Block Krylov methods for Hermitian linear systems

Diploma Thesis
Thomas Schmelzer

Block Krylov methods for Hermitian linear systems

Supervisors
Professor Martin H. Gutknecht, ETH Zürich
Professor Eberhard Schock, University of Kaiserslautern

30th August 2004

University of Kaiserslautern
Department of Mathematics
Gottlieb-Daimler-Straße
P.O. Box 3049, 67663 Kaiserslautern
Germany

Acknowledgements

This thesis would not have been possible without the assistance from many people who gave their support in different ways. To them I would like to convey my heartfelt gratitude and sincere appreciation. It gives me great pleasure to acknowledge the guidance, suggestions, constructive criticism and incredible patience provided by my supervisor Professor Martin Gutknecht. The various talks with him have always been a source of inspiration. Large parts of this work are based on his approach to this topic.

When I came to Zürich I skipped my plan to leave Zürich after one term very soon. I was overwhelmed and impressed by the ETH Zürich, the city of Zürich and Switzerland. My 18 months in Zürich have been wonderful. There was only a short interruption when I left Zürich to stay six weeks in Brisbane, Australia where I worked with Dr Roger Sidje and Professor Lloyd Nick Trefethen. During my studies in Oxford it was Professor Trefethen who convinced me to work on the fascinating field of numerical analysis. I am looking forward to starting my DPhil in a few weeks under his supervision.

As a student I was a globetrotter. However, my academic roots lie at the University of Kaiserslautern. The local graduate school in Kaiserslautern offered me the chance to study abroad for 2 years. I thank Dr Burckhard Strehl, Dr Falk Triebsch and Professor Eberhard Schock. Professor Schock is also the second supervisor of this thesis and introduced my to calculus and functional analysis.

I feel privileged to have been able to discuss my work with Professor Beresford Parlett who stayed as a visitor for several weeks in Zürich.

I acknowledge financial support for my studies provided by the Stifung der Deutschen Wirtschaft, the Karolina Rüedi Stiftung, the University of Kaiserslautern and the Erasmus scheme.

I am grateful to my friend and former flatmate Gustav H. Holzegel for the moral support during the last few months. Finally, I thank my parents, Rupert and Brigitte Schmelzer, for their continuous support in my academic endeavors.

Thank you all for your support in the completion of this degree.

Contents

1. Introduction

> During World War II a computer was a human being, usually female; several of them would work together to solve systems of difference equations derived from differential equations
>
> *(Beresford Parlett)*

In this thesis we generalize MinRes and SymmLQ for Hermitian linear systems with multiple right-hand sides.

1.1. The problem and its historical background

In 1975 Christopher Paige and Michael Saunders [21] proposed two iterative Krylov subspace methods called MinRes and SymmLQ for solving sparse Hermitian indefinite[1] linear systems

$$\mathbf{A}\mathbf{x} = \mathbf{b} \tag{1.1}$$

with $\mathbf{A} = \mathbf{A}^{\mathsf{H}} \in \mathbb{C}^{N \times N}$, $\mathbf{x} \in \mathbb{C}^{N}$ and $\mathbf{b} \in \mathbb{C}^{N}$.

In a more general context $\mathbf{A}$ can be regarded as a black box operator. For the methods discussed here it is only required to compute $\mathbf{Ay}$ for any $\mathbf{y} \in \mathbb{C}^{N}$. If $\mathbf{A}$ is indeed given in an explicit matrix form this multiplication is of complexity $O(N)$ instead of $O(N^2)$ if the matrix is sufficiently sparse. In particular these methods do not destroy the sparsity structure of the matrix. So although all these methods can be also applied to dense systems, they are especially suitable for sparse systems.

Almost 25 years earlier Eduard Stiefel and Magnus Hestenes [15] had introduced the method of conjugate gradients for symmetric positive definite systems. Paige and Saunders noted that this approach is closely related to the symmetric Lanczos process [16] and explained why the method of conjugate gradients can often be applied with success to indefinite systems. Although the idea of generalizing the method of conjugate gradients is quite attractive, in order to motivate all those ideas we will not walk this

[1] Since $\mathbf{A}$ is Hermitian all eigenvalues of $\mathbf{A}$ are real numbers, i.e.

$$\Lambda(\mathbf{A}) := \{\lambda \in \mathbb{C}\,;\ \det(\lambda\mathbf{I} - \mathbf{A}) = 0\} \subset \mathbb{R}$$

If $\Lambda(\mathbf{A}) \subset \mathbb{R}_{+}$ or $\Lambda(\mathbf{A}) \subset \mathbb{R}_{-}$ then $\mathbf{A}$ is called definite, otherwise indefinite.

historical paths, although we sketch them[2]. We will use an approach based on variational principles. After the review of this methods we will generalize them for solving sparse linear Hermitian systems with s multiple right-hand sides $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_\mathsf{s}$, i.e.

$$\mathbf{AX} = \mathbf{B} \tag{1.2}$$

with a nonsingular matrix $\mathbf{A} = \mathbf{A}^\mathsf{H} \in \mathbb{C}^{N \times N}$, $\mathbf{B} \in \mathbb{C}^{N \times \mathsf{s}}$ and $\mathbf{X} \in \mathbb{C}^{N \times \mathsf{s}}$. We denote by $\mathbf{x}_i$ and $\mathbf{b}_i$ the ith columns of $\mathbf{X}$ and $\mathbf{B}$ respectively; thus (1.2) is the compact form of s linear systems $\mathbf{A}\mathbf{x}_i = \mathbf{b}_i$.

The idea of block Krylov methods is to construct one common space for the approximations, spanned by the Krylov subspaces related to each of the s systems. It is more expensive to construct an orthonormal basis for the block Krylov space than to construct s orthonormal bases for each of the Krylov subspaces. We hope that this effect is weaker than the convergence in fewer iterations we expect as the dimension of the block Krylov space is up to s-times higher than those of the single Krylov subspaces.

There are various problems we have to be aware of in the case of blocks. The dimension of the block Krylov space can be smaller as the sum of the dimensions of the s Krylov subspaces. The speed of convergence varies for the different right hand sides. All this problems demand a degree of care that was not necessary for solving single systems. So the central question is:

Are all these efforts worth the trouble?

In order to answer this question we perform a series of experiments.

1.2. Possible applications

Linear systems have a central position in scientific computing. The discretization of a partial differential equation by means of finite differences or finite elements usually leads to large and sparse linear systems. One of the most prominent sources for symmetric but indefinite linear systems stems from an appropriate discretization of the Stokes equations [5].

In equation (1.1) the matrix $\mathbf{A}$ might represent a differential operator including the geometry of the problem. Boundary conditions might appear in the vector $\mathbf{b}$. The approximation is described by a linear combination in a finite dimensional function space. The vector of coefficients is $\mathbf{x}$. Approximating the same partial differential equation with different boundary conditions could be an application for a block Krylov method.

The discretization error can be estimated by various techniques. A second error is unavoidable when we approximate the solution of the linear system we got. It will make

[2] Those tangled paths are fascinating. A historical perspective is provided in [24, 28]

no sense if the second error is on a smaller scale as the discretization error. So the goal is to achieve a certain accuracy usually given by constraints known a priori.

Trefethen and Bau [33] identify here the "philosophical center of the scientific computing of the future: The traditional view of computer scientists is that a computational problem is finite. Over the years, however, this view has come to be appropriate to fewer and fewer problems. The best method for large-scale computational problems are usually approximate ones, methods that obtain a satisfactorily accurate solution in a short time rather than an exact one in a much longer or infinite time".

1.3. Outline

In Chapter 2 the Lanczos process[3] is introduced as a method to compute an orthonormal basis of a Krylov subspace using the Gram-Schmidt algorithm. Here the Lanczos process is the basis for sophisticated algorithms as MinRes and SymmLQ in order to solve Hermitian linear systems. However, in practice those methods are not as sophisticated as we expect from theory. For large matrices they tend to fail, that is, they do not converge. A remedy is preconditioning introduced in Chapter 3.

Chapter 4 starts with experiments visualizing the loss of orthogonality in the Lanczos process. By defining block Krylov spaces and the numerical column rank of a matrix we lay the foundations for the block Lanczos algorithm. It is analyzed in exact and non-exact arithmetic. The loss of orthogonality is observable in various experiments. The influence of the starting vectors and a possible deflation scheme are investigated. Comparing the Lanczos process with the Arnoldi methods gives further insights.

The next two chapters treat an efficient block QR decomposition based on Householder reflections and a block recurrence relation. Actually MinRes and SymmLQ are recurrence relations based on the QR decomposition of a matrix gained in the Lanczos process.

We deduce block versions of both methods and give numerical results. In the last chapter we give an overview about the results gained in this work.

[3] Often the Lanczos algorithm is regarded as a combination of the Gram-Schmidt algorithm and an additional Rayleigh-Ritz iteration approximating eigenpairs. If on a cocktail party this issue is addressed it is enough to know this definition due to Beresford Parlett.

2. Iterative Krylov space methods

It is striking how long it took for the true worth of some algorithms to be appreciated.

(Beresford Parlett)

In this chapter we focus on iterative Krylov methods for Hermitian systems. The chapter is based on lecture notes of Martin H. Gutknecht [13]. We introduce the concept of Krylov subspaces, which provides the framework to talk about methods as MinRes and SymmLQ.

2.1. Krylov subspaces

The iterative method of choice for solving a Hermitian system (1.1) if $\mathbf{A}$ is definite[1], is the method of conjugate gradients - the most prominent method of a family of so-called Krylov space methods. A Krylov subspace $\mathcal{K}_n(\mathbf{A}, \mathbf{v}_0)$ where $n \geq 1$ is spanned by a given vector $\mathbf{v}_0$ and increasing powers of $\mathbf{A}$ applied to $\mathbf{v}_0$, up to the $(n-1)$th power:

$$\mathcal{K}_n(\mathbf{A}, \mathbf{v}_0) = \mathsf{span}\left\{\mathbf{v}_0, \mathbf{A}\mathbf{v}_0, \ldots, \mathbf{A}^{n-1}\mathbf{v}_0\right\}$$

If the vectors $\mathbf{v}_0, \mathbf{A}\mathbf{v}_0, \ldots, \mathbf{A}^{n-1}\mathbf{v}_0$ are linearly independent we note that $\dim \mathcal{K}_n(\mathbf{A}, \mathbf{v}_0) = n$. As soon as $\mathbf{A}^n\mathbf{v}_0 \in \mathcal{K}_n(\mathbf{A}, \mathbf{v}_0)$ we obtain that $\mathcal{K}_{n+i}(\mathbf{A}, \mathbf{v}_0) = \mathcal{K}_n(\mathbf{A}, \mathbf{v}_0)$ for all $i \in \mathbb{N}$ and we get a linear combination

$$\mathbf{0} = c_0\mathbf{v}_0 + c_1\mathbf{A}\mathbf{v}_0 + \ldots + c_{n-1}\mathbf{A}^{n-1}\mathbf{v}_0 + c_n\mathbf{A}^n\mathbf{v}_0$$

where at least $c_n \neq 0$ and $c_0 \neq 0$, otherwise the multiplication by $\mathbf{A}^{-1}$ would contradict the assumption the vectors $\mathbf{v}_0, \mathbf{A}\mathbf{v}_0, \ldots, \mathbf{A}^{n-1}\mathbf{v}_0$ are linearly independent. Indeed as $c_0 \neq 0$ we get

$$\mathbf{A}^{-1}\mathbf{v}_0 = \frac{-1}{c_0}\sum_{i=1}^{n} c_i\mathbf{A}^{i-1}\mathbf{v}_0 \in \mathcal{K}_n(\mathbf{A}, \mathbf{v}_0).$$

The smallest index n with $n = \dim \mathcal{K}_n(\mathbf{A}, \mathbf{v}_0) = \dim \mathcal{K}_{n+1}(\mathbf{A}, \mathbf{v}_0)$ is called the **grade of A with respect to $\mathbf{v}_0$** and denoted by $\bar{\nu}(\mathbf{A}, \mathbf{v}_0)$. The above argument also shows that there is no smaller Krylov subspace which contains $\mathbf{A}^{-1}\mathbf{v}_0$; therefore

$$\bar{\nu}(\mathbf{A}, \mathbf{v}_0) = \min\left\{n : \ \mathbf{A}^{-1}\mathbf{v}_0 \in \mathcal{K}_n(\mathbf{A}, \mathbf{v}_0)\right\}.$$

[1] To be correct $\mathbf{A}$ has to be positive definite, but we apply the method of conjugate gradients to $-\mathbf{A}\mathbf{x} = -\mathbf{b}$ if $\mathbf{A}$ is negative definite.

The degree of the minimum polynomial of $\mathbf{A}$ is an upper bound for $\bar{\nu}(\mathbf{A}, \mathbf{v}_0)$. A major idea is to construct a sequence of approximations getting closer to the exact solution $\mathbf{x}_*$, such that[2]

$$\mathbf{x}_n \in \mathbf{x}_0 + \mathcal{K}_n(\mathbf{A}, \mathbf{r}_0) \tag{2.1}$$

where $\mathbf{x}_0$ is the initial approximation,

$$\mathbf{r}_i = \mathbf{b} - \mathbf{A}\mathbf{x}_i \tag{2.2}$$

is the ith residual and

$$\mathbf{f}_i = \mathbf{x}_i - \mathbf{x}_* \tag{2.3}$$

denotes the ith error. After $\bar{\nu}(\mathbf{A}, \mathbf{v}_0)$ iterations the exact solution is contained in the current affine Krylov subspace, i.e.

$$\mathbf{x}_* \in \mathbf{x}_0 + \mathcal{K}_{\bar{\nu}}(\mathbf{A}, \mathbf{r}_0) \tag{2.4}$$

as $\mathbf{x}_0 + \mathbf{A}^{-1}\mathbf{r}_0 = \mathbf{x}_0 + \mathbf{A}^{-1}(\mathbf{A}\mathbf{x}_* - \mathbf{A}\mathbf{x}_0) = \mathbf{x}_*$. This finite termination property does not hold in finite precision arithmetic.

2.2. The Lanczos algorithm

The construction of an orthonormal basis by the Arnoldi process or Lanczos process is the common starting point for all methods introduced here, that is,

$$\mathcal{K}_n(\mathbf{A}, \mathbf{v}_0) = \mathsf{span}\left\{\mathbf{v}_0, \mathbf{A}\mathbf{v}_0, \ldots, \mathbf{A}^{n-1}\mathbf{v}_0\right\} = \mathsf{span}\left\{\mathbf{y}_0, \mathbf{y}_1, \ldots, \mathbf{y}_{n-1}\right\}$$

where $n \le \bar{\nu}(\mathbf{A}, \mathbf{v}_0)$. For Hermitian matrices the Lanczos process constructs in exact arithmetic an orthonormal basis by applying the modified Gram-Schmidt method. The grade $\bar{\nu}(\mathbf{A}, \mathbf{v}_0)$ of $\mathbf{A}$ with respect to $\mathbf{v}_0$ is not a priori known. The iterative process stops once the Krylov space is exhausted.

Algorithm 1 (Lanczos algorithm) .
Let a Hermitian matrix $\mathbf{A}$ *and a vector* $\mathbf{y}_0$ *with* $\|\mathbf{y}_0\|_2 = 1$ *be given. For constructing a nested set of orthonormal bases* $\{\mathbf{y}_0, \mathbf{y}_1, \ldots, \mathbf{y}_m\}$ *for the nested Krylov subspaces* $\mathcal{K}_{m+1}(\mathbf{A}, \mathbf{y}_0)$ $(m = 1, 2, \ldots, \bar{\nu}(\mathbf{y}_0, \mathbf{A}) - 1)$ *compute, for* $n = 1, 2, \ldots, m$*:*

1. *Apply* $\mathbf{A}$ *to* $\mathbf{y}_{n-1} \perp \mathcal{K}_{n-1}(\mathbf{A}, \mathbf{y}_0)$*:*

$$\widetilde{\mathbf{y}}_n := \mathbf{A}\mathbf{y}_{n-1}. \tag{2.5}$$

[2] In SymmLQ we use $\mathbf{x}_n \in \mathbf{x}_0 + \mathbf{A}\mathcal{K}_n(\mathbf{A}, \mathbf{r}_0)$

2. Subtract the projection of $\widetilde{\mathbf{y}}_n$ on the last two basis vectors:

$$\begin{aligned} \widetilde{\mathbf{y}}_n &:= \widetilde{\mathbf{y}}_n - \mathbf{y}_{n-2}\beta_{n-2} \quad \textit{if } n > 1, && (2.6) \\ \alpha_{n-1} &:= \langle \mathbf{y}_{n-1}, \widetilde{\mathbf{y}}_n \rangle \ , && (2.7) \\ \widetilde{\mathbf{y}}_n &:= \widetilde{\mathbf{y}}_n - \mathbf{y}_{n-1}\alpha_{n-1} \, . && (2.8) \end{aligned}$$

3. Normalize the vector $\widetilde{\mathbf{y}}_n \perp \mathcal{K}_n(\mathbf{A}, \mathbf{y}_0)$ if $\|\widetilde{\mathbf{y}}_n\| > 0$:

$$\begin{aligned} \beta_{n-1} &:= \|\widetilde{\mathbf{y}}_n\| \, , && (2.9) \\ \mathbf{y}_n &:= \widetilde{\mathbf{y}}_n / \beta_{n-1} \, . && (2.10) \end{aligned}$$

The method stops if $\|\widetilde{\mathbf{y}}_n\| = 0$. Then $n = \bar{\nu}(\mathbf{A}, \mathbf{y}_0)$

Theorem 1 *The vectors $\left\{\mathbf{y}_0, \mathbf{y}_1, \ldots, \mathbf{y}_{\bar{\nu}(\mathbf{A},\mathbf{y}_0)-1}\right\}$ constructed by this algorithm are orthonormal. The first n vectors are a basis for $\mathcal{K}_n(\mathbf{A}, \mathbf{y}_0)$ for every $n = 1, 2, \ldots \bar{\nu}(\mathbf{A}, \mathbf{y}_0)$. Moreover,*

$$\mathbf{A}\mathbf{Y}_n = \mathbf{Y}_{n+1}\underline{\mathbf{T}}_n \tag{2.11}$$

where $n < \bar{\nu}(\mathbf{A}, \mathbf{y}_0)$,

$$\mathbf{Y}_n = \begin{pmatrix} \mathbf{y}_0 & \mathbf{y}_1 & \ldots & \mathbf{y}_{n-1} \end{pmatrix},$$

$$\underline{\mathbf{T}}_n = \left(\begin{array}{ccccc} \alpha_0 & \beta_0 & & & \\ \beta_0 & \alpha_1 & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & \beta_{n-2} \\ & & & \beta_{n-2} & \alpha_{n-1} \\ \hline & & & & \beta_{n-1} \end{array}\right) \in \mathbb{R}^{n+1\times n} \tag{2.12}$$

and $\beta_i > 0$ for all $i = 1, 2, \ldots n-1$.

We denote with $\mathbf{T}_n$ the upper Hermitian square matrix of $\underline{\mathbf{T}}_n$.

We use induction to prove the theorem. Later in this work we will use a slight modification of this proof when we introduce the block Lanczos process. So it is worth to spend some efforts here.

Proof: The vector $\mathbf{y}_0$ is an orthonormal basis for $\mathcal{K}_1(\mathbf{A}, \mathbf{y}_0)$. Applying (2.5) we get $\widetilde{\mathbf{y}}_1 := \mathbf{A}\mathbf{y}_0 \in \mathcal{K}_2(\mathbf{A}, \mathbf{y}_0)$. As

$$\overline{\alpha_0} = \overline{\langle \mathbf{y}_0, \mathbf{A}\mathbf{y}_0 \rangle} = \overline{\langle \mathbf{A}^{\mathsf{H}}\mathbf{y}_0, \mathbf{y}_0 \rangle} = \overline{\langle \mathbf{A}\mathbf{y}_0, \mathbf{y}_0 \rangle} = \langle \mathbf{y}_0, \mathbf{A}\mathbf{y}_0 \rangle = \alpha_0$$

we have $\alpha_0 \in \mathbb{R}$. By the definition of α_0 the vector $\widetilde{\mathbf{y}}_1$ (2.8) is orthogonal to $\mathbf{y}_0$

$$\langle \mathbf{y}_0, \widetilde{\mathbf{y}}_1 \rangle = \langle \mathbf{y}_0, \mathbf{A}_0\mathbf{y}_0 - \alpha_0 \mathbf{y}_0 \rangle = \langle \mathbf{y}_0, \mathbf{A}\mathbf{y}_0 \rangle - \alpha_0 = 0.$$

If $\widetilde{\mathbf{y}}_1 = \mathbf{0}$ (2.8) then $\mathbf{y}_0$ is an eigenvector of $\mathbf{A}$ and $\bar{\nu}(\mathbf{A}, \mathbf{y}_0) = 1$. Otherwise $\widetilde{\mathbf{y}}_1 \neq \mathbf{0}$ and in particular $\beta_0 > 0$. With $\mathbf{y}_1 = \widetilde{\mathbf{y}}_1 / \beta_0$ we get

$$\beta_0 \mathbf{y}_1 = \mathbf{A}\mathbf{y}_0 - \mathbf{y}_0 \alpha_0$$

or rearranged

$$\mathbf{A}\mathbf{Y}_1 = \mathbf{Y}_2 \begin{pmatrix} \alpha_0 \\ \beta_0 \end{pmatrix}.$$

Assume the statement holds for n. Then

- $\{\mathbf{y}_0, \ldots, \mathbf{y}_{n-1}\}$ is an orthonormal basis for $\mathcal{K}_n(\mathbf{A}, \mathbf{y}_0)$,
- $\mathbf{y}_{n-1} \perp \mathcal{K}_{n-1}(\mathbf{A}, \mathbf{y}_0)$,
- the three-term recurrence relation with real coefficients $\beta_{n-3}, \alpha_{n-2}$ and β_{n-2}

$$\mathbf{A}\mathbf{y}_{n-2} = \mathbf{y}_{n-3}\beta_{n-3} + \mathbf{y}_{n-2}\alpha_{n-2} + \mathbf{y}_{n-1}\beta_{n-2} \tag{2.13}$$

  holds.

Let $\widetilde{\mathbf{y}}_n = \mathbf{A}\mathbf{y}_{n-1} \in \mathcal{K}_{n+1}(\mathbf{A}, \mathbf{y}_0)$. We remark that $\widetilde{\mathbf{y}}_n \perp \mathcal{K}_{n-2}(\mathbf{A}, \mathbf{y}_0)$. Assume $\mathbf{g} \in \mathcal{K}_{n-2}(\mathbf{A}, \mathbf{y}_0)$, then

$$\langle \mathbf{g}, \widetilde{\mathbf{y}}_n \rangle = \left\langle \mathbf{A}^{\mathsf{H}}\mathbf{g}, \mathbf{y}_{n-1} \right\rangle = \langle \mathbf{A}\mathbf{g}, \mathbf{y}_{n-1} \rangle = 0.$$

as $\mathbf{A}\mathbf{g} \in \mathcal{K}_{n-1}(\mathbf{A}, \mathbf{y}_0)$ but $\mathbf{y}_{n-1} \perp \mathcal{K}_{n-1}(\mathbf{A}, \mathbf{y}_0)$. It is this property that makes the difference between the Lanczos and the Arnoldi process. If $\mathbf{A}$ is not Hermitian the vector $\widetilde{\mathbf{y}}_n$ has to be orthogonalized with respect to all basis vectors $\mathbf{y}_0, \ldots, \mathbf{y}_{n-1}$.

Using exactly the same argument as above we conclude that $\alpha_{n-1} \in \mathbb{R}$.

The vector $\widetilde{\mathbf{y}}_n$ (2.8) is orthogonal to the vectors $\mathbf{y}_{n-2}$ and $\mathbf{y}_{n-1}$ as

$$\langle \mathbf{y}_{n-2}, \mathbf{A}\mathbf{y}_{n-1} - \mathbf{y}_{n-1}\alpha_{n-1} - \mathbf{y}_{n-2}\beta_{n-2} \rangle = \langle \mathbf{A}\mathbf{y}_{n-2}, \mathbf{y}_{n-1} \rangle - \beta_{n-2} = 0$$

by using the recurrence relation (2.13) and

$$\langle \mathbf{y}_{n-1}, \mathbf{A}\mathbf{y}_{n-1} - \mathbf{y}_{n-1}\alpha_{n-1} - \mathbf{y}_{n-2}\beta_{n-2} \rangle = \langle \mathbf{y}_{n-1}, \mathbf{A}\mathbf{y}_{n-1} \rangle - \alpha_{n-1} = 0$$

by the definition of α_{n-1}. If $\beta_{n-1} = 0$ we conclude $\bar{\nu}(\mathbf{A}, \mathbf{y}_0) = n$. Otherwise the recurrence relation

$$\mathbf{y}_n\beta_{n-1} = \mathbf{A}\mathbf{y}_n - \mathbf{y}_{n-1}\alpha_{n-1} - \mathbf{y}_{n-2}\beta_{n-2}.$$

holds with real coefficients $\beta_{n-2}, \alpha_{n-1}$ and β_{n-1}. □

There is also an important generalization. Given a Hermitian positive definite matrix $\mathbf{M}$ it is possible to construct an $\mathbf{M}$-orthonormal bases for the nested Krylov subspaces $\mathcal{K}_{m+1}\left(\mathbf{M}^{-1}\mathbf{A}, \mathbf{y}_0\right)$ where $\|\mathbf{y}_0\|_{\mathbf{M}} = 1$. In each iteration it is enough to solve additionally a linear system $\mathbf{z} = \mathbf{M}^{-1}\mathbf{w}$. It is not necessary to compute the $\mathbf{M}$-inner products explicitly. Details are given in the book by Saad [26, Chapter 9.3.1]. The idea to approximate the system in a different Krylov subspace as $\mathcal{K}_m\left(\mathbf{M}^{-1}\mathbf{A}, \mathbf{y}_0\right)$ is widely used in practice.

The condition (2.1) yields a simple scheme

$$\mathbf{x}_n = \mathbf{x}_0 + \mathbf{Y}_n\mathbf{k}_n \tag{2.14}$$

and

$$\mathbf{r}_n = \mathbf{r}_0 - \mathbf{A}\mathbf{Y}_n\mathbf{k}_n \tag{2.15}$$

in which $\mathbf{k}_n$ contains the coordinates of $\mathbf{x}_n - \mathbf{x}_0$ in terms of the Lanczos basis. The freedom to choose those coordinates results in a variety of different Krylov methods.

2.3. The minimal residual method

The first idea is to minimize the 2-norm of the residual, that is we identify $\mathbf{k}_n$ in (2.15) such that

$$\|\mathbf{r}_n\|_2 = \min_{\mathbf{k}_n \in \mathbb{C}^n} \|\mathbf{r}_0 - \mathbf{A}\mathbf{Y}_n\mathbf{k}_n\|_2. \tag{2.16}$$

With $\rho_0 = \|\mathbf{r}_0\|_2$ we obtain $\mathbf{r}_0 = \mathbf{Y}_{n+1}\underline{\mathbf{e}}_1\rho_0$ and by inserting $\mathbf{A}\mathbf{Y}_n = \mathbf{Y}_{n+1}\underline{\mathbf{T}}_n$ into the relation (2.15)

$$\|\mathbf{r}_n\|_2^2 = \|\mathbf{Y}_{n+1}\underline{\mathbf{e}}_1\rho_0 - \mathbf{Y}_{n+1}\underline{\mathbf{T}}_n\mathbf{k}_n\|_2^2 = \|\underline{\mathbf{e}}_1\rho_0 - \underline{\mathbf{T}}_n\mathbf{k}_n\|_2^2.$$

The term $\mathbf{q}_n = \underline{\mathbf{e}}_1\rho_0 - \underline{\mathbf{T}}_n\mathbf{k}_n$ is called the nth quasiresidual. Minimizing $\|\mathbf{q}_n\|_2^2$ is an $(n+1) \times n$ least squares problem which could be solved by an update scheme using the tridiagonal structure of $\underline{\mathbf{T}}_n$. In every iteration only one Givens rotation has to be constructed.

Let $\underline{\mathbf{T}}_n = \mathbf{Q}_{n+1}\underline{\mathbf{R}}_n^{\mathrm{MR}}$ be a QR decomposition of $\underline{\mathbf{T}}_n$. Here $\mathbf{Q}_{n+1}$ is a unitary matrix of order $n+1$ and $\underline{\mathbf{R}}_n^{\mathrm{MR}}$ is a banded upper triangular $(n+1) \times n$ matrix whose last row is zero. The abbreviation MR should avoid any confusion with the matrix $\mathbf{R}_n$ of residual vectors. Here

$$\underline{\mathbf{R}}_n^{\mathrm{MR}} = \left(\begin{array}{ccccc} \widetilde{\alpha}_0 & \widetilde{\beta}_0 & \widetilde{\gamma}_0 & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & \widetilde{\gamma}_{n-3} \\ & & & \ddots & \widetilde{\beta}_{n-2} \\ & & & & \widetilde{\alpha}_{n-1} \\ \hline 0 & & \cdots & & 0 \end{array}\right) \quad \text{and} \quad \underline{\mathbf{T}}_n = \left(\begin{array}{ccccc} \alpha_0 & \beta_0 & & & \\ \beta_0 & \alpha_1 & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & \beta_{n-2} \\ & & & \beta_{n-2} & \alpha_{n-1} \\ \hline 0 & \cdots & \cdots & 0 & \beta_{n-1} \end{array}\right). \tag{2.17}$$

We denote its upper square $n \times n$ submatrix by $\mathbf{R}_n^{\mathrm{MR}}$. This matrix is invertible.

COROLLARY 2 *In exact arithmetic the smallest singular value of* $\mathbf{R}_n^{\mathrm{MR}}$ *is larger than the smallest singular value of* $\mathbf{A}$.

$$\inf(\mathbf{R}_n^{\mathrm{MR}}) \geq \inf(\mathbf{A}) \tag{2.18}$$

PROOF: It is

$$\inf(\mathbf{R}_n^{\mathrm{MR}}) = \inf(\underline{\mathbf{R}}_n^{\mathrm{MR}}) = \inf(\mathbf{Q}_{n+1}\underline{\mathbf{R}}_n^{\mathrm{MR}}) = \inf(\underline{\mathbf{T}}_n) = \inf(\mathbf{Y}_{n+1}\underline{\mathbf{T}}_n) = \inf(\mathbf{A}\mathbf{Y}_n).$$

Hence $\inf(\mathbf{R}_n^{\mathrm{MR}}) \geq \inf(\mathbf{A})$ is a consequence of Corollary 14 stated in the Appendix. □

We define[3]

$$\underline{\mathbf{h}}_n :\equiv \left(\frac{\mathbf{h}_n}{\widetilde{\eta}_n} \right) :\equiv \mathbf{Q}_{n+1}^{\mathsf{H}} \underline{\mathbf{e}}_1 \rho_0. \tag{2.19}$$

So $\widetilde{\eta}_n$ is the first entry of the last column of $\mathbf{Q}_{n+1}$ multiplied by ρ_0. In view of

$$\begin{aligned} \|\underline{\mathbf{e}}_1\rho_0 - \underline{\mathbf{T}}_n\mathbf{k}_n\|^2 &= \|\mathbf{Q}_{n+1}^{\mathsf{H}}\underline{\mathbf{e}}_1\rho_0 - \underline{\mathbf{R}}_n^{\mathrm{MR}}\mathbf{k}_n\|^2 \\ &= \|\underline{\mathbf{h}}_n - \underline{\mathbf{R}}_n^{\mathrm{MR}}\mathbf{k}_n\|^2 \\ &= \|\mathbf{h}_n - \mathbf{R}_n^{\mathrm{MR}}\mathbf{k}_n\|^2 + |\widetilde{\eta}_n|^2 \end{aligned} \tag{2.20}$$

we see that

$$\mathbf{k}_n = (\mathbf{R}_n^{\mathrm{MR}})^{-1}\mathbf{h}_n \tag{2.21}$$

is the solution of our least-squares problem and that the corresponding least-squares error equals

$$\|\underline{\mathbf{e}}_1\rho_0 - \underline{\mathbf{T}}_n\mathbf{k}_n\|_2^2 = |\widetilde{\eta}_n|^2. \tag{2.22}$$

Hence the minimum residual norm can therefore be found without computing $\mathbf{k}_n$ or the residual $\mathbf{r}_n$. We will determine the unitary matrix $\mathbf{Q}_{n+1}$ only in its factored form, namely as the product of n Givens rotations that are chosen to annihilate the subdiagonal elements of the tridiagonal matrix:

$$\mathbf{Q}_1 :\equiv 1, \quad \mathbf{Q}_{n+1} :\equiv \begin{pmatrix} \mathbf{Q}_n & \\ & 1 \end{pmatrix} \mathbf{G}_n \quad \text{where} \quad \mathbf{G}_n :\equiv \begin{pmatrix} \mathbf{I}_{n-1} & \\ & \mathbf{G}_n' \end{pmatrix}. \tag{2.23}$$

Here $\mathbf{G}_n'$ is a complex Givens rotation; in particular $\mathbf{G}_n'$ is a 2×2 unitary matrix. It is worth noting

$$\mathbf{G}_n^{\mathsf{H}} \begin{pmatrix} \mathbf{Q}_n^{\mathsf{H}} & \\ & 1 \end{pmatrix} \underline{\mathbf{T}}_n = \underline{\mathbf{R}}_n^{\mathrm{MR}}.$$

The multiplication of $\underline{\mathbf{T}}_n$ by diag $\left(\mathbf{Q}_n^{\mathsf{H}}, 1\right)$ annihilates all subdiagonal elements except the element in last column. Applying diag $\left(\mathbf{Q}_n^{\mathsf{H}}, 1\right)$ on the last column of $\underline{\mathbf{T}}_n$ boils down

[3] The horizontal rule is not a fraction rule but just a visual aid separating the first n components gathered in $\mathbf{h}_n$ from the last component $\widetilde{\eta}_n$.

to the multiplication with two Givens rotations constructed in the last two iterations, that is,

$$\begin{pmatrix} 0 \\ \vdots \\ 0 \\ \widetilde{\gamma}_{n-3} \\ \widetilde{\beta}_{n-2} \\ \mu_n \\ \nu_n \end{pmatrix} :\equiv \begin{pmatrix} \mathbf{Q}_n^{\mathsf{H}} & \\ & 1 \end{pmatrix} \underline{\mathbf{T}}_n \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix}$$

where

$$\begin{pmatrix} \widetilde{\gamma}_{n-3} \\ \widetilde{\beta}_{n-2} \\ \mu_n \end{pmatrix} = \begin{pmatrix} 1 & \\ & \mathbf{G}'^{\mathsf{H}}_{n-1} \end{pmatrix} \begin{pmatrix} \mathbf{G}'^{\mathsf{H}}_{n-2} & \\ & 1 \end{pmatrix} \begin{pmatrix} 0 \\ \beta_{n-2} \\ \alpha_{n-1} \end{pmatrix}. \tag{2.24}$$

and $\nu_n = \beta_{n-1}$. Annihilating the last entry we have to construct a Givens rotation $\mathbf{G}'_n$ such that

$$\begin{pmatrix} \mu_n \\ \nu_n \end{pmatrix} = \mathbf{G}'_n \begin{pmatrix} \widetilde{\alpha}_{n-1} \\ 0 \end{pmatrix}.$$

Therefore the first column of $\mathbf{G}'_n$ contains a multiple of the vector $\begin{pmatrix} \mu_n & \nu_n \end{pmatrix}^{\mathsf{T}}$. The phase factor of μ_n will appear in $\widetilde{\alpha}_{n-1}$ such that the entry $(\mathbf{G}'_n)_{1,1}$ is a positive real number. Scaling the first column of $\mathbf{G}'_n$ to unit length we gain $\widetilde{\alpha}_{n-1} = \sqrt{|\mu_n|^2 + |\nu_n|^2} e^{+i\phi_n}$ where $\mu_n = |\mu_n| e^{+i\phi_n}$. The second column of $\mathbf{G}'_n$ has to be orthogonal to the first to guarantee that $\mathbf{G}'_n$ is unitary. We end up with

$$\begin{pmatrix} \mu_n \\ \nu_n \end{pmatrix} = \frac{1}{\sqrt{|\mu_n|^2 + |\nu_n|^2}} \begin{pmatrix} |\mu_n| & -\overline{\nu_n e^{-i\phi_n}} \\ \nu_n e^{-i\phi_n} & |\mu_n| \end{pmatrix} \begin{pmatrix} \widetilde{\alpha}_{n-1} \\ 0 \end{pmatrix}.$$

There is no unique choice for $\mathbf{G}'_n$. The choice that there is a non-negative entry in the upper left corner is a widely used standard in linear algebra. With the coefficients

$$\boxed{\begin{array}{llr} c_n := \dfrac{|\mu_n|}{\sqrt{|\mu_n|^2 + |\nu_n|^2}}\,, & s_n := \dfrac{\nu_n e^{-i\phi_n}}{\sqrt{|\mu_n|^2 + |\nu_n|^2}} \dfrac{e^{+i\phi_n}|\mu_n|}{\mu_n} = c_n \dfrac{\nu_n}{\mu_n}\,, & \text{if } \mu_n \neq 0\,, \\[2ex] c_n := 0\,, & s_n := 1\,, & \text{if } \mu_n = 0\,. \end{array}} \tag{2.25}$$

we set

$$\mathbf{G}'_n = \begin{pmatrix} c_n & -\overline{s_n} \\ s_n & c_n \end{pmatrix} \qquad \text{with} \quad \mathbf{G}'^{\mathsf{H}}_n = \begin{pmatrix} c_n & \overline{s_n} \\ -s_n & c_n \end{pmatrix}$$

analogue to the real case, where c_n and s_n are the cosine and sine of the rotation angle. In particular $\widetilde{\alpha}_{n-1} = \mu_n c_n + \nu_n \overline{s_n}$.

In view of (2.19) and (2.23) updating $\underline{\mathbf{h}}_{n-1}$ is simple:

$$\underline{\mathbf{h}}_n = \begin{pmatrix} \mathbf{I}_{n-1} & 0 \\ 0 & \mathbf{G}'^{\mathsf{H}}_n \end{pmatrix} \begin{pmatrix} \mathbf{Q}_n^{\mathsf{H}} & 0 \\ 0 & 1 \end{pmatrix} \underline{\mathbf{e}}_1 \rho_0 = \begin{pmatrix} \mathbf{I}_{n-1} & 0 \\ 0 & \mathbf{G}'^{\mathsf{H}}_n \end{pmatrix} \begin{pmatrix} \mathbf{h}_{n-1} \\ \hline \widetilde{\eta}_{n-1} \\ 0 \end{pmatrix}.$$

We get

$$\underline{\mathbf{h}}_n = \left(\frac{\mathbf{h}_n}{\widetilde{\eta}_n} \right) = \left(\begin{array}{c} \mathbf{h}_{n-1} \\ \hline c_n\, \widetilde{\eta}_{n-1} \\ -s_n\, \widetilde{\eta}_{n-1} \end{array} \right). \tag{2.26}$$

In particular, since $\widetilde{\eta}_0 = \|\mathbf{r}_0\|$, it follows by induction that

$$\|\underline{\mathbf{e}}_1 \rho_0 - \underline{\mathbf{T}}_n \mathbf{k}_n\| = |\widetilde{\eta}_n| = |s_n \widetilde{\eta}_{n-1}| = |s_1\, s_2 \cdots s_n|\; \|\mathbf{r}_0\|. \tag{2.27}$$

We rewrite $\mathbf{x}_n = \mathbf{x}_0 + \mathbf{Y}_n \mathbf{k}_n$ by using (2.21) as

$$\mathbf{x}_n = \mathbf{x}_0 + \mathbf{Z}_n \mathbf{h}_n, \qquad \text{where} \quad \mathbf{Z}_n :\equiv \mathbf{Y}_n (\mathbf{R}_n^{\mathrm{MR}})^{-1} \tag{2.28}$$

contains the search directions $\mathbf{z}_0, \ldots, \mathbf{z}_{n-1}$, and therefore

$$\mathbf{x}_n := \mathbf{x}_{n-1} + \mathbf{z}_{n-1} c_n \widetilde{\eta}_{n-1}. \tag{2.29}$$

A recursion for the residual is given by[4]

$$\mathbf{r}_n := \mathbf{r}_{n-1} |s_n|^2 + \mathbf{y}_n c_n \widetilde{\eta}_n. \tag{2.30}$$

However, computing or updating the residual is unnecessary since its norm is equal to $|\widetilde{\eta}_n|$. If this number is smaller than an a priori given tolerance the residual it computed explicitly for a second verification in practice.

The matrix $\mathbf{R}_n^{\mathrm{MR}}$ is a banded upper tridiagonal matrix with bandwidth three. Therefore the relation

$$\mathbf{Y}_n = \mathbf{Z}_n \mathbf{R}_n^{\mathrm{MR}} \tag{2.31}$$

can be viewed as the matrix representation of a three-term recursion for generating the vectors $\{\mathbf{z}_k\}_{k=0}^{n-1}$:

$$\mathbf{z}_k := (\mathbf{y}_k - \mathbf{z}_{k-1} \widetilde{\beta}_{k-1} - \mathbf{z}_{k-2} \widetilde{\gamma}_{k-2}) / \widetilde{\alpha}_k. \tag{2.32}$$

Algorithm 2 (MinRes) .
For solving $\mathbf{A}\mathbf{x} = \mathbf{b}$ *with Hermitian* $\mathbf{A}$ *choose* $\mathbf{x}_0$*, and let* $\mathbf{r}_0 := \mathbf{b} - \mathbf{A}\mathbf{x}_0$*,* $\rho_0 := \|\mathbf{r}_0\|$*,* $\mathbf{y}_0 := \mathbf{r}_0/\rho_0$*,* $\mathbf{z}_{-2} := \mathbf{z}_{-1} := \mathbf{o}$*, and* $\widetilde{\eta}_0 := 1$.
Then, for $n = 1, \ldots, m$*:*

1. *Do one step of the symmetric Lanczos algorithm:*

$$\begin{array}{ll} \widetilde{\mathbf{y}}_n := \mathbf{A}\mathbf{y}_{n-1}\,, & \widetilde{\mathbf{y}}_n := \widetilde{\mathbf{y}}_n - \mathbf{y}_{n-2}\beta_{n-2} \quad \textit{if } n > 1, \\ \alpha_{n-1} := \langle \mathbf{y}_{n-1}, \widetilde{\mathbf{y}}_n \rangle\,, & \widetilde{\mathbf{y}}_n := \widetilde{\mathbf{y}}_n - \mathbf{y}_{n-1}\alpha_{n-1}\,, \\ \beta_{n-1} := \|\widetilde{\mathbf{y}}_n\|\,, & \mathbf{y}_n := \widetilde{\mathbf{y}}_n/\beta_{n-1}\,. \end{array}$$

[4] A proof is given for the general case in Chapter 7

2. *Let* $\widetilde{\alpha}_{n-1} := \alpha_{n-1}$*, and, if* $n > 1$, $\widetilde{\beta}_{n-1} := \beta_{n-1}$.
   If $n > 2$*, apply* $\mathbf{G}_{n-2}^{\mathsf{H}}$ *to the new last column of* $\underline{\mathbf{T}}_n$*:*

$$\begin{pmatrix} \widetilde{\gamma}_{n-3} \\ \widetilde{\beta}_{n-2} \end{pmatrix} := \begin{pmatrix} c_{n-2} & \overline{s_{n-2}} \\ -s_{n-2} & c_{n-2} \end{pmatrix} \begin{pmatrix} 0 \\ \beta_{n-2} \end{pmatrix};$$

   if $n > 1$*, apply* $\mathbf{G}_{n-1}^{\mathsf{H}}$ *to the last column of* $\mathbf{G}_{n-2}^{\mathsf{H}}\underline{\mathbf{T}}_n$*:*

$$\begin{pmatrix} \widetilde{\beta}_{n-2} \\ \widetilde{\alpha}_{n-1} \end{pmatrix} := \begin{pmatrix} c_{n-1} & \overline{s_{n-1}} \\ -s_{n-1} & c_{n-1} \end{pmatrix} \begin{pmatrix} \widetilde{\beta}_{n-2} \\ \alpha_{n-1} \end{pmatrix}.$$

3. *Let* $\mu_n := \widetilde{\alpha}_{n-1}$, $\nu_n := \beta_{n-1}$ *and compute* c_n *and* s_n *of the Givens rotation* $\mathbf{G}_n$ *according to* (2.25)*.*
4. *Apply the adjoint Givens rotation* $\mathbf{G}_n^{\mathsf{H}}$ *to update* $\underline{\mathbf{h}}_{n-1}$ *and the last two components of the modified last column of* $\underline{\mathbf{T}}_n$*:*

$$\eta_{n-1} := c_n \,\widetilde{\eta}_{n-1}\,, \quad \widetilde{\eta}_n := -s_n \,\widetilde{\eta}_{n-1}\,, \qquad \widetilde{\alpha}_{n-1} := c_n\mu_n + \overline{s_n}\nu_n\,.$$

5. *Compute* $\mathbf{z}_{n-1}$ *and* $\mathbf{x}_n$ *according to* (2.32) *and* (2.29)*.*
6. *If* $|\widetilde{\eta}_n| \leq \mathrm{tol}$*, the algorithm terminates and* $\mathbf{x}_n$ *is a sufficiently accurate approximate solution. In practice: 2nd verification by computing* $\|\mathbf{b} - \mathbf{A}\mathbf{x}_n\|_2$ *if* $|\widetilde{\eta}_n| \leq \mathrm{tol}$.

If $\mathbf{A}$ is not Hermitian we use the aforementioned Arnoldi process. The matrix $\underline{\mathbf{T}}_n$ is replaced by an upper Hessenberg matrix $\underline{\mathbf{H}}_n$. It is still puzzling why this modification of MinRes took 11 years. The resulting method is known as GMRes [27].

2.4. Ritz-Galerkin methods

Instead of minimizing the 2-norm of the residual we choose $\mathbf{k}_n$ such that the residual $\mathbf{r}_n$ is orthogonal to the current subspace: $\mathbf{r}_n \perp \mathcal{K}_n(\mathbf{A}, \mathbf{r}_0)$. This implies

$$0 = \mathbf{Y}_n^{\mathsf{H}}\mathbf{r}_n = \mathbf{Y}_n^{\mathsf{H}}(\mathbf{b} - \mathbf{A}\mathbf{x}_n) = \mathbf{Y}_n^{\mathsf{H}}(\mathbf{b} - \mathbf{A}(\mathbf{x}_0 + \mathbf{Y}_n\mathbf{k}_n)) = \mathbf{Y}_n^{\mathsf{H}}(\mathbf{r}_0 + \mathbf{A}\mathbf{Y}_n\mathbf{k}_n)$$

With $\rho_0 = \|\mathbf{r}_0\|_2$ and the identity (2.11) we gain

$$\mathbf{T}_n\mathbf{k}_n = \rho_0\mathbf{e}_1 \quad \text{and} \quad \mathbf{x}_n = \mathbf{x}_0 + \mathbf{Y}_n\mathbf{k}_n \tag{2.33}$$

If $\mathbf{A}$ is indefinite then it is possible that $\mathbf{T}_n$ is not invertible for a certain n. Hence a solution $\mathbf{k}_n$ does not exist or is not unique in every iteration.

If $\mathbf{A}$ is in addition positiv definite we can construct in an implicit way an Cholesky factorization for $\mathbf{T}_n$ without generating $\mathbf{T}_n$ itself. This idea is leading to the method of conjugate gradients.

In view of $\mathbf{r}_n \perp \mathcal{K}_n(\mathbf{A}, \mathbf{r}_0)$ we note that $\mathbf{A}(\mathbf{x}_n - \mathbf{x}_*) \perp \mathcal{K}_n(\mathbf{A}, \mathbf{r}_0)$ or $\mathbf{x}_n - \mathbf{x}_* \perp_{\mathbf{A}} \mathcal{K}_n(\mathbf{A}, \mathbf{r}_0)$, which implies that the error is $\mathbf{A}$-orthogonal to the Krylov subspace. If $\mathbf{A}$ is positiv definite this property is equivalent to observation that $\|\mathbf{x}_n - \mathbf{x}_*\|_{\mathbf{A}}$ is minimal.

So the method of conjugate gradients does not only construct a sequence of orthogonal residuals, it also provides a minimization property for the error $\mathbf{f}_n$. In the literature usually this minimization property is the starting point for an introduction to the method of conjugate gradients. There is even a third approach motivating this method as a modified steepest descent version.

An approach for solving (2.33) for indefinite matrices $\mathbf{A}$ is based on the identity

$$\mathbf{T}_n = \underline{\mathbf{T}}_n^{\mathsf{T}} \begin{pmatrix} \mathbf{I}_n \\ 0 \end{pmatrix} = (\underline{\mathbf{R}}_n^{\mathrm{MR}})^{\mathsf{T}} \mathbf{Q}_{n+1}^{\mathsf{T}} \begin{pmatrix} \mathbf{I}_n \\ 0 \end{pmatrix}.$$

Using the recursive character of $\mathbf{Q}_{n+1}$, that is relation (2.23) and the definition of a Givens rotation we remark

$$\mathbf{T}_n = (\underline{\mathbf{R}}_n^{\mathrm{MR}})^{\mathsf{T}} \begin{pmatrix} \mathbf{I}_{n-1} & & \\ & c_n & \overline{s_n} \\ & -s_n & c_n \end{pmatrix} \begin{pmatrix} \mathbf{Q}_n^{\mathsf{T}} & \\ & 1 \end{pmatrix} \begin{pmatrix} \mathbf{I}_n \\ 0 \end{pmatrix} = (\mathbf{R}_n^{\mathrm{MR}})^{\mathsf{T}} \begin{pmatrix} \mathbf{I}_{n-1} & \\ & c_n \end{pmatrix} \mathbf{Q}_n^{\mathsf{T}}. \tag{2.34}$$

The matrix $\mathbf{R}_n^{\mathrm{MR}}$ is invertible as long as $\beta_n \neq 0$. Hence the matrix $\mathbf{T}_n$ is not invertible if $c_n = 0$, which happens if $\mu_n = 0$ in equation (2.24). This decomposition is exploited in the next section.

2.5. The symmetric LQ method

Paige and Saunders [21] introduced with MinRes a method minimizing the 2-norm of the residual. In the same paper they remarked that the computation of a Cholesky factorization for $\mathbf{T}_n$ is numerically unstable if $\mathbf{T}_n$ is indefinite. They proposed an approach using instead the factorization $\mathbf{T}_n = \mathbf{L}_n \mathbf{Q}_n$[5], where $\mathbf{L}_n$ is lower triangular and $\mathbf{Q}_n$ is an orthogonal matrix. Their successful attempt to develop a stable extension of the method of conjugate gradients lead to an auxiliary iteration in

$$\mathcal{L}_n := \mathbf{A}\mathcal{K}_n(\mathbf{A}, \mathbf{r}_0) \subset \mathcal{K}_{n+1}(\mathbf{A}, \mathbf{r}_0). \tag{2.35}$$

Our approach to SymmLQ is the reverse path first walked by Paige and Saunders. Unfortunately SymmLQ has never gained much attention. Saad [26, Page 189] alludes

[5] We will instead use the QR factorization. Compare with the discussion by Fischer [5, Page 180]

to SymmLQ in only sentence. Actually, SymmLQ is a set of two iterations which has resulted in some confusion in literature[6].

The space $\mathcal{L}_n$ is attractive as it is possible to construct an iteration minimizing the 2-norm of the error $\mathbf{f}_n^{\mathsf{L}} = \mathbf{x}_n^{\mathsf{L}} - \mathbf{x}_*$, without knowing the exact solution $\mathbf{x}_*$. Our approach to SymmLQ starts with this variational principle

$$\|\mathbf{f}_n^{\mathsf{L}}\|_2 = \min_{\mathbf{x}_n^{\mathsf{L}} \in \mathcal{L}_n + \mathbf{x}_0} \|\mathbf{x}_n^{\mathsf{L}} - \mathbf{x}_\star\|_2 \tag{2.36}$$

The columns of $\mathbf{Y}_n$ are still an orthonormal basis of $\mathcal{K}_n(\mathbf{A}, \mathbf{r}_0)$, so that $\mathbf{x}_n^{\mathsf{L}} \in \mathcal{L}_n + \mathbf{x}_0$ has the representation

$$\mathbf{x}_n^{\mathsf{L}} - \mathbf{x}_0 = \mathbf{A}\mathbf{Y}_n\mathbf{k}_n^{\mathsf{L}}. \tag{2.37}$$

Hence the condition (2.36) is

$$\|\mathbf{f}_n^{\mathsf{L}}\|_2 = \min_{\mathbf{k}_n \in \mathbb{C}^n} \|\mathbf{A}\mathbf{Y}_n\mathbf{k}_n^{\mathsf{L}} - (\mathbf{x}_\star - \mathbf{x}_0)\|_2 \tag{2.38}$$

which is a least squares problem with the normal equations

$$\mathbf{Y}_n^{\mathsf{H}}\mathbf{A}^{\mathsf{H}}\mathbf{A}\mathbf{Y}_n\mathbf{k}_n^{\mathsf{L}} = \mathbf{Y}_n^{\mathsf{H}}\mathbf{A}^{\mathsf{H}}(\mathbf{x}_\star - \mathbf{x}_0) \tag{2.39}$$

the Galerkin condition

$$\mathbf{f}_n^{\mathsf{L}} = \mathbf{x}_n^{\mathsf{L}} - \mathbf{x}_\star \perp \mathbf{A}\mathcal{K}_n(\mathbf{A}, \mathbf{r}_0). \tag{2.40}$$

Therefore it is

$$\mathbf{r}_n^{\mathsf{L}} \perp \mathcal{K}_n(\mathbf{A}, \mathbf{r}_0).$$

The exact solution $\mathbf{x}_\star$ and the original error $\mathbf{x}_0 - \mathbf{x}_\star$ are multiplied by $\mathbf{A}^{\mathsf{H}} = \mathbf{A}$, so they need not to be known

$$\mathbf{Y}_n^{\mathsf{H}}\mathbf{A}^{\mathsf{H}}(\mathbf{x}_\star - \mathbf{x}_0) = \mathbf{Y}_n^{\mathsf{H}}\mathbf{A}(\mathbf{x}_\star - \mathbf{x}_0) = \mathbf{Y}_n^{\mathsf{H}}\mathbf{r}_0^{\mathsf{L}} = \rho_0\mathbf{e}_1.$$

On the left-hand side, using the Lanczos relation $\mathbf{A}\mathbf{Y}_n = \mathbf{Y}_{n+1}\underline{\mathbf{T}}_n$ and the QR decomposition of the symmetric matrix $\underline{\mathbf{T}}_n = \underline{\mathbf{Q}}_n\mathbf{R}_n^{\mathrm{MR}}$ with an $(n+1) \times n$ matrix $\underline{\mathbf{Q}}_n$ with orthonormal columns and an $n \times n$ upper triangular $\mathbf{R}_n^{\mathrm{MR}}$, we get

$$\begin{aligned}
\mathbf{Y}_n^{\mathsf{H}}\mathbf{A}^{\mathsf{H}}\mathbf{A}\mathbf{Y}_n &= \underline{\mathbf{T}}_n^{\mathsf{T}}\mathbf{Y}_{n+1}^{\mathsf{H}}\mathbf{Y}_{n+1}\underline{\mathbf{T}}_n \\
&= \underline{\mathbf{T}}_n^{\mathsf{T}}\underline{\mathbf{T}}_n = (\mathbf{R}_n^{\mathrm{MR}})^{\mathsf{T}}\underline{\mathbf{Q}}_n^{\mathsf{T}}\underline{\mathbf{Q}}_n\mathbf{R}_n^{\mathrm{MR}} \\
&= (\mathbf{R}_n^{\mathrm{MR}})^{\mathsf{T}}\mathbf{R}_n^{\mathrm{MR}}.
\end{aligned} \tag{2.41}$$

[6] The authors of a widely used collection of algorithms [1] claim that SymmLQ does not minimize anything. It is curious that one of the authors introduces SymmLQ in exactly the same way as proposed here in his monograph [34]. However, that is only a half of the truth in both cases.

$(\mathbf{R}_n^{\text{MR}})^{\mathsf{T}}\mathbf{R}_n^{\text{MR}}$ is just the Cholesky decomposition of the symmetric positive definite matrix $\underline{\mathbf{T}}_n^{\mathsf{T}}\underline{\mathbf{T}}_n$, here computed via the more stable QR decomposition of $\underline{\mathbf{T}}_n$. Altogether, the normal equations (2.39) are reduced to

$$(\mathbf{R}_n^{\text{MR}})^{\mathsf{T}}\,\mathbf{R}_n^{\text{MR}}\,\mathbf{k}_n^{\mathsf{L}} = \mathbf{e}_1\rho_0. \tag{2.42}$$

Setting $\mathbf{L}_n^{\text{MR}} :\equiv (\mathbf{R}_n^{\text{MR}})^{\mathsf{T}}$, inserting $\mathbf{k}_n^{\mathsf{L}}$ into (2.37), and using first $\mathbf{A}\mathbf{Y}_n = \mathbf{Y}_{n+1}\underline{\mathbf{T}}_n$ and then $\underline{\mathbf{T}}_n = \underline{\mathbf{Q}}_n\mathbf{R}_n^{\text{MR}}$, we obtain further

$$\begin{aligned}
\mathbf{x}_n^{\mathsf{L}} &= \mathbf{x}_0 + \mathbf{A}\mathbf{Y}_n(\mathbf{R}_n^{\text{MR}})^{-1}(\mathbf{L}_n^{\text{MR}})^{-1}\mathbf{e}_1\rho_0\\
&= \mathbf{x}_0 + \mathbf{Y}_{n+1}\underline{\mathbf{T}}_n(\mathbf{R}_n^{\text{MR}})^{-1}(\mathbf{L}_n^{\text{MR}})^{-1}\mathbf{e}_1\rho_0\\
&= \mathbf{x}_0 + \mathbf{Y}_{n+1}\underline{\mathbf{Q}}_n(\mathbf{L}_n^{\text{MR}})^{-1}\mathbf{e}_1\rho_0\,.
\end{aligned}$$

So, if we let

$$\mathbf{W}_n :\equiv \begin{pmatrix} \mathbf{w}_0 & \dots & \mathbf{w}_{n-1} \end{pmatrix} :\equiv \mathbf{Y}_{n+1}\underline{\mathbf{Q}}_n\,, \qquad \mathbf{g}_n :\equiv (\mathbf{L}_n^{\text{MR}})^{-1}\mathbf{e}_1\rho_0\,, \tag{2.43}$$

we finally get

$$\mathbf{x}_n^{\mathsf{L}} = \mathbf{x}_0 + \mathbf{W}_n\mathbf{g}_n = \mathbf{x}_{n-1}^{\mathsf{L}} + \mathbf{w}_{n-1}g_{n-1}. \tag{2.44}$$

Furthermore we introduce

$$\overline{\mathbf{W}}_{n+1} :\equiv \begin{pmatrix} \mathbf{w}_0 & \dots & \mathbf{w}_{n-1} & \overline{\mathbf{w}}_n \end{pmatrix} :\equiv \mathbf{Y}_{n+1}\mathbf{Q}_{n+1}$$

which implies

$$\mathbf{W}_n = \overline{\mathbf{W}}_{n+1}\begin{pmatrix} \mathbf{I}_n \\ \mathbf{0} \end{pmatrix}.$$

The matrix $\overline{\mathbf{W}}_n$ is easy to update by appending the last column of $\mathbf{Y}_{n+1}$ and applying the Givens transformation $\mathbf{G}_n$ to the last two columns, that is

$$\begin{pmatrix} \mathbf{w}_{n-1} & \overline{\mathbf{w}}_n \end{pmatrix} = \begin{pmatrix} \overline{\mathbf{w}}_{n-1} & \mathbf{y}_n \end{pmatrix}\mathbf{G}_n. \tag{2.45}$$

We could exhibit the relation $\mathbf{L}_n^{\text{MR}} = \left(\mathbf{R}_n^{\text{MR}}\right)^{\mathsf{T}}$ by writing the relation $\mathbf{L}_n^{\text{MR}}\mathbf{g}_n = \mathbf{e}_1\rho_0$ as $\mathbf{g}_n^{\mathsf{T}}\mathbf{R}_n^{\text{MR}} = \rho_0\mathbf{e}_1^{\mathsf{T}}$. The resulting three-term recursion for updating $\mathbf{g}_n$ is described in Chapter 6[7].

Paige and Saunders claim referring to a computation of a few lines that a slightly longer algebraic manipulation[8] shows that

$$\mathbf{r}_n^{\mathsf{L}} = \mathbf{y}_n\widetilde{\alpha}_n g_n - \mathbf{y}_{n+1}\widetilde{\gamma}_{n-1}g_{n-1}. \tag{2.46}$$

It is not possible to evaluate this term in the nth iteration. The update scheme limps two iterations. As $\mathbf{r}_n^{\mathsf{L}}$ is a linear combination of $\mathbf{y}_n$ and $\mathbf{y}_{n+1}$ it is obvious that $\mathbf{r}_n^{\mathsf{L}} \perp \mathcal{K}_n(\mathbf{A}, \mathbf{r}_0)$.

[7] There is described a more general case. Here the parameters are $N = 1, \mathbf{b}_n = \rho_0\mathbf{e}_1^{\mathsf{T}}, \mathbf{s}_n^{\square} = n, \mathbf{s}_i = 1$ and $\mathbf{P}_n = \mathbf{I}_n$.

[8] This manipulation is given in Chapter 8 for the general block case.

The second iteration in SymmLQ is the construction of the approximation given by (2.33), that is the residuals are orthogonal. Without almost any further costs it is possible to compute the approximation $\mathbf{x}_n$ and the corresponding residual $\mathbf{r}_n$. It is

$$\underline{\mathbf{T}}_n^{\mathsf{T}}\underline{\mathbf{T}}_n\mathbf{k}_n^{\mathsf{L}} = \mathbf{T}_n\mathbf{k}_n = \mathbf{e}_1\rho_0 \tag{2.47}$$

Using the Cholesky decomposition of $\underline{\mathbf{T}}_n^{\mathsf{T}}\underline{\mathbf{T}}_n$ (2.41), the identity (2.34) and replacing $\mathbf{k}_n^{\mathsf{L}}$ by $(\mathbf{R}_n^{\mathrm{MR}})^{-1}\mathbf{g}_n$ yields

$$\mathbf{g}_n = \begin{pmatrix} \mathbf{I}_{n-1} & \\ & c_n \end{pmatrix} \mathbf{Q}_n^{\mathsf{T}}\mathbf{k}_n$$

If $c_n \neq 0$ the update scheme for $\mathbf{x}_n$ is

$$\begin{aligned} \mathbf{x}_n &= \mathbf{x}_0 + \mathbf{Y}_n\mathbf{Q}_n\mathbf{Q}_n^{\mathsf{T}}\mathbf{k}_n \\ &= \mathbf{x}_0 + \overline{\mathbf{W}}_n \begin{pmatrix} \mathbf{I}_{n-1} & \\ & c_n^{-1} \end{pmatrix} \mathbf{g}_n \\ &= \mathbf{x}_0 + \mathbf{W}_{n-1}\mathbf{g}_{n-1} + \overline{\mathbf{w}}_{n-1}c_n^{-1}g_{n-1} \\ &= \mathbf{x}_{n-1}^{\mathsf{L}} + \overline{\mathbf{w}}_{n-1}c_n^{-1}g_{n-1}. \end{aligned} \tag{2.48}$$

If $c_n = 0$ we proceed without updating $\mathbf{x}_n$ and $\mathbf{r}_n$. If the matrix $\mathbf{A}$ is Hermitian positive definite the iterates $\mathbf{x}_n$ are equivalent to those gained in the method of conjugate gradients.

An update scheme for the corresponding residual is given by

$$\mathbf{r}_n = \mathbf{y}_n\beta_n \left(s_{n-1}g^{(n-2)} + \frac{c_{n-1}}{c_n}g^{(n-1)} \right). \tag{2.49}$$

It is possible to evaluate this term in the nth iteration if $c_n \neq 0$. As the vectors $\mathbf{y}_n$ are orthogonal the residuals are orthogonal as postulated for Ritz-Galerkin methods (2.33).

Algorithm 3 (SymmLQ) .
For solving $\mathbf{Ax} = \mathbf{b}$ *with Hermitian* $\mathbf{A}$ *choose* $\mathbf{x}_0$*, and let* $\mathbf{r}_0 := \mathbf{b} - \mathbf{Ax}_0$*,* $\rho_0 := \|\mathbf{r}_0\|$*,* $\overline{\mathbf{w}}_0 := \mathbf{y}_0 := \mathbf{r}_0/\rho_0$*, and* $g_{-2} := g_{-1} := \widetilde{\gamma}_{-1} := 0$*.*
Then, for $n = 1, \ldots, m$*:*

1. *Do one step of the symmetric Lanczos algorithm:*

$$\begin{aligned} &\widetilde{\mathbf{y}}_n := \mathbf{A}\mathbf{y}_{n-1}\,, && \widetilde{\mathbf{y}}_n := \widetilde{\mathbf{y}}_n - \mathbf{y}_{n-2}\beta_{n-2} \quad \textit{if } n > 1, \\ &\alpha_{n-1} := \langle \mathbf{y}_{n-1}, \widetilde{\mathbf{y}}_n \rangle\,, && \widetilde{\mathbf{y}}_n := \widetilde{\mathbf{y}}_n - \mathbf{y}_{n-1}\alpha_{n-1}\,, \\ &\beta_{n-1} := \|\widetilde{\mathbf{y}}_n\|\,, && \mathbf{y}_n := \widetilde{\mathbf{y}}_n/\beta_{n-1}\,. \end{aligned}$$

2. *Let* $\widetilde{\alpha}_{n-1} := \alpha_{n-1}$*, and, if* $n > 1$, $\widetilde{\beta}_{n-1} := \beta_{n-1}$.
   If $n > 2$*, apply* $\mathbf{G}_{n-2}^{\mathsf{H}}$ *to the new last column of* $\underline{\mathbf{T}}_n$*:*

$$\begin{pmatrix} \widetilde{\gamma}_{n-3} \\ \widetilde{\beta}_{n-2} \end{pmatrix} := \begin{pmatrix} c_{n-2} & s_{n-2} \\ -\overline{s_{n-2}} & c_{n-2} \end{pmatrix} \begin{pmatrix} 0 \\ \widetilde{\beta}_{n-2} \end{pmatrix};$$

   if $n > 1$*, apply* $\mathbf{G}_{n-1}^{\mathsf{H}}$ *to the last column of* $\mathbf{G}_{n-2}^{\mathsf{H}}\underline{\mathbf{T}}_n$*:*

$$\begin{pmatrix} \widetilde{\beta}_{n-2} \\ \widetilde{\alpha}_{n-1} \end{pmatrix} := \begin{pmatrix} c_{n-1} & s_{n-1} \\ -\overline{s_{n-1}} & c_{n-1} \end{pmatrix} \begin{pmatrix} \widetilde{\beta}_{n-2} \\ \widetilde{\alpha}_{n-1} \end{pmatrix}.$$

3. *Let* $\mu_n := \widetilde{\alpha}_{n-1}$, $\nu_n := \beta_{n-1}$ *and compute* c_n *and* s_n *of the Givens rotation* $\mathbf{G}_n$ *according to* (2.25).

4. *Apply the adjoint Givens rotation* $\mathbf{G}_n^{\mathsf{H}}$ *to update the last two components of the modified last column of* $\underline{\mathbf{T}}_n$ *(which turns into* $\mathbf{R}_n^{\mathrm{MR}}$*), then compute the last component* g_{n-1} *of* $\mathbf{g}_n$ *by the three-term recurrence induced by* $\mathbf{L}_n^{\mathrm{MR}} = (\mathbf{R}_n^{\mathrm{MR}})^{\mathsf{H}}$.

5. *Update the direction vectors* $\mathbf{w}_{n-1}$ *and* $\overline{\mathbf{w}}_n$ *according to* (2.45).

6. *Update the approximations* $\mathbf{x}_n^{\mathsf{L}}$, $\mathbf{x}_n$ *and the corresponding residuals* $\mathbf{r}_{n-2}^{\mathsf{L}}$, $\mathbf{r}_n$ *according to* (2.44),(2.48) *and* (2.46),(2.49).

7. *If* $\|\mathbf{r}_{n-2}^{\mathsf{L}}\|_2 \leq \mathrm{tol}$ *or* $\|\mathbf{r}_n\|_2 \leq \mathrm{tol}$*, the algorithm terminates and* $\mathbf{x}_{n-2}^{\mathsf{L}}$ *or* $\mathbf{x}_n$ *are a sufficiently accurate approximate solutions. In practice:* 2*nd verification by computing explicitly the corresponding residual.*

Note that the first three steps and part of the forth are the same as in MinRes. There are various other possibilities for the last two steps. For example it might be an idea to update $\mathbf{x}_n$ only if $\mathbf{r}_{n-2}^{\mathsf{L}}$ is sufficiently small.

In this chapter we introduced with MinRes and SymmLQ two methods computing three sequences of approximations for the solution of a Hermitian linear system. The sequence in MinRes gives minimal residuals. The constructed iterates in SymmLQ guarantee minimal errors or orthogonal residuals. However, all methods are based on the same termination property. If the residual is smaller than a certain tolerance the methods stops. It seems therefore reasonable to argue that MinRes is the best choice as the variational property matches the termination criterion in a perfect way.

However, things are much more complicated. It has been shown that rounding errors are propagated to the approximate solution with a factor proportional to the square of the condition of $\mathbf{A}$ in MinRes; in SymmLQ this factor is only proportional to the condition of $\mathbf{A}$ [29]. So SymmLQ is slower but more reliable than MinRes.

But Krylov methods tend to fail or converge very slowly if the condition is too large. A remedy, an art and a science is preconditioning introduced in the next chapter.

There are much more links between the methods we have introduced here. It is possible to construct the iterates in MinRes out of the iterates gained in SymmLQ. We refer the reader to an excellent book by Fischer [5].

3. Preconditioning

> The story of these methods very briefly: They flopped as direct methods and were brought back to life as effective iterative procedures only when coupled with some form of preconditioning.
>
> *(Beresford Parlett)*

This is only a very brief discussion of the concept of preconditioning for Hermitian indefinite systems. It is unsatisfying that there is still no reliable preconditioner for such systems.

3.1. Spectrum and convergence

The equation (2.1) implies a surjective mapping between the linear space of polynomials of degree less than $n-1$ and a Krylov subspace $\mathcal{K}_n(\mathbf{A}, \mathbf{r}_0)$, that is, there is a polynomial q_n of degree less than $n-1$ such that

$$\mathbf{x}_n = \mathbf{x}_0 + q_n(\mathbf{A})\mathbf{r}_0.$$

We remark that

$$\mathbf{r}_n = (\mathbf{I} - \mathbf{A}q_n(\mathbf{A}))\,\mathbf{r}_0 = p_n(\mathbf{A})\mathbf{r}_0 \tag{3.1}$$

where p_n is the polynomial mapping z to $1 - zq_n(z)$, in particular $p_n(0) = 1$[1]. Let P_n be the set of polynomials p of degree less than n and $p(0) = 1$. Then the variational principle of MinRes is[2]

$$\|\mathbf{r}_n\|_2 = \min_{p_n \in P_n} \|p_n(\mathbf{A})\mathbf{r}_0\|_2. \tag{3.2}$$

For Hermitian matrices $\mathbf{A}$ a first upper bound for the relative residual is given by

$$\frac{\|\mathbf{r}_n\|_2}{\|\mathbf{r}_0\|_2} \leq \min_{p \in P_n} \max_{\lambda \in \Lambda(\mathbf{A})} |p(\lambda)|. \tag{3.3}$$

If the eigenvalues of $\mathbf{A}$ are all contained in the interval $E = [c, d] \subset \mathbb{R}^+$ a coarser upper bound is induced by

$$\rho_n := \min_{p \in P_n} \max_{\lambda \in E} |p(\lambda)|$$

[1] There is a similar mapping for SymmLQ . Here $\mathbf{f}_n^{\mathsf{L}} = \left(\mathbf{I} - \mathbf{A}^2 q_n(\mathbf{A})\right)\mathbf{f}_0^{\mathsf{L}}$.

[2] The variational principle of SymmLQ is $\|\mathbf{f}_n^{\mathsf{L}}\|_2 = \min_{p_n \in P_n} \|p_n(\mathbf{A})\mathbf{f}_0^{\mathsf{L}}\|_2$ where P_n is the set of polynomials p of degree less than $n+1$, $p(0) = 1$ and $p'(0) = 0$.

which is a classical approximation problem with an explicit solution given by a suitably scaled and shifted Chebyshev polynomial. An analysis is given in [30]. Unfortunately things are much more complicated if the matrix $\mathbf{A}$ is indefinite, that is, the spectrum is contained in two intervals

$$E = [-a, -b] \cup [c, d]$$

where a, b, c and d are positive numbers. However, if both intervals are of equal length an explicit solution exists. Due to Lebedev [17] the unique polynomial minimizing ρ_n is given in terms of Chebyshev polynomials by

$$T_{\lfloor \frac{n}{2} \rfloor}(q(x)) \Big/ T_{\lfloor \frac{n}{2} \rfloor}(q(0)), \qquad q(x) = 1 + \frac{2(x+b)(x-c)}{bc - ad}$$

where $\lfloor \frac{n}{2} \rfloor$ is the biggest integer i such that $i \leq \frac{n}{2}$. The nth Chebyshev polynomial is

$$T_n(x) = \cos(n \arccos x).$$

An interpretation as a sine wave "wrapped around a cylinder and viewed from the side" and various other properties are given in [32]. We end up with

$$\rho_n = 2 \left(\frac{\sqrt{ad} - \sqrt{bc}}{\sqrt{ad} + \sqrt{bc}} \right)^{\lfloor \frac{n}{2} \rfloor}. \tag{3.4}$$

Assuming symmetry, i.e. $a = d$ and $b = c$, Wathen [35] noted that this residual reduction is the same as would be achieved after $\lfloor \frac{k}{2} \rfloor$ steps on a positive definite problem with eigenvalues in $[c^2, d^2]$, in particular solving the normal equations with the method of conjugate gradients would require a comparable amount of work as there are two matrix-vector products at each iteration.

Obviously the convergence is influenced by the position and number of eigenvalues on the real line. In the case of nonnormal[3] matrices minor perturbations can shift the spectrum in the complex plane a lot. This behavior is reflected in the pseudospectrum of a matrix [19]. Fortunately the eigenvalues of an Hermitian matrix are stable[4].

3.2. Left preconditioning for Hermitian systems

In order to stay consistent with the notation that a linear system $\mathbf{Ax} = \mathbf{b}$ is solved we denote the original system by $\widehat{\mathbf{A}}\widehat{\mathbf{x}} = \widehat{\mathbf{b}}$. The idea of preconditioning is to solve an equivalent system with a spectrum more suitable for the aforementioned methods. In practice Krylov methods often converge very slowly when applied to large real-world problems - if at all. Therefore they are nearly always applied with preconditioning. The simplest technique, called left preconditioning, is to multiply both sides of the

[3] A matrix is called normal if it is unitary diagonalizable, i.e. all Hermitian matrices are normal, but of course not all normal matrices are Hermitian

[4] The pseudospectrum of a Hermitian matrix is trivial, which is a consequence of the Bauer-Fike theorem

linear system with a matrix $\mathbf{C}$ or $\mathbf{C}^{-1}$, where $\mathbf{C}$ is given. In general $\mathbf{C}\widehat{\mathbf{A}}$ or $\mathbf{C}^{-1}\widehat{\mathbf{A}}$ is not Hermitian, even if $\mathbf{C}$ is so. We demand $\mathbf{C}$ is Hermitian positive definite, in particular $\mathbf{C}\widehat{\mathbf{A}}$ is Hermitian with respect to the inner product induced by $\mathbf{C}^{-1}$. So the Euclidean inner product is replaced in this approach. An alternative is to use the Cholesky decomposition of $\mathbf{C} = \mathbf{L}\mathbf{L}^{\mathsf{H}}$, that is to solve the system corresponding to $\mathbf{C}\widehat{\mathbf{A}}$

$$\mathbf{L}^{\mathsf{H}}\widehat{\mathbf{A}}\mathbf{L}\mathbf{L}^{-1}\widehat{\mathbf{x}} = \mathbf{L}^{\mathsf{H}}\widehat{\mathbf{b}} \tag{3.5}$$

or to solve the system corresponding to $\mathbf{C}^{-1}\widehat{\mathbf{A}}$

$$\mathbf{L}^{-1}\widehat{\mathbf{A}}\mathbf{L}^{-\mathsf{H}}\mathbf{L}^{\mathsf{H}}\widehat{\mathbf{x}} = \mathbf{L}^{-1}\widehat{\mathbf{b}}. \tag{3.6}$$

Both operators $\mathbf{L}^{\mathsf{H}}\widehat{\mathbf{A}}\mathbf{L}$ and $\mathbf{L}^{-1}\widehat{\mathbf{A}}\mathbf{L}^{-\mathsf{H}}$ are Hermitian and denoted by $\mathbf{A}$. Similar $\mathbf{b} = \mathbf{L}^{\mathsf{H}}\widehat{\mathbf{b}}$ or respectively $\mathbf{b} = \mathbf{L}^{-1}\widehat{\mathbf{b}}$ and $\mathbf{x} = \mathbf{L}^{-1}\widehat{\mathbf{x}}$ or $\mathbf{x} = \mathbf{L}^{-\mathsf{H}}\widehat{\mathbf{x}}$. The matrix $\mathbf{A}$ is never formed explicitly. Obviously the construction of $\mathbf{C}$ and its Cholesky decomposition should not be prohibitive. It is important that solving linear systems as $\mathbf{v} = \mathbf{L}\mathbf{w}$ is fast and effective. A good preconditioner for indefinite systems would shift the spectrum, such that the term $\sqrt{ad} - \sqrt{bc}/\sqrt{ad} + \sqrt{bc}$ in (3.4) is much smaller or would reduce the number of distinct eigenvalues dramatically as the relative residual vanishes after k steps, where k is the number of distinct eigenvalues. This is an active area of research.

3.2.1. The algebraic approach

The black box constructing an effective preconditioner $\mathbf{C}$ where $\widehat{\mathbf{A}}$ is the only input parameter is still not available. Although for some classes of matrices $\widehat{\mathbf{A}}$ there are methods for constructing suitable preconditioners, but unfortunately not for Hermitian indefinite matrices. Due to Saad [26, Page 339] the conjugate gradients approach applied to the normal equations may become a good alternative if the matrix $\widehat{\mathbf{A}}$ is strongly indefinite, i.e. when it has eigenvalues on both sides of the imaginary axis. Benzi and Tuma [2] conclude a survey with the result that the reliability of existing methods is still elusive.

3.2.2. The problem approach

Here the structure of the matrix $\widehat{\mathbf{A}}$ and background of the problem is exploited. It is possible to construct effective preconditioners for a certain class of Hermitian indefinite systems. Although problems arise in computational, fluid dynamics, optimization and many other fields of scientific computing they often share a common matrix structure. A lot of research has been done for problems of the form:

$$\begin{pmatrix} \mathbf{H} & \mathbf{J}^{\mathsf{H}} \\ \mathbf{J} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{pmatrix} = \begin{pmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \end{pmatrix} \tag{3.7}$$

They arise in computational fluid dynamics, optimization and many other fields of scientific computing.

3.3. Preconditioned iterations

The updated residual respectively the updated norm of the residual in MinRes and SymmLQ corresponds to the preconditioned residual

$$\mathbf{r}_n = \mathbf{b} - \mathbf{A}\mathbf{x}_n. \tag{3.8}$$

However if preconditioning is applied the stopping criterion should usually still be based on the original residuals instead of the preconditioned ones. The original residual is updated[5] by

$$\widehat{\mathbf{r}}_n = \mathbf{L}^{-\mathsf{H}}\mathbf{r}_n \tag{3.9}$$

if $\mathbf{A} = \mathbf{L}^{\mathsf{H}}\widehat{\mathbf{A}}\mathbf{L}$ or respectively

$$\widehat{\mathbf{r}}_n = \mathbf{L}\mathbf{r}_n \tag{3.10}$$

if $\mathbf{A} = \mathbf{L}^{-1}\widehat{\mathbf{A}}\mathbf{L}^{-\mathsf{H}}$.

The desired residual is computed explicitly for a 2nd verification if the corresponding norm predicted by (3.8), (3.9) or (3.10) is smaller than an a priori given tolerance.

Algorithm 4 (Iterative solver with preconditioning) .
Let a Hermitian matrix $\mathbf{A}$*, a Hermitian positive definite preconditioner* $\mathbf{C}$*, a right-hand side* $\widehat{\mathbf{b}}$ *and an initial approximation* $\widehat{\mathbf{x}}_0$ *be given.*

1. *Compute the Cholesky decomposition of* $\mathbf{C}$ *and initialize the preconditioned iteration due to* (3.5) *(or* (3.6)*).*

$$\begin{aligned}\mathbf{C} &= \mathbf{L}\mathbf{L}^{\mathsf{H}}\\ \mathbf{A} &= \mathbf{L}^{-1}\widehat{\mathbf{A}}\mathbf{L}^{-\mathsf{H}}\\ \mathbf{x}_0 &= \mathbf{L}^{\mathsf{H}}\widehat{\mathbf{x}}_0\\ \mathbf{b} &= \mathbf{L}^{-1}\widehat{\mathbf{b}}\\ \mathbf{r}_0 &= \mathbf{b} - \mathbf{A}\mathbf{x}_0.\end{aligned}$$

2. *Execute an iteration of a linear solver as* MinRes *or* SymmLQ*. Update the preconditioned residual* $\mathbf{r}_n$ *by using equation* (2.30)*,* (2.46) *or* (2.49)*.*
3. *Update the original residual by* (3.9) *or* (3.10) *and compute its norm. If this norm is smaller than the a priori given tolerance an explicit computation is used for a second verification, that is* $\widehat{\mathbf{r}}_n = \widehat{\mathbf{b}} - \widehat{\mathbf{A}}\mathbf{L}^{-\mathsf{H}}\mathbf{x}_n$.

[5] The current implementation of MinRes in Matlab does not exploit this possibility. In each iteration the unpreconditioned residual is explicitly computed by an expensive matrix-vector multiplication.

Note that it is without efforts to generalize this algorithm for the block case, i.e. $\mathbf{b} \in \mathbb{C}^{N\times \mathsf{s}}$. The problems are hidden in the linear solvers and the corresponding update process of the preconditioned residual. The speed of convergence might vary for different right-hand sides. If $\|\widehat{\mathbf{r}}_n^{(i)}\|_2$ is smaller than an a priori given tolerance the ith system is solved and can be deflated.

4. A block Lanczos process

Why did more than 20 years pass before the properties of the Lanczos algorithm were understood? My suggestion is that the difficulties in matrix computations may not be **deep** but they are **subtle**.

(Beresford Parlett)

In block methods a common space for approximation is used. In this work that is a direct sum of Krylov spaces:

$$\mathcal{B}_n\left(\mathbf{A}, \mathbf{r}_0\right) = \bigoplus_{i=1}^{\mathsf{s}} \mathcal{K}_n\left(\mathbf{A}, \mathbf{r}_0^{(i)}\right) = \mathsf{span}\left\{\mathbf{r}_0, \mathbf{A}\mathbf{r}_0, \ldots, \mathbf{A}^{n-1}\mathbf{r}_0\right\}. \tag{4.1}$$

The scope of a block Lanczos process is to construct an orthonormal basis for this space. A deflation scheme detecting and deleting linearly dependent or almost linearly dependent vectors in the list $\mathbf{r}_0, \mathbf{A}\mathbf{r}_0, \ldots, \mathbf{A}^{n-1}\mathbf{r}_0$ is introduced. We perform a series of experiments in order to investigate and understand the influence of the initial starting vectors and the loss of orthogonality.

4.1. The loss of orthogonality

In the case of exact arithmetic the Lanczos process (Algorithm 1) creates an $\mathbf{A}$-invariant subspace. Once the vector $\mathbf{A}\mathbf{y}_{n-1}$ is contained in the span of the vectors $\mathbf{y}_{n-2}$ and $\mathbf{y}_{n-1}$ we end up with

$$\mathbf{A}\mathbf{Y}_n = \mathbf{Y}_n\mathbf{T}_n.$$

where $\mathbf{Y}_n$ is an orthogonal matrix.

However, the vectors constructed by this method tend to lose orthogonality. A problem responsible for the fact that the Lanczos algorithm was regarded for almost 20 years as a method with a certain theoretical beauty but without almost any impact in practice.

Experiment 1 *Let* $\mathbf{A}$ *a* 100×100 *discrete Laplacian*[1] *on an uniform grid in a square. The initial vector* $\mathbf{y}_0$ *has random entries uniformly distributed in interval* $[0, 1]$*. In the notation of* Matlab*:*

[1] Details are given in the Appendix

$n = 10;\ N = n^2;$
$A = gallery('poisson', n);$
$y0 = rand(N, 1);$

The Lanczos process does not stop. The algorithm is producing a set of local orthogonal vectors far away from being linear independent or even orthogonal. The subdiagonal entries do not indicate any loss of orthogonality. Nevertheless MinRes *does converge. In Figure 4.14 superlinear convergence [34, Page 50–57] is observable.*

The Arnoldi process with double projection constructs a sequence of almost perfect orthogonal vectors. Even the small subdiagonal entries in the iterations between 90 *and* 100 *do not destroy orthogonality. This effect is explained in one of the next experiments.*

Experiment 2 *Here we work with a sparse* 100×100 *random matrix and an initial vector as described in experiment 1.*

$d = 0.1;$
$N = 100;$
$A = sprand(N, N, d) + sqrt(-1) * sprand(N, N, d);$
$A = 0.5 * (A' + A);$

The results are rather similar to those gained in the experiment 1. It seems there is a beautiful pattern in the structure of the matrix $\mathbf{V} = \log|\mathbf{Y}_{250}^{\mathsf{H}}\mathbf{Y}_{250} - \mathbf{I}_{250}|$.

Figure 4.15 shows the typical convergence behavior for such random matrices. Krylov solvers are not effective for this class of matrices. Edelman [4] states that it is a mistake to link psychologically random matrices with the intuitive notion of a "typical" matrix or the vague concept of "any old matrix".

The result is different is we use small matrices or work with special start vectors $\mathbf{y}_0$.

Experiment 3 *The matrix* $\mathbf{A}$ *is a* 50×50 *sparse random matrix. Here the start vector* $\mathbf{y}_0$ *is a random linear combination of* 15 *eigenvectors of* $\mathbf{A}$*, hence this* 15 *vectors are an orthonormal basis for an* $\mathbf{A}$*-invariant subspace.*

$k = 15;$
$[V, D] = eig(A);$
$p = randperm(N);$
$y0 = V(:, p(1:k)) * rand(k, 1);$

In this case the results are closer to the predictions from theory. The Lanczos algorithm constructs the Krylov basis vectors $\mathbf{y}_0, \ldots, \mathbf{y}_{14}$ *with rather accurate precision. They correspond to the upper left block in Figure 4.7. The algorithm should stop as we expect* $\beta_{14} = 0$. *We proceed although in this experiment* $\beta_{14} \approx 10^{-7}$. *The vector* $\mathbf{y}_{15}$ *corresponds to the discontinuous step in Figure 4.7.*

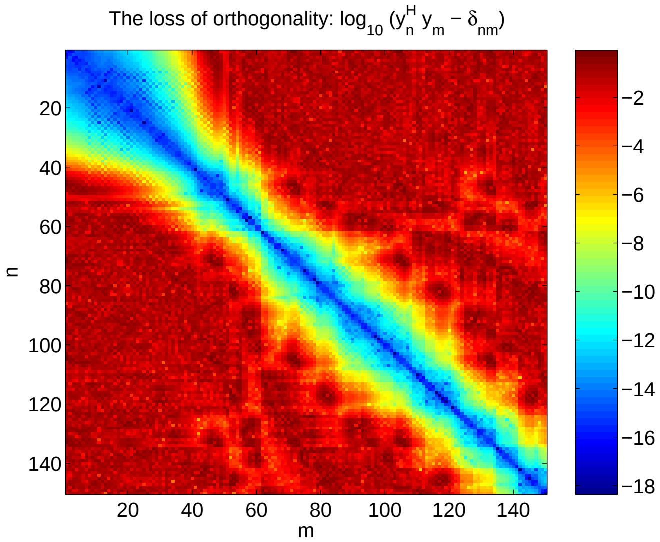


Figure 4.1.: Experiment 1: The loss of orthogonality in the Krylov basis using the Lanczos process.

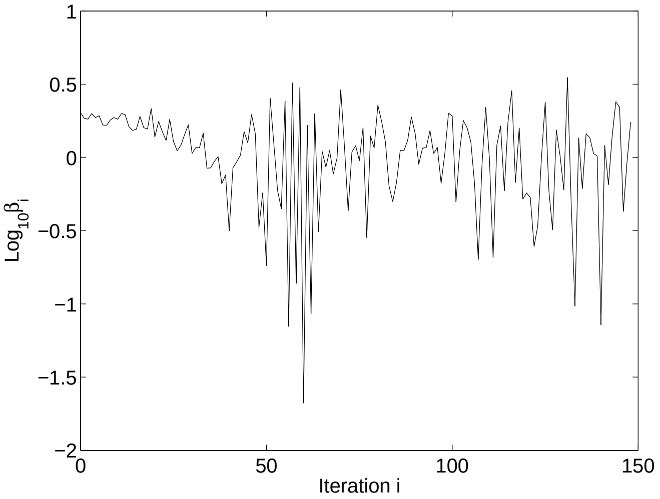


Figure 4.2.: Experiment 1: The subdiagonal entries are larger than 10^{-2}.

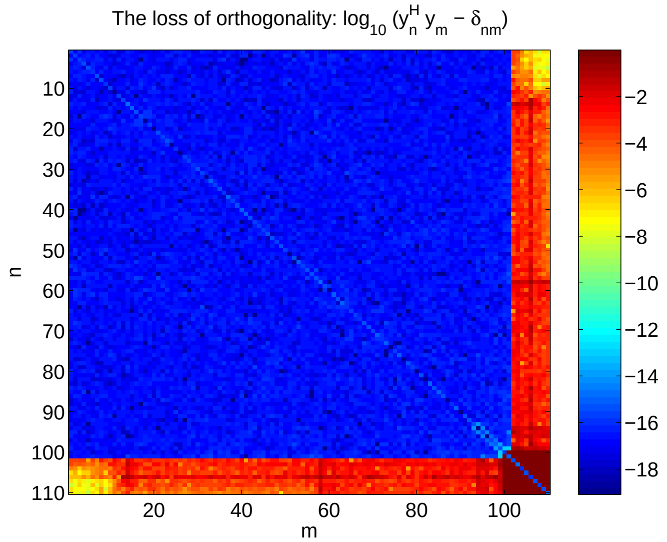


Figure 4.3.: Experiment 1: The loss of orthogonality in the Krylov basis using the Arnoldi process with double projection.

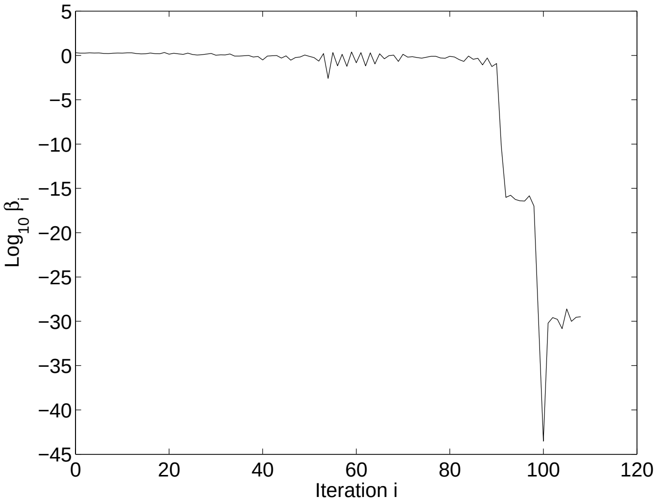


Figure 4.4.: Experiment 1: Extreme small subdiagonal entries of the corresponding upper Hessenberg matrix.

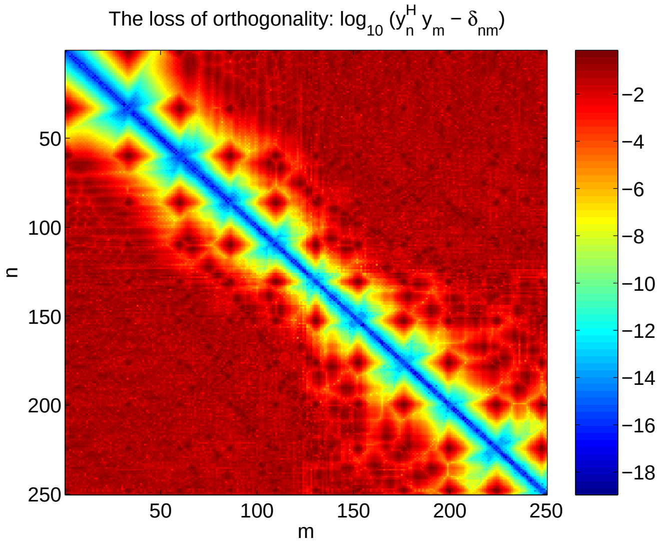


Figure 4.5.: Experiment 2: The loss of orthogonality in the Krylov basis using the Lanczos Process.

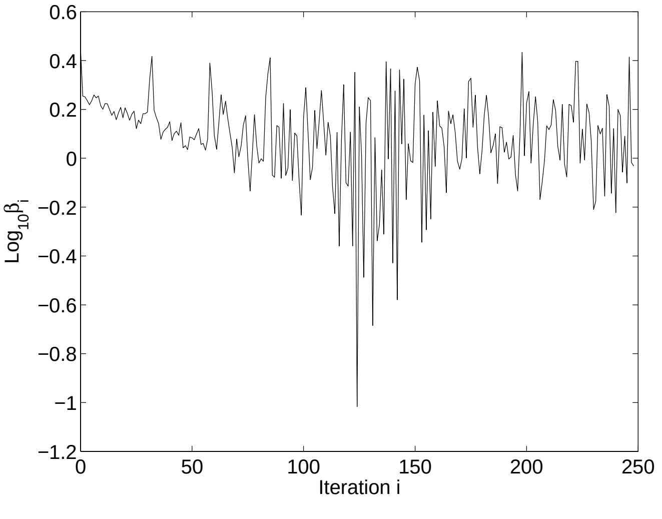


Figure 4.6.: Experiment 2: The subdiagonal entries are larger than 0.1.

The small subdiagonal entry indicates that the Krylov space $\mathcal{K}_{15}(\mathbf{A}, \mathbf{y}_0)$ *has been successfully generated. The vector* $\mathbf{y}_{15}$ *is not orthogonal to all previous constructed vectors. It is*

$$\mathbf{A}\mathbf{y}_{14} - \mathbf{y}_{13}\beta_{13} - \mathbf{y}_{14}\alpha_{14} = \widetilde{\mathbf{y}}_{15}.$$

Assuming a machine precision of 10^{-16} *and assuming* $\beta_{14} = 10^{-7}$ *the vector* $\widetilde{\mathbf{y}}_{15}$ *has only* 9 *remaining significant figures due to cancellation effects by subtracting two nearly equal vectors. Hence the vector* $\mathbf{y}_{15}$ *has only* 9 *significant digits. This explains why* $\mathbf{y}_{15}$ *is not orthogonal with respect to* $\mathbf{y}_{13}$ *and* $\mathbf{y}_{14}$ *although it should be at least perfect orthogonal with respect to* $\mathbf{y}_{14}$*. For the vectors* $\mathbf{y}_0, \ldots, \mathbf{y}_{12}$ *the situation is even worse.*

Using the Arnoldi process we have the same effect concerning the dip in the subdiagonal entries. However, there is no loss of orthogonality. The exhausted Krylov space is augmented in an almost perfect way. The almost vanishing projected vector is orthogonalized with respect to all previous vectors. The successive projections guarantee orthogonality in theory, but in practice we observe that after a few projections the vector is no longer orthogonal to those vectors used for constructing the first projections. Repeating the projections yields an impressive accuracy. See Figure 4.9 and 4.10.

The approximation gained in the exhausted Krylov space is of rather good quality. If we proceed superlinear convergence is lost but the residual can be reduced even more. See Figure 4.16.

Experiment 4 *The matrix* $\mathbf{A}$ *is a* 100×100 *discrete Laplacian as in experiment 1. Here the start vector* $\mathbf{y}_0$ *is a random linear combination of* 20 *eigenvectors of* $\mathbf{A}$*, hence this* 20 *vectors are an orthonormal basis for an* $\mathbf{A}$*-invariant subspace.*

```
$k = 20;$
$[V, D] = eig(A);$
$p = randperm(N);$
$y0 = V(:, p(1:k)) * rand(k, 1);$
```

However, here we note that the Krylov space is exhausted already after 15 *iterations. Hence the vector* $\mathbf{y}_{15}$ *is indetermined. The effect is explained by fact that* $\mathbf{A}$ *has degenerated eigenvalues. The eigenvalues*[2] *of* $\mathbf{A}$ *are*

$$\lambda_{i,j} = 2\,(2 - \cos i\pi h - \cos j\pi h) \quad i, j = 1, \ldots, n \tag{4.2}$$

where $h = (n+1)^{-1}$ *is the meshwidth of grid in the unit square. An eigenvalue is degenerated if it has at least multiplicity* 2*. If* $i + j = n + 1$ *then* $\lambda_{i,j} = 4$*, i.e. this eigenvalue has even multiplicity* n*. In Figure 4.11 the corresponding eigenvalues of the eigenvectors in the linear combination.*

$$\mathbf{y}_0 = \sum_{j=1}^{20} \gamma_j \mathbf{w}_j$$

[2] A proof is given in the appendix

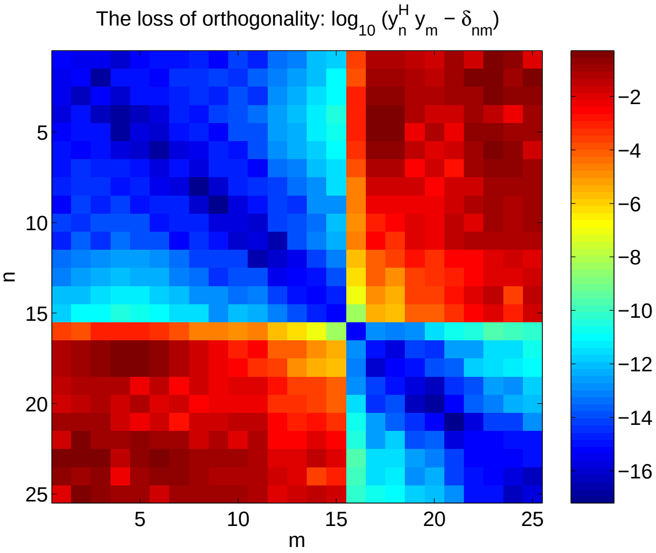


Figure 4.7.: Experiment 3: The loss of orthogonality in the Krylov basis using the Lanczos process.

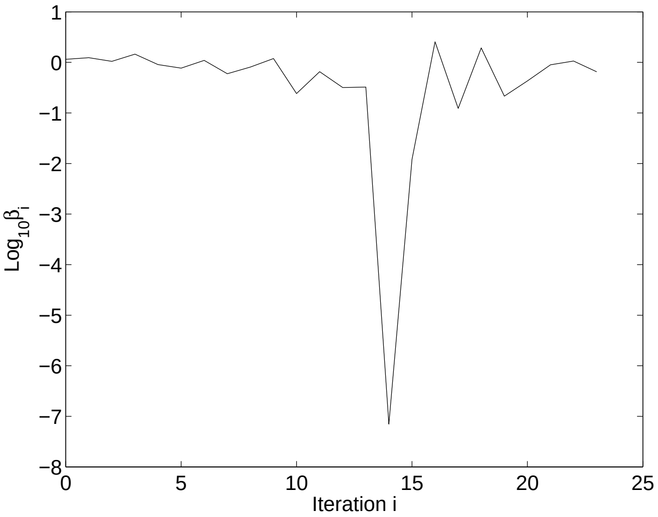


Figure 4.8.: Experiment 3: A dip in the subdiagonal entries indicates here an exhausted Krylov space.

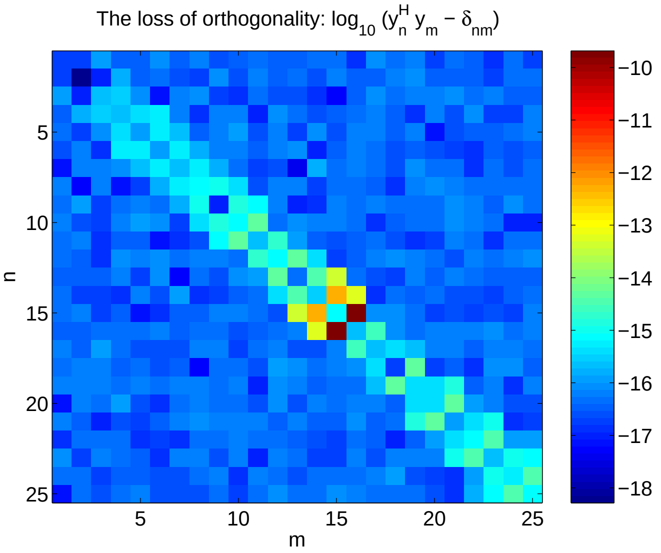


Figure 4.9.: Experiment 3: The loss of orthogonality in the Krylov basis using the Arnoldi process with simple projection.

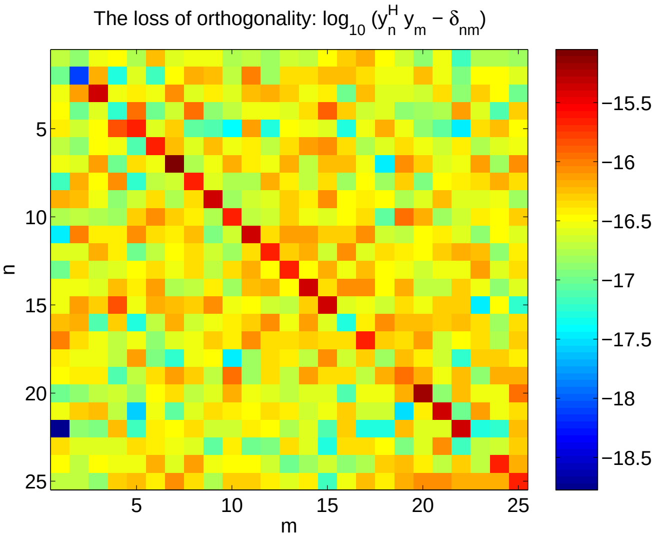


Figure 4.10.: Experiment 3: The loss of orthogonality in the Krylov basis using the Arnoldi process with double projection. The dark diagonal results from cancellation effects by subtracting two nearly equal numbers.

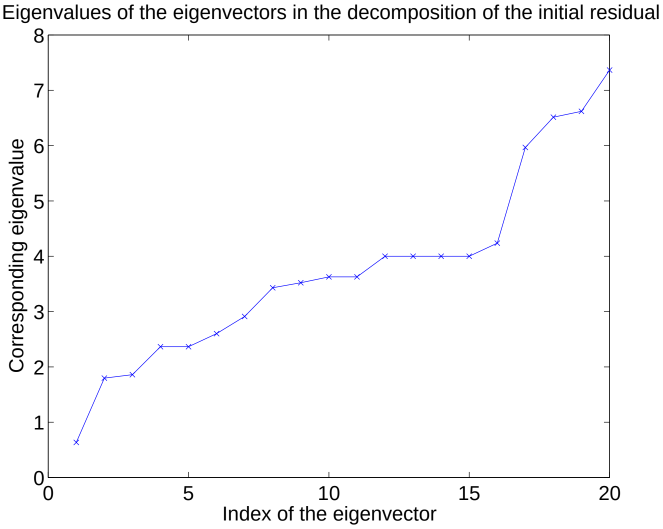


Figure 4.11.: Experiment 4: Corresponding eigenvalues for the eigenvectors in the decomposition of $\mathbf{y}_0$.

are marked. However, as the eigenvectors $\mathbf{w}_{12}, \ldots, \mathbf{w}_{15}$ *share the same eigenvalue also any linear combination of those vectors is an eigenvector. Hence we can reduce the number of eigenvectors in the above decomposition to* 17*. As there are two further horizontal lines in the graph we can reduce the number of eigenvectors to* 15*.*

All other effects are already described in experiment 3.

The approximation gained in the exhausted Krylov space is very accurate. There is no need to proceed with the indetermined vectors. See Figure 4.17.

It is worth noting that there are two reasons for the loss of orthogonality in the Lanczos algorithm. First, the converged Ritz vectors act as magnetic poles. Paige [20] showed that at the same time as orthogonality is lost, a Ritz pair converges to an eigenpair of $\mathbf{A}$. The Ritz vectors are $\mathbf{Y}_n\mathbf{z}_i$, where $\mathbf{z}_i$ are the eigenvectors of $\mathbf{T}_n$.

Although it is possible to avoid this with reorthogonalization is usually not combined with solvers for linear systems. Reorthogonalization is used when Lanczos is applied to compute the eigenpairs of a matrix $\mathbf{A}$. It has been proved by Greenbaum and Strakos [11] that rounding errors in the Lanczos process may have a delaying effect on the convergence of iterative solvers but do not prevent eventual convergence in general.

A second reason for the loss of orthogonality is ignorance of small subdiagonal entries in $\underline{\mathbf{T}}_n$. A small subdiagonal entry indicates that the Krylov space is exhausted. The corresponding basis vector $\mathbf{y}_i$ is indetermined. This effect appears if the grade $\bar{\nu}\left(\mathbf{A}, \mathbf{y}_0\right)$ of $\mathbf{A}$ with respect to $\mathbf{y}_0$ is small, hence the matrix is small or $\mathbf{y}_0$ is lying in a small eigenspace of $\mathbf{A}$. Actually in this case all possible Ritz pairs have converged to those

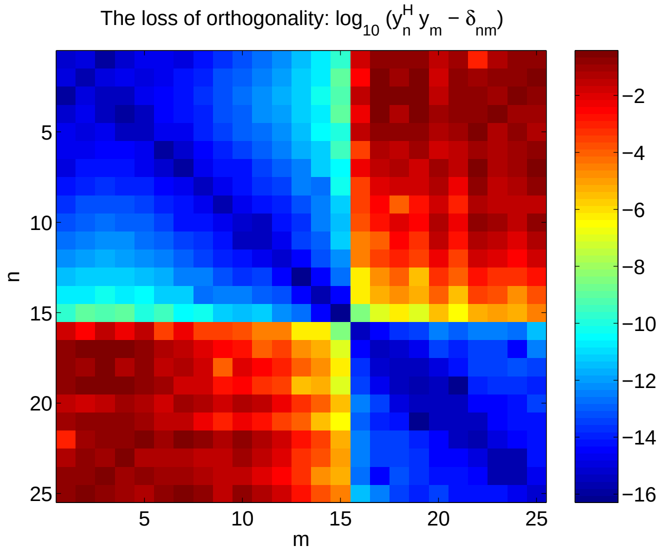


Figure 4.12.: Experiment 4: The loss of orthogonality in the Krylov basis using the Lanczos process.

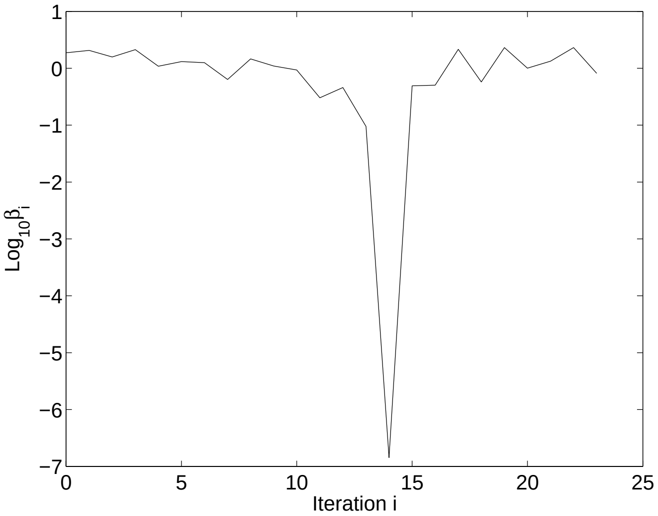


Figure 4.13.: Experiment 4: A dip in the subdiagonal entries indicates an exhausted Krylov space.

eigenpairs which appear in the decomposition of $\mathbf{y}_0$. Hence one might interpret this as a special case of above argument.

However, an exhausted Krylov space is a numerical $\mathbf{A}$-invariant subspace. Any sophisticated linear solver should be able to find a good approximation within this space. Proceeding with the indetermined vector $\mathbf{y}_i$ should not be necessary, but there are no serious problem to expect although superlinear convergence is lost. The estimated relative residual gained in the recursion can be reduced to any arbitrary small size. A coarse limit for the accuracy of the real residual is given by the machine precision.

In this work and also in practice the behavior of MinRes and SymmLQ beyond the successful construction of a numerical $\mathbf{A}$-invariant subspace is of less interest. In practice a method as MinRes stops if the approximation is of reasonable quality. A stopping criterion in the Lanczos algorithm seems to be not necessary, but it is of more interest in the block case.

The Arnoldi and Lanczos process behave very similar before an $\mathbf{A}$-invariant subspace has been constructed. Using double projections within the Arnoldi algorithm it is possible to maintain orthogonality even beyond this point. However, for Hermitian linear systems with a single right-hand side there is no advantage to expect if the Arnoldi process with GMRes instead of Lanczos and MinRes is applied. The drawbacks of GMRes are higher costs in terms of computational work and memory requirements.

4.2. Block vectors

Definition. A **block vector** is a matrix $\mathbf{y} = \begin{pmatrix} \mathbf{y}^{(1)} & \dots & \mathbf{y}^{(\mathsf{s})} \end{pmatrix} \in \mathbb{C}^{N \times \mathsf{s}}$. ▲

Definition. The block vectors $\mathbf{y} \in \mathbb{C}^{N \times \mathsf{s}_1}$ and $\mathbf{x} \in \mathbb{C}^{N \times \mathsf{s}_2}$ are **orthogonal** if

$$\mathbf{y}^{\mathsf{H}}\mathbf{x} = \mathbf{0}$$

▲

This definition does not imply that the columns of $\mathbf{y}$ or $\mathbf{x}$ are orthogonal.

Definition. The block vectors $\mathbf{y}_1 \in \mathbb{C}^{N \times \mathsf{s}_1}, \dots, \mathbf{y}_n \in \mathbb{C}^{N \times \mathsf{s}_n}$ are **orthonormal** if

$$\mathbf{y}_i^{\mathsf{H}}\mathbf{y}_j = \delta_{i,j}\mathbf{I}_{\mathsf{s}_i \times \mathsf{s}_j}$$

▲

Lemma 3 *The block vectors $\mathbf{y}_1 \in \mathbb{C}^{N \times \mathsf{s}_1}, \dots, \mathbf{y}_n \in \mathbb{C}^{N \times \mathsf{s}_n}$ are orthonormal if and only if the block*

$$\mathbf{y}^{\mathsf{H}}\mathbf{y} = \mathbf{I}_{\mathsf{s}^{\Box} \times \mathsf{s}^{\Box}}$$

where $\mathbf{y} = \begin{pmatrix} \mathbf{y}_1 & \dots & \mathbf{y}_n \end{pmatrix}$ and $\mathsf{s}^{\Box} = \sum_{i=1}^{n} \mathsf{s}_i$

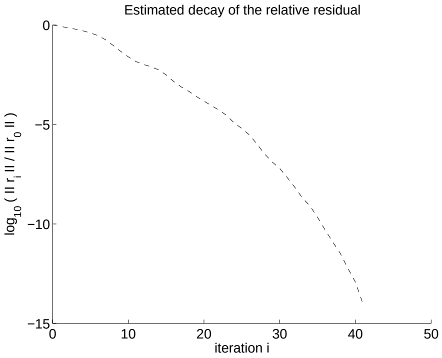


Figure 4.14.: Experiment 1

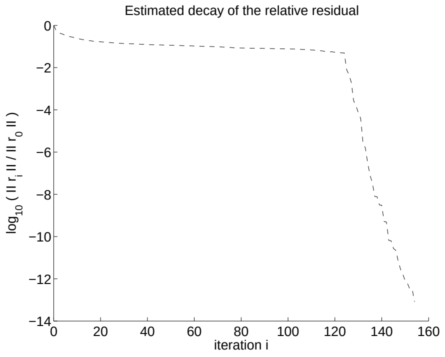


Figure 4.15.: Experiment 2

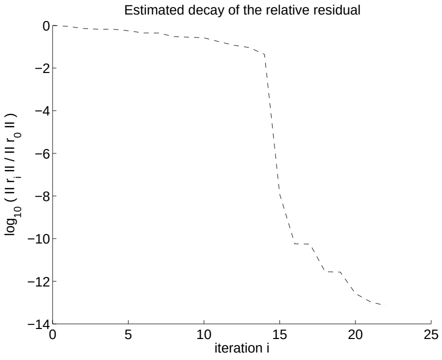


Figure 4.16.: Experiment 3

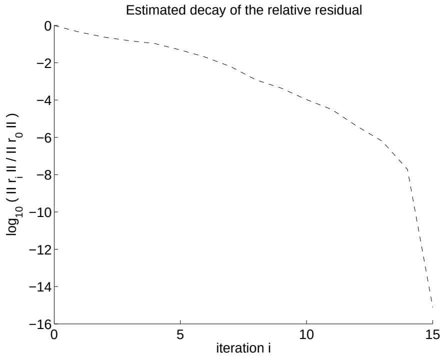


Figure 4.17.: Experiment 4

For all plots of the relative residuals we have used MinRes. The relative residuals are estimated, that is the update formula for the residual is applied. The stopping criterion is based on a relative residual smaller than 10^{-13}.

So orthonormality relies on the inner product of $\mathbb{C}^N$, although we can also equip the space of block vectors with an inner product.

Definition. For block vectors, the inner product $\langle . , . \rangle_F$ and the norm $\|.\|_F$ it induces are defined by

$$\langle \mathbf{x}, \mathbf{y} \rangle_F :\equiv \text{trace}\, \mathbf{x}^{\mathsf{H}}\, \mathbf{y}\,, \qquad \|\mathbf{x}\|_F :\equiv \sqrt{\langle \mathbf{x}, \mathbf{x} \rangle_F} :\equiv \sqrt{\text{trace}\, \mathbf{x}^{\mathsf{H}}\, \mathbf{x}}\,.$$

▲

If

$$\mathbf{x} = \left(\begin{array}{ccc} x^{(1)} & \dots & x^{(\mathsf{s})} \end{array} \right) = \left(\begin{array}{c} \xi_{i,j} \end{array} \right) \quad \in \mathbb{C}^{N \times \mathsf{s}}\,,$$

then

$$\|\mathbf{x}\|_F = \sqrt{\sum_{j=1}^{\mathsf{s}} \|x^{(j)}\|_2^2} = \sqrt{\sum_{j=1}^{\mathsf{s}} \sum_{i=1}^{N} |\xi_{i,j}|^2}\,.$$

We call it Frobenius norm, although a block vector does not represent a linear map.

Lemma 4 *The Frobenius norm is consistent, i.e.*

$$\|\mathbf{AB}\|_F \le \|\mathbf{A}\|_F \|\mathbf{B}\|_F \tag{4.3}$$

and it is an upper bound for the matrix norm induced by the 2*-norm*

$$\|\mathbf{A}\|_2 \le \|\mathbf{A}\|_F \tag{4.4}$$

where $\|\mathbf{A}\|_2 = \max_{\|\mathbf{x}\|_2 = 1} \|\mathbf{Ax}\|_2 = \max\{|\lambda_1|, \dots, |\lambda_N|\}$.

Proof: Compare with Theorem 4.11 and Theorem 4.16 in [31]. □

Lemma 5 *Let* $\mathbf{C} \in \mathbb{C}^{N \times \mathsf{s}}$ *where* $\mathbf{C} = \left(\begin{array}{ccc} \mathbf{c}^{(1)} & \dots & \mathbf{c}^{(\mathsf{s})} \end{array} \right)$ *and* $\|\mathbf{c}^{(i)}\|_2 = 1$ *for all* $i = 1, \dots \mathsf{s}$ *then for* $\mathbf{x} \in \mathbb{C}^{\mathsf{s} \times k}$ *where* $k \in \mathbb{N}$

$$\|\mathbf{Cx}\|_F \le \sqrt{\mathsf{s}} \|\mathbf{x}\|_F \tag{4.5}$$

If the columns of $\mathbf{C}$ *are in addition orthonormal then*

$$\|\mathbf{Cx}\|_F = \|\mathbf{x}\|_F \tag{4.6}$$

Proof: The definition of $\|.\|_F$ implies

$$\|\mathbf{Cx}\|_F^2 = \sum_{i=1}^{k} \|\mathbf{Cx}^{(i)}\|_2^2 = \sum_{i=1}^{k} \| \sum_{j=1}^{\mathsf{s}} \mathbf{c}^{(j)} x_j^{(i)} \|_2^2$$

where $\mathbf{x}^{(i)}$ is the ith column of $\mathbf{x}$ and $x_j^{(i)}$ is the jth entry in this column. The triangle inequality yields

$$\sum_{i=1}^{k} \| \sum_{j=1}^{\mathsf{s}} \mathbf{c}^{(j)} x_j^{(i)} \|_2^2 \leq \sum_{i=1}^{k} \left(\sum_{j=1}^{\mathsf{s}} \| \mathbf{c}^{(j)} x_j^{(i)} \|_2 \right)^2 = \sum_{i=1}^{k} \left(\sum_{j=1}^{\mathsf{s}} |x_j^{(i)}| \right)^2 = \sum_{i=1}^{k} \|\mathbf{x}^{(i)}\|_1^2.$$

The 1-Norm can be bounded by $\|\mathbf{x}^{(i)}\|_1 \leq \sqrt{\mathsf{s}} \|\mathbf{x}^{(i)}\|_2$

$$\sum_{i=1}^{k} \|\mathbf{x}^{(i)}\|_1^2 \leq \mathsf{s} \sum_{i=1}^{k} \|\mathbf{x}^{(i)}\|_2^2 = \mathsf{s} \|\mathbf{x}\|_F^2.$$

If the columns of $\mathbf{C}$ are in addition orthonormal then

$$\|\mathbf{C}\mathbf{x}\|_F^2 = \sum_{i=1}^{k} \|\mathbf{C}\mathbf{x}^{(i)}\|_2^2 = \sum_{i=1}^{k} \|\mathbf{x}^{(i)}\|_2^2 = \|\mathbf{x}\|_F^2.$$

□

Definition. The column rank of a block vector $\mathbf{y} = \left(\begin{array}{ccc} \mathbf{y}^{(1)} & \ldots & \mathbf{y}^{(\mathsf{s})} \end{array} \right) \in \mathbb{C}^{N \times \mathsf{s}}$ is

$$\mathsf{rank}\, \mathbf{y} = \dim \mathsf{span} \left\{ \mathbf{y}^{(1)}, \ldots, \mathbf{y}^{(\mathsf{s})} \right\}$$

▲

Given a block vector it is essential for the further work to estimate the rank of a block vector. The point is to decide what the rank is in the presence of error in non exact arithmetic. Common sense is this answer:

Definition. Given a small tolerance tol the numerical column rank of a block vector $\widetilde{\mathbf{y}}_0$ with respect to this tolerance is the number of singular values of $\widetilde{\mathbf{y}}_0$ greater than this tolerance. ▲

Hence the smallest singular value of matrix with full numerical column rank is greater than the tolerance tol. The idea is not to compute but to estimate the singular values of a given matrix $\widetilde{\mathbf{y}}_0$. A powerful software package [25] is available and used in our implementation to construct a full rank-revealing QR decomposition as proposed by Chan [3]:

$$\boxed{\widetilde{\mathbf{y}}_0 =: \left(\begin{array}{cc} \mathbf{y}_0 & \mathbf{y}_0^{\Delta} \end{array} \right) \left(\begin{array}{cc} \boldsymbol{\rho}_0 & \boldsymbol{\rho}_0^{\square} \\ \mathbf{0} & \boldsymbol{\rho}_0^{\Delta} \end{array} \right) \boldsymbol{\pi}_0^{\mathsf{T}} =: \left(\begin{array}{cc} \mathbf{y}_0 & \mathbf{y}_0^{\Delta} \end{array} \right) \left(\begin{array}{c} \boldsymbol{\eta}_0 \\ \boldsymbol{\eta}_0^{\Delta} \end{array} \right),} \quad (4.7)$$

where:

$\boldsymbol{\pi}_0$ is an $\mathsf{s} \times \mathsf{s}$ permutation matrix.

$\mathbf{y}_0$ is an $N \times \mathsf{s}_0$ matrix with full numerical column rank.

$\mathbf{y}_0^{\triangle}$ is an $N \times \mathsf{s} - \mathsf{s}_0$ matrix whose columns span an approximation to the null space of $\widetilde{\mathbf{y}}_0$, if $\widetilde{\mathbf{y}}_0$ is a square block,

$\boldsymbol{\rho}_0$ is an $\mathsf{s}_0 \times \mathsf{s}_0$ upper triangular, nonsingular matrix.

$\boldsymbol{\rho}_0^{\square}$ is an $\mathsf{s}_0 \times (\mathsf{s} - \mathsf{s}_0)$ matrix.

$\boldsymbol{\rho}_0^{\triangle}$ is an upper triangular $(\mathsf{s} - \mathsf{s}_0) \times (\mathsf{s} - \mathsf{s}_0)$ matrix. It is $\|\boldsymbol{\rho}_0^{\triangle}\|_F = O(\sigma_{\mathsf{s}_0+1})$, where σ_{s_0+1} is the largest singular value of $\widetilde{\mathbf{y}}_0$ smaller than tol.

If the matrix $\widetilde{\mathbf{y}}_0$ has full numerical rank, i.e. $\mathsf{s}_0 = \mathsf{s}$, the blocks $\mathbf{y}_0^{\triangle}$, $\boldsymbol{\rho}_0^{\triangle}$, $\boldsymbol{\rho}_0^{\square}$, $\boldsymbol{\eta}_0^{\triangle}$ are empty.

4.3. Block Krylov subspaces

Definition. Given $\mathbf{A} \in \mathbb{C}^{N \times N}$ and $\mathbf{v}_0 \in \mathbb{C}^{N \times \mathsf{s}}$, the **block Krylov subspaces** $\mathcal{B}_n^{\square}$ generated by $\mathbf{A}$ from $\mathbf{v}_0$ are

$$\mathcal{B}_n^{\square}(\mathbf{A}, \mathbf{v}_0) := \underbrace{\mathcal{B}_n(\mathbf{A}, \mathbf{v}_0) \times \mathcal{B}_n(\mathbf{A}, \mathbf{v}_0) \times \ldots \mathcal{B}_n(\mathbf{A}, \mathbf{v}_0)}_{\mathsf{s}\,\text{times}} \tag{4.8}$$

▲

The vectors $\mathbf{r}_n^{(i)}$ are contained in $\mathcal{B}_{n+1}(\mathbf{A}, \mathbf{r}_0)$ for all $i = 1, 2 \ldots, \mathsf{s}$ if and only if the block vector $\mathbf{r}_n \in \mathcal{B}_n^{\square}$. Observing that

$$\mathcal{B}_n(\mathbf{A}, \mathbf{v}_0) = \left\{ \sum_{k=0}^{n-1} \mathbf{A}^k \mathbf{v}_0 \boldsymbol{\beta}_k \,;\ \boldsymbol{\beta}_k \in \mathbb{C}^{\mathsf{s} \times 1} \right\}. \tag{4.9}$$

we gain

$$\mathcal{B}_n^{\square}(\mathbf{A}, \mathbf{v}_0) = \left\{ \sum_{k=0}^{n-1} \mathbf{A}^k \mathbf{v}_0 \gamma_k \,;\ \gamma_k \in \mathbb{C}^{\mathsf{s} \times \mathsf{s}} \right\}. \tag{4.10}$$

A basis of $\mathcal{B}_n^{\square}(\mathbf{A}, \mathbf{v}_0)$ would be a set of block vectors such that every block vector in $\mathcal{B}_n^{\square}(\mathbf{A}, \mathbf{v}_0)$ is a unique linear combination of the block vectors contained in the basis. Although the construction of such a set is not difficult our approach is a bit different. Instead of an ordinary linear combination we use an approach based on the term block basis.

Definition. Given a set of block vectors $\mathbf{v}_1 \in \mathbb{C}^{N \times \mathsf{s}_1}, \mathbf{v}_2 \in \mathbb{C}^{N \times \mathsf{s}_2}, \ldots, \mathbf{v}_n \in \mathbb{C}^{N \times \mathsf{s}_n}$ where $\mathsf{rank}\, \mathbf{v}_i = \mathsf{s}_i \leq \mathsf{s}$ for all $i = 1, 2, \ldots, n$ the block span of those block vectors is defined by

$$\text{block span}\,(\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n) = \left\{ \sum_{i=1}^{n} \mathbf{v}_i \gamma_i \quad ;\ \gamma_i \in \mathbb{C}^{\mathsf{s}_i \times \mathsf{s}} \right\}.$$

Let $\mathcal{B} =$ block span $(\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n)$ then the set of block vectors $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n$ is a **block basis** of $\mathcal{B}$ if and only if

$$\sum_{i=0}^{n} \mathbf{v}_i \boldsymbol{\gamma}_i = \mathbf{0}$$

implies $\boldsymbol{\gamma}_i = \mathbf{0}$ for all $i = 1, 2, \ldots n$. The block basis is called **orthonormal** if the block vectors are orthonormal. ▲

As a consequence of (4.10) we note

Lemma 6

$$\mathcal{B}_n^{\square}(\mathbf{A}, \mathbf{v}_0) = \text{block span}\left(\mathbf{v}_0, \mathbf{A}\mathbf{v}_0, \ldots, \mathbf{A}^{n-1}\mathbf{v}_0\right)$$

Keeping in mind our goal to construct an orthonormal basis of $\mathcal{B}_n(\mathbf{A}, \mathbf{v}_0)$ we formulate

Lemma 7 *Let the block vectors*

$$\mathbf{y}_0 = \left(\begin{array}{ccc} \mathbf{y}_0^{(1)} & \ldots & \mathbf{y}_0^{(\mathsf{s}_0)} \end{array} \right), \quad \ldots, \quad \mathbf{y}_{n-1} = \left(\begin{array}{ccc} \mathbf{y}_{n-1}^{(1)} & \ldots & \mathbf{y}_{n-1}^{(\mathsf{s}_{n-1})} \end{array} \right)$$

be an orthonormal block basis for $\mathcal{B}_n^{\square}(\mathbf{A}, \mathbf{v}_0)$*. Then* $\mathbf{y}_0^{(1)}, \ldots, \mathbf{y}_0^{(\mathsf{s}_0)}, \mathbf{y}_1^{(1)}, \ldots, \mathbf{y}_{n-1}^{(\mathsf{s}_{n-1})}$ *is an orthonormal basis of* $\mathcal{B}_n(\mathbf{A}, \mathbf{v}_0)$*.*

Proof: Due to Lemma 3 the vectors $\mathbf{y}_0^{(1)}, \ldots, \mathbf{y}_0^{(\mathsf{s}_0)}, \mathbf{y}_1^{(1)}, \ldots, \mathbf{y}_{n-1}^{(\mathsf{s}_{n-1})}$ are orthonormal. Let $\mathbf{x} \in \mathcal{B}_n(\mathbf{A}, \mathbf{v}_0)$. Interpreted as a first column of a block vector it is a unique linear combination of all columns of those block vectors listed in the block basis $\mathbf{y}_0, \ldots, \mathbf{y}_{n-1}$. □

Definition. $\mathsf{s}_n^{\square}(\mathbf{A}, \mathbf{v}_0) = \dim \mathcal{B}_n(\mathbf{A}, \mathbf{v}_0)$ ▲

So if $\mathbf{y}_0^{(1)}, \ldots, \mathbf{y}_0^{(\mathsf{s}_0)}, \mathbf{y}_1^{(1)}, \ldots, \mathbf{y}_{n-1}^{(\mathsf{s}_{n-1})}$ is an orthonormal basis for $\mathcal{B}_n(\mathbf{A}, \mathbf{v}_0)$ then $\mathsf{s}_n^{\square}(\mathbf{A}, \mathbf{v}_0) = \sum_{i=0}^{n-1} \mathsf{s}_i$.

Definition. The smallest index n with $n = \mathsf{s}_n^{\square}(\mathbf{A}, \mathbf{v}_0) = \mathsf{s}_{n+1}^{\square}(\mathbf{A}, \mathbf{v}_0)$ is called the **block grade of A with respect to** $\mathbf{v}_0$ and denoted by $\bar{\nu}^{\square}(\mathbf{A}, \mathbf{v}_0)$. ▲

4.4. A block Lanczos process with deflations

With the above framework we can introduce the block Lanczos process in an analogue manner as we used when we introduced the Lanczos process. The block Lanczos process creates at least in exact arithmetic an orthonormal block basis $\mathbf{y}_0, \mathbf{y}_1, \ldots, \mathbf{y}_{n-1}$ for $\mathcal{B}_n^{\square}(\mathbf{A}, \mathbf{v}_0)$:

$$\mathcal{B}_n^{\square}(\mathbf{A}, \mathbf{v}_0) = \text{block span}\left\{\mathbf{v}_0, \mathbf{A}\mathbf{v}_0, \ldots, \mathbf{A}^{n-1}\mathbf{v}_0\right\} = \text{block span}\left\{\mathbf{y}_0, \mathbf{y}_1, \ldots, \mathbf{y}_{n-1}\right\}$$

where:

$n \leq \bar{\nu}^{\square}(\mathbf{A}, \mathbf{v}_0)$ with $\mathbf{v}_0 \in \mathbb{C}^{N \times \mathsf{s}}$,

$\left\{\mathbf{y}_i \in \mathbb{C}^{N \times \mathsf{s}_i}\right\}$ is an orthonormal block basis with $\mathsf{s} \geq \mathsf{s}_0 \geq \mathsf{s}_1 \geq \ldots \geq \mathsf{s}_i \geq \ldots \geq \mathsf{s}_{n-1}$.

Algorithm 5 (Hermitian Block Lanczos algorithm) .
Let a Hermitian matrix $\mathbf{A}$ *and an orthonormal block vector* $\mathbf{y}_0$ *be given. For constructing a nested set of orthonormal block bases* $\{\mathbf{y}_0, \mathbf{y}_1, \ldots, \mathbf{y}_m\}$ *for the nested Block Krylov subspaces* $\mathcal{B}_{m+1}^{\square}(\mathbf{A}, \mathbf{y}_0)$ $(m = 1, 2, \cdots \leq \bar{\nu}^{\square}(\mathbf{A}, \mathbf{y}_0) - 1)$ *compute, for* $n = 1, 2, \ldots, m$*:*

1. *Apply* $\mathbf{A}$ *to* $\mathbf{y}_{n-1} \perp \mathcal{B}_{n-1}^{\square}(\mathbf{A}, \mathbf{y}_0)$*:*

$$\widetilde{\mathbf{y}}_n := \mathbf{A}\mathbf{y}_{n-1}. \tag{4.11}$$

2. *Subtract the projection of* $\widetilde{\mathbf{y}}_n$ *on the last two basis block vectors:*

$$\begin{aligned} \widetilde{\mathbf{y}}_n &:= \widetilde{\mathbf{y}}_n - \mathbf{y}_{n-2}\boldsymbol{\beta}_{n-2}^{\mathsf{H}} \quad \textit{if } n > 1, & (4.12)\\ \boldsymbol{\alpha}_{n-1} &:= \mathbf{y}_{n-1}^{\mathsf{H}}\widetilde{\mathbf{y}}_n\,, & (4.13)\\ \widetilde{\mathbf{y}}_n &:= \widetilde{\mathbf{y}}_n - \mathbf{y}_{n-1}\boldsymbol{\alpha}_{n-1}\,. & (4.14) \end{aligned}$$

3. *QR factorization of* $\widetilde{\mathbf{y}}_n \perp \mathcal{B}_n^{\square}(\mathbf{A}, \mathbf{y}_0)$ *with* $\mathsf{rank}\,\widetilde{\mathbf{y}}_n = \mathsf{s}_n \leq \mathsf{s}_{n-1}$*:*

$$\widetilde{\mathbf{y}}_n =: \begin{pmatrix} \mathbf{y}_n & \mathbf{y}_n^{\triangle} \end{pmatrix} \begin{pmatrix} \boldsymbol{\rho}_n & \boldsymbol{\rho}_n^{\square} \\ \mathbf{0} & \boldsymbol{\rho}_n^{\triangle} \end{pmatrix} \boldsymbol{\pi}_n^{\mathsf{T}} =: \begin{pmatrix} \mathbf{y}_n & \mathbf{y}_n^{\triangle} \end{pmatrix} \begin{pmatrix} \boldsymbol{\beta}_{n-1} \\ \boldsymbol{\beta}_{n-1}^{\triangle} \end{pmatrix}, \tag{4.15}$$

where:

$\boldsymbol{\pi}_n$ *is an* $\mathsf{s}_{n-1} \times \mathsf{s}_{n-1}$ *permutation matrix.*

$\mathbf{y}_n$ *is an* $N \times \mathsf{s}_n$ *matrix with full numerical column rank going into the basis.*

$\mathbf{y}_n^{\triangle}$ *is an* $N \times (\mathsf{s}_{n-1} - \mathsf{s}_n)$ *matrix that will be deflated,*

$\boldsymbol{\rho}_n$ *is an* $\mathsf{s}_n \times \mathsf{s}_n$ *upper triangular, nonsingular matrix.*

$\boldsymbol{\rho}_n^{\square}$ *is an* $\mathsf{s}_n \times (\mathsf{s}_{n-1} - \mathsf{s}_n)$ *matrix.*

$\boldsymbol{\rho}_n^{\triangle}$ *is an upper triangular* $(\mathsf{s}_{n-1} - \mathsf{s}_n) \times (\mathsf{s}_{n-1} - \mathsf{s}_n)$ *matrix.*
It is $\|\boldsymbol{\rho}_n^{\triangle}\|_F = O(\sigma_{\mathsf{s}_n+1})$*, where* σ_{s_n+1} *is the largest singular value of* $\widetilde{\mathbf{y}}_n$ *smaller than* tol.

The permutations are encapsulated in the block coefficients $\boldsymbol{\beta}_i$. Let

$$\mathbf{P}_n :\equiv \text{block diag}\left(\boldsymbol{\pi}_1, \ldots, \boldsymbol{\pi}_n\right)$$

be the permutation matrix that describes all these permutations. Note that $\mathbf{P}_n^{\mathsf{T}} = \mathbf{P}_n^{-1}$. If tol $= 0$ we speak of **exact deflation**. The algorithm uses block permutations as well as deflation. It is therefore not immediately apparent that the fundamental Lanczos relationships still hold.

4.4.1. Block Lanczos in exact arithmetic

Theorem 8 *With exact deflation the block vectors* $\left\{\mathbf{y}_0, \mathbf{y}_1, \ldots, \mathbf{y}_{\bar{\nu}^{\Box}(\mathbf{A},\mathbf{y}_0)-1}\right\}$ *constructed by this algorithm are orthonormal. The first n block vectors are a block basis for* $\mathcal{B}_n^{\Box}\left(\mathbf{A}, \mathbf{y}_0\right)$ *for every* $n = 1, 2, \ldots \bar{\nu}^{\Box}\left(\mathbf{A}, \mathbf{y}_0\right)$. *Moreover,*

$$\mathbf{A}\mathbf{Y}_n = \mathbf{Y}_{n+1}\underline{\mathbf{T}}_n \tag{4.16}$$

where $n < \bar{\nu}^{\Box}\left(\mathbf{A}, \mathbf{y}_0\right)$,

$$\mathbf{Y}_n = \begin{pmatrix} \mathbf{y}_0 & \mathbf{y}_1 & \cdots & \mathbf{y}_{n-1} \end{pmatrix},$$

$$\underline{\mathbf{T}}_n = \left(\begin{array}{ccccc} \boldsymbol{\alpha}_0 & \boldsymbol{\beta}_0^{\mathsf{H}} & & & \\ \boldsymbol{\beta}_0 & \boldsymbol{\alpha}_1 & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & \boldsymbol{\beta}_{n-2}^{\mathsf{H}} \\ & & & \boldsymbol{\beta}_{n-2} & \boldsymbol{\alpha}_{n-1} \\ \hline & & & & \boldsymbol{\beta}_{n-1} \end{array}\right) \in \mathbb{C}^{\mathsf{s}_{n+1}^{\Box} \times \mathsf{s}_n^{\Box}} \tag{4.17}$$

and $\boldsymbol{\alpha}_i = \boldsymbol{\alpha}_i^{\mathsf{H}}$ *for all* $i = 1, 2, \ldots n-1$.

We denote by $\mathbf{T}_n$ the upper Hermitian square matrix of $\underline{\mathbf{T}}_n$. The proof is a slight modification of the proof we used when we introduced the Lanczos process.

Proof: The block vector $\mathbf{y}_0$ is an orthonormal block basis for $\mathcal{B}_1^{\Box}\left(\mathbf{A}, \mathbf{y}_0\right)$. Applying (4.11) we note that $\widetilde{\mathbf{y}}_1 = \mathbf{A}\mathbf{y}_0 \in \mathcal{B}_2^{\Box}\left(\mathbf{A}, \mathbf{y}_0\right)$. As

$$\boldsymbol{\alpha}_0^{\mathsf{H}} = \left(\mathbf{y}_0^{\mathsf{H}}\mathbf{A}\mathbf{y}_0\right)^{\mathsf{H}} = \left(\mathbf{A}\mathbf{y}_0\right)^{\mathsf{H}}\mathbf{y}_0 = \mathbf{y}_0^{\mathsf{H}}\mathbf{A}\mathbf{y}_0 = \boldsymbol{\alpha}_0,$$

we have $\boldsymbol{\alpha}_0^{\mathsf{H}} = \boldsymbol{\alpha}_0$. By the definition of $\boldsymbol{\alpha}_0$ the block vector $\widetilde{\mathbf{y}}_1$ (4.14) is orthogonal to $\mathbf{y}_0$

$$\mathbf{y}_0^{\mathsf{H}}\widetilde{\mathbf{y}}_1 = \mathbf{y}_0^{\mathsf{H}}\left(\mathbf{A}_0\mathbf{y}_0 - \mathbf{y}_0\boldsymbol{\alpha}_0\right) = \mathbf{y}_0^{\mathsf{H}}\mathbf{A}_0\mathbf{y}_0 - \boldsymbol{\alpha}_0 = \mathbf{0}.$$

If $\widetilde{\mathbf{y}}_1 = \mathbf{0}$ (4.14) then the columns of $\mathbf{y}_0$ span an invariant subspace of $\mathbf{A}$ and $\bar{\nu}^{\Box}\left(\mathbf{A}, \mathbf{y}_0\right) = 1$.

With exact deflation $\boldsymbol{\rho}_1^{\Delta} = \mathbf{0}$ or empty (4.15), i.e. $\widetilde{\mathbf{y}}_1 = \mathbf{y}_1\boldsymbol{\beta}_0$, where $\boldsymbol{\beta}_0$ has full row rank. So $\mathbf{y}_0^{\mathsf{H}}\mathbf{y}_1\boldsymbol{\beta}_0 = \mathbf{0}$ implies $\mathbf{y}_0^{\mathsf{H}}\mathbf{y}_1 = \mathbf{0}$. The block $\mathbf{y}_1$ is orthonormal by the definition of $\begin{pmatrix} \mathbf{y}_1 & \mathbf{y}_1^{\Delta} \end{pmatrix}$.

Comparing (4.14) and (4.15) we note that

$$\mathbf{A}\mathbf{y}_0 - \mathbf{y}_0\boldsymbol{\alpha}_0 = \mathbf{y}_1\boldsymbol{\beta}_0$$

or rearranged

$$\mathbf{A}\mathbf{Y}_1 = \mathbf{Y}_2 \begin{pmatrix} \boldsymbol{\alpha}_0 \\ \boldsymbol{\beta}_0 \end{pmatrix},$$

so (4.16) holds for $n = 1$.

Assume the statement holds for n. Then

- $\{\mathbf{y}_0, \ldots, \mathbf{y}_{n-1}\}$ is an orthonormal block basis for $\mathcal{B}_n^{\square}(\mathbf{A}, \mathbf{y}_0)$,
- $\mathbf{y}_{n-1} \perp \mathcal{B}_{n-1}^{\square}(\mathbf{A}, \mathbf{y}_0)$
- and the three-term recurrence relation

$$\mathbf{A}\mathbf{y}_{n-2} = \mathbf{y}_{n-3}\boldsymbol{\beta}_{n-3}^{\mathsf{H}} + \mathbf{y}_{n-2}\boldsymbol{\alpha}_{n-2} + \mathbf{y}_{n-1}\boldsymbol{\beta}_{n-2} \tag{4.18}$$

  holds with $\boldsymbol{\beta}_{n-3} \in \mathbb{C}^{\mathsf{s}_{n-2}\times\mathsf{s}_{n-3}}, \boldsymbol{\alpha}_{n-2} = \boldsymbol{\alpha}_{n-2}^{\mathsf{H}} \in \mathbb{C}^{\mathsf{s}_{n-2}\times\mathsf{s}_{n-2}}$ and $\boldsymbol{\beta}_{n-2} \in \mathbb{C}^{\mathsf{s}_{n-1}\times\mathsf{s}_{n-2}}$.

Let $\widetilde{\mathbf{y}}_n = \mathbf{A}\mathbf{y}_{n-1} \in \mathcal{B}_{n+1}^{\square}(\mathbf{A}, \mathbf{y}_0)$. First we remark that $\widetilde{\mathbf{y}}_n \perp \mathcal{B}_{n-2}^{\square}(\mathbf{A}, \mathbf{y}_0)$. Assume $\mathbf{g} \in \mathcal{B}_{n-2}^{\square}(\mathbf{A}, \mathbf{y}_0)$. We have

$$\mathbf{g}^{\mathsf{H}}\widetilde{\mathbf{y}}_n = \mathbf{g}^{\mathsf{H}}\mathbf{A}\mathbf{y}_{n-1} = (\mathbf{A}\mathbf{g})^{\mathsf{H}}\,\mathbf{y}_{n-1} = \mathbf{0}.$$

as $\mathbf{A}\mathbf{g} \in \mathcal{B}_{n-1}^{\square}(\mathbf{A}, \mathbf{y}_0)$ but $\mathbf{y}_{n-1} \perp \mathcal{B}_{n-1}^{\square}(\mathbf{A}, \mathbf{y}_0)$. It is this property that makes the difference between the block Lanczos and the block Arnoldi process. If $\mathbf{A}$ is not Hermitian the block vector $\widetilde{\mathbf{y}}_n$ has to be orthogonalized with respect to all basis block vectors $\mathbf{y}_0, \ldots, \mathbf{y}_{n-1}$.

Using exactly the same argument as above for $\boldsymbol{\alpha}_0$ we conclude that $\boldsymbol{\alpha}_{n-1} = \boldsymbol{\alpha}_{n-1}^{\mathsf{H}}$.

The block vector $\widetilde{\mathbf{y}}_n$ (4.14) is also orthogonal to the block vectors $\mathbf{y}_{n-2}$ and $\mathbf{y}_{n-1}$ as

$$\mathbf{y}_{n-2}^{\mathsf{H}}\widetilde{\mathbf{y}}_n = \mathbf{y}_{n-2}^{\mathsf{H}}\left(\mathbf{A}\mathbf{y}_{n-1} - \mathbf{y}_{n-1}\boldsymbol{\alpha}_{n-1} - \mathbf{y}_{n-2}\boldsymbol{\beta}_{n-2}^{\mathsf{H}}\right) = (\mathbf{A}\mathbf{y}_{n-2})^{\mathsf{H}}\,\mathbf{y}_{n-1} - \boldsymbol{\beta}_{n-2}^{\mathsf{H}} = \mathbf{0}$$

by using the recurrence relation (4.18) and

$$\mathbf{y}_{n-1}^{\mathsf{H}}\widetilde{\mathbf{y}}_n = \mathbf{y}_{n-1}^{\mathsf{H}}\left(\mathbf{A}\mathbf{y}_{n-1} - \mathbf{y}_{n-1}\boldsymbol{\alpha}_{n-1} - \mathbf{y}_{n-2}\boldsymbol{\beta}_{n-2}\right) = \mathbf{y}_{n-1}^{\mathsf{H}}\mathbf{A}\mathbf{y}_{n-1} - \boldsymbol{\alpha}_{n-1} = \mathbf{0}$$

by the definition of $\boldsymbol{\alpha}_{n-1}$.

If $\widetilde{\mathbf{y}}_n = \mathbf{0}$ (4.14) then $\bar{\nu}^{\square}(\mathbf{A}, \mathbf{y}_0) = n$ and the algorithm stops. Otherwise with exact deflation $\boldsymbol{\rho}_n^{\Delta} = \mathbf{0}$ or empty in (4.15), i.e. $\widetilde{\mathbf{y}}_n = \mathbf{y}_n\boldsymbol{\beta}_{n-1}$, where $\boldsymbol{\beta}_{n-1}$ has full row rank. So $\mathbf{y}_{n-1}^{\mathsf{H}}\mathbf{y}_n\boldsymbol{\beta}_{n-1} = \mathbf{0}$ and $\mathbf{y}_{n-2}^{\mathsf{H}}\mathbf{y}_n\boldsymbol{\beta}_{n-1} = \mathbf{0}$ imply $\mathbf{y}_{n-1}^{\mathsf{H}}\mathbf{y}_n = \mathbf{0}$ and $\mathbf{y}_{n-2}^{\mathsf{H}}\mathbf{y}_n = \mathbf{0}$. The block $\mathbf{y}_n$ is orthonormal by the definition of $\begin{pmatrix} \mathbf{y}_n & \mathbf{y}_n^{\Delta} \end{pmatrix}$.

Comparing (4.14) and (4.15) we note that

$$\widetilde{\mathbf{y}}_n = \mathbf{y}_n \boldsymbol{\beta}_{n-1} = \mathbf{A}\mathbf{y}_{n-1} - \mathbf{y}_{n-2}\boldsymbol{\beta}_{n-2}^{\mathsf{H}} - \mathbf{y}_{n-1}\boldsymbol{\alpha}_{n-1}$$

which is the desired result. □

4.4.2. Experiments using exact deflation

For the ordinary Lanczos method we saw in the first experiment that the subdiagonal entries remain rather large although the Krylov vectors are far away from being linear independent or even orthogonal.

Experiment 5 *Let* $\mathbf{A}$ *a* 400×400 *discrete Laplacian on an uniform grid in a square. The initial block vector* $\mathbf{y}_0$ *has random entries uniformly distributed in interval* $[0,1]$ *and* 5 *columns.*

$n = 20;\ N = n^2;$
$A = gallery('poisson', n);$
$y0t = rand(N, 5);$

The result is rather similar to the first experiment (see Figure 4.2). The smallest singular values of the subdiagonal block $\beta_0, \ldots, \beta_{148}$ *are all larger than* 10^{-1}*. In particular the singular values do not indicate any loss of orthogonality in the block Lanczos process. As reason for the loss of orthogonality we identify again the converged Ritz vectors acting as magnetic poles.*

In order to observe deflation we have to construct an experiment where small singular values arise in the block Lanczos process. Ignoring the small subdiagonal entries has led to a loss of orthogonality in experiment 3.

Experiment 6 *Here we use a block vector* $\widetilde{\mathbf{y}}_0$ *where the columns are random linear combinations of the same* 30 *eigenvectors of* $\mathbf{A}$ *which is* 100×100 *sparse random matrix. Hence this* 30 *eigenvectors are an orthonormal basis for the* $\mathbf{A}$*-invariant subspace* $\mathcal{B}_{30}(\mathbf{A}, \widetilde{\mathbf{y}}_0)$*.*

$k = 30;$
$[V, D] = eig(A);$
$p = randperm(N);$
$y0t = V(:, p(1:k)) * rand(k, 3);$

The block Lanczos algorithm constructs the block Krylov vectors $\mathbf{y}_0, \ldots, \mathbf{y}_9$ *with very accurate precision. The block orthogonalization is more precise than working with the ith column of* $\widetilde{\mathbf{y}}_0$ *in order to generate the equivalent space* $\mathcal{K}_{30}\left(\mathbf{A}, \widetilde{\mathbf{y}}_0^{(i)}\right)$*. It seems there is a connection between the dip in the singular values and the peak in the condition of the diagonal blocks* $\widetilde{\boldsymbol{\alpha}}_i$*, which appear in the QR decomposition of* $\underline{\mathbf{T}}_n$*. Later we will*

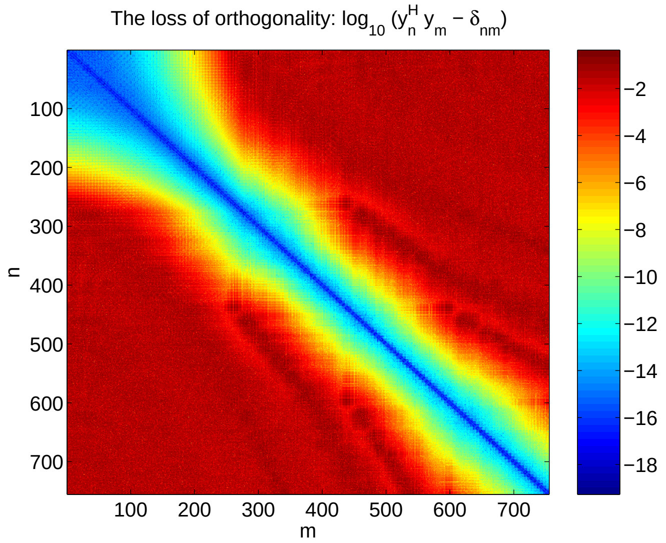


Figure 4.18.: Experiment 5: Colormap of the matrix $\mathbf{V} = \log|\mathbf{Y}_{150}^{\mathsf{H}}\mathbf{Y}_{150}^{\mathsf{H}} - \mathbf{I}_{150}|$. Note that $\mathbf{V}$ is a 750×750 matrix. The block vectors $\mathbf{y}_i$ where $i = 1, \ldots, 149$ have been constructed using the block Lanczos algorithm with exact deflation.

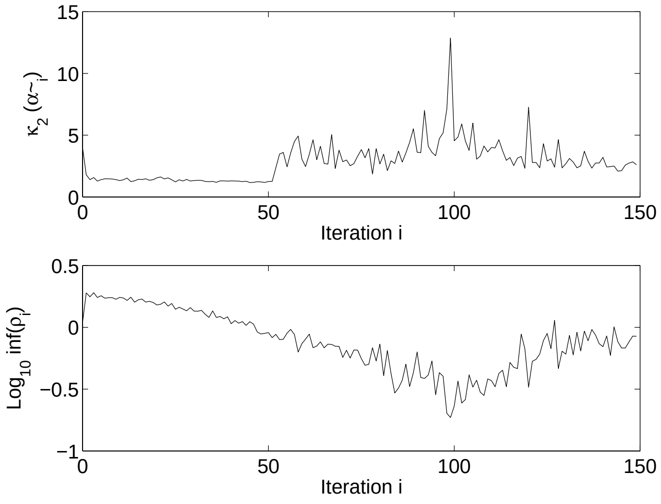


Figure 4.19.: Experiment 6: The smallest singular values of the subdiagonal blocks $\boldsymbol{\beta}_i$ are all larger than 10^{-1}.

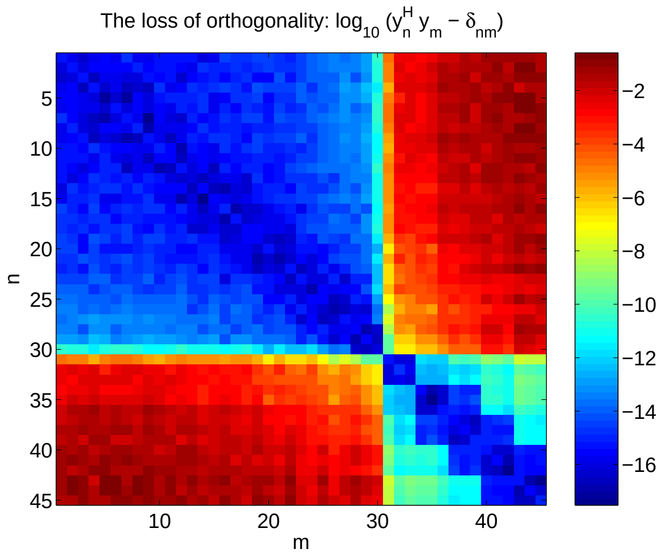


Figure 4.20.: Experiment 6: Colormap of the matrix $\mathbf{V} = \log|\mathbf{Y}_{15}^{\mathsf{H}}\mathbf{Y}_{15}^{\mathsf{H}} - \mathbf{I}_{15}|$. Note that $\mathbf{V}$ is a 45×45 matrix. The block vectors $\mathbf{y}_i$ where $i = 1, \dots, 14$ have been constructed using the block Lanczos algorithm with exact deflation.

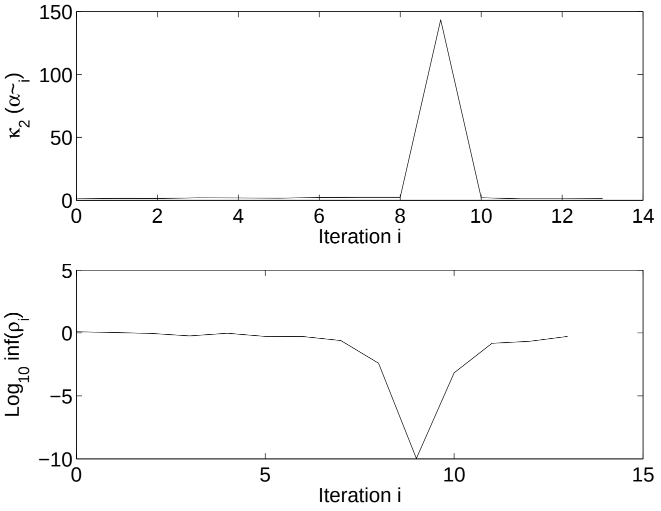


Figure 4.21.: Experiment 6: An exhausted block Krylov space is indicated.

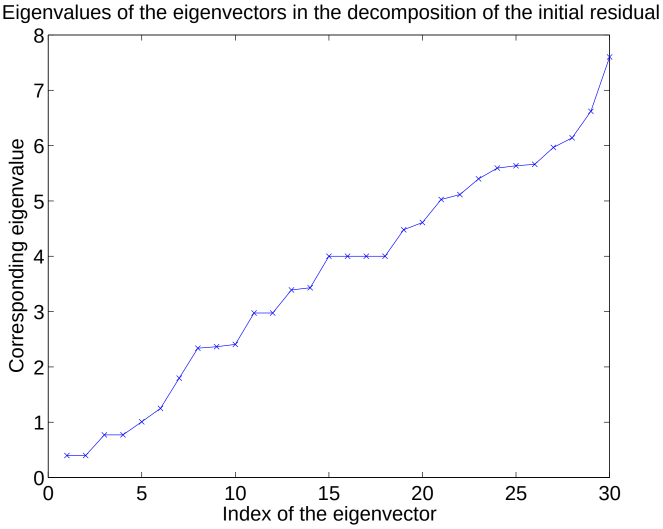


Figure 4.22.: Experiment 7: Corresponding eigenvalues for the eigenvectors in the decomposition of $\widetilde{\mathbf{y}}_0$.

discuss this relation in detail. Furthermore note that in this case we had to compute 27 *matrix-vector multiplications in order to construct the basis for the common space of approximation. Using the ordinary Lanczos algorithm unable to detect the common background of the single vectors we would end up with constructing the same space thrice. Every time we would need at least* 29 *matrix-vector multiplications.*

Experiment 7 *Here we use the same setup as in experiment 6. However, here* **A** *is a* 100×100 *discrete Laplacian. In experiment 4 we had a certain improvement as distinct eigenvectors shared almost the same eigenvalue. Here the situation is similar, but there is no real improvement.*

$n = 10;\ N = n^2;$
$A = gallery('poisson', n);$
$k = 30;$
$[V, D] = eig(A);$
$p = randperm(N);$
$y0t = [V(:, p(1:k)) * rand(k, 3);$

Every column of the initial block vector $\mathbf{y}_0$ *is a linear combination of* 30 *eigenvectors of* $\mathbf{A}$.

$$\mathbf{y}_0^{(i)} = \sum_{j=1}^{30} \gamma_j^{(i)} \mathbf{w}_j \qquad i = 1, 2, 3.$$

In Figure 4.22 the eigenvectors $\mathbf{w}_{15}, \ldots, \mathbf{w}_{18}$ *almost share a common eigenvalue. Each projection of the columns of the initial block into this eigenspace can be described by a* *linear combination of the same three orthonormal eigenvectors. Hence we expect that the smallest* $\mathbf{A}$*-invariant subspace containing all columns of* $\mathbf{y}_0$ *has dimension* 29*. This is perfectly matched by the experiment. The ordinary Lanczos process would profit a lot more from the fact that distinct eigenvectors share common eigenvalues. It would need approximately* 24 *iterations to construct an* $\mathbf{A}$*-invariant subspace for each corresponding column. This effect is explained in 4.*

The double dip in the singular values appears as a column of $\mathbf{y}_9$ *and two columns of* $\mathbf{y}_{10}$ *are indetermined.*

If all columns of the initial block vectors are in distinct spaces then we have the opposite case.

Experiment 8 *The matrix* $\mathbf{A}$ *is again a sparse* 100×100 *random matrix. Here we use a block vector* $\widetilde{\mathbf{y}}_0$ *where each column of the* 3 *columns is random linear combinations of* 20 *distinct eigenvectors of* $\mathbf{A}$*, that is the columns are orthogonal. Hence this* 60 *eigenvectors are an orthonormal basis for the* $\mathbf{A}$*-invariant subspace* $\mathcal{B}_{60}\left(\mathbf{A}, \widetilde{\mathbf{y}}_0\right)$*.*

$k = 20;$
$[V, D] = eig(A);$
$p = randperm(N);$
$y0t = [V(:, p(1:k)) * rand(k, 1), V(:, p(k+1:2*k)) * rand(k, 1), ...$

The next experiment is most important one in this thesis. It is central for understanding the need of deflation.

Experiment 9 *The matrix* $\mathbf{A}$ *is again a sparse* 100×100 *random matrix. Here we use a block vector* $\widetilde{\mathbf{y}}_0$ *where each of the first two columns is a random linear combinations of* 20 *distinct eigenvectors of* $\mathbf{A}$*. The third column is a linear combination of* 5 *other eigenvectors. Hence this* 45 *eigenvectors are an orthonormal basis for the* $\mathbf{A}$*-invariant subspace*

$$\mathcal{B}_{20}\left(\mathbf{A}, \widetilde{\mathbf{y}}_0\right) = \mathcal{K}_{20}\left(\mathbf{A}, \widetilde{\mathbf{y}}_0^{(1)}\right) \oplus \mathcal{K}_{20}\left(\mathbf{A}, \widetilde{\mathbf{y}}_0^{(2)}\right) \oplus \mathcal{K}_{5}\left(\mathbf{A}, \widetilde{\mathbf{y}}_0^{(3)}\right). \qquad (4.19)$$

$k = 20;$
$[V, D] = eig(A);$
$p = randperm(N);$
$y01 = V(:, p(1:k)) * rand(k, 1);$
$y02 = V(:, p(k+1:2*k)) * rand(k, 1);$
$y03 = V(:, p(2*k+1:2*k+5)) * rand(5, 1);$
$y0t = [y01, y02, y03];$

Constructing the block vectors $\mathbf{y}_0, \ldots, \mathbf{y}_4$ *we expect no problems. The situation is similar to Experiment 8. However, the Krylov subspace* $\mathcal{K}_5\left(\mathbf{A}, \widetilde{\mathbf{y}}_0^{(3)}\right)$ *is exhausted. The smallest*

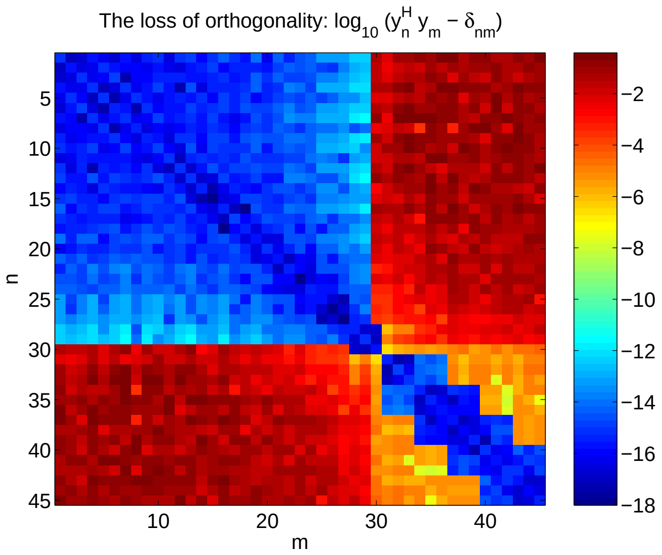


Figure 4.23.: Experiment 7: Colormap of the matrix $\mathbf{V} = \log |\mathbf{Y}_{15}^{\mathsf{H}} \mathbf{Y}_{15}^{\mathsf{H}} - \mathbf{I}_{15}|$. Note that $\mathbf{V}$ is a 45×45 matrix. The block vectors $\mathbf{y}_i$ where $i = 1, \ldots, 14$ have been constructed using the block Lanczos algorithm with exact deflation.

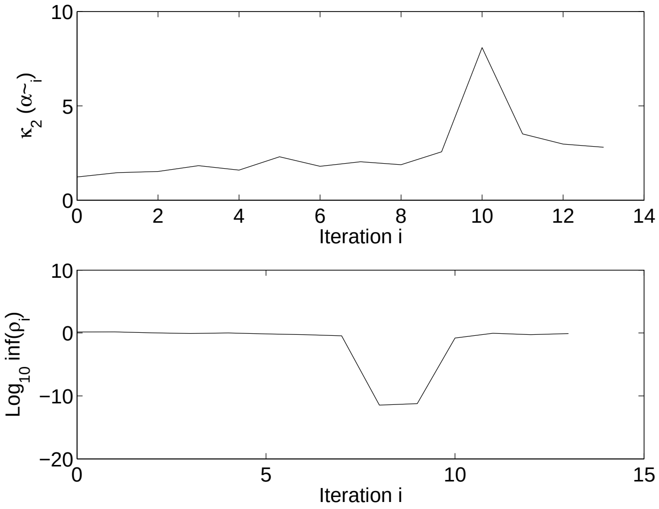


Figure 4.24.: Experiment 7: The double dip indicates two exhausted Krylov spaces.

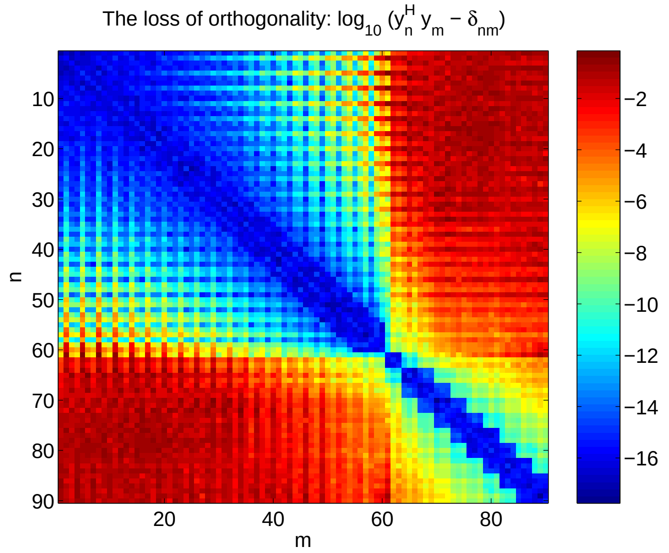


Figure 4.25.: Experiment 8: Colormap of the matrix $\mathbf{V} = \log|\mathbf{Y}_{30}^{\mathsf{H}}\mathbf{Y}_{30}^{\mathsf{H}} - \mathbf{I}_{30}|$. Note that $\mathbf{V}$ is a 90 × 90 matrix. The block vectors $\mathbf{y}_i$ where $i = 1, \ldots, 14$ have been constructed using the block Lanczos algorithm with exact deflation.

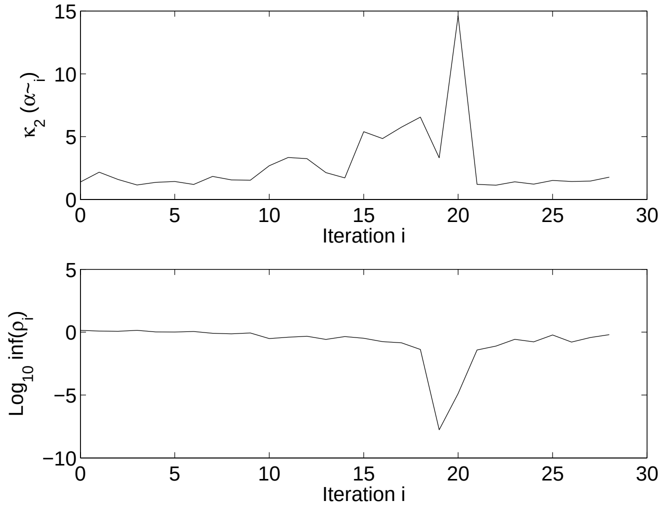


Figure 4.26.: Experiment 8: An exhausted block Krylov space is indicated.

eigenvalue of $\boldsymbol{\beta}_4$ is close to 10^{-10}. Proceeding without deflation we construct a highly indetermined vector in order to complete the block vector $\mathbf{y}_5$.

The hope is, that this vector does not disturb the Lanczos process, that is it does not influence to construction of the Krylov subspaces $\mathcal{K}_n\left(\mathbf{A}, \widetilde{\mathbf{y}}_0^{(1)}\right)$ and $\mathcal{K}_n\left(\mathbf{A}, \widetilde{\mathbf{y}}_0^{(2)}\right)$. In particular we expect that the corresponding columns in the block vector $\mathbf{y}_6$ remain orthogonal to all previous constructed vectors. However, this experiment shows that orthogonality is lost.

Let $\mathbf{y}_0, \ldots, \mathbf{y}_{n-2}, \hat{\mathbf{y}}_{n-1}$ as set of orthonormal block vectors. The block vectors $\mathbf{y}_0, \ldots, \mathbf{y}_{n-2}$ are of width s, $\hat{\mathbf{y}}_{n-1}$ is of width s_{n-1}. It is

$$\mathbf{y}_{n-1} = \begin{pmatrix} \hat{\mathbf{y}}_{n-1} & \mathbf{n} \end{pmatrix}$$

where $\mathbf{n}$ stands for an arbitrary normalized vector orthogonal to all columns of $\hat{\mathbf{y}}_{n-1}$. This is exactly the situation in experiment 9. It is

$$\widetilde{\mathbf{y}}_n = \begin{pmatrix} \mathbf{A}\hat{\mathbf{y}}_{n-1} & \mathbf{A}\mathbf{n} \end{pmatrix}.$$

The block Lanczos algorithm projects this block vector onto the block vectors $\mathbf{y}_{n-2}$ and $\mathbf{y}_{n-1}$. The experiment 9 seems to imply that orthogonality with respect to $\mathbf{y}_{n-2}$ is abruptly completely lost. Using the recurrence relation for $\mathbf{A}\mathbf{y}_{n-2}$ it is

$$\begin{aligned}
\mathbf{y}_{n-2}^{\mathsf{H}}\widetilde{\mathbf{y}}_n &= \mathbf{y}_{n-2}^{\mathsf{H}}\left(\mathbf{A}\mathbf{y}_{n-1} - \mathbf{y}_{n-1}\boldsymbol{\alpha}_{n-1} - \mathbf{y}_{n-2}\boldsymbol{\beta}_{n-2}^{\mathsf{H}}\right) \\
&= \left(\mathbf{A}\mathbf{y}_{n-2}\right)^{\mathsf{H}}\mathbf{y}_{n-1} - \mathbf{y}_{n-2}^{\mathsf{H}}\mathbf{y}_{n-1}\boldsymbol{\alpha}_{n-1} - \boldsymbol{\beta}_{n-2}^{\mathsf{H}} \\
&= \left(\mathbf{y}_{n-3}\boldsymbol{\beta}_{n-3}^{\mathsf{H}} + \mathbf{y}_{n-2}\boldsymbol{\alpha}_{n-2} + \mathbf{y}_{n-1}\boldsymbol{\beta}_{n-2}\right)^{\mathsf{H}}\mathbf{y}_{n-1} - \mathbf{y}_{n-2}^{\mathsf{H}}\mathbf{y}_{n-1}\boldsymbol{\alpha}_{n-1} - \boldsymbol{\beta}_{n-2}^{\mathsf{H}} \\
&= \boldsymbol{\beta}_{n-3}\mathbf{y}_{n-3}^{\mathsf{H}}\mathbf{y}_{n-1} + \boldsymbol{\alpha}_{n-2}^{\mathsf{H}}\mathbf{y}_{n-2}^{\mathsf{H}}\mathbf{y}_{n-1} - \mathbf{y}_{n-2}^{\mathsf{H}}\mathbf{y}_{n-1}\boldsymbol{\alpha}_{n-1} \\
&= \begin{pmatrix} \mathbf{0} & \ldots\mathbf{0} & \left(\boldsymbol{\beta}_{n-3}\mathbf{y}_{n-3}^{\mathsf{H}} + \boldsymbol{\alpha}_{n-2}^{\mathsf{H}}\mathbf{y}_{n-2}^{\mathsf{H}}\right)\mathbf{n} \end{pmatrix} + \begin{pmatrix} \mathbf{0} & \ldots\mathbf{0} & \mathbf{y}_{n-2}^{\mathsf{H}}\mathbf{n} \end{pmatrix}\boldsymbol{\alpha}_{n-1}
\end{aligned}$$

This yields that there is no column of $\widetilde{\mathbf{y}}_n$ is orthogonal to $\mathbf{y}_{n-2}^{\mathsf{H}}$ as

$$\begin{pmatrix} \mathbf{0} & \ldots\mathbf{0} & \mathbf{y}_{n-2}^{\mathsf{H}}\mathbf{n} \end{pmatrix}\boldsymbol{\alpha}_{n-1} = \mathbf{y}_{n-2}^{\mathsf{H}}\mathbf{n}\mathbf{n}^{\mathsf{H}}\left(\mathbf{A}\mathbf{y}_{n-1} - \mathbf{y}_{n-2}\boldsymbol{\beta}_{n-2}^{\mathsf{H}}\right)$$

is a matrix with rank 1 where each column is a multiple of $\mathbf{y}_{n-2}^{\mathsf{H}}\mathbf{n} \neq \mathbf{0}$. As the entries of the vector $\mathbf{y}_{n-2}^{\mathsf{H}}\mathbf{n}$ are of approximately same size we obtain two rather similar stripes in row 19 and 20 of Figure 4.28. The last experiment implies that it is necessary to use a deflation scheme. Otherwise the Lanczos algorithm will abruptly lose orthogonality when an indetermined vector is not deflated.

The above argument does not hold for the block Arnoldi algorithm by Gutknecht [14]. It also implies that it is possible to work with exact deflation only at the start of an iteration.

So far small singular values indicated an exhausted Krylovspace. There is also a second reason for a small singular value. Different Krylov spaces can "collide". A further experiment demonstrates this behavior:

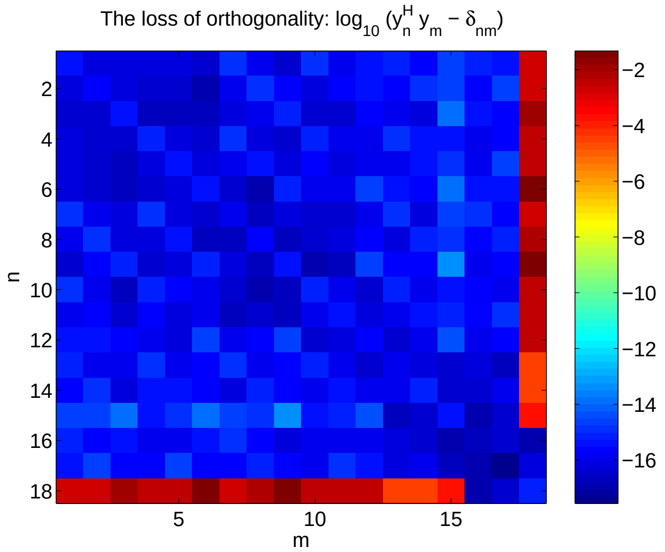


Figure 4.27.: Experiment 9: The vector corresponding to a singular value of approximately 10^{-10} is highly indetermined. It is not orthogonal to the vectors of the previous blocks. However, it is orthogonal to the two other vectors of the block vector $\mathbf{y}_5$.

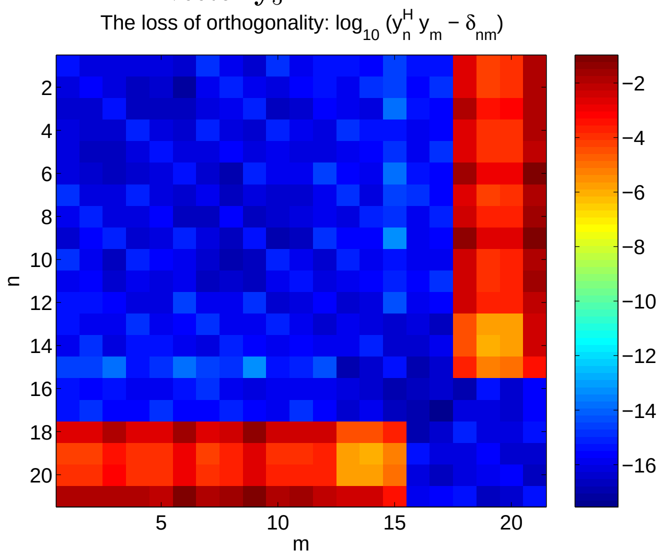


Figure 4.28.: Experiment 9: The block vector $\mathbf{y}_6$ is far away from being orthogonal to all previous blocks.

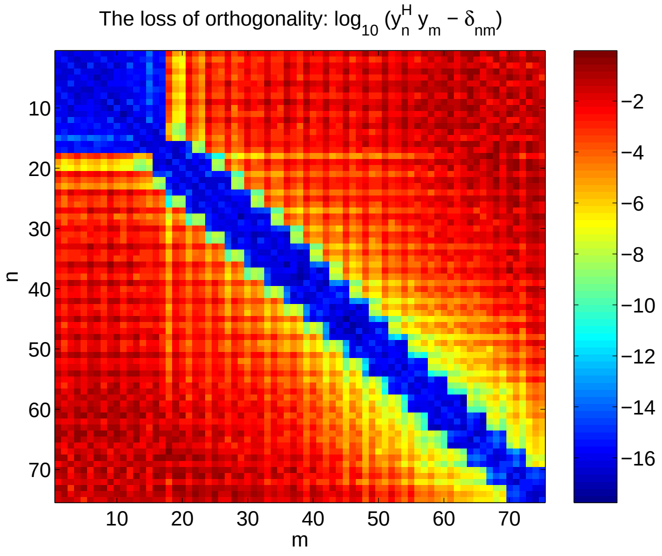


Figure 4.29.: Experiment 9: Colormap of the matrix $\mathbf{V} = \log|\mathbf{Y}_{25}^{\mathsf{H}}\mathbf{Y}_{25}^{\mathsf{H}} - \mathbf{I}_{25}|$. Orthogonality is completely after ignoring the exhausted Krylov space.

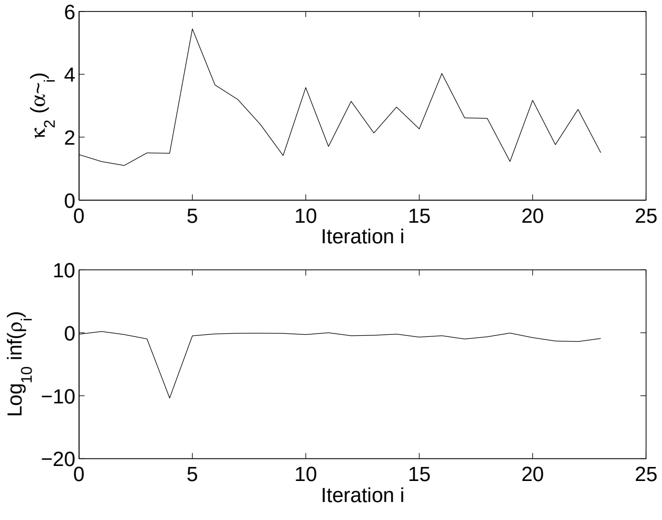


Figure 4.30.: Experiment 9: The peak indicates that the Krylov subspace $\mathcal{K}_5\left(\mathbf{A}, \widetilde{\mathbf{y}}_0^{(3)}\right)$ is exhausted.

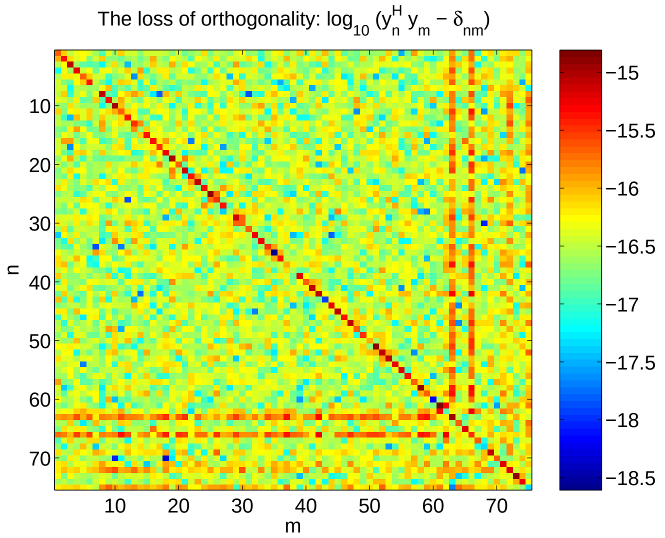


Figure 4.31.: Experiment 9: Colormap of the matrix $\mathbf{V} = \log|\mathbf{Y}_{25}^{\mathsf{H}}\mathbf{Y}_{25}^{\mathsf{H}} - \mathbf{I}_{25}|$ using the block Arnoldi process without deflation.

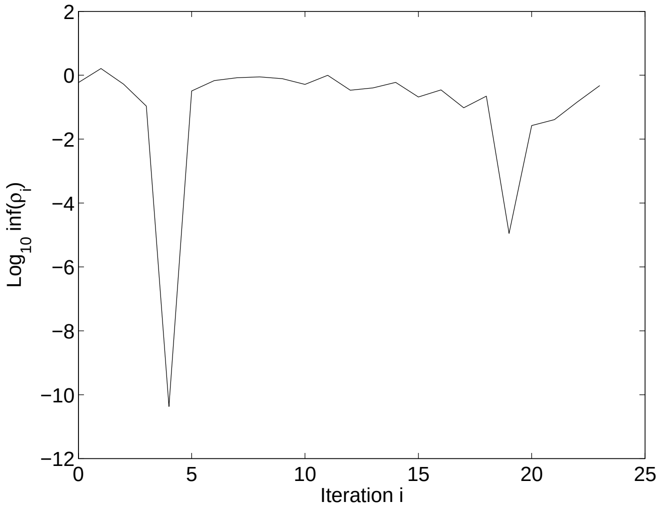


Figure 4.32.: Experiment 9: The two dips perfectly match the expectation of indetermined vectors in $\mathbf{y}_5$ and $\mathbf{y}_{20}$.

Experiment 10 *The matrix* $\mathbf{A}$ *is a sparse* 100×100 *random matrix and let* $\mathbf{w}_1$ *a random linear combination of* 20 *distinct eigenvectors of* $\mathbf{A}$*. As a second right-hand side we use the vector* $\mathbf{w}_2 = \mathbf{A}^5 \mathbf{w}_1$*.*

```
k = 20;
[V, D] = eig(A);
p = randperm(N);
w1 = V(:, p(1 : k)) * (rand(k, 1));
w2 = A * (A * (A * (A * y0)));
y0t = [w1, w2];
```

It is

$$\mathcal{K}_{20}\left(\mathbf{A}, \mathbf{w}_1\right) = \mathcal{K}_{20}\left(\mathbf{A}, \mathbf{w}_2\right)$$

as both right-hand sides are linear combinations of the same 20 *eigenvectors. As* $\mathbf{w}_2 \in \mathcal{K}_5\left(\mathbf{A}, \mathbf{w}_1\right)$ *it is not possible to orthogonalize the* 5 *basis vectors of* $\mathcal{K}_5\left(\mathbf{A}, \mathbf{w}_1\right)$ *with respect to* $\mathbf{w}_2$*. We expect a small singular value for block* $\mathbf{y}_4$*. Proceeding without deflation results in similar problems as in the experiment 9.*

4.4.3. Inexact deflation

Definition. Using a tolerance $\mathrm{tol} \geq 0$ in the block Lanczos process the set of block vectors constructed by the Hermitian block Lanczos algorithm $\mathbf{y}_0, \mathbf{y}_1 \ldots \mathbf{y}_{n-1}$ spans for every $n = 1, 2 \ldots$ the space

$$\mathcal{B}^{\square}_{n,\mathrm{tol}}\left(\mathbf{A}, \mathbf{y}_0\right) = \text{block span}\left\{\mathbf{y}_0, \mathbf{y}_1 \ldots \mathbf{y}_{n-1}\right\}$$

and

$$\mathcal{B}_{n,\mathrm{tol}}\left(\mathbf{A}, \mathbf{y}_0\right) = \mathsf{span}\left\{\mathbf{y}_0^{(1)}, \ldots, \mathbf{y}_0^{(\mathsf{s}_0)}, \mathbf{y}_1^{(1)}, \ldots, \mathbf{y}_{n-1}^{(\mathsf{s}_{n-1})}\right\}$$

▲

Corollary 9 *In exact arithmetic*

$$\mathcal{B}^{\square}_{n}\left(\mathbf{A}, \mathbf{y}_0\right) = \mathcal{B}^{\square}_{n,0}\left(\mathbf{A}, \mathbf{y}_0\right)$$

In non exact arithmetic both spaces could be rather distinct due to the loss of orthogonality. The space constructed by the algorithm using exact deflation might contain basis vectors not lying in the space $\mathcal{B}^{\square}_{n}\left(\mathbf{A}, \mathbf{y}_0\right)$. On the other hand the algorithm might fail to match all directions in $\mathcal{B}^{\square}_{n}\left(\mathbf{A}, \mathbf{y}_0\right)$. It is worth noting:

Corollary 10

$$\mathcal{B}^{\square}_{n,\mathrm{tol}_1}\left(\mathbf{A}, \mathbf{y}_0\right) \subset \mathcal{B}^{\square}_{n,\mathrm{tol}_2}\left(\mathbf{A}, \mathbf{y}_0\right)$$

if $\mathrm{tol}_1 \geq \mathrm{tol}_2$*. The same holds for* $\mathcal{B}_{n,\mathrm{tol}}\left(\mathbf{A}, \mathbf{y}_0\right)$*.*

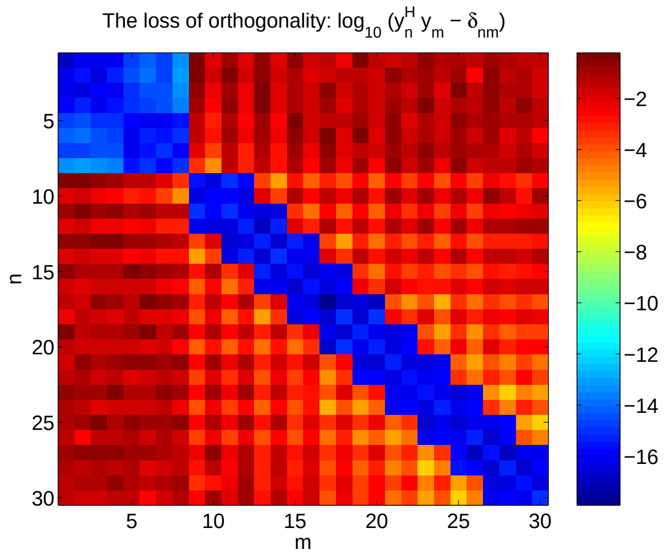


Figure 4.33.: Experiment 10: Colormap of the matrix $\mathbf{V} = \log|\mathbf{Y}_{15}^{\mathsf{H}}\mathbf{Y}_{15}^{\mathsf{H}} - \mathbf{I}_{15}|$. For constructing the block Krylov basis the block Lanczos algorithm with exact deflation has been used.

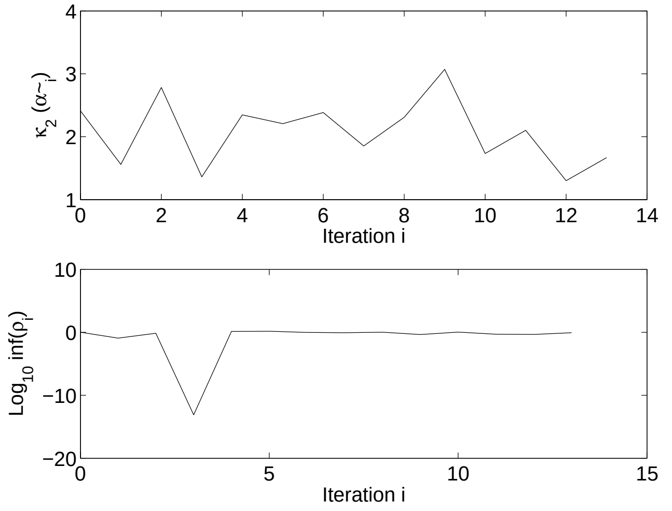


Figure 4.34.: Experiment 10: The peak indicates here a collision of two Krylov subspaces.

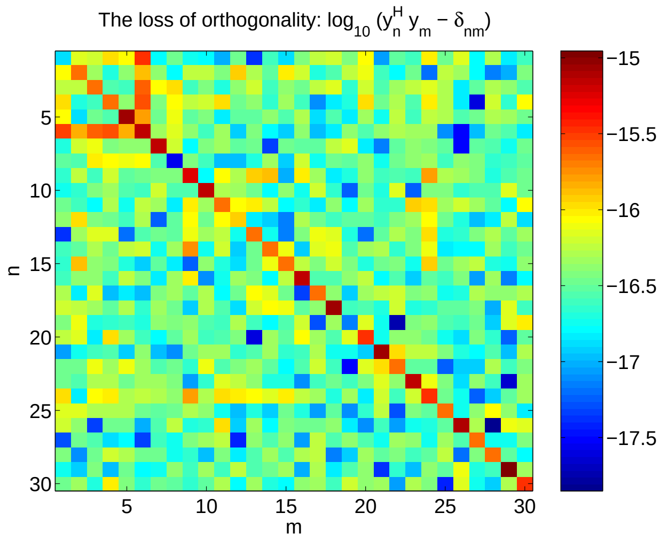


Figure 4.35.: Experiment 10: Colormap of the matrix $\mathbf{V} = \log|\mathbf{Y}_{15}^{\mathsf{H}}\mathbf{Y}_{15}^{\mathsf{H}} - \mathbf{I}_{15}|$. For constructing the block Krylov basis the block Arnoldi algorithm with exact deflation has been used.

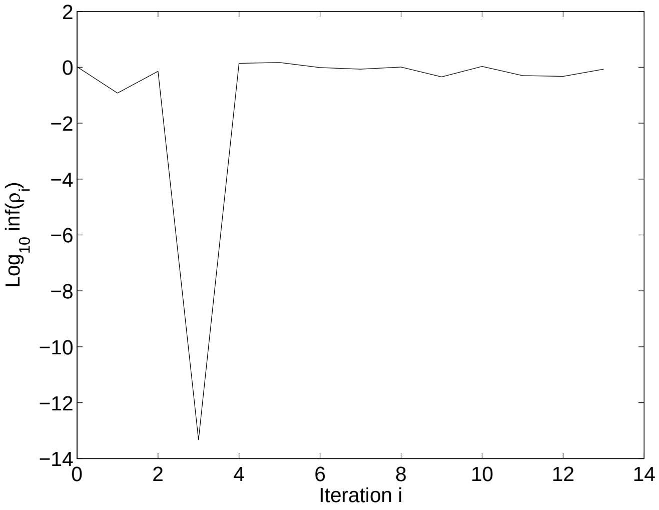


Figure 4.36.: Experiment 10: The peak indicates that the Krylov subspace $\mathcal{K}_5\left(\mathbf{A}, \widetilde{\mathbf{y}}_0^{(3)}\right)$ is exhausted.

Inexact deflation occurs if without loss of generality the vector $\widetilde{\mathbf{y}}_n^{(1)}$ is linearly or almost linearly dependent on the the columns of

$$\widetilde{\mathbf{y}}'_n = \left(\begin{array}{ccc} \widetilde{\mathbf{y}}_n^{(2)} & \dots & \widetilde{\mathbf{y}}_n^{(\mathsf{s}_n)} \end{array} \right).$$

Freund [7] claims that a vector $\widetilde{\mathbf{y}}_n^{(1)}$ being linearly or almost linearly dependent on columns of $\widetilde{\mathbf{y}}'_n$ implies that all vectors $\mathbf{A}^k\widetilde{\mathbf{y}}_n^{(1)}$, $k \geq 0$ are also linearly dependent or almost linearly dependent on the columns of $\widetilde{\mathbf{y}}'_n, \mathbf{A}\widetilde{\mathbf{y}}'_n, \dots, \mathbf{A}^k\widetilde{\mathbf{y}}'_n$. It is assumed that almost linearly dependent vectors share a common dominating eigenvector, that is the power iteration starting with almost linearly dependent vectors would converge against the same eigenvector. This is a reasonable argument in practice nevertheless one has to be aware as the following example illustrates:

Example. Let

$$\mathbf{A} = \left(\begin{array}{cc} 1 & 0 \\ 0 & 1000 \end{array} \right), \qquad \widetilde{\mathbf{y}}_0 = \left(\begin{array}{cc} 1 & 1 \\ 0.001 & 0 \end{array} \right).$$

The column vectors of $\widetilde{\mathbf{y}}_0$ are almost linearly dependent, i.e. the method might deflate the first column and proceed with

$$\mathbf{y}_0 = \left(\begin{array}{c} 1 \\ 0 \end{array} \right).$$

Due to the above argument

$$\mathbf{A} \left(\begin{array}{c} 1 \\ 0.001 \end{array} \right) = \left(\begin{array}{c} 1 \\ 1 \end{array} \right)$$

should be almost linearly dependent on the columns of $\mathbf{y}_0, \mathbf{A}\mathbf{y}_0 = \mathbf{y}_0$. Here $\mathbf{y}_0$ is an eigenvector of $\mathbf{A}$, in particular $\mathbf{y}_0$ has no component in the direction of the second eigenvector. Although the columns of $\widetilde{\mathbf{y}}_0$ are almost linearly dependent they do not share a common dominating eigenvector. ◊

It is not practicable to react within the Lanczos process. A reliable solver as block MinRes should instead cope with that problem if convergence cannot be observed.

Experiment 11 *Here we repeat the Experiment 9 with a deflation tolerance of* tol $= 10^{-5}$. *Hence the indetermined vector in* $\mathbf{y}_5$ *will be deflated.*

Experiment 12 *Here we repeat the Experiment 10 with a deflation tolerance of* tol $= 10^{-5}$ *Hence we expect that the indetermined vector in* $\mathbf{y}_4$ *will be deflated.*

4.4.4. A corrected Lanczos relationship

The fundamental Lanczos relationship (4.16) is valid as long as we assume $\boldsymbol{\rho}_i^{\Delta} = \mathbf{0}$ or empty for all $i = 1, \dots, \bar{\nu}^{\Box}(\mathbf{A}, \mathbf{y}_0) - 1$. In the case of deflation $\|\boldsymbol{\rho}_i^{\Delta}\|_F < O(\text{tol})$. Assuming tol > 0 the argument we used in the proof does not hold anymore. Instead we have to regard the deflated block vectors more precisely. Let

$$\mathbf{Y}_n^{\Delta} :\equiv \left(\begin{array}{cccc} \mathbf{y}_0^{\Delta} & \mathbf{y}_1^{\Delta} & \dots & \mathbf{y}_{n-1}^{\Delta} \end{array} \right) \tag{4.20}$$

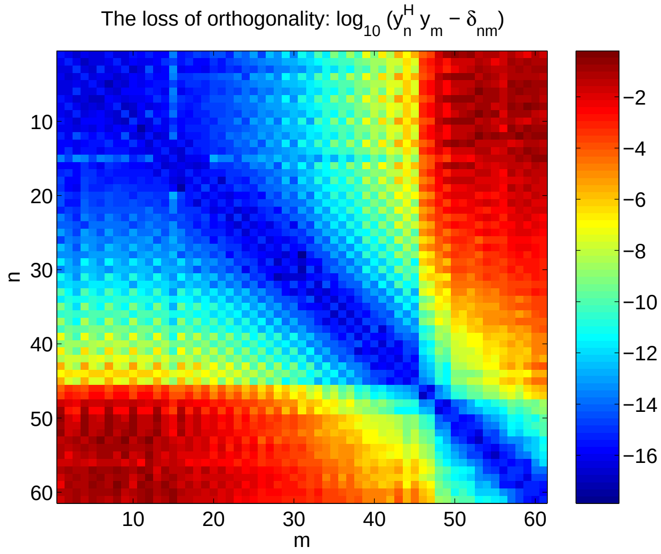


Figure 4.37.: Experiment 11: Block Lanczos with a deflation tolerance of $\mathrm{tol} = 10^{-5}$.

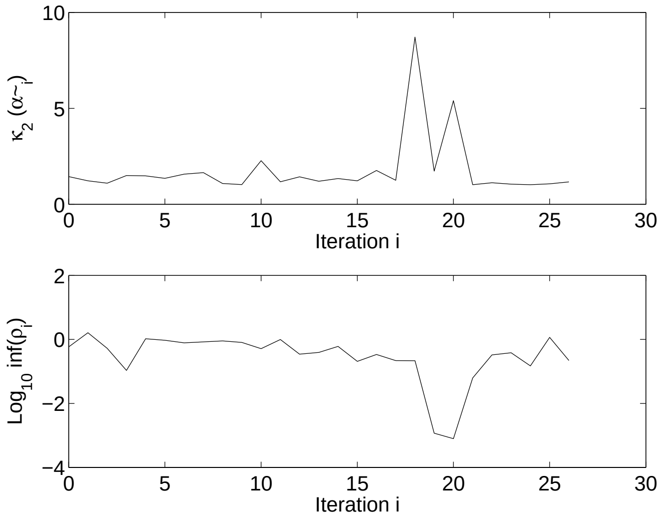


Figure 4.38.: Experiment 11: Compare with Figure 4.30.

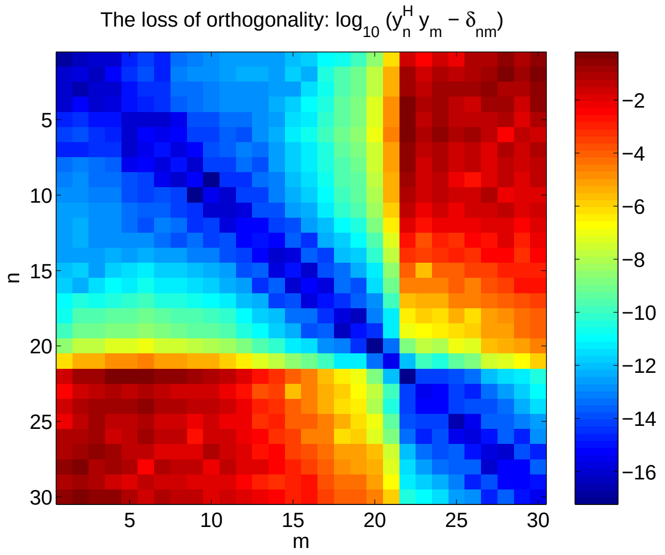


Figure 4.39.: Experiment 12: Block Lanczos with a deflation tolerance of tol $= 10^{-5}$.

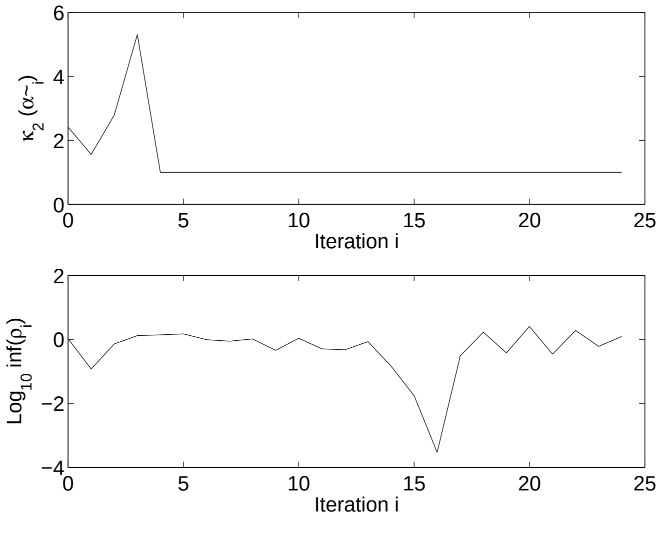


Figure 4.40.: Experiment 12: Compare with Figure 4.34.

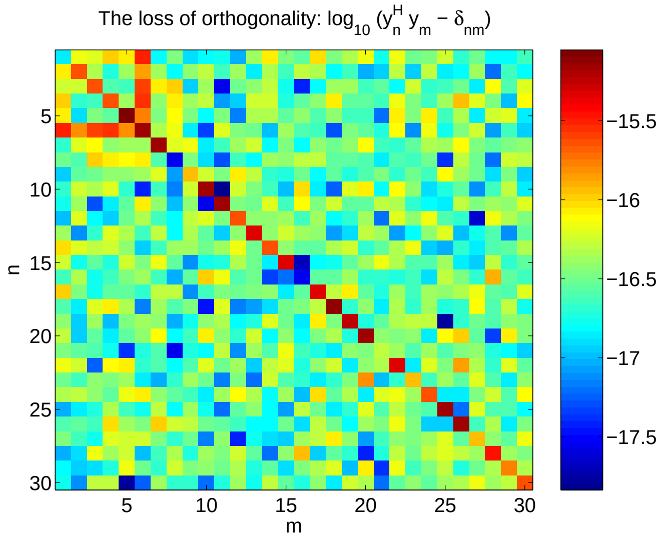


Figure 4.41.: Experiment 12: Block Arnoldi with a deflation tolerance of $\mathrm{tol} = 10^{-5}$.

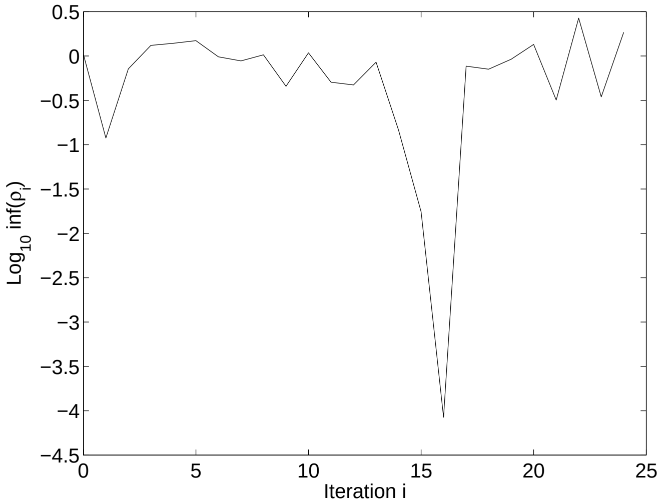


Figure 4.42.: Experiment 12: Compare with Figure 4.36.

where $\mathbf{y}_0^\Delta$ is a $N \times (\mathsf{s} - \mathsf{s}_0)$ matrix. The number of columns of $\mathbf{y}_i^\Delta$ is the number of deflations in the iteration i of the block Lanczos process. We denote it with d_i, i.e. $\mathsf{d}_0 = \mathsf{s} - \mathsf{s}_0$ and $\mathsf{d}_i = \mathsf{s}_i - \mathsf{s}_{i+1}$ if $i > 0$. Let

$$\underline{\mathbf{T}}_n^\Delta :\equiv \begin{pmatrix} \mathbf{0}_{\mathsf{d}_0 \times \mathsf{s}_0} & \mathbf{0}_{\mathsf{d}_0 \times \mathsf{s}_1} & \cdots & \mathbf{0}_{\mathsf{d}_0 \times \mathsf{s}_{n-1}} \\ \boldsymbol{\beta}_0^\Delta & \mathbf{0}_{\mathsf{d}_1 \times \mathsf{s}_1} & \cdots & \mathbf{0}_{\mathsf{d}_1 \times \mathsf{s}_{n-1}} \\ & \boldsymbol{\beta}_1^\Delta & \ddots & \vdots \\ & & \ddots & \mathbf{0}_{\mathsf{d}_{n-1} \times \mathsf{s}_{n-1}} \\ & & & \boldsymbol{\beta}_{n-1}^\Delta \end{pmatrix}. \tag{4.21}$$

Note that the blocks of $\underline{\mathbf{T}}_n^\Delta$ are as wide as those of $\underline{\mathbf{T}}_n$, but typically much less high.

Theorem 11

$$\mathbf{A}\mathbf{Y}_n = \mathbf{Y}_{n+1}\underline{\mathbf{T}}_n + \mathbf{Y}_{n+1}^\Delta \underline{\mathbf{T}}_n^\Delta. \tag{4.22}$$

Recall that $\mathbf{Y}_{n+1}$ has orthonormal columns, but, in general, those of $\mathbf{Y}_{n+1}^\Delta$ are not orthogonal, just of 2-norm 1.

In case of exact deflation only, $\underline{\mathbf{T}}_n^\Delta = \mathbf{0}$, but $\underline{\mathbf{T}}_n$ still has blocks of different size.

We denote by $\tilde{\mathsf{d}}_n$ the number of non-vanishing blocks $\boldsymbol{\beta}_0^\Delta, \ldots, \boldsymbol{\beta}_{n-1}^\Delta$, i.e. $\tilde{\mathsf{d}}_n$ counts the number of iterations in the block Lanczos process with deflation. Then

$$\|\underline{\mathbf{T}}_n^\Delta\|_F^2 = \sum_{i=0}^{n-1} \|\boldsymbol{\beta}_i^\Delta\|_F^2 = \tilde{\mathsf{d}}_n O(\mathrm{tol})^2 \tag{4.23}$$

We denote by $\mathsf{d}_n^\Box$ the number of deflations up to step n, i.e.

$$\mathsf{d}_n^\Box = \sum_{i=0}^{n-1} \mathsf{d}_i \tag{4.24}$$

This is analogous to the definition of $\mathsf{s}_n^\Box$. As a consequence of Lemma 5 we note

Lemma 12

$$\|\mathbf{Y}_{n+1}^\Delta \underline{\mathbf{T}}_n^\Delta\|_F \le \sqrt{\mathsf{d}_{n+1}^\Box \tilde{\mathsf{d}}_n}\, O(\mathrm{tol}) \le \mathsf{s}\, O(\mathrm{tol}). \tag{4.25}$$

4.5. A general block iteration

The general block iteration is

$$\mathbf{x}_n^{(i)} \in \mathbf{x}_0^{(i)} + \mathcal{B}_{n,\mathrm{tol}}\left(\mathbf{A}, \mathbf{r}_0\right). \tag{4.26}$$

This condition implies a simple scheme

$$\mathbf{x}_n^{(i)} = \mathbf{x}_0^{(i)} + \mathbf{Y}_n \mathbf{k}_n^{(i)} \tag{4.27}$$

and

$$\mathbf{r}_n^{(i)} = \mathbf{r}_0^{(i)} - \mathbf{A}\mathbf{Y}_n \mathbf{k}_n^{(i)} \tag{4.28}$$

in which $\mathbf{k}_n^{(i)}$ contains the coordinates of $\mathbf{x}_n^{(i)} - \mathbf{x}_0^{(i)}$ in terms of the block Lanczos basis. Again the freedom to choose those coordinates results in a variety of different block Krylov methods treated in the chapters 7 and 8 . In block notation we formulate (4.27) and (4.28) as

$$\mathbf{x}_n = \mathbf{x}_0 + \mathbf{Y}_n \mathbf{k}_n \tag{4.29}$$

and

$$\mathbf{r}_n = \mathbf{r}_0 - \mathbf{A}\mathbf{Y}_n \mathbf{k}_n. \tag{4.30}$$

5. A block QR decomposition

> Any orthogonal matrix can be written as the product of reflector matrices. Thus the class of reflections is rich enough for all occasions and yet each member it charachterized by a single vector which serves to describe its mirror.
>
> *(Beresford Parlett)*

For MinRes and SymmLQ it is essential to compute a QR decomposition of the matrix $\underline{\mathbf{T}}_n$ which is done by an update scheme constructing in every step a single Givens rotation. Here we factorize instead a block tridiagonal matrix $\underline{\mathbf{T}}_n\mathbf{P}_n$. Still we are using an update scheme, but now a single Givens rotation is not enough. An idea by Gutknecht is to apply in every step a set of complex Householder reflections. Some implementation details are given and we compare the method with an algorithm based on Givens rotations by Freund [7].

5.1. An update scheme for the tridiagonal block QR decomposition

Let $\underline{\mathbf{T}}_n\mathbf{P}_n = \mathbf{Q}_{n+1}\underline{\mathbf{R}}_n^{\mathrm{MR}}$ where $\mathbf{Q}_{n+1}$ is an unitary $\mathsf{s}_{n+1}^{\square} \times \mathsf{s}_{n+1}^{\square}$ matrix and $\underline{\mathbf{R}}_n^{\mathrm{MR}}$ is an upper block banded triangular $\mathsf{s}_{n+1}^{\square} \times \mathsf{s}_n^{\square}$ matrix with full column rank. The block tridiagonal matrix $\underline{\mathbf{T}}_n\mathbf{P}_n$ is

$$\underline{\mathbf{T}}_n\mathbf{P}_n = \left(\begin{array}{ccccc}
\boldsymbol{\alpha}_0\boldsymbol{\pi}_1 & \boldsymbol{\beta}_0^{\mathsf{H}}\boldsymbol{\pi}_2 & & & \\
\boldsymbol{\beta}_0\boldsymbol{\pi}_1 & \boldsymbol{\alpha}_1\boldsymbol{\pi}_2 & \boldsymbol{\beta}_1^{\mathsf{H}}\boldsymbol{\pi}_3 & & \\
 & \boldsymbol{\beta}_1\boldsymbol{\pi}_2 & \ddots & \ddots & \\
 & & \ddots & \ddots & \boldsymbol{\beta}_{n-2}^{\mathsf{H}}\boldsymbol{\pi}_n \\
 & & & \boldsymbol{\beta}_{n-2}\boldsymbol{\pi}_{n-1} & \boldsymbol{\alpha}_{n-1}\boldsymbol{\pi}_n \\
\hline
 & & & & \boldsymbol{\beta}_{n-1}\boldsymbol{\pi}_n
\end{array}\right) \tag{5.1}$$

where:

$\boldsymbol{\alpha}_i\boldsymbol{\pi}_{i+1}$ is an $\mathsf{s}_i \times \mathsf{s}_i$ block vector and

$\boldsymbol{\beta}_i\boldsymbol{\pi}_{i+1}$ is an $\mathsf{s}_{i+1} \times \mathsf{s}_i$ upper trapezoidal block vector.

The matrix $\underline{\mathbf{R}}_n^{\mathrm{MR}}$ has the form

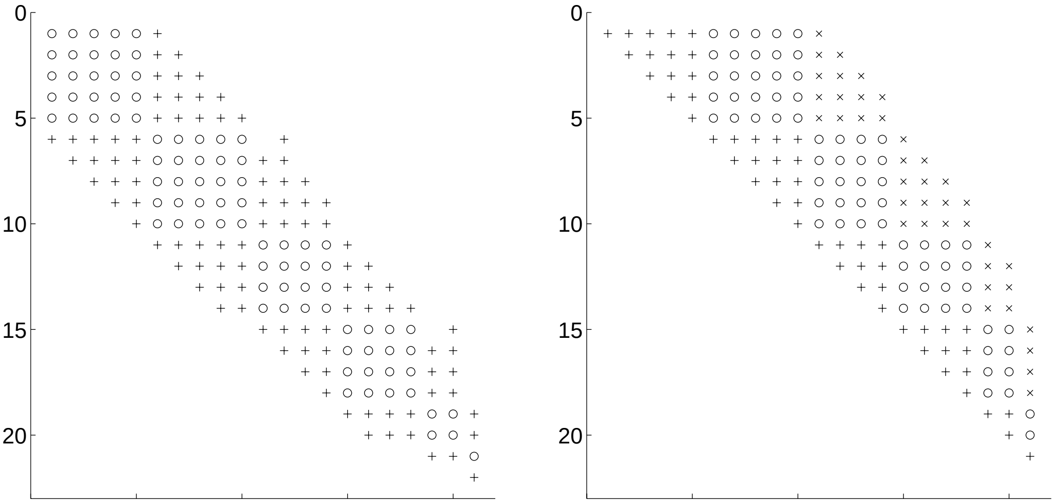
$$\underline{\mathbf{R}}_n^{\mathrm{MR}} = \left(\begin{array}{ccccc} \widetilde{\boldsymbol{\alpha}}_0 & \widetilde{\boldsymbol{\beta}}_0 & \widetilde{\boldsymbol{\gamma}}_0 & & \\ & \widetilde{\boldsymbol{\alpha}}_1 & \widetilde{\boldsymbol{\beta}}_1 & \ddots & \\ & & \ddots & \ddots & \widetilde{\boldsymbol{\gamma}}_{n-3} \\ & & & \ddots & \widetilde{\boldsymbol{\beta}}_{n-2} \\ & & & & \widetilde{\boldsymbol{\alpha}}_{n-1} \\ \hline \multicolumn{5}{c}{\mathbf{0}_{\mathsf{s}_n \times \mathsf{s}_n^{\square}}} \end{array}\right) \tag{5.2}$$

where:

$\widetilde{\boldsymbol{\alpha}}_i$ is an $\mathsf{s}_i \times \mathsf{s}_i$ upper triangular block vector with full rank,

$\widetilde{\boldsymbol{\beta}}_i$ is an $\mathsf{s}_i \times \mathsf{s}_{i+1}$ block vector and

$\widetilde{\boldsymbol{\gamma}}_i$ is an $\mathsf{s}_i \times \mathsf{s}_{i+2}$ lower trapezoidal block vector.



(a) *The block tridiagonal structure of a matrix* $\underline{\mathbf{T}}_n\mathbf{P}_n$.

(b) *The corresponding block structure of the matrix* $\underline{\mathbf{R}}_n^{\mathrm{MR}}$.

Again we determine the unitary matrix $\mathbf{Q}_{n+1}$ in its factored form. With

$$\mathbf{Q}_1 = \mathbf{I}_{\mathsf{s}_0} \tag{5.3}$$

we introduce the recurrence relation

$$\mathbf{Q}_{n+1} :\equiv \left(\begin{array}{c|c} \mathbf{Q}_n & \mathbf{0}_{\mathsf{s}_n^{\square} \times \mathsf{s}_n} \\ \hline \mathbf{0}_{\mathsf{s}_n \times \mathsf{s}_n^{\square}} & \mathbf{I}_{\mathsf{s}_n} \end{array}\right) \mathbf{U}_n = \text{block diag } \left(\mathbf{Q}_n, \mathbf{I}_{\mathsf{s}_n}\right) \mathbf{U}_n \tag{5.4}$$

where

$$\mathbf{U}_n :\equiv \left(\begin{array}{c|c} \mathbf{I}_{\mathsf{s}_{n-1}^{\square}} & \mathbf{0}_{\mathsf{s}_{n-1}^{\square} \times (\mathsf{s}_{n-1} + \mathsf{s}_n)} \\ \hline \mathbf{0}_{(\mathsf{s}_{n-1} + \mathsf{s}_n) \times \mathsf{s}_{n-1}^{\square}} & \hat{\mathbf{U}}_n \end{array}\right) = \text{block diag } \left(\mathbf{I}_{\mathsf{s}_{n-1}^{\square}}, \hat{\mathbf{U}}_n\right). \tag{5.5}$$

Here $\hat{\mathbf{U}}_n$ is an unitary $(\mathsf{s}_{n-1}+\mathsf{s}_n)\times(\mathsf{s}_{n-1}+\mathsf{s}_n)$ matrix. Given a sequence of unitary transformations $\hat{\mathbf{U}}_1,\ldots\hat{\mathbf{U}}_n$ it is possible to compute $\mathbf{Q}_{n+1}$ with a simple scheme. If the matrix $\mathbf{Q}_n$ has the form

$$\mathbf{Q}_n = \begin{pmatrix} \mathbf{q}_0 & \mathbf{q}_1 & \ldots & \mathbf{q}_{n-2} & \widetilde{\mathbf{q}}_{n-1} \end{pmatrix}$$

where $\mathbf{q}_i \in \mathbb{C}^{\mathsf{s}_n^{\square}\times\mathsf{s}_i}$ and

$$\hat{\mathbf{U}}_n = \begin{pmatrix} \hat{\mathbf{U}}_{n,u} \\ \hat{\mathbf{U}}_{n,d} \end{pmatrix} = \begin{pmatrix} \hat{\mathbf{U}}_{n,u,l} & \hat{\mathbf{U}}_{n,u,r} \\ \hat{\mathbf{U}}_{n,d,l} & \hat{\mathbf{U}}_{n,d,r} \end{pmatrix} \tag{5.6}$$

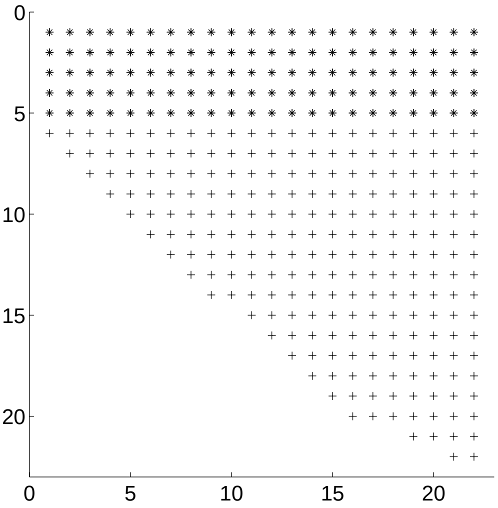
where:

$\hat{\mathbf{U}}_{n,u}$	is an $\mathsf{s}_{i-1}\times\mathsf{s}_{i-1}+\mathsf{s}_i$ matrix.
$\hat{\mathbf{U}}_{n,d}$	is an $\mathsf{s}_{i}\times\mathsf{s}_{i-1}+\mathsf{s}_i$ matrix.
$\hat{\mathbf{U}}_{n,u,l}$	is an $\mathsf{s}_{i-1}\times\mathsf{s}_{i-1}$ matrix.
$\hat{\mathbf{U}}_{n,u,r}$	is an $\mathsf{s}_{i-1}\times\mathsf{s}_{i}$ matrix.
$\hat{\mathbf{U}}_{n,d,l}$	is an $\mathsf{s}_{i}\times\mathsf{s}_{i-1}$ matrix.
$\hat{\mathbf{U}}_{n,d,r}$	is an $\mathsf{s}_{i}\times\mathsf{s}_{i}$ matrix.

then

$$\mathbf{Q}_{n+1} = \begin{pmatrix} \mathbf{q}_0 & \mathbf{q}_1 & \cdots & \mathbf{q}_{n-2} & \widetilde{\mathbf{q}}_{n-1}\hat{\mathbf{U}}_{n,u,l} & \widetilde{\mathbf{q}}_{n-1}\hat{\mathbf{U}}_{n,u,r} \\ & & & & \hat{\mathbf{U}}_{n,d,l} & \hat{\mathbf{U}}_{n,d,r} \end{pmatrix} \tag{5.7}$$

In particular the matrix $\mathbf{Q}_{n+1}$ has a trapezoidal structure. The update scheme (5.7) yields the following algorithm:



(c) *The block trapezoidal structure of the matrix* $\mathbf{Q}_{n+1}$. *The entries marked with stars have a central role in block* MinRes

ALGORITHM 6 (EXPLICIT CONSTRUCTION OF $\mathbf{Q}_{n+1}$) .
Let $\hat{\mathbf{U}}_1, \dots \hat{\mathbf{U}}_n$ *the constructed sequence of unitary matrices where* $\hat{\mathbf{U}}_i$ *is of order* $\mathsf{s}_{i-1}+\mathsf{s}_i$. *Let* $\mathbf{Q}_{n+1} = \mathbf{I}_{\mathsf{s}_{n+1}^{\square}}$. *Then, for* $i = 1, \dots, n$*:*

- *Apply the upper part of* $\hat{\mathbf{U}}_i$

$$\mathbf{Q}_{n+1}\left(1:\mathsf{s}_i^{\square}, \mathsf{s}_{i-1}^{\square}+1:\mathsf{s}_{i+1}^{\square}\right) = \mathbf{Q}_{n+1}\left(1:\mathsf{s}_i^{\square}, \mathsf{s}_{i-1}^{\square}+1:\mathsf{s}_i^{\square}\right)\hat{\mathbf{U}}_{i,u} \tag{5.8}$$

- *Apply the lower part of* $\hat{\mathbf{U}}_i$

$$\mathbf{Q}_{n+1}\left(\mathsf{s}_i^{\square}+1:\mathsf{s}_{i+1}^{\square}, \mathsf{s}_{i-1}^{\square}+1:\mathsf{s}_{i+1}^{\square}\right) = \hat{\mathbf{U}}_{i,d} \tag{5.9}$$

The multiplication by $\mathbf{Q}_n^{\mathsf{H}}$ annihilates all subdiagonal elements except those below the diagonal of the last s_{n-1} columns, or if you regard $\underline{\mathbf{T}}_n\mathbf{P}_n$ as a matrix with n block columns it does not annihilate the subdiagonal elements of the blocks in the last block column, i.e.

$$\text{block diag}\left(\mathbf{Q}_n^{\mathsf{H}}, \mathbf{I}_{\mathsf{s}_n}\right)\underline{\mathbf{T}}_n\mathbf{P}_n = \left(\begin{array}{cccccc}
\widetilde{\boldsymbol{\alpha}}_0 & \widetilde{\boldsymbol{\beta}}_0 & \widetilde{\boldsymbol{\gamma}}_0 & & & \\
 & \widetilde{\boldsymbol{\alpha}}_1 & \widetilde{\boldsymbol{\beta}}_1 & \ddots & & \\
 & & \ddots & \ddots & \ddots & \\
 & & & \ddots & \ddots & \widetilde{\boldsymbol{\gamma}}_{n-3} \\
 & & & & \widetilde{\boldsymbol{\alpha}}_{n-2} & \widetilde{\boldsymbol{\beta}}_{n-2} \\
 & & & & & \boldsymbol{\mu}_n \\
\hline
 & & & & & \boldsymbol{\beta}_{n-1}\boldsymbol{\pi}_n
\end{array}\right).$$

We regard only the entries in the last block column, respectively the entries in the last s_{n-1} columns:

$$\begin{pmatrix} \widetilde{\boldsymbol{\gamma}}_{n-3} \\ \widetilde{\boldsymbol{\beta}}_{n-2} \\ \boldsymbol{\mu}_n \end{pmatrix} = \text{block diag}\left(\mathbf{I}_{\mathsf{s}_{n-3}}, \hat{\mathbf{U}}_{n-1}^{\mathsf{H}}\right)\text{block diag}\left(\hat{\mathbf{U}}_{n-2}^{\mathsf{H}}, \mathbf{I}_{\mathsf{s}_{n-1}}\right)\begin{pmatrix} \mathbf{0}_{\mathsf{s}_{n-3}\times\mathsf{s}_{n-1}} \\ \boldsymbol{\beta}_{n-2}^{\mathsf{H}}\boldsymbol{\pi}_n \\ \boldsymbol{\alpha}_{n-1}\boldsymbol{\pi}_n \end{pmatrix}. \tag{5.10}$$

Annihilating all subdiagonal elements we have to construct an unitary matrix $\mathbf{U}_n'$ such that

$$\begin{pmatrix} \boldsymbol{\mu}_n \\ \boldsymbol{\beta}_{n-1}\boldsymbol{\pi}_n \end{pmatrix} = \hat{\mathbf{U}}_n\begin{pmatrix} \widetilde{\boldsymbol{\alpha}}_{n-1} \\ \mathbf{0}_{\mathsf{s}_n\times\mathsf{s}_{n-1}} \end{pmatrix} \tag{5.11}$$

where $\widetilde{\boldsymbol{\alpha}}_{n-1}$ is an upper triangular nonsingular block vector.

Algorithm 7 (An update scheme for the QR decomposition) . *Constructing the upper triangular matrix* $\mathbf{R}_m$ *(5.2) by applying a sequence of unitary matrices* $\hat{\mathbf{U}}_1, \dots, \hat{\mathbf{U}}_m$ *on the matrix* $\underline{\mathbf{T}}_m \mathbf{P}_m$ *5.1). For* $n = 1, \dots, m$*:*

1. *Let* $\widetilde{\boldsymbol{\alpha}}_{n-1} := \boldsymbol{\alpha}_{n-1}\boldsymbol{\pi}_n$*, and, if* $n > 1$*,* $\widetilde{\boldsymbol{\beta}}_{n-2} := \boldsymbol{\beta}^{\mathsf{H}}_{n-2}\boldsymbol{\pi}_n$*.*
   If $n > 2$*, apply* $\mathbf{U}^{\mathsf{H}}_{n-2}$ *to the new last column of* $\underline{\mathbf{T}}_n$*:*

$$\begin{pmatrix} \widetilde{\boldsymbol{\gamma}}_{n-3} \\ \widetilde{\boldsymbol{\beta}}_{n-2} \end{pmatrix} := \hat{\mathbf{U}}^{\mathsf{H}}_{n-2} \begin{pmatrix} \mathbf{0}_{\mathsf{s}_{n-3}\times \mathsf{s}_{n-1}} \\ \widetilde{\boldsymbol{\beta}}_{n-2} \end{pmatrix};$$

   if $n > 1$*, apply* $\mathbf{U}^{\mathsf{H}}_{n-1}$ *to the last column of* $\mathbf{U}^{\mathsf{H}}_{n-2}\underline{\mathbf{T}}_n$*:*

$$\begin{pmatrix} \widetilde{\boldsymbol{\beta}}_{n-2} \\ \widetilde{\boldsymbol{\alpha}}_{n-1} \end{pmatrix} := \hat{\mathbf{U}}^{\mathsf{H}}_{n-1} \begin{pmatrix} \widetilde{\boldsymbol{\beta}}_{n-2} \\ \widetilde{\boldsymbol{\alpha}}_{n-1} \end{pmatrix}.$$

2. *Let* $\boldsymbol{\mu}_n := \widetilde{\boldsymbol{\alpha}}_{n-1}$ *and compute* $\hat{\mathbf{U}}^{\mathsf{H}}_n$ *according to* (5.11)*.*
3. *Compute* $\mathbf{Q}_{n+1}$ *according to algorithm 6.*

There are various possible ways to construct such a matrix $\hat{\mathbf{U}}_n$. Freund [7] applies Givens rotations as one single right-hand side should be a special case of the general block problem. In the case of one right-hand side it is enough to apply a single Givens rotation as we have seen before. So the philosophy behind this approach is to generalize this special case [6]. For our goals the most efficient way is to use a product of complex Householder reflections.

5.2. Complex Householder reflections

Let $\mathbf{y} = \begin{pmatrix} y_1 & \dots y_n \end{pmatrix}^{\mathsf{T}}$ a complex vector. The scope is to construct an unitary matrix $\mathbf{U} \in \mathbb{C}^{n\times n}$ such that

$$\mathbf{U}\mathbf{y} = \alpha \mathbf{e}_1$$

where $\alpha \in \mathbb{C}$. As $\mathbf{U}$ is unitary we note $|\alpha| = \|\mathbf{y}\|_2$. Let $\mathbf{v} \in \mathbb{C}^n$ then the matrix

$$\mathbf{H}_{\mathbf{v}} = \mathbf{I}_n - 2\frac{\mathbf{v}\,\mathbf{v}^{\mathsf{H}}}{\langle \mathbf{v}, \mathbf{v}\rangle} = \mathbf{I}_n + \beta\,\mathbf{v}\,\mathbf{v}^{\mathsf{H}}$$

where $\beta = -2/\langle \mathbf{v}, \mathbf{v}\rangle \in \mathbb{R}$ is called a Householder reflection. The matrix $\mathbf{H}_{\mathbf{v}}$ describes a reflection on the layer orthogonal to $\mathbf{v}$. We note $\mathbf{H}_{\mathbf{v}}$ is Hermitian and unitary, i.e. $\mathbf{H}^{\mathsf{H}}_{\mathbf{v}} = \mathbf{H}_{\mathbf{v}}$ and $\mathbf{H}_{\mathbf{v}}\mathbf{H}^{\mathsf{H}}_{\mathbf{v}} = \mathbf{I}_n$. The vector $\mathbf{y}$ is mapped to

$$\mathbf{H}_{\mathbf{v}}\mathbf{y} = \mathbf{y} + \beta\mathbf{v}\,\langle \mathbf{v}, \mathbf{y}\rangle\,. \tag{5.12}$$

Keeping in mind our scope we demand

$$\mathbf{H}_\mathbf{v}\mathbf{y} = \alpha\mathbf{e}_1.$$

This yields

$$\mathbf{y} - \alpha\mathbf{e}_1 = -\beta\left\langle \mathbf{v}, \mathbf{y}\right\rangle \mathbf{v}.$$

In particular $\mathbf{v} \in \mathsf{span}\left\{\mathbf{y} - \alpha\mathbf{e}_1\right\}$. As $\mathbf{H}_\mathbf{v} = \mathbf{H}_{\lambda\mathbf{v}}$ for all $\lambda \in \mathbb{C} - \{0\}$ it is $\mathbf{v} = \mathbf{y} - \alpha\mathbf{e}_1$ without loss of generality. This choice implies

$$\left\langle \mathbf{v}, \mathbf{y}\right\rangle = -\beta^{-1} \in \mathbb{R}$$

Let $y_1 = |y_1|e^{i\theta}$ and $\alpha = \|\mathbf{y}\|_2 e^{i\theta_\alpha}$

$$\left\langle \mathbf{v}, \mathbf{y}\right\rangle = \left\langle \mathbf{y} - \alpha\mathbf{e}_1, \mathbf{y}\right\rangle = \|\mathbf{y}\|_2^2 - \|\mathbf{y}\|_2 e^{-i\theta_\alpha} e^{i\theta}|y_1|$$

So either $\alpha = -\|\mathbf{y}\|_2 e^{i\theta}$ or $\alpha = +\|\mathbf{y}\|_2 e^{i\theta}$. The first choice is better, otherwise cancellation effects in the first component of $\mathbf{v}$ might occur. Parlett [23] presents a thorough discussion on the choice of the sign when computing Householder reflectors. Finally we note

$$\left\langle \mathbf{v}, \mathbf{y}\right\rangle = \|\mathbf{y}\|_2\left(\|\mathbf{y}\|_2 + |y_1|\right) = -\beta^{-1}.$$

Algorithm 8 (Implicit construction of $\mathbf{H}_\mathbf{v}$) .
Let $\mathbf{y} = \begin{pmatrix} y_1 & \dots & y_n \end{pmatrix}^\mathsf{T}$ *a complex vector. A Householder reflection* $\mathbf{H}_\mathbf{v}$ *of order* n *is constructed such that* $\mathbf{H}_\mathbf{v}\mathbf{y} = \alpha\mathbf{e}_1$*. Let* $y_1 = |y_1|e^{i\theta}$

- *Compute* α *and* β

$$\alpha = -\|\mathbf{y}\|_2 e^{i\theta} \qquad \beta = \frac{-1}{\|\mathbf{y}\|_2\left(\|\mathbf{y}\|_2 + |y_1|\right)} \tag{5.13}$$

- *Compute the vector* $\mathbf{v}$

$$\mathbf{v} = \mathbf{y} - \alpha\mathbf{e}_1 \tag{5.14}$$

It is not necessary to compute the actual matrix $\mathbf{H}_\mathbf{v}$. It is better to store the vector $\mathbf{v}$ and the coefficient β and apply the identity (5.12). Lehoucq [18] compares different variants for the choice of the vector $\mathbf{v}$ and the corresponding coefficient β. The above scheme is due to Wilkinson [37, pp. 49-50]. A slight modification is used in EISPACK[1]

[1] EISPACK was a collection of Fortran subroutines that compute the eigenvalues and eigenvectors of matrices. Lehoucq compares the different computations of an elementary unitary matrix in EISPACK, LINPACK, NAG and LAPACK.

5.3. QR decomposition of a trapezoidal matrix

The idea is to use the trapezoidal structure of the matrix $\boldsymbol{\nu}_n$ in equation (5.11). Suppose $\mathsf{s}_{n-1} = 5$ and $\mathsf{s}_n = 4$, i.e.

$$\begin{pmatrix} \boldsymbol{\mu}_n \\ \boldsymbol{\nu}_n \end{pmatrix} = \left(\begin{array}{ccccc} \circ & \circ & \circ & \circ & \circ \\ \circ & \circ & \circ & \circ & \circ \\ \circ & \circ & \circ & \circ & \circ \\ \circ & \circ & \circ & \circ & \circ \\ \circ & \circ & \circ & \circ & \circ \\ \hline \circ & \circ & \circ & \circ & \circ \\ & \circ & \circ & \circ & \circ \\ & & \circ & \circ & \circ \\ & & & \circ & \circ \end{array} \right).$$

We determine s_{n-1} Householder reflections $\mathbf{H}_{1,n}, \ldots, \mathbf{H}_{\mathsf{s}_{n-1},n}$ such that

$$\begin{pmatrix} \widetilde{\boldsymbol{\alpha}}_{n-1} \\ \mathbf{0}_{\mathsf{s}_n \times \mathsf{s}_{n-1}} \end{pmatrix} = \mathbf{H}_{\mathsf{s}_{n-1},n} \ldots \mathbf{H}_{1,n} \begin{pmatrix} \boldsymbol{\mu}_n \\ \boldsymbol{\nu}_n \end{pmatrix} \tag{5.15}$$

where $\widetilde{\boldsymbol{\alpha}}_{n-1}$ is an upper triangular matrix. In particular

$$\hat{\mathbf{U}}_n = \mathbf{H}_{1,n} \ldots \mathbf{H}_{\mathsf{s}_{n-1},n}. \tag{5.16}$$

Assume that Householder reflections $\mathbf{H}_{1,n}, \mathbf{H}_{2,n}$ have been computed such that

$$\mathbf{H}_{2,n}\mathbf{H}_{1,n} \begin{pmatrix} \boldsymbol{\mu}_n \\ \boldsymbol{\nu}_n \end{pmatrix} = \left(\begin{array}{ccccc} \circ & \circ & \circ & \circ & \circ \\ & \circ & \circ & \circ & \circ \\ & & \bullet & \circ & \circ \\ & & \bullet & \circ & \circ \\ & & \bullet & \circ & \circ \\ \hline & & \bullet & \circ & \circ \\ & & \bullet & \circ & \circ \\ & & \bullet & \circ & \circ \\ & & & \circ & \circ \end{array} \right)$$

The highlighted vector generates the next Householder reflection. In step i this vector has the length

$$l_{i,n} = \underbrace{\mathsf{s}_{n-1} - i + 1}_{\text{length of the upper part}} + \underbrace{\min(i, \mathsf{s}_n)}_{\text{length of the lower part}}$$

and the last entry is in row

$$e_{i,n} = l_{i,n} + i - 1.$$

In this example we have

$$l_{3,n} = 5 - 3 + 1 + 3 = 6 \qquad e_{i,n} = 6 + 3 - 1 = 8.$$

The highlighted vector generates a $l_{i,n} \times l_{i,n}$ unitary matrix $\hat{\mathbf{H}}_{i,n}$. In particular

$$\mathbf{H}_{i,n} = \operatorname{diag}\left(\mathbf{I}_{i-1}, \hat{\mathbf{H}}_{i,n}, \mathbf{I}_{\mathsf{s}_n - \min(i,\mathsf{s}_n)}\right)$$

When we apply this reflection we only compute those entries which are not invariant. In this example the first two and the last row would be not influenced at all. All we have to do is to apply the reflection $\hat{\mathbf{H}}_{i,n}$ on the submatrix whose left column is exactly given by the vector generating $\hat{\mathbf{H}}_{i,n}$. Here the submatrix is highlighted:

$$\left(\begin{array}{ccccc}
\circ & \circ & \circ & \circ & \circ \\
 & \circ & \circ & \circ & \circ \\
 & & \bullet & \bullet & \bullet \\
 & & \bullet & \bullet & \bullet \\
 & & \bullet & \bullet & \bullet \\
\hline
 & & \bullet & \bullet & \bullet \\
 & & \bullet & \bullet & \bullet \\
 & & \bullet & \bullet & \bullet \\
 & & & \circ & \circ
\end{array}\right)$$

This submatrix is updated and we proceed with the construction of the next reflection. The matrix $\hat{\mathbf{U}}_n$ is constructed by a reverse application of all those reflections, i.e. starting with $\mathbf{H}_{\mathsf{s}_{n-1},n}$ on $\mathbf{I}_{\mathsf{s}_{n-1}+\mathsf{s}_n \times \mathsf{s}_{n-1}+\mathsf{s}_n}$ following equation (5.16). An alternative approach might be to conjugate $\mathbf{H}_{\mathsf{s}_{n-1},n} \dots \mathbf{H}_{1,n}\mathbf{I}_{\mathsf{s}_{n-1}+\mathsf{s}_n \times \mathsf{s}_{n-1}+\mathsf{s}_n}$ which is less attractive due to Trefethen [33, Page 74].

Algorithm 9 (Explicit construction of $\hat{\mathbf{U}}_n$) .
Let $\boldsymbol{\mu}_n$ a square block of order s_{n-1} and $\boldsymbol{\nu}_n$ an upper trapezoidal $\mathsf{s}_n \times \mathsf{s}_{n-1}$ block. In a implicit way we construct s_{n-1} Householder reflections such that (5.15) holds and determine the matrix $\hat{\mathbf{U}}_n$ (5.16). Let $\hat{\mathbf{U}}_n = \mathbf{I}_{\mathsf{s}_{n-1}+\mathsf{s}_n}$ and $\mathbf{M} = \begin{pmatrix} \boldsymbol{\mu}_n & \boldsymbol{\nu}_n \end{pmatrix}^{\mathsf{T}}$.

For $i = 1, \dots, \mathsf{s}_{n-1}$:

- *Compute $l_{i,n}$ and $e_{i,n}$*

$$l_{i,n} = \mathsf{s}_{n-1} - i + 1 + \min(i, \mathsf{s}_n) \qquad e_{i,n} = l_{i,n} + i - 1 \tag{5.17}$$

- *Create implicit the Householder reflection $\hat{\mathbf{H}}_{i,n}$ with the vector*

$$\mathbf{y}_{i,n} = \mathbf{M}(i : e_{i,n}, i) \tag{5.18}$$

- *Apply $\hat{\mathbf{H}}_{i,n}$ on the corresponding submatrix of $\mathbf{M}$*

$$\mathbf{M}(i : e_{i,n}, i : \mathsf{s}_{n-1}) = \hat{\mathbf{H}}_{i,n}\mathbf{M}(i : e_{i,n}, i : \mathsf{s}_{n-1}) \tag{5.19}$$

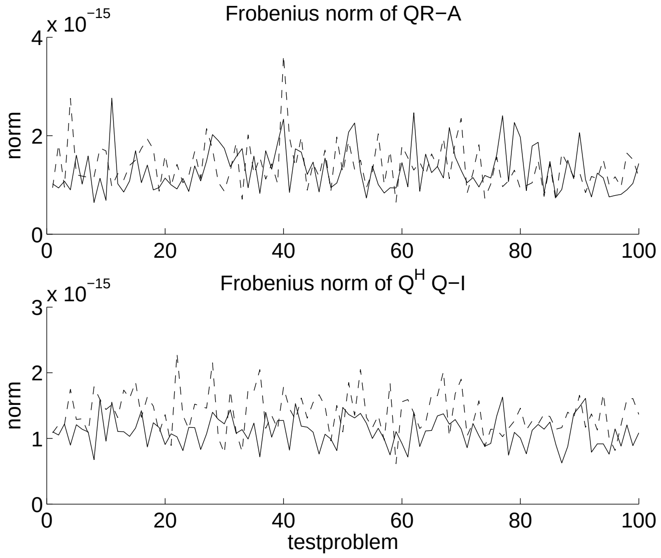


Figure 5.1.: Experiment 13: The solid line represents results gained by using Householder reflections. The dashed line corresponds to Givens rotations.

For $i = \mathsf{s}_{n-1}, \ldots, 1$*:*

- *Update* $\hat{\mathbf{U}}_n$ *by* $\hat{\mathbf{H}}_{i,n}$

$$\hat{\mathbf{U}}_n \left(i : e_{i,n}, i : \mathsf{s}_{n-1} + \mathsf{s}_n\right) = \hat{\mathbf{H}}_{i,n} \hat{\mathbf{U}}_n \left(i : e_{i,n}, i : \mathsf{s}_{n-1} + \mathsf{s}_n\right) \tag{5.20}$$

5.4. Householder reflections vs. Givens rotations

Given an upper trapezoidal matrix with $\mathsf{s}_1 = \mathsf{s}_2 = n$ the construction of $\hat{\mathbf{U}}_n$ and the upper triangular matrix $\mathbf{R}$ is of complexity $O(n^3)$. However, the asymptotic behavior is not of interest here as we usually work with a rather small matrices.

Experiment 13 *Here we work with a set of* 100 *random* 10×5 *upper trapezoidal matrices. Concerning accuracy both methods seem to be on the same level. See Figure 5.1.*

Experiment 14 *In order to compare the speed we have measured the cpu time for* 100 *random matrices of width* w.

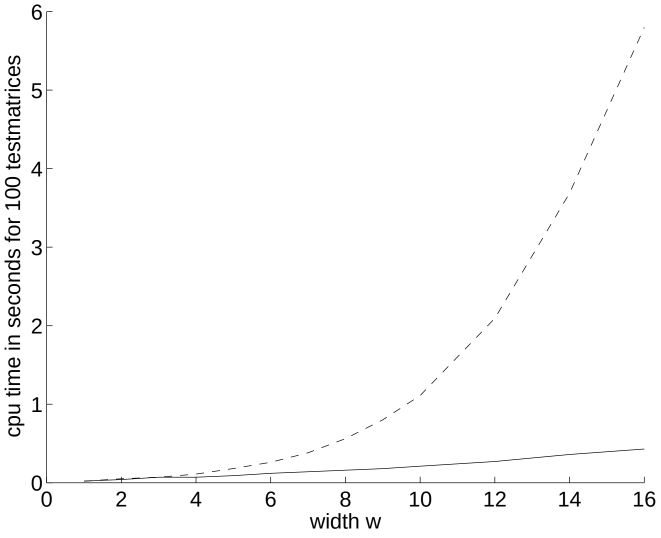


Figure 5.2.: Experiment 14: The solid line represents results gained by using Householder reflections. The dashed line corresponds to Givens rotations.

Obviously Householder reflections are much faster as soon as the width $w \geq 4$. The Givens rotations in the block QMR algorithm by Freund and Malhotra [7] should be replaced by Householder reflections.

6. A block three-term recurrence relation

Although three-term recurrence relations have a central position on the wide field of Krylov methods, we regard only a recurrence relations induced by the matrix

$$\mathbf{R}_n^{\mathrm{MR}}\mathbf{P}_n^{\mathsf{T}} = \begin{pmatrix} \widetilde{\boldsymbol{\alpha}}_0\boldsymbol{\pi}_1^{\mathsf{T}} & \widetilde{\boldsymbol{\beta}}_0\boldsymbol{\pi}_2^{\mathsf{T}} & \widetilde{\boldsymbol{\gamma}}_0\boldsymbol{\pi}_3^{\mathsf{T}} & & \\ & \widetilde{\boldsymbol{\alpha}}_1\boldsymbol{\pi}_2^{\mathsf{T}} & \widetilde{\boldsymbol{\beta}}_1\boldsymbol{\pi}_3^{\mathsf{T}} & \ddots & \\ & & \ddots & \ddots & \widetilde{\boldsymbol{\gamma}}_{n-3}\boldsymbol{\pi}_n^{\mathsf{T}} \\ & & & \ddots & \widetilde{\boldsymbol{\beta}}_{n-2}\boldsymbol{\pi}_n^{\mathsf{T}} \\ & & & & \widetilde{\boldsymbol{\alpha}}_{n-1}\boldsymbol{\pi}_n^{\mathsf{T}} \end{pmatrix} \tag{6.1}$$

where:

$\widetilde{\boldsymbol{\alpha}}_i$ is an $\mathsf{s}_i \times \mathsf{s}_i$ upper triangular block vector with full rank,

$\widetilde{\boldsymbol{\beta}}_i$ is an $\mathsf{s}_i \times \mathsf{s}_{i+1}$ block vector and

$\widetilde{\boldsymbol{\gamma}}_i$ is an $\mathsf{s}_i \times \mathsf{s}_{i+2}$ lower trapezoidal block vector.

6.1. The recurrence relation

Given a block vector $\mathbf{b}_n \in \mathbb{C}^{N\times \mathsf{s}_n^{\square}}$ with

$$\mathbf{b}_n = \begin{pmatrix} \mathbf{b}^{(0)} & \dots & \mathbf{b}^{(n-1)} \end{pmatrix} \tag{6.2}$$

where $\mathbf{b}^{(i)} \in \mathbb{C}^{N\times \mathsf{s}_i}$, then

$$\mathbf{b}_n = \mathbf{x}_n\mathbf{R}_n^{\mathrm{MR}}\mathbf{P}_n^{\mathsf{T}} \tag{6.3}$$

is an equation for the unknown vector $\mathbf{x}_n \in \mathbb{C}^{N\times \mathsf{s}_n^{\square}}$. Note that N is here an arbitrary integer. Obviously

$$\mathbf{x}_n = \mathbf{b}_n\mathbf{P}_n\left(\mathbf{R}_n^{\mathrm{MR}}\right)^{-1}. \tag{6.4}$$

The idea is to read equation (6.3) as a set of recurrence relations, i.e.

$$\begin{aligned} \mathbf{b}^{(0)} &= \left(\mathbf{x}^{(0)}\widetilde{\boldsymbol{\alpha}}_0\right)\boldsymbol{\pi}_1^{\mathsf{T}} \\ \mathbf{b}^{(1)} &= \left(\mathbf{x}^{(0)}\widetilde{\boldsymbol{\beta}}_0 + \mathbf{x}^{(1)}\widetilde{\boldsymbol{\alpha}}_1\right)\boldsymbol{\pi}_2^{\mathsf{T}} \\ \mathbf{b}^{(2)} &= \left(\mathbf{x}^{(0)}\widetilde{\boldsymbol{\gamma}}_0 + \mathbf{x}^{(1)}\widetilde{\boldsymbol{\beta}}_1 + \mathbf{x}^{(2)}\widetilde{\boldsymbol{\alpha}}_2\right)\boldsymbol{\pi}_3^{\mathsf{T}} \\ \mathbf{b}^{(3)} &= \left(\mathbf{x}^{(1)}\widetilde{\boldsymbol{\gamma}}_1 + \mathbf{x}^{(2)}\widetilde{\boldsymbol{\beta}}_2 + \mathbf{x}^{(3)}\widetilde{\boldsymbol{\alpha}}_3\right)\boldsymbol{\pi}_4^{\mathsf{T}} \\ &\vdots \end{aligned}$$

Hence the solution is

$$\begin{aligned}
\mathbf{x}^{(0)} &= \mathbf{b}^{(0)}\boldsymbol{\pi}_1\widetilde{\boldsymbol{\alpha}}_0^{-1} \\
\mathbf{x}^{(1)} &= \left(\mathbf{b}^{(1)}\boldsymbol{\pi}_2 - \mathbf{x}^{(0)}\widetilde{\boldsymbol{\beta}}_0\right)\widetilde{\boldsymbol{\alpha}}_1^{-1} \\
\mathbf{x}^{(2)} &= \left(\mathbf{b}^{(2)}\boldsymbol{\pi}_3 - \mathbf{x}^{(0)}\widetilde{\gamma}_0 - \mathbf{x}^{(1)}\widetilde{\boldsymbol{\beta}}_1\right)\widetilde{\boldsymbol{\alpha}}_2^{-1} \\
\mathbf{x}^{(3)} &= \left(\mathbf{b}^{(3)}\boldsymbol{\pi}_4 - \mathbf{x}^{(1)}\widetilde{\gamma}_1 - \mathbf{x}^{(2)}\widetilde{\boldsymbol{\beta}}_2\right)\widetilde{\boldsymbol{\alpha}}_3^{-1} \\
&\vdots
\end{aligned}$$

In the methods we use we append in every iteration a block vector on $\mathbf{b}_n$, i.e.

$$\mathbf{b}_{n+1} = \begin{pmatrix} \mathbf{b}_n & \mathbf{b}^{(n)} \end{pmatrix}. \tag{6.5}$$

It is therefore enough to evaluate

$$\mathbf{x}^{(n)} = \left(\mathbf{b}^{(n)}\boldsymbol{\pi}_{n+1} - \mathbf{x}^{(n-2)}\widetilde{\gamma}_{n-2} - \mathbf{x}^{(n-1)}\widetilde{\boldsymbol{\beta}}_{n-1}\right)\widetilde{\boldsymbol{\alpha}}_n^{-1} \tag{6.6}$$

in each step in order to solve the system (6.3). Instead of computing the inverse of $\widetilde{\boldsymbol{\alpha}}_n^{-1}$ we solve the system

$$\widetilde{\boldsymbol{\alpha}}_n^{\mathsf{T}}\left(\mathbf{x}^{(n)}\right)^{\mathsf{T}} = \left(\mathbf{b}^{(n)}\boldsymbol{\pi}_{n+1} - \mathbf{x}^{(n-2)}\widetilde{\gamma}_{n-2} - \mathbf{x}^{(n-1)}\widetilde{\boldsymbol{\beta}}_{n-1}\right)^{\mathsf{T}}$$

6.2. Ill-conditioned diagonal blocks

A problem might arise if $\widetilde{\boldsymbol{\alpha}}_n$ is ill-conditioned. In the ordinary Lanczos process it is $\widetilde{\alpha}_n \in \mathbb{R}$, i.e. this problem can not occur there.

Corollary 13 *In exact arithmetic all diagonal entries of* $\mathbf{R}_n^{\mathrm{MR}}$ *are greater than* $\inf(\mathbf{R}_n^{\mathrm{MR}}) \geq \inf(\mathbf{A})$.

Proof: Due to Corollary 2 it is $\inf(\mathbf{R}_n^{\mathrm{MR}}) \geq \inf(\mathbf{A})$. As $\inf(\mathbf{R}_n^{\mathrm{MR}}) \leq \|\mathbf{R}_n^{\mathrm{MR}}\mathbf{e}_1\|_2 = |\mathbf{R}_{n_{1,1}}^{\mathrm{MR}}|$ the claim is true for the first diagonal entry. Assuming the first $k-1$ diagonal entries are greater than $\inf(\mathbf{R}_n^{\mathrm{MR}})$ it is possible to construct a normed vector $\mathbf{w}$ where the only the first k entries do not vanish. The first $k-1$ components are chosen such that

$$\left(\mathbf{R}_n^{\mathrm{MR}}\mathbf{w}\right)^{\mathsf{T}} = \begin{pmatrix} 0 & \dots & 0 & \mathbf{R}_{n_{k,k}}^{\mathrm{MR}} w_k & 0 & \dots \end{pmatrix}^{\mathsf{T}}.$$

Hence $\|\mathbf{R}_n^{\mathrm{MR}}\mathbf{w}\|_2 = |\mathbf{R}_{n_{k,k}}^{\mathrm{MR}} w_k| \geq \inf(\mathbf{R}_n^{\mathrm{MR}})$. As $|w_k| \leq 1$ it is $|\mathbf{R}_{n_{k,k}}^{\mathrm{MR}}| \geq \inf(\mathbf{R}_n^{\mathrm{MR}})$. □

The construction of $\widetilde{\boldsymbol{\alpha}}_n$ in (5.11) yields that $\inf(\widetilde{\boldsymbol{\alpha}}_n) \geq \inf(\boldsymbol{\beta}_n)$. If the smallest singular value of $\boldsymbol{\beta}_n$ is greater than the deflation tolerance then $\boldsymbol{\beta}_n$ is a square matrix. In the case of deflation $\inf(\boldsymbol{\beta}_n) = 0$. Although $\inf(\boldsymbol{\beta}_n)$ is rather small in some of the experiments performed in chapter 4 the corresponding condition of $\widetilde{\boldsymbol{\alpha}}_n$ does not exceed any critical borders. There the block $\boldsymbol{\mu}_n$ compensates small singular values in $\boldsymbol{\beta}_n$. However, it is worth noting that the deflation scheme does not guarantee small condition numbers for the blocks $\widetilde{\boldsymbol{\alpha}}_n$. In non-exact arithmetic the condition of a block $\widetilde{\boldsymbol{\alpha}}_n$ can become arbitrarily large. The idea is to determine the condition of $\widetilde{\boldsymbol{\alpha}}_i$ in the ith iteration. If it exceeds a certain a priori given limit a restart is a possible remedy.

7. A block minimal residual method

A generalization of MinRes for several right-hand sides based on the block Lanczos process is introduced. The perturbation by deflation in the basis of the block Krylov space is analyzed. Minimizing the remaining quasiresidual is almost trivial however linear dependency might be exploited.

For every system the 2-norm of the residual is minimized, that is we identify $\mathbf{k}_n^{(i)}$ in (4.28) such that

$$\|\mathbf{r}_n^{(i)}\|_2 = \min_{\mathbf{k}_n^{(i)} \in \mathbb{C}^{\mathsf{s}_n^{\square}}} \|\mathbf{r}_0^{(i)} - \mathbf{A}\mathbf{Y}_n\mathbf{k}_n^{(i)}\|_2 \quad \forall \quad i = 1, \dots \mathsf{s}$$

where the columns of $\mathbf{Y}_n$ are an orthonormal basis for $\mathcal{B}_n(\mathbf{A}, \mathbf{r}_0)$. Hence

$$\|\mathbf{r}_n\|_F = \min_{\mathbf{k}_n \in \mathbb{C}^{\mathsf{s}_n^{\square} \times \mathsf{s}}} \|\mathbf{r}_0 - \mathbf{A}\mathbf{Y}_n\mathbf{k}_n\|_F. \tag{7.1}$$

As $\mathbf{r}_0 = \mathbf{Y}_{n+1}\underline{\mathbf{e}}_1\boldsymbol{\eta}_0 + \mathbf{Y}_{n+1}^{\Delta}\underline{\mathbf{e}}_1^{\Delta}\boldsymbol{\eta}_0^{\Delta}$ we obtain by inserting $\mathbf{A}\mathbf{Y}_n = \mathbf{Y}_{n+1}\underline{\mathbf{T}}_n + \mathbf{Y}_{n+1}^{\Delta}\underline{\mathbf{T}}_n^{\Delta}$:

$$\|\mathbf{r}_n\|_F = \min_{\mathbf{k}_n \in \mathbb{C}^{\mathsf{s}_n^{\square} \times \mathsf{s}}} \|\mathbf{Y}_{n+1}\mathbf{e}_1\boldsymbol{\eta}_0 + \mathbf{Y}_{n+1}^{\Delta}\underline{\mathbf{e}}_1^{\Delta}\boldsymbol{\eta}_0^{\Delta} - \left(\mathbf{Y}_{n+1}\underline{\mathbf{T}}_n + \mathbf{Y}_{n+1}^{\Delta}\underline{\mathbf{T}}_n^{\Delta}\right)\mathbf{k}_n\|_F. \tag{7.2}$$

7.1. The quasiresidual and the deflated quasiresidual

Deflation in the basis of the block Krylov space leads to a small correction term in the residual. If the norm of this term exceeds the the tolerance for accepting an approximation the system might fail to converge.

Before we minimized a quasiresidual $\mathbf{q}_n$ given by $\mathbf{r}_n = \mathbf{Y}_{n+1}\mathbf{q}_n$, in particular $\|\mathbf{q}_n\|_2 = \|\mathbf{r}_n\|_2$. Reordering the terms in (7.2) yields

$$\mathbf{r}_n = \mathbf{Y}_{n+1}\underbrace{\left(\underline{\mathbf{e}}_1\boldsymbol{\eta}_0 - \underline{\mathbf{T}}_n\mathbf{k}_n\right)}_{:=\mathbf{q}_n} + \mathbf{Y}_{n+1}^{\Delta}\underbrace{\left(\underline{\mathbf{e}}_1^{\Delta}\boldsymbol{\eta}_0^{\Delta} - \underline{\mathbf{T}}_n^{\Delta}\mathbf{k}_n\right)}_{:=\mathbf{q}_n^{\Delta}} \tag{7.3}$$

If the second term called **deflated quasiresidual** is small the minimization of $\mathbf{q}_n$ is still an attractive approach. Using Lemma 5 it is

$$\|\mathbf{Y}_{n+1}^{\Delta}\mathbf{q}_n^{\Delta}\|_F \leq \sqrt{\mathsf{d}_{n+1}^{\square}}\|\mathbf{q}_n^{\Delta}\|_F$$

The triangle inequality and the consistency of the Frobenius norm imply

$$\|\mathbf{q}_n^{\Delta}\|_F \leq \|\underline{\mathbf{e}}_1^{\Delta}\|_F\|\boldsymbol{\eta}_0^{\Delta}\|_F + \|\underline{\mathbf{T}}_n^{\Delta}\|_F\|\mathbf{k}_n\|_F$$

With $\|\underline{\mathbf{e}}_1^\Delta\|_F = \sqrt{\mathsf{s}-\mathsf{s}_0}$, $\|\boldsymbol{\eta}_0^\Delta\|_F = O(\text{tol})$ and the upper bound for $\|\underline{\mathbf{T}}_n^\Delta\|_F$ given in Lemma 12 we end up with

$$\|\mathbf{q}_n^\Delta\|_F \leq \sqrt{\mathsf{d}_{n+1}^\Box}\left(\sqrt{\mathsf{s}-\mathsf{s}_0} + \sqrt{\tilde{\mathsf{d}}_n}\|\mathbf{k}_n\|_F\right) O(\text{tol}) = O(\text{tol})$$

Hence using a large tolerance might hamper convergence however it is necessary to deflate otherwise orthogonality in the Lanczos process is abruptly lost and therefore a small tolerance should be used. Here we identify already the central problem of this approach. A restart for those systems which failed to converge is an option suggested by Gutknecht. However, it is difficult during an iteration, that is before the maximal number of iterations is reached, to decide whether a system will converge or will fail to achieve the required precision. In the case of exact deflation the deflated quasiresidual vanishes.

7.2. The minimization of the quasiresidual

By using the full QR decomposition of $\underline{\mathbf{T}}_n\mathbf{P}_n$ (5.1), it is

$$\mathbf{q}_n = \underline{\mathbf{e}}_1\boldsymbol{\eta}_0 - \underline{\mathbf{T}}_n\mathbf{k}_n = \underline{\mathbf{e}}_1\boldsymbol{\eta}_0 - \mathbf{Q}_{n+1}\underline{\mathbf{R}}_n^{\text{MR}}\mathbf{P}_n^T\mathbf{k}_n$$

The upper square $\mathsf{s}_n^\Box \times \mathsf{s}_n^\Box$ submatrix of $\underline{\mathbf{R}}_n^{\text{MR}}$ is denoted by $\mathbf{R}_n^{\text{MR}}$. We define

$$\underline{\mathbf{h}}_n :\equiv \left(\frac{\mathbf{h}_n}{\widetilde{\eta}_n}\right) :\equiv \mathbf{Q}_{n+1}^{\mathsf{H}}\underline{\mathbf{e}}_1\boldsymbol{\eta}_0.$$

In view of

$$\begin{aligned}\|\mathbf{q}_n\|_F^2 &= \|\mathbf{Q}_{n+1}^{\mathsf{H}}\underline{\mathbf{e}}_1\boldsymbol{\eta}_0 - \underline{\mathbf{R}}_n^{\text{MR}}\mathbf{P}_n^T\mathbf{k}_n\|_F^2 \\ &= \|\underline{\mathbf{h}}_n - \underline{\mathbf{R}}_n^{\text{MR}}\mathbf{P}_n^T\mathbf{k}_n\|_F^2 \\ &= \|\mathbf{h}_n - \mathbf{R}_n^{\text{MR}}\mathbf{P}_n^T\mathbf{k}_n\|_F^2 + \|\widetilde{\eta}_n\|_F^2\end{aligned} \tag{7.4}$$

we see that

$$\mathbf{k}_n = \mathbf{P}_n(\mathbf{R}_n^{\text{MR}})^{-1}\mathbf{h}_n \tag{7.5}$$

is the solution of our least-squares problem and that the corresponding least-squares error equals

$$\|\underline{\mathbf{e}}_1\boldsymbol{\eta}_0 - \underline{\mathbf{T}}_n\mathbf{k}_n\|_F^2 = \|\widetilde{\eta}_n\|_F^2 \tag{7.6}$$

We rewrite $\mathbf{x}_n = \mathbf{x}_0 + \mathbf{Y}_n\mathbf{k}_n$ as

$$\mathbf{x}_n = \mathbf{x}_0 + \mathbf{Z}_n\mathbf{h}_n \qquad \text{where} \quad \mathbf{Z}_n :\equiv \mathbf{Y}_n\mathbf{P}_n(\mathbf{R}_n^{\text{MR}})^{-1} \tag{7.7}$$

contains the search directions $\mathbf{z}_0, \ldots, \mathbf{z}_{n-1}$. The matrix $\mathbf{R}_n^{\text{MR}}$ is a banded upper block tridiagonal matrix with bandwidth three. Therefore the relation

$$\mathbf{Y}_n = \mathbf{Z}_n\mathbf{R}_n^{\text{MR}}\mathbf{P}_n^{\mathsf{T}}$$

can be viewed as the matrix representation of a block three-term recursion for generating the vectors $\{\mathbf{z}_k\}_{k=0}^{n-1}$:

$$\widetilde{\boldsymbol{\alpha}}_k^{\mathsf{T}} \mathbf{z}_k^{\mathsf{T}} = \Big(\mathbf{y}_k \boldsymbol{\pi}_{k+1} - \mathbf{z}_{n-2} \widetilde{\gamma}_{k-2} - \mathbf{z}_{n-1} \widetilde{\boldsymbol{\beta}}_{k-1} \Big)^{\mathsf{T}} \tag{7.8}$$

7.3. An update scheme

An update scheme for $\underline{\mathbf{h}}_n$ and the quasiresidual $\mathbf{Y}_{n+1}\mathbf{q}_n$ is proposed. With (5.3) it is

$$\underline{\mathbf{h}}_0 = \mathbf{Q}_1^{\mathsf{H}} \underline{\mathbf{e}}_1 \boldsymbol{\eta}_0 = \boldsymbol{\eta}_0.$$

The unitary matrix $\mathbf{Q}_{n+1}$ is a product of Householder reflections (5.4), thus updating $\underline{\mathbf{h}}_{n-1}$ is simple:

$$\begin{aligned} \underline{\mathbf{h}}_n &= \text{block diag} \left(\mathbf{I}_{\mathsf{s}^{\square}_{n-1}}, \hat{\mathbf{U}}_n^{\mathsf{H}} \right) \text{block diag} \left(\mathbf{Q}_n^{\mathsf{H}}, \mathbf{I}_{\mathsf{s}_n} \right) \underline{\mathbf{e}}_1 \boldsymbol{\eta}_0 \\ &= \text{block diag} \left(\mathbf{I}_{\mathsf{s}^{\square}_{n-1}}, \hat{\mathbf{U}}_n^{\mathsf{H}} \right) \left(\begin{array}{c} \mathbf{h}_{n-1} \\ \hline \widetilde{\boldsymbol{\eta}}_{n-1} \\ \mathbf{0} \end{array} \right) \end{aligned}$$

Using the block form of $\hat{\mathbf{U}}_n$ (5.6) it is

$$\underline{\mathbf{h}}_n = \left(\begin{array}{c} \mathbf{h}_{n-1} \\ \hline \hat{\mathbf{U}}_{n,u}^{\mathsf{H}} \widetilde{\boldsymbol{\eta}}_{n-1} \end{array} \right).$$

Hence

$$\mathbf{h}_n = \left(\begin{array}{c} \mathbf{h}_{n-1} \\ \hat{\mathbf{U}}_{n,u,l}^{\mathsf{H}} \, \widetilde{\boldsymbol{\eta}}_{n-1} \end{array} \right) \tag{7.9}$$

and

$$\widetilde{\boldsymbol{\eta}}_n = \hat{\mathbf{U}}_{n,u,r}^{\mathsf{H}} \, \widetilde{\boldsymbol{\eta}}_{n-1}. \tag{7.10}$$

Example. Let $\hat{\mathbf{U}}_n$ a Givens rotation. Then

$$\hat{\mathbf{U}}_n = \begin{pmatrix} c_n & -\overline{s_n} \\ s_n & c_n \end{pmatrix}, \qquad \hat{\mathbf{U}}_{n,u} = \begin{pmatrix} c_n & -\overline{s_n} \end{pmatrix}, \qquad \text{and} \qquad \hat{\mathbf{U}}_{n,u}^{\mathsf{H}} = \begin{pmatrix} c_n \\ -s_n \end{pmatrix}$$

Hence

$$\underline{\mathbf{h}}_n = \left(\begin{array}{c} \mathbf{h}_{n-1} \\ \hline c_n \, \widetilde{\eta}_{n-1} \\ -s_n \, \widetilde{\eta}_{n-1} \end{array} \right).$$

which coincides with (2.26). ♢

With (7.9) an update scheme for $\mathbf{x}_n$ (7.7) is

$$\mathbf{x}_n = \mathbf{x}_{n-1} + \mathbf{z}_n \mathbf{U}_{n,u,l}^{\mathsf{H}} \, \widetilde{\boldsymbol{\eta}}_{n-1}. \tag{7.11}$$

In view of (7.3) an update scheme for the residual is by using (7.4) and (7.5):

$$\begin{aligned}\mathbf{r}_n = \mathbf{Y}_{n+1}\mathbf{q}_n &= \mathbf{Y}_{n+1}\left(\underline{\mathbf{e}}_1\boldsymbol{\eta}_0 - \mathbf{Q}_{n+1}\underline{\mathbf{R}}_n^{\mathrm{MR}}\mathbf{P}_n^{\mathsf{T}}\mathbf{P}_n\left(\mathbf{R}_n^{\mathrm{MR}}\right)^{-1}\mathbf{h}_n\right)\\ &= \mathbf{Y}_{n+1}\mathbf{Q}_{n+1}\left(\mathbf{Q}_{n+1}^{\mathsf{H}}\underline{\mathbf{e}}_1\boldsymbol{\eta}_0 - \begin{pmatrix}\mathbf{h}_n\\ \mathbf{0}\end{pmatrix}\right)\\ &= \mathbf{Y}_{n+1}\mathbf{Q}_{n+1}\mathbf{I}_{n+1}^{\mathsf{s}_n}\widetilde{\eta}_n\end{aligned}$$

where

$$\mathbf{I}_{n+1}^{\mathsf{s}_n} = \left(\begin{array}{ccc}\multicolumn{3}{c}{\mathbf{0}_{\mathsf{s}_n^{\square}\times\mathsf{s}_n}}\\ \hline 1 & & \\ & \ddots & \\ & & 1\end{array}\right).$$

One might interpret $\mathbf{I}_{n+1}^{\mathsf{s}_n}$ as a "delete" operator. An application on the right side of square matrix of order $\mathsf{s}_{n+1}^{\square}$ gives the last s_n columns of this square matrix, i.e. it deletes the first $\mathsf{s}_n^{\square}$ columns. With (5.4) and (5.5) we see further that

$$\begin{aligned}\mathbf{r}_n &= \begin{pmatrix}\mathbf{Y}_n & \mathbf{y}_n\end{pmatrix}\text{block diag }\left(\mathbf{Q}_n, \mathbf{I}_{\mathsf{s}_n}\right)\text{block diag }\left(\mathbf{I}_{\mathsf{s}_{n-1}^{\square}}, \hat{\mathbf{U}}_n\right)\mathbf{I}_{n+1}^{\mathsf{s}_n}\widetilde{\eta}_n\\ &= \begin{pmatrix}\mathbf{Y}_n & \mathbf{y}_n\end{pmatrix}\text{block diag }\left(\mathbf{Q}_n, \mathbf{I}_{\mathsf{s}_n}\right)\begin{pmatrix}\mathbf{0}\\ \vdots\\ \mathbf{0}\\ \hat{\mathbf{U}}_{n,u,r}\\ \hat{\mathbf{U}}_{n,d,r}\end{pmatrix}\widetilde{\eta}_n\\ &= \left(\mathbf{Y}_n\mathbf{Q}_n\mathbf{I}_n^{\mathsf{s}_{n-1}}\hat{\mathbf{U}}_{n,u,r} + \mathbf{y}_n\hat{\mathbf{U}}_{n,d,r}\right)\widetilde{\eta}_n\end{aligned}$$

We define

$$\mathbf{p}_1 := \mathbf{Y}_1\mathbf{Q}_1\mathbf{I}_1^{\mathsf{s}_0} = \mathbf{y}_0$$

as start for the recurrence relation

$$\mathbf{p}_{n+1} = \mathbf{p}_n\hat{\mathbf{U}}_{n,u,r} + \mathbf{y}_n\hat{\mathbf{U}}_{n,d,r}. \tag{7.12}$$

This yields

$$\mathbf{r}_n = \mathbf{p}_{n+1}\widetilde{\eta}_n. \tag{7.13}$$

Example. Let $\hat{\mathbf{U}}_n$ a Givens rotation. Then

$$\hat{\mathbf{U}}_n = \begin{pmatrix}c_n & -\overline{s_n}\\ s_n & c_n\end{pmatrix}, \qquad \widetilde{\eta}_n = -s_n\widetilde{\eta}_{n-1}.$$

Hence

$$\mathbf{p}_{n+1} = -\mathbf{p}_n\overline{s_n} + \mathbf{y}_n c_n$$

and

$$\mathbf{r}_n = \mathbf{r}_{n-1}|s_n|^2 + \mathbf{y}_n c_n\widetilde{\eta}_n.$$

◊

7.4. Linear dependent quasiresiduals

Exploiting an exact or nearby linear dependency of residuals might reduce the computational work and the memory requirement. However, a dependency is discovered only in the initialization step as

$$\mathbf{r}_0 = \begin{pmatrix} \mathbf{y}_0 & \mathbf{y}_0^{\Delta} \end{pmatrix} \begin{pmatrix} \boldsymbol{\rho}_0 & \boldsymbol{\rho}_0^{\Box} \\ \mathbf{0} & \boldsymbol{\rho}_0^{\Delta} \end{pmatrix} \boldsymbol{\pi}_0^{\mathsf{T}} = \begin{pmatrix} \mathbf{y}_0 & \mathbf{y}_0^{\Delta} \end{pmatrix} \begin{pmatrix} \boldsymbol{\eta}_0 \\ \boldsymbol{\eta}_0^{\Delta} \end{pmatrix} \tag{7.14}$$

The idea is to solve those systems where the initial residual is orthogonal to $\mathbf{y}_0^{\Delta}$. An approach by Gutknecht uses the identities

$$\mathbf{r}_0 \boldsymbol{\pi}_0 = \left(\mathbf{y}_0 \boldsymbol{\rho}_0 \;\middle|\; \mathbf{y}_0 \boldsymbol{\rho}_0^{\Box} + \mathbf{y}_0^{\Delta} \boldsymbol{\rho}_0^{\Delta} \right)$$

and

$$\mathbf{r}_0 \boldsymbol{\pi}_0 = (\mathbf{b} - \mathbf{A}\mathbf{x}_0)\,.$$

By splitting the permutation matrix $\boldsymbol{\pi}_0 = \left(\boldsymbol{\pi}_0^{\Box} \;\middle|\; \boldsymbol{\pi}_0^{\Delta} \right)$ we gain two equalities for the left and right part of the block vector. Both equalities are multiplied by $\mathbf{A}^{-1}$. Hence

$$\mathbf{A}^{-1}\mathbf{y}_0 \boldsymbol{\rho}_0 = \underbrace{\mathbf{A}^{-1}\mathbf{b}\boldsymbol{\pi}_0^{\Box}}_{\equiv:\, \mathbf{x}_*^{\Box}} - \underbrace{\mathbf{x}_0 \boldsymbol{\pi}_0^{\Box}}_{\equiv:\, \mathbf{x}_0^{\Box}} \tag{7.15}$$

and

$$\underbrace{\mathbf{A}^{-1}\mathbf{b}\boldsymbol{\pi}_0^{\Delta}}_{\equiv:\, \mathbf{x}_*^{\Delta}} = \underbrace{\mathbf{x}_0 \boldsymbol{\pi}_0^{\Delta}}_{\equiv:\, \mathbf{x}_0^{\Delta}} + \left(\mathbf{A}^{-1}\mathbf{y}_0\boldsymbol{\rho}_0\right) \left(\boldsymbol{\rho}_0^{-1}\boldsymbol{\rho}_0^{\Box}\right) + \mathbf{A}^{-1}\mathbf{y}_0^{\Delta}\boldsymbol{\rho}_0^{\Delta}\,. \tag{7.16}$$

Here $\mathbf{x}_0^{\Box}$, $\mathbf{x}_0^{\Delta}$, $\mathbf{x}_*^{\Box}$, and $\mathbf{x}_*^{\Delta}$ contain the non-deleted and the deleted columns of the initial approximation and the exact solution, respectively. Likewise, $\mathbf{x}_n^{\Box}$ and $\mathbf{x}_n^{\Delta}$ will contain the undeleted and the deleted columns of the current approximation. The term $\mathbf{A}^{-1}\mathbf{y}_0^{\Delta}\boldsymbol{\rho}_0^{\Delta}$ is of $O(\text{tol})$ and will be neglected.

In the second term on the right-hand side of (7.16) we insert the expression for $\mathbf{A}^{-1}\mathbf{y}_0\boldsymbol{\rho}_0$ obtained from (7.15) after replacing there $\mathbf{x}_*^{\Box}$ by its current approximation $\mathbf{x}_n^{\Box}$.

Then we consider the resulting two terms as the current approximation $\mathbf{x}_n^{\Delta}$ of $\mathbf{x}_*^{\Delta} = \mathbf{A}^{-1}\mathbf{b}\boldsymbol{\pi}_0^{\Delta}$:

$$\mathbf{x}_n^{\Delta} :\equiv \mathbf{x}_0^{\Delta} + \left(\mathbf{x}_n^{\Box} - \mathbf{x}_0^{\Box}\right) \left(\boldsymbol{\rho}_0^{-1}\boldsymbol{\rho}_0^{\Box}\right)\,. \tag{7.17}$$

So, in case of a (typically only approximate) linear dependence of columns of $\mathbf{r}_0$, we can express some of the columns (stored in $\mathbf{x}_n^{\Delta}$) of $\mathbf{x}_n$ in terms of the other columns (stored in $\mathbf{x}_n^{\Box}$) and the $\mathsf{s}_0 \times (\mathsf{s} - \mathsf{s}_0)$ matrix $\boldsymbol{\rho}_0^{-1}\boldsymbol{\rho}_0^{\Delta}$.

Note that these relations for deflation in the initialization step are not limited to block MinRes, but are applicable to many other block Krylov methods.

However, in the current version of block MinRes this option is not exploited for two reasons. A possible dependency can be discovered only at the start. In block MinRes there are no restarts as in block GMRes. A further explicit computation of the residual should be avoided by all means. Furthermore the idea does reduce the number of multiplications by $\mathbf{A}$.

7.5. The algorithm

It is now without efforts to combine the ideas to create block MinRes. In practice this algorithm is embedded into the general preconditioning scheme for iterative solvers proposed in Algorithm 4.

Algorithm 10 (block MinRes) .
Let a Hermitian matrix $\mathbf{A} \in \mathbb{C}^{N\times N}$, *a block vector* $\mathbf{b} \in \mathbb{C}^{N\times \mathsf{s}}$ *and an initial approximation* $\mathbf{x}_0$ *be given. Let* $\mathbf{r}_0 := \mathbf{b} - \mathbf{A}\mathbf{x}_0$, *let*

$$\mathbf{r}_0 = \mathbf{y}_0\boldsymbol{\eta}_0 \quad (\textit{QR factorization: } \boldsymbol{\eta}_0 \in \mathbb{C}^{\mathsf{s}\times\mathsf{s}}, \mathbf{y}_0 \in \mathbb{C}^{N\times\mathsf{s}})$$

Then, for $n = 1, \ldots$:

1. *Compute* $\mathbf{y}_n$, $\boldsymbol{\alpha}_{n-1}$, $\boldsymbol{\pi}_n$ *and* $\boldsymbol{\beta}_{n-1}$ *by one step of the Hermitian block Lanczos algorithm (see Algorithm 5).*
2. *Update the QR decomposition of* $\underline{\mathbf{T}}_n\mathbf{P}_n$ *by one step of the Algorithm 7.*
3. *Construct* $\hat{\mathbf{U}}_n$ *by applying Algorithm 9.*
4. *Construct the new search directions* $\mathbf{z}_n$ *by applying the recurrence relation* (7.8).
5. *Update* $\widetilde{\boldsymbol{\eta}}_n$ *by using* (7.10) *and update* $\mathbf{x}_n$ *by* (7.11).
6. *Update the residual* $\mathbf{r}_n$ *by using* (7.12) *and* (7.13).

8. A block symmetric LQ method

We introduce a block version of SymmLQ. By neglecting all terms of $O(\mathrm{tol})$ it is straightforward to generalize the results gained in Section 2.5 for the block case. Our approach starts again with the auxiliary iterates $\mathbf{x}_n^{\mathsf{L}}$ with a minimal error property. An update formula for the corresponding residual $\mathbf{r}_n^{\mathsf{L}}$ is given. It is also possible to generalize the update scheme for the Galerkin approximations $\mathbf{x}_n$ (2.33) and the residual $\mathbf{r}_n$.

8.1. The normal equations

We define

$$\mathcal{M}_{n,tol}\left(\mathbf{A}, \mathbf{r}_0\right) := \mathbf{A}\mathcal{B}_{n,tol}\left(\mathbf{A}, \mathbf{r}_0\right) = \mathcal{B}_{n,tol}\left(\mathbf{A}, \mathbf{A}\mathbf{r}_0\right) \subset \mathcal{B}_{n+1,tol}\left(\mathbf{A}, \mathbf{r}_0\right)$$

and

$$\mathcal{M}^{\square}_{n,tol}\left(\mathbf{A}, \mathbf{r}_0\right) := \mathbf{A}\mathcal{B}^{\square}_{n,tol}\left(\mathbf{A}, \mathbf{r}_0\right).$$

The variational principle for the auxiliary iterates is

$$\|\mathbf{f}_n^{(i),\mathsf{L}}\|_2 = \min_{\mathbf{x}_n^{(i),\mathsf{L}} \in \mathcal{M}_{n,tol}(\mathbf{A},\mathbf{r}_0)+\mathbf{x}_0^{(i)}} \|\mathbf{x}_n^{(i),\mathsf{L}} - \mathbf{x}_\star^{(i)}\|_2.$$

Using the concept of block vectors we get

$$\|\mathbf{f}_n^{\mathsf{L}}\|_F = \min_{\mathbf{x}_n^{\mathsf{L}} \in \mathcal{M}^{\square}_{n,tol}(\mathbf{A},\mathbf{r}_0)+\mathbf{x}_0} \|\mathbf{x}_n^{\mathsf{L}} - \mathbf{x}_\star\|_F. \tag{8.1}$$

The constraint $\mathbf{x}_n^{\mathsf{L}} \in \mathcal{M}^{\square}_{n,tol}\left(\mathbf{A}, \mathbf{r}_0\right) + \mathbf{x}_0$ is matched by the representation

$$\mathbf{x}_n^{\mathsf{L}} - \mathbf{x}_0 = \mathbf{A}\mathbf{Y}_n\mathbf{k}_n^{\mathsf{L}}. \tag{8.2}$$

Hence condition (8.1) takes the form

$$\|\mathbf{f}_n^{\mathsf{L}}\|_F = \min_{\mathbf{k}_n^{\mathsf{L}} \in \mathbb{C}^{\mathsf{s}_n^{\square} \times \mathsf{s}}} \|\mathbf{A}\mathbf{Y}_n\mathbf{k}_n^{\mathsf{L}} - (\mathbf{x}_\star - \mathbf{x}_0)\|_F \tag{8.3}$$

which is a least squares problem with the normal equations

$$\mathbf{Y}_n^{\mathsf{H}}\mathbf{A}^{\mathsf{H}}\mathbf{A}\mathbf{Y}_n\mathbf{k}_n^{\mathsf{L}} = \mathbf{Y}_n^{\mathsf{H}}\mathbf{A}^{\mathsf{H}}(\mathbf{x}_\star - \mathbf{x}_0) \tag{8.4}$$

and the Galerkin condition

$$\mathbf{f}_n^{\mathsf{L}} = \mathbf{x}_n^{\mathsf{L}} - \mathbf{x}_\star \perp \mathbf{A}\mathcal{B}^{\square}_{n,tol}\left(\mathbf{A}, \mathbf{r}_0\right), \qquad \text{hence} \qquad \mathbf{r}_n^{\mathsf{L}} \perp \mathcal{B}^{\square}_{n,tol}\left(\mathbf{A}, \mathbf{r}_0\right). \tag{8.5}$$

The term on the right-hand side of (8.4) is:

$$\mathbf{Y}_n^{\mathsf{H}}\mathbf{A}^{\mathsf{H}}(\mathbf{x}_\star - \mathbf{x}_0) = \mathbf{Y}_n^{\mathsf{H}}\mathbf{A}(\mathbf{x}_\star - \mathbf{x}_0) = \mathbf{Y}_n^{\mathsf{H}}\mathbf{r}_0.$$

As $\mathbf{r}_0 = \mathbf{Y}_n\mathbf{e}_1\boldsymbol{\eta}_0 + \mathbf{Y}_n^\Delta\mathbf{e}_1^\Delta\boldsymbol{\eta}_0^\Delta$ we obtain

$$\mathbf{Y}_n^{\mathsf{H}}\mathbf{A}^{\mathsf{H}}(\mathbf{x}_\star - \mathbf{x}_0) = \mathbf{e}_1\boldsymbol{\eta}_0 + \mathbf{Y}_n^{\mathsf{H}}\mathbf{Y}_n^\Delta\mathbf{e}_1^\Delta\boldsymbol{\eta}_0^\Delta$$

where $\|\mathbf{Y}_n^{\mathsf{H}}\mathbf{Y}_n^\Delta\mathbf{e}_1^\Delta\boldsymbol{\eta}_0^\Delta\|_F$ is of $O(\text{tol})$.

On the left-hand side, the Lanczos relationship $\mathbf{A}\mathbf{Y}_n = \mathbf{Y}_{n+1}\underline{\mathbf{T}}_n + \mathbf{Y}_{n+1}^\Delta\underline{\mathbf{T}}_n^\Delta$ implies

$$\begin{aligned}\mathbf{Y}_n^{\mathsf{H}}\mathbf{A}^{\mathsf{H}}\mathbf{A}\mathbf{Y}_n &= \left(\mathbf{Y}_{n+1}\underline{\mathbf{T}}_n + \mathbf{Y}_{n+1}^\Delta\underline{\mathbf{T}}_n^\Delta\right)^{\mathsf{H}}\left(\mathbf{Y}_{n+1}\underline{\mathbf{T}}_n + \mathbf{Y}_{n+1}^\Delta\underline{\mathbf{T}}_n^\Delta\right)\\ &= \underline{\mathbf{T}}_n^{\mathsf{H}}\underline{\mathbf{T}}_n + \underline{\mathbf{T}}_n^{\mathsf{H}}\mathbf{Y}_{n+1}^{\mathsf{H}}\mathbf{Y}_{n+1}^\Delta\underline{\mathbf{T}}_n^\Delta + \underline{\mathbf{T}}_n^{\Delta^{\mathsf{H}}}\mathbf{Y}^{\Delta^{\mathsf{H}}}_{n+1}\mathbf{Y}_{n+1}\underline{\mathbf{T}}_n + \underline{\mathbf{T}}_n^{\Delta^{\mathsf{H}}}\mathbf{Y}^{\Delta^{\mathsf{H}}}_{n+1}\mathbf{Y}_{n+1}^\Delta\underline{\mathbf{T}}_n^\Delta.\end{aligned}$$

Terms of $O(\text{tol})$ are neglected and thus the modified normal equations are

$$\underline{\mathbf{T}}_n^{\mathsf{H}}\underline{\mathbf{T}}_n\mathbf{k}_n^{\mathsf{L}} = \mathbf{e}_1\boldsymbol{\eta}_0 \tag{8.6}$$

8.2. Iterative solution of the normal equations

The QR decomposition of $\underline{\mathbf{T}}_n\mathbf{P}_n = \underline{\mathbf{Q}}_n\mathbf{R}_n^{\mathrm{MR}}$ with an $\mathsf{s}_{n+1}^\square \times \mathsf{s}_n^\square$ matrix $\underline{\mathbf{Q}}_n$ with orthonormal columns and an $\mathsf{s}_n^\square \times \mathsf{s}_n^\square$ upper triangular $\mathbf{R}_n^{\mathrm{MR}}$ implies

$$\underline{\mathbf{T}}_n^{\mathsf{H}}\underline{\mathbf{T}}_n = \mathbf{P}_n\left(\underline{\mathbf{T}}_n\mathbf{P}_n\right)^{\mathsf{H}}\underline{\mathbf{T}}_n\mathbf{P}_n\mathbf{P}_n^{\mathsf{T}} = \mathbf{P}_n(\mathbf{R}_n^{\mathrm{MR}})^{\mathsf{H}}\underline{\mathbf{Q}}_n^{\mathsf{H}}\underline{\mathbf{Q}}_n\mathbf{R}_n^{\mathrm{MR}}\mathbf{P}_n^{\mathsf{T}} = \mathbf{P}_n(\mathbf{R}_n^{\mathrm{MR}})^{\mathsf{H}}\mathbf{R}_n^{\mathrm{MR}}\mathbf{P}_n^{\mathsf{T}}\,. \tag{8.7}$$

$\mathbf{P}_n(\mathbf{R}_n^{\mathrm{MR}})^{\mathsf{H}}\mathbf{R}_n^{\mathrm{MR}}\mathbf{P}_n^{\mathsf{T}}$ is the Cholesky decomposition of the Hermitian positive definite matrix $\underline{\mathbf{T}}_n^{\mathsf{H}}\underline{\mathbf{T}}_n$, here computed via the more stable QR decomposition of $\underline{\mathbf{T}}_n\mathbf{P}_n$. Altogether, (8.6) reduces to

$$\mathbf{P}_n(\mathbf{R}_n^{\mathrm{MR}})^{\mathsf{H}}\mathbf{R}_n^{\mathrm{MR}}\mathbf{P}_n^{\mathsf{T}}\mathbf{k}_n^{\mathsf{L}} = \mathbf{e}_1\boldsymbol{\eta}_0 \tag{8.8}$$

Setting $\mathbf{L}_n^{\mathrm{MR}} :\equiv (\mathbf{R}_n^{\mathrm{MR}})^{\mathsf{H}}$ and inserting $\mathbf{k}_n^{\mathsf{L}}$ into (8.2) we obtain

$$\mathbf{x}_n^{\mathsf{L}} = \mathbf{x}_0 + \mathbf{A}\mathbf{Y}_n\mathbf{P}_n(\mathbf{R}_n^{\mathrm{MR}})^{-1}(\mathbf{L}_n^{\mathrm{MR}})^{-1}\mathbf{P}_n^{\mathsf{H}}\mathbf{e}_1\boldsymbol{\eta}_0.$$

Let

$$\mathbf{g}_n :\equiv (\mathbf{L}_n^{\mathrm{MR}})^{-1}\mathbf{P}_n^{\mathsf{H}}\mathbf{e}_1\boldsymbol{\eta}_0 = (\mathbf{L}_n^{\mathrm{MR}})^{-1}\mathbf{e}_1\boldsymbol{\pi}_1^{\mathsf{T}}\boldsymbol{\eta}_0\,, \tag{8.9}$$

then with $\mathbf{A}\mathbf{Y}_n = \mathbf{Y}_{n+1}\underline{\mathbf{T}}_n + \mathbf{Y}_{n+1}^\Delta\underline{\mathbf{T}}_n^\Delta$ we get

$$\mathbf{x}_n^{\mathsf{L}} = \mathbf{x}_0 + \left(\mathbf{Y}_{n+1}\underline{\mathbf{T}}_n\mathbf{P}_n\mathbf{P}_n^{\mathsf{T}} + \mathbf{Y}_{n+1}^\Delta\underline{\mathbf{T}}_n^\Delta\right)\mathbf{P}_n(\mathbf{R}_n^{\mathrm{MR}})^{-1}\mathbf{g}_n.$$

The terms of $O(\text{tol})$ are again swept under the carpet. Using the QR decomposition of $\underline{\mathbf{T}}_n\mathbf{P}_n$ we obtain

$$\mathbf{x}_n^{\mathsf{L}} = \mathbf{x}_0 + \mathbf{Y}_{n+1}\underline{\mathbf{Q}}_n\mathbf{g}_n$$

So, if we let

$$\mathbf{W}_n :\equiv \begin{pmatrix} \mathbf{w}_0 & \dots & \mathbf{w}_{n-1} \end{pmatrix} :\equiv \mathbf{Y}_{n+1}\underline{\mathbf{Q}}_n \tag{8.10}$$

we finally get

$$\mathbf{x}_n^{\mathsf{L}} = \mathbf{x}_0 + \mathbf{W}_n\mathbf{g}_n = \mathbf{x}_{n-1}^{\mathsf{L}} + \mathbf{w}_{n-1}\mathbf{g}^{(n-1)}. \tag{8.11}$$

Furthermore we introduce

$$\overline{\mathbf{W}}_{n+1} :\equiv \begin{pmatrix} \mathbf{w}_0 & \dots & \mathbf{w}_{n-1} & \overline{\mathbf{w}}_n \end{pmatrix} :\equiv \mathbf{Y}_{n+1}\mathbf{Q}_{n+1}$$

which implies

$$\mathbf{W}_n = \overline{\mathbf{W}}_{n+1}\begin{pmatrix} \mathbf{I}_n \\ \mathbf{0} \end{pmatrix}.$$

The matrix $\overline{\mathbf{W}}_n$ is easy to update by appending the last column of $\mathbf{Y}_{n+1}$ and applying the Givens transformation $\mathbf{G}_n$ to the last two columns, that is

$$\begin{pmatrix} \mathbf{w}_{n-1} & \overline{\mathbf{w}}_n \end{pmatrix} = \begin{pmatrix} \overline{\mathbf{w}}_{n-1} & \mathbf{y}_n \end{pmatrix}\hat{\mathbf{U}}_n. \tag{8.12}$$

It is worth to study this block three-term recursion relation. As

$$(\mathbf{L}_n^{\mathrm{MR}})\mathbf{g}_n = \begin{pmatrix} \widetilde{\boldsymbol{\alpha}}_0 & & & & & \\ \widetilde{\boldsymbol{\beta}}_0^{\mathsf{H}} & \widetilde{\boldsymbol{\alpha}}_1 & & & & \\ \widetilde{\boldsymbol{\gamma}}_0^{\mathsf{H}} & \widetilde{\boldsymbol{\beta}}_1^{\mathsf{H}} & \ddots & & & \\ & \ddots & \ddots & \ddots & & \\ & & \ddots & \ddots & \widetilde{\boldsymbol{\alpha}}_{n-1} & \\ & & & \widetilde{\boldsymbol{\gamma}}_{n-3}^{\mathsf{H}} & \widetilde{\boldsymbol{\beta}}_{n-2}^{\mathsf{H}} & \widetilde{\boldsymbol{\alpha}}_{n-1} \end{pmatrix}\begin{pmatrix} \mathbf{g}^{(0)} \\ \vdots \\ \mathbf{g}^{(n-1)} \end{pmatrix} = \begin{pmatrix} \boldsymbol{\pi}_1^{\mathsf{T}}\boldsymbol{\eta}_0 \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{pmatrix} \tag{8.13}$$

we get by setting $\widetilde{\boldsymbol{\gamma}}_{-1} = \mathbf{0}$ and $\mathbf{g}^{(-1)} = \mathbf{0}$

$$\widetilde{\boldsymbol{\gamma}}_{n-2}^{\mathsf{H}}\mathbf{g}^{(n-2)} + \widetilde{\boldsymbol{\beta}}_{n-1}^{\mathsf{H}}\mathbf{g}^{(n-1)} + \widetilde{\boldsymbol{\alpha}}_n\mathbf{g}^{(n)} = \mathbf{0} \qquad n \geq 1.$$

We could exhibit the relation $\mathbf{L}_n^{\mathrm{MR}} = \left(\mathbf{R}_n^{\mathrm{MR}}\right)^{\mathsf{H}}$ by writing the relation $\mathbf{L}_n^{\mathrm{MR}}\mathbf{g}_n = \mathbf{e}_1\boldsymbol{\pi}_1^{\mathsf{T}}\boldsymbol{\eta}_0$ as $\mathbf{g}_n^{\mathsf{H}}\mathbf{R}_n^{\mathrm{MR}} = \boldsymbol{\eta}_0^{\mathsf{H}}\boldsymbol{\pi}_1\mathbf{e}_1^{\mathsf{T}}$. The resulting three-term recursion is described in chapter 6.

Again the goal is to construct an update scheme for the residual without any further explicit multiplication of a block vector by $\mathbf{A}$. It is

$$\mathbf{r}_n^{\mathsf{L}} = \mathbf{b} - \mathbf{A}\mathbf{x}_n^{\mathsf{L}} = \mathbf{b} - \mathbf{A}\left(\mathbf{A}\mathbf{Y}_n\mathbf{k}_n^{\mathsf{L}} + \mathbf{x}_0\right) = \mathbf{r}_0 - \mathbf{A}^2\mathbf{Y}_n\mathbf{k}_n^{\mathsf{L}}.$$

The coefficient block vector is given by

$$\mathbf{k}_n^{\mathsf{L}} = \mathbf{P}_n(\mathbf{R}_n^{\mathrm{MR}})^{-1}\mathbf{g}_n \tag{8.14}$$

and $\mathbf{r}_0 = \mathbf{Y}_{n+2}\mathbf{e}_1\boldsymbol{\eta}_0$, thus

$$\mathbf{r}_n^{\mathsf{L}} = \mathbf{Y}_{n+2}\mathbf{e}_1\boldsymbol{\eta}_0 - \mathbf{A}\mathbf{Y}_{n+1}\underline{\mathbf{T}}_n\mathbf{P}_n(\mathbf{R}_n^{\mathrm{MR}})^{-1}\mathbf{g}_n$$

where we have neglected the term $\mathbf{A}\mathbf{Y}^{\Delta}_{n+1}\underline{\mathbf{T}}^{\Delta}_{n}\mathbf{P}_n(\mathbf{R}^{\mathrm{MR}}_n)^{-1}\mathbf{g}_n$. A second application of the Lanczos identity (4.22) implies with (8.8) and (8.14)

$$\mathbf{r}^{\mathsf{L}}_n = \mathbf{Y}_{n+2}\left(\mathbf{P}_{n+2}(\mathbf{R}^{\mathrm{MR}}_{n+2})^{\mathsf{H}}\mathbf{g}_{n+2} - \underline{\mathbf{T}}_{n+1}\underline{\mathbf{T}}_{n}\mathbf{P}_n(\mathbf{R}^{\mathrm{MR}}_n)^{-1}\mathbf{g}_n\right).$$

We introduce a further "delete" operator by defining:

$$\underline{\underline{\mathbf{I}}}_{n} := \begin{pmatrix} \mathbf{I}_n \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix} \in \mathsf{s}^{\square}_{n+2} \times \mathsf{s}^{\square}_{n}.$$

This operator deletes the last two block columns of a matrix if applied on the right-hand side, respectively the last two rows if applied on the left-hand side. It is

$$\underline{\mathbf{T}}_{n} = \underline{\underline{\mathbf{I}}}^{\mathsf{T}}_{n+1}\begin{pmatrix} \underline{\mathbf{T}}_{n} \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix} = \underline{\underline{\mathbf{I}}}^{\mathsf{T}}_{n+1}\underline{\mathbf{T}}_{n+2}\underline{\underline{\mathbf{I}}}_{n} \quad \text{and} \quad \underline{\mathbf{T}}_{n+1} = \underline{\mathbf{T}}^{\mathsf{H}}_{n+2}\underline{\underline{\mathbf{I}}}_{n+1}.$$

This yields

$$\underline{\mathbf{T}}_{n+1}\underline{\mathbf{T}}_{n} = \underline{\mathbf{T}}^{\mathsf{H}}_{n+2}\underline{\underline{\mathbf{I}}}_{n+1}\underline{\underline{\mathbf{I}}}^{\mathsf{T}}_{n+1}\underline{\mathbf{T}}_{n+2}\underline{\underline{\mathbf{I}}}_{n} = \underline{\mathbf{T}}^{\mathsf{H}}_{n+2}\begin{pmatrix} \mathbf{I}_{n+1} & & \\ & \mathbf{0} & \\ & & \mathbf{0} \end{pmatrix}\begin{pmatrix} \underline{\mathbf{T}}_{n} \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix}.$$

Obviously by using (8.7) it is

$$\underline{\mathbf{T}}_{n+1}\underline{\mathbf{T}}_{n} = \underline{\mathbf{T}}^{\mathsf{H}}_{n+2}\mathbf{I}_{n+3}\begin{pmatrix} \underline{\mathbf{T}}_{n} \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix} = \underline{\mathbf{T}}^{\mathsf{H}}_{n+2}\underline{\mathbf{T}}_{n+2}\underline{\underline{\mathbf{I}}}_{n} = \mathbf{P}_{n+2}(\mathbf{R}^{\mathrm{MR}}_{n+2})^{\mathsf{H}}\mathbf{R}^{\mathrm{MR}}_{n+2}\mathbf{P}^{\mathsf{H}}_{n+2}\underline{\underline{\mathbf{I}}}_{n}.$$

For the residual this result implies

$$\mathbf{r}^{\mathsf{L}}_n = \mathbf{Y}_{n+2}\mathbf{P}_{n+2}(\mathbf{R}^{\mathrm{MR}}_{n+2})^{\mathsf{H}}\left(\mathbf{g}_{n+2} - \mathbf{R}^{\mathrm{MR}}_{n+2}\mathbf{P}^{\mathsf{H}}_{n+2}\underline{\underline{\mathbf{I}}}_{n}\mathbf{P}_n(\mathbf{R}^{\mathrm{MR}}_n)^{-1}\mathbf{g}_n\right).$$

With the properties of $\underline{\underline{\mathbf{I}}}_{n}$ we get

$$\mathbf{r}^{\mathsf{L}}_n = \mathbf{Y}_{n+2}\mathbf{P}_{n+2}(\mathbf{R}^{\mathrm{MR}}_{n+2})^{\mathsf{H}}\left(\mathbf{g}_{n+2} - \underline{\underline{\mathbf{I}}}_{n}\mathbf{g}_n\right)$$

or written in slightly different notation:

$$\mathbf{r}^{\mathsf{L}}_n = \mathbf{Y}_{n+2}\mathbf{P}_{n+2}\begin{pmatrix} \widetilde{\boldsymbol{\alpha}}_0 & & & & & \\ \widetilde{\boldsymbol{\beta}}^{\mathsf{H}}_0 & \widetilde{\boldsymbol{\alpha}}_1 & & & & \\ \widetilde{\boldsymbol{\gamma}}^{\mathsf{H}}_0 & \widetilde{\boldsymbol{\beta}}^{\mathsf{H}}_1 & \ddots & & & \\ & \ddots & \ddots & \ddots & & \\ & & \ddots & \ddots & \widetilde{\boldsymbol{\alpha}}_n & \\ & & & \widetilde{\boldsymbol{\gamma}}^{\mathsf{H}}_{n-1} & \widetilde{\boldsymbol{\beta}}^{\mathsf{H}}_n & \widetilde{\boldsymbol{\alpha}}_{n+1} \end{pmatrix}\begin{pmatrix} \mathbf{0} \\ \vdots \\ \mathbf{0} \\ \mathbf{g}^{(n)} \\ \mathbf{g}^{(n+1)} \end{pmatrix}$$

Hence

$$\mathbf{r}^{\mathsf{L}}_n = \mathbf{y}_n\boldsymbol{\pi}_{n+1}\widetilde{\boldsymbol{\alpha}}_n\mathbf{g}^{(n)} + \mathbf{y}_{n+1}\boldsymbol{\pi}_{n+2}\left(\widetilde{\boldsymbol{\beta}}^{\mathsf{H}}_n\mathbf{g}^{(n)} + \widetilde{\boldsymbol{\alpha}}_{n+1}\mathbf{g}^{(n+1)}\right)$$

Using the relation (8.14) we finally end up with

$$\mathbf{r}^{\mathsf{L}}_n = \mathbf{y}_n\boldsymbol{\pi}_{n+1}\widetilde{\boldsymbol{\alpha}}_n\mathbf{g}^{(n)} - \mathbf{y}_{n+1}\boldsymbol{\pi}_{n+2}\boldsymbol{\gamma}^{\mathsf{H}}_{n-1}\mathbf{g}^{(n-1)}. \tag{8.15}$$

8.3. The Galerkin approximations

Here we use the iterates $\mathbf{x}_n^L$ in order to solve the Ritz-Galerkin problem (2.33), that is

$$\underline{\mathbf{T}}_n^{\mathsf{T}}\underline{\mathbf{T}}_n\mathbf{k}_n^{\mathsf{L}} = \mathbf{T}_n\mathbf{k}_n = \mathbf{e}_1\boldsymbol{\eta}_0 \tag{8.16}$$

It might happen that $\mathbf{k}_n$ is not unique or does not exist. Therefore we generalize first the identity (2.34). It is

$$\begin{aligned}
\mathbf{T}_n &= \mathbf{P}_n\mathbf{P}_n^{\mathsf{T}}\underline{\mathbf{T}}_n^{\mathsf{H}}\begin{pmatrix}\mathbf{I}_{\mathsf{s}_n^{\square}}\\ \mathbf{0}\end{pmatrix}\\
&= \mathbf{P}_n(\underline{\mathbf{R}}_n^{\mathrm{MR}})^{\mathsf{H}}\mathbf{Q}_{n+1}^{\mathsf{H}}\begin{pmatrix}\mathbf{I}_{\mathsf{s}_n^{\square}}\\ \mathbf{0}\end{pmatrix}\\
&= \mathbf{P}_n(\underline{\mathbf{R}}_n^{\mathrm{MR}})^{\mathsf{H}}\mathbf{U}_n^{\mathsf{H}}\begin{pmatrix}\mathbf{Q}_n^{\mathsf{H}}\\ \mathbf{0}\end{pmatrix}\\
&= \mathbf{P}_n(\mathbf{R}_n^{\mathrm{MR}})^{\mathsf{H}}\begin{pmatrix}\mathbf{I}_{\mathsf{s}_{n-1}^{\square}} & \\ & \hat{\mathbf{U}}_{n,u,l}^{\mathsf{H}}\end{pmatrix}\mathbf{Q}_n^{\mathsf{H}}
\end{aligned} \tag{8.17}$$

Using the Cholesky decompositions for the normal equations (8.7) and (8.14) we end up with

$$\mathbf{P}_n(\mathbf{R}_n^{\mathrm{MR}})^{\mathsf{H}}\mathbf{R}_n^{\mathrm{MR}}\mathbf{P}_n^{\mathsf{T}}\mathbf{P}_n(\mathbf{R}_n^{\mathrm{MR}})^{-1}\mathbf{g}_n = \mathbf{P}_n(\mathbf{R}_n^{\mathrm{MR}})^{\mathsf{H}}\begin{pmatrix}\mathbf{I}_{\mathsf{s}_{n-1}^{\square}} & \\ & \hat{\mathbf{U}}_{n,u,l}^{\mathsf{H}}\end{pmatrix}\mathbf{Q}_n^{\mathsf{H}}\mathbf{k}_n$$

and therefore

$$\mathbf{g}_n = \begin{pmatrix}\mathbf{I}_{\mathsf{s}_{n-1}^{\square}} & \\ & \hat{\mathbf{U}}_{n,u,l}^{\mathsf{H}}\end{pmatrix}\mathbf{Q}_n^{\mathsf{H}}\mathbf{k}_n.$$

If the matrix $\hat{\mathbf{U}}_{n,u,l}^{\mathsf{H}}$ is invertible it is

$$\begin{pmatrix}\mathbf{I}_{\mathsf{s}_{n-1}^{\square}} & \\ & \left(\hat{\mathbf{U}}_{n,u,l}^{\mathsf{H}}\right)^{-1}\end{pmatrix}\mathbf{g}_n = \mathbf{Q}_n^{\mathsf{H}}\mathbf{k}_n.$$

We define

$$\mathbf{v}^{(n-1)} := \left(\hat{\mathbf{U}}_{n,u,l}^{\mathsf{H}}\right)^{-1}\mathbf{g}^{(n-1)}. \tag{8.18}$$

The vector $\mathbf{v}^{(n-1)}$ may not exist in every iteration as $\hat{\mathbf{U}}_{n,u,l}^{\mathsf{H}}$ is not invertible. If its exist it is possible to update the Galerkin approximation. As proposed for Ritz-Galerkin methods it is

$$\begin{aligned}
\mathbf{x}_n &= \mathbf{x}_0 + \mathbf{Y}_n\mathbf{k}_n\\
&= \mathbf{x}_0 + \mathbf{Y}_n\mathbf{Q}_n\mathbf{Q}_n^{\mathsf{H}}\mathbf{k}_n\\
&= \mathbf{x}_0 + \mathbf{W}_{n-1}\mathbf{g}_{n-1} + \overline{\mathbf{w}}_{n-1}\left(\hat{\mathbf{U}}_{n,u,l}^{\mathsf{H}}\right)^{-1}\mathbf{g}^{(n-1)}\\
&= \mathbf{x}_{n-1}^{\mathsf{L}} + \overline{\mathbf{w}}_{n-1}\mathbf{v}^{(n-1)}.
\end{aligned} \tag{8.19}$$

It remains to give an update formula for the corresponding residual

$$\begin{aligned}
\mathbf{r}_n &= \mathbf{b} - \mathbf{A}\left(\mathbf{x}_0 + \mathbf{Y}_n \mathbf{k}_n\right) \\
&= \mathbf{r}_0 - \mathbf{Y}_{n+1} \underline{\mathbf{T}}_n \mathbf{k}_n \\
&= \mathbf{Y}_n \mathbf{e}_1 \boldsymbol{\eta}_0 - \begin{pmatrix} \mathbf{Y}_n & \mathbf{y}_n \end{pmatrix} \begin{pmatrix} & \mathbf{T}_n & \\ \mathbf{0} \quad \dots & \mathbf{0} & \boldsymbol{\beta}_{n-1} \end{pmatrix} \mathbf{k}_n \\
&= \mathbf{Y}_n \mathbf{T}_n \mathbf{k}_n - \mathbf{Y}_n \mathbf{T}_n \mathbf{k}_n - \mathbf{y}_n \begin{pmatrix} \mathbf{0} & \dots & \mathbf{0} & \boldsymbol{\beta}_{n-1} \end{pmatrix} \mathbf{k}_n \\
&= -\mathbf{y}_n \begin{pmatrix} \mathbf{0} & \dots & \mathbf{0} & \boldsymbol{\beta}_{n-1} \end{pmatrix} \mathbf{k}_n
\end{aligned}$$

Hence we effectively have to compute the last entry $\mathbf{k}_n$. As

$$\mathbf{k}_n = \begin{pmatrix} \mathbf{Q}_{n-1} & \\ & \mathbf{I}_{\mathsf{s}_n} \end{pmatrix} \begin{pmatrix} \mathbf{I}_{\mathsf{s}^{\square}_{n-2}} & & \\ & \hat{\mathbf{U}}_{n-1,u,l} & \hat{\mathbf{U}}_{n-1,u,r} \\ & \hat{\mathbf{U}}_{n-1,d,l} & \hat{\mathbf{U}}_{n-1,d,r} \end{pmatrix} \begin{pmatrix} \mathbf{I}_{\mathsf{s}^{\square}_{n-1}} & \\ & \left(\hat{\mathbf{U}}^{\mathsf{H}}_{n,u,l}\right)^{-1} \end{pmatrix} \mathbf{g}_n$$

the residual is given by

$$\mathbf{r}_n = -\mathbf{y}_n \boldsymbol{\beta}_{n-1} \left(\hat{\mathbf{U}}_{n-1,d,l} \mathbf{g}^{(n-2)} + \hat{\mathbf{U}}_{n-1,d,r} \mathbf{v}^{(n-1)}\right). \tag{8.20}$$

8.4. The algorithm

It is now without efforts to combine the ideas to create block SymmLQ. In practice this algorithm is also embedded into the general preconditioning scheme for iterative solvers proposed in Algorithm 4.

Algorithm 11 (block SymmLQ) .
Let a Hermitian matrix $\mathbf{A} \in \mathbb{C}^{N \times N}$*, a block vector* $\mathbf{b} \in \mathbb{C}^{N \times \mathsf{s}}$ *and an initial approximation* $\mathbf{x}_0$ *be given. Let* $\mathbf{r}_0 := \mathbf{b} - \mathbf{A}\mathbf{x}_0$*, let*

$$\mathbf{r}_0 = \mathbf{y}_0 \boldsymbol{\eta}_0 \quad (\textit{QR factorization: } \boldsymbol{\eta}_0 \in \mathbb{C}^{\mathsf{s} \times \mathsf{s}}, \mathbf{y}_0 \in \mathbb{C}^{N \times \mathsf{s}})$$

and let $\overline{\mathbf{w}}_0 = \mathbf{y}_0$*. Then, for* $n = 1, \dots,$*:*

1. *Compute* $\mathbf{y}_n$*,* $\boldsymbol{\alpha}_{n-1}$*,* $\boldsymbol{\pi}_n$ *and* $\boldsymbol{\beta}_{n-1}$ *by one step of the Hermitian block Lanczos algorithm (see Algorithm 5).*
2. *Update the QR decomposition of* $\underline{\mathbf{T}}_n \mathbf{P}_n$ *by one step of the Algorithm 7.*
3. *Construct* $\hat{\mathbf{U}}_n$ *by applying Algorithm 9.*
4. *Compute* $\mathbf{g}^{(n-1)}$ *by using* (8.13)*.*
5. *Compute* $\mathbf{w}_{n-1}$ *and* $\overline{\mathbf{w}}_n$ *due to* (8.12)*.*

6. *Update* $\mathbf{x}_n^{\mathsf{L}}$ *by using* (8.11).
7. *Update the residual* $\mathbf{r}_n^{\mathsf{L}}$ *by using* (8.15).
8. *If* $\hat{\mathbf{U}}_{n,u,l}^{\mathsf{H}}$ *is invertible compute the Galerkin approximation also. Compute* $\mathbf{v}^{(n-1)}$ (8.18) *and update* $\mathbf{x}_n$ *and* $\mathbf{r}_n$ *by using* (8.19) *and* (8.20)

9. Numerical Results

Here we apply block MinRes and MinRes to those experiments constructed already in Chapter 4. We neglect the barely measurable differences between block SymmLQ and block MinRes. The experiments constructed in Chapter 4 are designed for our purposes to illustrate a certain behavior of the methods. We decided to use also an application from the real world for our benchmark. We discuss the discrete Stokes equations from fluid dynamics. The estimated residuals are gained by the update formulas of MinRes and block MinRes.

9.1. Experiments using exact deflation

All effects in the following experiments are in principle already explained in Chapter 4. The results are linked to the behavior of the block Lanczos process. Nevertheless it is worth to execute and document those experiments also. In all experiments we tried to achieve an accuracy of 10^{-13} or 10^{-10} with respect to the relative residual.

In Experiment 5 we already observe a typical behavior. The speed of convergence does not vary for different random right-hand sides as every side is a random linear combination of all eigenvectors which often correspond to degenerated eigenvalues. All singular values remain rather large and there are no problems with the condition of the diagonal blocks $\widetilde{\boldsymbol{\alpha}}_i$. We observe superlinear convergence.

Block MinRes is even more superior if the right-hand sides lie in a common small eigenspace as the block Lanczos process profits from the fast construction of an orthonormal basis as explained in Experiment 6. The behavior of the smallest singular value is plotted in Figure 4.21. For Experiment 7 we expect that the advantage of a common eigenspace is less dominating as the block Lanczos algorithm does not profit from double eigenvalues as Lanczos does. There is no advantage to expect if all right-hand sides lie in distinct eigenspaces as in Experiment 8.

The abrupt loss of orthogonality in Experiment 9 extremely hampers convergence. Block MinRes can not compete with MinRes in this case. For Experiment 10 problems are even more severe. Working with blocks of width larger than 2 the method is very unreliable without deflation if small singular values appear. However, often the singular values remain rather large as observed in Experiment 5.

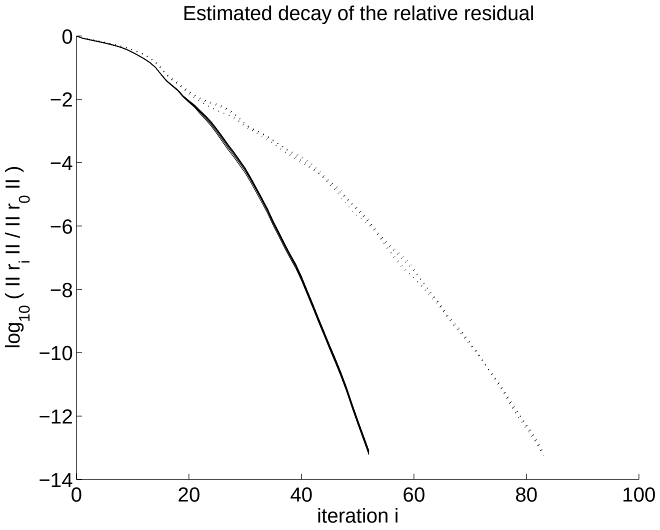


Figure 9.1.: Experiment 5: The relative residuals for block MinRes (solid lines) and MinRes (dotted lines)

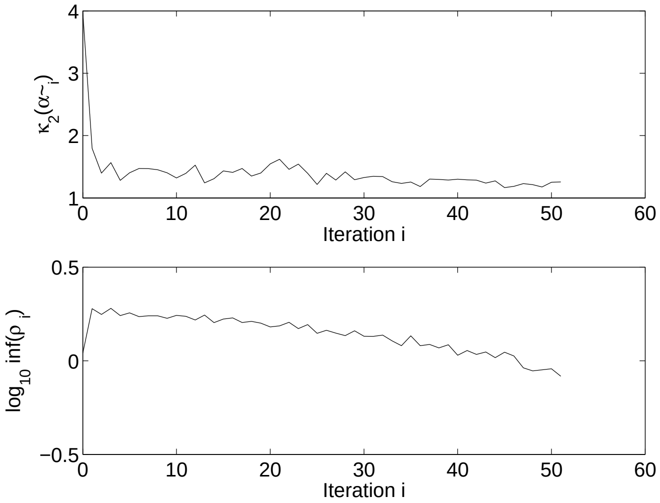


Figure 9.2.: Experiment 5: All singular values remain rather large.

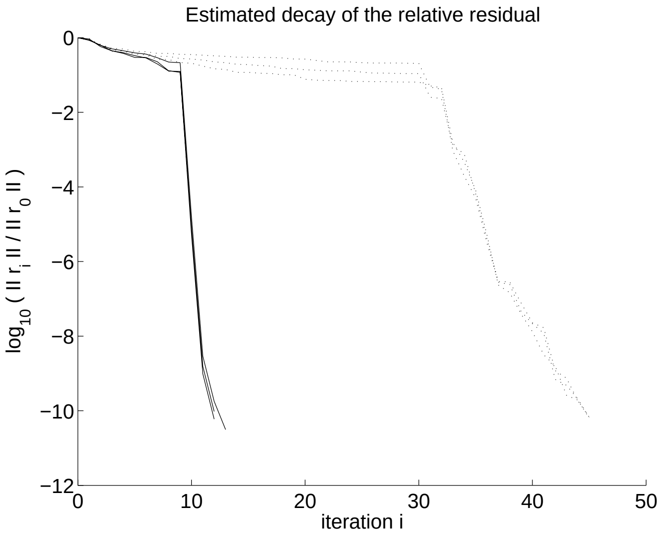


Figure 9.3.: Experiment 6: All right-hand sides lie in a common small eigenspace. The relative residuals for block MinRes (solid lines) and MinRes (dotted lines)

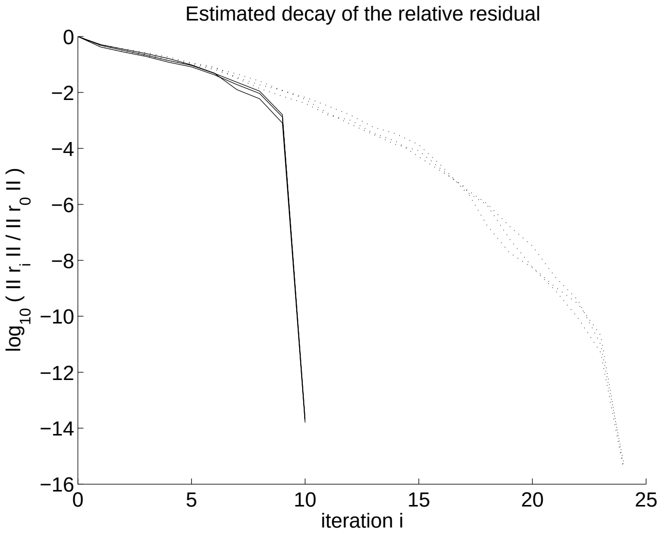


Figure 9.4.: Experiment 7: All right-hand sides lie in a common small eigenspace. The relative residuals for block MinRes (solid lines) and MinRes (dotted lines)

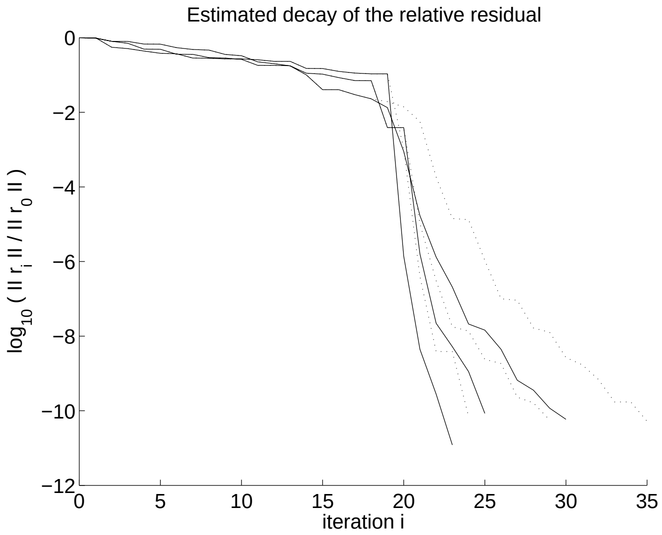


Figure 9.5.: Experiment 8: All right-hand sides lie in distinct eigenspaces. The relative residuals for block MinRes (solid lines) and MinRes (dotted lines)

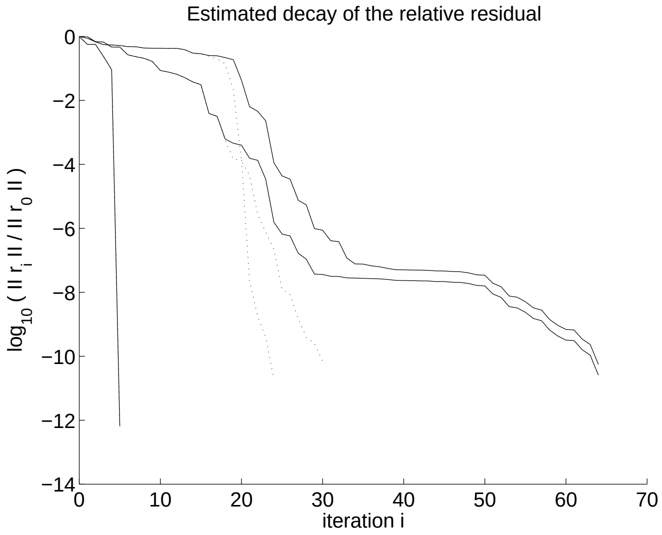


Figure 9.6.: Experiment 9: Extreme slow convergence due to refrained deflation. The relative residuals for block MinRes (solid lines) and MinRes (dotted lines)

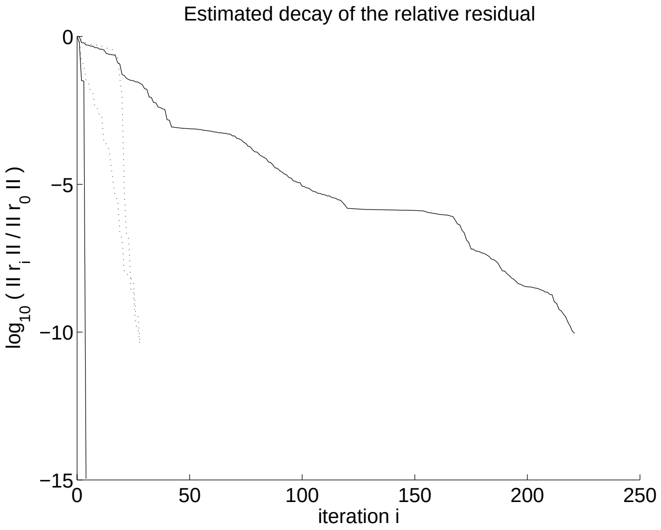


Figure 9.7.: Experiment 10: The relative residuals for block MinRes (solid lines) and MinRes (dotted lines)

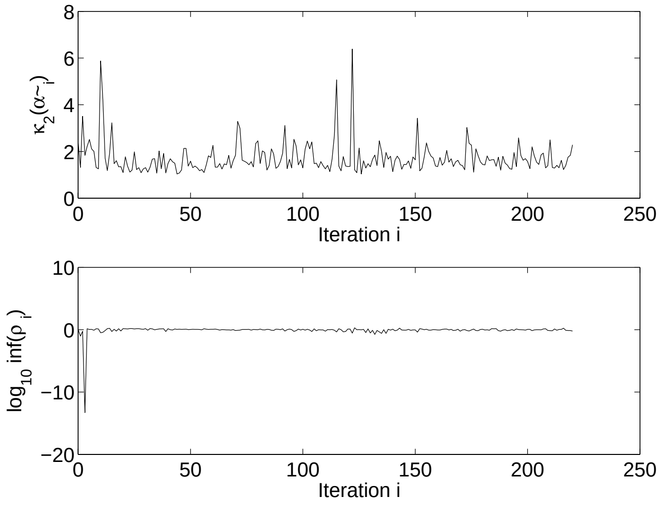


Figure 9.8.: Experiment 10: Ignoring the small singular value has disastrous effects.

9.2. Experiments using inexact deflation

Deflation is of interest only if small singular values are observed. To understand the negative effects we first return to case of a single system. In Experiment 3 only a deflation tolerance larger than 10^{-7} would imply deflation (see Figure 4.8). In this case the algorithm would stop with a successful construction of $\mathbf{Y}_{15}$. But the 2-norm of the corresponding residual is approximately 10^{-8} (see Figure 4.16). Hence any deflation tolerance smaller than 10^{-7} would prevent block MinRes to achieve an accuracy better than 10^{-8} for this experiment. In this case deflation hampers convergence.

A similar problem is observable in Experiment 9. Using a deflation tolerance of 10^{-7} will lead to deflation only in the basis vector $\mathbf{y}_5$. However using a deflation tolerance larger than 10^{-5} results in a second deflation in $\mathbf{y}_{20}$. The method then fails to achieve an accuracy of 10^{-10} with respect to the relative residual. It is a problem that deflation is too early for that kind of problems discussed in Chapter 4, that is, the demanded accuracy is not yet achieved when deflation takes place. However, deflation has to be applied immediately otherwise orthogonality is abruptly lost.

9.3. Applications

9.3.1. Convex quadratic programming

The goal of quadratic programming is to minimize a certain function f that might represent a risk or costs. Based on quadratic programming Harry Markowitz introduced an approach for portfolio optimization in his paper "Portfolio Selection" which appeared in the 1952 Journal of Finance. Thirty-eight years later, he shared a Nobel Prize with Merton Miller and William Sharpe for what has become a broad theory for portfolio selection.

A typical convex quadratic programming problem is

$$\min_{\mathbf{x}\in\mathbb{C}^n} f(\mathbf{x}) = \frac{1}{2}\langle \mathbf{x}, \mathbf{H}\mathbf{x}\rangle + \langle \mathbf{g}, \mathbf{x}\rangle \quad \text{subject to} \quad \mathbf{J}\mathbf{x} = \mathbf{t} \tag{9.1}$$

where $\mathbf{H} \in \mathbb{C}^{n\times n}$ is Hermitian positive definite, $\mathbf{B} \in \mathbb{C}^{m\times n}$ is the full row rank matrix of linear constraints and vectors $\mathbf{g}$ and $\mathbf{t}$ have appropriate dimensions. Any finite solution of (9.1) is a stationary point of the Lagrangian function

$$L(\mathbf{x}, \boldsymbol{\mu}) = \frac{1}{2}\langle \mathbf{x}, \mathbf{H}\mathbf{x}\rangle + \langle \mathbf{g}, \mathbf{x}\rangle - \langle \boldsymbol{\mu}, \mathbf{J}\mathbf{x} - \mathbf{t}\rangle$$

where the entries of the vector $\boldsymbol{\mu}$ are referred to as **Lagrangian multipliers**. In the stationery point the partial derivatives of L with respect to $\mathbf{x}$ and $\boldsymbol{\mu}$ vanish, that is

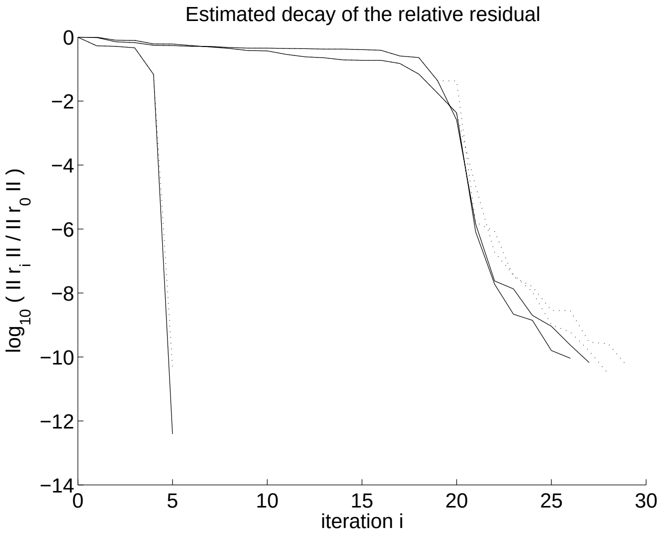


Figure 9.9.: Experiment 9: The relative residuals for block MinRes (solid lines) and MinRes (dotted lines) using a deflation tolerance of 10^{-6}.

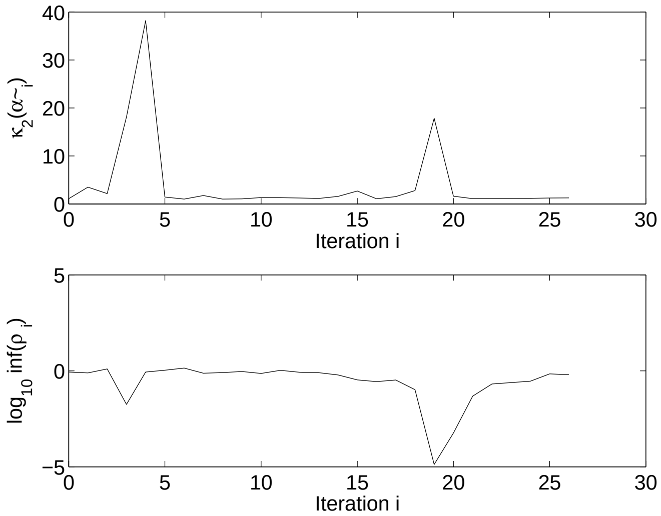


Figure 9.10.: Experiment 9: Deflation only in the basis vector $\mathbf{y}_5$. Compare with Figure 4.38.

in the stationary point $n + m$ linear equations are satisfied. These are known as the Karush-Kuhn-Tucker (KKT) conditions:

$$\begin{pmatrix} \mathbf{H} & \mathbf{J}^{\mathsf{H}} \\ \mathbf{J} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{x} \\ -\boldsymbol{\mu} \end{pmatrix} = \begin{pmatrix} -\mathbf{g} \\ \mathbf{t} \end{pmatrix} = \mathbf{B} \tag{9.2}$$

In the model of Markowitz the coefficients of the vector $\mathbf{g}$ represent the estimated reward for a certain investment. Hence with a block method it would be possible to compute different configurations at once. However, before we get lost in financial details we solve a selection of problems[1] from the CUTEr collection [10][2]

Experiment 15 *Let* $\mathbf{A}$ *the* 210×210 *matrix DPKL01 from the CUTEr collection [10] and let* $\mathbf{B}$ *a block vector of* 5 *random columns with entries uniformly distributed in interval* $[0, 1]$ *Here block* MinRes *is faster by a factor slightly larger than the number of right-hand sides. It takes less matrix vector multiplications to solve* 5 *systems with the block version than to solve one single system with* MinRes*.*

However, we expect problems as soon as a right-hand side is lying in a small eigenspace and inducing a small singular value after a few iterations.

Experiment 16 *Let* $\mathbf{A}$ *the* 210×210 *matrix DPKL01 from the CUTEr collection [10] and let* $\mathbf{b}$ *a block vector with 5 columns. The first column is a random linear combination of* 4 *eigenvectors, the second column is a linear combination of* 8 *eigenvectors and the remaining columns are linear combinations of* 12 *eigenvectors. We work without deflation.*

9.3.2. The discrete Stokes equations

The Stokes equations are "the" source of symmetric but indefinite linear systems. They describe a slow viscous incompressible flow in a 2-dimensional domain Ω with a Lipschitz continuous boundary Γ. In this work Ω is the unit square. The continuous Stokes problem is to find a velocity field $\mathbf{u} : \Omega \mapsto \mathbb{R}^2$ and a pressure $p : \Omega \mapsto \mathbb{R}$ satisfying:

$$\begin{aligned} -\Delta\mathbf{u} + \operatorname{grad} p &= \mathbf{f} \qquad (9.3) \\ \operatorname{div} \mathbf{u} &= 0 \quad \text{in} \quad \Omega \qquad (9.4) \end{aligned}$$

with

$$\mathbf{u} = 0 \quad \text{on} \quad \Gamma \tag{9.5}$$

[1] called DPKL01, DUAL1, DUAL2, DUAL3, GOULDQP3, MOSARQP2

[2] I thank Sue Dollar for sending me the matrices in an appropriate format

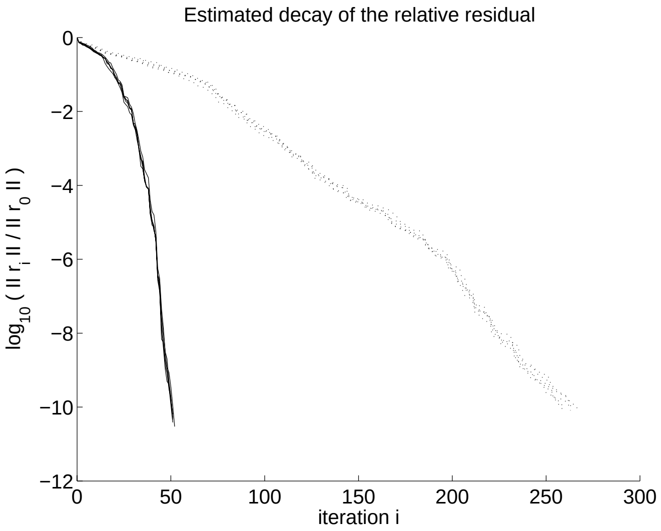


Figure 9.11.: Experiment 15: The relative residuals for block MinRes (solid lines) and MinRes (dotted lines)

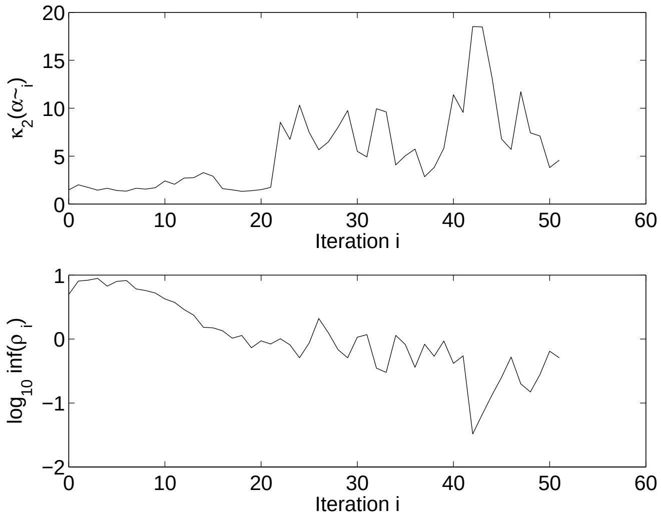


Figure 9.12.: Experiment 15: All singular values remain rather large.

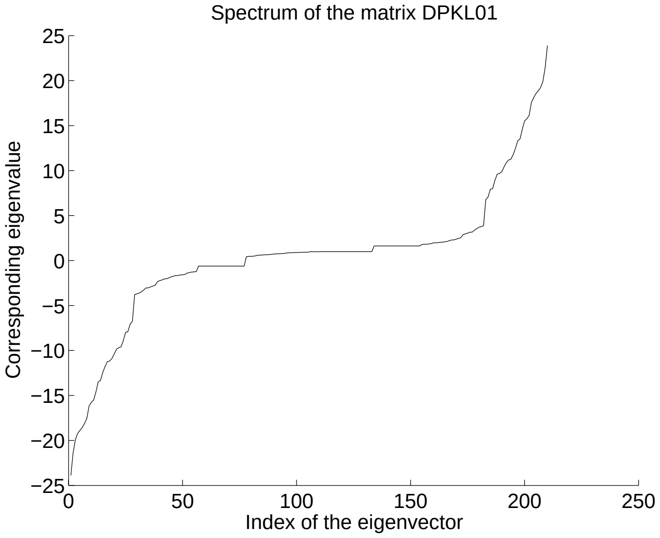


Figure 9.13.: Experiment 15: The spectrum of the matrix DPKL01

External forces are modeled by $\mathbf{f}:\Omega\mapsto\mathbb{R}^2$. We assume $\mathbf{f}\in L_2(\Omega)^2$, that is the squares of the component functions f_1 and f_2 are integrable[3]. The function space $L_2(\Omega)$ is a Hilbert space with the inner product

$$\langle p,q\rangle_{L_2} := \int_\Omega p\,q\,d(x,y) \tag{9.6}$$

The idea is to state the problem in a weaker variational form. The concept of weak formulations is beautiful. However, we can provide here only a short sketch. We define

$$C^{n,2}(\Omega) := \left(C^n(\Omega)\cap C^0(\bar{\Omega})\right)^2 . \tag{9.7}$$

If $\mathbf{u}\in C^{2,2}(\Omega)$, $p\in C^1(\Omega)$ solve (9.3)-(9.5) then $\mathbf{u}$ and p are called classical solutions. Equation (9.3) implies

$$-\int_\Omega \begin{pmatrix} \Delta u_1 & \Delta u_2 \end{pmatrix} \mathbf{v}\,d(x,y) + \int_\Omega (\operatorname{grad} p)^{\mathsf{T}}\,\mathbf{v}\,d(x,y) = \left\langle \mathbf{f}^{\mathsf{T}}\mathbf{v},1\right\rangle_{L_2} \quad \text{for all} \quad \mathbf{v}\in C^{2,2}(\Omega) \tag{9.8}$$

By defining the bilinear forms

$$\begin{aligned} a(\mathbf{u},\mathbf{v}) &:= \int_\Omega (\operatorname{grad} u_1)^{\mathsf{T}} \operatorname{grad} v_1 + (\operatorname{grad} u_2)^{\mathsf{T}} \operatorname{grad} v_2\, d(x,y) && (9.9)\\ b(\mathbf{v},q) &:= -\int_\Omega q \operatorname{div}\mathbf{v}\, d(x,y) && (9.10)\end{aligned}$$

partial integration of (9.11) yields

$$a(\mathbf{u},\mathbf{v}) + b(\mathbf{v},p) = \left\langle \mathbf{f}^{\mathsf{T}}\mathbf{v},1\right\rangle_{L_2} \quad \text{for all} \quad \mathbf{v}\in C^{2,2}(\Omega) \tag{9.11}$$

[3] Strictly speaking f_1 and f_2 are representing an equivalent class of measurable functions differing on a set of measure zero and whose squares are Lebesgue integrable.

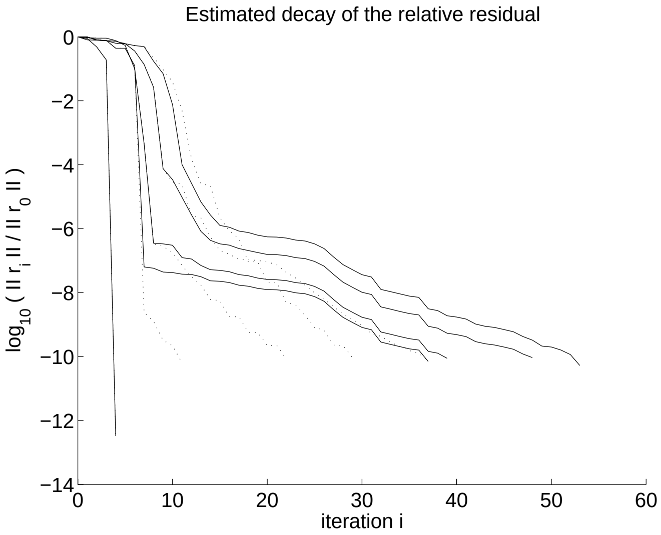


Figure 9.14.: Experiment 16: The relative residuals for block MinRes (solid lines) and MinRes (dotted lines)

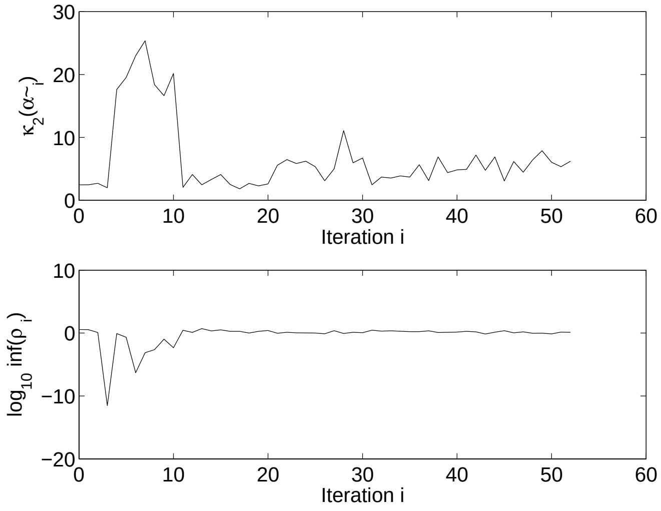


Figure 9.15.: Experiment 16: The small singular values are ignored.

This is already a first weaker formulation of (9.3). Here it is sufficient that $\mathbf{u} \in C^{1,2}(\Omega)$ The bilinear form a is an inner product. However, the space $C^{1,2}(\Omega)$ is not complete with respect to the norm induced by a. The closure of $C^{1,2}(\Omega)$ with respect to the norm induced by a is the Sobolev space $H^1(\Omega) \times H^1(\Omega)$. In order to justify (9.5) we work in the subspace

$$H_0^1(\Omega) = \left\{u \in H^1(\Omega) \ : u(x,y) = 0 \quad \text{on} \quad \Gamma\right\}$$

The pressure p in only defined up to constant, hence we impose the condition

$$\int_\Omega p \, d(x,y) = 0$$

to ensure uniqueness. With the same arguments as above we introduce the pressure test space

$$Q := \left\{q \in L_2(\Omega) \ : \int_\Omega q \, d(x,y) = 0\right\}$$

Then the weak formulation of (9.3)-(9.5) reads:

Find $\mathbf{u} \in H_0^1(\Omega) \times H_0^1(\Omega)$ and $p \in Q$ such that

$$\begin{aligned} a(\mathbf{u},\mathbf{v}) + b(\mathbf{v},p) &= \left\langle \mathbf{f}^{\mathsf{T}}\mathbf{v}, 1\right\rangle_{L_2} \quad \text{for all} \quad \mathbf{v} \in H_0^1(\Omega) \times H_0^1(\Omega) && (9.12) \\ b(\mathbf{u},q) &= 0 \quad \text{for all} \quad q \in Q && (9.13) \end{aligned}$$

Although this formulation is weaker in terms of restrictions to $\mathbf{u}$ and p we gained a lot. The abstract setting in Hilbert spaces enables use to use results from functional analysis for existence and uniqueness of a solution. We refer the reader to a rather complete discussion in a book by Fischer [5].

The idea of finite element methods (FEM) is to approximate the solution in finite dimensional subspaces of the testspaces $H_0^1(\Omega) \times H_0^1(\Omega)$ and Q. Here the finite element spaces consist of piecewise linear functions on a mesh with rectangular elements. We use a coarse grid for the pressure where every rectangle is subdivided into four rectangles by joining the midsides to obtain the velocity approximation. It is convenient to use nodal basis functions, that is every nodal point in the mesh corresponds with a basis function that is 1 on this nodal and 0 on the other nodals. An approximation of u_1, u_2 or p is a finite linear combination in terms of the basis functions. Here the basis functions are also used as testfunctions. Formulating the equations in the finite dimensional setting as a matrix problem is straightforward but rather tedious[4]. Details are again provided by Fischer [5]. We end up with a KKT system.

$$\mathbf{A}\begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{pmatrix} = \begin{pmatrix} \mathbf{A}_e & \mathbf{B}^{\mathsf{T}} \\ \mathbf{B} & \mathbf{0} \end{pmatrix}\begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{pmatrix} = \begin{pmatrix} \mathbf{F} \\ \mathbf{0} \end{pmatrix} = \mathbf{D} \tag{9.14}$$

However, here the matrix $\mathbf{A}_e$ is symmetric positive definite.

[4] The author would like to thank David Silvester for generating the matrices

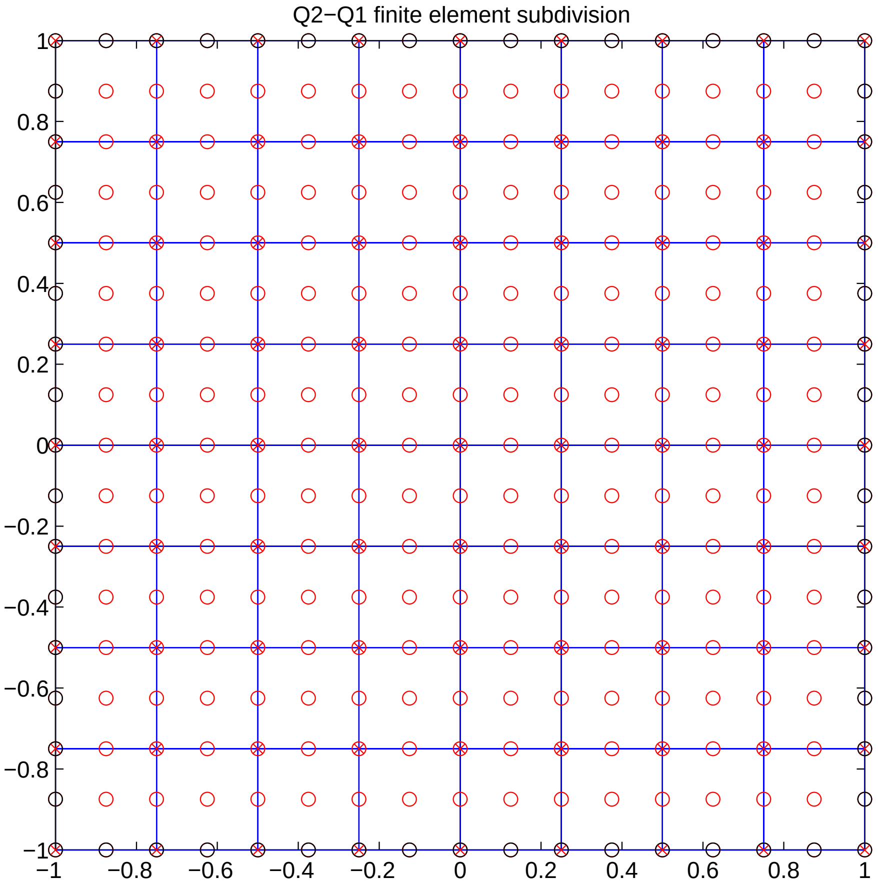


Figure 9.16.: A mesh for the discrete Stokes equation: (○) degrees of freedom for the velocity; (×) degrees of freedom for the pressure.

Because of the special form of the right-hand side $\mathbf{D}$ it is unlikely and physically unmotivated that a column of $\mathbf{D}$ is a linear combination of only a few eigenvectors of $\mathbf{A}$. It is physically unmotivated because an eigenvector of $\mathbf{A}$ does not correspond to a solution of the discrete Stokes equations and the external force $\mathbf{F}$ would be a linear combination of velocity fields corresponding to those eigenvectors. Hence MinRes does typically converge very slowly for the Stokes equation. Therefore we use preconditioning and expect a good performance of block MinRes. Due to Wathen [36] an attractive approach for the preconditioning of the Stokes equations is to use a special diagonal matrix where

$$\mathbf{M} = \begin{pmatrix} \mathbf{M_A} & \mathbf{0} \\ \mathbf{0} & \mathbf{M_Q} \end{pmatrix} \tag{9.15}$$

where $\mathbf{M_A}$ is the diagonal of $\mathbf{A}_e$ and $\mathbf{M_Q}$ is the diagonal of the pressure mass matrix $\mathbf{Q}$. The pressure mass matrix is defined by

$$\mathbf{Q} := \left[\langle M_i, M_j \rangle_{L_2}\right]_{i,j=1,\ldots,m}$$

where $\{M_j\}_{j=1}^m$ denotes a "pressure basis".

Experiment 17 *Let* $\mathbf{A}$ *the* 659×659 *matrix* (9.14) *describing the discrete Stokes on the mesh of Figure 9.16. Let* $\mathbf{F}$ *a random block vector with* 5 *columns and an appropriate number of rows. We apply a preconditioner of the form described in* (9.15)*.*

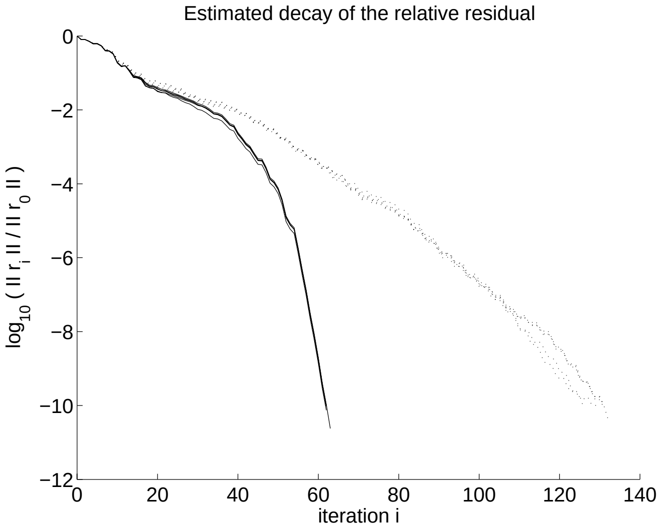


Figure 9.17.: Experiment 17: The relative residuals for block MinRes (solid lines) and MinRes (dotted lines)

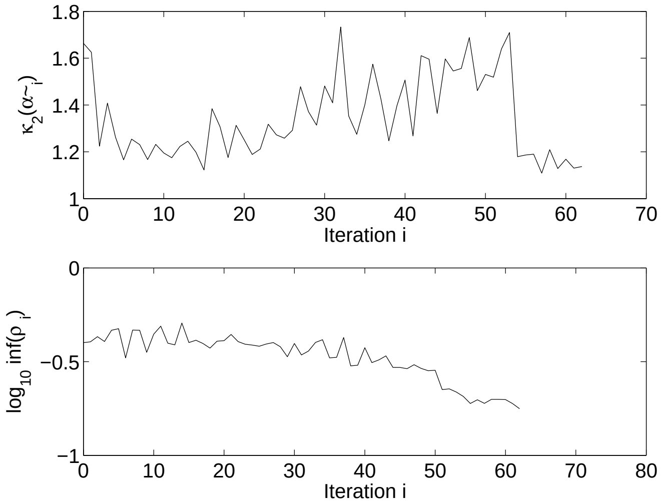


Figure 9.18.: Experiment 17: All singular values remain rather large.

10. Conclusions

It is not without risk to apply block version of MinRes and SymmLQ.

The loss of orthogonality in the ordinary Lanczos process is no tragedy. Paige [20] connected it with converged Ritz pairs. Greenbaum and Strakos [11] showed that the loss of orthogonality may have a delaying effect but does not prevent convergence in general.

Freund and Malhotra [7] state that "for block Krylov-subspace methods deflation is crucial in order to delete linearly and almost linearly dependent vectors in the underlying block Krylov sequences".

We have never been satisfied with that rather short remark. In the ordinary Lanczos process we do not care about the loss of orthogonality but in the block version we introduce a deflation scheme in order to avoid it? The effects of refrained deflation are visualized and described in Experiment 9. An abrupt loss of orthogonality is unavoidable if a subdiagonal block of the matrix $\underline{\mathbf{T}}_n$ has a small singular value, which corresponds to an "indetermined" vector. This vector completely destroys orthogonality in all subsequent steps. It extremely hampers convergence if there is convergence at all (see Figure 9.9).

We explained in which situations this problem arises. Small singular values appear if a Krylov space is exhausted or a collision of two of them is unavoidable. This depends on the decomposition of the initial residuals (which are usually the right-hand sides) in the eigenvector basis of $\mathbf{A}$. To make things more reliable you might start with a random initial approximation.

The experiments for the Arnoldi process give rise to the hope that it is not necessary to work with deflations in the block Arnoldi process which is the basis for block GMRes. There it is not crucial to delete almost linearly dependent vectors in the underlying block Krylov sequences. An augmented space might be an alternative as we saw that deflation hampers convergence also. It has adverse effects and a carefully chosen tolerance is necessary. However, here further work has to be done. An advantage of deflation is the reduction of matrix-vector multiplications per iteration.

In Experiment 7 we showed that the block Lanczos algorithm does not profit from degenerated eigenvalues to the same degree as the Lanczos process does.

In Chapter 5 we introduced an efficient way to compute the QR decomposition of the matrix $\underline{\mathbf{T}}_n$ superior to the approach used by Freund based on Givens rotations. The problem of ill-conditioned diagonal blocks described in the next chapter does not seem to be very severe. In all further experiments we could not observe any problems here.

Generalizing MinRes and SymmLQ is an exercise of multiplying matrices. Although both schemes are closely related they had different fortune in last 30 years. It seems MinRes is very popular. Probably because GMRes has gained so much attention. Yet SymmLQ had some identity problems. What is SymmLQ? Is it a scheme producing Galerkin approximations or minimal errors in the "wrong" Krylov subspace? It is both.

In our experiments we could not observe any differences between block MinRes and block SymmLQ worth to get documented here. In the experiments we also observed a dramatic acceleration (see Figure 9.11) when no "indetermined" vectors disturb the iteration. For s right-hand sides it is possible to achieve a reduction of the matrix-vector multiplication by a factor slightly larger than s. This happens when the right-hand sides are linear combinations of the same eigenvectors. Furthermore orthogonality is better preserved in the block Lanczos process (see Experiment 6) which will reduce the number of matrix-vector multiplications also. However, that is the best thinkable case. Usually it is not possible to achieve the factor s.

The discrete Stokes equation is a possible area of application. Using preconditioning we observed that no Krylov subspace is exhausted already after a few steps (see Figure 9.17). In such cases a block method might be a superior alternative. Additionally the eigenvectors of discrete Stokes equations do not correspond to solutions of the continuous Stokes equations.

We highlighted the snares for block MinRes and block SymmLQ but the good news is, they seem to be rare in practice. A more reliable alternative is MinRes or SymmLQ. Both methods have extremely low memory requirements as it not necessary to store more than three Krylov basis vectors. But actually the goal was to develop an alternative for MinRes - not the other way round. A further idea might be to use block GMRes. Based on the robust block Arnoldi process we expect good results even without deflation. However the well-known drawbacks are high memory requirements (if used without restarts) and the expensive projections in every iteration.

However, if memory requirements are no issue then an approach by Parlett [22] might be a clever alternative. The idea is to solve the first system and to store all basis vectors gained in the iteration. We project the remaining right-hand sides in the space spanned by all basis vectors gained so far. This method seems to be very appealing in terms of computational work. As it is based on the ordinary Lanczos process there is no need to use rank-revealing QR-decompositions.

If no preconditioning is used, then the symmetric QMR algorithm of Freund and Nachtigal [8] and MinRes are mathematically equivalent. Hence all problems discussed here also apply for the block QMR algorithm [7]. The block QMR algorithm is based on a block version of the Non-Hermitian Lanczos process [12]. There are also various other problems in this case.

During this work we assumed that the initial start vector for the block Lanczos process is the initial residual. But it might be an option also to start with 20 vectors although there are only 5 right-hand sides. An efficient choice of those vectors is an open problem.

We counted matrix-vector multiplications by hand. Multiplying a $N \times 5$ vector by an $N \times N$ matrix are 5 matrix-vector multiplications. But does is take the 5 seconds if one matrix-vector multiplication takes one second? The answer is somewhere hidden in the memory management system of your computer.

Or in the wise words of Beresford Parlett:

> For every computing environment and every large symmetric matrix $\mathbf{A}$ there is a "magic" integer $p = p\,(\text{environment}, \mathbf{A})$ such that the cost of multiplying p vectors by $\mathbf{A}$ is less than 10 percent more than multiplying one vector by $\mathbf{A}$.

A. Appendix

Here we review basics from linear algebra and list the block Arnoldi algorithm.

A.1. Inner products and Hermitian matrices

A Hermitian form γ is a mapping $\gamma : \mathbb{C}^n \times \mathbb{C}^n \mapsto \mathbb{C}$, which fulfills:

- $\gamma\left(\mathbf{x}, \mathbf{y}\right) = \overline{\gamma\left(\mathbf{y}, \mathbf{x}\right)}$
- $\gamma\left(\lambda\mathbf{u} + \mu\mathbf{v}, \mathbf{w}\right) = \lambda\gamma\left(\mathbf{u}, \mathbf{w}\right) + \mu\gamma\left(\mathbf{v}, \mathbf{w}\right)$

In particular $\gamma\left(\mathbf{x}, \mathbf{x}\right) \in \mathbb{R}$. A Hermitian form γ is positive definite if $\gamma(\mathbf{x}, \mathbf{x}) > 0$ for all $\mathbb{C}^n \ni \mathbf{x} \neq 0$. An **inner product** is a positive definite Hermitian form. The **canonical inner product** of two vectors $\mathbf{x} = (x_1, \ldots, x_n)$, $\mathbf{y} = (y_1, \ldots, y_n) \in \mathbb{C}^n$ is defined by

$$\langle \mathbf{x}, \mathbf{y} \rangle = \sum_{i=1}^{n} \overline{x_i} y_i.$$

Two vectors x, $y \in \mathbb{C}^n$ are called **orthogonal** if

$$\langle \mathbf{x}, \mathbf{y} \rangle = 0.$$

A set of linearly independent vectors $\mathbf{a}_1, \mathbf{a}_2 \ldots \mathbf{a}_m$ in $\mathbb{C}^n$ are called **orthonormal** if

$$\langle \mathbf{a}_i, \mathbf{a}_j \rangle = \delta_{ij}.$$

Set $\mathbf{A} \in \mathbb{C}^{n \times n}$ then the adjoint matrix $\mathbf{A}^{\mathsf{H}}$ is defined such that for all $\mathbf{x}$, $\mathbf{y} \in \mathbb{C}^n$

$$\left\langle \mathbf{A}^{\mathsf{H}}\mathbf{x}, \mathbf{y} \right\rangle = \left\langle \mathbf{x}, \mathbf{A}\mathbf{y} \right\rangle.$$

A matrix $\mathbf{A}$ is **Hermitian** or **selfadjoint**if $\mathbf{A}^{\mathsf{H}} = \mathbf{A}$. The adjoint matrix is the transposed complex conjugate, that is

$$(\mathbf{A}^{\mathsf{H}})_{ij} = \overline{(\mathbf{A})_{ji}} \tag{A.1}$$

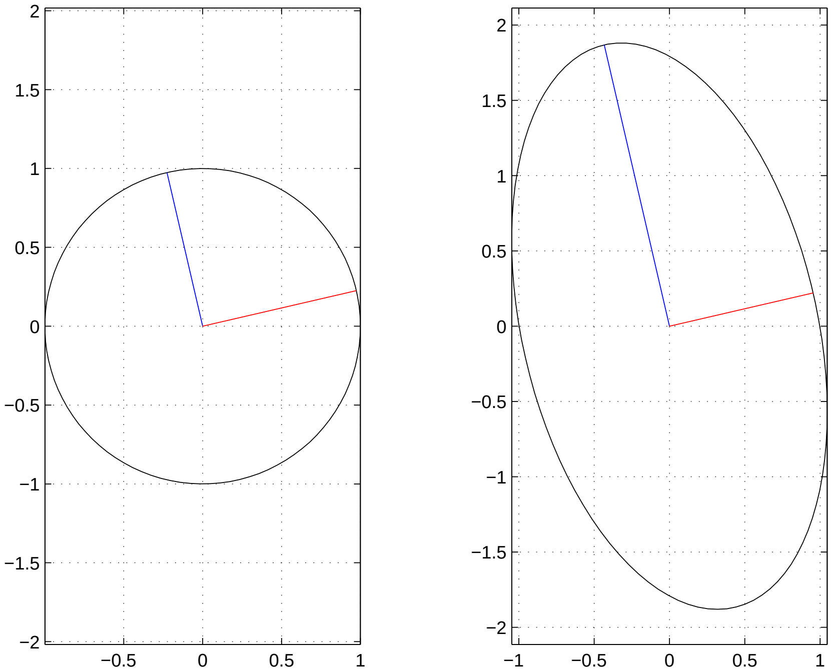
The form $\langle \mathbf{x}, \mathbf{A}\mathbf{y} \rangle$ is Hermitian if and only if $\mathbf{A}$ is Hermitian. A Hermitian matrix $\mathbf{A}$ is positive definite if and only if the form $\langle \mathbf{x}, \mathbf{A}\mathbf{y} \rangle$ is positive definite.

A.2. The singular value decomposition

An Hermitian matrix $\mathbf{A}$ maps the unit ball on a hyperellipsoid E defined by

$$E = \{\mathbf{Ax} \,:\, \|\mathbf{x}\|_2 = 1\}\,. \tag{A.2}$$

The mapped orthonormal eigenvectors of $\mathbf{A}$ are semi-axes of E. The length of a semi-axes is the modulus of the corresponding eigenvalue. The singular value decomposition generalizes this idea. A set of orthonormal vectors $\mathbf{V} = \begin{pmatrix} \mathbf{v}_1 & \dots & \mathbf{v}_n \end{pmatrix} \in \mathbb{C}^{n\times n}$ called **right singular vectors** is mapped by a complex rectangular $m \times n$ matrix $\mathbf{A}$ on a set of orthonormal vectors $\mathbf{U} = \begin{pmatrix} \mathbf{u}_1 & \dots & \mathbf{u}_m \end{pmatrix} \in \mathbb{C}^{m\times m}$ called **left singular vectors**, that is



(a) *The mapped orthonormal eigenvectors of a symmetric* 2×2 *matrix are semi-axes of the hyperellipsoid* E.

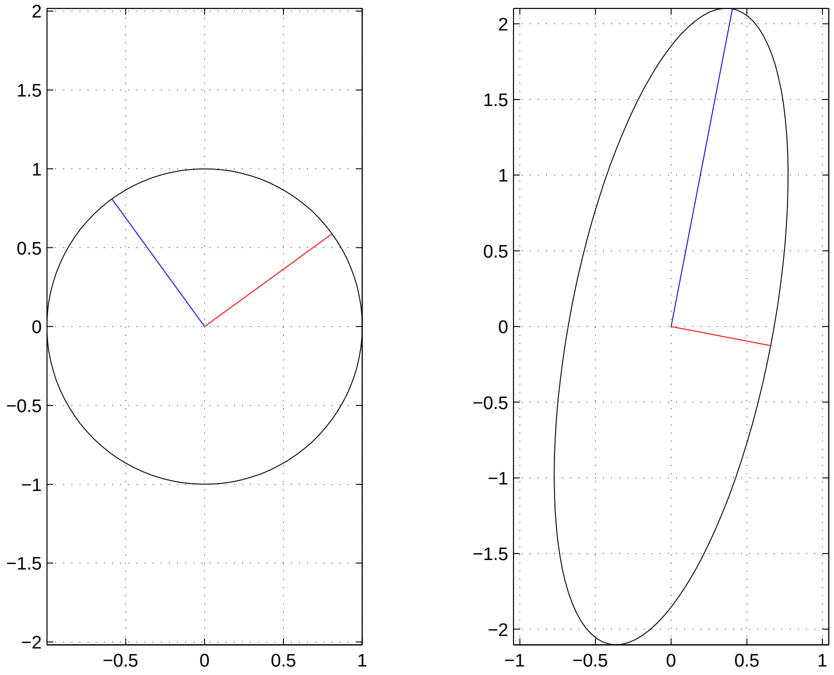
$$\mathbf{AV} = \mathbf{U}\ \mathrm{diag}\ (\sigma_1, \dots, \sigma_p) \qquad \text{where} \qquad p = \min\{m, n\} \tag{A.3}$$

The matrix $\mathrm{diag}\ (\sigma_1, \dots, \sigma_p)$ is of order $m \times n$. Here $\mathbf{u}_j \sigma_j$ where $j = 1, \dots, p$ is a semi-axis of the hyperellipsoid E. The length σ_j is non negative and called **singular value**. A proof of existence is given in Theorem 2.5.2 in [9]. It is possible and common sense to order and to align the singular vectors such that

$$\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_p \geq 0 \tag{A.4}$$

We denote by $\inf(\mathbf{R})$ the smallest singular value of a matrix $\mathbf{R}$. Hence

$$\inf(\mathbf{R}) = \min_{\substack{\mathbf{w}\in\mathbb{C}^n \\ \|\mathbf{w}\|_2 = 1}} \|\mathbf{Rw}\|_2 \tag{A.5}$$



(b) *The orthonormal right singular vectors (in the left diagram) of a non symmetric real* 2×2 *matrix are mapped on the left singular vectors which span semi-axes of the hyperellipsoid* E*.*

Corollary 14 *Let* $\mathbf{Y} \in \mathbb{C}^{N \times m}$ *a matrix with orthonormal columns then*

$$\inf(\mathbf{AY}) \geq \inf(\mathbf{A}). \tag{A.6}$$

Proof: We note $\|\mathbf{Yw}\|_2 = \|\mathbf{w}\|_2 = 1$. In particular

$$\inf(\mathbf{AY}) \geq \min_{\substack{\mathbf{u} \in \mathbb{C}^N \\ \|\mathbf{u}\|_2 = 1}} \|\mathbf{Au}\|_2 = \min\{|\lambda_1|, \ldots, |\lambda_N|\} = \inf(\mathbf{A})$$

where $\lambda_1, \ldots, \lambda_N$ is the set of eigenvalues of $\mathbf{A}$. □

Many further details are covered in the textbook by Trefethen and Bau [33].

A.3. The 2-norm condition of a matrix

The 2-norm condition[1] is a measure for the sensitivity of a linear system. It is given by the elongation of the aforementioned hyperellipsoid E. Hence

$$\kappa(\mathbf{A}) := \kappa_2(\mathbf{A}) := \frac{\sigma_1(\mathbf{A})}{\inf(\mathbf{A})} \tag{A.7}$$

[1] Throughout this work we call it sloppy just condition

If $\kappa(\mathbf{A})$ is large, then $\mathbf{A}$ is said to be ill-conditioned or close to be singular. However it remains to discuss what is large.

A.4. The block Arnoldi algorithm

The block Arnoldi algorithms is closely related to the block Lanczos process. It is not necessary that the matrix $\mathbf{A}$ is Hermitian. The Lanczos process is actually a special version of this Arnoldi algorithm. As aforementioned for Hermitian matrices $\boldsymbol{\eta}_{k,n-1} = \mathbf{0}$ if $k < n-2$. An advantage of the Lanczos algorithm is that it is not necessary to store the Krylov block basis vectors. A further drawback of the Arnoldi method are higher costs in terms of computational work. Gutknecht [14] has used the version proposed here for a block version of GMRes.

Orthogonality is better preserved if a double projection is used. Here second step in the algorithm is repeated in each loop.

Algorithm 12 (Block Arnoldi algorithm) .
Let an $N \times N$ matrix $\mathbf{A}$ and an orthonormal block vector $\mathbf{y}_0$ be given. For constructing a nested set of orthonormal block bases $\{\mathbf{y}_0, \mathbf{y}_1, \dots, \mathbf{y}_m\}$ for the nested Block Krylov subspaces $\mathcal{B}^{\square}_{m+1}(\mathbf{A}, \mathbf{y}_0)$ $(m = 1, 2, \dots \leq \bar{\nu}^{\square}(\mathbf{A}, \mathbf{y}_0) - 1)$ compute, for $n = 1, 2, \dots, m$:

1. *Apply $\mathbf{A}$ to $\mathbf{y}_{n-1} \perp \mathcal{B}^{\square}_{n-1}(\mathbf{A}, \mathbf{y}_0)$:*

$$\widetilde{\mathbf{y}}_n := \mathbf{A}\mathbf{y}_{n-1}. \tag{A.8}$$

2. *Project $\widetilde{\mathbf{y}}_n$ onto the basis block vectors $\mathbf{y}_0, \dots \mathbf{y}_{n-1}$, for $k = 0, 1, \dots, n-1$:*

$$\boldsymbol{\eta}_{k,n-1} := \mathbf{y}_k^{\mathsf{H}} \widetilde{\mathbf{y}}_n \tag{A.9}$$

$$\widetilde{\mathbf{y}}_n := \widetilde{\mathbf{y}}_n - \mathbf{y}_k \boldsymbol{\eta}_{k,n-1} \tag{A.10}$$

3. *QR factorization of $\widetilde{\mathbf{y}}_n \perp \mathcal{B}^{\square}_n(\mathbf{A}, \mathbf{y}_0)$ with $\operatorname{rank} \widetilde{\mathbf{y}}_n = \mathsf{s}_n \leq \mathsf{s}_{n-1}$:*

$$\widetilde{\mathbf{y}}_n =: \begin{pmatrix} \mathbf{y}_n & \mathbf{y}^{\Delta}_n \end{pmatrix} \begin{pmatrix} \boldsymbol{\rho}_n & \boldsymbol{\rho}^{\square}_n \\ \mathbf{0} & \boldsymbol{\rho}^{\Delta}_n \end{pmatrix} \boldsymbol{\pi}^{\mathsf{T}}_n =: \begin{pmatrix} \mathbf{y}_n & \mathbf{y}^{\Delta}_n \end{pmatrix} \begin{pmatrix} \boldsymbol{\beta}_{n-1} \\ \boldsymbol{\beta}^{\Delta}_{n-1} \end{pmatrix}, \tag{A.11}$$

where:

$\boldsymbol{\pi}_n$	*is an $\mathsf{s}_{n-1} \times \mathsf{s}_{n-1}$ permutation matrix.*
$\mathbf{y}_n$	*is an $N \times \mathsf{s}_n$ matrix with full numerical column rank going into the basis.*
$\mathbf{y}^{\Delta}_n$	*is an $N \times (\mathsf{s}_{n-1} - \mathsf{s}_n)$ matrix that will be deflated,*
$\boldsymbol{\rho}_n$	*is an $\mathsf{s}_n \times \mathsf{s}_n$ upper triangular, nonsingular matrix.*
$\boldsymbol{\rho}^{\square}_n$	*is an $\mathsf{s}_n \times (\mathsf{s}_{n-1} - \mathsf{s}_n)$ matrix.*
$\boldsymbol{\rho}^{\Delta}_n$	*is an upper triangular $(\mathsf{s}_{n-1} - \mathsf{s}_n) \times (\mathsf{s}_{n-1} - \mathsf{s}_n)$ matrix.* *It is $\|\boldsymbol{\rho}^{\Delta}_n\|_F = O(\sigma_{\mathsf{s}_n+1})$, where σ_{s_n+1} is the largest singular value of $\widetilde{\mathbf{y}}_n$ smaller than* tol.

A.5. The eigenvalues of the Poisson matrix

Throughout large parts of this work we have used the discrete Laplacian or Poisson matrix as a model problem. We consider the domain $\Omega = (0,1)^2$ with boundary $\delta\Omega$. The classic Poisson equation is

$$-\Delta u = f \tag{A.12}$$

where Δ is the Laplacian

$$\Delta = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \tag{A.13}$$

and $f : \Omega \cup \delta\Omega \to \mathbb{R}$ is a continuous function. The homogeneous Poisson equation, e.g. $f \equiv 0$ is known as the Laplace equation. We apply Dirichlet boundary conditions, that is,

$$u \equiv 0 \quad \text{on} \quad \partial\Omega.$$

An uniform mesh with mesh size h is introduced. So

$$(x_r, y_s) := (rh, sh)$$

with $r, s = 0, 1, \dots, n+1 = \frac{1}{h}$. With U_{rs} we associate $u(x_r, y_s)$. We use a central difference scheme to gain Δ_h - the discrete version of Δ.

$$(\Delta_h U)_{rs} = \frac{+U_{r+1,s} + U_{r-1,s} + U_{r,s+1} + U_{r,s-1} - 4U_{rs}}{h^2} \tag{A.14}$$

Therefore the Poisson equation $\Delta u = f$ with $u \equiv 0 \quad$ on $\partial\Omega$ becomes

$$-(\Delta_h U)_{rs} = f_{rs} \qquad \forall\, r, s = 1, \dots, n \tag{A.15}$$

and

$$U_{rs} = 0 \qquad \forall\, r, s = 0, n+1. \tag{A.16}$$

Using the lexicographic ordering[2] for the uniform mesh we obtain

$$-\mathbf{MU} = h^2\mathbf{f} \tag{A.17}$$

with

$$\mathbf{M} = \begin{pmatrix} \mathbf{T} & \mathbf{I} & & & \\ \mathbf{I} & \mathbf{T} & \mathbf{I} & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & \mathbf{I} \\ & & & \mathbf{I} & \mathbf{T} \end{pmatrix}$$

where $\mathbf{I}$ is the $n \times n$ unit matrix. $\mathbf{T}$ is of the same size and has the form

$$\mathbf{T} = \begin{pmatrix} -4 & 1 & & \\ 1 & -4 & \ddots & \\ & \ddots & \ddots & 1 \\ & & 1 & -4 \end{pmatrix}.$$

[2] see [32, Page 68]. The matrix $U_{r,s}$ is reshaped by mapping $U_{r,s}$ to U_{sr+s}

In this work $-\mathbf{M}$ is the discrete Laplacian. The eigenvalues of $-\mathbf{M}$ are

$$\lambda_{r,s} = 2\left(2 - \cos r\pi h - \cos s\pi h\right) \quad r, s = 1, \ldots, n. \tag{A.18}$$

It is enough to show that the vector

$$\mathbf{v}_{i,j} = \sin\left(ir\pi h\right)\sin\left(js\pi h\right) \quad i, j = 1, \ldots, n. \tag{A.19}$$

is a corresponding eigenvector, i.e.

$$-\mathbf{v}_{i+1,j} - \mathbf{v}_{i-1,j} - \mathbf{v}_{i,j+1} - \mathbf{v}_{i,j-1} + 4\mathbf{v}_{i,j} = \lambda_{r,s}\mathbf{v}_{i,j} \quad i, j = 1, \ldots, n.$$

Using the identity

$$\sin\left(\alpha + \beta\right)\sin\left(\alpha - \beta\right) = 2\sin\alpha\cos\beta.$$

the result is obvious.

Bibliography

[1] R. Barrett, M. Berry, T. F. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, and H. A. van der Vorst, **Templates for the solution of linear systems: Building blocks for iterative methods**, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 1994.

[2] M. Benzi and M. Tuma, **Preconditioning symmetric indefinite linear systems**, XV Householder Symposium, Peebles, Scotland, June 17–21 (2002).

[3] T. F. Chan, **Rank revealing QR factorizations**, Numerical Linear Algebra with Applications **88/89** (1987), 67–82.

[4] A. Edelman, **Eigenvalue roulette and random test matrices**, Linear Algebra for Large Scale and Real-Time Applications (M. S. Moonen, G. H. Golub, and B. L. R. De Moor, eds.), NATO ASI Series, 1992, pp. 365–368.

[5] B. Fischer, **Polynomial based iteration methods for symmetric linear systems**, Wiley-Teubner, Chichester, Stuttgart, 1996.

[6] R. W. Freund, **QR Zerlegung im Lanczos Prozess**, private note, 2004.

[7] R. W. Freund and M. Malhotra, **A block QMR algorithm for non-Hermitian linear systems with multiple right-hand sides**, Linear Algebra and Its Applications **254** (1997), 119–157.

[8] R. W. Freund and N. M. Nachtigal, **A new Krylov-subspace method for symmetric indefinite linear systems**, Proceedings of the 14th IMACS World Congress on Computational and Applied Mathematics (W. F. Ames, ed.), IMACS, 1994, pp. 1253–1256.

[9] G. H. Golub and C. F. van Loan, **Matrix computations**, third ed., Johns Hopkins University Press, Baltimore, MD, USA, 1996.

[10] N. Gould, D. Orban, and P. Toint, **CUTEr, a constrained and unconstrained testing environment, revisited**, Tech. Report (2001).

[11] A. Greenbaum and Z. Strakos, **Predicting the behavior of finite precision Lanczos and conjugate gradient computations**, SIAM J. Matrix Anal. Appl. **13** (1992), 121–137.

[12] M. H. Gutknecht, **A completed theory of the the unsymmetric lanczos process and related algorithms, part i**, SIAM J. Matrix Anal. Appl. **13** (1992), 594–639.

[13] ———, **Lecture notes: Iterative methods for linear systems**, ETH Zürich, 2003.

[14] ———, **Accuracy and effectiveness issues in block Krylov space methods**, Talk given at ICIAM 2003, Syndey.

[15] M. R. Hestenes and E. Stiefel, **Methods of conjugate gradients for solving linear systems**, Journal of Research of the National Bureau of Standards **49** (1952), 409–436.

[16] C. Lanczos, **An iteration method for the solution of the eigenvalue problem of linear differential and integral operators**, Journal of Research of the National Bureau of Standards **45** (1950), 255–282.

[17] V. I. Lebedev, **An iteration method for the solution of operator equations with their spectrum lying on several intervals**, USSR Comput. Math. and Math. Phys **9** (1969), 17–24, cited in [35].

[18] R. B. Lehoucq, **The computations of elementary unitary matrices**, ACM Trans. Math. Software **22** (1996), 393–400.

[19] N. M. Nachtigal, S. C. Reddy, and L. N. Trefethen, **How fast are nonsymmetric matrix iterations**, SIAM J. Matrix Anal. Appl. **13** (1992), 778–792.

[20] C. C. Paige, **The computation of eigenvalues and eigenvectors of very large sparse matrices**, Ph.D. thesis, University of London, 1971.

[21] C. C. Paige and M. A. Saunders, **Solution of sparse indefinite systems of linear equations**, SIAM J. Numer. Anal. (1975).

[22] B. N. Parlett, **A new look at the Lanczos algorithm for solving symmetric systems of linear equations**, Linear Algebra and its Applications **29**, 323–346.

[23] ———, **Analysis of algorithms for reflectors in bisectors**, SIAM Review **13** (1971), no. 2, 197–208.

[24] ———, **Very early days of matrix computations**, SIAM News **36** (2003), no. 9, 2 and 10.

[25] P. C. Hansen R. D. Fierro and P. S. K. Hansen, **UTV Tools: MATLAB templates for rank-revealing UTV decompositions**, Numerical Algorithms **20** (1999), 165–194.

[26] Y. Saad, **Iterative methods for sparse linear systems**, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2003.

[27] Y. Saad and M. Schultz, **GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems**, SIAM J. Sci. Stat. Comput. **7** (1986), 856–869.

[28] Y. Saad and H. A. van der Vorst, **Iterative solution of linear systems in the 20th century**, J. Comp. Appl. Math. **123** (2000), 1–33.

[29] G. L. G. Sleijpen, H. A. van der Vorst, and J. Modersitzki, **The main effects of rounding errors in krylov solvers for symmetric definite linear systems**, SIAM J. Matrix Anal. Appl. **22** (2000), no. 3, 726–751.

[30] G. W. Stewart, **Afternotes goes to graduate school**, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 1998.

[31] ______, **Matrix algorithms I: Basic decompositions**, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 1998.

[32] L. N. Trefethen, **Spectral methods in matlab**, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2000.

[33] L. N. Trefethen and D. Bau, III, **Numerical linear algebra**, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 1997.

[34] H. A. van der Vorst, **Iterative Krylov methods for large linear systems**, Cambridge University Press, Cambridge, UK, 2003.

[35] A. J. Wathen, B. Fischer, and D. J. Silvester, **The convergence of iterative solution methods for symmetric and indefinite linear systems**, Numerical Analysis 1997 (D.F. Griffiths and G.A. Watson, eds.), Pitman Research Notes in Mathematics Series, Addison Wesley Longman, Harlow, England, 1997, pp. 230–243.

[36] A. J. Wathen and D. J. Silvester, **Fast iterative solution of stabilised stokes systems part ii: using simple diagonal preconditioners**, SIAM J. Numer. Anal. **30** (1993), 630–649.

[37] J. H. Wilkinson, **The algebraic eigenvalue problem**, Oxford University Press, 1965.

Ich versichere, dass ich diese Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen als Hilfsmittel benutzt habe.

Zürich am Montag, den 30. August 2004

Thomas Schmelzer