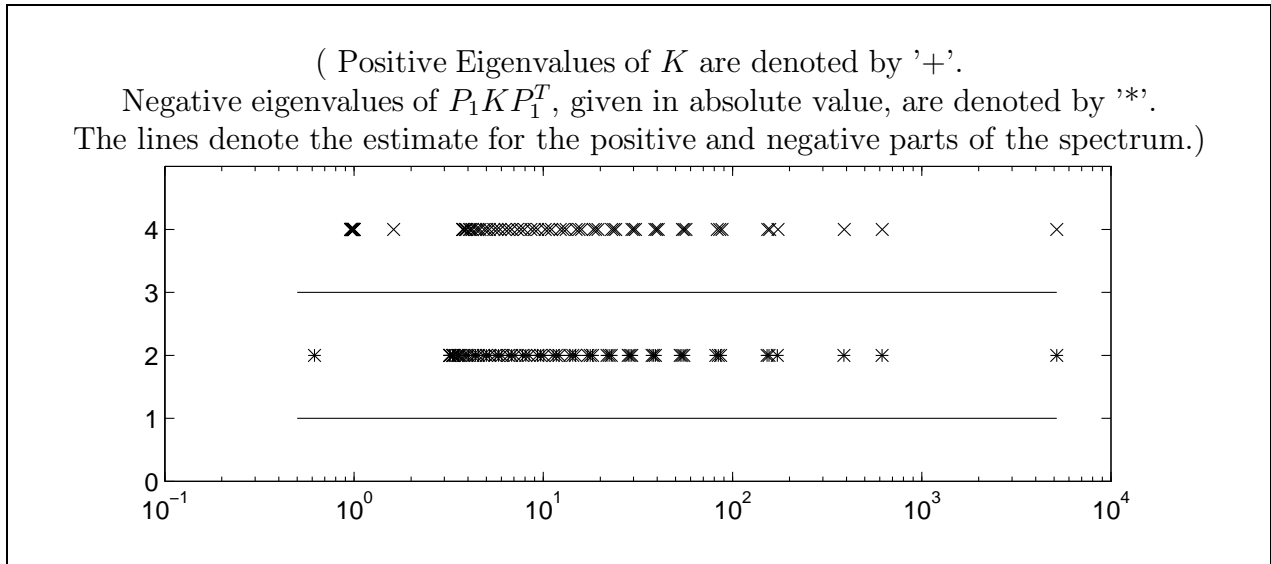


Figure 7.13: The eigenvalues of $P_1^{-1} K P_1^{-T}$ with $D_y = 10^4 \cdot I$, $D_u = 0$, $\alpha = 1$ for $n_x = n_y = 20$.

(In all computations, $n_x = n_y$.)

grid size	5	10	15	20	25	30
dimension	92	282	572	962	1452	2042
MINRES	50	98	194	289	449	530
SYMMLQ	50	98	187	283	410	524

Table 7.14: Iterations of MINRES and SYMMLQ for $P_1^{-1} K P_1^{-T}$ with $\alpha = 1$ and $D_y = 10^4$, $D_u = 0$.

(First diagram: Residuals of the iterates.
 Second diagram: The absolute error in the components of the solution vector.
 Third diagram: The relative error in the components of the solution vector.)

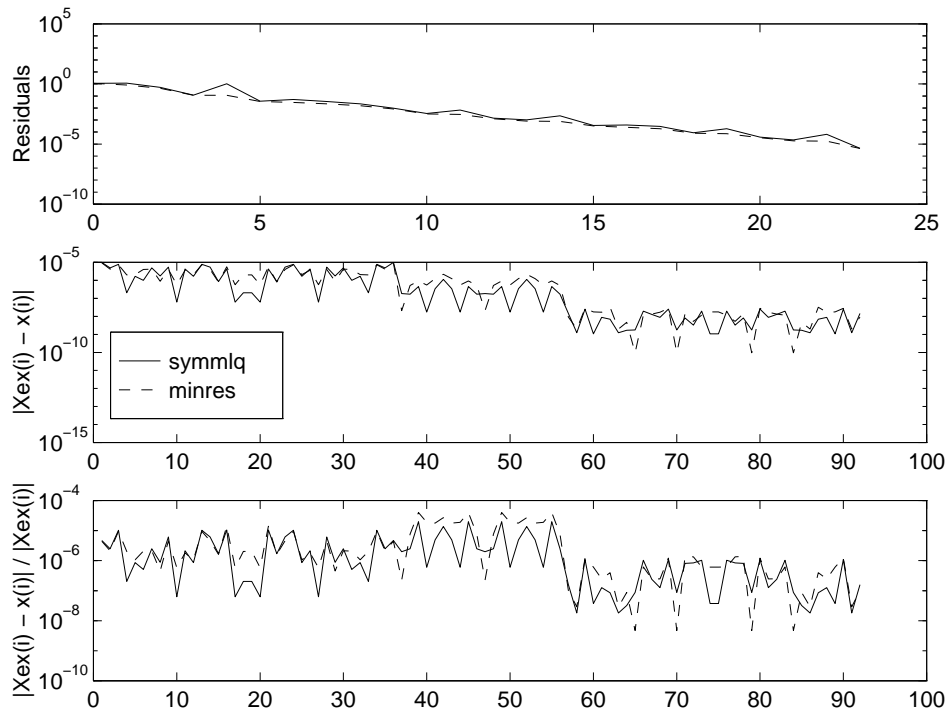


Figure 7.12: The residuals, the absolute and the relative error of MINRES- and SYMMLQ-iterates on the system $P_1^{-1} K P_1^{-T}$ for $n_x = n_y = 5$ with $\alpha = 1, D_y = 0, D_u = 0$.

7.7 Numerical Results with the Second Preconditioner

The second preconditioner is given by

$$P_2^{-1} = \begin{pmatrix} P_y^{-1} & 0 & 0 \\ 0 & P_u^{-1} & 0 \\ -P_y^{-1} & -P_y^T \tilde{A}^{-1} B P_u^{-1} P_u^{-T} & P_y^T \tilde{A}^{-1} \end{pmatrix},$$

where, as before, P_y and P_u denote the square roots of the diagonals of H_y and H_u , respectively, and \tilde{A} an approximate inverse of A . The preconditioned Karush–Kuhn–Tucker matrix is

$$(P_2^*)^{-1} K (P_2^*)^{-T} = \begin{pmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & -(I + \tilde{B} \tilde{B}^T) \end{pmatrix}$$

with $\tilde{B} = H_y^{1/2} A^{-1} B H_u^{-1/2}$ in the ideal case. We used an exact LU -factorization for A in our computations, so that we have $\tilde{B} = P_y^T A^{-1} B P_u^{-T}$ and actually work on the system

$$\begin{pmatrix} P_y^{-1} H_y P_y^{-T} & 0 & I - P_y^{-T} H_y P_y^{-1} \\ 0 & P_u^{-1} H_u P_u^{-T} & I - P_u^{-T} H_u P_u^{-1} \\ I - P_y^{-1} H_y P_y^{-T} & I - P_u^{-1} H_u P_u^{-T} & -2I + P_y^{-1} H_y P_y^{-T} + \tilde{B} (P_u^{-1} H_u P_u^{-T} - 2I) \tilde{B}^T \end{pmatrix}.$$

Case 1: $\alpha = 1$, $D_y = 0$, $D_u = 0$

In the case of a regularization parameter $\alpha = 1$, the second preconditioner provides us with a considerable reduction of the spectrum compared to that of the original matrix. The positive part of the spectrum is not exactly one, because we do not use the inverses of H_y , H_u . Nevertheless, the result

$$\Lambda(P_u^{-1} H_u P_u^{-T}) \subset [0.5, 2] \quad \text{and} \quad \Lambda(P_y^{-1} H_y P_y^{-T}) \subset [0.5, 2]$$

is a considerable improvement. For the negative part of the spectrum we get a small lower bound because the singular values of $P_y^T A^{-1} B P_u^{-T}$ are small. The bounds for the spectrum are given in Table 7.15. The extremal eigenvalues for the preconditioned case seem to be identical for all grid sizes. The eigenvalues and singular values of the preconditioned submatrices are plotted in Figure 7.14 for a grid with $n_x = n_y = 20$. The relation between the singular values of $\tilde{B} = P_y^T A^{-1} B P_u^{-T}$ and the eigenvalues of $-(I + \tilde{B} \tilde{B}^T)$ (see Lemma 6.2.1) can be seen clearly. The distribution of the eigenvalues of the entire preconditioned system is shown in Figure 7.15. The condition numbers for the submatrices and the whole system are given in Table 7.16. They seem to be independent of the grid size and are considerably smaller than for the original system. In Table 7.17 we give the number of iterations MINRES and SYMMLQ need to solve the system with $P_2^{-1} K P_2^{-T}$.

(In all computations, $n_x = n_y$.)

n_x	h	Computed Spectrum			
5	2.83e-1	-4.00e+0	-1.00e+0	5.00e-1	2.00e+0
10	1.41e-1	-4.00e+0	-1.00e+0	5.00e-1	2.00e+0
20	7.07e-2	-4.00e+0	-1.00e+0	5.00e-1	2.00e+0
30	4.71e-2	-4.00e+0	-1.00e+0	5.00e-1	2.00e+0

Table 7.15: Computed spectrum of $P_2^{-1}KP_2^{-T}$ with $\alpha = 1$, $D_y = 0$, $D_u = 0$.

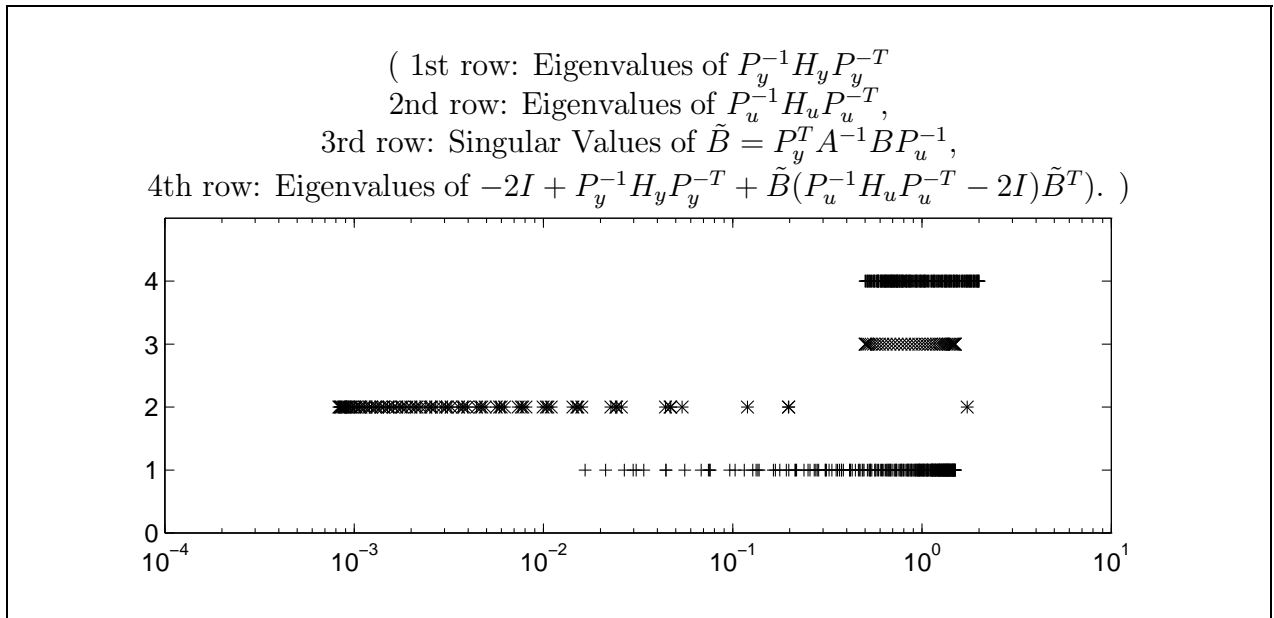


Figure 7.14: The eigenvalues and singular values of the preconditioned submatrices in $P_2^{-1}KP_2^{-T}$ for $n_x = n_y = 20$, $\alpha = 1$, $D_y = 0$, $D_u = 0$.

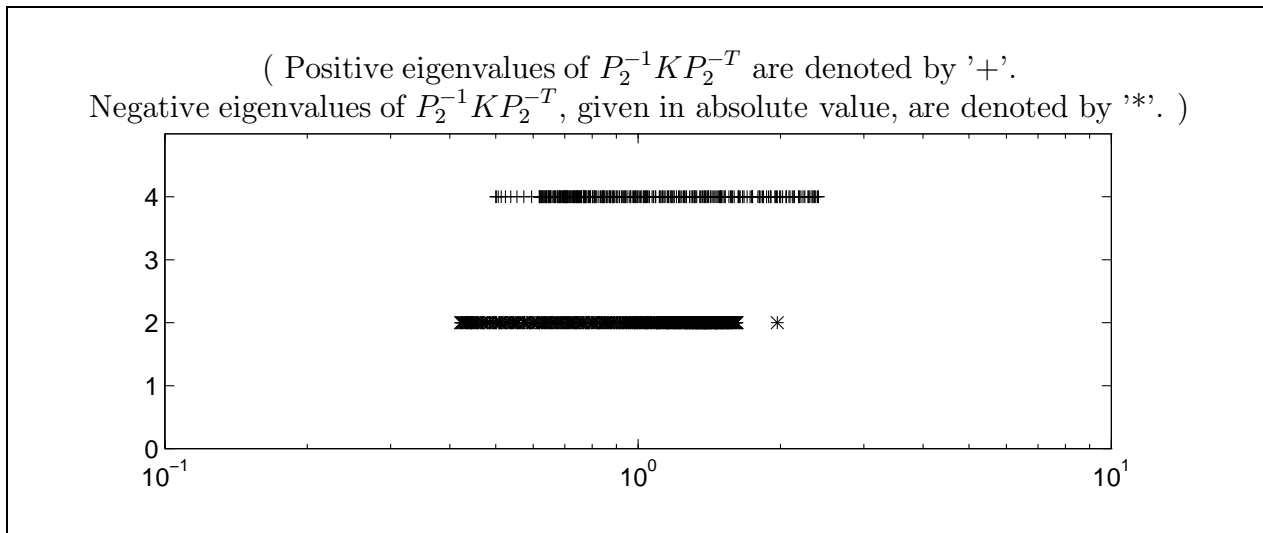


Figure 7.15: The eigenvalues of the preconditioned KKT-matrix $P_2^{-1}KP_2^{-T}$ for $n_x = n_y = 20$, $\alpha = 1$, $D_y = 0$, $D_u = 0$.

(In all computations, $n_x = n_y$.)

grid size	5	10	20	30
$\kappa(P_2^{-1}KP_2^{-T})$	8.01e+0	8.00e+0	8.00e+0	8.00e+0
$\kappa(P_y^{-1}H_yP_y^{-T})$	4.00e+0	4.00e+0	4.00e+0	4.00e+0
$\kappa(P_u^{-1}H_uD_u^{-T})$	3.00e+0	3.00e+0	3.00e+0	3.00e+0
$\kappa(\tilde{B} = P_y^T A^{-1}BP_u^{-T})$	2.58e+2	7.36e+2	2.09e+3	3.84e+3
$-(I + \tilde{B}\tilde{B}^T)$	4.00e+0	4.00e+0	4.00e+0	4.00e+0

Table 7.16: Condition numbers of the preconditioned system $P_2^{-1}KP_2^{-T}$ and the submatrices for different gridsizes; $\alpha = 1$, $D_y = 0$, $D_u = 0$.

(In all computations, $n_x = n_y$.)						
grid size	5	10	15	20	25	30
dimension	92	282	572	962	1452	2042
MINRES	24	35	37	37	35	35
SYMMLQ	24	35	36	35	35	33

Table 7.17: Iterations of MINRES and SYMMLQ for $P_2^{-1}KP_2^{-T}$ with $\alpha = 1$, $D_y = 0$, $D_u = 0$.

Case 2: $\alpha \ll 1$, $D_y = 0$, $D_u = 0$

If the regularization parameter α becomes small, the lower bound on the negative part of the spectrum decreases with the reciprocal of α . The effects on the bounds of the spectrum of the entire system are visible in Table 7.18. Thus the condition number of the preconditioned system is hardly reduced compared to the condition number of the original system. The iterations MINRES and SYMMLQ need are given in Table 7.20. The solvers can solve systems with $P_2^{-1}KP_2^{-1}$ for small values α considerably better than with the original system. Their performance for small α on different grids is shown in Table 7.19.

Case 3: $\alpha = 1$, $D_y = 0$, $D_u = 10^4 \cdot I$

If the diagonal of H_u is increased, the performance of the second preconditioner is good. The off-diagonal entries in $I - P_u^{-1}H_uP_u^{-T}$ are smaller than in Case 1, and the spectrum of the lower block is shrunk with respect to the spectrum in Case 1. The spectrum of $P_2^{-1}KP_2^{-T}$ is narrow. Therefore we can expect that MINRES and SYMMLQ only need a small number of iterations. The iteration numbers are given in Table 7.20. In fact the iteration count is almost similar to that in Case 1, where no large entries occur in H_u . The iterations the solvers need here seem to be independent of the grid size.

Case 4: $\alpha = 1$, $D_y = 10^4 \cdot I$, $D_u = 0$

In the analysis of the preconditioner P_1 we have seen that an increase in the diagonal of H_y affects the performance of MINRES and SYMMLQ on the preconditioned system $P_1^{-1}KP_1^{-T}$ almost as much as on the original K . Unfortunately, the second preconditioner does not improve the situation. The spectrum of the preconditioned matrix $P_2^{-1}KP_2^{-T}$, given in Figure 7.16, is very large, and so we can anticipate a large number of iterations. These are given in Table 7.21.

(In all computations, $n_x = n_y$.)

n_x	h	Computed Spectrum			
5	2.83e-1	-3.00e+5	-1.00e+0	5.00e-1	2.00e+0
10	1.41e-1	-3.00e+5	-1.00e+0	5.00e-1	2.00e+0
20	7.07e-2	-3.00e+5	-1.00e+0	5.00e-1	2.00e+0
30	4.71e-2	-3.00e+5	-1.00e+0	5.00e-1	2.00e+0

Table 7.18: Computed spectrum of $P_2^{-1}KP_2^{-T}$ with $\alpha = 10^{-5}$, $D_y = 0$, $D_u = 0$.

(In all computations, $n_x = n_y$.)

grid size	5	10	15	20	25	30
MINRES	10^{-6}	10^{-6}	10^{-7}	10^{-7}	10^{-8}	10^{-8}
SYMMLQ	10^{-6}	10^{-6}	10^{-7}	10^{-8}	10^{-8}	10^{-8}

Table 7.19: Largest value of α for that MINRES and SYMMLQ can no longer compute a solution to the system with $P_2^{-1}KP_2^{-1}$ ($D_y = 0$, $D_u = 0$) within the required accuracy in less than the maximal number of steps.

(In all computations, $n_x = n_y$.)

grid size	5	10	15	20	25	30
dimension	92	282	572	962	1452	2042
MINRES	21	33	35	37	35	35
SYMMLQ	21	33	35	35	33	33

Table 7.20: Iterations of MINRES and SYMMLQ for $P_2^{-1}KP_2^{-T}$ with $D_u = 10^4 \cdot I$, $\alpha = 1$, $D_y = 0$.

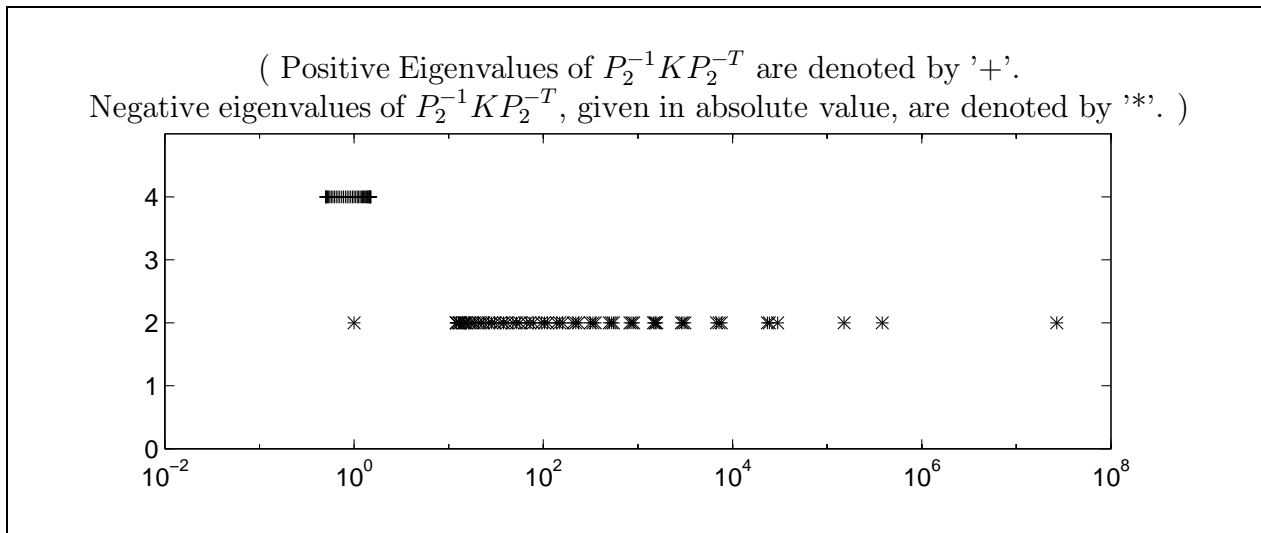


Figure 7.16: The eigenvalues of the KKT matrix $P_2^{-1}KP_2^{-T}$ with $D_y = 10^4 \cdot I$, $\alpha = 1$, $D_u = 0$ for $n_x = n_y = 20$.

(In all computations, $n_x = n_y$.)

grid size	5	10	15	20	25	30
dimension	92	282	572	962	1452	2042
MINRES	61	146	235	323	453	583
SYMMLQ	61	143	233	323	447	547

Table 7.21: Iterations of MINRES and SYMMLQ for $P_2^{-1}KP_2^{-T}$ with $\alpha = 1$ and $D_y = 10^4 \cdot I$, $D_u = 0$.

Quality of the Solution

In this section we use the notation introduced in Section 6.3.5, i.e. by e_1 , e_2 and e_3 we denote the y -, u - and p -components of the error, respectively. By K_2 we denote the preconditioned system $P_2^{-1}KP_2^{-T}$.

In Case 1 ($\alpha = 1$, $D_y = 0$, $D_u = 0$), MINRES and SYMMLQ generally need around 30 steps to reach a solution with a residual smaller than 10^{-5} . The overall absolute error is smaller than 10^{-5} (see Figure 7.17) and substantially lower in parts of the solution vector. If the parameter α is small, MINRES and SYMMLQ need almost 300 steps. This can be seen in Figure 7.18). The quality of the solution deteriorates considerably. The overall absolute error increases to a large amount, but not uniformly in all components. The partitioning of the solution vector into the different components is now clearly visible in the absolute error. While the error in the y - and p -components stay essentially the same compared to the error for $\alpha = 1$, the error in the u -components is increased by a factor between 10^2 and 10^3 . We

have seen in the analysis in Section 6.3.5 that the error \tilde{e}_3 (the preconditioned error) has the potential to rise if the eigenvalues of $-(I + \tilde{B}\tilde{B}^T)$ become small. This component only influences the estimate for the error e_3 , and it is damped out by the action of $h_y^{1/2}A^{-1}$. The u -components, i.e. e_2 , however, are given as $H_u^{-1/2}\tilde{e}_2$. While \tilde{e}_2 is of order ϵ , the diagonal of H_u is dominated by the factor α . This is the reason for the increase in the absolute error in the u -component.

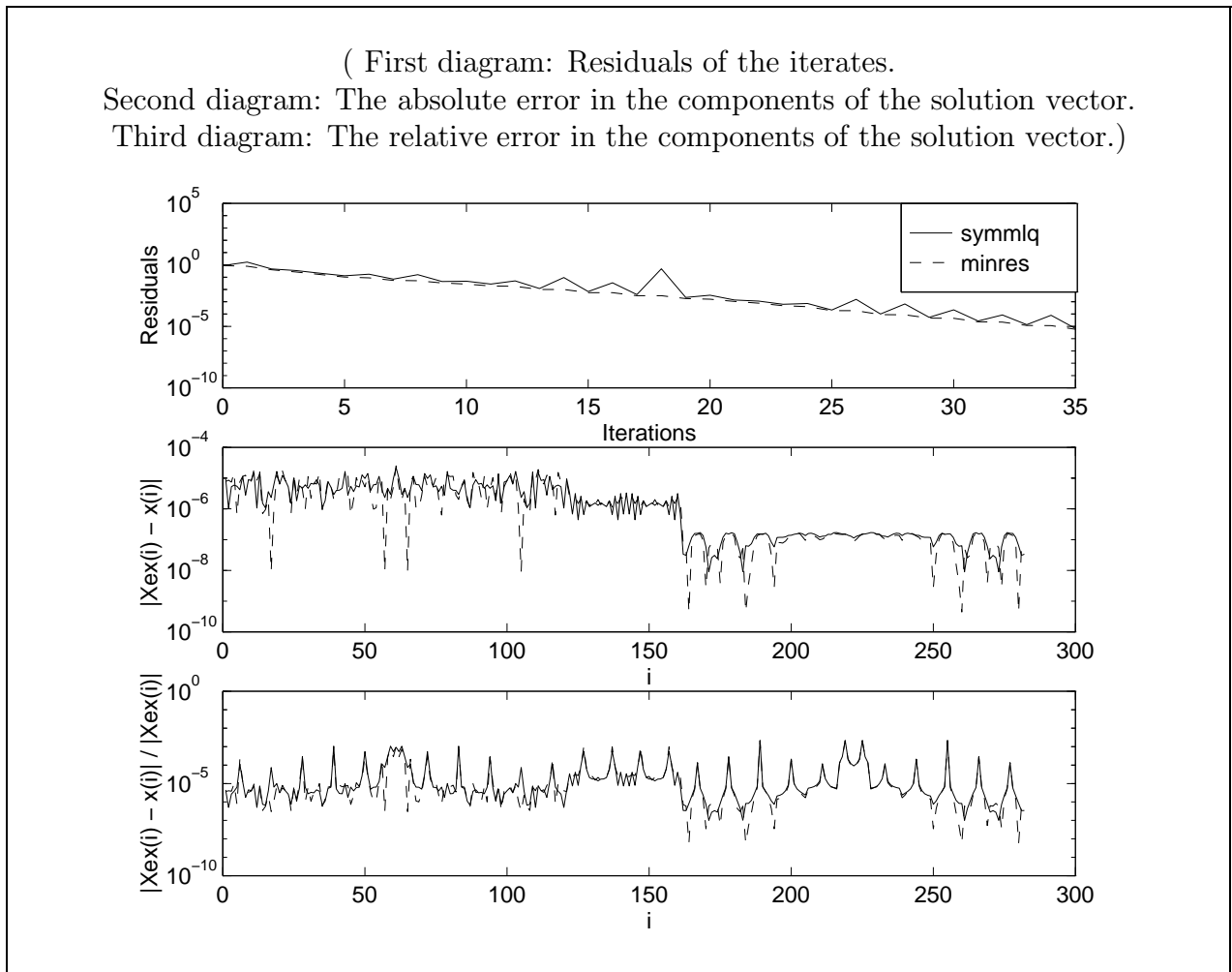


Figure 7.17: The residuals, the absolute and the relative error of MINRES- and SYMMLQ-iterates on the system $P_2^{-1}KP_2^{-T}$ for $n_x = n_y = 10$ with $D_y = 0$, $D_u = 0$, $\alpha = 1$.

(First diagram: Residuals of the iterates.
 Second diagram: The absolute error in the components of the solution vector.
 Third diagram: The relative error in the components of the solution vector.)

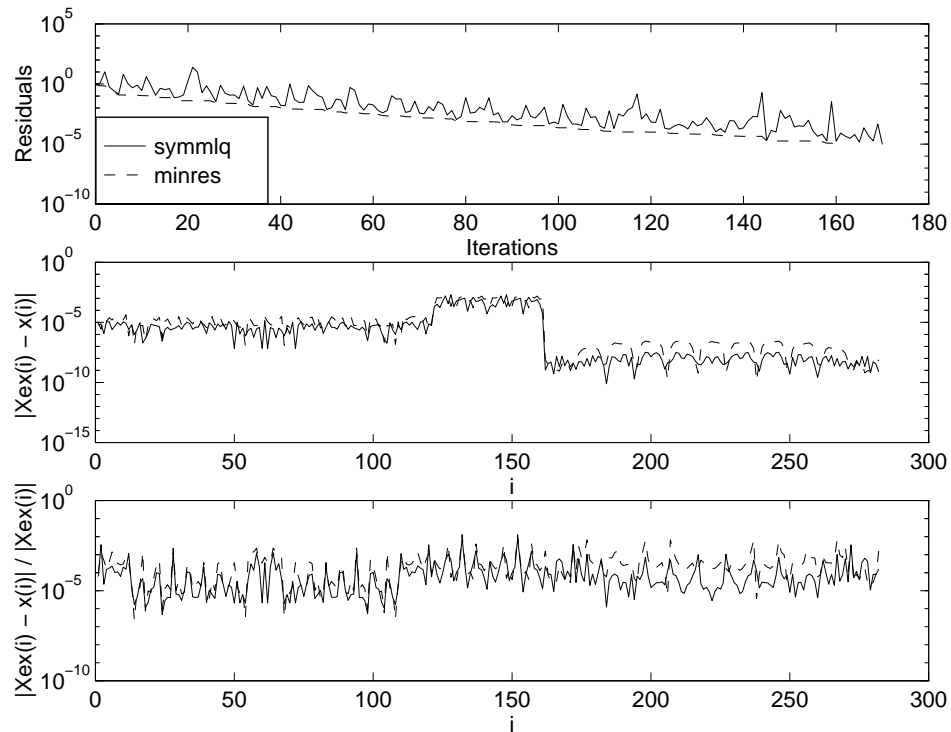


Figure 7.18: The residuals, the absolute and the relative error of MINRES- and SYMMLQ-iterates on the system $P_2^{-1}K P_2^{-T}$ for $n_x = n_y = 10$ with $D_y = 0$, $D_u = 0$, $\alpha = 10^{-5}$.

7.8 Numerical Results with the Third Preconditioner

The ideal third preconditioner is given by

$$(P_3^*)^{-1} = \begin{pmatrix} I_m & 0 & -1/2 H_y A^{-1} \\ 0 & 0 & A^{-1} \\ -(A^{-1}B)^T & I_n & (A^{-1}B)^T H_y A^{-1} \end{pmatrix},$$

the ideally preconditioned system is

$$(P_3^*)^{-1} K (P_3^*)^{-T} = \begin{pmatrix} 0 & I_m & 0 \\ I_m & 0 & 0 \\ 0 & 0 & -B^T A^{-T} H_y A^{-1} B + H_u \end{pmatrix}.$$

Case 1: $\alpha = 1$, $D_y = 0$, $D_u = 0$

Under the action of the third preconditioner, the spectrum of the system shrinks considerably. The bounds on the spectrum of $P_3^{-1} K P_3^{-T}$ are given in Table 7.22. The eigenvalues are plotted in Figure 7.20. The eigenvalues of the system are clustered around 1 and -1 , and another bundle of eigenvalues is located around $h = 0.5 * 10^{-2}$. This is the influence of H_u in the lower block. Figure 7.19 shows the eigenvalues of the submatrices in the system $P_3^{-1} K P_3^{-T}$. The eigenvalues of $(A^{-1}B)^T H_y (A^{-1}B) = W^T H W - H_u$ are small. The eigenvalues of H_u dominate the distribution of the eigenvalues of $W^T H W$. The condition number of the preconditioned system is considerably smaller than the condition number of the original system. The condition numbers of the system and of the dominant submatrix $W^T H W$ for different grid sizes are given in Table 7.24. MINRES and SYMMLQ need only a small number of iterations on the system $P_3^{-1} K P_3^{-T}$ to reach a solution with a residual smaller than the required 10^{-5} . The number of iterations seems to be independent of the grid size. They are given in Table 7.23.

Case 2: $\alpha \ll 1$, $D_y = 0$, $D_u = 0$

The eigenvalues of $(\tilde{A}^{-1}B)^T H_y (\tilde{A}^{-1}B)$ are small. Under the influence of a small parameter α , the eigenvalues of H_u move towards zero – and with them the eigenvalues of $W^T H W$. This can be seen clearly in Table 7.25 and Figure 7.21. This directly affects the eigenvalues of $W^T H W$, as can be seen in Figure 7.22. But although the small positive eigenvalues of the system move towards zero, the solvers can deal very well with small parameters α in the system $P_3^{-1} K P_3^{-T}$. The iteration numbers are given in Table 7.26. They are only slightly higher than in Case 1, and they are substantially lower than for the two other preconditioned systems for small parameters α .

Case 3: $\alpha = 1$, $D_y = 0$, $D_u \gg I$

The situation can be judged favorably if the diagonal of H_u rises. We considered uniform increases in the diagonal of H_u . If the diagonal of H_u is increased by 10^4 , the system has

(In all computations, $n_x = n_y$.)

n_x	h	Computed Spectrum			
5	2.83e-1	-1.00e+0	-1.00e+0	6.67e-2	1.00e+0
10	1.41e-1	-1.00e+0	-1.00e+0	3.33e-2	1.00e+0
20	7.07e-2	-1.00e+0	-1.00e+0	1.67e-2	1.00e+0
30	4.71e-2	-1.00e+0	-1.00e+0	8.33e-3	1.00e+0

Table 7.22: Computed spectrum of $P_3^{-1}KP_3^{-T}$ with $\alpha = 1$, $D_y = 0$, $D_u = 0$.

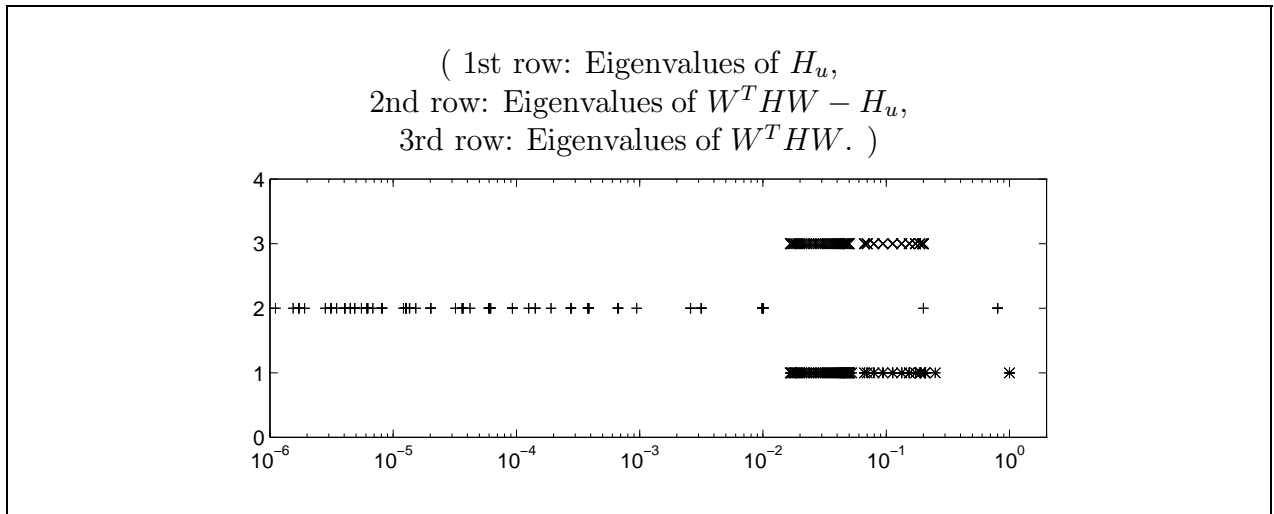


Figure 7.19: The eigenvalues of the submatrices in $P_3^{-1}KP_3^{-T}$ for $n_x = n_y = 20$, $\alpha = 1$, $D_y = 0$, $D_u = 0$.

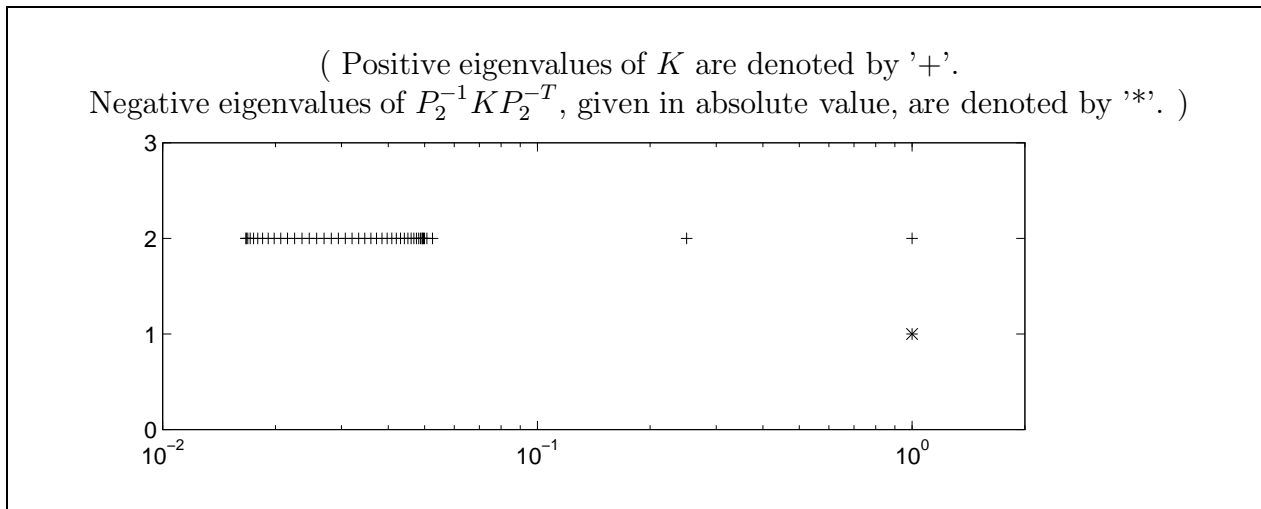


Figure 7.20: The eigenvalues of the preconditioned KKT-matrix $P_3^{-1} K P_3^{-T}$ for $n_x = n_y = 20$, $\alpha = 1$, $D_y = 0$, $D_u = 0$.

(In all computations, $n_x = n_y$.)

grid size	5	10	15	20	25	30
dimension	92	282	572	962	1452	2042
MINRES	7	6	5	5	5	4
SYMMLQ	7	6	5	5	5	4

Table 7.23: Iterations of MINRES and SYMMLQ on $P_3^{-1} K P_3^{-T}$ with $\alpha = 1$, $D_y = 0$, $D_u = 0$.

(In all computations, $n_x = n_y$.)

grid size	$P_3^{-1} K P_3^{-T}$	$W^T H W$
5	1.5e+1	1.5e+1
10	1.5e+1	3.00e+1
15	1.5e+1	4.50e+1
20	1.5e+1	6.00e+1
25	1.5e+1	7.50e+1
30	1.5e+1	9.00e+1

Table 7.24: Condition numbers of $P_3^{-1} K P_3^{-T}$ and $W^T H W$ with $\alpha = 1$, $D_y = 0$, $D_u = 0$.

(In all computations, $n_x = n_y$.)

n_x	h	Computed Spectrum			
5	2.83e-1	-1.00e+0	-1.00e+0	4.72e-6	1.00e+0
10	1.41e-1	-1.00e+0	-1.00e+0	5.82e-7	1.00e+0
20	7.07e-2	-1.00e+0	-1.00e+0	1.82e-7	1.00e+0
30	4.71e-2	-1.00e+0	-1.00e+0	8.75e-8	1.00e+0

Table 7.25: Computed spectrum of $P_3^{-1}KP_3^{-T}$ with $\alpha = 10^{-5}$, $D_y = 0$, $D_u = 0$.

(In all computations, $n_x = n_y$.)

n_x	h	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}	10^{-9}	10^{-10}
5	2.83e-1	6	9	8	10	10	10	10	10	10	10
10	1.41e-1	6	8	9	10	10	10	9	10	10	10
20	7.07e-2	5	5	8	9	10	9	10	9	9	9
30	4.71e-2	5	7	8	9	9	9	9	9	9	9

Table 7.26: Iterations of MINRES on $P_3^{-1}KP_3^{-T}$ for decreasing values of α with $D_y = 0$, $D_u = 0$. The values of α are given on the top line.

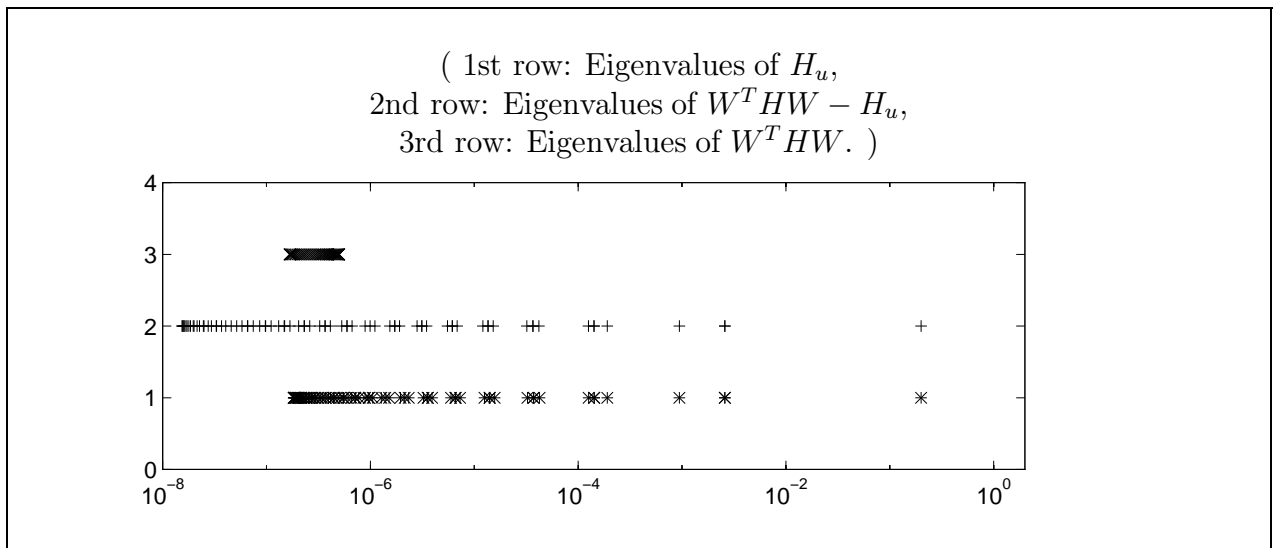


Figure 7.21: The eigenvalues of the submatrices in $P_3^{-1}KP_3^{-T}$ for $n_x = n_y = 20$, $\alpha = 10^{-5}$, $D_y = 0$, $D_u = 0$.

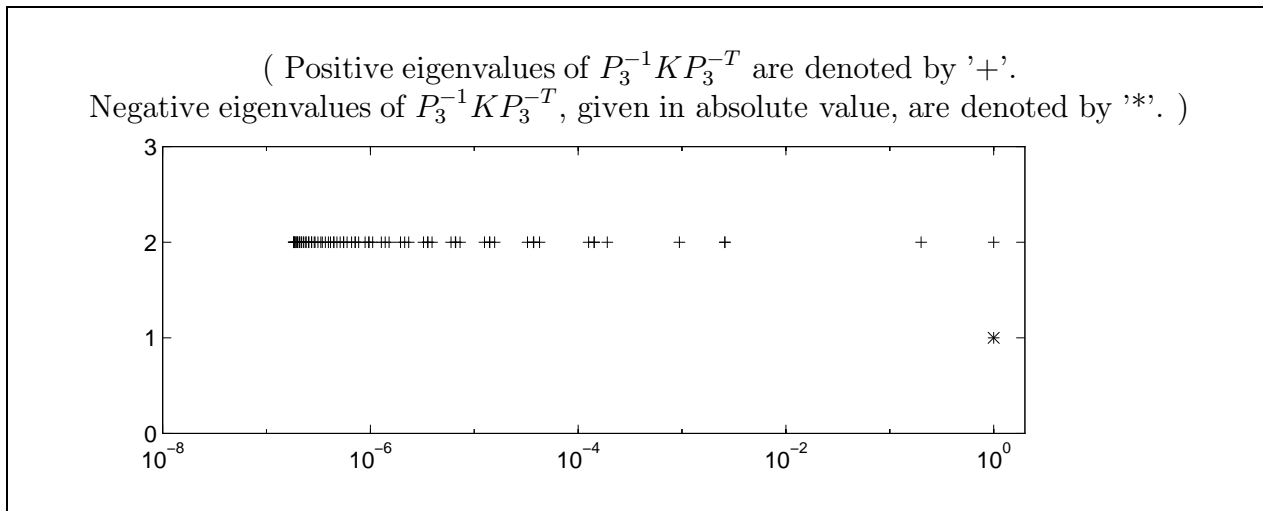


Figure 7.22: The eigenvalues of the preconditioned KKT-matrix $P_3^{-1}KP_3^{-T}$ for $n_x = n_y = 20$, $\alpha = 10^{-5}$, $D_y = 0$, $D_u = 0$.

three clusters of eigenvalues; one cluster at 1, one at -1 , and a third one at 10^4 . In this case, MINRES and SYMMLQ need even less iterations than in Case 1. The iterations are given in Table 7.27.

Case 4: $\alpha = 1$, $D_y \gg I$, $D_u = 0$

The preconditioned system $P_3^{-1}KP_3^{-T}$ is very sensitive to an increase in the diagonal in H_y . If the diagonal of H_y is increased uniformly, this does not lead to a cluster of large eigenvalues as in the preceding case. Instead, the large eigenvalues of H_y are modified by the action of A^{-1} and B such that they are widely spread. The iterations MINRES and SYMMLQ need for an increase of 10^4 are given in Table 7.28. Even though less iterations than for the systems $P_1^{-1}KP_1^{-T}$ and $P_2^{-1}KP_2^{-T}$ are needed if the diagonal of H_y is increased by 10^4 , the iterations on this system reach the maximal number $2m + n$ for smaller increases in H_y than on the two other preconditioned systems. In fact, MINRES and SYMMLQ require the maximal number of iterations on this system for a uniform increase in H_y of the order of 10^6 .

Quality of the Solution

In this section we use the notation introduced in Section 6.4.4, i.e. by e_1 , e_2 and e_3 we denote the y -, u - and p -components of the error, respectively. By K_3 we denote the preconditioned system $P_3^{-1}KP_3^{-T}$.

MINRES and SYMMLQ generally only need a small number of iterations to reach a solution for the system K_3 . For $\alpha = 1$, they reach a solution with a residual smaller than 10^{-5} within 6 steps. This is shown in Figure 7.23. The absolute error e_2 in the u -component is of the order of the residual. The absolute error e_1 and e_3 in the other two components are smaller. The error e_1 is in the largest components of the order 10^{-6} , and e_3 is smaller than 10^{-7} .

(In all computations, $n_x = n_y$.)						
grid size	5	10	15	20	25	30
dimension	92	282	572	962	1452	2042
MINRES	5	4	4	4	4	4
SYMMLQ	5	4	4	4	4	4

Table 7.27: Iterations of MINRES and SYMMLQ on $P_3^{-1}KP_3^{-T}$ with $D_u = 10^4 \cdot I$, $\alpha = 1$, $D_y = 0$.

(In all computations, $n_x = n_y$.)						
grid size	5	10	15	20	25	30
dimension	92	282	572	962	1452	2042
MINRES	44	67	120	203	275	366
SYMMLQ	44	56	120	167	286	355

Table 7.28: Iterations of MINRES and SYMMLQ for $P_3^{-1}KP_3^{-T}$ with $D_y = 10^4 \cdot I$, $\alpha = 1$, $D_u = 0$.

We see in Figure 7.24 the changes that occur for a smaller parameter α . A substantial deterioration of the accuracy in the solution occurs. The eigenvalues of $W^T HW$ become very small with $\alpha = 10^{-7}$ because the eigenvalues of H_u are multiplied by α . From the analysis in Section 6.4.4 we know that the absolute error e_2 can for small eigenvalues of $W^T HW$ be substantially larger than the residual. In fact, under the influence of the small eigenvalues the error e_2 is now between 10^0 and 10^{-1} . This is an increase by a factor 10^5 compared to the solution for $\alpha = 1$. The error in the other two components is affected by the small eigenvalues as well. However, the increase in e_1 and e_3 is smaller. The overall error e_1 now is of order 10^{-2} – this corresponds to an increase by a factor 10^4 –, and the error e_3 is still smaller than 10^{-5} . The influence of the small eigenvalues of $W^T HW$ is modified by the action of $A^{-1}B$ for e_1 , and by the action of $A^{-T}H_y A^{-1}B$ for the error e_3 in the p -components. This damps out the large error in \tilde{e}_2 , the error in the preconditioned solution.

(First diagram: Residuals of the iterates.
 Second diagram: The absolute error in the components of the solution vector.
 Third diagram: The relative error in the components of the solution vector.)

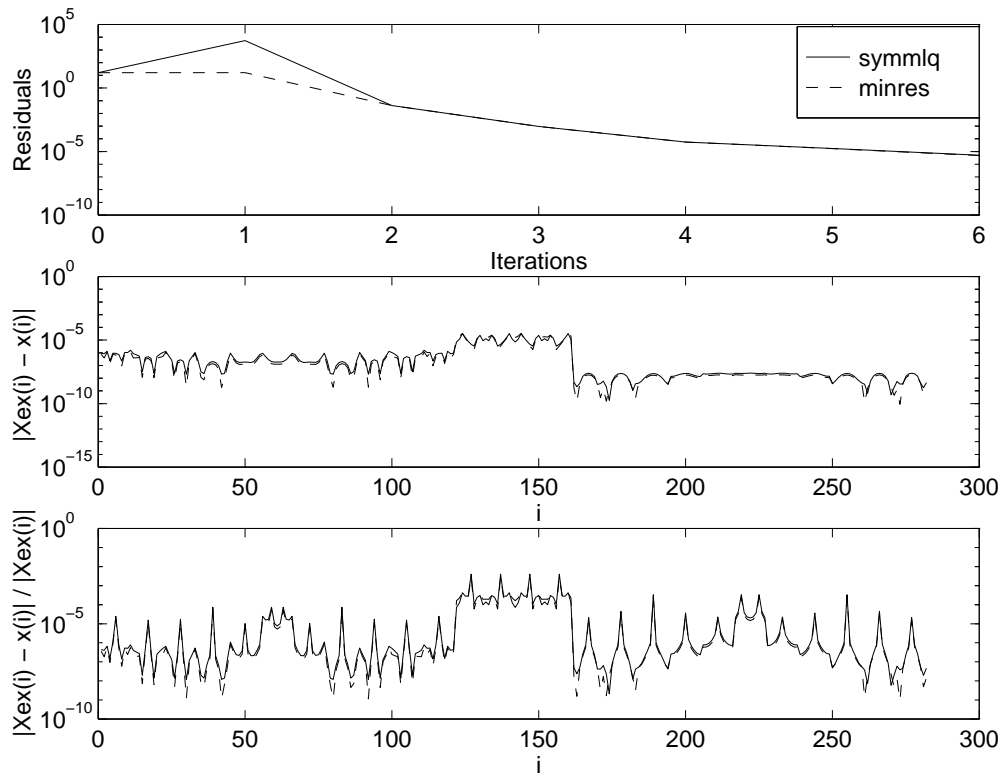


Figure 7.23: The residuals, the absolute and the relative error of MINRES- and SYMMLQ-iterates on the system $P_3^{-1} K P_3^{-T}$ for $n_x = n_y = 10$ with $D_y = 0$, $D_u = 0$, $\alpha = 1$.

(First diagram: Residuals of the iterates.
 Second diagram: The absolute error in the components of the solution vector.
 Third diagram: The relative error in the components of the solution vector.)

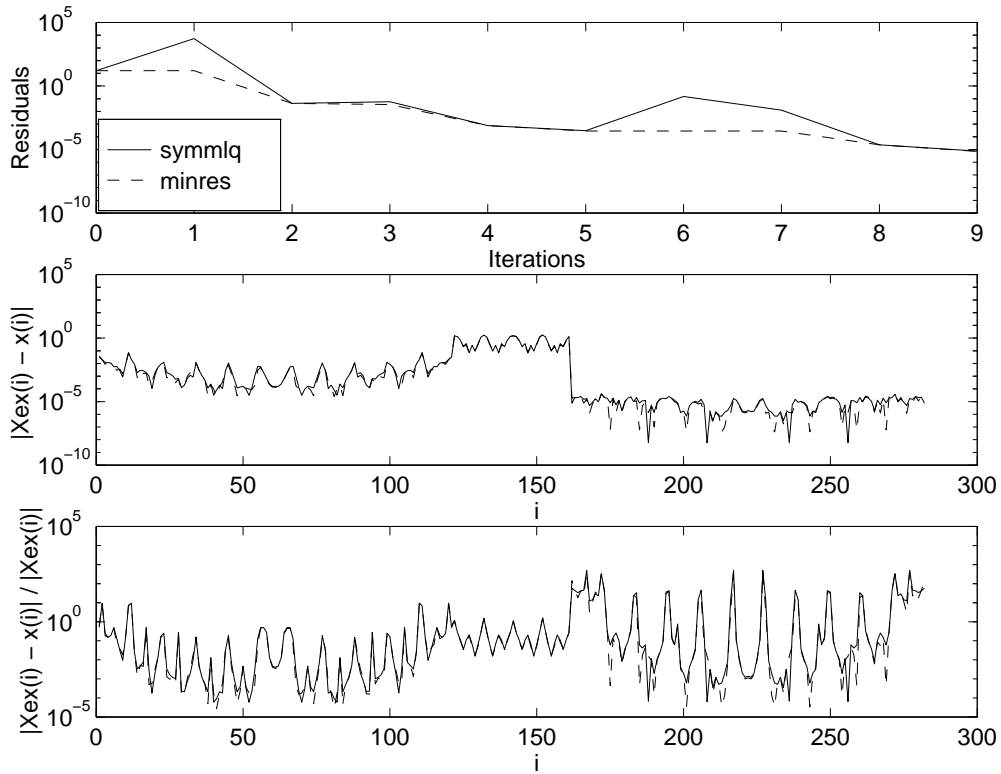


Figure 7.24: The residuals, the absolute and the relative error of MINRES- and SYMMLQ-iterates on the system $P_3^{-1} K P_3^{-T}$ for $n_x = n_y = 10$ with $D_y = 0$, $D_u = 0$, $\alpha = 10^{-7}$.

Chapter 8

Conclusion and Future Work

8.1 Conclusion

In this work we derived preconditioners for symmetric indefinite systems of the form

$$K = \begin{pmatrix} H_y & 0 & A^T \\ 0 & H_u & B^T \\ A & B & 0 \end{pmatrix}. \quad (8.1)$$

The system we were interested in arise in linear quadratic optimal control problems. In this case

$$H_y = M_y + D_y \quad \text{and} \quad H_u = \alpha \cdot M_u + D_u$$

and A is nonsingular. Our preconditioners exploit the structure of the problem and are composed of preconditioners for the submatrices H_y , H_u and A .

Preconditioners have been derived in a general form and formally analyzed. In the construction of the preconditioners, we not only paid attention to the task of 'favorable' changes in the spectrum of the system matrix, but also to the costs of applying the preconditioners. We consider as favorable changes of the eigenvalue distribution those that facilitate the convergence of MINRES and SYMMLQ. Most of the results concerning the iterative solution methods and their convergence are known and can be found in the literature. However, these results are adapted and presented in a form suitable to motivate the design and allow the analysis of the preconditioners. Most of the material covered in the derivation and analysis of the preconditioners is original work.

A typical example that gives rise to a system of the form (8.1) is the Neumann boundary control for an elliptic equation. We used this example to illustrate our results. In order to test our preconditioners, we had to choose particular preconditioners for the submatrices. The choice of these particular preconditioners is of course as problem-dependent as the general construction of preconditioners for systems of a certain structure. We took the respective diagonals of the matrices H_y and H_u as their preconditioners, and we computed a sparse LU -factorization of A . The iterations of MINRES and SYMMLQ on the original

system generally rose with increasing dimensions. In our numerical experiments we have seen that by preconditioning the number of iterations was considerably reduced to a small constant in some cases. In these 'good' cases, the number of iterations needed by MINRES and SYMMLQ seems to be independent of the grid size, and thus of the dimension of the underlying system. Here, the first preconditioner typically changed the system matrix in such a way that around 20 iterations were necessary, while it were around 30 for the second preconditioner, and 10 for the third. In the evaluation of these results we have to take into account that the application of the third preconditioner is essentially twice as expensive as the application of the other two preconditioners. What we referred to as 'good' cases are the situations where $\alpha = 1$ and $D_y = 0$. The preconditioners can handle well a rise in the diagonal of H_u . However, two cases are problematic. If the parameter α becomes small, the first and second preconditioner were able to reduce the iterations only to a certain extent. The third preconditioner seems to perform well even in the presence of a small α , but, again, we have to take into account that this preconditioner is essentially twice as costly as the other two. Many questions are still open if the diagonal of H_y increases by a large amount. In this situation, the performance of all three preconditioners was considerably less favorable than in the preceding cases. In most situations, the preconditioners designed for the system (8.1) lead to a substantial improvement.

8.2 Future Work

In our numerical experiments we used an LU -factorization of the submatrix A in the Karush–Kuhn–Tucker system. The construction of our preconditioners does not require an exact factorization, not even of one of the submatrices. A numerical study of the performance of our preconditioners when an approximation \tilde{A} of A is available will be done in the future. Then the preconditioned systems in the numerical experiments will be of an even more general form than those we worked with already.

We did not yet apply an interior–point method, but only tried to simulate its effects on the system we dealt with. The implementation of an interior–point method will be done in the near future. Then the improved conditioning of the system, resulting from the use of preconditioners, can be really studied for this case. Here, especially an increase in the diagonal of the matrix H_y (this can correspond to the degenerate case in linear programming) is of interest. We hope that our preconditioners lead to a significant improvement in the performance of MINRES and SYMMLQ. In the analysis of our numerical experiments we have mentioned constantly that an increase in the diagonal of the matrix H_y still causes problems. However, if interior–point methods are applied, only some of the entries become large. So we can hope that our preconditioners give good results.

In the analysis of the first preconditioner we relied on the estimate

$$\|M_y^{1/2}A^{-1}BM_u^{-1/2}\| \leq c$$

that is not proven for the general case in the present work. The estimate holds true in

our application and can be shown to be valid in a more general framework. The general derivation will be done in [2].

Another interesting point that is still open for future work is the extension of preconditioning to indefinite preconditioners. In this work we only considered positive definite preconditioners. However, indefinite preconditioners are possible as well. The *Quasi-Minimum Residual Method*, QMR, is a Krylov subspace method for general nonsymmetric matrices that can handle non-positive definite preconditioners (see [5]). For symmetric matrices and a positive definite preconditioner, the preconditioned version of QMR is identical to the preconditioned MINRES algorithm.

Bibliography

- [1] O. AXELSSON, *Iterative Solution Methods*, Cambridge University Press, Cambridge, London, New York, 1994.
- [2] A. BATTERMANN AND M. HEINKENSCHLOSS, *Preconditioners for Karush–Kuhn–Tucker Systems Arising in Optimal Control*, Tech. Rep. ICAM Report in preparation, Interdisciplinary Center for Applied Mathematics, Blacksburg VA 24061, 1996.
- [3] L. COLLATZ AND W. WETTERLING, *Optimierungsaufgaben*, Springer-Verlag, Berlin, Heidelberg, New York, 1971.
- [4] A. S. EL-BAKRY, R. A. TAPIA, T. TSUCHIYA, AND Y. ZHANG, *On the formulation and theory of the primal–dual Newton interior–point method for nonlinear programming*, TR92–40, Dept. of Computational & Applied Mathematics, Rice University, Houston, Texas, 1992. (revised May, 1993).
- [5] R. W. FREUND AND F. JARRE, *A qmr-based interior-point method for solving linear programs*, Mathematical Programming, Series B, (to appear).
- [6] P. E. GILL, W. MURRAY, D. B. PONCELEÓN, AND M. A. SAUNDERS, *Preconditioners for indefinite systems arising in optimization*, SIAM J. Matrix Anal. Appl., 13 (1992), pp. 292–311.
- [7] ———, *Solving reduced KKT systems in barrier methods for linear programming*, in Numerical Analysis 1993, D. F. Griffith and G. A. Watson, eds., Pitman Research Notes Mathematics, Vol. 303, 1994, pp. 89–104.
- [8] G. GOLUB AND C. F. VAN LOAN, *Matrix Computations*, John Hopkins University Press, Baltimore, London, 1989.
- [9] M. HEINKENSCHLOSS, *Krylov subspace methods for the solution of linear systems and linear least squares problems*, Tech. Rep. Lecture Notes, Interdisciplinary Center for Applied Mathematics, Virginia Polytechnic Institute and State University, Blacksburg, VA 24061, 1995.
- [10] R. HORST, *Nichtlineare Optimierung*, Carl Hanser Verlag, München, Wien, 1979.

- [11] C. JOHNSON, *Numerical Solution of Partial Differential Equations by the Finite Element Method*, Cambridge University Press, Cambridge, New York, Melbourne, Sidney, 1987.
- [12] J. L. LIONS, *Optimal Control of Systems Governed by Partial Differential Equations*, Springer Verlag, Berlin, Heidelberg, New York, 1971.
- [13] C. C. PAIGE AND M. A. SAUNDERS, *Solution of sparse indefinite systems of linear equations*, SIAM J. Numer. Anal., 12 (1975), pp. 617–629.
- [14] T. RUSTEN AND R. WINTHER, *A preconditioned iterative method for saddlepoint problems*, SIAM J. Matrix Anal. Appl., 13, No. 3 (1992), pp. 887 – 904.
- [15] J. STOER, *Solution of large linear systems of equations by conjugate gradient type methods*, in *Mathematical Programming, The State of The Art*, A. Bachem, M. Grötschel, and B. Korte, eds., Springer Verlag, Berlin, Heidelberg, New-York, 1983, pp. 540–565.
- [16] A. WATHEN, B. FISCHER, AND D. SILVESTER, *The convergence rate of the minimum residual method for the Stokes problem*, Numerische Mathematik, 71 (1995), pp. 121–134.
- [17] M. H. WRIGHT, *Interior point methods for constrained optimization*, in *Acta Numerica 1992*, A. Iserles, ed., Cambridge University Press, Cambridge, London, New York, 1992, pp. 341–407.
- [18] Y. ZHANG, R. A. TAPIA, AND F. POTRA, *On the superlinear convergence of interior point algorithms for a general class of problems*, SIAM J. on Optimization, 3 (1993), pp. 413–422.

Vita

Astrid Battermann was born on July 23rd, 1973, in Hannover, Germany. She studied Applied Mathematics and Business Administration from October 1992 until July 1995 at the Universität Trier, Trier, Germany. She came to Virginia Polytechnic Institute and State University, Blacksburg, Virginia, U. S. A., as an exchange student in August 1995 and received the M. S. in July 1996. Currently, she is pursuing the Diplom at the Universität Trier, Trier, Germany.