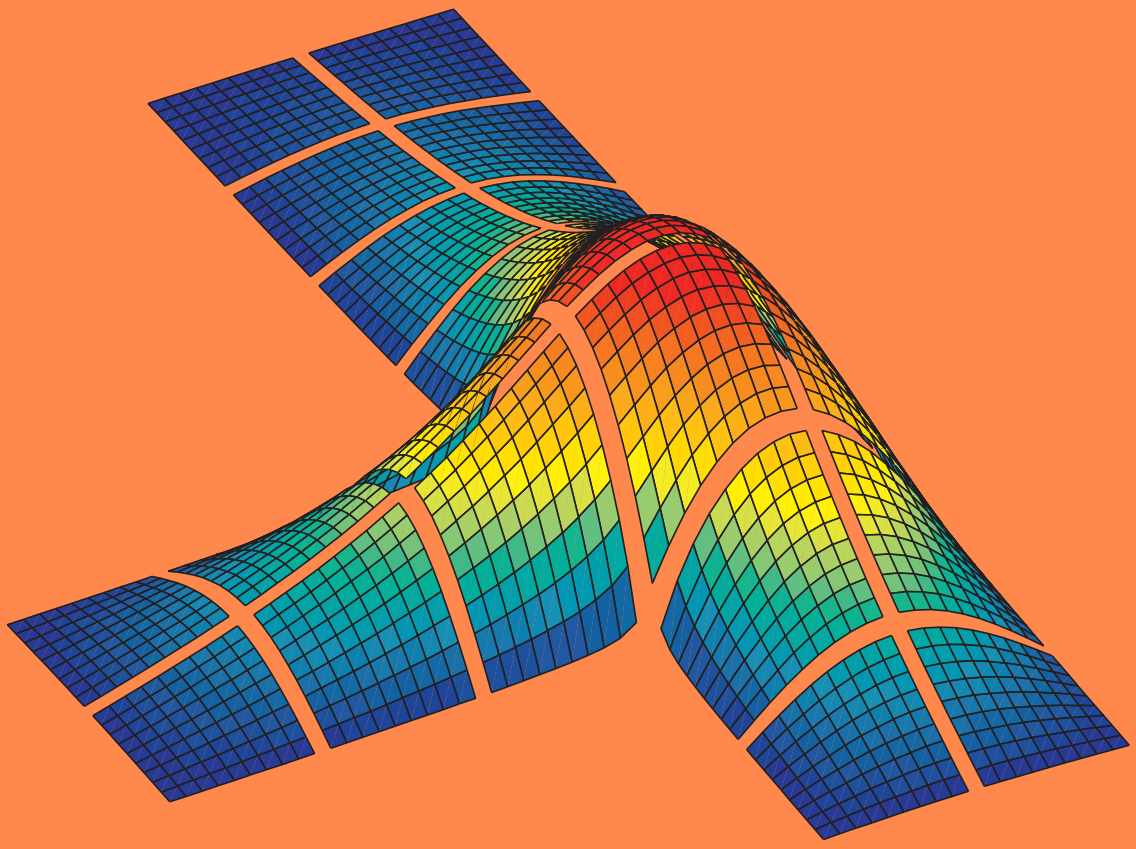
# Domain decomposition in the Jacobi-Davidson method for eigenproblems



Menno Genseberger

# Domain decomposition in the Jacobi-Davidson method for eigenproblems

Domeindecompositie in
de Jacobi-Davidson methode
voor eigenproblemen

(met een samenvatting in het Nederlands)

Proefschrift

ter verkrijging van de graad van doctor aan de Universiteit Utrecht
op gezag van de Rector Magnificus, Prof. dr. W. H. Gispen,
ingevolge het besluit van het College voor Promoties in het openbaar
te verdedigen op maandag 10 september 2001 des middags te 12.45 uur

door

Menno Genseberger

geboren op 8 september 1972, te Amsterdam

Promotor: Prof. dr. H. A. van der Vorst
Co-promotor: dr. G. L. G. Sleijpen

Faculteit der Wiskunde en Informatica
Universiteit Utrecht

Het onderzoek beschreven in dit proefschrift komt voort uit een samenwerkingsverband tussen het Mathematisch Instituut van de Universiteit Utrecht en de cluster Modelling, Analysis and Simulation (MAS) van het Centrum voor Wiskunde en Informatica (CWI) in Amsterdam. Financieel is het mogelijk gemaakt door de Nederlandse Organisatie voor Wetenschappelijk Onderzoek (NWO).

---

2000 Mathematics Subject Classification: 65F15, 65N25, 65N55

---

Genseberger, Menno
Domain decomposition in the Jacobi-Davidson method for eigenproblems
Proefschrift Universiteit Utrecht - Met een samenvatting in het Nederlands.

---

ISBN 90 6196 507 1

---

# Contents

# Chapter 1

# Introduction

Large scale eigenvalue problems play an important role in the scientific investigation of a variety of phenomena. The different phenomena in question are not only of interest to scientists, but they are also frequent news items as e.g. climate change and earthquakes. For the computation of solutions to large scale eigenvalue problems the last two decades considerable progression has been made in the development of numerical methods. But still research needs to be done for these methods. This thesis concerns an approach to relieve the amount of computational work for one of the most attractive methods.

This introduction starts with a brief sketch of the background of those large scale eigenvalue problems and indicates some typical characteristics (§1.1). Then an overview is given of methods for the numerical computation of solutions to eigenvalue problems (§1.2). The major computational component of the method on which this thesis focusses is some special kind of linear system. Therefore, also numerical methods for solving linear systems are considered in appropriate detail (§1.3). Emphasis will be on the construction of a preconditioner based on domain decomposition (§1.4). After this survey the thesis is outlined in §1.5.

## 1.1 The origin of eigenvalue problems

Eigenvalue problems show up in a diversity of scientific disciplines. For an impression of its importance we give a short list of applications:

- rotating plasma equilibria in tokamaks (fusion research, see for instance [27, 30])
- states and interactions of particles in quantum chemistry or molecular physics (for instance energy states of atoms [36], energy states of molecules [13], laser-molecule interactions [64], and molecular dynamics [1])
- effects of earthquakes on buildings (structural engineering, for a recent paper see for instance [62])
- stability analysis of ocean circulation patterns - ocean circulation is expected to be an eminent factor in climate variability (climatology/oceanography, for a recent paper see for instance [20])
- coronal loops of the sun (astrophysics, see for instance [3, 30])

See for an illustration the supplementary card that is included in this thesis.

In most situations the object of study is described by some partial differential equation. For a numerical treatment the concerning equations are discretized. The partial differential equations of some of these phenomena constitute an eigenvalue problem in itself, for the others stability analysis of solutions of the concerning equations leads to an eigenvalue problem, which in standard form is represented as:

$$\mathbf{A}\,\mathbf{x} = \lambda\,\mathbf{x}. \tag{1.1}$$

Other types of eigenvalue problems are also possible, e.g. the generalized one $\mathbf{A}\,\mathbf{x} = \lambda\,\mathbf{B}\,\mathbf{x}$. Note that many of them can be written in standard form. We will restrict ourselves to standard eigenproblems.

Characteristic properties of (1.1) are:

- large dimensions: realistic modelling and simulations require a fine grid discretization, resulting in an eigenvalue problem with a large number (in the order of millions) of unknowns,
- sparse and banded matrices: discretization of differential operators (for instance via finite differences) leads to sparse or banded linear operators,
- many applications require knowledge of a few (extremal) eigenpairs only.

The first property has put heavy pressure on the recent development of numerical methods for (1.1) whereas the two other properties somewhat relief the pain. This will be discussed next.

## 1.2 Numerical methods for eigenvalue problems

For the computation of solutions to an eigenvalue problem the available methods globally split into two classes:

- dense methods (like QR for the standard eigenvalue problem (1.1) [23, 24] and QZ for generalized eigenvalue problems [35])
- subspace methods (like Lanczos [31], Arnoldi [2], Davidson [15], and Jacobi-Davidson [46])

The first class suits well for eigenvalue problems with relatively small dense matrices. In general these methods compute all eigenvalues and the corresponding eigenvectors. As the amount of computational work is of order $n^3$ and the required memory of order $n^2$, they become impractical for large $n$-dimensional eigenvalue problems. In those situations one may address a subspace method. Such a method condenses relevant information of the high dimensional eigenvalue problem by projecting it on a subspace of relatively small dimension. It computes an approximate solution to the large eigenvalue problem from the small projected eigenvalue problem by means of a dense method.

Next we describe the Jacobi-Davidson method and we will show how the subspace methods Arnoldi and Davidson (we will consider the Lanczos method as a special case of Arnoldi) are related.

### 1.2.1 Accelerated inexact Newton methods

The Jacobi-Davidson method can be viewed as an accelerated inexact Newton method [25]. Typically, an accelerated inexact Newton method consists of the following components:

- projection of the original problem on a search subspace and computation of an approximate solution from the projected problem,
- expansion of the search subspace with a new search direction obtained from a correction equation.

The first component is the accelerating part: not only the most recent search direction but also previous search directions are taken into consideration for the computation of a new approximate solution to the original problem.
The second component forms the inexact Newton part: the correction equation describes a Newton step. In general it is inexact as only approximate solutions of the correction equation are computed, for instance with some iterative method.

In practice, accelerated inexact Newton methods are very effective because often rather accurate approximate solutions to a large problem can be obtained from a projected problem of relatively small size. Then it is essential how the search subspace is constructed. This construction is described iteratively by the correction equation and therefore the correction equation deserves special attention.

The correction equation "connects" the original problem with the projected problem. It tries to describe a correction vector with important characteristic behavior of the original problem not already present in the projected problem. Expansion of the search subspace with such a correction vector then improves the projected system: a better approximate solution can be obtained from the original problem projected on the new search subspace.

Many accelerated Newton type methods have a correction equation that involves a linear operator of equal dimension as the original problem. Then, for large scale problems the computation of solutions to the correction equation may become a computational bottleneck. In those situations exact solutions can be impractical because of time and/or memory limitations. Fortunately, also approximate solutions of the correction equation can be used for the expansion of the search subspace. The accuracy of such a solution affects the quality of the expansion: in general a crude approximation is less effective than an approximation close to the exact solution. As a direct consequence the accuracy of the new approximate solution to the original problem depends on the accuracy of the approximate solution of the correction equation.

### 1.2.2 The correction equation of Jacobi-Davidson

The main difference of Jacobi-Davidson with Arnoldi and Davidson is the correction equation. All three methods compute an approximate solution $(\theta, \mathbf{u})$ to an exact solution $(\lambda, \mathbf{x})$ of the eigenvalue problem (1.1) by projecting (1.1) on the search subspace. Without loss of generality $\mathbf{u}$ is assumed to be normalized.

Arnoldi's method has no correction equation: mathematically it simply expands the search subspace with the residual $\mathbf{r} \equiv (\mathbf{A} - \theta \mathbf{I})\ \mathbf{u}$.

In his original paper [15], Davidson suggested to precondition this residual first by the diagonal $\mathbf{D}$ of $\mathbf{A}$, shifted by $-\theta$, and expand the search subspace with the resulting vector $(\mathbf{D} - \theta \mathbf{I})^{-1}\ \mathbf{r}$. This idea was later refined to more general preconditioners $\mathbf{M} \approx \mathbf{A} - \theta \mathbf{I}$ (see for instance [33, 42, 14, 52, 53]).

The crucial distinction between Davidson and Jacobi-Davidson is that the latter imposes an extra requirement to keep things properly during the iteration process. This requirement is based on the observation that the operator $\mathbf{A} - \theta \mathbf{I}$ becomes singular when $\theta$ converges to an eigenvalue $\lambda$. An accurate preconditioner $\mathbf{M} \approx \mathbf{A} - \theta \mathbf{I}$ then also suffers from becoming singular and preconditioning by just performing the inverse action of $\mathbf{M}$ is not proper. The Jacobi-Davidson method proposes the following remedy: compute a correction vector $\mathbf{t} \perp \mathbf{u}$ from

$$(\mathbf{I} - \mathbf{u}\mathbf{u}^*)(\mathbf{A} - \theta \mathbf{I})(\mathbf{I} - \mathbf{u}\mathbf{u}^*)\ \mathbf{t} = -\mathbf{r}. \tag{1.2}$$

For interpretation of what is going on in (1.2), first consider the asymptotic case $\theta = \lambda$, for a simple eigenvalue $\lambda$ and $\mathbf{u} = \mathbf{x}$ in the projections on the left-hand side but $\mathbf{u} \neq \mathbf{x}$ in $\mathbf{r} \equiv (\mathbf{A} - \theta \mathbf{I})\ \mathbf{u}$ on the right-hand side. (The argument does also hold for $\mathbf{u} \neq \mathbf{x}$ on the left-hand side (see the original discussion in [46]), for ease of presentation we consider a less general situation here.) Then formally the operator on the left in (1.2) acts in the space orthogonal to the eigenvector $\mathbf{x}$ and, as this is precisely the direction in which $\mathbf{A} - \lambda \mathbf{I}$ is

singular, is well defined there. For this asymptotic case the exact solution is equal to

$$\mathbf{t} = -\left(\mathbf{A} - \lambda\,\mathbf{I}\right)^{\dagger}\,\mathbf{r} = -\left(\mathbf{A} - \lambda\,\mathbf{I}\right)^{\dagger}\,\left(\mathbf{A} - \lambda\,\mathbf{I}\right)\,\mathbf{u}, \tag{1.3}$$

here $^{\dagger}$ denotes the pseudo-inverse of a matrix. If one takes a closer look on the right-hand side of (1.3) the following can be observed: the operator $\left(\mathbf{A} - \lambda\,\mathbf{I}\right)$ removes the component from $\mathbf{u}$ in the direction of $\mathbf{x}$ and the operator $\left(\mathbf{A} - \lambda\,\mathbf{I}\right)^{\dagger}$ transforms all remaining components back. As a consequence, with the $\mathbf{t}$ from (1.3) the update $\mathbf{u} + \mathbf{t}$ is equal to $\gamma\,\mathbf{x}$ for some scalar $\gamma$ and the eigenvector one is looking for is obtained in just one step.

In practice $(\theta, \mathbf{u}) \approx (\lambda, \mathbf{x})$ and to facilitate interpretation of (1.2) for those situations the correction equation is first written in its augmented formulation [44, §3.4]:

$$\begin{bmatrix} \mathbf{A} - \theta\,\mathbf{I} & \mathbf{u} \\ \mathbf{u}^{*} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{t} \\ \varepsilon \end{bmatrix} = \begin{bmatrix} -\mathbf{r} \\ 0 \end{bmatrix}. \tag{1.4}$$

Here the requirement $\mathbf{t} \perp \mathbf{u}$ is represented explicitly in the last row of the augmented matrix. In view of an inexact Newton method applied to a general nonlinear problem (an eigenvalue problem is weakly nonlinear) $\varepsilon$ can be interpreted as a stepsize control variable, its value tends to zero when convergence to a "limit point" $(\lambda, \mathbf{x})$ takes place and this prevents the augmented matrix in (1.4) from getting singular. This strategy of adding an extra constraint to retain stability is also known as arc length method (for a recent application see for instance [38]).

The Jacobi-Davidson method belongs to that class of accelerated inexact Newton methods for which the dimension of the correction equation is proportional to the dimension of the original problem. For those methods it was mentioned in §1.2.1 that, especially for large scale problems, the computation of exact solutions to the correction equation may become impractical and it remains to compute approximate solutions. The correction equation of Jacobi-Davidson is a special type of linear system. First for ordinary linear systems it will be discussed how approximate solutions can be computed with Krylov subspace methods.

## 1.3 Krylov subspace methods for linear systems

For the computation of approximate solutions to large sparse linear systems Krylov subspace methods have become very popular. For many applications, a Krylov subspace method in combination with a preconditioner may be a good alternative for direct methods.

Given a linear system

$$\mathbf{A}\,\mathbf{x} = \mathbf{b}, \tag{1.5}$$

a Krylov subspace method generates iteratively a Krylov subspace $\mathcal{K}_m\left(\mathbf{A}, \mathbf{v}_0\right)$ built by powers of matrix $\mathbf{A}$ applied to the startvector $\mathbf{v}_0$:

$$\mathcal{K}_m\left(\mathbf{A}, \mathbf{v}_0\right) \equiv \text{span}\left(\mathbf{v}_0, \mathbf{A}\,\mathbf{v}_0, \mathbf{A}^2\,\mathbf{v}_0, \ldots, \mathbf{A}^{m-1}\,\mathbf{v}_0\right),$$

and computes an approximate solution to (1.5) with respect to $\mathcal{K}_m(\mathbf{A}, \mathbf{v}_0)$. There are different choices possible for the computation of the approximate solution and so there are different Krylov subspace methods, like CG, BiCG, QMR, GCR, Bi-CGSTAB, GMRES, and FOM. For example: FOM computes an approximate solution such that its residual is orthogonal to the Krylov subspace whereas GMRES minimizes the residual with respect to the Krylov subspace.

Compared to eachother Krylov subspace methods have their pros and cons. But overall, when compared with a direct method for the solution of (1.5) a considerable reduction in computational work may be achieved, particularly for large sparse $\mathbf{A}$. By taking powers of $\mathbf{A}$, dominant eigenvalues and -vectors of $\mathbf{A}$ will soon show up. So already for a small dimension these dominant eigenvectors are represented well in the Krylov subspace. If in addition these dominant eigenvectors are the main components of the exact solution of (1.5) then an approximate solution with respect to such a low dimensional Krylov subspace is quite accurate. But this ideal situation is not always the case. Then the incorporation of a preconditioner may be beneficial.

### 1.3.1 Preconditioning

When the matrix $\mathbf{A}$ of the linear system (1.5) is not easily invertible, one may try to construct a nonsingular preconditioner $\mathbf{M} \approx \mathbf{A}$ of which the inverse action can be computed more easily. Here "$\mathbf{M} \approx \mathbf{A}$" means that $\mathbf{M}$ possesses about the same important spectral properties of $\mathbf{A}$. Which spectral properties are of importance also depends on the solution of (1.5). For example, it may happen that the dominant eigenvalues and corresponding eigenvectors of $\mathbf{A}$ are too dominant and a proper damping of these components by $\mathbf{M}^{-1}$ suffices.

Application of a Krylov subspace method to the (here for simplicity left-) preconditioned linear system

$$\mathbf{M}^{-1}\,\mathbf{A}\,\mathbf{x} = \mathbf{M}^{-1}\,\mathbf{b} \tag{1.6}$$

leads to a Krylov subspace

$$\mathcal{K}'_m\left(\mathbf{M}^{-1}\,\mathbf{A}, \mathbf{v}_0\right) \equiv \text{span}\left(\mathbf{v}_0, \mathbf{M}^{-1}\,\mathbf{A}\,\mathbf{v}_0, \left(\mathbf{M}^{-1}\,\mathbf{A}\right)^2\,\mathbf{v}_0, \ldots, \left(\mathbf{M}^{-1}\,\mathbf{A}\right)^{m-1}\,\mathbf{v}_0\right).$$

Note that the systems (1.5) and (1.6) are equivalent, both yield the same solution. It is for the iterative computation of approximate solutions that a preconditioner is incorporated. Convergence of the preconditioned Krylov subspace method depends again on how fast the most important components of the solution show up in the Krylov subspace. Here the only restriction for $\mathbf{M}$ is that it is nonsingular and one may exploit this freedom to increase the speed of convergence.

Because of its importance it is stressed again that a good preconditioner heavily relies on the system which needs to be solved. For ordinary linear systems (1.5) numerous preconditioning techniques have been developped. Three classes can be distinguished:

- algebraic techniques: the preconditioner is based on algebraic properties (sparsity pattern, matrix entries, etc.) of the matrix (for instance incomplete factorizations like Incomplete Cholesky, ILU, MILU),

- domain decomposition techniques: the preconditioner is based on subproblems that originate from decomposition of the large problem into smaller coupled ones (for instance Schwarz methods), and

- multilevel techniques: the preconditioner is constructed by transformation of the matrix to different scales (for instance FFT, multigrid, wavelets).

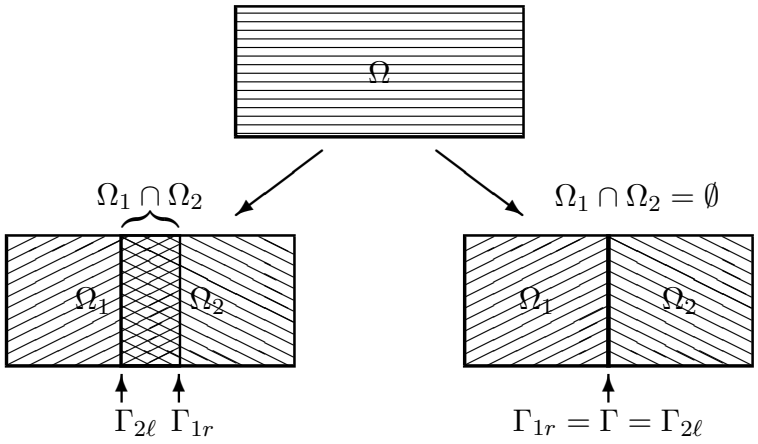
Application of an algebraic technique requires some additional properties of the original matrix (in many cases the matrix needs to be positive definite). The linear operator $\mathbf{A} - \theta \mathbf{I}$ in the correction equation (1.2) of Jacobi-Davidson is indefinite: $\mathbf{A}$ is shifted by an approximate eigenvalue $\theta$. For $\theta$ somewhere in the interior of the spectrum, $\mathbf{A} - \theta \mathbf{I}$ is even highly indefinite. This property of $\mathbf{A} - \theta \mathbf{I}$ prevents in general a succesfull application of an algebraic preconditioning technique to correction equation (1.2). An additional problem may be that the value of $\theta$ changes in each outer iteration. For a further discussion on algebraic techniques and the correction equation of Jacobi-Davidson, see for instance [47, §2.1].

Domain decomposition techniques and multilevel techniques have become popular tools. By comparing them, both types of techniques show advantages and disadvantages. Methods have been proposed that combine advantages of both techniques (see, for instance, surveys [12, 49]). There are even attempts to fit multilevel and domain decomposition in a general framework [63].

The choice in this thesis is a domain decomposition approach. Most important reason for this choice is that it enables a subdivision of a large problem into smaller (coupled) subproblems in a natural way in the following sense: there is a direct relationship between the "algebraic" subdivision of the matrix and vectors into smaller objects and the "geometric" division of the "physical" problem into subproblems. Here the matrix represents the discretization of the continuous equations that describe the "physical" problem. The kind of applications mentioned in §1.1 suits well in such an approach. Decomposition into smaller subproblems is also necessary: as already indicated the corresponding eigenvalue problems are of very large scale. Then a parallel approach to the large problem is possible, for instance on a parallel computing environment. Furthermore, the computation of solutions to the small subproblems is usually less hard.

Next the basic ideas of the common domain decomposition techniques are presented for a two subdomain case.

FIGURE 1.1. *Overlapping (left) and nonoverlapping (right) decomposition of a domain $\Omega$ into $\Omega = \Omega_1 \cup \Omega_2$.*




## 1.4 Domain decomposition

Recently the possibilities of a (massively) parallel approach with supercomputers renewed attention for domain decomposition methods. Already in the 19th century domain decomposition was considered as an approach to subdivide a hard problem in more easy to solve parts [43]. This approach forms the basis of the developments of the last two decades.

### 1.4.1 Schwarz methods

The concepts will be illustrated for a two subdomain case. Consider a physical problem described by a partial differential equation

$$\mathcal{L}\,\varphi = f. \tag{1.7}$$

Here $\varphi = \varphi(x, y)$ is defined on a two dimensional rectangular domain $\Omega$. For simplicity the conditions for $\varphi$ on the external boundary of $\Omega$ are left out of the discussion.

An *overlapping* Schwarz method (the left decomposition in Fig. 1.1) decomposes $\Omega$ in the subdomains $\Omega_1$ and $\Omega_2$ such that $\Omega_1 \cup \Omega_2 = \Omega$ and $\Omega_1 \cap \Omega_2$ is a (small) two dimensional region. On these subdomains, the system (1.7) can be written as two coupled subsystems:

$$\mathcal{L}_1\,\varphi_1 = f_1 \quad \text{on} \quad \Omega_1 \tag{1.8}$$

and

$$\mathcal{L}_2\,\varphi_2 = f_2 \quad \text{on} \quad \Omega_2. \tag{1.9}$$

The index $i$ refers to subdomain $\Omega_i$, on which subsystem (1.8), (1.9), for $i = 1, 2$, respectively, is defined. Note that the subsystems need additional internal boundary conditions. On

the two dimensional overlap $\Omega_1 \cap \Omega_2$, solutions $\varphi_1$ and $\varphi_2$ of the subsystems are equal:

$$\varphi_1|_{\Omega_1 \cap \Omega_2} = \varphi_2|_{\Omega_1 \cap \Omega_2}\,, \tag{1.10}$$

i.e. they are matched by some continuity requirement. As a consequence the value of $\varphi_1$ equals the value of $\varphi_2$ at the internal boundary $\Gamma_{1r}$ of $\Omega_1$. The same is true at the internal boundary $\Gamma_{2\ell}$ of $\Omega_2$. So appropriate internal boundary conditions for (1.8) and (1.9), respectively are

$$\varphi_1|_{\Gamma_{1r}} = \varphi_2|_{\Gamma_{1r}}$$

and

$$\varphi_2|_{\Gamma_{2\ell}} = \varphi_1|_{\Gamma_{2\ell}}\,,$$

respectively.

For the numerical computation of a solution to (1.7) (sub)domain $\Omega$ ($\Omega_i$) is covered by an appropriate (sub)grid $\widehat{\Omega}$ ($\widehat{\Omega}_i$). The operator and functions in (1.7) are discretized to $\widehat{\mathcal{L}}, \widehat{\varphi}$, and $\widehat{f}$ by means of finite differences or finite elements. Analogous to the continuous case an index refers to a subgrid. It is assumed that $\widehat{\Omega}_1 \cap \widehat{\Omega}_2 = \widehat{\Omega_1 \cap \Omega_2}$. The method consists of computing solutions on the subgrids $\widehat{\Omega}_1$ and $\widehat{\Omega}_2$ only and using values of $\widehat{\varphi}_1$ and $\widehat{\varphi}_2$ from a previous iteration on the other subgrid:

$$\widehat{\mathcal{L}}_1\, \widehat{\varphi}_1^{(i)} = \widehat{f}_1 \quad \text{on} \quad \widehat{\Omega}_1, \tag{1.11}$$

$$\widehat{\varphi}_1^{(i)}\Big|_{\widehat{\Gamma}_{1r}} = \widehat{\varphi}_2^{(i-1)}\Big|_{\widehat{\Gamma}_{1r}}\,, \tag{1.12}$$

and

$$\widehat{\mathcal{L}}_2\, \widehat{\varphi}_2^{(i)} = \widehat{f}_2 \quad \text{on} \quad \widehat{\Omega}_2, \tag{1.13}$$

$$\widehat{\varphi}_2^{(i)}|_{\widehat{\Gamma}_{2\ell}} = \widehat{\varphi}_1^{(i-1)}\Big|_{\widehat{\Gamma}_{2\ell}}\,. \tag{1.14}$$

Information from one subgrid to the other is transferred via (1.12) and (1.14). The speed of convergence of an overlapping Schwarz method depends on the amount of overlap.

A *nonoverlapping* Schwarz method (the right decomposition in Fig. 1.1) decomposes $\Omega$ into two nonoverlapping subdomains $\Omega_1$ and $\Omega_2$ such that $\Omega_1 \cup \Gamma \cup \Omega_2 = \Omega$. Here $\Gamma$ is the internal boundary that equals the right boundary $\Gamma_{1r}$ of $\Omega_1$ and the left boundary $\Gamma_{2\ell}$ of $\Omega_2$. Now, coupling between the subsystems

$$\mathcal{L}_1\, \varphi_1 = f_1 \quad \text{on} \quad \Omega_1 \tag{1.15}$$

and

$$\mathcal{L}_2\, \varphi_2 = f_2 \quad \text{on} \quad \Omega_2 \tag{1.16}$$

is maintained by the internal boundary conditions

$$\mathcal{G}\ \varphi_1|_{\Gamma_{1r}} = \mathcal{G}\ \varphi_2|_{\Gamma_{2\ell}} \tag{1.17}$$

with $\mathcal{G}$ an appropriate operator.

Discretization yields the following algorithm:
Compute $\widehat{\varphi}_1^{(i)}$ from

$$\widehat{\mathcal{L}}_1\,\widehat{\varphi}_1^{(i)} = \widehat{f}_1 \quad \text{on} \quad \widehat{\Omega}_1, \tag{1.18}$$

$$\widehat{\mathcal{G}}_{12}\;\widehat{\varphi}_1^{(i)}\Big|_{\widehat{\Gamma}_{1r}} = \widehat{\mathcal{G}}_{12}\;\widehat{\varphi}_2^{(i-1)}\Big|_{\widehat{\Gamma}_{2\ell}}, \tag{1.19}$$

and compute $\widehat{\varphi}_2^{(i)}$ from

$$\widehat{\mathcal{L}}_2\,\widehat{\varphi}_2^{(i)} = \widehat{f}_2 \quad \text{on} \quad \widehat{\Omega}_2, \tag{1.20}$$

$$\widehat{\mathcal{G}}_{21}\;\widehat{\varphi}_2^{(i)}\Big|_{\widehat{\Gamma}_{2\ell}} = \widehat{\mathcal{G}}_{21}\;\widehat{\varphi}_1^{(i-1)}\Big|_{\widehat{\Gamma}_{1r}}. \tag{1.21}$$

The difference with the overlapping Schwarz method is that here information from one subgrid to the other is transferred via discretized boundary conditions (1.19) and (1.21) instead of a shared interchange region $\widehat{\Omega}_1 \cap \widehat{\Omega}_2$. The speed of convergence of a nonoverlapping Schwarz method depends on the boundary conditions on the internal boundary. For an operator on the internal boundary that describes the boundary conditions corresponding to the exact solution, the method needs only one iteration. But, this needs unknown information. In practice, popular choices for the operator in (1.19) and (1.21) are the identity (a Dirichlet condition) and the first order (discretized) derivative (a Neumann condition). Then, for instance "Neumann-Dirichlet coupling" means a Neumann condition described by $\widehat{\mathcal{G}}_{12}$ in (1.19) and a Dirichlet condition described by $\widehat{\mathcal{G}}_{21}$ in (1.21). A more sophisticated choice is some linear combination of a Dirichlet and Neumann condition: a mixed or Robin condition.

By elimination of the subsystems (1.18) and (1.20) a reduced system on the interface $\widehat{\Gamma}_{1r} = \widehat{\Gamma}_{2\ell}$ remains which is the Schur complement. Also overlapping Schwarz methods can be formulated as specific Schur complements (see [4, 11]; a more recent paper is [61]).

Overlapping Schwarz methods have the disadvantage that the overlap results in computational overhead.

Nonoverlapping Schwarz methods, rely heavily on appropriate conditions on the internal boundaries. It is possible to tune this coupling for better convergence, as will be explained now.

### 1.4.2 Tuning of the coupling

The idea of a nonoverlapping Schwarz method with more sophisticated internal boundary conditions was generalized by Tang [57] and by Tan and Borsboom [55, 56].

The enhancement of matrices and vectors, as introduced by Tang in [57], enables a clear and compact formulation of the domain decomposition method. In the two subdomain case of §1.4.1, the nonoverlapping subgrids $\widehat{\Omega}_1$ and $\widehat{\Omega}_2$ are expanded with additional gridpoints along the internal boundary $\widehat{\Gamma}_{1r} = \widehat{\Gamma}_{1\ell}$. Unknowns are defined on the additional gridpoints.

These additional unknowns constitute the enhancement of the vector with unknowns. In order to fit the extra unknowns in the discretized system, the matrix that represents the discretized operator is enhanced. For that purpose Tang made a splitting of the subblocks that couple the unknowns along the internal boundary and enhanced the right-hand side vector with a copy of the right-hand side values corresponding to those subblocks.

Tan and Borsboom [55, 56] refined this concept by defining a double set of additional gridpoints near the interface. Then no splitting of subblocks of the discretized operator, that describes the original system, needs to be made as coupling between unknowns and corresponding additional unknowns near the interface is defined by extra equations independently of the discretization.

By enhancing the unknowns near/along the internal boundary, extra degrees of freedom are created reflected by coupling parameters. These parameters can be interpreted as weights that describe (for many parameters even higher order) mixed boundary conditions at the internal boundary. The idea is to tune these parameters to speed up the convergence of the domain decomposition method. This tuning needs some knowledge of the (physical) system from which the linear system arises via discretization. By performing for a modelproblem an analysis, optimal coupling parameters can be determined. These values can be used to estimate appropriate coupling parameters for more complex problems. This approach has been very successful for elliptic problems also in case of advection dominated problems [55, 56].

## 1.5 Outline of this thesis

Before one tries to apply the domain decomposition technique [55, 56] to the correction equation of the Jacobi-Davidson method one should be aware of the following two aspects.

First it should be emphasized that accelerated inexact Newton methods, in particular Jacobi-Davidson in combination with approximate solutions of the correction equation, are nested iterative methods. As a consequence the accuracy obtained in the innerloop (a truncated iterative method for the correction equation) affects the outerloop (the iterations of Jacobi-Davidson).

Secondly, due to the shift $\theta$, the linear operator in the correction equation is usually indefinite. As the domain decomposition technique from Tang and Tan & Borsboom is developped for positive definite matrices, this technique cannot be applied right away to the correction equation and further investigation is needed.

The thesis is organized as follows.

First in chapter 2 alternative correction equations for the original Jacobi-Davidson without any preconditioning are studied.
This study is motivated by an analogy with nested methods GMRESR [59] and GCRO [18] for systems of linear equations as discussed in [25]. Furthermore, it may yield a remedy for the case of $\theta$ close to a multiple eigenvalue.

After this pilot study the stage is set for the incorporation of the domain decomposition technique in the Jacobi-Davidson method.

The heart of this thesis is contained in chapter 3. There the concepts of the domain decomposition technique from Tang [57] and Tan & Borsboom [55, 56] are reformulated and adapted for the correction equation.
An analysis is performed for a two dimensional advection-diffusion model eigenvalue problem with constant coefficients. With this knowledge in mind, optimal coupling parameters can be determined for different types of coupling. The predicted performances for these couplings are verified by numerical experiments. It turned out that with the coupling only the positive definite part of the operator in the correction equation can be controlled. For the remaining, negative definite part a strategy is developped and verified by a numerical experiment. The chapter concludes with a number of numerical experiments to indicate the overall performance of the Jacobi-Davidson method in combination with solutions of the correction equation obtained via the domain decomposition technique.

Where the approach in chapter 3 is of conceptual type, chapter 4 discusses practical aspects.
In many applications the eigenvalue problems exhibit coefficients that vary over the physical domain. It is shown experimentally how results from chapter 3 for the case of constant coefficients may be applied to the case of variable coefficients. Several characteristic numerical experiments accompany the discussion. Then attention is turned to more complicated geometries.

In the final chapter it is shown that once a preconditioner based on domain decomposition is constructed, one may take more advantage of it by considering the different levels of nesting in the Jacobi-Davidson method. For a high degree of parallelism, i.e. for a large number of subdomains, the observed phenomenon becomes significant.

Chapter 2 has appeared as:

> M. GENSEBERGER AND G. L. G. SLEIJPEN, *Alternative correction equations in the Jacobi-Davidson method*, Numer. Linear Algebra Appl., 6 (1999), pp. 235–253.

Chapter 3 is submitted for publication.

# Chapter 2

# Alternative correction equations in the Jacobi-Davidson method

Menno Genseberger and Gerard Sleijpen

**Abstract**

The correction equation in the Jacobi-Davidson method is effective in a subspace orthogonal to the current eigenvector approximation, whereas for the continuation of the process only vectors orthogonal to the search subspace are of importance. Such a vector is obtained by orthogonalizing the (approximate) solution of the correction equation against the search subspace. As an alternative, a variant of the correction equation can be formulated that is restricted to the subspace orthogonal to the current search subspace. In this chapter, we discuss the effectiveness of this variant.

Our investigation is also motivated by the fact that the restricted correction equation can be used for avoiding stagnation in case of defective eigenvalues. Moreover, this equation plays a key role in the inexact TRQ method [51].

**Keywords:** Eigenvalues and eigenvectors, Jacobi-Davidson method

**AMS subject classification:** 65F15, 65N25

## 2.1 Introduction

For the computation of a few eigenvalues with associated eigenvectors of large $n$-dimensional linear eigenvalue problems

$$\mathbf{A}\mathbf{x} = \lambda\mathbf{x} \tag{2.1}$$

subspace methods have become very popular. The application of a subspace method is attractive when the method is able to calculate accurate solutions to (2.1) from relatively low dimensional subspaces, i.e. $m \ll n$ with $m$ the dimension of the subspace. Keeping $m$ small enables a reduction in computational time and memory usage.

There are many ways to construct a subspace and different options are possible for a subspace method. Globally three stages can be distinguished in such a method:

- Calculation of an approximation to the eigenpair inside the search subspace.
- Computation of new information about the behavior of operator $\mathbf{A}$.
- Expansion of the search subspace with vector(s) containing this information.

In the Jacobi-Davidson method [46], Sleijpen and Van der Vorst propose to look for new information in the space orthogonal to the approximate eigenvector. A correction equation

$$(\mathbf{I}_n - \mathbf{u}_m\mathbf{u}_m^*)(\mathbf{A} - \theta_m\mathbf{I}_n)(\mathbf{I}_n - \mathbf{u}_m\mathbf{u}_m^*)\mathbf{t} = -\mathbf{r}_m, \tag{2.2}$$

is defined on this space. Here $(\theta_m, \mathbf{u}_m)$ is the current approximate eigenpair with residual $\mathbf{r}_m \equiv \mathbf{A}\mathbf{u}_m - \theta_m\mathbf{u}_m$. A correction $\mathbf{t}$ to the approximate eigenvector $\mathbf{u}_m$ is obtained by solving (2.2) approximately. Then the search subspace $\mathcal{V}_m$ is expanded to $\mathcal{V}_{m+1}$ with the component of $\mathbf{t}$ orthogonal to $\mathcal{V}_m$. One of the eigenvalues $\theta_{m+1}$ of the projection of matrix $\mathbf{A}$ on the new search subspace is selected. Inside $\mathcal{V}_{m+1}$ the so-called Ritz pair $(\theta_{m+1}, \mathbf{u}_{m+1})$ is considered to be an optimal approximation to the wanted eigenpair $(\lambda, \mathbf{x})$.

As the residual of a Ritz pair is orthogonal to the subspace this special choice of the approximation introduces some freedom for the projection of the correction equation. Another possibility is looking for a correction in the space orthogonal to the search subspace constructed so far. If the Ritz pair is indeed the "best" approximation inside the search subspace, then we should expect that really new information lies in the orthogonal complement of $\mathcal{V}_m$. This suggests a more restrictive correction equation

$$(\mathbf{I}_n - \mathbf{V}_m\mathbf{V}_m^*)(\mathbf{A} - \theta_m\mathbf{I}_n)(\mathbf{I}_n - \mathbf{V}_m\mathbf{V}_m^*)\mathbf{t} = -\mathbf{r}_m, \tag{2.3}$$

that will be investigated here. In equation (2.3), $\mathbf{V}_m$ is an $n$ by $m$ matrix of which the columns form an orthonormal basis of the current search subspace $\mathcal{V}_m$.

Although the approach in (2.3) does not seem to be unnatural, it is not clear whether it is more effective in practical computations. In general, the solutions of (2.2) and (2.3) do not lead to the same expansion of the search subspaces. Therefore, a different convergence behavior of the Jacobi-Davidson process is to be expected.

The projection in (2.3) is more expensive, but the method for solving the correction equation may profit from projecting on a smaller subspace. To see this, note that $\mathbf{A} - \theta_m\mathbf{I}_n$ is

nearly singular if $\theta_m \approx \lambda$. Restricting $\mathbf{A} - \theta_m \mathbf{I}_n$ to the space orthogonal to the approximate eigenvector $\mathbf{u}_m$ will give a well-conditioned operator in case $\lambda$ is simple and fairly well isolated from the other eigenvalues. Projecting on the space orthogonal to $\mathcal{V}_m$ may further improve the conditioning. If eigenvalues cluster around the target eigenvalue $\lambda$ then the associated eigenvectors should be removed as well. The search subspace may be expected to contain good approximations also of these eigenvectors [26, §3.4] and projecting on the space orthogonal to $\mathcal{V}_m$ may lead to a well-conditioned operator also in case of clustering eigenvectors. A reduction may be expected in the number of steps that are needed to solve the correction equation to a certain accuracy if an iterative linear solver is used. It also improves the stability of the linear solver. These effects may compensate for the more expensive steps. For precisely these reasons, a strategy is followed in [22, 17] where $\mathbf{u}_m$ in (2.2) is replaced by the matrix of all Ritz vectors that could be associated with eigenvalues in a cluster near the target eigenvalue.

GMRESR[1] [59] and GCRO[2] [18] are nested methods for solving linear systems $\mathbf{A}\mathbf{x} = \mathbf{b}$ iteratively. They both use GCR in the "outer loop" to update the approximate solution and GMRES in the "inner loop" to compute a new search direction from a correction equation. As argued in [25], Jacobi-Davidson with (2.2) can be viewed as the eigenvalue version of GMRESR, while Jacobi-Davidson with (2.3) is the analogue of GCRO. GCRO employs the search subspace to improve the convergence of GMRES for the solution of a correction equation (see also [19]). Experiments in [18, 5] for linear systems of equations show that GCRO can be more effective than GMRESR: for linear problems it appears to be worthwhile to use more expensive projections. Is this also the case for eigenvalue problems? If, for a linear system, the correction equation is solved exactly then both GMRESR and GCRO produce the exact solution of the linear system in the next step. However, eigenvalue problems are not linear and even if all correction equations are solved exactly still a number of steps may be needed to find accurate approximations of an eigenpair. Replacing $\mathbf{u}_m$ in (2.2) by $\mathbf{V}_m$ may lead to an increase in the number of iteration steps. The loss in speed of convergence may not be compensated by the advantage of a better conditioned correction equation (2.3). In practical computations the situation is even more complicated since the correction equations will be solved only with a modest accuracy.

Jacobi-Davidson itself may also profit from projecting on a smaller subspace. If the Ritz value is a defective eigenvalue of the interaction matrix $\mathbf{V}_m^* \mathbf{A} \mathbf{V}_m$ then the correction equation (2.2) may have a solution in the current search subspace. In such a case the search subspace is not expanded and Jacobi-Davidson stagnates. Correction equation (2.3) will give a proper expansion vector and stagnation can be avoided [48]. In practical computations, where the correction equations are not solved exactly, it is observed that stagnation also can be avoided by a strategical and occasional use of (2.3).

Equation (2.3) also plays a key role in the inexact Truncated RQ iteration [51] of Sorensen and Yang (see also §§2.2.3 and 2.4.1). This provides another motivation for studying the effect of using (2.3) in Jacobi-Davidson.

This chapter is organized as follows. First, in §2.2 we recall some facts about projecting

[1] Generalized Minimum Residual Recursive
[2] Generalized Conjugate Residual with Orthogonalization in the inner iteration

the eigenvalue problem. An alternative derivation of a more general correction equation is given to motivate the correction equation (2.3). It appears that (2.3) and the original correction equation (2.2) are the extremal cases of this general correction equation. Next, in §2.3, an illustration is given in which the two correction equations can produce different results. We will show that, if the process is started with a Krylov subspace then the two exact solutions of the correction equations lead to mathematically equivalent results (§2.4). We will also argue that in other situations the correction equation (2.3) will lead to slower convergence. In §2.5 we conclude with some numerical experiments; partially as an illustration of the preceding, partially to observe what happens if things are not computed in high precision and whether round-off errors play a role of importance.

## 2.2 The framework: the Jacobi-Davidson method

We start with a brief summary of the Rayleigh-Ritz procedure. This procedure, where the large eigenvalue problem is projected on a small one, serves as a starting point for the derivation of a more general correction equation. We will consider the two extremal cases of this equation. One corresponds to the correction equation of the original Jacobi-Davidson method, the other one is employed in the inexact Truncated RQ iteration.

### 2.2.1 Interpolation: Rayleigh-Ritz procedure

Suppose some $m$-dimensional subspace $\mathcal{V}_m$ is available. Let $\mathbf{V}_m$ be an $n \times m$ dimensional matrix such that the column-vectors of $\mathbf{V}_m$ form an orthonormal basis of $\mathcal{V}_m$. The orthogonal projection of $\mathbf{A}$ on the subspace (the Rayleigh quotient or interaction matrix) will then be $H_m \equiv \mathbf{V}_m^* \mathbf{A} \mathbf{V}_m$.

Furthermore suppose that we selected a Ritz pair $(\theta_m, \mathbf{u_m})$ of $\mathbf{A}$ with respect to $\mathcal{V}_m$, i.e. a scalar $\theta_m$ and a vector $\mathbf{u}_m \in \mathcal{V}_m$ such that the residual $\mathbf{r}(\theta_m, \mathbf{u}_m) \equiv \mathbf{r}_m \equiv \mathbf{A}\mathbf{u}_m - \theta_m \mathbf{u}_m$ is orthogonal to $\mathcal{V}_m$. A Ritz pair can considered to be an optimal approximation inside the subspace to an eigenpair $(\lambda, \mathbf{x})$ of the matrix $\mathbf{A}$ in some sense (in [37, §11.4] this is argued for the real symmetric case).

The Ritz values are equal to the eigenvalues of $H_m$. Therefore they can be computed by solving the $m$-dimensional linear eigenvalue problem $H_m s = \theta s$. The Ritz vector associated with $\theta$ is $\mathbf{V}_m s$.

### 2.2.2 Extrapolation: correction equation

How well does the Ritz pair $(\theta_m, \mathbf{u}_m)$ approximate an eigenpair $(\lambda, \mathbf{x})$ of matrix $\mathbf{A}$? With a view restricted to the subspace there would be no better alternative. But outside $\mathcal{V}_m$ a remainder $\mathbf{r}_m$ is left. The norm of this residual gives an indication about the quality of the approximation. Let us try to minimize this norm.

For that purpose, consider $\mathbf{u}' = \mathbf{u}_m + \mathbf{t}$ and $\theta' = \theta_m + \varepsilon$. Define the residual $\mathbf{r}' \equiv \mathbf{A}\mathbf{u}' - \theta'\mathbf{u}' = \mathbf{r}_m + \mathbf{A}\mathbf{t} - \theta_m \mathbf{t} - \varepsilon \mathbf{u}_m - \varepsilon \mathbf{t}$. If we view $\varepsilon$ and $\mathbf{t}$ as first order corrections

then $\varepsilon\mathbf{t}$ represents some second order correction (cf. [36], [52]). Ignoring this contribution results in

$$\mathbf{r}' = \mathbf{r}_m + (\mathbf{A} - \theta_m \mathbf{I}_n)\mathbf{t} - \varepsilon \mathbf{u}_m. \tag{2.4}$$

Consider some subspace $\mathcal{W}$ such that $\mathbf{u}_m \in \mathcal{W} \subseteq \mathcal{V}_m$. With $\mathbf{W}$, a matrix of which the column-vectors form an orthonormal basis for $\mathcal{W}$, we decompose (2.4) (cf. [44]) in

$$\mathbf{W}\mathbf{W}^*\mathbf{r}' = \mathbf{W}\mathbf{W}^*(\mathbf{A} - \theta_m \mathbf{I}_n)\mathbf{t} - \varepsilon \mathbf{u}_m,$$

the component of $\mathbf{r}'$ in $\mathcal{W}$, and in

$$(\mathbf{I}_n - \mathbf{W}\mathbf{W}^*)\mathbf{r}' = (\mathbf{I}_n - \mathbf{W}\mathbf{W}^*)(\mathbf{A} - \theta_m \mathbf{I}_n)\mathbf{t} + \mathbf{r}_m, \tag{2.5}$$

the component of $\mathbf{r}'$ orthogonal to $\mathcal{W}$.

The new direction $\mathbf{t}$ will be used to expand the subspace $\mathcal{V}_m$ to $\mathcal{V}_{m+1}$. An approximation $(\theta_{m+1}, \mathbf{u}_{m+1})$ is computed with respect to $\mathcal{V}_{m+1}$. Because $\mathcal{W} \subseteq \mathcal{V}_m \subseteq \mathcal{V}_{m+1}$ the residual $\mathbf{r}_{m+1}$ of this Ritz pair is also orthogonal to $\mathcal{W}$. This means that if we write $(\theta_{m+1}, \mathbf{u}_{m+1}) = (\theta_m + \varepsilon, \mathbf{u}_m + \mathbf{t})$ then only (2.5) gives a contribution to the norm of $\mathbf{r}_{m+1}$:

$$\|\mathbf{r}_{m+1}\| = \|(\mathbf{I}_n - \mathbf{W}\mathbf{W}^*)(\mathbf{A} - \theta_m \mathbf{I}_n)\mathbf{t} + \mathbf{r}_m\|. \tag{2.6}$$

So to get a smaller norm in the next step we should calculate $\mathbf{t}$ such that

$$(\mathbf{I}_n - \mathbf{W}\mathbf{W}^*)(\mathbf{A} - \theta_m \mathbf{I}_n)\mathbf{t} = -\mathbf{r}_m. \tag{2.7}$$

Note that if $\mathbf{t} = \mathbf{u}_m$ then there is no expansion of the search space. So it can be assumed that $\mathbf{t} \neq \mathbf{u}_m$. As we are free to scale $\mathbf{u}_m$ to any length, we can require that $\mathbf{t} \perp \mathbf{u}_m$. From this it follows that if $\mathbf{t} \neq \mathbf{u}_m$ then equation (2.7) and

$$(\mathbf{I}_n - \mathbf{W}\mathbf{W}^*)(\mathbf{A} - \theta_m \mathbf{I}_n)(\mathbf{I}_n - \mathbf{u}_m \mathbf{u}_m^*)\mathbf{t} = -\mathbf{r}_m \tag{2.8}$$

can considered to be equivalent.

Drawback may be that the linear systems in (2.7) and (2.8) are underdetermined. The operators $(\mathbf{I}_n - \mathbf{W}\mathbf{W}^*)(\mathbf{A} - \theta_m \mathbf{I}_n)$ and $(\mathbf{I}_n - \mathbf{W}\mathbf{W}^*)(\mathbf{A} - \theta_m \mathbf{I}_n)(\mathbf{I}_n - \mathbf{u}_m \mathbf{u}_m^*)$ map $\mathbf{t}$ on a lower dimensional subspace $\mathcal{W}$. The operator $(\mathbf{I}_n - \mathbf{W}\mathbf{W}^*)(\mathbf{A} - \theta_m \mathbf{I}_n)(\mathbf{I}_n - \mathbf{W}\mathbf{W}^*)$ acts only inside the space orthogonal to $\mathcal{W}$. We expect this operator to have a more favourable distribution of eigenvalues for the iterative method. In that case the correction equation reads

$$(\mathbf{I}_n - \mathbf{W}\mathbf{W}^*)(\mathbf{A} - \theta_m \mathbf{I}_n)(\mathbf{I}_n - \mathbf{W}\mathbf{W}^*)\mathbf{t} = -\mathbf{r}_m. \tag{2.9}$$

If the correction equation is solved (approximately) by a Krylov subspace method where the initial guess is $\mathbf{0}$, then no difference will be observed between (2.7) and (2.9). The reason why is that $(\mathbf{I}_n - \mathbf{W}\mathbf{W}^*)^2 = \mathbf{I}_n - \mathbf{W}\mathbf{W}^*$.

### 2.2.3 Extremal cases

After $m$ steps of the subspace method, $\mathcal{V}_m$ contains besides $\mathbf{u}_m$, $m-1$ other independent directions. Consequence: different subspaces $\mathcal{W}$ can be used in equation (2.7) provided that span$(\mathbf{u}_m) \subseteq \mathcal{W} \subseteq \mathcal{V}_m$. Here we will consider the extremal cases $\mathcal{W} = $ span$(\mathbf{u}_m)$ and $\mathcal{W} = \mathcal{V}_m$.

The first case corresponds with the original Jacobi-Davidson method [46]:

$$(\mathbf{I}_n - \mathbf{u}_m\mathbf{u}_m^*)(\mathbf{A} - \theta_m\mathbf{I}_n)(\mathbf{I}_n - \mathbf{u}_m\mathbf{u}_m^*)\mathbf{t} = -\mathbf{r}_m.$$

The operator in this equation can be seen as a mapping in the orthogonal complement of $\mathbf{u}_m$.

Let us motivate the other case. Suppose $\mathcal{W}$ is a subspace contained in, but not equal to $\mathcal{V}_m$. Then $(\mathbf{I}_n - \mathbf{W}\mathbf{W}^*)$ projects still some components of $(\mathbf{A} - \theta_m\mathbf{I}_n)\mathbf{t}$ inside $\mathcal{V}_m$. These components will not contribute to a smaller norm in (2.6). To avoid this overhead of already known information it is tempting to take $\mathcal{W} = \mathcal{V}_m$:

$$(\mathbf{I}_n - \mathbf{V}_m\mathbf{V}_m^*)(\mathbf{A} - \theta_m\mathbf{I}_n)(\mathbf{I}_n - \mathbf{u}_m\mathbf{u}_m^*)\mathbf{t} = -\mathbf{r}_m. \tag{2.10}$$

Furthermore, if $\mathcal{W} = \mathcal{V}_m$ then equation (2.9) becomes

$$(\mathbf{I}_n - \mathbf{V}_m\mathbf{V}_m^*)(\mathbf{A} - \theta_m\mathbf{I}_n)(\mathbf{I}_n - \mathbf{V}_m\mathbf{V}_m^*)\mathbf{t} = -\mathbf{r}_m.$$

In the following with JD and JDV we will denote the Jacobi-Davidson method which uses (2.2) and (2.3) respectively as correction equation. The *exact* solution of (2.2) will be denoted by $\mathbf{t}_{\text{JD}}$, while $\mathbf{t}_{\text{JDV}}$ denotes the *exact* solution of (2.3). With an "*exact*" process we refer to a process in exact arithmetic in which all correction equations are solved exactly. Note that both $\mathbf{t}_{\text{JD}}$ and $\mathbf{t}_{\text{JDV}}$ are solutions of (2.10). As we will illustrate in an example in §2.3, the solution set of (2.10) may consist of more than two vectors. In fact, this set will be an affine space of dimension dim$(\mathcal{V}_m)$, while generally (2.2) and (2.3) will have unique solutions. For this reason, we will refer to equation (2.10) as the "in between" equation.

An equation similar to (2.3) appears in the truncated RQ-iteration of Sorensen and Yang [51]. In every step of this method the solution of the so-called TRQ equations is required. For the application of an iterative solver the authors recommend to use

$$(\mathbf{I}_n - \mathbf{V}_m\mathbf{V}_m^*)\,(\mathbf{A} - \mu\mathbf{I}_n)\,(\mathbf{I} - \mathbf{V}_m\mathbf{V}_m^*)\,\widehat{\mathbf{w}} = \mathbf{f}_m \tag{2.11}$$

instead of the TRQ equations. Here $\mu$ is some shift which may be chosen to be fixed for some TRQ-iteration steps whereas in Jacobi-Davidson $\theta_m$ is an optimal shift which differs from step to step. Also here Sorensen and Yang expect (2.11) to give better results due to the fact that
$(\mathbf{I}_n - \mathbf{V}_m\mathbf{V}_m^*)\,(\mathbf{A} - \mu\mathbf{I}_n)\,(\mathbf{I} - \mathbf{V}_m\mathbf{V}_m^*)$ has a more favourable eigenvalue distribution than $\mathbf{A} - \mu\mathbf{I}$ when $\mu$ is near an eigenvalue of $\mathbf{A}$ (see also the remark at the end of §2.4.1).

### 2.2.4 Convergence rate

The derivation in §2.2.2 of the alternative correction equations may suggest that expansion with an exact solution $\mathbf{t}$ of (2.10) would result in quadratic convergence (cf. [50]) like the original Jacobi-Davidson method ([46, §4.1], [44, Th.3.2]). Let us take a closer look.

As in §2.2.2, consider the residual $\mathbf{r}_{m+1}$ associated with $(\theta_{m+1}, \mathbf{u}_{m+1}) = (\theta_m + \varepsilon, \mathbf{u}_m + \mathbf{t})$.
If $\mathbf{t} \perp \mathbf{u}_m$ is the exact solution of (2.2) and $\varepsilon$ is chosen such that $\mathbf{r}_{m+1}$ is orthogonal to $\mathbf{u}_m$ then it can be checked that $\mathbf{r}_{m+1}$ is equal to a quadratic term ($\mathbf{r}_{m+1} = -\varepsilon\mathbf{t}$), which virtually proves quadratic convergence. (Note: we are dealing not only with the directions $\mathbf{u}_m$ and $\mathbf{t}$ but with a search subspace from which the new approximation is computed, there could be an update for $\mathbf{u}_m$ that is even better than $\mathbf{t}$.)
If $\mathbf{t}$ solves (2.10) exactly then, by construction, the component of the residual orthogonal to $\mathcal{V}_m$ consists of a second order term. However, generally the component of $\mathbf{r}_{m+1}$ in the space $\mathcal{V}_m$ contains first order terms (see §2.3) and updating $\mathbf{u}_m$ with this exact solution $\mathbf{t}$ of (2.10) does not lead to quadratic convergence. One may hope for better updates in the space spanned by $\mathcal{V}_m$ and $\mathbf{t}$, but, as we will see in our numerical experiments in §2.5.1.1, equation (2.3), and therefore also (2.10), do not lead to quadratic convergence in general.

## 2.3 Two examples

The two following simple examples give some insight into the differences between the three correction equations (2.2), (2.10), and (2.3).

### 2.3.1 Different expansion of the subspace

Consider the following matrix

$$\mathbf{A} = \begin{pmatrix} 0 & \beta & \mathbf{c}_1^* \\ 0 & \alpha & \mathbf{c}_2^* \\ \mathbf{d}_1 & \mathbf{d}_2 & \mathbf{B} \end{pmatrix},$$

with $\alpha$ and $\beta$ scalars, $\mathbf{c}_1, \mathbf{c}_2, \mathbf{d}_1$ and $\mathbf{d}_2$ vectors and $\mathbf{B}$ a non-singular matrix of appropriate size.

Suppose we already constructed the subspace $\mathcal{V}_2 = \text{span}(\mathbf{e}_1, \mathbf{e}_2)$ and the selected Ritz vector $\mathbf{u}_2$ is $\mathbf{e}_1$. Then the associated Ritz value $\theta_2$ equals $0$,

$$\mathbf{r}_2 = \begin{pmatrix} 0 \\ 0 \\ \mathbf{d}_1 \end{pmatrix},$$

while $(\mathbf{I} - \mathbf{e}_1\mathbf{e}_1^*)\mathbf{A}(\mathbf{I} - \mathbf{e}_1\mathbf{e}_1^*)$, $(\mathbf{I} - \mathbf{V}_2\mathbf{V}_2^*)\mathbf{A}(\mathbf{I} - \mathbf{e}_1\mathbf{e}_1^*)$, and $(\mathbf{I} - \mathbf{V}_2\mathbf{V}_2^*)\mathbf{A}(\mathbf{I} - \mathbf{V}_2\mathbf{V}_2^*)$ are equal to

$$\begin{pmatrix} 0 & 0 & \mathbf{0}^* \\ 0 & \alpha & \mathbf{c}_2^* \\ \mathbf{0} & \mathbf{d}_2 & \mathbf{B} \end{pmatrix}, \quad \begin{pmatrix} 0 & 0 & \mathbf{0}^* \\ 0 & 0 & \mathbf{0}^* \\ \mathbf{0} & \mathbf{d}_2 & \mathbf{B} \end{pmatrix}, \quad \text{and} \quad \begin{pmatrix} 0 & 0 & \mathbf{0}^* \\ 0 & 0 & \mathbf{0}^* \\ \mathbf{0} & \mathbf{0} & \mathbf{B} \end{pmatrix},$$

respectively. From this it is seen that JD computes its correction from

$$\begin{pmatrix} \alpha & \mathbf{c}_2^* \\ \mathbf{d}_2 & \mathbf{B} \end{pmatrix} \begin{pmatrix} \gamma \\ \mathbf{t}' \end{pmatrix} = - \begin{pmatrix} 0 \\ \mathbf{d}_1 \end{pmatrix},$$

the "in between" from

$$\begin{pmatrix} \mathbf{d}_2 & \mathbf{B} \end{pmatrix} \begin{pmatrix} \gamma \\ \mathbf{t}' \end{pmatrix} = -\mathbf{d}_1,$$

and JDV from

$$\mathbf{B}\mathbf{t}' = -\mathbf{d}_1.$$

Let $\mathbf{t}_i'$ be the solution of $\mathbf{B}\mathbf{t}_i' = -\mathbf{d}_i$ ($i = 1, 2$). Then the component of $\mathbf{t}_{\text{JDV}}$ for JDV orthogonal to $\mathcal{V}_2$ is represented by $\mathbf{t}_1'$ (to be more precise, $\mathbf{t}_{\text{JDV}} = (0, 0, \mathbf{t}_1'^{\text{T}})^{\text{T}}$), while the orthogonal component for JD is represented by a combination of $\mathbf{t}_1'$ and $\mathbf{t}_2'$: $\mathbf{t}_{\text{JD}} = (0, \gamma, (\mathbf{t}_1' + \gamma\mathbf{t}_2')^{\text{T}})^{\text{T}}$. So in general, when $\mathbf{d}_2$ is not a multiple of $\mathbf{d}_1$ and when $\gamma \neq 0$, JD and JDV will not produce the same expansion of $\mathcal{V}_2$. Note that $(\mathbf{I} - \mathbf{e}_1\mathbf{e}_1^*)\mathbf{A}(\mathbf{I} - \mathbf{e}_1\mathbf{e}_1^*)$ is non-singular on $\mathbf{e}_1^\perp$ if and only if $\alpha \neq -\mathbf{c}_2^*\mathbf{t}_2'$. The "in between" differs from JD and JDV in that it has no extra constraint for $\gamma$. Taking $\gamma = -\mathbf{c}_2^*\mathbf{t}_1'/(\alpha + \mathbf{c}_2^*\mathbf{t}_2')$ gives JD, taking $\gamma = 0$ gives JDV.

Finally, as an illustration of §2.2.4, we calculate the new residual associated with $\mathbf{u}_3 = \mathbf{u}_2 + \mathbf{t}$ and $\theta_3 = \theta_2 + \varepsilon$. We take $\beta = 0$. The new residual for the "in between" equals

$$\mathbf{r}_3 = \begin{pmatrix} \mathbf{c}_1^*\mathbf{t}' - \varepsilon \\ \alpha\gamma + \mathbf{c}_2^*\mathbf{t}' - \varepsilon\gamma \\ -\varepsilon\mathbf{t}' \end{pmatrix}.$$

If $\gamma = -\mathbf{c}_2^*\mathbf{t}_1'/(\alpha + \mathbf{c}_2^*\mathbf{t}_2')$ (as for JD) then the choice $\varepsilon = \mathbf{c}_1^*\mathbf{t}'$ reduces the terms in $\mathbf{r}_3$ to second order ones, while no clever choice for $\varepsilon$ can achieve this if $\gamma$ is not close to $-\mathbf{c}_2^*\mathbf{t}_1'/(\alpha + \mathbf{c}_2^*\mathbf{t}_2')$.

### 2.3.2 Stagnation

The example in this section shows that JD may stagnate where JDV expands.
Consider the matrix $\mathbf{A}$ of §2.3.1, but now take $\beta = 1$, $\alpha = 0$ and $\mathbf{d}_2 = \mathbf{d}_1$.

As initial space, we take $\mathcal{V}_1 = \text{span}\{e_1\}$. Then $\mathbf{u}_1 = e_1$ and $\mathbf{r}_1 = (0, 0, \mathbf{d}_1^{\text{T}})^{\text{T}}$. Any of the three approaches find $-e_2$ as expansion vector: $\mathcal{V}_2 = \text{span}\{e_1, e_2\}$. Now $\mathbf{u}_2$ is again $e_1$ and JD stagnates: $\mathbf{t}_{\text{JD}} = -e_2$ belongs already to $\mathcal{V}_2$ and does not lead to an expansion of $\mathcal{V}_2$. The JDV correction vector $\mathbf{t}_{\text{JDV}}$ is equal to $(0, 0, (\mathbf{B}^{-1}\mathbf{d}_1)^{\text{T}})^{\text{T}}$ and expands $\mathcal{V}_2$.

## 2.4 Exact solution of the correction equations

If, in the example in §2.3.1, $\mathbf{d}_1$ and $\mathbf{d}_2$ are in the same direction, or equivalently, if the residuals of the Ritz vectors are in the same direction, then exact JD and exact JDV calculate effectively the same expansion vector. One may wonder whether this also may happen in

more general situations. Before we discuss this question, we characterize the situation in which all residuals are in the same direction.

All residuals of Ritz vectors with respect to some subspace $\mathcal{V}_m$ are in the same direction if and only if the components orthogonal to $\mathcal{V}_m$ of the vectors $\mathbf{A}\mathbf{v}$ are in the same direction for all $\mathbf{v} \in \mathcal{V}_m$. It is easy to see and well known that $\mathcal{V}_m$ has this last property if it is a Krylov subspace generated by $\mathbf{A}$ (i.e., $\mathcal{V}_m = \mathcal{K}_m(\mathbf{A}, \mathbf{v}_0) = \text{span}(\{\mathbf{A}^i\mathbf{v}_0 \mid i < m\})$ for some positive integer $m$ and some vector $\mathbf{v}_0$). The converse is also true as stated in the following lemma. We will tacitly assume that all Krylov subspaces that we will consider in the remainder of this chapter, are generated by $\mathbf{A}$.

LEMMA 1 *For a subspace $\mathcal{V}_m$ the following properties are equivalent.*
(a) *$\mathcal{V}_m$ is a Krylov subspace,*
(b) *$\mathbf{A}\mathcal{V}_m \subset \text{span}(\mathcal{V}_m, \mathbf{v})$ for some $\mathbf{v} \in \mathbf{A}\mathcal{V}_m$.*

*Proof.* We prove that (b) implies (a). The implication "(a) $\Rightarrow$ (b)" is obvious.
If the columns of the $n$ by $m$ matrix $\mathbf{V}_m$ form a basis of $\mathcal{V}_m$ then (b) implies that $\mathbf{A}\mathbf{V}_m = [\mathbf{V}_m, \mathbf{v}]H$ for some $m+1$ by $m$ matrix $H$. There is an orthogonal $m$ by $m$ matrix $Q$ such that $\widetilde{H} := Q'^* H Q$ is upper Hessenberg. Here $Q'$ is the $m+1$ by $m+1$ orthogonal matrix with $m$ by $m$ left upper block $Q$ and $(m+1, m+1)$ entry equal to 1. $Q$ can be constructed as product of Householder reflections.[3] Hence $\mathbf{A}\widetilde{\mathbf{V}}_m = [\widetilde{\mathbf{V}}_m, \mathbf{v}]\widetilde{H}$, where $\widetilde{\mathbf{V}}_m \equiv \mathbf{V}_m Q$. Since $\widetilde{H}$ upper Hessenberg, this implies that $\mathcal{V}_m$ is a Krylov subspace (of order $m$) generated by the first column of $\widetilde{\mathbf{V}}_m$. □

We will see in Cor. 4 that exact JD and exact JDV coincide after restart with a set of Ritz vectors taken from a Krylov subspace. The proof uses the fact, formulated in Cor. 1, that any collection of Ritz vectors of $\mathbf{A}$ with respect to a single Krylov subspace span a Krylov subspace themselves. This fact can be found in [34, §3] and is equivalent to the statement in [53, Th.3.4] that Implicit Restarted Arnoldi and unpreconditioned Davidson (i.e., Davidson with the trivial preconditioner $\mathbf{I}_n$) generate the same search subspaces. However, the proof below is more elementary.

COROLLARY 1 *If $\mathcal{V}_m$ is a Krylov subspace and if $\{(\theta_m^{(i)}, \mathbf{u}_m^{(i)}) \mid i \in J\}$ is a subset of Ritz pairs of $\mathbf{A}$ with respect to $\mathcal{V}_m$ then the Ritz vectors $\mathbf{u}_m^{(i)}$ ($i \in J$) span a Krylov subspace.*

*Proof.* Assume that $\mathcal{V}_m$ is a Krylov subspace. Then (b) of Lemma 1 holds and, in view of the Gram-Schmidt process, we may assume that the vector $\mathbf{v}$ in (b) is orthogonal to $\mathcal{V}_m$ .
Since $\mathbf{A}\mathbf{u}_m^{(i)} - \theta_m^{(i)}\mathbf{u}_m^{(i)} \perp \mathcal{V}_m$, (b) of Lemma 1 implies that $\mathbf{A}\mathbf{u}_m^{(i)} - \theta_m^{(i)}\mathbf{u}_m^{(i)} \in \text{span}(\mathbf{v})$. Hence $\mathbf{A}\mathbf{u}_m^{(i)} \in \text{span}(\mathcal{U}, \mathbf{v})$, where $\mathcal{U}$ is the space spanned by the Ritz vectors $\mathbf{u}_m^{(i)}$ ($i \in J$), and the corollary follows from Lemma 1. □

[3] Here the refections are defined from their right action on the $m+1$ by $m$ matrix and work on the rows from bottom to top, whereas in the standard reduction to Hessenberg form of a square matrix they are defined from their left action and work on the columns from left to right.

### 2.4.1 Expanding a Krylov subspace

In this section, $\mathcal{V}_m$ is a subspace, $\mathbf{V}_m$ a matrix of which the columns form an orthonormal basis of $\mathcal{V}_m$, $(\theta_m, \mathbf{u}_m)$ a Ritz pair of $\mathbf{A}$ with respect to $\mathcal{V}_m$, and $\mathbf{r}_m$ is the associated residual. Further, we assume that $(\mathbf{I}_n - \mathbf{V}_m\mathbf{V}_m^*)(\mathbf{A} - \theta_m\mathbf{I}_n)(\mathbf{I}_n - \mathbf{V}_m\mathbf{V}_m^*)$ is non-singular on $\mathcal{V}_m^\perp$, that is (2.3) has a unique solution, and we assume that $\mathbf{r}_m \neq \mathbf{0}$, that is $\mathbf{u}_m$ is not converged yet.
The assumption $\mathbf{r}_m \neq \mathbf{0}$ implies that $\mathbf{t}_{\text{JDV}} \neq \mathbf{0}$ and $\mathbf{A}\mathbf{u}_m \notin \mathcal{V}_m$.

Note that (cf. [46], [39])

$$\mathbf{t}_{\text{JD}} = -\mathbf{u}_m + \varepsilon(\mathbf{A} - \theta_m\mathbf{I}_n)^{-1}\mathbf{u}_m \quad \text{for} \quad \varepsilon = \frac{\mathbf{u}_m^*\mathbf{u}_m}{\mathbf{u}_m^*(\mathbf{A} - \theta_m\mathbf{I}_n)^{-1}\mathbf{u}_m}. \tag{2.12}$$

THEOREM 1 *Consider the following properties.*
(a) $\mathcal{V}_m$ *is a Krylov subspace.*
(b) $\text{span}(\mathcal{V}_m, \mathbf{t}) \subset \text{span}(\mathcal{V}_m, \mathbf{t}_{\text{JDV}})$ *for all solutions* $\mathbf{t}$ *of* (2.10).
(c) $\text{span}(\mathcal{V}_m, \mathbf{t}_{\text{JD}})$ *is a Krylov subspace.*
*Then* (a) $\Leftrightarrow$ (b) $\Rightarrow$ (c).

*Proof.* Consider a solution $\mathbf{t}$ of (2.10). We first show the intermediate result that

$$\text{span}(\mathcal{V}_m, \mathbf{t}) = \text{span}(\mathcal{V}_m, \mathbf{t}_{\text{JDV}}) \quad \Leftrightarrow \quad \gamma\mathbf{A}\mathbf{u}_m + \mathbf{A}\mathbf{V}_m(\mathbf{V}_m^*\mathbf{t}) \in \mathcal{V}_m \quad \text{for some } \gamma \neq 1. \tag{2.13}$$

If we decompose $\mathbf{t}$ in

$$\mathbf{t} = \widetilde{\mathbf{t}} + \mathbf{V}_m s \quad \text{with} \quad \widetilde{\mathbf{t}} \equiv (\mathbf{I}_n - \mathbf{V}_m\mathbf{V}_m^*)\mathbf{t} \quad \text{and} \quad s \equiv \mathbf{V}_m^*\mathbf{t} \tag{2.14}$$

then we see that (2.10) is equivalent to

$$(\mathbf{I}_n - \mathbf{V}_m\mathbf{V}_m^*)(\mathbf{A} - \theta\mathbf{I}_n)(\mathbf{I}_n - \mathbf{V}_m\mathbf{V}_m^*)\widetilde{\mathbf{t}} = -\mathbf{r}_m - (\mathbf{I}_n - \mathbf{V}_m\mathbf{V}_m^*)(\mathbf{A} - \theta\mathbf{I}_n)\mathbf{V}_m s. \tag{2.15}$$

The vectors $\widetilde{\mathbf{t}}$ and $\mathbf{t}$ lead to the same expansion of $\mathcal{V}_m$. A combination of (2.3) and (2.15) shows that $\mathbf{t}_{\text{JDV}}$ and $\mathbf{t}$ lead to the same expansion of $\mathcal{V}_m$ if and only if

$$(1 - \gamma')\mathbf{r}_m + (\mathbf{I}_n - \mathbf{V}_m\mathbf{V}_m^*)(\mathbf{A} - \theta\mathbf{I}_n)\mathbf{V}_m s = \mathbf{0} \quad \text{for some scalar } \gamma' \neq 0; \tag{2.16}$$

use the non-singularity restriction for the "if-part". Since $(\mathbf{I}_n - \mathbf{V}_m\mathbf{V}_m^*)\mathbf{V}_m = \mathbf{0}$, (2.16) is equivalent to $(1 - \gamma')\mathbf{A}\mathbf{u}_m + \mathbf{A}\mathbf{V}_m s \in \mathcal{V}_m$, which proves (2.13).

"(a) $\Rightarrow$ (b)": Since $\mathbf{r}_m \neq \mathbf{0}$, we see that $\mathbf{A}\mathbf{u}_m \notin \mathcal{V}_m$. Therefore, if (a) holds then (see Lemma 1) we have that $\mathbf{A}\mathbf{V}_m(\mathbf{V}_m^*\mathbf{t}) \in \text{span}(\mathcal{V}_m, \mathbf{A}\mathbf{u}_m)$ and (2.13) shows that (b) holds.

"(b) $\Rightarrow$ (c)": Note that the kernel $\mathcal{N}$ of the operator in (2.10) consists of the vectors $\mathbf{s} \equiv \mathbf{t} - \mathbf{t}_{\text{JDV}}$ with $\mathbf{t}$ a solution of (2.10). Since (2.3) has a unique solution, we see that none of the non-trivial vectors in $\mathcal{N}$ is orthogonal to $\mathcal{V}_m$. Therefore, the space $\mathcal{N}$ and the space of all vectors $\mathbf{V}_m^*\mathbf{s}$ ($\mathbf{s} \in \mathcal{N}$) have the same dimension which is one less than the dimension of $\mathcal{V}_m$. From (2.13) we see that (b) implies that $\mathbf{A}\mathbf{V}_m(\mathbf{V}_m^*\mathbf{s}) \in \text{span}(\mathcal{V}_m, \mathbf{A}\mathbf{u}_m)$ for all $\mathbf{s} \in \mathcal{N}$. Since $\mathbf{s} = \mathbf{t} - \mathbf{t}_{\text{JDV}} \perp \mathbf{u}_m$, we see that $\mathbf{u}_m$ is independent of $\mathbf{A}\mathbf{V}_m(\mathbf{V}_m^*\mathbf{s})$ for all

$\mathbf{s} \in \mathcal{N}$. Therefore, in view of the dimensions of the spaces involved we may conclude that $\mathbf{A}\mathcal{V}_m \in \text{span}(\mathcal{V}_m, \mathbf{A}\mathbf{u}_m)$, which, by Lemma 1, proves (a).

"(a) $\Rightarrow$ (c)": If $\mathcal{V}_m$ is a Krylov subspace of order $m$ generated by $\mathbf{v}_0$, that is if (a) holds, then, also in view of (2.12), we have that

$$\text{span}(\mathcal{V}_m, \mathbf{t}_{\text{JD}}) = \text{span}(\mathcal{V}_m, (\mathbf{A}-\theta\mathbf{I}_n)^{-1}\mathbf{u}_m) \subset \{q(\mathbf{A})[(\mathbf{A}-\theta\mathbf{I}_n)^{-1}\mathbf{v}_0] \mid q \text{ pol. degree } \leq k\}.$$

The inclusion follows easily from the representation of $\mathcal{V}_m$ as $\mathcal{V}_m = \{p(A)\mathbf{v}_0 | p \text{ pol. degree } < k\}$. If $(\mathbf{A} - \theta\mathbf{I}_n)^{-1}\mathbf{u}_m \notin \mathcal{V}_m$ then a dimension argument shows that the subspaces coincide which proves that $\text{span}(\mathcal{V}_m, \mathbf{t}_{\text{JD}})$ is a Krylov subspace. If $(\mathbf{A} - \theta\mathbf{I}_n)^{-1}\mathbf{u}_m \in \mathcal{V}_m$ then there is no expansion and the Krylov structure is trivially preserved. □

Lemma 1 implies that any $n-1$ dimensional subspace is a Krylov subspace. In particular, $\text{span}(\mathcal{V}_m, \mathbf{t}_{\text{JD}})$ is a Krylov subspace if $\mathcal{V}_m$ is $n-2$-dimensional and it does not contain $\mathbf{t}_{\text{JD}}$. From this argument it can be seen that (c) does not imply (a).

Since $\mathbf{t}_{\text{JD}}$ is also a solution of (2.10), we have the following.

COROLLARY 2 *If $\mathcal{V}_m$ is a Krylov subspace then* $\text{span}(\mathcal{V}_m, \mathbf{t}_{\text{JD}}) \subset \text{span}(\mathcal{V}_m, \mathbf{t}_{\text{JDV}})$. □

If $\theta_m$ is simple then $\mathbf{t}_{\text{JD}} \notin \mathcal{V}_m$ and the expanded subspaces in Cor. 2 coincide. However, as the example in §2.3.2 shows, JD may not always expand the subspace. Note that, in accordance with (c) of Th. 1, the subspace $\mathcal{V}_2$ in this example is a Krylov subspace (generated by $\mathbf{A}$ and $\mathbf{v}_0 = e_2 - e_1$).

Cor. 2 does not answer the question whether $\mathbf{t}_{\text{JD}}$ and $\mathbf{t}_{\text{JDV}}$ lead to the same expansion of $\mathcal{V}_m$ only if $\mathcal{V}_m$ is a Krylov subspace. The example in §2.3 shows that the answer can be negative, namely if $\mathbf{t}_{\text{JD}} \perp \mathcal{V}_m$: then $\gamma = \mathbf{V}_m^*\mathbf{t}_{\text{JD}} = 0$. The answer can also be negative in cases where $\mathbf{t}_{\text{JD}} \not\perp \mathcal{V}_m$, provided that the dimension of the subspace $\mathcal{V}_m$ is larger than 2. The following theorem characterizes partially the situation where we obtain the same expansion. Note that $\mathcal{V}_m$ is a Krylov subspace if and only if the dimension of $\mathbf{A}\mathcal{V}_m \cap \mathcal{V}_m$ is at most one less than the dimension of $\mathcal{V}_m$ (see Lemma 1).

THEOREM 2 *If* $\text{span}(\mathcal{V}_m, \mathbf{t}_{\text{JD}}) \subset \text{span}(\mathcal{V}_m, \mathbf{t}_{\text{JDV}})$ *then* $\mathbf{A}\mathcal{V}_m \cap \mathcal{V}_m \neq \{\mathbf{0}\}$ *or* $\mathbf{t}_{\text{JD}} \perp \mathcal{V}_m$.

*Proof.* If $\mathbf{t}_{\text{JD}}$ and $\mathbf{t}_{\text{JDV}}$ give the same expansion then (2.13) shows that $\gamma\mathbf{A}\mathbf{u}_m + \mathbf{A}\mathbf{V}_m(\mathbf{V}_m^*\mathbf{t}_{\text{JD}}) \in \mathcal{V}_m$. Apparently, $\mathbf{A}\mathcal{V}_m \cap \mathcal{V}_m \neq \{\mathbf{0}\}$ or $\gamma = 0$ and $\mathbf{V}_m^*\mathbf{t}_{\text{JD}} = 0$. A similar argument applies to the case where $\mathbf{t}_{\text{JD}} \in \mathcal{V}_m$. □

In practical situations, where $\mathcal{V}_m$ is constructed from inexact solutions of the correction equations it will be unlikely that $\mathbf{A}\mathcal{V}_m$ will have a non-trivial intersection with $\mathcal{V}_m$ (unless the dimension of $\mathcal{V}_m$ is larger than $n/2$). Usually $\mathbf{t}_{\text{JD}} \not\perp \mathcal{V}_m$. Therefore, the exact expansion vectors $\mathbf{t}_{\text{JD}}$ and $\mathbf{t}_{\text{JDV}}$ will not lead to same expansions, and we may not expect that inexact expansion vectors will produce the same expansions.

The correction equation (2.11) in inexact TRQ is based on a Krylov subspace: the matrix $\mathbf{V}_m$ in this algorithm is produced by the Arnoldi procedure whenever equation (2.11) has to be solved.

### 2.4.2 Starting with one vector

As any one dimensional subspace is a Krylov subspace, one consequence of Theorem 1 is the following corollary. The proof follows by an inductive combination of Th. 1(c) and Cor. 2.

COROLLARY 3 *Exact JD and exact JDV started with the same vector* $\mathbf{u}_1$ *are mathematically equivalent as long as exact JD expands, i.e., they produce the same sequence of search subspaces in exact arithmetic.*

### 2.4.3 (Re-)Starting with several Ritz vectors

Once we start JD and JDV with one vector the dimension of the search subspace starts increasing. After a number of steps a restart strategy must be followed to keep the required storage limited and the amount of work related to the search subspace low. The question is which information should be thrown away and which should be kept in memory. A popular strategy is to select those Ritz pairs that are close to a specified shift/target. Cor. 1 and an inductive application of Theorem 1 imply that, with a one-dimensional initial start and restarts with the selected Ritz vectors, restarted exact JD and restarted exact JDV are mathematically equivalent.

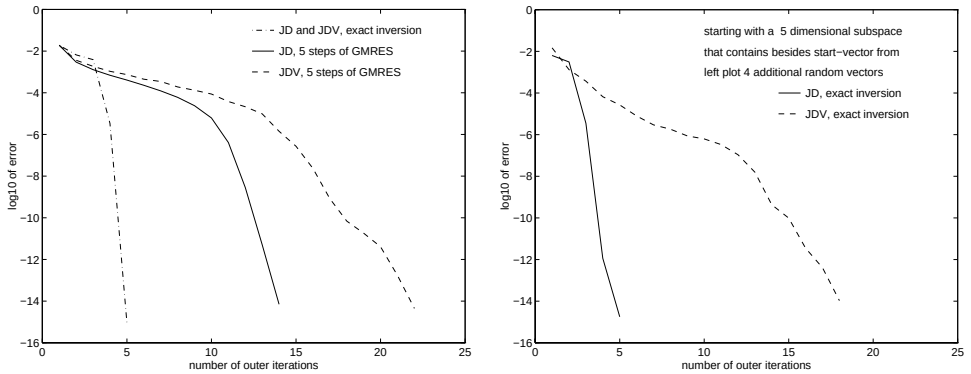
COROLLARY 4 *Exact JD and exact JDV are mathematically equivalent as long as exact JD expands if they are both started with the same set of Ritz vectors of* $\mathbf{A}$ *with respect to one Krylov subspace.*

In practice, we have to deal with round off errors and the correction equations can only be solved with a modest accuracy. Therefore, even if we start with one vector or a Krylov subspace, the subsequent search subspaces will not be Krylov and the results in the above corollaries do not apply. If a search subspace is not Krylov, then from Th. 1 we learn that the "in between" variant may lead to expansions different from those of JDV. Th. 2 indicates that also JD will differ from JDV.

## 2.5 Numerical experiments

Here a few numerical experiments will be presented. We will see that JDV and JD show comparable speed of convergence also in finite precision arithmetic as long as the correction equations are solved in high precision (§2.5.1.1). JDV converges much slower than JD if the Krylov structure of the search subspace is seriously perturbed. We will test this by starting with a low dimensional random space (§2.5.1.1). We will also see this effect in our experiments where we solved the correction equations only in modest accuracy (§2.5.1.2). Moreover, we will be interested in the question whether the slower convergence of JDV in case of inaccurate solutions of the correction equations can be compensated by a better performance of the linear solver for the correction equation (§2.5.2.1). Further, some stability issues will be addressed (§2.5.1.3).

FIGURE 2.1. *Convergence plots for Example 1. Differences between* JD *and* JDV *when not solving the correction equation exactly* (*left plot*) *and when starting with an unstructured 5-dimensional subspace* (*right plot*)*. The plots show the* $\log_{10}$ *of the error* $|\theta_m - \lambda|$ *in the Ritz value* $\theta_m$ *versus the iteration number* $m$*.*




## 2.5.1 Example 1

In the experiments in this section 2.5.1, we apply the Jacobi-Davidson method on a tridiagonal matrix of order 100 with diagonal entries 2.4 and off-diagonal entries 1 ([46, Ex. 1]). Our aim is the largest eigenvalue $\lambda = 4.3990\ldots$. We start with a vector with all entries equal to 0.1.

### 2.5.1.1 Exact solution of the correction equation

When solving the correction equations exactly no difference between JD and JDV is observed (dash-dotted line in left plot in Fig. 2.1) which is in accordance with Cor. 3. The plots show the $\log_{10}$ of the error $|\theta_m - \lambda|$ in the Ritz value $\theta_m$ versus the iteration number $m$.

To see the effect of starting with an arbitrary subspace of dimension larger than 1 we added four random vectors to the start vector with all entries equal to $0.1$. The right plot in Fig. 2.1 shows the convergence of exact JD (solid curve) and JDV (dashed curve). Here the results of `seed(253)` in our MATLAB-code are presented (other seeds showed similar convergence behavior). The correction equations have been solved "exactly", that is to machine precision. As anticipated in §2.4.1 (see Th. 2) the convergence behavior of JDV now clearly differs from that of JD. Moreover, the speed of convergence of JDV seems to be much lower than of JD (linear rather than cubic? See §2.2.4). Apparently, expanding with $\mathbf{t}_{\text{JDV}}$ rather than with $\mathbf{t}_{\text{JD}}$ may slow down the convergence of Jacobi-Davidson considerably in case the initial subspace is not a Krylov subspace.

Note that JD performs slightly better with the five-dimensional start than with the one-dimensional start (compare the solid curve in the right plot with de dashed-dotted curve in the left plot). This may be caused by the extra (noisy) search directions.

#### 2.5.1.2 Approximate solution of the correction equation

If the correction equations are not solved in high precision, we may not expect the constructed search subspaces $\mathcal{V}_m$ to be Krylov subspaces, even if the process is started with a Krylov subspace. Consequently $\mathbf{t}_{\text{JD}}$ and $\mathbf{t}_{\text{JDV}}$, and therefore their inexact approximations, will not lead to the same expansions of $\mathcal{V}_m$. In view of the experimental result in §2.5.1.1, we expect the inexact JDV to converge slower than inexact JD.

Again we start with one vector, but we use only 5 steps of GMRES to get an approximate solution of the correction equation in each outer iteration. The solid line (JD) and the dashed line (JDV) in the left plot of Fig. 2.1 show the results. JDV needs significantly more outer iterations for convergence than JD.

#### 2.5.1.3 Loss of orthogonality

The (approximate) solution of (2.2) in JD will in general not be orthogonal to $\mathbf{V}_m$. Therefore, this solution is orthonormalized against $\mathbf{V}_m$ before it is used to expand $\mathbf{V}_m$ to $\mathbf{V}_{m+1}$. We refer to this step in the algorithm as *post*-orthogonalization (of the solution of the correction equation). In JDV, however, if the correction equation (2.3) is solved with, for instance, GMRES, then the (approximate) solution should be orthogonal to $\mathbf{V}_m$ and post-orthogonalization, i.e., the explicit orthogonalization before expanding $\mathbf{V}_m$, should be superfluous. This observation would offer a possibility of saving inner products. Here we investigate what the effect is of omitting the post-orthogonalization in JDV.

Again JDV is applied on the simple test matrix with the same starting vector as before and the correction equations are solved approximately with 5 steps of GMRES. As initial approximate solution for GMRES we take the zero vector.

From the experiment we learn that without post-orthogonalization the basis of the search subspace in JDV loses its orthogonality. As a measure for the orthonormality of $\mathbf{V}_m$ we took (see [37, §13.8]) $\kappa_m \equiv \|I_m - \mathbf{V}_m^* \mathbf{V}_m\|$. Table 2.1 lists the values of the error $|\lambda - \theta_m|$ and the quantity $\kappa_m$ for the first 10 outer iterations. Column two and three ("with post-ortho.") show the results for the implementation of JDV where the approximate solution of the correction equation is explicitly orthogonalized against $\mathbf{V}_m$ before it is used to expand this matrix. In the columns four and five ("without post-ortho.") we see that if the post-orthogonalization is omitted then the loss of orthonormality starts influencing the error significantly after just 5 outer iterations. After 8 iterations the orthonormality is completely lost. This phenomenon can be explained as follows.

The residual of the selected Ritz pair is computed as $\mathbf{r}_m = \mathbf{A}\mathbf{u}_m - \theta_m \mathbf{u}_m$. Therefore, in finite precision arithmetic, the residual will not be as orthogonal to the search subspace as intended even if $\mathbf{V}_m$ would have been orthonormal. For instance, at the second iteration of our experiment, we have an error $\|\mathbf{V}_2^* \mathbf{r}_2\|$ equal to $1.639\text{e}{-13}$. With the norm of the residual equal to $0.02145$ this results in a relative error of $7.640\text{e}{-12}$. Note that, specifically at convergence, rounding errors in $\mathbf{r}_m$ may be expected to lead to relatively big errors. In each solve of the correction equation (2.3), GMRES is started with initial approximate $\mathbf{0}$ and the vector $\mathbf{r}_m$ is taken as the initial residual in the GMRES process.

TABLE 2.1. *The need of post-orthogonalization when using* JDV. *For the simple test, the* JDV *correction equation* (2.3) *is solved approximately with 5 steps of* GMRES. *The table shows the error* $|\lambda - \theta_m|$ *in the Ritz value* $\theta_m$ *and the "orthonormality" of the basis* $\mathbf{V}_m$ *of the search subspaces* ($\kappa_m = \|I_m - \mathbf{V}_m^* \mathbf{V}_m\|$) *for the implementation with post-orthogonalization of the solution of the correction equation* (*column two and three*), *without post-orthogonalization* (*column four and five*), *and without post-orthogonalization, but with pre-orthogonalization of the left-hand side vector of the correction equation* (*column six and seven*).

| | with post-ortho. | | without post-ortho. | | with pre-ortho. | |
|---|---|---|---|---|---|---|
| m | $\lvert\lambda - \theta_m\rvert$ | $\kappa_m$ | $\lvert\lambda - \theta_m\rvert$ | $\kappa_m$ | $\lvert\lambda - \theta_m\rvert$ | $\kappa_m$ |
| 1 | 1.903e−02 | 2.220e−16 | 1.903e−02 | 2.220e−16 | 1.903e−02 | 2.220e−16 |
| 2 | 3.611e−03 | 2.289e−15 | 3.611e−03 | 3.690e−14 | 3.611e−03 | 3.690e−14 |
| 3 | 1.856e−03 | 2.314e−15 | 1.856e−03 | 1.426e−11 | 1.856e−03 | 4.567e−14 |
| 4 | 1.076e−03 | 2.314e−15 | 1.076e−03 | 2.649e−09 | 1.076e−03 | 4.866e−14 |
| 5 | 7.480e−04 | 2.316e−15 | 7.480e−04 | 6.621e−07 | 7.480e−04 | 5.920e−14 |
| 6 | 4.464e−04 | 2.316e−15 | 4.423e−04 | 1.125e−04 | 4.464e−04 | 6.534e−14 |
| 7 | 3.454e−04 | 2.317e−15 | 4.135e−04 | 2.710e−02 | 3.454e−04 | 7.490e−14 |
| 8 | 1.909e−04 | 2.317e−15 | 3.135e+00 | 9.732e−01 | 1.909e−04 | 9.546e−14 |
| 9 | 1.317e−04 | 2.317e−15 | 7.004e+00 | 1.940e+00 | 1.317e−04 | 9.548e−14 |
| 10 | 8.747e−05 | 2.317e−15 | 1.094e+01 | 2.920e+00 | 8.747e−05 | 1.232e−13 |

Since $\mathbf{r}_m$ is supposed to be orthogonal against $\mathbf{V}_m$, this vector is not explicitly orthogonalized against $\mathbf{V}_m$, and the normalized $\mathbf{r}_m$ is simply taken as the first Arnoldi vector. In the subsequent GMRES steps the Arnoldi vectors are obtained by orthogonalization against $\mathbf{V}_m$ followed by orthogonalization against the preceding Arnoldi vectors. However, since the first Arnoldi vector will not be orthogonal against $\mathbf{V}_m$, the approximate GMRES solution will not be orthogonal against $\mathbf{V}_m$. Adding this "skew" vector to the basis of the search subspace will add to the non-orthogonality in the basis.
Columns six and seven ("with pre-ortho.") of Table 2.1 show that post-orthogonalization can be omitted as long as the residual $\mathbf{r}_m$ is sufficiently orthogonal with respect to $\mathbf{V}_m$: the post-orthogonalization is omitted here, but the right-hand side vector of the correction equation, the residual $\mathbf{r}_m$, is orthogonalized explicitly against $\mathbf{V}_m$ before solving the correction equation (pre-orthogonalization). Since pre- and post-orthogonalization are equally expensive and since pre-orthogonalization appears to be slightly less stable (compare the $\kappa_m$'s in column 3 with those in column 7 of Table 2.1), pre-orthogonalization is not an attractive alternative, but the experimental results confirm the correctness of the above arguments.

Note that our test matrix here is only of order $100$ and the effect of losing orthogonality may become even more important for matrices of higher order.

Also in JD the finite precision residual $\mathbf{r}_m$ of the Ritz pair will not be orthogonal to the search subspace. Since even in exact arithmetic you may not expect the solution of the JD correction equation (2.2) to be orthogonal to $\mathbf{V}_m$, post-orthogonalization is essential in the JD variant. In our experiment, using finite precision arithmetic, we did not observe any significant loss of orthogonality in the column vectors of $\mathbf{V}_m$. Nevertheless, we also checked

whether pre-orthogonalization of $\mathbf{r}_m$ before solving the correction equation would enhance the convergence of JD. This was not the case: JD converged equally fast with and without pre-orthogonalization.

In the remaining experiments we used post-orthogonalization in JDV, too.

### 2.5.2 Example 2

In this section we consider a slightly more realistic eigenvalue problem. We are interested in the question whether the projections on the orthogonal complement of $\mathbf{V}_m$ in the JDV approach may significantly improve the performance of the linear solver for the correction equation.

For $\mathbf{A}$ we take the SHERMAN1 matrix from the Harwell-Boeing collection [21]. The matrix is real unsymmetric of order $1000$. All eigenvalues appear to be real and in the interval [-5.0449,-0.0003]. About 300 eigenvalues are equal to -1. We want to find a few eigenvalues with associated eigenvectors that are closest to the target $\sigma$. Our target $\sigma$ is set to -2.5. Note that the "target" eigenvalues are in the "interior" of the spectrum, which make them hard to find, no matter the numerical method employed.

In general, when started with a single vector, the Ritz values in the initial stage of the process will be relatively inaccurate approximations of the target eigenvalue $\lambda$, that is, if $\lambda$ is the eigenvalue closest to $\sigma$ then for the first few $m$ we will have that $|\theta_m - \lambda|/|\sigma - \lambda| \gg 1$. Therefore, as argued in [44, §9.4] (see also [25, §4.0.1]), it is more effective to replace initially $\theta_m$ in the correction equation by $\sigma$ (similar observations can be found in [33, §6] and [52, §3.1]). As the search subspace will not contain significant components of the target eigenvectors in this initial stage, the projections in (2.2) and (2.3) are not expected to be effective. Therefore, we expanded the search subspace in the first few steps of our process by approximate solutions of the equation

$$(\mathbf{A} - \sigma \mathbf{I}_n)\mathbf{t} = -\mathbf{r}_m, \tag{2.17}$$

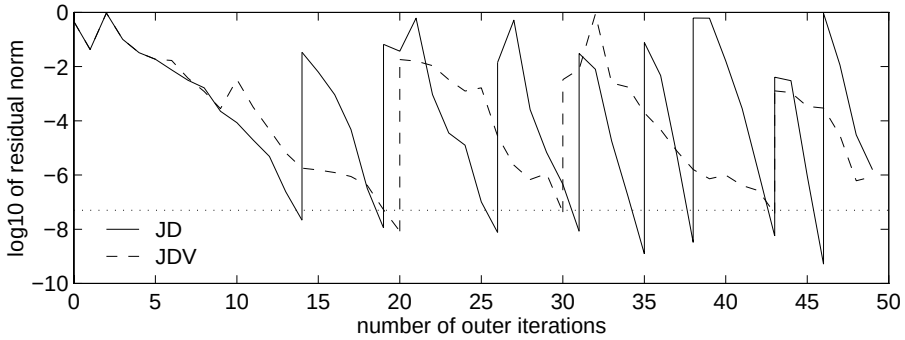
which can be viewed as a generalized Davidson approach.

In the computations we did not use any preconditioning. We started JD and JDV with the same vector, the vector of norm one of which all entries are equal. The algorithms were coded in C and run on a Sun SPARCstation 4 using double precision.

#### 2.5.2.1 Solving the correction equation in lower precision

Fig. 2.2 shows the $\log_{10}$ of the residual norm for JD (the solid curve) and for JDV (the dashed curve). In this example, all correction equations (including (2.17)) have been solved with 50 steps of GMRES except where GMRES reached a residual accuracy of $10^{-14}$ in an earlier stage. In the first 5 steps of the outer iteration we took the approximate solution of the Davidson correction equation (2.17) as the expansion vector. As the correction equations are not solved exactly, we expect that JD will need less outer iterations than JDV (see §§2.4.1 and 2.5.1.2), which is confirmed by the numerical results in the figure.

FIGURE 2.2. *The convergence history for the computation of eigenpairs with eigenvalue closest to* $-2.5$ *of the matrix* SHERMAN1. *The plot shows the* $\log_{10}$ *of the subsequent residual norms for* JD (*solid curve*) *and* JDV (*dashed curve*) *versus the iteration number* $m$. *A search for a next eigenpair is started when a Ritz pair is accepted as eigenpair* (*i.e., if* $\|\mathbf{r}_m\|_2 \leq 5\,10^{-8}$). *The correction equations are approximately solved with 50 steps of* GMRES.




As argued in §2.1, the projections on the orthogonal complement of $\mathbf{V}_m$ in the JDV correction equation (2.3) may improve the conditioning (or more general, the spectral properties) of the operator in the correction equation. This may allow a more efficient or a more accurate way of solving the correction equation. Here we test numerically whether a better performance of the linear solver for the correction equations can compensate for a loss of speed of convergence in the outer iteration. In the figures in Fig. 2.3 we show how the performance of JD and JDV and the computational costs relate. As a measure for the costs we take the number of matrix-vector multiplications: we plot the $\log_{10}$ of the residual norm versus the number of matrix-vector multiplications by $\mathbf{A}$ (or by $\mathbf{A} - \theta_m \mathbf{I}_n$). Note that this way of measuring the costs favours JDV, since the projections in JDV are more costly than in JD. Nevertheless, we will see that JD outperforms JDV.

We solve all correction equations with GMRES$_\ell$, that is with $\ell$ steps of GMRES, except where GMRES reaches a residual accuracy of $10^{-14}$ in an earlier stage. For $\ell$ we took 200 (top figure), 50 (middle figure), and 25 (bottom figure). In the first few outer iterations the Davidson correction equation (2.17) is solved approximately (2 outer iterations for $\ell = 200$ and 5 for $\ell = 50$ and for $\ell = 25$). When a Ritz pair is accepted as eigenpair (i.e., if $\|\mathbf{r}_m\| \leq 5\,10^{-8}$), a search is started for the next eigenpair. The accepted Ritz pairs are kept in the search subspace. Explicit deflation is used only in the correction equation (see [26]). Note that the correction equations (2.3) in JDV need no modification to accommodate the deflation, because accepted Ritz vectors are kept in the search space.

If GMRES would converge faster on JDV correction equations than on JD correction equations, then GMRES would need less steps for solving (2.3) in case the residual accuracy of $10^{-14}$ would be reached in less than $\ell$ GMRES steps, while in the other case it would produce more effective expansion vectors in JDV. With more effective expansion vectors the number of outer iterations may be expected to decrease. In both cases, there would be a positive effect on the number of matrix-vector multiplications needed in JDV.

TABLE 2.2. *Costs for the computation of two eigenpairs of* SHERMAN1 *with* JD *and* JDV. *The costs* (*b*) *for the computation of the second eigenpair* ($\lambda = -2.51545\ldots$) *include the costs* (*a*) *for the computation of the first eigenpair* ($\lambda = -2.49457\ldots$).

| method for the correction equation | | number of outer iterations | | number of matrix-vector multiplications | | wallclock time in seconds | |
|---|---|---|---|---|---|---|---|
| | | JD | JDV | JD | JDV | JD | JDV |
| $\text{GMRES}_{200}$ | (*a*) | 4 | 4 | 798 | 790 | 64.1 | 64.3 |
| | (*b*) | 7 | 7 | 1401 | 1393 | 114.7 | 119.5 |
| $\text{GMRES}_{50}$ | (*a*) | 14 | 20 | 715 | 1021 | 21.5 | 51.2 |
| | (*b*) | 19 | 30 | 970 | 1531 | 35.0 | 121.1 |
| $\text{GMRES}_{25}$ | (*a*) | 26 | 37 | 677 | 963 | 41.3 | 143.0 |
| | (*b*) | 33 | 47 | 859 | 1223 | 83.2 | 301.4 |

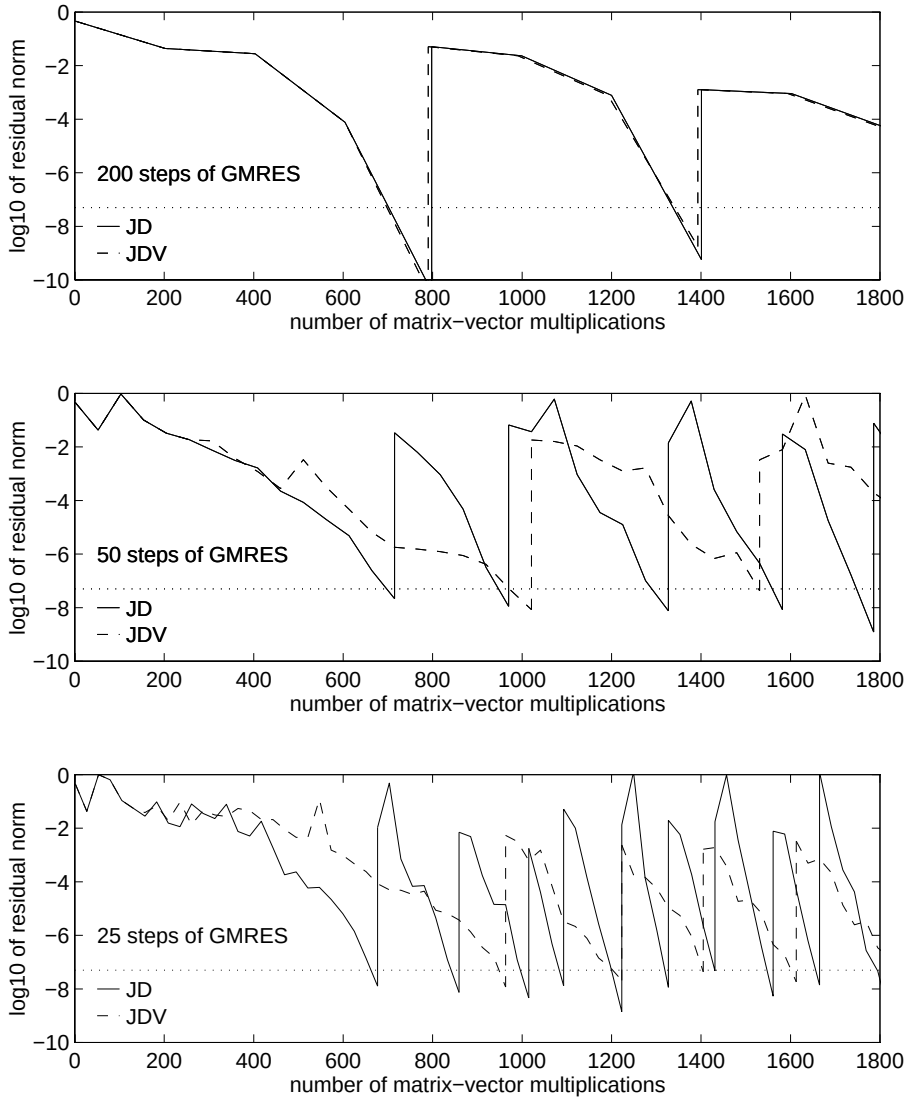
In Table 2.2 the number of outer iterations, the number of matrix-vector multiplications and the amount of time needed for the computation for the first two eigenpairs ($\lambda = -2.49457\ldots$ and $\lambda = -2.51545\ldots$) are presented.

When solving the correction equation with 200 steps of GMRES no difference between JD and JDV is observed (upper plot in Fig. 2.3). Apparently with 200 steps of GMRES the correction equations are solved in high precision and the results are in line with the theory and our previous experience. This can also be seen from Table 2.2. For the first eigenvalue JD uses 8 more matrix-vector multiplications than the 790 from JDV. On the other hand JDV takes a bit more time (about 0.2 seconds) than JD. From this we may conclude that, compared with the costs of the matrix-vector multiplications and the QR-algorithm for the computation of the eigenvalues of the projected matrix, the extra vector-vector operations involved in the correction equation of JDV are not very expensive.

Although JD and JDV need the same amount of time for convergence when using 200 steps of GMRES, the same eigenpairs can be computed in much less time. If 50 steps of GMRES are used, JD takes only 21.45 seconds for computing the first eigenpair whereas JDV takes 2.5 times that amount.

The differences between the two methods become more significant if we lower the precision of the solver for the correction equation by using only 25 steps of GMRES. With the same amount of matrix-vector multiplications the number of eigenpairs found by JD is much higher than JDV. Note, that the measured time for both JD and JDV in the case of $\text{GMRES}_{25}$ is more than in the case of $\text{GMRES}_{50}$ whereas the number of matrix-vector multiplications is less. The reason for this can only be the fact that in the case of $\text{GMRES}_{25}$ more outer iterations are needed, every outer iteration the eigenvalues of the projected matrix are computed with a QR-algorithm.

FIGURE 2.3. *The effect of reducing the precision of the solution method for the correction equation. The figures display the convergence history for the computation of eigenpairs with eigenvalue closest to* $-2.5$ *of the matrix* SHERMAN1. *Plotted are the* $\log_{10}$ *of the subsequent residual norms for* JD (*solid curve*) *and* JDV (*dashed curve*) *versus the number of matrix-vector multiplications. The correction equations are approximately solved with 200* (*top figure*), *50* (*center figure*) *and 25* (*bottom figure*) *steps of* GMRES.



## 2.6 Conclusions

In GMRESR, an iterative method for solving linear systems of equations, it pays to restrict the correction equations to the orthogonal complement of the space spanned by the search vectors. This approach, called GCRO, leads to new search directions that are automatically orthogonal with respect to the old ones. Although the restricted correction equations require more complicated projections with higher computational costs per matrix-vector multiplication, the number of matrix-vector multiplications may decrease tremendously leading to a better overall performance [18, 5]. In this chapter, we investigated the question whether such an approach would be equally effective for the Jacobi-Davidson method for solving the eigenvalue problem. Note that eigenvalue problems are weakly non-linear.

When starting with a Krylov subspace and solving the correction equations exactly the standard approach (JD) of Jacobi-Davidson and its variant JDV with the more restricted correction equations, are mathematically equivalent (§2.4). However, in practical situations, where the correction equations are solved only in modest accuracy with finite precision arithmetic, the JDV variant appears to converge much more slowly than JD. Although the restricted correction equations in JDV may have spectral properties that are more favourable for linear solvers, a better performance of the linear solvers for the correction equation in JDV may not compensate for the slower convergence.

# Chapter 3

# Using domain decomposition in the Jacobi-Davidson method

Menno Genseberger, Gerard Sleijpen, and Henk van der Vorst

**Abstract**

The Jacobi-Davidson method is suitable for computing solutions of large $n$-dimensional eigenvalue problems. It needs (approximate) solutions of specific $n$-dimensional linear systems. Here we propose a strategy based on a nonoverlapping domain decomposition technique in order to reduce the wall clock time and local memory requirements. For a model eigenvalue problem we derive optimal coupling parameters. Numerical experiments show the effect of this approach on the overall Jacobi-Davidson process. The implementation of the eventual process on a parallel computer is beyond the scope of this chapter.

**Keywords:** Eigenvalue problems, domain decomposition, Jacobi-Davidson, Schwarz method, nonoverlapping, iterative methods.

**2000 Mathematics Subject Classification:** 65F15, 65N25, 65N55.

## 3.1 Introduction

The Jacobi-Davidson method [46] is a valuable approach for the solution of large (generalized) linear eigenvalue problems. The method reduces the large problem to a small one by projecting it on an appropriate low dimensional subspace. Approximate solutions for eigenpairs of the large problem are obtained from the small problem by means of a Rayleigh-Ritz principle. The key to the Jacobi-Davidson method is how the subspace is expanded. To keep the dimension of the subspace, and consequently the size of the small problem, low it is essential that all necessary information of the wanted eigenpair(s) is collected in the subspace after a small number of iterations. Therefore, the subspace should be expanded with a vector that contains important information not already present in the subspace. The correction equation of the Jacobi-Davidson method aims at prescribing such a vector.

But in itself, the correction equation poses a large linear problem, with size equal to the size of the originating large eigenvalue problem. Because of this, most of the computational work of the Jacobi-Davidson method arises from solving the correction equation. In practice the eigenvalue problem is often so large that an accurate solution of the correction equation is too expensive. However, often approximate solutions of the correction equation suffice to obtain sufficiently fast convergence of the Jacobi-Davidson method. The speed of this convergence depends on the accuracy of the approximate solution. Jacobi-Davidson lends itself to be used in combination with a preconditioned iterative solver for the correction equation. In such a case the quality of the preconditioner is critical.

Nonoverlapping domain decomposition methods for *linear systems* have been studied well in the literature. Because of the absence of overlapping regions they have computational advantages compared to domain decomposition methods with overlap. But much depends on the coupling that should be chosen carefully.

In this chapter we will show how a nonoverlapping domain decomposition technique [55, 56] can be incorporated in the correction equation of Jacobi-Davidson, when applied to PDE type of eigenvalue problems. The technique is based on work by Tang and by Tan and Borsboom for linear systems.

For a linear system Tang [57] proposed to enhance the system with duplicates in order to enable an additive Schwarz method with minimal overlap (for more recent publications, see for example [16], [32] and [28]). Tan and Borsboom [55, 56] refined this idea by introducing more flexibility for the unknowns near the interfaces between the subdomains. In this way additional degrees of freedom are created, reflected by coupling equations for the unknowns near the interfaces and their virtual counterparts. Now, the key point is to tune these interface conditions for the given problem in order to improve the speed of convergence of the iterative solution method. This approach is very effective for classes of linear systems stemming from advection-diffusion problems [55, 56].

The operator in the correction equation involves the matrix of the large eigenvalue problem shifted by an approximate eigenvalue. In the computational process, this shift will become arbitrarily close to the desired eigenvalue. This is a situation that requires special attention when applying the domain decomposition technique.

An eigenvalue problem is a mildly nonlinear problem. Therefore, for the computation

of solutions to the eigenvalue problem one needs a nonlinear solver, for instance, a Newton method. In fact, Jacobi-Davidson can be seen as an accelerated inexact Newton method [45]. Here, we shall, as explained above, combine the Jacobi-Davidson method with a Krylov solver for the correction equation. A preconditioner for the Krylov solver is constructed with domain decomposition. A similar type of nesting, but for general nonlinear systems, can be found in the Newton-Krylov-Schwarz algorithms by Cai, Gropp, Keyes et al. in [9] and [10]. In these two papers the subdomains have overlap, therefore there is no analysis for the tuning of the coupling between subdomains. Furthermore, the eigenvalue problem is nonlinear but with a specific structure; we will exploit this structure.

This chapter is organized as follows. First, we recall the enhancement technique for domain decomposition in §3.2. Then, in §3.3 we discuss the Jacobi-Davidson method. We outline how the technique can be applied to the correction equation and how the projections in the correction equation should be handled. For a model eigenvalue problem we investigate, in §3.4, in detail how the coupling equations should be chosen for optimal performance. It will turn out that the shift plays a critical role. Section §3.5 gives a number of illustrative numerical examples.

## 3.2 Domain decomposition

### 3.2.1 Canonical enhancement of a linear system

Tang [57] has proposed the concept of matrix enhancement, which gives elegant possibilities for the formulation of effective domain decomposition of the underlying PDE problem. The idea is to decompose the grid into nonoverlapping subgrids and to expand the subgrids by introducing additional gridpoints and additional unknows along the interfaces of the decomposition. This approach artificially creates some overlap on gridpoint level and the overlap is minimal. For hyperbolic systems of PDEs, this approach was further refined by Tan in [56] and by Tan and Borsboom in [55]. Discretization of the PDE leads to a linear system of equations. Tang duplicates and adjusts those equations in the system that couple across the interfaces. Tan and Borsboom introduce a double set of additional gridpoints along the interfaces in order to keep each equation confined to one expanded subgrid. As a consequence, none of the equations has to be adjusted. Then they enhanced the linear system by 'new' equations that can be viewed as discretized boundary conditions for the internal boundaries (along the interfaces). Since the last approach offers more flexibility, this is the one we follow.

We start with the linear nonsingular system

$$\mathbf{B}\,\mathbf{y} = \mathbf{d}, \tag{3.1}$$

that results from discretization of a given PDE over some domain. Now, we partition the

matrix $\mathbf{B}$, and the vectors $\mathbf{y}$ and $\mathbf{d}$ correspondingly,

$$\begin{bmatrix} \mathbf{B}_{11} & \mathbf{B}_{1\ell} & \mathbf{B}_{1r} & \mathbf{B}_{12} \\ \mathbf{B}_{\ell 1} & B_{\ell\ell} & B_{\ell r} & \mathbf{B}_{\ell 2} \\ \mathbf{B}_{r1} & B_{r\ell} & B_{rr} & \mathbf{B}_{r2} \\ \mathbf{B}_{21} & \mathbf{B}_{2\ell} & \mathbf{B}_{2r} & \mathbf{B}_{22} \end{bmatrix}, \quad \begin{bmatrix} \mathbf{y}_1 \\ y_\ell \\ y_r \\ \mathbf{y}_2 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} \mathbf{d}_1 \\ d_\ell \\ d_r \\ \mathbf{d}_2 \end{bmatrix}.$$

The labels are not chosen arbitrarily: we associate with label 1 (and 2, respectively) elements-/operations of the linear system corresponding to subdomain 1 (2, respectively) and with label $\ell$ (resp. $r$) elements/operations corresponding to the left (resp. right) of the interface between the two subdomains. The central blocks $B_{\ell\ell}$, $B_{\ell r}$, $B_{r\ell}$ and $B_{rr}$ are square matrices of equal size, say, $n_i$ by $n_i$. They correspond to the unknowns along the interface. Since the number of unknowns along the interface will typically be much smaller than the total number of unknows, $n_i$ will be much smaller than $n$, the size of $\mathbf{B}$.

For a typical discretization, the matrix $\mathbf{B}$ is banded and the unknowns are only locally coupled. Therefore, under adequate ordering we can manage that $\mathbf{B}_{r1}, \mathbf{B}_{21}, \mathbf{B}_{12}$ and $\mathbf{B}_{\ell 2}$ are zero. For this situation, we define the 'canonical enhancement' $\mathbf{B}_I$ of $\mathbf{B}$, $\underset{\sim}{\mathbf{y}}$ of $\mathbf{y}$, and $\underline{\mathbf{d}}$ of $\mathbf{d}$, by

$$\mathbf{B}_I \equiv \begin{bmatrix} \mathbf{B}_{11} & \mathbf{B}_{1\ell} & \mathbf{B}_{1r} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{B}_{\ell 1} & B_{\ell\ell} & B_{\ell r} & 0 & 0 & \mathbf{0} \\ \mathbf{0} & I & 0 & -I & 0 & \mathbf{0} \\ \mathbf{0} & 0 & -I & 0 & I & \mathbf{0} \\ \mathbf{0} & 0 & 0 & B_{r\ell} & B_{rr} & \mathbf{B}_{r2} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{B}_{2\ell} & \mathbf{B}_{2r} & \mathbf{B}_{22} \end{bmatrix}, \quad \underset{\sim}{\mathbf{y}} \equiv \begin{bmatrix} \mathbf{y}_1 \\ y_\ell \\ \widetilde{y}_r \\ \widetilde{y}_\ell \\ y_r \\ \mathbf{y}_2 \end{bmatrix}, \text{ and } \quad \underline{\mathbf{d}} \equiv \begin{bmatrix} \mathbf{d}_1 \\ d_\ell \\ 0 \\ 0 \\ d_r \\ \mathbf{d}_2 \end{bmatrix}. \tag{3.2}$$

One easily verifies that $\mathbf{B}_I$ is also nonsingular and that $\underset{\sim}{\mathbf{y}}$ is the unique solution of

$$\mathbf{B}_I \underset{\sim}{\mathbf{y}} = \underline{\mathbf{d}}, \tag{3.3}$$

with $\underset{\sim}{\mathbf{y}} \equiv (\mathbf{y}_1^T, y_\ell^T, y_r^T, y_\ell^T, y_r^T, \mathbf{y}_2^T)^T$.

With this linear system we can associate a simple iterative scheme for the two coupled subblocks:

$$\begin{bmatrix} \mathbf{B}_{11} & \mathbf{B}_{1\ell} & \mathbf{B}_{1r} \\ \mathbf{B}_{\ell 1} & B_{\ell\ell} & B_{\ell r} \\ \mathbf{0} & I & 0 \end{bmatrix} \begin{bmatrix} \mathbf{y}_1^{(i+1)} \\ y_\ell^{(i+1)} \\ \widetilde{y}_r^{(i+1)} \end{bmatrix} = \begin{bmatrix} \mathbf{d}_1 \\ d_\ell \\ \widetilde{y}_\ell^{(i)} \end{bmatrix},$$
$$\begin{bmatrix} 0 & I & \mathbf{0} \\ B_{r\ell} & B_{rr} & \mathbf{B}_{r2} \\ \mathbf{B}_{2\ell} & \mathbf{B}_{2r} & \mathbf{B}_{22} \end{bmatrix} \begin{bmatrix} \widetilde{y}_\ell^{(i+1)} \\ y_r^{(i+1)} \\ \mathbf{y}_2{}^{(i+1)} \end{bmatrix} = \begin{bmatrix} \widetilde{y}_r^{(i)} \\ d_r \\ \mathbf{d}_2 \end{bmatrix}. \tag{3.4}$$

These systems can be solved in parallel and we can view this as a simple additive Schwarz iteration (with no overlap and Dirichlet-Dirichlet coupling). The extra unknowns $\widetilde{y}_\ell$ and $\widetilde{y}_r$,

in the enhanced vector $\underset{\sim}{\mathbf{y}}$, will serve for communication between the subdomains during the iterative solution process of the linear system: after each iteration step subdomain 1 and 2 exchange the pairs $(y_\ell, \widetilde{y}_r)$ and $(\widetilde{y}_\ell, y_r)$. After termination of the iterative process, we have to undo the enhancement. We could simply skip the values of the additional elements, but since these carry also information one of the alternatives could be the following one.
With an approximate solution

$$\underset{\sim}{\mathbf{y}}^{(i)} = (\mathbf{y}_1^{(i)\,T}, y_\ell^{(i)\,T}, \widetilde{y}_r^{(i)\,T}, \widetilde{y}_\ell^{(i)\,T}, y_r^{(i)\,T}, \mathbf{y}_2^{(i)\,T})^{\,T}$$

of (3.3), we may associate the approximate solution $\mathbf{R}\underset{\sim}{\mathbf{y}}$ of (3.1) given by

$$\mathbf{R}\underset{\sim}{\mathbf{y}} \equiv (\mathbf{y}_1^{(i)\,T}, \tfrac{1}{2}(y_\ell^{(i)} + \widetilde{y}_\ell^{(i)})^{\,T}, \tfrac{1}{2}(y_r^{(i)} + \widetilde{y}_r^{(i)})^{\,T}, \mathbf{y}_2^{(i)\,T})^{\,T},$$

that is, we simply average the two sets of unknowns that should have been equal to each other at full convergence.

### 3.2.2 Interface coupling matrix

From (3.2) we see that the interface unknowns and the additional interface unknowns are coupled in a straightforward way by

$$\begin{bmatrix} I & 0 \\ 0 & -I \end{bmatrix} \begin{bmatrix} y_\ell \\ \widetilde{y}_r \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & -I \end{bmatrix} \begin{bmatrix} \widetilde{y}_\ell \\ y_r \end{bmatrix}, \tag{3.5}$$

but, of course, we may replace the coupling matrix by any other nonsingular interface coupling matrix $C$:

$$C \equiv \begin{bmatrix} C_{\ell\ell} & C_{\ell r} \\ -C_{r\ell} & -C_{rr} \end{bmatrix}. \tag{3.6}$$

This leads to the following block system

$$\mathbf{B}_C\underset{\sim}{\mathbf{y}} = \begin{bmatrix} \mathbf{B}_{11} & \mathbf{B}_{1\ell} & \mathbf{B}_{1r} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{B}_{\ell 1} & B_{\ell\ell} & B_{\ell r} & 0 & 0 & \mathbf{0} \\ \mathbf{0} & C_{\ell\ell} & C_{\ell r} & -C_{\ell\ell} & -C_{\ell r} & \mathbf{0} \\ \mathbf{0} & -C_{r\ell} & -C_{rr} & C_{r\ell} & C_{rr} & \mathbf{0} \\ \mathbf{0} & 0 & 0 & B_{r\ell} & B_{rr} & \mathbf{B}_{r2} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{B}_{2\ell} & \mathbf{B}_{2r} & \mathbf{B}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{y}_1 \\ y_\ell \\ \widetilde{y}_r \\ \widetilde{y}_\ell \\ y_r \\ \mathbf{y}_2 \end{bmatrix} = \underline{\mathbf{d}}. \tag{3.7}$$

In a domain decomposition context, we will have for the approximate solution $\underset{\sim}{\mathbf{y}}$ that $\widetilde{y}_r \approx y_r$ and $\widetilde{y}_\ell \approx y_\ell$. If we know some analytic properties about the local behavior of the true solution $\mathbf{y}$ across the interface, for instance, smoothness up to some degree, then we may try to identify a convenient coupling matrix $C$ that takes advantage of this knowledge. We want preferably a $C$ so that

$$\begin{aligned} & -C_{\ell\ell}\widetilde{y}_\ell - C_{\ell r}y_r \;\approx\; -C_{\ell\ell}y_\ell - C_{\ell r}\widetilde{y}_r \approx 0 \\ \text{and} \quad & -C_{r\ell}y_\ell - C_{rr}\widetilde{y}_r \;\approx\; -C_{r\ell}\widetilde{y}_\ell - C_{rr}y_r \approx 0. \end{aligned}$$

In this case (3.7) is almost decoupled into two independent smaller linear systems (identified by the two boxes). We may expect fast convergence for the corresponding additive Schwarz iteration.

### 3.2.3 Solution of the coupled subproblems

The goal of the enhancement of the matrix of a given linear system, together with a convenient coupling matrix $C$, is to get two smaller mildly coupled subsystems that can be solved in parallel.

Additive Schwarz for the linear system (3.7) leads to the following iterative scheme

$$\begin{bmatrix} \mathbf{B}_{11} & \mathbf{B}_{1\ell} & \mathbf{B}_{1r} \\ \mathbf{B}_{\ell 1} & B_{\ell\ell} & B_{\ell r} \\ \mathbf{0} & C_{\ell\ell} & C_{\ell r} \end{bmatrix} \begin{bmatrix} \mathbf{y}_1^{(i+1)} \\ y_\ell^{(i+1)} \\ \widetilde{y}_r^{(i+1)} \end{bmatrix} = \begin{bmatrix} \mathbf{d}_1 \\ d_r \\ g_r^{(i)} \end{bmatrix},$$
$$\begin{bmatrix} C_{r\ell} & C_{rr} & \mathbf{0} \\ B_{r\ell} & B_{rr} & \mathbf{B}_{r2} \\ \mathbf{B}_{2\ell} & \mathbf{B}_{2r} & \mathbf{B}_{22} \end{bmatrix} \begin{bmatrix} \widetilde{y}_\ell^{(i+1)} \\ y_r^{(i+1)} \\ \mathbf{y}_2^{(i+1)} \end{bmatrix} = \begin{bmatrix} g_\ell^{(i)} \\ d_\ell \\ \mathbf{d}_2 \end{bmatrix}, \tag{3.8}$$

and

$$g_r^{(i)} = C_{\ell\ell}\, \widetilde{y}_\ell^{(i)} + C_{\ell r}\, y_r^{(i)}, \quad g_\ell^{(i)} = C_{r\ell}\, y_\ell^{(i)} + C_{rr} \widetilde{y}_r^{(i)}. \tag{3.9}$$

The additive Schwarz method can be represented as a block Jacobi iteration method. To see this, consider the matrix splitting $\mathbf{B}_C = \mathbf{M}_C - \mathbf{N}$, where

$$\mathbf{M}_C \equiv \begin{bmatrix} \mathbf{M}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{M}_2 \end{bmatrix},$$

with $\mathbf{M}_1$ the matrix at the top in (3.8) and $\mathbf{M}_2$ the matrix at the bottom ($\mathbf{M}_C$ is the boxed part of $\mathbf{B}_C$ in (3.7), $-\mathbf{N}$ the part outside the boxes). We assume that $C$ *is such that* $\mathbf{M}_C$ *is nonsingular*. The approximate solution $\underset{\sim}{\mathbf{y}}^{(i+1)}$ of (3.7) at step $i+1$ of the block Jacobi method,

$$\underset{\sim}{\mathbf{y}}^{(i+1)} = \underset{\sim}{\mathbf{y}}^{(i)} + \mathbf{M}_C^{-1}\, \underset{\sim}{\mathbf{r}}^{(i)} \quad \text{with} \quad \underset{\sim}{\mathbf{r}}^{(i)} \equiv \underline{\mathbf{d}} - \mathbf{B}_C \underset{\sim}{\mathbf{y}}^{(i)}, \tag{3.10}$$

corresponds to the approximate solutions at step $i+1$ of the additive Schwarz method. In view of the fact that one wants to have $g_r^{(i)}$ and $g_\ell^{(i)}$ as small as possible in norm, the starting value $\underset{\sim}{\mathbf{y}}^{(0)} \equiv \mathbf{0}$ is convenient, but it is conceivable to construct other starting values for which the two vectors are small in norm (for instance, after a restart of some acceleration scheme).

Jacobi is a one step method and the updates from previous steps are discarded. The updates can also be stored in a space $\mathcal{V}_m$ and be used to obtain more accurate approximations. This leads to a subspace method that, at step $m$, searches for the approximate solution in the space $\mathcal{V}_m$, which is precisely equal to the Krylov subspace $\mathcal{K}_m(\mathbf{M}_C^{-1}\mathbf{B}_C, \mathbf{M}_C^{-1}\underline{\mathbf{d}})$. For instance, GMRES [41] finds the approximation in $\mathcal{V}_m$ with the smallest residual, and may be useful if only a few iterations are to be expected.

Krylov subspace methods can be interpreted as accelerators of the domain decomposition method (3.10). The resulting method can also be seen as a preconditioned Krylov subspace method where, in this case, the preconditioner is based on domain decomposition: the matrix $\mathbf{M}_C$. This preconditioning approach where a system of the form $\mathbf{M}_C^{-1}\mathbf{B}_C\underset{\sim}{\mathbf{x}} = \underset{\sim}{\mathbf{r}}^{(0)}$ is solved, is referred to as left preconditioning. Here $\underset{\sim}{\mathbf{r}}^{(0)} \equiv \mathbf{M}_C^{-1}(\underline{\mathbf{d}} - \mathbf{B}_C\underset{\sim}{\mathbf{y}}^{(0)})$ and $\underline{\mathbf{y}} = \underset{\sim}{\mathbf{y}}^{(0)} + \underset{\sim}{\mathbf{x}}$,

Since $\mathbf{M}_C^{-1}\mathbf{B}_C = \mathbf{I} - \mathbf{M}_C^{-1}\mathbf{N}$, the search subspace $\mathcal{V}_m$ coincides with the Krylov subspace
$\mathcal{K}_m(\mathbf{M}_C^{-1}\mathbf{N}, \mathbf{M}_C^{-1}\underline{\mathbf{d}})$. The rank of both $\mathbf{N}$ and $\mathbf{M}_C^{-1}\mathbf{N}$ is equal to the dimension of $C$ which, in this case where $C$ is nonsingular, is $2n_i$. This shows that the dimension of $\mathcal{V}_m$ is at most $2n_i$. Therefore, the exact solution $\underline{\mathbf{y}}$ of (3.7) belongs to $\mathcal{V}_m$ for $m \geq 2n_i$ and GMRES finds $\underline{\mathbf{y}}$ in at most $2n_i$ steps. (For further discussion see, for instance, [7, §3.2], [61, §2], and [6].)

### 3.2.4 Right preconditioning

We can also use $\mathbf{M}_C$ as a right preconditioner. In this case solution $\underline{\mathbf{y}}$ of (3.7) is obtained as $\underline{\mathbf{y}} = \underset{\sim}{\mathbf{y}}^{(0)} + \mathbf{M}_C^{-1}\underset{\sim}{\mathbf{x}}$ where $\underset{\sim}{\mathbf{x}}$ is solved from

$$\mathbf{B}_C\mathbf{M}_C^{-1}\underset{\sim}{\mathbf{x}} = \underset{\sim}{\mathbf{r}}^{(0)} \quad \text{with} \quad \underset{\sim}{\mathbf{r}}^{(0)} \equiv \underline{\mathbf{d}} - \mathbf{B}_C\underset{\sim}{\mathbf{y}}^{(0)}. \tag{3.11}$$

Right preconditioning has some advantages for domain decomposition. To see this, first note that any vector of the form $\mathbf{N}\underset{\sim}{\mathbf{v}}$ '*vanishes outside the artificial boundary*', that is, only the $\tilde{\cdot}_r$ and $\tilde{\cdot}_\ell$ component of this vector are nonzero. Since $\mathbf{B}_C\mathbf{M}_C^{-1} = \mathbf{I} - \mathbf{N}\mathbf{M}_C^{-1}$, multiplication by this operator preserves the property of vanishing outside the artificial boundary. Moreover, if $\underset{\sim}{\mathbf{y}}^{(0)} \equiv \mathbf{M}_C^{-1}\underline{\mathbf{d}}$, then $\underset{\sim}{\mathbf{r}}^{(0)} = \underline{\mathbf{d}} - \mathbf{B}_C\underset{\sim}{\mathbf{y}}^{(0)} = \mathbf{N}\mathbf{M}_C^{-1}\underline{\mathbf{d}}$ vanishes outside the artificial boundary.

Therefore, if, for $\underset{\sim}{\mathbf{y}}^{(0)} \equiv \mathbf{M}_C^{-1}\underline{\mathbf{d}}$, equation (3.11) is solved with a Krylov subspace method with an initial guess that vanishes outside the artificial boundary, for instance $\underset{\sim}{\mathbf{x}}^{(0)} = \mathbf{0}$, then all the intermediate vectors also vanish outside the artificial boundary. Consequently, only vectors of size $2n_i$ have to be stored and the vector updates and dot products are $2n_i$ dimensional operations. Note that, in this way, right preconditioning can be interpreted as a Schur complement method (for our notation see [61], equivalence properties between Schur and Schwarz methods were already reported in [4, 11]).

For appropriate $\underset{\sim}{\mathbf{y}}^{(0)}$, the left preconditioned equation can also be formulated in a $2n_i$ dimensional subspace. However, with respect to the standard basis, it is not so easy to identify the corresponding subspace. We will use the $2n_i$ dimensional subspace, characterized by right preconditioning as corresponding to the artificial boundary, for the derivation of properties of the eigensystem of the iteration matrix.

### 3.2.5 Convergence analysis

As a consequence of (3.10), the errors $\mathbf{e}^{(i)} \equiv \underline{\mathbf{y}} - \underset{\sim}{\mathbf{y}}^{(i)}$ in the block Jacobi method satisfy:

$$\underset{\sim}{\mathbf{e}}^{(i+1)} = (\mathbf{I} - \mathbf{M}_C^{-1}\mathbf{B}_C)\underset{\sim}{\mathbf{e}}^{(i)} = \mathbf{M}_C^{-1}\mathbf{N}\underset{\sim}{\mathbf{e}}^{(i)}. \tag{3.12}$$

Therefore, the convergence rate of the Jacobi iteration depends on the spectral properties of the '*error propagation matrix*' $\mathbf{M}_C^{-1}\mathbf{N}$. These properties also determine the convergence behavior of other Krylov subspace methods. With right preconditioning, we have to work with $\underset{\sim}{\mathbf{x}} - \underset{\sim}{\mathbf{x}}^{(i)}$, which would lead to the error propagation matrix $\mathbf{N}\mathbf{M}_C^{-1}$, but this matrix has the same eigenvalues as the previous one, so we can analyse either of them with the same result.
For the Jacobi iteration, the spectral radius of $\mathbf{M}_C^{-1}\mathbf{N}$ (or of $\mathbf{N}\mathbf{M}_C^{-1}$ in the right preconditioned situation) should be strictly less than 1. For other methods, as GMRES, clustering of the eigenvalues of the error propagation matrix around $0$ is a desirable property for fast convergence.

The kernel of $\mathbf{N}$ forms the space of eigenvectors of $\mathbf{M}_C^{-1}\mathbf{N}$ that are associated with eigenvalue $0$.
Consider an eigenvalue $\sigma \neq 0$ of $\mathbf{M}_C^{-1}\mathbf{N}$ with eigenvector $\underset{\sim}{\mathbf{z}} \equiv (\mathbf{z}_1^T, z_\ell^T, \widetilde{z}_r^T, \widetilde{z}_\ell^T, z_r^T, \mathbf{z}_2^T)^T$:

$$\mathbf{M}_C^{-1}\mathbf{N}\underset{\sim}{\mathbf{z}} = \sigma\underset{\sim}{\mathbf{z}}. \tag{3.13}$$

Since $\mathbf{N}$ maps all components, except for the $\widetilde{\cdot}_\ell$ and $\widetilde{\cdot}_r$ ones, to zero, we have that all components of $\mathbf{M}_C\underset{\sim}{\mathbf{z}}$, except for the $\widetilde{\cdot}_\ell$ and $\widetilde{\cdot}_r$ components, are zero. The eigenvalue problem $\sigma\mathbf{M}_C\underset{\sim}{\mathbf{z}} = \mathbf{N}\underset{\sim}{\mathbf{z}}$ can be decomposed into two coupled problems:

$$\sigma \begin{bmatrix} \mathbf{B}_{11} & \mathbf{B}_{1\ell} & \mathbf{B}_{1r} \\ \mathbf{B}_{\ell 1} & B_{\ell\ell} & B_{\ell r} \\ \mathbf{0} & C_{\ell\ell} & C_{\ell r} \end{bmatrix} \begin{bmatrix} \mathbf{z}_1 \\ z_\ell \\ \widetilde{z}_r \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ 0 \\ g_r \end{bmatrix}, \quad \sigma \begin{bmatrix} C_{r\ell} & C_{rr} & \mathbf{0} \\ B_{r\ell} & B_{rr} & \mathbf{B}_{r2} \\ \mathbf{B}_{2\ell} & \mathbf{B}_{2r} & \mathbf{B}_{22} \end{bmatrix} \begin{bmatrix} \widetilde{z}_\ell \\ z_r \\ \mathbf{z}_2 \end{bmatrix} = \begin{bmatrix} g_\ell \\ 0 \\ \mathbf{0} \end{bmatrix}, \tag{3.14}$$

with

$$g_r \equiv C_{\ell\ell}\,\widetilde{z}_\ell + C_{\ell r}\,z_r, \quad g_\ell \equiv C_{r\ell}\,z_\ell + C_{rr}\widetilde{z}_r. \tag{3.15}$$

In the context of PDEs, the systems in (3.14) can be interpreted as representing homogeneous partial differential equations with inhomogeneous boundary conditions along the artificial boundary: the left system for domain 1, the right system for domain 2. The values $g_r$ and $g_\ell$ at the artificial boundaries are defined by (3.15): the value $g_r$ for domain 1 is determined by the solution of the PDE at domain 2, while the solution of the PDE at domain 1 determines the value at the internal boundary of domain 2.

We have the following properties, which help to identify the relevant part of the eigensystem:

(i) $\mathbf{N}$ is an $n + 2n_i$ by $n + 2n_i$ matrix. Since $C$ is nonsingular, we have that $\text{rank}(\mathbf{N}) = 2n_i$, and it follows that $\dim(\ker(\mathbf{N})) = n$. Hence, $\sigma = 0$ is an eigenvalue with geometric multiplicity $n$.

(ii) Since $\text{rank}(\mathbf{N}) = 2n_i$, there are at most $2n_i$ nonzero eigenvalues $\sigma$, counted according to algebraic multiplicity.

(iii) If $\sigma$ is a nonzero eigenvalue then the corresponding components $g_r$ and $g_\ell$ are nonzero. To see this, take $g_r = 0$. Then from (3.14) we have that $(\mathbf{z}_1^T, z_\ell^T, \widetilde{z}_r^T)^T = 0$. Hence, $g_\ell = 0$, so that $\underset{\sim}{\mathbf{z}}$ would be zero.

(iv) If $\sigma$ is an eigenvalue with corresponding nonzero components $g_r$ and $g_\ell$ then $-\sigma$ is an eigenvalue with eigenvector with components $g_r$ and $-g_\ell$ (use (3.14) and (3.15)).

(v) The vector $\widetilde{\widetilde{z}}_\ell \equiv (z_\ell^T, \widetilde{z}_r^T)^T$ is linearly independent of $\widetilde{\widetilde{z}}_r \equiv (\widetilde{z}_\ell^T, z_r^T)^T$. To prove this, suppose that $\alpha\,\widetilde{\widetilde{z}}_\ell = \beta\,\widetilde{\widetilde{z}}_r$ for some $\alpha, \beta \neq 0$. Then, from (3.14) it follows that $\mathbf{B}\widetilde{\mathbf{z}} = 0$ where

$$\widetilde{\mathbf{z}} \equiv (\alpha\,\mathbf{z}_1^T, \alpha\,z_\ell^T, \alpha\,\widetilde{z}_r^T, \beta\,\mathbf{z}_2^T)^T = (\alpha\,\mathbf{z}_1^T, \beta\,\widetilde{z}_\ell^T, \beta\,z_r^T, \beta\,\mathbf{z}_2^T)^T.$$

As $\mathbf{B}$ is nonsingular, we have $\widetilde{\mathbf{z}} = 0$. Hence, $\underset{\sim}{\mathbf{z}} = \mathbf{0}$ and $\underset{\sim}{\mathbf{z}}$ is not an eigenvector.

Consequently the value of $\sigma$ cannot be equal to $\pm 1$. To prove this, suppose that $\sigma = 1$. Then by combining the last row of the left part and the first row of the right part of (3.14) with (3.15), we find that $C(\widetilde{\widetilde{z}}_\ell - \widetilde{\widetilde{z}}_r) = 0$. Since $C$ is nonsingular, this implies that $\widetilde{\widetilde{z}}_\ell = \widetilde{\widetilde{z}}_r$, i.e. the vectors are linearly dependent. The value $-1$ for $\sigma$ is then excluded on account of property (iv).

The magnitude of $\sigma$ dictates the error reduction. From (3.14) and (3.15) it follows that

$$\begin{aligned} \sigma(C_{\ell\ell}z_\ell + C_{\ell r}\widetilde{z}_r) &= g_r = C_{\ell\ell}\widetilde{z}_\ell + C_{\ell r}z_r \\ \sigma(C_{r\ell}\widetilde{z}_\ell + C_{rr}z_r) &= g_\ell = C_{r\ell}z_\ell + C_{rr}\widetilde{z}_r, \end{aligned} \tag{3.16}$$

which leads to

$$|\sigma|^2 = \frac{(C_{\ell\ell}\widetilde{z}_\ell + C_{\ell r}z_r)^*(C_{r\ell}z_\ell + C_{rr}\widetilde{z}_r)}{(C_{\ell\ell}z_\ell + C_{\ell r}\widetilde{z}_r)^*(C_{r\ell}\widetilde{z}_\ell + C_{rr}z_r)}. \tag{3.17}$$

From (3.16) we conclude that multiplying both $C_{\ell\ell}$ and $C_{\ell r}$ by a nonsingular matrix does not affect the value of $\sigma$. Likewise, both $C_{r\ell}$ and $C_{rr}$ may be multiplied by (another) singular matrix with no effect on $\sigma$. This can be exploited to bring the $C$ matrices to some convenient form.

The one-dimensional case. We first study the one-dimensional case, because this will not only give some insight in how to reduce $\sigma$, but it will also be useful to control local situations in the two-dimensional case.
In this situation the problem simplifies: the matrices $C_{\ell\ell}$, $C_{\ell r}$, $C_{r\ell}$, and $C_{rr}$ are scalars, and so are the vector parts $z_\ell$, $z_r$, $\widetilde{z}_\ell$, and $\widetilde{z}_r$. Because of the freedom to scale the matrices (scalars), we may take $C$ as

$$C = \begin{bmatrix} C_{\ell\ell} & C_{\ell r} \\ -C_{r\ell} & -C_{rr} \end{bmatrix} = \begin{bmatrix} 1 & \alpha_\ell \\ -\alpha_r & -1 \end{bmatrix}. \tag{3.18}$$

With $\mu_\ell \equiv \widetilde{z}_r/z_\ell$, $\mu_r \equiv \widetilde{z}_\ell/z_r$, we have from (3.17) that

$$|\sigma|^2 = \left| \frac{\mu_r + \alpha_\ell}{1 + \alpha_\ell\mu_\ell} \cdot \frac{\alpha_r + \mu_\ell}{\alpha_r\mu_r + 1} \right|. \tag{3.19}$$

The $\mu$-values will be interpreted as local growth factors at the artificial boundary: $\mu_\ell$ shows how $\underset{\sim}{\mathbf{z}}$ changes at the artificial boundary of the left domain; $\mu_r$ shows the same for the right domain.

Note that $\widetilde{\widetilde{z}}_\ell$ depends linearly on $\widetilde{\widetilde{z}}_r$ if $\mu_r\mu_\ell = 1$. Since this situation is excluded on account of property (v), we have that $\mu_r\mu_\ell \neq 1$. The best choice for the minimization of $\sigma$ in (3.19) is obviously $\alpha_\ell = -\mu_r$ and $\alpha_r = -\mu_\ell$, leading to $\sigma = 0$, which gives optimal damping.

The optimal choice for $\alpha_\ell$ and $\alpha_r$ results in a coupling that annihilates the 'outflow' $g_r$ and $g_\ell$ of the two domains. This leads effectively to two uncoupled subdomains: an ideal situation.

More dimensions. In the realistic case of a more dimensional overlap ($n_i > 1$), there is no choice for $\alpha_\ell$ and $\alpha_r$ (i.e., $C_{\ell\ell} = I$, $C_{\ell r} = \alpha_\ell I$, etc.) that leads to an error reduction matrix with only trivial eigenvalues. But, the conclusion that the outflow should be minimized in some average sense for the best error reduction is here also correct. In our application in §3.4, we will identify coupling matrices $C$ that lead to satisfactory clustering of most of the eigenvalues $\sigma$, of the error propagation matrix, around $0$. We will do so by selecting the $\alpha_r$ and $\alpha_\ell$ as suitable averages of the local growth factors $\mu_r$ and $\mu_\ell$.

## 3.3 The eigenvalue problem

### 3.3.1 The Jacobi-Davidson method

For the computation of a solution to an eigenvalue problem the Jacobi-Davidson method [46], is an iterative method that in each iteration:

1. computes an approximation for an eigenpair from a given subspace, using a Rayleigh-Ritz principle,

2. computes a correction for the eigenvector from a so-called correction equation,

3. expands the subspace with the computed correction.

The correction equation mentioned in step 2 is characteristic for the Jacobi-Davidson method, for example, the Arnoldi method [2, 40] simply expands the subspace with the residual for the approximated eigenpair, and the Davidson method [15] expands the subspace with a preconditioned residual. The success of the Jacobi-Davidson method depends on how fast good approximations for the correction equation can be obtained and it is for that purpose that we will try to exploit the enhancement techniques discussed in the previous section.

Therefore, we will consider this correction equation in some more detail. We will do this for the standard eigenvalue problem

$$\mathbf{A}\,\mathbf{x} = \lambda\,\mathbf{x}. \tag{3.20}$$

Given an approximate eigenpair $(\theta, \mathbf{u})$ (with residual $\mathbf{r} \equiv \theta\mathbf{u} - \mathbf{A}\mathbf{u}$) that is close to some wanted eigenpair $(\lambda, \mathbf{x})$, a correction $\mathbf{t}$ for the normalized $\mathbf{u}$ is computed from the correction equation:

$$\mathbf{t} \perp \mathbf{u}, \qquad (\mathbf{I} - \mathbf{u}\mathbf{u}^*)\,(\mathbf{A} - \theta\,\mathbf{I})\,(\mathbf{I} - \mathbf{u}\mathbf{u}^*)\,\mathbf{t} = \mathbf{r}, \tag{3.21}$$

or in augmented formulation ([44, §3.4])

$$\begin{bmatrix} \mathbf{A} - \theta\,\mathbf{I} & \mathbf{u} \\ \mathbf{u}^* & 0 \end{bmatrix} \begin{bmatrix} \mathbf{t} \\ \varepsilon \end{bmatrix} = \begin{bmatrix} \mathbf{r} \\ 0 \end{bmatrix}. \tag{3.22}$$

In many situations it is quite expensive to solve this correction equation accurately and fortunately it is also not always necessary to do so. A common technique is to compute an approximation for $\mathbf{t}$ by a few steps of a preconditioned iterative method, such as GMRES or Bi-CGSTAB.

When a preconditioner $\mathbf{M}$ for $\mathbf{A} - \theta\,\mathbf{I}$ is available, then $(\mathbf{I} - \mathbf{u}\,\mathbf{u}^*)\mathbf{M}(\mathbf{I} - \mathbf{u}\,\mathbf{u}^*)$ can be used as left preconditioner for (3.21). This leads to the linear system (see, [46, §4])

$$\mathbf{P}\,\mathbf{M}^{-1}\,(\mathbf{A} - \theta\,\mathbf{I})\,\mathbf{P}\,\mathbf{t} = \mathbf{P}\,\mathbf{M}^{-1}\,\mathbf{r} \quad \text{where} \quad \mathbf{P} \equiv \mathbf{I} - \frac{\mathbf{M}^{-1}\,\mathbf{u}\,\mathbf{u}^*}{\mathbf{u}^*\,\mathbf{M}^{-1}\,\mathbf{u}}. \tag{3.23}$$

The operator at the left hand side in (3.23) involves two (skew) projectors $\mathbf{P}$. However, when we start the iterative solution process for (3.23) with initial guess $\mathbf{0}$, then $\mathbf{P}\mathbf{t}$ may be replaced with $\mathbf{t}$ at each iteration of a Krylov iteration method: projection at the right can be skipped in each step of the Krylov subspace solver.

Right preconditioning, which has advantages in the domain decomposition approach, can be carried out in a similar way, with similar reductions in the application of $\mathbf{P}$, as we will see in §3.3.3 below. However, because the formulas with right preconditioning look slightly more complicated, we will present our arguments mainly for left preconditioning.

### 3.3.2 Enhancement of the correction equation

We use the domain decomposition approach as presented in §3.2 to solve the correction equation (3.21). Again, we will assume that we have two subdomains and we will use the same notations for the enhanced vectors. With $\mathbf{B} \equiv \mathbf{A} - \theta\,\mathbf{I}$ this leads to the enhanced Jacobi-Davidson correction equation

$$\underline{\mathbf{t}} \perp \underline{\mathbf{u}}, \quad (\mathbf{I} - \underline{\mathbf{u}}\,\underline{\mathbf{u}}^*)\,\mathbf{B}_C\,(\mathbf{I} - \underline{\mathbf{u}}\,\underline{\mathbf{u}}^*)\,\underline{\mathbf{t}} = \underline{\mathbf{r}} \tag{3.24}$$

with $\underline{\mathbf{u}} \equiv (\mathbf{u}_1^T, u_\ell^T, 0^T, 0^T, u_r^T, \mathbf{u}_2^T)^T$, and likewise $\underline{\mathbf{r}} \equiv (\mathbf{r}_1^T, r_\ell^T, 0^T, 0^T, r_r^T, \mathbf{r}_2^T)^T$. The dimension of the zero parts, indicated by $0$, is assumed to be the same as the dimension of $u_\ell$ (and $u_r$).

To see why this is correct, apply the enhancements of §3.2 to the augmented formulation (3.22) of the correction equation, and use the fact that the augmented and the projected form are equivalent. We assume $\mathbf{u}$ to be normalized. Then $\underline{\mathbf{u}}$ is normalized as well.

With

$$(\mathbf{I} - \underline{\mathbf{u}}\,\underline{\mathbf{u}}^*)\mathbf{M}_C(\mathbf{I} - \underline{\mathbf{u}}\,\underline{\mathbf{u}}^*) \tag{3.25}$$

as the left preconditioner, we obtain

$$\mathbf{P}\,\mathbf{M}_C^{-1}\mathbf{B}_C\,\mathbf{P}\,\underline{\mathbf{t}} = \mathbf{P}\,\mathbf{M}_C^{-1}\underline{\mathbf{r}} \quad \text{with} \quad \mathbf{P} \equiv \mathbf{I} - \frac{\mathbf{M}_C^{-1}\underline{\mathbf{u}}\,\underline{\mathbf{u}}^*}{\underline{\mathbf{u}}^*\mathbf{M}_C^{-1}\underline{\mathbf{u}}}. \tag{3.26}$$

In comparison with the error propagation (3.12) of the block Jacobi method for ordinary linear systems, the error propagation matrix $\mathbf{M}_C^{-1}\mathbf{N}$ is now embedded by the projections $\mathbf{P}$. These projections prevent the operator in the correction equation from getting (nearly) singular: as $\theta$ approximates the wanted eigenvalue $\lambda$, in the asymptotic case $\theta$ is even equal to $\lambda$, $\mathbf{B}$ gets close to singular in the direction of the wanted eigenvector $\mathbf{x}$. For ordinary linear systems this possibility is excluded by imposing $\mathbf{B}$ to be nonsingular (see remark (v) in §3.2.5). Here we have to allow a singular $\mathbf{B}$. In our analysis of the propagation matrix of the correction equation, for the model problem in §3.4.3, in first instance we will ignore the projections. Afterwards, we will justify this (both analytically (§3.4.3) as well as numerically (§3.5.2)).

*Note.* We have enhanced the correction equation. Another option is to start with an enhancement of the eigenvalue problem itself. However, this does not result in essential differences (see chapter 5). If the correction equations for these two different approaches are solved exactly, then the approaches are even equivalent.

### 3.3.3 Right preconditioning

In §3.2.4 we have showed that, without projections, right preconditioning for domain decomposition leads to an equation that is defined by its behavior on the artificial boundary only. Although the projections slightly complicate matters, the computations for the projected equation can also be restricted to vectors corresponding to the artificial boundary, as we will see below. Moreover, similar to the situation for left preconditioning, right preconditioning requires only one projection per iteration of a Krylov subspace method. In this section, we will use the underscore notation for vectors in order to emphasize that they are defined in the enhanced space.

First we analyze the action of the right preconditioned matrix.
The inverse on $\underline{\mathbf{u}}^{\perp}$ of the projected preconditioner in (3.25) is equal to (cf. [44, §7.1.1] and [26])

$$\mathbf{P}\mathbf{M}_C^{-1} = \left(\mathbf{I} - \frac{\mathbf{M}_C^{-1}\underline{\mathbf{u}}\,\underline{\mathbf{u}}^*}{\underline{\mathbf{u}}^*\mathbf{M}_C^{-1}\underline{\mathbf{u}}}\right)\mathbf{M}_C^{-1} = \mathbf{M}_C^{-1}\left(\mathbf{I} - \frac{\underline{\mathbf{u}}\,\underline{\mathbf{u}}^*\mathbf{M}_C^{-1}}{\underline{\mathbf{u}}^*\mathbf{M}_C^{-1}\underline{\mathbf{u}}}\right), \tag{3.27}$$

with $\mathbf{P}$ as in (3.26). This expression represents the Moore–Penrose inverse of the operator in (3.25), on the entire space. Note that $\underline{\mathbf{u}}^*\mathbf{P} = \mathbf{0}$ (by definition of $\mathbf{P}$) and $\underline{\mathbf{u}}^*\mathbf{N} = \mathbf{0}$ (by definition of $\underline{\mathbf{u}}$ and $\mathbf{N}$). Therefore, for the operator that is involved in right preconditioning (cf. (3.11)), we have that

$$\begin{aligned}
&(\mathbf{I} - \underline{\mathbf{u}}\,\underline{\mathbf{u}}^*)\mathbf{B}_C(\mathbf{I} - \underline{\mathbf{u}}\,\underline{\mathbf{u}}^*)\mathbf{P}\mathbf{M}_C^{-1} \\
&\qquad = (\mathbf{I} - \underline{\mathbf{u}}\,\underline{\mathbf{u}}^*)\mathbf{B}_C\mathbf{P}\mathbf{M}_C^{-1} \\
&\qquad = (\mathbf{I} - \underline{\mathbf{u}}\,\underline{\mathbf{u}}^*)\mathbf{B}_C\mathbf{M}_C^{-1}\left(\mathbf{I} - \frac{\underline{\mathbf{u}}\,\underline{\mathbf{u}}^*\mathbf{M}_C^{-1}}{\underline{\mathbf{u}}^*\mathbf{M}_C^{-1}\underline{\mathbf{u}}}\right), \\
&\qquad = \mathbf{I} - \underline{\mathbf{u}}\,\underline{\mathbf{u}}^* - (\mathbf{I} - \underline{\mathbf{u}}\,\underline{\mathbf{u}}^*)\mathbf{N}\mathbf{P}\mathbf{M}_C^{-1} \\
&\qquad = \mathbf{I} - \underline{\mathbf{u}}\,\underline{\mathbf{u}}^* - \mathbf{N}\mathbf{P}\mathbf{M}_C^{-1}.
\end{aligned} \tag{3.28}$$

Hence, this operator maps a vector $\underset{\sim}{\mathbf{v}}$ that is orthogonal to $\underline{\mathbf{u}}$ to the vector

$$(\mathbf{I} - \underline{\mathbf{u}}\,\underline{\mathbf{u}}^*)\mathbf{B}_C(\mathbf{I} - \underline{\mathbf{u}}\,\underline{\mathbf{u}}^*)\mathbf{P}\mathbf{M}_C^{-1}\underset{\sim}{\mathbf{v}} = \underset{\sim}{\mathbf{v}} - \mathbf{N}\mathbf{P}\mathbf{M}_C^{-1}\underset{\sim}{\mathbf{v}}$$

that is also orthogonal to $\underline{\mathbf{u}}$.

Therefore, right preconditioning for (3.24) can be carried out in the following steps (cf. §3.2.4):

1. Compute $\quad \underset{\sim}{\mathbf{t}}^{(0)} \equiv \mathbf{P}\mathbf{M}_C^{-1}\underline{\mathbf{r}} \quad$ and $\quad \underset{\sim}{\mathbf{r}}^{(0)} \equiv \mathbf{N}\underset{\sim}{\mathbf{t}}^{(0)}$.
2. Compute an (approximate) solution $\underset{\sim}{\mathbf{s}}^{(m)}$ of

   $$(\mathbf{I} - \mathbf{N}\mathbf{P}\mathbf{M}_C^{-1})\underset{\sim}{\mathbf{s}} = \underset{\sim}{\mathbf{r}}^{(0)},$$

   with ($m$ steps of) a Krylov subspace method with initial guess $\mathbf{0}$.
3. Update $\underset{\sim}{\mathbf{t}}^{(0)}$ to the (approximate) solution $\underline{\mathbf{t}}$ of (3.24):

   $$\underline{\mathbf{t}} = \underset{\sim}{\mathbf{t}}^{(0)} + \mathbf{P}\mathbf{M}_C^{-1}\underset{\sim}{\mathbf{s}}^{(m)}.$$

As in §3.2.4, the intermediate vectors in the solution process for the equation in step 2 vanish outside the artificial boundary. Therefore, for the solution of the right preconditioned enhanced correction equation, only $2n_i$-dimensional vectors have to be stored, and the vector updates and dot products are also for vectors of length $2n_i$.

## 3.4 Tuning of the coupling matrix for a model problem

Now we will address the problem whether it is possible to reduce the computing time for the Jacobi-Davidson process, by an appropriate choice of the coupling matrix $C$. We have, in §3.2, introduced the decomposition of a linear system, into two coupled subsystems, in an algebraic way. In this section we will demonstrate how knowledge of the physical equations from which the linear system originates can be used for tuning of the coupling parameters.

### 3.4.1 The model problem

As a model problem we will consider the two-dimensional advection-diffusion operator:

$$\mathcal{L}(\widehat{\varphi}) \equiv a\frac{\partial^2}{\partial x^2}\,\widehat{\varphi} + b\frac{\partial^2}{\partial y^2}\,\widehat{\varphi} + u\frac{\partial}{\partial x}\,\widehat{\varphi} + v\frac{\partial}{\partial y}\,\widehat{\varphi} + c\widehat{\varphi}, \tag{3.29}$$

that is defined on the open domain $\Omega = (0, \omega_x) \times (0, \omega_y)$ in $\mathbb{R}^2$, with constants $a > 0$, $b \geq 0$, $c, u$ and $v$. We will further assume Dirichlet boundary conditions: $\widehat{\varphi} = 0$ on $\partial\Omega$ of $\Omega$. We are interested in some eigenvalue $\widehat{\lambda} \in \mathbb{C}$ and corresponding eigenfunction $\widehat{\varphi}$ of $\mathcal{L}$:

$$\begin{cases} \mathcal{L}(\widehat{\varphi}) = \widehat{\lambda}\,\widehat{\varphi} & \text{on} \quad \Omega, \\ \widehat{\varphi} = 0 & \text{on} \quad \partial\Omega. \end{cases} \tag{3.30}$$

We will use the insights, obtained with this simple model problem, for the construction of couplings for more complicated partial differential operators.

*Discretization.* We discretize $\mathcal{L}$ with central differences with stepsize $h = (h_x, h_y) = (\frac{\omega_x}{n_x+1}, \frac{\omega_y}{n_y+1})$ for the second order part and stepsize $2h = (2h_x, 2h_y)$ for the first order part, where $n_x$ and $n_y$ are positive integers:

$$\widehat{L}(\widehat{\varphi}) \equiv a\frac{\delta_x^2}{h_x^2}\,\widehat{\varphi} + b\frac{\delta_y^2}{h_y^2}\,\widehat{\varphi} + u\frac{\delta_x}{2h_x}\,\widehat{\varphi} + v\frac{\delta_y}{2h_y}\,\widehat{\varphi} + c\widehat{\varphi}. \tag{3.31}$$

The operator $\dfrac{\delta_x}{h_x}$ denotes the central difference operator, defined as

$$\frac{\delta_x}{h_x}\,\widehat{\psi}(x,y) \equiv \frac{\widehat{\psi}(x+\frac{1}{2}h_x, y) - \widehat{\psi}(x-\frac{1}{2}h_x, y)}{h_x},$$

and $\dfrac{\delta_y}{h_y}$ is defined similar. This leads to the discretized eigenvalue problem

$$\begin{cases} L(\varphi) = \lambda\varphi & \text{on} \quad \Omega_h, \\ \varphi = 0 & \text{on} \quad \partial\Omega_h, \end{cases} \tag{3.32}$$

where $\Omega_h$ and $\partial\Omega_h$ is the uniform rectangular grid of points $(j_x h_x, j_y h_y)$ in $\Omega$ and in $\partial\Omega$, respectively. We have skipped the hat $\widehat{\cdot}$ in order to indicate that the functions are restricted to the appropriate grid, and that the operator $L$ is restricted to grid functions. The vector $\varphi$ is defined on $\Omega_h \cup \partial\Omega_h$.

We use the boundary conditions $\varphi = 0$ at $\partial\Omega_h$ for the elimination of these values of $\varphi$ from $L(\varphi) = \lambda\varphi$.

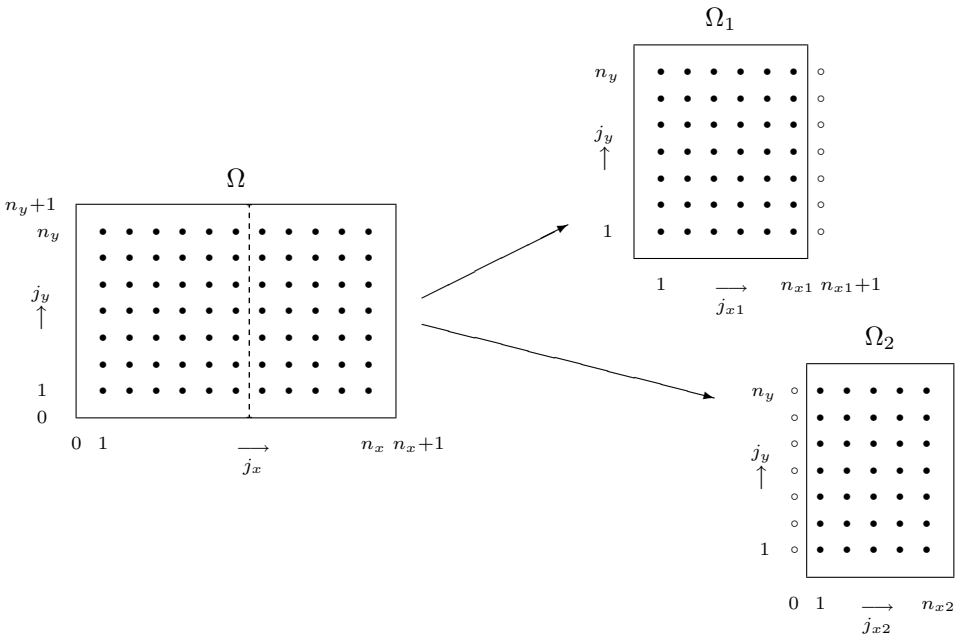
Identification of grid functions with vectors and of operators on grid functions with matrices leads to an eigenvalue problem as in (3.20) of dimension $n \equiv n_x \cdot n_y$: the eigenvector $\mathbf{x}$ corresponds to the eigenfunction $\varphi$ restricted to $\Omega_h$. The matrix $\mathbf{A}$ corresponds to the operator $L$ from which the boundary conditions have been eliminated. In our application, we obtain the corresponding vectors by enumeration of the grid points from bottom to top first (i.e., the $y$-coordinates first) and then from left to right ([58, §6.3]). In our further analysis, we will switch from one representation to another (grid function or vector), selecting the representation that is the most convenient at that moment.

### 3.4.2 Decomposition of the physical domain

For some $0 < \omega_{x1} < \omega_x$ we decompose the domain $\Omega$ in two subdomains $\Omega_1 \equiv (0, \omega_{x1}] \times (0, \omega_y)$ and $\Omega_2 \equiv (\omega_{x1}, \omega_x) \times (0, \omega_y)$.

Let $n_{x1}$ be the number of grid points in the $x$ direction in $\Omega_1$. Then $\Omega_1 \cap \Omega_h$ and $\Omega_2 \cap \Omega_h$ is an $n_{x1} \times n_y$ and $n_{x2} \times n_y$ grid respectively with $n_{x1} + n_{x2} = n_x$. To number the grid points in the $x$ direction, we use local indices $j_{x1}$, $1 \le j_{x1} \le n_{x1}$, and $j_{x2}$, $1 \le j_{x2} \le n_{x2}$, in $\Omega_1$ and $\Omega_2$ respectively.

FIGURE 3.1. *Decomposition of the domain $\Omega$ into two subdomains $\Omega_1$ and $\Omega_2$.*
*The bullets* (•) *represent the grid points of the original grid. The circles* (o) *represent the extra grid points at the internal boundary. The indices $j_x$ and $j_y$ refer to numbering in the $x$ direction and $y$ direction respectively of the grid points in the grids: the pair $(j_x, j_y)$ corresponds to point $(j_x h_x, j_y h_y)$ in $\Omega$. For the numbering of the grid points in the $x$ direction in the two subdomains a local index is used: $j_{x1} = j_x$ in $\Omega_1$ ($0 \leq j_{x1} \leq n_{x1} + 1$) and $j_{x2} = j_x - n_{x1}$ in $\Omega_2$ ($0 \leq j_{x2} \leq n_{x2} + 2$).*




Because of the 5 point star discretization, the unknowns at the last row of grid points ($j_{x1} = n_{x1}$) in the $y$ direction in $\Omega_1$ are coupled with those at the first row of grid points ($j_{x2} = 1$) in the $y$ direction in $\Omega_2$, and vice versa. The unknowns for $j_{x1} = n_{x1}$ are denoted by the vector $y_\ell$, and the unknowns for $j_{x2} = 1$ are denoted by $y_r$, just as in §3.2. Now we enhance the system with the unknowns $\widetilde{y}_r$ and $\widetilde{y}_\ell$, which, in grid terminology, correspond to a virtual new row of gridpoints to the right of $\Omega_1$, and the left of $\Omega_2$, respectively. These new virtual gridpoints serve as boundary points for the domains $\Omega_1$ and $\Omega_2$. See Fig. 3.1 for an illustration.

The vectors $y_\ell$, $y_r$, $\widetilde{y}_\ell$, and $\widetilde{y}_r$ are $n_y$ dimensional (the $n_i$ in §3.2.1 is now equal to $n_y$). The $2n_y$ by $2n_y$ matrix $C$, that couples $y_\ell$, $\widetilde{y}_r$, $\widetilde{y}_\ell$, and $y_r$ can be interpreted as discretized boundary conditions of the differential operator at the internal newly created boundary between $\Omega_1$ and $\Omega_2$ [55, 56].
Note that the internal boundary conditions are explicitly expressed in the total system matrix $\mathbf{B}_C$, through $C$, whereas the external boundary conditions have been used to eliminate the values at the external boundary (see §3.4.1).

### 3.4.3 Eigenvectors of the error propagation matrix

We will now analyze the eigensystem of the error reduction matrix $\mathbf{M}_C^{-1}\mathbf{N}$ (see §3.2.5) and discuss appropriate coupling conditions (that is, the internal boundary conditions) as represented by the matrix $C$. Here, the matrices $\mathbf{M}_C$ and $\mathbf{N}$ are defined for $\mathbf{B} \equiv \mathbf{A} - \theta\mathbf{I}$, as explained in §§3.2.2-3.2.3, for some approximate eigenvalue $\theta$ (cf., §§3.3.1-3.3.2). The matrix $\mathbf{A}$ corresponds to $L$, as explained in §3.4.1.

First, we will discuss in section §3.4.3.1 the case of one spatial dimension (i.e., no $y$ variable). The results for the one-dimensional case are easy to interpret. Moreover, since the two-dimensional eigenvalue problem in (3.30) is a tensor product of two one-dimensional problems, the results for the one-dimensional case can conveniently be used for the analysis in §3.4.3.2 of the two-dimensional problem.

#### 3.4.3.1 The one-dimensional case

In this section, we will discuss the case of one spatial dimension: there is no $y$ variable. To simplify notations, we will skip the index $x$ for this case.

Suppose that we have an approximate eigenvalue $\theta$ for some eigenvalue $\theta$ of $\mathbf{B}$.
To simplify formulas, we shift the approximate eigenvalue by $c$. The matrix $\mathbf{B}$ in §3.2.5 corresponds to the three point stencil of the finite difference operator

$$a\,\frac{\delta^2}{h^2} + u\,\frac{\delta}{2h} - \theta.$$

For the eigensystem of $\mathbf{M}_C^{-1}\mathbf{N}$, we have to solve the systems in (3.14) for an $\widetilde{x}_r \neq 0$ and $\widetilde{x}_\ell \neq 0$, that is, we have to compute solutions $\psi_1$ and $\psi_2$ for the discretized PDE on domain 1 and domain 2, respectively (cf. §3.2.5). The functions $\psi_1$ and $\psi_2$ should satisfy

$$\left[a\,\frac{\delta^2}{h^2} + u\,\frac{\delta}{2h} - \theta\right]\psi_p(j_p h) = 0 \quad \text{for} \quad 1 \leq j_p \leq n_p \quad \text{and} \quad p = 1, 2. \tag{3.33}$$

The conditions on the external boundaries imply that

$$\psi_1(0) = 0 \quad \text{and} \quad \psi_2(n_2 h + h) = 0.$$

For the solutions of (3.33), we try functions of the form $\psi(jh) = \zeta^j$. Then $\zeta$ satisfies

$$\left(1 + \tfrac{uh}{2a}\right)\zeta - 2D + \left(1 - \tfrac{uh}{2a}\right)\zeta^{-1} = 0 \quad \text{with} \quad D \equiv 1 + \frac{h^2}{2a}\theta. \tag{3.34}$$

Let $\zeta_+$ and $\zeta_-$ denote the roots of this equation, such that $|\zeta_+| \geq |\zeta_-|$. In the regular case where $\zeta_+ \neq \zeta_-$, the solutions $\psi_1$ and $\psi_2$ are, apart from scaling, given by

$$\psi_1(j_1 h) = \zeta_+^{j_1} - \zeta_-^{j_1} \quad \text{and} \quad \psi_2(j_2 h) = \zeta_-^{j_2 - n_2 - 1} - \zeta_+^{j_2 - n_2 - 1}.$$

We distinguish three different situations:

(i) *Harmonic behavior:* $\zeta_- = \bar{\zeta}_+ \notin \mathbb{R}$.
If $\zeta_0 \in \mathbb{R}$ and $\tau \in [0, 2\pi)$ are such that $\zeta_+ = \zeta_0 \exp(i\tau)$. Then, up from scaling factors,

$$\psi_1(j_1 h) = \zeta_0^{j_1} \sin(\tau j_1) \quad \text{and} \quad \psi_2(j_2 h) = \zeta_0^{j_2} \sin(\tau(j_2 - n_2 - 1)).$$

(ii) *Degenerated harmonic behavior:* $\zeta_+ = \zeta_-$.
In this case we have, apart from scaling factors,

$$\psi_1(j_1 h) = j_1 \zeta_0^{j_1} \quad \text{and} \quad \psi_2(j_2 h) = (n_2 + 1 - j_2)\zeta_0^{j_2}.$$

(iii) *Dominating behavior:* $|\zeta_+| > |\zeta_-|$.
Near the artificial boundary, that is for $j_1 \approx n_1$ and $j_2 \approx 1$, we have apart from scaling factors that

$$\psi_1(j_1 h) = \zeta_+^{j_1} \left(1 - \left(\frac{\zeta_-}{\zeta_+}\right)^{j_1}\right) \approx \zeta_+^{j_1}$$

and

$$\psi_2(j_2 h) = \zeta_-^{j_2 - n_2 - 1} \left(1 - \left(\frac{\zeta_-}{\zeta_+}\right)^{n_2 + 1 - j_2}\right) \approx c\zeta_-^{j_2},$$

so that, apart from a scaling factor again, $\psi_2(j_2 h) \approx \zeta_-^{j_2}$.
How accurate the approximation is depends on the ratio $|\zeta_-|/|\zeta_+|$ and on the size of $n_1$ and $n_2$.

The coupling matrix $C$ is 2 by 2 ($n_i = 1$). We consider a $C$ as in (3.18). Then, according to (3.19), the absolute value of the eigenvalue $\sigma$ is given by

$$|\sigma|^2 = \left|\frac{\alpha_\ell + \mu_r}{1 + \alpha_r \mu_r}\right| \left|\frac{\alpha_r + \mu_\ell}{1 + \alpha_\ell \mu_\ell}\right|, \tag{3.35}$$

where $\mu_\ell = \psi_1(n_1 h + h)/\psi_1(n_1 h)$ and $\mu_r = \psi_2(0)/\psi_2(h)$: $z_\ell$ in (3.14) corresponds to $\psi_1(n_1 h)$, $\widetilde{z}_r$ to $\psi_1(n_1 h + h)$, etcetera.

In the case of dominating behavior (cf. (iii)), we have that $\mu_\ell \approx \zeta_+$ and $\mu_r \approx 1/\zeta_-$. As observed in (iii), the accuracy of the approximation depends on the ratio $|\zeta_-|/|\zeta_+|$ and on the values of $n_1$ and $n_2$. But already for modest (and realistic) values of these quantities, we obtain useful estimates, and we may expect a good error reduction for the choice $\alpha_\ell = -1/\zeta_-$ and $\alpha_r = -\zeta_+$. The parameters $\zeta_+$ and $\zeta_-$ would also appear in a local mode analysis: they do not depend on the external boundary condition nor on the position of the artificial boundary.

The value for $|\sigma|$ in (3.35) is equal to one when $\mu_r = 1/\bar{\mu}_\ell$, regardless $\alpha_\ell$ and $\alpha_r$ (assuming these are real). If we would follow the local mode approach for the situations (i) and (ii), that is, if we would estimate $\mu_\ell$ by $\zeta_+$ and $\mu_r$ by $1/\zeta_-$, then we would encounter such values for $\mu_\ell$ and $\mu_r$. In specific situations, we may do better by using the expressions for

$\psi_1$ and $\psi_2$ in (i) and (ii), that is, we may find coupling parameters $\alpha_\ell$ and $\alpha_r$ that lead to an eigenvalue $\sigma$ with $|\sigma| < 1$. However, then we need information on the external boundary conditions and the position of the artificial boundary. Certainly in the case of a higher spatial dimension, this is undesirable. Moreover, if $\theta$ is an exact eigenvalue of $\mathbf{A}$ then we are in the situation in (i): the functions $\psi_1$ and $\psi_2$ are multiples of the components on domain 1 and domain 2, respectively, of the eigenfunction and $\sigma = 1$ (see (v) in §3.2.5 and the remark in §3.3.2). In this case there is no value of $\alpha_\ell$ and $\alpha_r$ for which $|\sigma| < 1$.

We define $\nu \equiv (2a+uh)/(2a-uh)$. In order to simplify the forthcoming discussion for two spatial dimensions, observe that, in the case of dominating growth (iii), that is, $\mu_\ell \approx \zeta_+$ and $\mu_r \approx 1/\zeta_-$, (3.35) implies that

$$|\sigma|^2 \approx \left| \frac{\widetilde{\alpha}_\ell + \widetilde{\zeta}}{1 + \widetilde{\alpha}_\ell \widetilde{\zeta}} \right| \left| \frac{\widetilde{\alpha}_r + \widetilde{\zeta}}{1 + \widetilde{\alpha}_r \widetilde{\zeta}} \right|, \text{ where } \widetilde{\alpha}_\ell \equiv \frac{\alpha_\ell}{\sqrt{\nu}},\ \widetilde{\alpha}_r \equiv \sqrt{\nu}\alpha_r,\ \widetilde{\zeta} \equiv \sqrt{\nu}\zeta_+. \tag{3.36}$$

Here we have used that $\zeta_+ \cdot \zeta_- = 1/\nu$, which follows from (3.34).

If, for the Laplace operator (where $u = 0$ and $c = 0$), we use Ritz values for the approximate eigenvalues $\theta$, then $\theta$ takes values between $\lambda^{(n)}$ and $\lambda^{(0)}$. Hence, $\theta \in (-4a/h^2, 0)$, and the roots $\zeta_+$ and $\zeta_-$ are always complex conjugates. We will see in the next subsections that, for two spatial dimensions, the Ritz values that are of interest lead to a dominant root, also for the Laplace operator, and we will see that local mode analysis is then a convenient tool for the identification of effective coupling parameters.

#### 3.4.3.2 Two dimensions

Similar to the one-dimensional case we are interested in functions $\chi_1$ and $\chi_2$ such that,

$$L(\chi_p) = 0 \quad \text{on} \quad \Omega_h \cap \Omega_p, \quad p = 1, 2, \tag{3.37}$$

and that satisfy the external boundary conditions. But now $\chi_1$ and $\chi_2$ are functions that depend on both the $x$- and $y$ direction whereas the operator $L$ (here $L$ is introduced in §3.4.1) acts in these two directions. Since the finite difference operator $\frac{\delta_x}{h_x}$ acts only in the $x$ direction and $\frac{\delta_y}{h_y}$ acts only in the $y$ direction, their actions are independent of each other. Therefore, in this case of constant coefficients[1], we can write the operator $L$ in equation (3.37) as a sum of tensor product of one-dimensional operators:

$$L = L_x \otimes \mathbf{I} + \mathbf{I} \otimes L_y, \tag{3.38}$$

where

$$L_x \equiv a\frac{\delta_x^2}{h_x^2} + u\frac{\delta_x}{2h_x} \quad \text{and} \quad L_y \equiv b\frac{\delta_y^2}{h_y^2} + v\frac{\delta_y}{2h_y} + c - \theta. \tag{3.39}$$

$L_x$ and $L_y$ incorporate the action of $L$ in the $x$ direction and $y$ direction respectively.

[1] It is sufficient if $a$ and $u$ are constants as functions of $y$, $b$ and $v$ are constants as function of $x$, and $c$ is a product of a function in $x$ and a function in $y$.

Since the domain $\Omega$ is rectangular and since on each of the four boundary sides of $\Omega$ we have the same boundary conditions, the tensor product decomposition of $L$ corresponds to a tensor product decomposition of the matrix $\mathbf{A}$.

We try to construct solutions of (3.37) by tensor product functions, that is by functions $\chi_p$ of the form

$$\chi_p(j_{xp}h_x, j_yh_y) = \psi_p(j_{xp}h_x) \otimes \varphi(j_yh_y) = \psi_p(j_{xp}h_x) \cdot \varphi(j_yh_y).$$

For $\varphi$ we select eigenfunctions $\varphi^{(l)}$ of the operator $L_y$ that satisfy the boundary conditions for the $y$ direction. Then

$$L(\chi_p) = (L_x\psi_p) \otimes \varphi^{(l)} + \psi_p \otimes \lambda^{(l)}\varphi^{(l)} = (L_x + \lambda^{(l)})(\psi_p) \otimes \varphi^{(l)},$$

where $\lambda^{(l)}$ is the eigenvalue of $L_y$ that corresponds to $\varphi^{(l)}$. Apparently, for each eigensolution of the '$y$-operator' $L_y$, the problem of finding solutions of (3.37) reduces to a one-dimensional problem as discussed in the previous subsection: find $\psi_p$ such that

$$(L_x + \lambda^{(l)})(\psi_p) = \left[a\frac{\delta_x^2}{h_x^2} + u\frac{\delta_x}{2h_x} + \lambda^{(l)}\right]\psi_p = 0, \tag{3.40}$$

and that satisfy the external boundary conditions in the $x$ direction. To express the dependency of the solutions $\psi_p$ on the selected eigenfunction of $L_y$, we denote the solution as $\psi_p^{(l)}$.

Now, consider matrixpairs $(C_{\ell r}, C_{\ell\ell})$ and $(C_{r\ell}, C_{rr})$ for which the eigenfunctions $\varphi^{(l)}$ of $L_y$ are also eigenfunctions:

$$C_{\ell r}\varphi^{(l)} = \alpha_\ell^{(l)}C_{\ell\ell}\varphi^{(l)} \quad \text{and} \quad C_{r\ell}\varphi^{(l)} = \alpha_r^{(l)}C_{rr}\varphi^{(l)}. \tag{3.41}$$

Examples of such matrices are scalar multiples of the identity matrix (for instance, $C_{\ell r} = \alpha_\ell^{(l)}I$ and $C_{\ell\ell} = I$), but there are others as well, as we will see in §3.4.4. For such a $C$ there is a 1–1 correspondence for each function $\varphi^{(l)}$ on the two subdomains: a component in the direction of $\psi_1^{(l)} \otimes \varphi^{(l)}$ on subdomain 1 is transferred by $\mathbf{M}_C^{-1}\mathbf{N}$ to a component in the direction of $\psi_2^{(l)} \otimes \varphi^{(l)}$ on subdomain 2 and vice versa. More precisely, if $C$ is such that (3.41) holds and if $\psi^{(l)} \equiv (c_l\psi_1^{(l)}, \psi_2^{(l)})^T$ for some scalar $c_l$ then, by construction of $\psi^{(l)}$, $\mathbf{M}_C$ maps $\psi^{(l)} \otimes \varphi^{(l)}$ onto a vector that is zero except for the $\widetilde{\cdot}_\ell$ and $\widetilde{\cdot}_r$ components (cf. (3.14)) which are equal to

$$c_l\left(\psi_1^{(l)}(n_{1x}h_x) + \alpha_\ell^{(l)}\psi_1^{(l)}(n_{1x}h_x + h_x)\right)C_{\ell\ell}\varphi^{(l)} \tag{3.42}$$

and

$$\left(\alpha_r^{(l)}\psi_2^{(l)}(0) + \psi_2^{(l)}(h_x)\right)C_{rr}\varphi^{(l)}, \tag{3.43}$$

respectively. In its turn, $\mathbf{N}$ maps $\psi^{(l)} \otimes \varphi^{(l)}$ onto a vector that is zero except for the $\widetilde{\cdot}_\ell$ and $\widetilde{\cdot}_r$ components (cf. (3.14) and (3.15)) which are equal to

$$\left(\psi_2^{(l)}(0) + \alpha_\ell^{(l)}\psi_2^{(l)}(h_x)\right)C_{\ell\ell}\varphi^{(l)} \tag{3.44}$$

and

$$c_l \left( \alpha_r^{(l)} \psi_1^{(l)}(n_{1x}h_x) + \psi_1^{(l)}(n_{1x}h_x + h_x) \right) C_{rr} \varphi^{(l)}, \tag{3.45}$$

respectively. By a combination of (3.42) and (3.44), and (3.43) and (3.45), respectively, one can check that, for an appropriate scalar $c_l$, $\psi^{(l)} \otimes \varphi^{(l)}$ is an eigenvector of $\mathbf{M}_C^{-1}\mathbf{N}$ with corresponding eigenvalue $\sigma^{(l)}$ such that

$$|\sigma^{(l)}|^2 = \left| \frac{\alpha_\ell^{(l)} + \mu_r^{(l)}}{1 + \alpha_r^{(l)} \mu_r^{(l)}} \right| \left| \frac{\alpha_r^{(l)} + \mu_\ell^{(l)}}{1 + \alpha_\ell^{(l)} \mu_\ell^{(l)}} \right|, \tag{3.46}$$

where (here we assumed that $\psi_1^{(l)}(n_{1x}h_x) \neq 0$ and $\psi_2^{(l)}(h_x) \neq 0$)

$$\mu_\ell^{(l)} \equiv \psi_1^{(l)}(n_{1x}h_x + h_x)/\psi_1^{(l)}(n_{1x}h_x) \quad \text{and} \quad \mu_r^{(l)} \equiv \psi_2^{(l)}(0)/\psi_2^{(l)}(h_x).$$

Note that the expression for $\sigma^{(l)}$ does not involve the value of $c_l$. From property (iv) in §3.2.5 we know that $\psi_-^{(l)} \otimes \varphi^{(l)}$ where $\psi_-^{(l)} \equiv (c_l \psi_1^{(l)}, -\psi_2^{(l)})^T$ is also an eigenvector with eigenvalue $-\sigma^{(l)}$.
As span$\{\psi^{(l)}, \psi_-^{(l)}\}$ = span$\{(\psi_1^{(l)}, 0)^T, (0, \psi_2^{(l)})^T\}$ the functions $\psi^{(l)} \otimes \varphi^{(l)}$ and $\psi_-^{(l)} \otimes \varphi^{(l)}$ are linearly independent and

$$\begin{aligned}
&\text{span}\{\psi^{(1)} \otimes \varphi^{(1)}, \psi_-^{(1)} \otimes \varphi^{(1)}, \ldots, \psi^{(n_y)} \otimes \varphi^{(n_y)}, \psi_-^{(n_y)} \otimes \varphi^{(n_y)}\} = \\
&\text{span}\{\begin{pmatrix} \psi_1^{(1)} \otimes \varphi^{(1)} \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ \psi_2^{(1)} \otimes \varphi^{(1)} \end{pmatrix}, \ldots \\
&\ldots, \begin{pmatrix} \psi_1^{(n_y)} \otimes \varphi^{(n_y)} \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ \psi_2^{(n_y)} \otimes \varphi^{(n_y)} \end{pmatrix}\}.
\end{aligned}$$

From this it follows that the total number of linear independently eigenfunctions of the form $\psi^{(l)} \otimes \varphi^{(l)}$ is equal to $2\,n_y$. Note that our approach with tensorproduct functions leads to the required result: once we know the $n_y$ functions $\varphi^{(1)}, \ldots, \varphi^{(n_y)}$, we can, up to scalars, construct all eigenvectors of $\mathbf{M}_C^{-1}\mathbf{N}$ that correspond to the case (ii) in §3.2.5, i.e. the eigenvectors with, in general, nonzero eigenvalues.[2]

Apparently, the problem of finding the two times $n_y$ nontrivial eigensolutions of $\mathbf{M}_C^{-1}\mathbf{N}$ breaks up into $n_y$ 'one'-dimensional problems. For each $l$, the matrix $\mathbf{M}_C^{-1}\mathbf{N}$ has two eigenvalues $\sigma^{(l)}$ and $-\sigma^{(l)}$ of which the eigenvectors have components that, on domain $p$, correspond to a scalar multiple of $\psi_p^{(l)} \otimes \varphi^{(l)}$ $(p = 1, 2)$.

Errors will be transferred in the iterative solution process of (3.7) from one subdomain to the other. These errors can be decomposed in eigenvectors of $\mathbf{M}_C^{-1}\mathbf{N}$, that is, they can be expressed on subdomain $p$ $(p = 1, 2)$ as linear combination of the functions $\psi_p^{(l)} \otimes \varphi^{(l)}$. The component of the error on domain $p$ in the direction of $\psi_p^{(l)} \otimes \varphi^{(l)}$ is transferred in each

[2] For $\alpha_\ell^{(l)} \to -\mu_r^{(l)}$ or $\alpha_r^{(l)} \to -\mu_\ell^{(l)}$ one of the nonzero eigenvalues degenerates to a defective zero eigenvalue. But then still this construction yields all nonzero eigenvalues. To avoid a technical discussion we give no details here.

step of the iteration process precisely to the component in the direction of $\psi_{3-p}^{(l)} \otimes \varphi^{(l)}$ on domain $3-p$. In case of the block Jacobi method, transference damps this component by a factor $|\sigma^{(l)}|$.

From §3.4.3.1 we know that $\psi_p^{(\ell)}$, the component in the $x$ direction of an eigenvector of $\mathbf{M}_C^{-1}\,\mathbf{N}$, behaves (degenerated) harmonic or dominated. In Fig. 3.2 these typical situations are illustrated.

Here, as in the case of one spatial dimension (§3.4.3.1), the size of the eigenvalues $\sigma^{(l)}$ is determined by the growth factor $\mu_\ell^{(l)}$ of $\psi_1^{(l)}$ and $\mu_r^{(l)}$ of $\psi_2^{(l)}$ in (3.46).
In case of dominated behavior, these factors can adequately be estimated by the dominating root of the appropriate characteristic equation (cf. (3.34)). The scalars, that is, the matrices $C_{\ell r}$ and $C_{r\ell}$ can be tuned to minimize the $|\sigma^{(l)}|$. This will be the subject of our next section. As we explained in §3.4.3.1, we see no practical way to tune our coefficients in case of harmonic behavior. However, in our applications the number of eigenvalues that can not be controlled is limited as we will see in our next subsection. Except for a few eigenvalues, the eigenvalues of the error reduction matrix $\mathbf{M}_C^{-1}\mathbf{N}$ will be small in absolute value: the eigenvalues cluster around 0. If $\theta$ is equal to an eigenvalue $\lambda$ of $\mathbf{A}$, then 1 is an eigenvalue of $\mathbf{M}_C^{-1}\mathbf{N}$ (see (v) in §3.2.5 and §3.3.2) and $\mathbf{M}_C^{-1}\mathbf{B}_C$ is singular. However, the projections that have been discussed in §3.3.2, will remove this singularity. An accurate approximation $\theta$ of $\lambda$ (a desirable situation) corresponds to a near singular matrix $\mathbf{M}_C^{-1}\mathbf{B}_C$, and here, the projection will also improve the conditioning of the matrix.

### 3.4.4 Optimizing the coupling

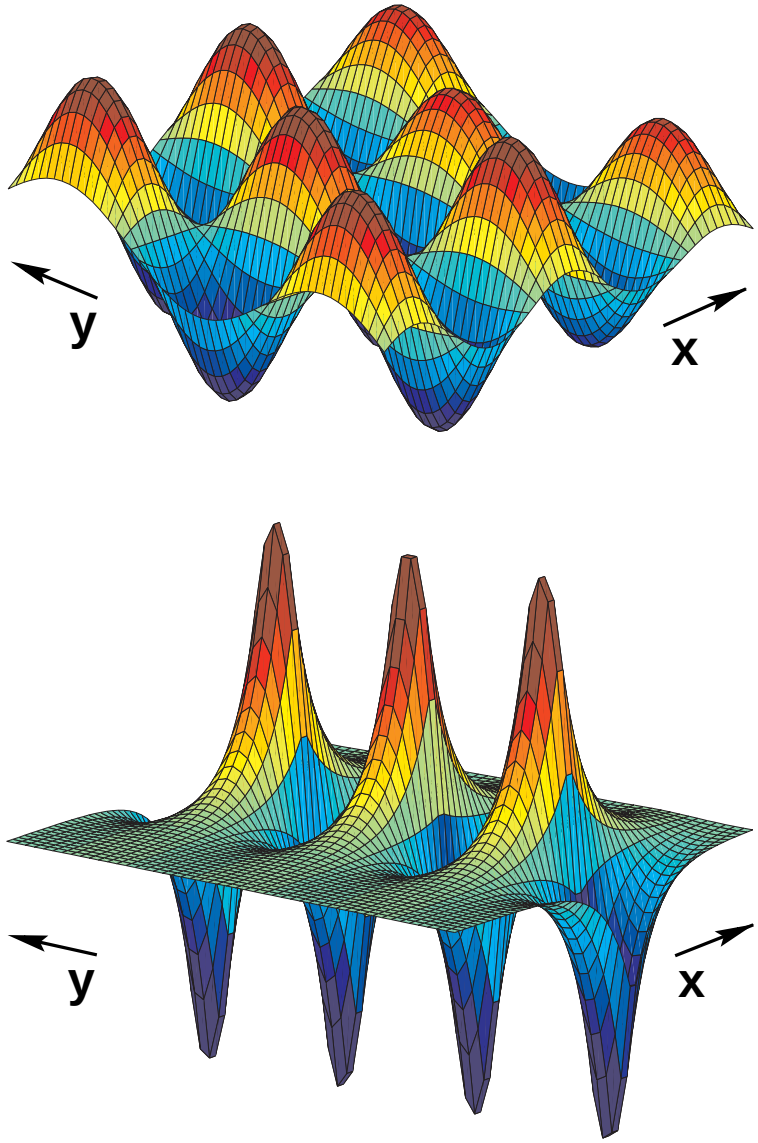
In this section, we will discuss the construction of a coupling matrix $C$ that leads to a clustering of eigenvalues $\sigma^{(l)}$ of $\mathbf{M}_C^{-1}\mathbf{N}$ around 0. We give details for the Laplace operator. We will concentrate on the error modes $\psi_p^{(l)} \otimes \varphi^{(l)}$ on domain $p$ with dominated growth in the $x$ direction, that is, modes for which $\psi_p^{(l)}$ exhibits the dominated behavior as described in (iii) of §3.4.3.1. For these modes and for $C$ as in (3.18) and (3.41), we have that (cf., (3.36) and (3.46))

$$|\sigma^{(l)}|^2 \approx \left|\frac{\widetilde{\alpha}_\ell^{(l)} + \widetilde{\zeta}^{(l)}}{1 + \widetilde{\alpha}_\ell^{(l)}\widetilde{\zeta}^{(l)}}\right| \left|\frac{\widetilde{\alpha}_r^{(l)} + \widetilde{\zeta}^{(l)}}{1 + \widetilde{\alpha}_r^{(l)}\widetilde{\zeta}^{(l)}}\right|. \tag{3.47}$$

Here, for $\nu \equiv (2a + uh_x)/(2a - uh_x)$, the quantities $\widetilde{\alpha}_\ell^{(l)}$, $\widetilde{\alpha}_r^{(l)}$ and $\widetilde{\zeta}^{(l)}$ are defined as in (3.36): $\widetilde{\alpha}_\ell^{(l)} \equiv \alpha_\ell^{(l)}/\sqrt{\nu}$, $\widetilde{\alpha}_r^{(l)} \equiv \sqrt{\nu}\,\alpha_r^{(l)}$, $\widetilde{\zeta}^{(l)} \equiv \sqrt{\nu}\,\zeta_+^{(l)}$, where here $\zeta_+^{(l)}$ is the dominant root of (3.34) for $\lambda' = \lambda^{(l)}$. Note that, in view of the symmetry in the expression for $|\sigma^{(l)}|^2$, it suffices to study a $C$ for which $\widetilde{\alpha}_\ell^{(l)} = \widetilde{\alpha}_r^{(l)}$.

Let $E$ be the set of $l$'s in $\{1, \ldots, n_y\}$ for which the $\psi_p^{(l)}$ exhibit dominated growth, or, equivalently, for which the characteristic equation associated with the operator $L_x + \lambda^{(l)}$ in (3.40) (cf., (3.34)) has a dominant root $\zeta_+^{(l)}$: $E \equiv \{l = 1, \ldots, n_y \mid |\zeta_+^{(l)}| > |\zeta_-^{(l)}|\}$. We are

FIGURE 3.2. *Eigenvector of the error propagation matrix* $\mathbf{M}_C^{-1}\,\mathbf{N}$ *for the two-dimensional two subdomain case. In the* $x$ *direction, the direction perpendicular to the interface between the subdomains, it typically behaves harmonic (top picture) or dominated (bottom picture). For explanation see* §*3.4.3.2.*



interested in $\alpha^{(l)} \equiv \widetilde{\alpha}_\ell^{(l)} = \widetilde{\alpha}_r^{(l)}$ for which

$$\sigma_{\text{opt}} \equiv \max\left\{ \left| \frac{\alpha^{(l)} + \zeta}{1 + \alpha^{(l)}\zeta} \right| \,\middle|\, \zeta \in \widehat{E} \right\} \quad \text{with} \quad \widehat{E} \equiv \{\sqrt{\nu}\,\zeta_+^{(l)} \mid l \in E\} \tag{3.48}$$

is 'as small as possible'.

#### 3.4.4.1 Simple coupling

For the choice $C_{\ell r} = \sqrt{\nu}\,\alpha I$ and $C_{r\ell} = (\alpha/\sqrt{\nu})I$, we can easily analyze the situation. Then $\alpha^{(l)} = \alpha$ for all $l$ and we should find the $\alpha = \alpha_{\text{opt}}$ that minimizes $\max|(\alpha+\zeta)/(1+\alpha\zeta)|$. We assume that $|uh_x| < 2a$. Note that then $\sqrt{\nu}$ times the dominant characteristic roots are real and $> 1$. Therefore, the two extremal values

$$\mu \equiv \min \widehat{E} \quad \text{and} \quad M \equiv \max \widehat{E} \tag{3.49}$$

determine the size of the maximum. This leads to

$$-\alpha_{\text{opt}} = 1 + \frac{\sqrt{(\mu^2-1)(M^2-1)}}{\mu + M} + \frac{(\mu-1)(M-1)}{\mu+M} > 1 \tag{3.50}$$

and

$$\sigma_{\text{opt}} = \frac{\sqrt{M^2-1} - \sqrt{\mu^2-1}}{M\sqrt{\mu^2-1} + \mu\sqrt{M^2-1}} > 0. \tag{3.51}$$

*Laplace operator.* To get a feeling for what we can expect, we interpret and discuss the results for the Laplace operator, that is, we now take $u = v = c = 0$. Further, we concentrate on the computation of (one of) the largest eigenvalue of $L$ and we assume that $\theta$ is close to the target eigenvalue. Then

$$\lambda^{(l)} = -\frac{2b}{h_y^2}\left(1 - \cos\left(\pi \frac{l}{n_y+1}\right)\right) - \theta. \tag{3.52}$$

First we derive a lower bound for $\mu$ and an upper bound for $M$.
For $D^{(l)} \equiv 1 - \frac{h_x^2}{2a}\lambda^{(l)}$ (cf., (3.34)), we have that $|D^{(l)}| > 1$, or, equivalently, $|\zeta_+^{(l)}| > |\zeta_-^{(l)}|$, if and only if $\lambda^{(l)} < 0$. Hence $l_e \equiv \min E$ is the smallest integer $l$ for which $\lambda^{(l)} < 0$ and

$$l_e \equiv \lfloor \widetilde{l}_e \rfloor + 1 \quad \text{where} \quad \widetilde{l}_e \equiv \frac{2}{\pi}(n_y+1)\arcsin\left(\frac{h_y}{2}\sqrt{\frac{-\theta}{b}}\right).$$

(The noninteger value $l = \widetilde{l}_e$ is the 'solution' of $\lambda^{(l)} = 0$.) For $h_y \ll 1$, $\widetilde{l}_e \approx \dfrac{\omega_y}{\pi}\sqrt{\dfrac{-\theta}{b}}$.

For an impression on the error reduction that can be achieved with a suitable coupling, we are interested in lower bounds for $\mu - 1$ that are as large as possible. With $\delta \equiv D^{(l_e)} - 1$

we have that $\mu - 1 = \delta + \sqrt{2\delta + \delta^2} \geq \sqrt{2\delta}$. Therefore, we are interested in positive lower bounds for $\delta$:

$$\begin{aligned} \delta &= \rho^2(\cos(\pi\frac{\widetilde{l}_e}{n_y+1}) - \cos(\pi\frac{l_e}{n_y+1})) \geq \pi\rho^2\frac{l_e-\widetilde{l}_e}{n_y+1}\sin(\pi\frac{\widetilde{l}_e}{n_y+1}) \\ &\geq 2\pi\frac{b}{a}\widetilde{l}_e(l_e-\widetilde{l}_e)\left(\frac{h_x}{\omega_y}\right)^2 \qquad \text{where} \quad \rho \equiv \frac{h_x}{h_y}\sqrt{\frac{b}{a}}. \end{aligned}$$

The bound for $\delta$ depends on the distance of $\widetilde{l}_e$ to the integers, which can be arbitrarily small. This means that, even for the optimal coupling parameters, the (absolute value of the) eigenvalue $\sigma^{(l_e)}$ can be arbitrarily close to one. Since, for optimal coupling, the damping that we achieve for the smallest $l$ in $E$ is the same as for the largest, it seems to be undesirable to concentrate on damping the error modes associated with $l_e$ as much as possible. Therefore, we remove $l_e$ from the set $E$ and concentrate on damping the error modes associated with $l$ in $E' \equiv E\backslash\{l_e\}$. For the $\delta$ and $\mu$ associated with this slightly reduced set $E'$ we have that

$$\mu - 1 \geq \sqrt{2\delta} \geq 2\kappa h_x \quad \text{where} \quad \kappa \equiv \frac{1}{\omega_y}\sqrt{\pi l_e\frac{b}{a}}. \tag{3.53}$$

The lower bound for $\mu - 1$ is sharp for $h \to 0$ with $\rho$ fixed, i.e., for given $\rho$, $h = (h_x, h_y)$ is such that $h_x = h_y\rho\sqrt{a/b}$.

An upper bound for $M$ follows from the observations that $\theta < 0$ and that the cosine takes values between $-1$ and 1: we have that $D^{(l)} \leq 1 + 2\rho^2$ and

$$M - 1 \leq 2\rho^2 + \sqrt{4\rho^2 + 4\rho^4}.$$

Put

$$M' \equiv \sqrt{\frac{M-1}{M+1}} \leq \sqrt[4]{\frac{\rho^2}{1+\rho^2}}.$$

Then, for $h \to (0,0)$ such that $\rho$ is fixed, we have that

$$-\alpha_{\text{opt}} = 1 + 2M'\sqrt{\kappa h_x} + \mathcal{O}(h_x) \quad \text{and} \quad 1 - \sigma_{\text{opt}} = 2\frac{\sqrt{\kappa h_x}}{M'} + \mathcal{O}(h_x).$$

Here we used the fact that

$$-\alpha_{\text{opt}} = 1 + \sqrt{2(\mu-1)}M' + \mathcal{O}(\mu-1) \quad \text{and} \quad 1 - \sigma_{\text{opt}} = \sqrt{2(\mu-1)}/M' + \mathcal{O}(\mu-1)$$

for $\mu \to 1$ (see (3.50) and (3.51)).

So, for small stepsizes $h$, the 'best' 'asymptotic error reduction factor' $\sigma_{\text{opt}}$ is less than one with a difference from one that is proportional to the square root of $h_y$.

#### 3.4.4.2 Control of nondominant modes: deflation

We tried to cluster the eigenvalues of $\mathbf{M}_C^{-1}\mathbf{B}$ around one as much as possible. With $\alpha = \alpha_{\text{opt}}$, at most $l_e$ eigenvalues may be located outside the disk with radius $\sigma_{\text{opt}}$ and center one. After an initial $l_e$ steps we may expect the convergence of GMRES to be determinated by $\sigma_{\text{opt}}$ (provided that the basis of eigenvectors is not too skew). Therefore, as long as $l_e$ is a modest integer, we expect GMRES to converge well in this situation. We will now argue that, in realistic situations, $l_e$ will be modest as compared to the index of the eigenvalue of $\mathbf{A}$ in which we are interested. For clearness of arguments, we assume the stepsizes to be small: $h \to (0,0)$ with $\rho$ fixed: $\lambda^{(l_e)} \approx -b\pi^2(l_e/\omega_y)^2 - \theta$.

Suppose that, for some $\tau > 0$, we are interested in the smallest eigenvalue $\lambda$ of $\mathbf{A}$ that is larger than $-\tau$. Since, in the Jacobi-Davidson process, $\theta$ converges to $\lambda$, $\theta$ will eventually be larger than $-\tau$. We concentrate on this 'asymptotic' situation.[3]
Then, $l_e \leq C_1(\tau') + 1$, where

$$C_1(\tau') \equiv \#\{l \in \mathbb{N} \mid l^2 \leq \tau'\} \quad \text{and} \quad \tau' \equiv \tau \frac{\omega_y^2}{b\pi^2}.$$

The number of eigenvalues $\lambda^{(m_x,m_y)} \approx -a\pi^2(m_x/\omega_x)^2 - b\pi^2(m_y/\omega_y)^2$ of $\mathbf{A}$ that are larger than $-\tau$ is approximately equal to

$$C_2(\tau') \equiv \#\{(m_x,m_y) \in \mathbb{N}^2 \mid \; m_y^2 + \frac{a}{b}\frac{\omega_y^2}{\omega_x^2} m_x^2 \leq \tau'\}.$$

Since $C_1(\tau')^2 \lesssim 2\frac{\omega_y}{\omega_x}\sqrt{\frac{a}{b}}\, C_2(\tau')$, the number $l_e + 1$ of error modes that we do not try to control with appropriate coupling coefficients is proportional to the *square root* of the index number of the wanted eigenvalue (if the eigenvalues have been increasingly ordered). For instance, if $a = b$, $\omega_x = \omega_y$, and $\tau' = 15$, then eight eigenvalues of $\mathbf{A}$ are larger than $-\tau$, and we do not 'control' four modes. One of these modes corresponds with the wanted eigenvalue and is 'controlled' by the projections in the correction equation of the Jacobi-Davidson process.

In practice, deflation will be used for the computation of the, say, eight eigenvalue of $\mathbf{A}$. The first seven eigenvalues will be computed first and will be deflated from $\mathbf{A}$. In such an approach, the three modes that we did not try to control in our coupling, will be controlled by the projection on the space orthogonal to the detected eigenvectors. See §3.5.2.2 for a numerical example.

We analyzed the situation where the domain has been decomposed into two subdomains. Of course, in practice, we will interested in a decomposition of more subdomains. In these situations, the number of modes that we did not try to control by the coupling, will be proportional to the number of artificial boundaries. For numerical results, see §3.5.4. Deflation

[3] The Jacobi-Davidson process can often be started in practice with an approximate eigenvector that is already close to the wanted eigenvector. Then $\theta$ will be close to $\lambda$. For instance, if one is interested in a number of eigenvalues close to some target value, then the search for the second and following eigenvectors will be started with a search subspace that has been constructed for the first eigenvector. This search subspace will be 'rich' with components in the direction of the eigenvectors that are wanted next (see [26, §3.4]).

will be more important if the number subdomains is larger. Note that the observations in the §§3.4.3.1 and 3.4.3.2 on the error modes that exhibit dominated behavior also apply to the situation of more than two subdomains: the essential observation in case of dominated growth is that, on one subdomain, the influence of the 'dominated' component (as represented by $\zeta_-^{(l)}$) is negligible at the artificial boundary regardless the boundary condition at the other end of the subdomain.

#### 3.4.4.3 Stronger couplings

In §3.4.3.2, we considered coupling matrices $C$ with eigenvectors related to ones of $L_y$, the $y$-component of the finite difference operator $L$. Instances of such matrices can easily be formed by using $L_y$ itself.

For ease of notation we consider the Laplace operator. Inclusion of first order terms only results in extra factors $\nu$ (cf. (3.36) in §3.4.3.1). Consider the matrices

$$C_{\ell\ell} = C_{rr} = 1 + \gamma L_y \quad \text{and} \quad C_{\ell r} = C_{r\ell} = \alpha + \beta L_y, \tag{3.54}$$

where $\alpha$, $\beta$, and $\gamma$ are appropriate scalars. With $\beta$ and $\gamma$, we introduce interaction parallel to the interface in the coupling. Then $\alpha_\ell^{(l)}$ in (3.41) is equal to

$$\alpha_\ell^{(l)} = q_\ell(\lambda^{(l)}) \quad \text{where} \quad q_\ell(\lambda) \equiv \frac{\alpha + \beta\lambda}{1 + \gamma\lambda}. \tag{3.55}$$

Note that the dominant root $\zeta_+^{(l)}$ (cf. (3.34) with $\lambda' = \lambda^{(l)}$) depends on $\lambda^{(l)}$: $\zeta_+^{(l)} = w_\ell(\lambda^{(l)})$ for some function $w_\ell$. Hence, we are interested in finding scalars $\alpha$, $\beta$, and $\gamma$ for which

$$\sigma'_{\text{opt}} \equiv \max_\lambda \left| \frac{q_\ell(\lambda) + w_\ell(\lambda)}{1 + q_\ell(\lambda) w_\ell(\lambda)} \right| \tag{3.56}$$

is as small as possible. Here $\lambda$ ranges over the set of eigenvalues $\lambda^{(l)}$ of $L_y$ that lead to a dominant root $\zeta_+^{(l)} = w_\ell(\lambda^{(l)})$. For $\beta = \gamma = 0$ we have the 'simple coupling' as discussed above. For the coupling at the right side of the artificial boundary, we have similar expressions. Finding the minimum of (3.56) is a non-linear problem (in $\alpha$, $\beta$ and $\gamma$; $q_\ell$ is rational and $q_\ell$ is in the denominator) and can not analytically be solved. But a numerical solution can be obtained with, for instance, a modified Rémès algorithm. We discuss our results for a simple example in order to illustrate how much can be gained by including interactions parallel to the artificial boundary in the coupling.

*Example.* Table 3.1 shows values for $\sigma'_{\text{opt}}$ for the Laplace operator on the unit square ($a = b = 1$, $u = v = c = 0$, $\Omega = (0,1) \times (0,1)$), with $\theta = -34\pi^2$ (then $l_e = 6$ and 24 eigenvalues are larger than $\theta$), $n_x = 180$, $n_y = 120$ and $\omega_{x1} = \frac{1}{3}$. In case 1 in the table, we took $\beta = \gamma = 0$ and we optimized with respect to $\alpha$. This case corresponds to the 'simple coupling' as discussed above. We learn from column 2 of Table 3.1 that an additional parameter $\beta$ allows a considerable reduction of the damping factor.

TABLE 3.1. *The table shows the values that can be achieved for the damping $\sigma'_{opt}$ in* (3.56) *for the Laplace equation on the the unit square by optimizing the coupling in* (3.54) *with respect to some of the parameters $\alpha$, $\beta$ and $\gamma$. For explanation see the example in §3.4.4.*

| | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| optimized w.r.t. | $\alpha$ | $\alpha$, $\beta$ | $\alpha$, $\gamma$ | $\alpha$, $\beta$, $\gamma$ |
| $\sigma'_{\text{opt}}$ | 0.696 | 0.157 | 0.376 | 0.093 |

With $\beta = \gamma = 0$ the explicit coupling is in the $x$ direction only, this corresponds to a two point stencil for the boundary conditions on the artificial boundary. The parameter $\beta$ introduces a coupling in the $y$ directions which corresponds to a four point stencil for the artificial boundary conditions. If in addition $\gamma \neq 0$, the coupling corresponds to a six point stencil. Extension from a two to a four point stencil appears to be more effective than the extension from a four to a six point stencil (a reduction of $\sigma'_{\text{opt}}$ from $0.696$ to $0.157$ as compared to a reduction from $0.157$ to $0.093$ in Table 3.1). The parameter $\beta \neq 0$ gives a coupling of the internal boundary conditions on the artificial interface (the $\circ$'s in Fig. 3.1), while $\gamma$ gives a coupling of the internal boundary conditions on points of the original domain (the $\bullet$'s in Fig. 3.1 closest to the cut). Note that an optimal $\beta$ (with $\gamma = 0$) gives better values than an optimal $\gamma$ (with $\beta = 0$).

Experimentally we verified that the values for $\sigma'_{\text{opt}}$ obtained with a 'local mode analysis' (where we neglected $\zeta_-$ terms) correspond rather well with the actual radius of the cluster of eigenvalues of $\mathbf{M}_C^{-1}\mathbf{N}$: except for the first $l_e + 1$ eigenvalues, in all cases all eigenvalues of $\mathbf{M}_C^{-1}\mathbf{N}$ are in the disc with center 0 and radius $\sigma'_{\text{opt}}$. Since we did not optimize for the first $l_e$ eigenvalues, it is no surprise that these eigenvalues are not in the disc. The $l_e+1$th eigenvalue corresponds to the situation where $|\zeta_+^{(l)}|$ is closest to $|\zeta_-^{(l)}|$ and then the predictions of the local mode analysis may expected to be the least reliable. For an experiment with larger stepsize see §3.5.2.3.

## 3.5 Numerical experiments

The experiments presented in this section illustrate the numerical behavior of the Jacobi-Davidson method in combination with the domain decomposition method, as described in §3.3 and §3.4. We will focus on some characteristic properties. All experiments are performed with MATLAB 5.3.0 on a Sun Sparc Ultra 5 workstation.

In §3.5.1 we will discuss the circumstances under which experiments have been performed. Because Jacobi-Davidson is a nested iterative method, an inexact solution of the correction equation affects the outerloop. Therefore, we will also check how the exact process behaves and which stage of the process is most sensitive to inexact solution.

Then, in §3.5.2, we consider the spectrum of the error propagator for the asymptotic situation $\theta = \lambda$. This spectrum contains all information for understanding the convergence behavior of the Jacobi iteration method. The predictions of §3.4.4 on the optimized coupling are verified and we investigate the effect of deflation.

TABLE 3.2. *Convergence of Jacobi-Davidson, with accurate solution of the correction equation, towards the eigenvalue of smallest absolute value (=largest eigenmode) of the discretized ($n = 99, h = 0.01$) eigenvalue problem for the one-dimensional Laplace operator.*

| step | selected Ritz value | residual selected Ritz pair | number of correct digits selected Ritz value |
|---|---|---|---|
| 1 | -3992.4322622 | 9.74e+03 | -3.6 |
| 2 | -1487.8343933 | 3.99e+03 | -3.2 |
| 3 | -581.73159839 | 1.62e+03 | -2.8 |
| 4 | -283.84104294 | 7.22e+02 | -2.4 |
| 5 | -123.01979659 | 3.23e+02 | -2.1 |
| 6 | -42.762088608 | 1.15e+02 | -1.5 |
| 7 | -17.253205686 | 4.49e+01 | -0.87 |
| 8 | -9.8982441731 | 7.41e+00 | 1.5 |
| 9 | -9.8687926855 | 5.15e-04 | 9.8 |
| 10 | -9.8687926854 | 6.26e-12 | 12 |

The next question is how the Jacobi-Davidson method behaves when inexact solutions for the correction equation are obtained with Jacobi iterations. In §3.5.3 we compare different types of coupling, and left and right preconditioning. Furthermore, we consider GMRES as an accelerator of the Jacobi iterative method.

We conclude, in §3.5.4, with an experiment that shows what happens when we have more than two subdomains.

### 3.5.1 Reference process

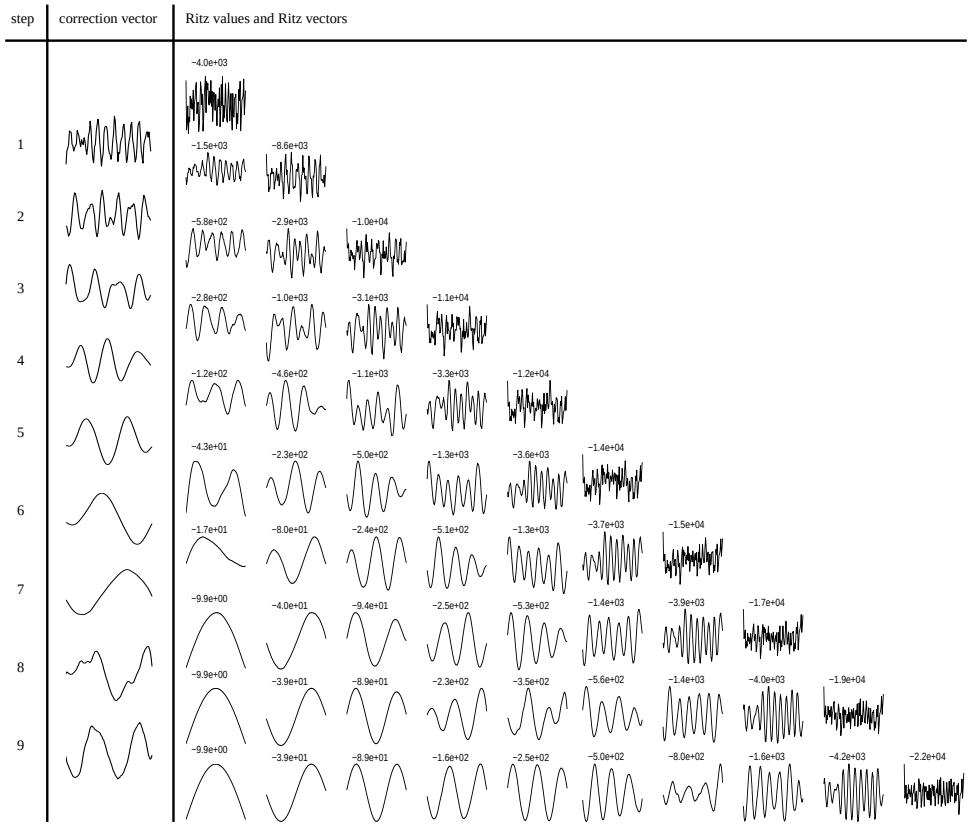
We first consider the standard Jacobi-Davidson method, when applied to the discretized eigenvalue problem for the Laplace operator. No domain is decomposed and correction vectors are obtained by accurate solution of the correction equation.

The first experiment gives a global impression of the speed of convergence. For that purpose we confine ourselves to the one-dimensional case, described in §3.4.3.1. We take $n = 99, h = 0.01$. For the starting vector of the Jacobi-Davidson process, we take a random vector generated in MATLAB (with seed equal to 226). We want to compute the eigenvalue of smallest absolute value ($\lambda_1 = -\left(200 \sin \frac{\pi}{200}\right)^2 = -9.86879268536\ldots$). The corresponding eigenvector describes the largest eigenmode of the discretized PDE.

Table 3.2 and Fig. 3.3 show what happens in the iteration process. The second column of Table 3.2 gives the selected Ritz value $\theta$ for the correction equation, the third column gives the 2-norm of the residual $\mathbf{r} \equiv \mathbf{A}\mathbf{u} - \theta\mathbf{u}$ of the corresponding Ritz pair $(\theta, \mathbf{u})$, and the fourth column lists the number of correct digits of the Ritz value: $-\log^{10} |\lambda - \theta|$.

From Table 3.2 we observe that Jacobi-Davidson needs about 8 steps before the (theoretically cubic) convergence to the desired eigenvalue sets in. This might have been expected: as

FIGURE 3.3. *Convergence behavior of Jacobi-Davidson with accurate solution of the correction equation, when applied to the discretized ($n = 99, h = 0.01$) eigenvalue problem for the one-dimensional Laplace operator. The process is started with one random vector. In each step a correction vector is computed (second column) by which the search subspace is expanded. In the third column all Ritz values of the search subspaces before/after expansion are printed. Right below this number the corresponding Ritz vector is graphically displayed.*



the startvector is random it is likely that the components of all eigenmodes are about equally represented in the startvector. Therefore, in the beginning the eigenvalues with larger absolute value will dominate for a while. In Fig. 3.3 we display the Ritz vectors after each iteration of the Jacobi-Davidson process. The corresponding eigenmodes are of high frequency, which explains the order of appearance of Ritz vectors (high frequencies dominate initially).

A proper target value in the correction equation (3.21), instead of the Ritz value, may help to overcome the initial phase of slow convergence, but this is beyond the scope of this chapter. Our concern is the question how much the process is affected when the correction equation is solved approximately by performing accurate solves on the subdomains only and by tuning the interface conditions. A less accurate solution of the correction equation will, in general, result in more steps of Jacobi-Davidson (outer iterations) for the same precision for the approximate eigenpair. In particular, we do not want to extend the 'slow phase' by destroying the 'fast phase' with too inaccurate solution steps. We take the 'exact' Jacobi-Davidson process in Table 3.2 as our reference. In order to see what happens in the final, potentially fast phase, we select a parabola shaped startvector.

TABLE 3.3. *Convergence history of Jacobi-Davidson applied to the discretized eigenvalue problem of the two-dimensional Laplace operator ($n_x = 63, n_y = 31, \omega_x = 2$ and $\omega_y = 1$) with accurate solutions of the correction equation.*

| step | $\theta$ | $\theta - \lambda$ | $\|\mathbf{r}\|_2$ | $\|\mathbf{r}'\|_2$ |
|---|---|---|---|---|
| 1 | -12.4896 | -1.61e-01 | 4.19e+00 | 4.19e+00 |
| 2 | -12.3286 | -9.65e-07 | 8.55e-03 | 6.10e-03 |
| 3 | -12.3286 | -1.55e-13 | 1.76e-10 | 1.19e-10 |
| 4 | -12.3286 | -1.33e-13 | 7.71e-14 | 3.90e-14 |

In the next subsections we will mainly consider the more interesting two-dimensional case, with physical sizes $\omega_x = 2$ and $\omega_y = 1$. The number of grid points in $x$- and $y$ direction are $n_x = 63$ and $n_y = 31$, so $h_x : h_y = 1 : 1$. The eigenvalue corresponding to the largest eigenmode of the discretized Laplace operator is equal to $-12.328585\ldots$. In Table 3.3 the convergence history for Jacobi-Davidson to this eigenpair is presented when starting with the parabolic vector

$$\{(\frac{j_x}{n_x+1}(1-\frac{j_x}{n_x+1}), \frac{j_y}{n_y+1}(1-\frac{j_y}{n_y+1})) \;\mid\; 1 \le j_x \le n_x, 1 \le j_y \le n_y\}, \quad (3.57)$$

and with accurate solutions of the correction equation. The second column of this table shows the selected Ritz value for the correction equation, the third column the error $\theta - \lambda$ for this Ritz value, and the fourth column gives the 2-norm of the residual $\mathbf{r}$ for the corresponding normalized Ritz pair. Jia and Stewart [29] have pointed out that for $\theta$, and given the information in the subspace $\mathcal{V}$, a better, in residual sense, approximate eigenvector can be computed; the norm of the residual of this so-called refined Ritz vector is given by the

quantity

$$\|\mathbf{r}'\|_2 = \min_{\mathbf{u}\in\mathcal{V}} \|\mathbf{A}\mathbf{u} - \theta\mathbf{u}\|,$$

represented in the fifth column in Table 3.3.

These experiments set the stage for the domain decomposition experiments.

## 3.5.2 Spectrum of the error propagator

From §3.2.5 we know that the convergence properties of the Jacobi iterative method depend on the spectrum of the error propagator $\mathbf{M}_C^{-1}\mathbf{N}$. Therefore, we will investigate these spectra for some typical situations. We consider the asymptotic case $\theta = \lambda$. Although $\theta$ approximates $\lambda$ in practice, during the iteration process $\theta$ becomes very close to $\lambda$, and that is the reason we think that the asymptotic case gives a good indication.

### 3.5.2.1 Predicted and computed spectra

First we consider the determination of the parameter $\alpha_{\text{opt}}$ (3.50) for the simple optimized coupling. The value of $\alpha_{\text{opt}}$ depends on the extremal values $\mu$ and $M$ of the collection of dominant roots $\widehat{E}$ (3.48) for which $\alpha_{\text{opt}}$ is optimized. The value $\mu$ depends amongst others on $\theta$, and $M$ only depends on $h_x, h_y$, and on the coefficients $a$ and $b$.

We illustrate the sensitivity of $\alpha_{\text{opt}}$ w.r.t. the lower bound $\mu$, for $\theta$ equal to the largest eigenvalue $\lambda^{(1,1)}$ of the Laplace operator, with $\omega_x = 2, \omega_y = 1, n_x = 63, n_y = 31$ and $n_{x1} = 26$. For a dominant root $\zeta_+^{(l)}$, $\lambda^{(l)}$ in (3.52) should be smaller than 0. Then $\dfrac{4b}{h_y^2}\sin^2\left(\dfrac{\pi}{2}\dfrac{l_e}{n_y+1}\right) > \theta$. Since $\theta \approx \dfrac{5}{4}\pi^2$ and $\dfrac{4b}{h_y^2}\sin^2\left(\dfrac{\pi}{2}\dfrac{l_e}{n_y+1}\right) \approx l_e^2\pi^2$, we have approximately that $l_e^2 > \dfrac{5}{4}$. The smallest such integer $l_e$ is $l_e = 2$. In order to show that this is a sharp value for $l_e$ and thus a sharp lower bound for the $\mu$ (3.53), we shall compare the case $l_e = 2$ with the case for the smaller value $l_e = 1.2$. We also included the case $l_e = 4$, where apart from the mode $l_y = 1$, the modes $l_y = 2$ and $l_y = 3$ are excluded from the optimization process (i.e. for the computation of an optimal $\alpha$).

For these three cases ($l_e = 2, l_e = 4$, and $l_e = 1.2$) we have computed the corresponding $\alpha$ ($\alpha = -1.6287\ldots, \alpha = -2.1279\ldots$, and $\alpha = -1.2800\ldots$, respectively). In Fig. 3.4 the predicted amplification of the error propagator $\mathbf{M}_C^{-1}\mathbf{N}$ for these values of $\alpha$ are shown. Here we calculated for each mode (with wavenumber $l_y$) the expected amplification $|\sigma^{(l_y)}|$ with expression (3.46). Indeed, we see that (for $l_e = 2$) the second leftmost circle ($l_y = 2$) in Fig. 3.4 represents the same value as for the rightmost circle ($l_y = 31$), which was our goal. If $l_e$ is close to 1, then because the mode $l_y = 1$ can not be damped at all, the overall damping for $l_e = 1.2$ is predicted to be less, whereas $l_e = 4$ should lead to a better damping of the remaining modes $l_y = 4, \ldots, 31$ that are taken into account, which is confirmed in Fig. 3.4 for different values of $\alpha$.

Fig. 3.5 shows the *exact* nonzero eigenvalues $\sigma$ of $\mathbf{M}_C^{-1}\mathbf{N}$ sorted by magnitude for different values of $\alpha$. We also plotted in this figure the *predicted* nonzero eigenvalues sorted by magnitude. We see that the predictions are very accurate.

FIGURE 3.4. *Predicted amplification of the error propagator* $\mathbf{M}_C^{-1}\mathbf{N}$ *with simple optimized coupling for the largest eigenvalue* $\lambda^{(1,1)}$ *of the Laplace operator for* $l_e = 2$, $l_e = 4$, *and* $l_e = 1.2$. *For explanation, see* §3.5.2.1.

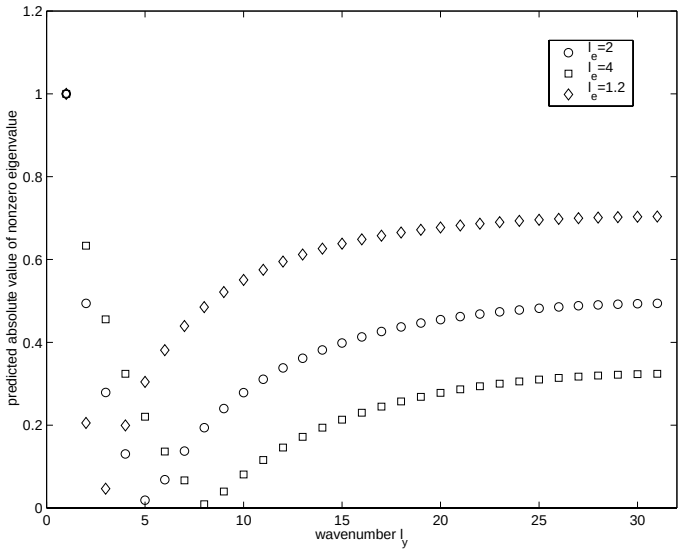
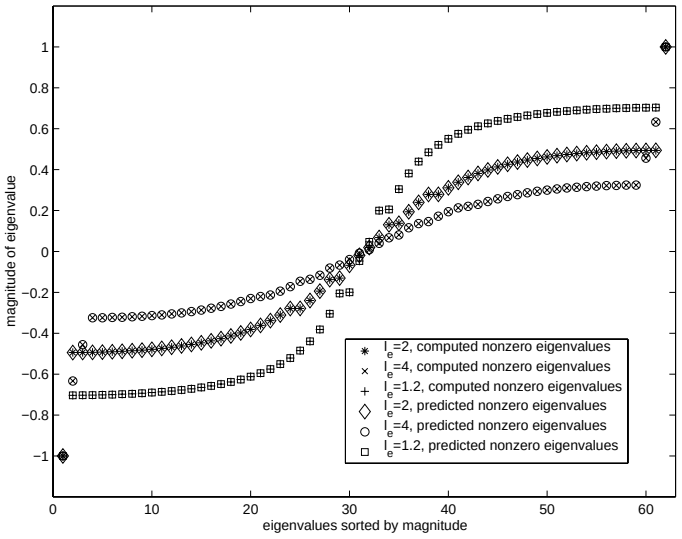



FIGURE 3.5. *Predicted and computed nonzero eigenvalues of the error propagator* $\mathbf{M}_C^{-1}\mathbf{N}$ *with simple optimized coupling for the largest eigenvalue* $\lambda^{(1,1)}$ *of the Laplace operator for* $l_e = 2$, $l_e = 4$, *and* $l_e = 1.2$. *For explanation, see* §3.5.2.1.



In Fig. 3.5 we see also the effect of the value $l_e$ on the eigenvalues. Again, we see that it is better to overestimate $l_e$ than underestimate. The point symmetry in Fig. 3.5 is due to the fact that if $\sigma$ is an eigenvalue of $\mathbf{M}_C^{-1}\mathbf{N}$ then $-\sigma$ is also an eigenvalue (remark (iv) of §3.2.5). Furthermore, note that for each process one eigenvalue is equal to 1, independent of $\alpha$. By a combination of remark (v) of §3.2.5 and the discussion at the end of §3.3.2, we see that the corresponding eigenvector is of the form $\underline{\mathbf{y}}$ that corresponds to the eigenvector $\mathbf{y}$ that we are looking for with our Jacobi-Davidson process. Hence the occurrence of 1 in the spectrum is not a problem: the projections in the correction equation take care of this, as we will show now.

#### 3.5.2.2 Deflation

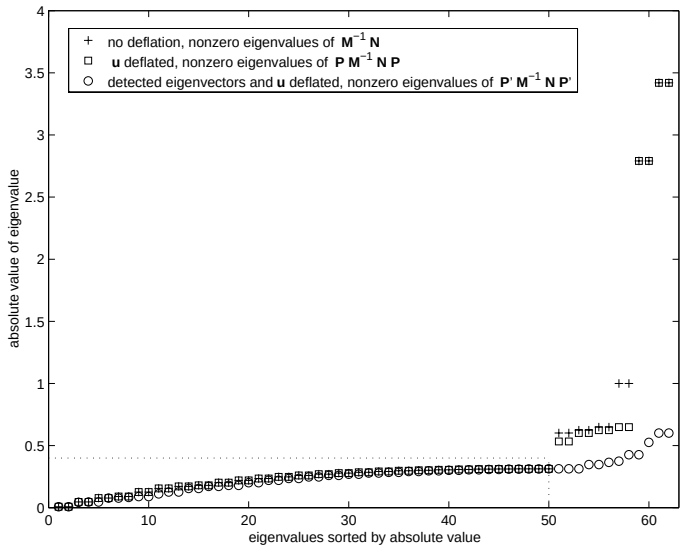
Now we show, by means of an example, how deflation improves the condition of the preconditioned correction equation (3.26). For the discretized Laplace operator we take $\omega_x = \omega_y = 1, n_x = n_y = 31, n_{x1} = 15$ and $\theta = \lambda^{(4,4)}$. There are 19 eigenvalues larger than $\lambda^{(4,4)}$. If we determine the $\alpha_{\text{opt}}$ for the simple optimized coupling, then $\widetilde{l_e} \approx 5.6944$. So the modes $l_y = 1, \ldots, 6$ are not taken into account for the optimization of $\alpha$, since they do not show dominant behavior. Hence we do not necessarily damp these modes with the resulting $\alpha_{\text{opt}}$.
One of them, more precisely the mode $l_y = 4$, is connected to the $y$-component of the eigenvector $\varphi^{(4,4)}$ corresponding to $\lambda^{(4,4)}$: this mode can not be controlled at all with $\alpha$ because the operator $\mathbf{A}$ is shifted by $\lambda^{(4,4)}$ and therefore singular in the direction of $\varphi^{(4,4)}$. In the correction equation (3.26) the operator stays well-conditioned due to the projection $\mathbf{P}$ that deflates exactly the direction $\mathbf{u} = \varphi^{(4,4)}$. Since the error propagator originates from the enhanced operator in the correction equation, this projection is actually incorporated in the error propagator (§3.3.2): $\mathbf{P}\mathbf{M}_C^{-1}\mathbf{N}\mathbf{P}$.
The other non-dominant modes $l_y = 1, 2, 3, 5, 6$, can not be controlled by $\alpha_{\text{opt}}$. But, as remarked in §3.4.4, in practice one starts the computation with the largest eigenvalues and when arrived at $\lambda^{(4,4)}$, the 19 largest eigenvalues with corresponding eigenvectors are already computed and will be deflated from the operator $\mathbf{B}$. Deflation in the enhanced correction equation is performed by the projection

$$\mathbf{P}' \equiv \mathbf{I} - \mathbf{M}_C^{-1}\underline{\mathbf{X}}\left(\underline{\mathbf{X}}^*\mathbf{M}_C^{-1}\underline{\mathbf{X}}\right)^{-1}\underline{\mathbf{X}}^*.$$

Here $\underline{\mathbf{X}} \equiv (\mathbf{X}_1^T, X_\ell^T, 0^T, 0^T, X_r^T, \mathbf{X}_2^T)^T$, where $\mathbf{X} \equiv (\mathbf{X}_1^T, X_\ell^T, X_r^T, \mathbf{X}_2^T)^T$ is a matrix of which the columns form an orthonormal basis for the space spanned by the 19 already computed eigenvectors and the approximate 20th eigenvector. This implies that we are dealing with the error propagator $\mathbf{P}'\mathbf{M}_C^{-1}\mathbf{N}\mathbf{P}'$.

For $\alpha_{\text{opt}}$ we computed the nonzero eigenvalues of $\mathbf{M}_C^{-1}\mathbf{N}$, $\mathbf{P}\mathbf{M}_C^{-1}\mathbf{N}\mathbf{P}$ and $\mathbf{P}'\mathbf{M}_C^{-1}\mathbf{N}\mathbf{P}'$. In Fig. 3.6 their absolute values are plotted. The '+'-s (no deflation) indicate that the most right 12 eigenvalues have not been controlled by $\alpha_{\text{opt}}$. This is in agreement with the fact that the modes $l_y = 1, \ldots, 6$ have not been taken into account for the determination of $\alpha_{\text{opt}}$: to each mode $l_y$ there correspond exactly two eigenvalues $-\sigma^{(l_y)}$ and $+\sigma^{(l_y)}$. Two eigenvalues

FIGURE 3.6. *The effect of deflation on the nonzero eigenvalues of the error propagator with simple optimized coupling. For explanation, see §3.5.2.2. The dotted lines indicate the area of Fig. 3.7.*




have absolute value $1$ (position $57$ and $58$ on the horizontal axis). They correspond to the eigenvector $\varphi^{(4,4)}$ of $\mathbf{A}$.
The '$\square$'-s show that deflation with $\mathbf{u}$ makes these absolute values become less than $1$. But, with deflation by $\mathbf{u}$, the other uncontrolled eigenvalues stay where they were without deflation; four absolute values are even larger than $2.5$. Fortunately, deflation with the $19$ already computed eigenvectors drastically reduces these absolute values, as the '$\circ$'-s show.

From this example we learned that deflation may help to cluster the part of the spectrum that we can not control with the coupling parameters, and therefore improves the conditioning of the preconditioned correction equation. The remaining part of the spectrum, that is the eigenvalues that are in control (indicated by the dotted lines in Fig. 3.6), may be damped even more. This will be subject of the next section.

#### 3.5.2.3 Stronger coupling

At the end of §3.4.4, it was illustrated that the inclusion of interactions parallel to the artificial boundary provides more coupling parameters by which a better coupling can be realized. We will apply this now to the example in §3.5.2.2 in order to investigate how much we can improve the spectrum of the error propagator and how accurate the value of the predicted amplification $\sigma'_{\text{opt}}$ is for the different types of coupling.

Table 3.4 contains the values of the coupling parameters and the predicted amplification $\sigma'_{\text{opt}}$ for the different types of coupling when $l_e = 7$, as in §3.5.2.2. These values are obtained

TABLE 3.4. *Values of coupling parameters and predicted amplification $\sigma'_{opt}$ for four types of optimized coupling. For explanation, see §3.5.2.3.*

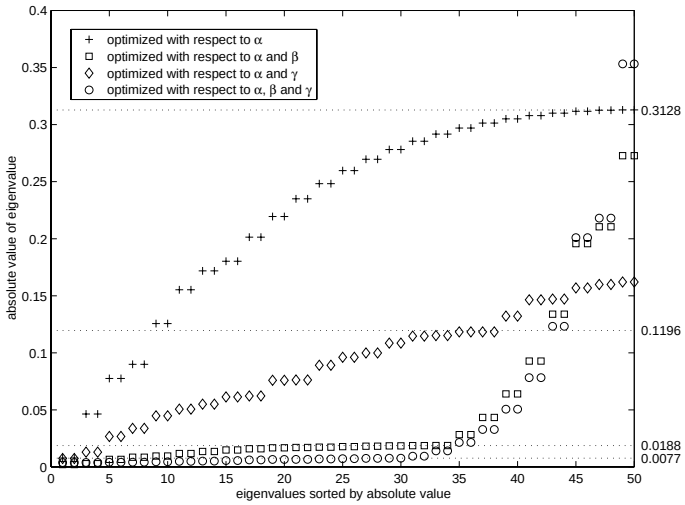
| type no. | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| optimized w.r.t. | $\alpha$ | $\alpha, \beta$ | $\alpha, \gamma$ | $\alpha, \beta, \gamma$ |
| $\alpha$ | $-2.138$ | $-0.4988$ | $-1.373$ | $-0.2080$ |
| $\beta$ | | $0.001375$ | | $0.001959$ |
| $\gamma$ | | | $0.0002230$ | $-0.0001352$ |
| predicted $\sigma'_{\text{opt}}$ | $0.3128$ | $0.01875$ | $0.1196$ | $0.007686$ |

by application of a Rémès algorithm to expression (3.56). As in the final example of §3.4.4, we see that be the best coupling is predicted to be of type $4$, followed by type $2$, and then type $3$. But, the question remains what the exact spectrum may be for these types op coupling.

We computed the exact nonzero eigenvalues of the error propagator $\mathbf{M}_C^{-1}\mathbf{N}$ for the four types of coupling from Table 3.4. From §3.5.2.2, we know that with the coupling parameters we only control the $2n_y - 12 = 50$ nonzero eigenvalues of the error propagator with lowest absolute value. Therefore, we exclude the $12$ other nonzero eigenvalues from our further discussion. In Fig. 3.7 the $50$ eigenvalues with lowest absolute value are plotted. The corresponding predicted values of $\sigma'_{\text{opt}}$ are indicated by dotted lines in Fig. 3.7. From inspection of the eigenvectors, we have verified that for the four different types of coupling, the $12$ eigenvalues with highest absolute value that are excluded correspond to the modes $l_y = 1, \ldots, 6$. (Computation of the eigenvectors is rather time consuming. Therefore, we restricted ourselves here to a grid that is coarser than the one in the example at the end of §3.4.4.)

Indeed, as predicted, it pays off to include more coupling parameters. For type $1$ the predicted value of $\sigma'_{\text{opt}}$ is almost exact. The value for type $3$ seems to be accurate for the eigenvalues at positions $1, \ldots, 38$. For types $2$ and $4$, the value becomes less accurate after position $34$. We believe that this is because of neglecting the $\zeta_-$ terms in the expression for $\sigma'_{\text{opt}}$: for types $2$ and $4$ the eigenvectors, that correspond to the eigenvalues with position larger than $34$, have a low value of $l_y$. In our quest for optimizing the spectral radius of the error propagator, we have now arrived at a level where we can no longer ignore the contributions of the terms $\zeta_-$. This is confirmed by inspecting the eigenvectors: the eigenvalues that deviate from the predicted $\sigma'_{\text{opt}}$ have eigenvectors that correspond to low values of $l_y$. But still, the predicted $\sigma'_{\text{opt}}$ gives a good indication for the quality of the coupling and will be better for finer grids.

FIGURE 3.7. *The effect of different types of optimized coupling on the nonzero eigenvalues of the error propagator. The values of the coupling parameters are given in Table 3.4. The corresponding predicted values of $\sigma'_{opt}$ are indicated by dotted lines. For explanation, see §3.5.2.3.*




## 3.5.3 Effect on the overall process

In §3.5.2 spectra of the error propagator have been studied. These spectra provide information on the convergence behavior of the Jacobi iterative method. Now we turn our attention to the overall Jacobi-Davidson method itself. We are interested in how approximate solutions of the correction equation, obtained with a linear solver ('the innerloop'), affects the Jacobi-Davidson process ('the outerloop').

Here we consider two types of coupling:

1. the simple optimized coupling with one coupling parameter $\alpha$,

2. the Neumann-Dirichlet coupling.

Although we have seen in §3.4.4 and §3.5.2.3, that there exist better choices for the coupling, we believe that the overall process with the simple optimized coupling gives a good indication of what we may expect for the stronger optimized couplings. The choice for the Neumann-Dirichlet coupling is motivated by the fact that it is commonly used in domain decomposition methods.

The testproblem will be the same as the one in §3.5.2.1. First we discuss the Jacobi iterative method as a solver for the correction equation. We do this for both the left and right preconditioned variant. Then we compare the results with those obtained by the GMRES method.

TABLE 3.5. *Convergence history of Jacobi-Davidson applied to the discretized eigenvalue problem of the two-dimensional Laplace operator for approximate solutions to the correction equation obtained with left (left) and right (right) preconditioned Jacobi iterations on two subdomains and simple optimized coupling. For explanation see §3.5.3.1.*

optimized coupling, $l_e = 2$

| | left DD-preconditioned | | | | right DD-preconditioned | | | |
|---|---|---|---|---|---|---|---|---|
| step | $\theta - \lambda$ | $\|\mathbf{r}\|_2$ | $\|\mathbf{r}'\|_2$ | $\alpha$ | $\theta - \lambda$ | $\|\mathbf{r}\|_2$ | $\|\mathbf{r}'\|_2$ | $\alpha$ |
| | 3 Jacobi inner iterations | | | | 2 Jacobi inner iterations | | | |
| 1 | -1.61e-01 | 4.19e+00 | 4.19e+00 | -1.6275 | -1.61e-01 | 4.19e+00 | 4.19e+00 | -1.6275 |
| 2 | -4.98e-03 | 3.14e+00 | 2.55e+00 | -1.6287 | -4.98e-03 | 3.14e+00 | 2.55e+00 | -1.6287 |
| 3 | -2.20e-04 | 1.90e-01 | 1.81e-01 | -1.6287 | -2.20e-04 | 1.90e-01 | 1.81e-01 | -1.6287 |
| 4 | -1.62e-07 | 7.12e-03 | 6.74e-03 | -1.6287 | -1.62e-07 | 7.12e-03 | 6.74e-03 | -1.6287 |
| 5 | -2.13e-12 | 4.16e-05 | 3.91e-05 | -1.6287 | -2.09e-12 | 4.16e-05 | 3.91e-05 | -1.6287 |
| 6 | -1.53e-13 | 1.36e-06 | 9.37e-07 | -1.6287 | -1.47e-13 | 1.36e-06 | 9.37e-07 | -1.6287 |
| 7 | -1.62e-13 | 8.43e-09 | 5.78e-09 | -1.6287 | -1.81e-13 | 8.43e-09 | 5.78e-09 | -1.6287 |
| 8 | -1.39e-13 | 1.19e-10 | 8.84e-11 | | -1.44e-13 | 1.19e-10 | 8.84e-11 | |
| | 4 Jacobi inner iterations | | | | 3 Jacobi inner iterations | | | |
| 1 | -1.61e-01 | 4.19e+00 | 4.19e+00 | -1.6275 | -1.61e-01 | 4.19e+00 | 4.19e+00 | -1.6275 |
| 2 | -4.23e-03 | 2.89e+00 | 2.43e+00 | -1.6287 | -4.23e-03 | 2.89e+00 | 2.43e+00 | -1.6287 |
| 3 | -2.70e-05 | 6.42e-02 | 6.20e-02 | -1.6287 | -2.70e-05 | 6.42e-02 | 6.20e-02 | -1.6287 |
| 4 | -5.95e-09 | 1.02e-03 | 7.36e-04 | -1.6287 | -5.95e-09 | 1.02e-03 | 7.36e-04 | -1.6287 |
| 5 | -1.53e-13 | 2.84e-06 | 2.61e-06 | -1.6287 | -1.58e-13 | 2.84e-06 | 2.61e-06 | -1.6287 |
| 6 | -1.76e-13 | 2.81e-08 | 1.54e-08 | -1.6287 | -9.95e-14 | 2.81e-08 | 1.54e-08 | -1.6287 |
| 7 | -1.44e-13 | 8.33e-12 | 8.30e-12 | | -1.42e-13 | 8.34e-12 | 8.28e-12 | |

#### 3.5.3.1 The Jacobi iterative process

In §3.5.2.1 we have computed the spectra of the error propagator $\mathbf{M}_C^{-1}\mathbf{N}$, for $\alpha_{\text{opt}}$ and two other near optimal values of $\alpha$. We further investigate these three cases for the Jacobi iterative process.

Table 3.5 shows the convergence behavior of Jacobi-Davidson, when the correction equation is solved with the Jacobi iterative method and with coupling parameter $\alpha_{\text{opt}}$, obtained for $l_e = 2$. The left (on the left) and right (on the right) preconditioned variant are presented. Moreover, we have varied the number of Jacobi inner iterations.

When we compare the top part of Table 3.5 with the bottom part, then we see that more Jacobi inner iterations lead to less outer iterations for the same precision. More Jacobi iterations yields a better approximation of the correction vector and a better approximation of the correction vector results in fewer Jacobi-Davidson steps. When we compare the left part with the right part in Table 3.5, then we see that $m$ steps with right preconditioned Jacobi iterations produces exactly the same results as with $m + 1$ left preconditioned Jacobi iterations. This is explained by stage 1 in §3.3.3 of right preconditioning: one extra preconditioning step is performed.

From §3.5.2.1 we know that the spectra of the error propagator are less optimal for $l_e = 4$ and $l_e = 1.2$, and therefore Jacobi will perform not as good as for $l_e = 2$. How does this

TABLE 3.6. *Convergence history of Jacobi-Davidson applied to the discretized eigenvalue problem for the two-dimensional Laplace operator for approximate solutions to the correction equation obtained with 3 left preconditioned Jacobi iterations on two subdomains and two almost optimal simple couplings. For explanation see §3.5.3.1.*

| step | $\theta-\lambda$ | $\|\mathbf{r}\|_2$ | $\|\mathbf{r}'\|_2$ | $\alpha$ | $\theta-\lambda$ | $\|\mathbf{r}\|_2$ | $\|\mathbf{r}'\|_2$ | $\alpha$ |
|---|---|---|---|---|---|---|---|---|
| | $l_e=4$ | | | | $l_e=1.2$ | | | |
| 1 | -1.61e-01 | 4.19e+00 | 4.19e+00 | -2.1274 | -1.61e-01 | 4.19e+00 | 4.19e+00 | -1.2729 |
| 2 | -2.93e-03 | 2.27e+00 | 2.00e+00 | -2.1279 | -1.33e-02 | 5.03e+00 | 3.31e+00 | -1.2794 |
| 3 | -1.12e-03 | 5.92e-01 | 4.62e-01 | -2.1279 | -1.92e-06 | 2.96e-02 | 2.94e-02 | -1.2800 |
| 4 | -1.46e-05 | 6.50e-02 | 5.83e-02 | -2.1279 | -4.11e-10 | 6.69e-04 | 5.57e-04 | -1.2800 |
| 5 | -4.02e-10 | 5.91e-04 | 5.71e-04 | -2.1279 | -1.18e-12 | 5.35e-05 | 3.97e-05 | -1.2800 |
| 6 | -2.47e-12 | 6.71e-05 | 4.05e-05 | -2.1279 | -1.24e-13 | 1.45e-06 | 1.21e-06 | -1.2800 |
| 7 | -1.47e-13 | 1.82e-07 | 1.14e-07 | -2.1279 | -3.13e-13 | 9.31e-08 | 5.82e-08 | -1.2800 |
| 8 | -1.67e-13 | 2.84e-10 | 2.82e-10 | | -1.46e-13 | 2.83e-09 | 2.09e-09 | -1.2800 |
| 9 | | | | | -1.72e-13 | 1.24e-10 | 1.09e-10 | |

affect the Jacobi-Davidson process? In Table 3.6 data are presented for three left preconditioned Jacobi iterations in each outer iteration, for $l_e = 4$ (left) and $l_e = 1.2$ (right). We should compare this with the top left part of Table 3.5. From this we see, that also Jacobi-Davidson performs less well for less optimal couplings.

Now we consider the Neumann-Dirichlet coupling. In our enhancement terminology (cf. §3.2.2) this can be interpreted as a Neumann boundary condition on the left: $C_{\ell\ell} = I$ and $C_{\ell r} = -I$, and a Dirichlet boundary condition on the right: $C_{r\ell} = I$ and $C_{rr} = I$. For dominated behavior (cf. §3.4.3.1 (iii), and §3.4.4 (3.48)) and for two subdomains it follows from (3.16) that

$$\sigma^2 \approx \frac{\overline{(\zeta-1)}\,(1+\zeta)}{\overline{(1-\zeta)}\,(\zeta+1)} = -1.$$

From this we see that for $\theta = \lambda^{(1,1)}$, the error propagator has, besides $-1$ and $+1$, only eigenvalues near $-\sqrt{-1}$ and $\sqrt{-1}$. Hence, the eigenvectors of $\mathbf{M}_C^{-1}\mathbf{N}$ will hardly be damped. Therefore, the Jacobi iteration will not perform well with Neumann-Dirichlet coupling. From Table 3.7 we see that Jacobi-Davidson clearly suffers from this effect.

TABLE 3.7. *Convergence history of Jacobi-Davidson applied to the discretized eigenvalue problem for the two-dimensional Laplace operator for approximate solutions to the correction equation obtained with left (left) and right (right) preconditioned Jacobi iterations on two subdomains and Neumann-Dirichlet coupling. For explanation see §3.5.3.1.*

Neumann-Dirichlet coupling

| | left DD-preconditioned | | | right DD-preconditioned | | |
|---|---|---|---|---|---|---|
| step | $\theta - \lambda$ | $\|\mathbf{r}\|_2$ | $\|\mathbf{r}'\|_2$ | $\theta - \lambda$ | $\|\mathbf{r}\|_2$ | $\|\mathbf{r}'\|_2$ |
| | 4 Jacobi inner iterations | | | 3 Jacobi inner iterations | | |
| 1 | -1.61e-01 | 4.19e+00 | 4.19e+00 | -1.61e-01 | 4.19e+00 | 4.19e+00 |
| 2 | -5.07e-02 | 8.72e+00 | 3.98e+00 | -5.07e-02 | 8.72e+00 | 3.98e+00 |
| 3 | -1.79e-02 | 4.85e+00 | 3.29e+00 | -1.79e-02 | 4.85e+00 | 3.29e+00 |
| 4 | -1.20e-02 | 2.40e+00 | 2.03e+00 | -1.20e-02 | 2.40e+00 | 2.03e+00 |
| 5 | -4.55e-03 | 2.69e+00 | 1.68e+00 | -4.55e-03 | 2.69e+00 | 1.68e+00 |
| 6 | -2.93e-04 | 6.90e-01 | 6.13e-01 | -2.93e-04 | 6.90e-01 | 6.13e-01 |
| 7 | -1.40e-04 | 3.74e-01 | 3.29e-01 | -1.40e-04 | 3.74e-01 | 3.29e-01 |
| 8 | -2.00e-05 | 2.10e-01 | 1.74e-01 | -2.00e-05 | 2.10e-01 | 1.74e-01 |
| 9 | -4.11e-06 | 7.32e-02 | 6.63e-02 | -4.11e-06 | 7.32e-02 | 6.63e-02 |
| 10 | -8.12e-07 | 3.88e-02 | 3.49e-02 | -8.12e-07 | 3.88e-02 | 3.49e-02 |
| 11 | -1.54e-07 | 1.41e-02 | 1.12e-02 | -1.54e-07 | 1.41e-02 | 1.12e-02 |
| 12 | -1.50e-08 | 5.84e-03 | 5.28e-03 | -1.50e-08 | 5.84e-03 | 5.28e-03 |
| 13 | -3.20e-09 | 2.62e-03 | 1.59e-03 | -3.19e-09 | 2.62e-03 | 1.58e-03 |
| 14 | -7.27e-10 | 1.22e-03 | 1.01e-03 | -3.68e-10 | 9.02e-04 | 8.00e-04 |
| 15 | -1.31e-10 | 5.86e-04 | 5.38e-04 | -1.30e-10 | 5.82e-04 | 5.35e-04 |
| 16 | -2.34e-11 | 2.63e-04 | 1.72e-04 | -2.35e-11 | 2.63e-04 | 1.72e-04 |
| 17 | -2.26e-12 | 5.03e-05 | 4.78e-05 | -4.16e-13 | 5.03e-05 | 4.78e-05 |
| 18 | -7.46e-13 | 2.08e-05 | 1.65e-05 | -5.68e-14 | 2.08e-05 | 1.65e-05 |
| 19 | -1.63e-13 | 3.90e-06 | 3.21e-06 | -7.53e-13 | 3.88e-06 | 3.19e-06 |
| 20 | 4.12e-13 | 1.49e-06 | 1.25e-06 | 1.14e-13 | 1.27e-06 | 1.04e-06 |
| 21 | 9.95e-13 | 8.53e-07 | 7.63e-07 | 6.25e-13 | 3.60e-07 | 2.54e-07 |
| 22 | -6.79e-13 | 2.55e-07 | 1.30e-07 | -3.91e-13 | 2.30e-07 | 1.25e-07 |
| 23 | 4.01e-13 | 3.81e-08 | 3.56e-08 | -7.11e-14 | 3.81e-08 | 3.56e-08 |
| 24 | 7.11e-14 | 1.18e-08 | 8.40e-09 | -5.47e-13 | 1.18e-08 | 8.39e-09 |
| 25 | 4.90e-13 | 1.45e-09 | 1.41e-09 | 2.98e-13 | 1.19e-09 | 1.16e-09 |
| 26 | 6.98e-13 | 6.58e-10 | 6.30e-10 | -5.90e-13 | 5.02e-10 | 4.80e-10 |

TABLE 3.8. *Convergence history of Jacobi-Davidson applied to the discretized eigenvalue problem for the two-dimensional Laplace operator for approximate solutions to the correction equation obtained with left (left) and right (right) preconditioned GMRES on two subdomains and simple optimized coupling. For explanation see §3.5.3.2.*

optimized coupling, $l_e = 2$

| | left DD-preconditioned | | | | right DD-preconditioned | | | |
|---|---|---|---|---|---|---|---|---|
| step | $\theta - \lambda$ | $\|\mathbf{r}\|_2$ | $\|\mathbf{r}'\|_2$ | $\alpha$ | $\theta - \lambda$ | $\|\mathbf{r}\|_2$ | $\|\mathbf{r}'\|_2$ | $\alpha$ |
| | GMRES(3) | | | | GMRES(2) | | | |
| 1 | -1.61e-01 | 4.19e+00 | 4.19e+00 | -1.6275 | -1.61e-01 | 4.19e+00 | 4.19e+00 | -1.6275 |
| 2 | -2.72e-05 | 1.67e-01 | 1.67e-01 | -1.6287 | -3.74e-05 | 1.16e-01 | 1.16e-01 | -1.6287 |
| 3 | -3.05e-08 | 6.68e-03 | 6.23e-03 | -1.6287 | -5.89e-08 | 6.63e-03 | 5.43e-03 | -1.6287 |
| 4 | -3.06e-11 | 2.72e-04 | 2.71e-04 | -1.6287 | -1.46e-11 | 1.19e-04 | 1.13e-04 | -1.6287 |
| 5 | 1.78e-15 | 1.72e-06 | 1.66e-06 | -1.6287 | -1.56e-13 | 1.46e-06 | 1.26e-06 | -1.6287 |
| 6 | -2.59e-13 | 1.34e-08 | 1.03e-08 | -1.6287 | -1.69e-13 | 6.81e-09 | 5.71e-09 | -1.6287 |
| 7 | -1.26e-13 | 7.94e-10 | 6.71e-10 | | -7.28e-14 | 4.38e-11 | 4.03e-11 | |
| | GMRES(4) | | | | GMRES(3) | | | |
| 1 | -1.61e-01 | 4.19e+00 | 4.19e+00 | -1.6275 | -1.61e-01 | 4.19e+00 | 4.19e+00 | -1.6275 |
| 2 | -1.52e-06 | 3.07e-02 | 3.02e-02 | -1.6287 | -1.34e-06 | 2.76e-02 | 2.71e-02 | -1.6287 |
| 3 | -1.39e-12 | 3.35e-05 | 3.32e-05 | -1.6287 | -4.85e-12 | 4.30e-05 | 4.13e-05 | -1.6287 |
| 4 | -1.42e-13 | 1.87e-07 | 1.76e-07 | -1.6287 | -1.42e-13 | 7.62e-07 | 7.31e-07 | -1.6287 |
| 5 | -1.79e-13 | 1.21e-09 | 1.17e-09 | -1.6287 | -1.19e-13 | 3.20e-09 | 3.19e-09 | -1.6287 |
| 6 | -1.85e-13 | 4.64e-12 | 4.09e-12 | | -1.28e-13 | 1.10e-11 | 1.05e-11 | |

#### 3.5.3.2 GMRES

At the end of §3.2.3 we noted that Krylov subspace methods can be viewed as accelerators of the Jacobi iterative method. If we apply GMRES for the solution of the correction equation, instead of Jacobi iterations as in §3.5.3.1, then we should expect at least the same speed of convergence in the inner iteration. As a consequence, the speed of convergence of the Jacobi-Davidson (outer) iteration should be not worse but presumably better.

Our expectations are confirmed by the results in Table 3.8, for the simple optimized coupling and in Table 3.9 for the Neumann-Dirichlet coupling. For the same type of coupling one should compare the data for GMRES($m$) with $m$ Jacobi iterations: GMRES optimizes over the Krylov subspace spanned by powers of the (preconditioned) operator, whereas Jacobi uses only the last iteration vector for the computation of a solution to the linear system.

Note that with left preconditioned GMRES(4) and with Neumann-Dirichlet coupling, we have almost recovered the exact Jacobi-Davidson process from §3.5.1. This can be explained as follows. The eigenvalue distribution of the error propagator has besides $-1$ and $+1$, all other eigenvalues clustered around $\pm\sqrt{-1}$ for two subdomains. However, for four distinct eigenvalues, GMRES needs four steps at most for convergence. So the spectral properties of the error propagator for two subdomains with Neumann-Dirichlet coupling are worse for the Jacobi iterative method but ideal for the acceleration part of GMRES. This is not a typical situation. In §3.5.4 we will see how the picture changes for more subdomains and with less accurate preconditioners.

TABLE 3.9. *Convergence history of Jacobi-Davidson applied to the discretized eigenvalue problem for the two-dimensional Laplace operator for approximate solutions to the correction equation obtained with left (left) and right (right) preconditioned GMRES on two subdomains and Neumann-Dirichlet coupling. For explanation see §3.5.3.2.*

Neumann-Dirichlet coupling

| | left DD-preconditioned | | | right DD-preconditioned | | |
|---|---|---|---|---|---|---|
| step | $\theta - \lambda$ | $\|\mathbf{r}\|_2$ | $\|\mathbf{r}'\|_2$ | $\theta - \lambda$ | $\|\mathbf{r}\|_2$ | $\|\mathbf{r}'\|_2$ |
| | GMRES(3) | | | GMRES(2) | | |
| 1 | -1.61e-01 | 4.19e+00 | 4.19e+00 | -1.61e-01 | 4.19e+00 | 4.19e+00 |
| 2 | -1.20e-04 | 3.80e-01 | 3.80e-01 | -5.87e-05 | 8.67e-02 | 8.48e-02 |
| 3 | -5.48e-05 | 2.00e-01 | 1.96e-01 | -7.21e-09 | 2.19e-03 | 2.18e-03 |
| 4 | -1.13e-06 | 2.78e-02 | 1.73e-02 | -1.71e-13 | 1.57e-06 | 1.22e-06 |
| 5 | -1.99e-08 | 5.95e-03 | 4.43e-03 | -1.49e-13 | 3.25e-08 | 3.09e-08 |
| 6 | -8.17e-12 | 7.64e-05 | 7.48e-05 | -1.74e-13 | 3.10e-12 | 2.98e-12 |
| 7 | -1.79e-13 | 3.88e-06 | 3.83e-06 | | | |
| 8 | -1.99e-13 | 1.41e-07 | 1.32e-07 | | | |
| 9 | -1.14e-13 | 1.90e-09 | 1.61e-09 | | | |
| 10 | -1.71e-13 | 4.80e-11 | 2.58e-11 | | | |
| | GMRES(4) | | | GMRES(3) | | |
| 1 | -1.61e-01 | 4.19e+00 | 4.19e+00 | -1.61e-01 | 4.19e+00 | 4.19e+00 |
| 2 | -9.65e-07 | 8.55e-03 | 6.10e-03 | -9.65e-07 | 8.55e-03 | 6.10e-03 |
| 3 | -1.44e-13 | 5.84e-10 | 5.79e-10 | -1.55e-13 | 5.35e-10 | 5.30e-10 |
| 4 | -1.21e-13 | 8.56e-14 | 1.01e-14 | -1.49e-13 | 3.92e-14 | 4.12e-14 |

### 3.5.4 More subdomains

We describe an experiment that illustrates what happens when the number of subdomains is increased. For each number of subdomains we keep the preconditioner fixed.

Our modelproblem is a channel that is made larger by extending new subdomains. We compute the largest eigenvalue and corresponding eigenvector of the Laplace operator on this channel. After adding a subdomain, this results in a different eigenvalue problem. For $p$ subdomains the physical size and number of gridpoints in the $y$ direction are taken to be fixed: $\omega_y = 1$ and $n_y = 63$, whereas in the $x$ direction they increase: $\omega_x = p$ and $n_x = 63 + (p-1) \cdot 64$ for $1 \leq p \leq 6$.

Now, the idea is that the DD-preconditioner consists of block matrices defined on the enhanced subdomain grids. For the channel this results in one block matrix of size $(63 + 1) \times (63 + 1)$ (corresponding to the first subdomain on the left), $p - 2$ block matrices of size $(64+2) \times (64+2)$ (corresponding to the $p-2$ intermediate subdomains) and one block matrix of size $(64 + 1) \times (64 + 1)$ (corresponding to the last subdomain on the right). If we select the same coupling between all subdomains, then we need to know the inverse action of $3$ blocks (corresponding to the left, right, and a single intermediate subdomain). Furthermore, we construct the preconditioner only for the value of $\theta_1$ of the first Jacobi-Davidson step. This fixed preconditioner is used for all iteration steps.

In order to be able to interpret the results properly, we have checked how Jacobi-Davidson with accurate solutions to the correction equation on the undecomposed domain (the 'exact' process) behaves. In Fig. 3.8 and Fig. 3.9 this is represented by the solid line.

We consider simple optimized (type 1), strong optimized (type 4), and Neumann-Dirichlet couplings. In each Jacobi-Davidson step we solve the correction equation approximately by right preconditioned GMRES(3). The number of nonzero eigenvalues of the error propagator is proportional to the number of subdomains. Because of this, it is reasonable that with a fixed number of inner iterations the accuracy will deteriorate for more subdomains.

Fig. 3.8 represents the convergence history of Jacobi-Davidson for the 'exact process' and for the inexact processes with different types of coupling, when starting with the vector (3.57). The 'exact process' does not change significantly for increasing values of $p$. For the inexact processes, the number of outer iterations increases when the number of subdomains increases (as expected). For the simple optimized coupling one can roughly say that convergence on $p$ subdomains requires $5 + p$ outer iterations. The strong optimized coupling needs about $1 - 2$ iterations less. But for the Neumann-Dirichlet coupling the results do not show such a linear relationship: when increasing from $2$ to $3$ or from $3$ to $4$ subdomains, the number of outer iterations almost doubles.

When we compare the right bottom part of Table 3.9 with the two subdomain case in Fig. 3.8, then we see what happens when the preconditioner is less accurate for Neumann-Dirichlet coupling: the exact Jacobi-Davidson process can not longer be reproduced. Because the shift $\theta_1$ in $\mathbf{M}_C$ is not equal to the shift $\theta$ in $\mathbf{B}_C$, the eigenvalues of the error propagator that were close to $\pm\sqrt{-1}$ (cf. §3.5.3) start to deviate. This results in worse circumstances for GMRES.

From these results we conclude that the optimized couplings outperform the Neumann-Dirichlet coupling for more than $2$ subdomains and a less accurate preconditioner

FIGURE 3.8. *Convergence history of Jacobi-Davidson applied to the discretized eigenvalue problem for the two-dimensional Laplace operator for accurate solutions to the correction equation and increasing values of $\omega_x$ and $n_x$ (solid lines) versus approximate solutions to the correction equation obtained from right preconditioned GMRES(3) with strong optimized (type 4) coupling (dashed lines with '$\circ$'), simple optimized (type 1) coupling (dash-dotted lines with '$\square$') and Neumann-Dirichlet coupling (dotted lines with '$*$') on an increasing number of subdomains. For explanation see §3.5.4.*

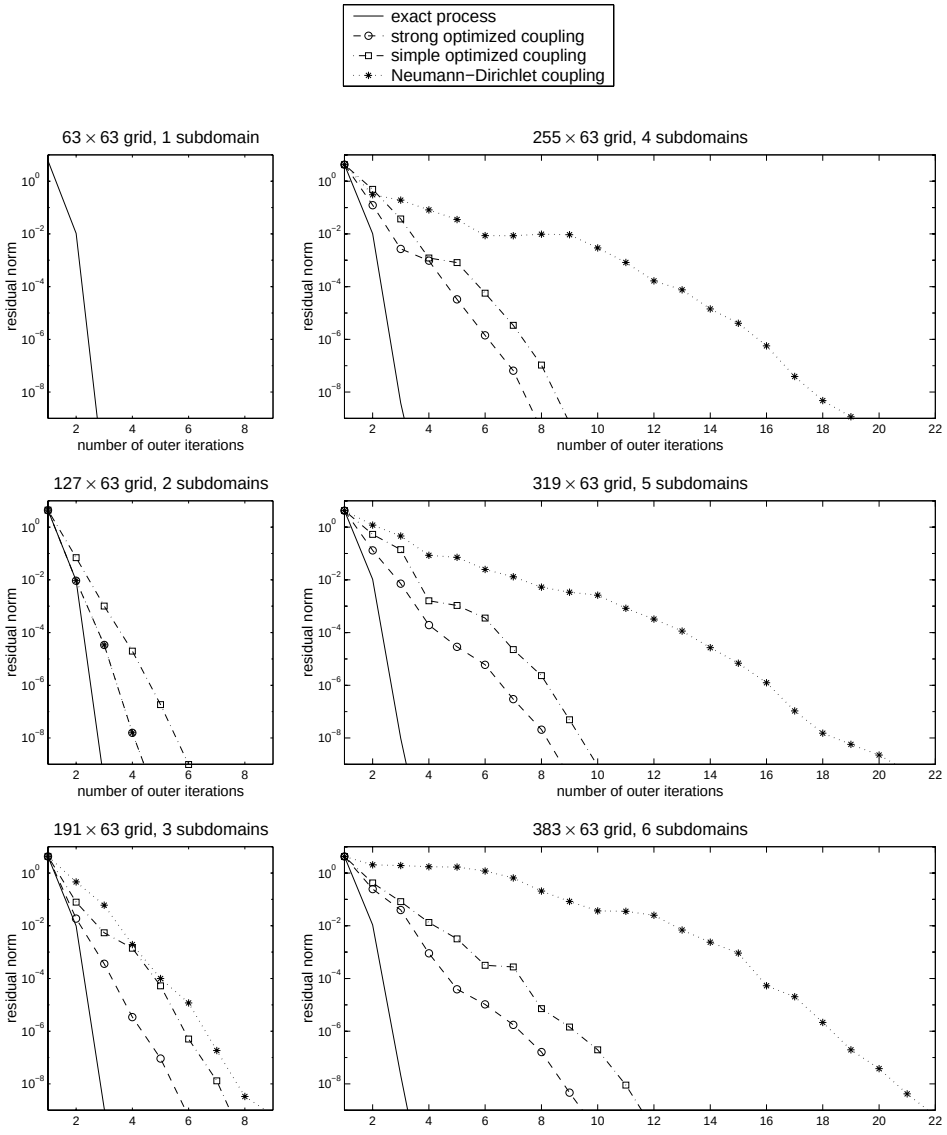
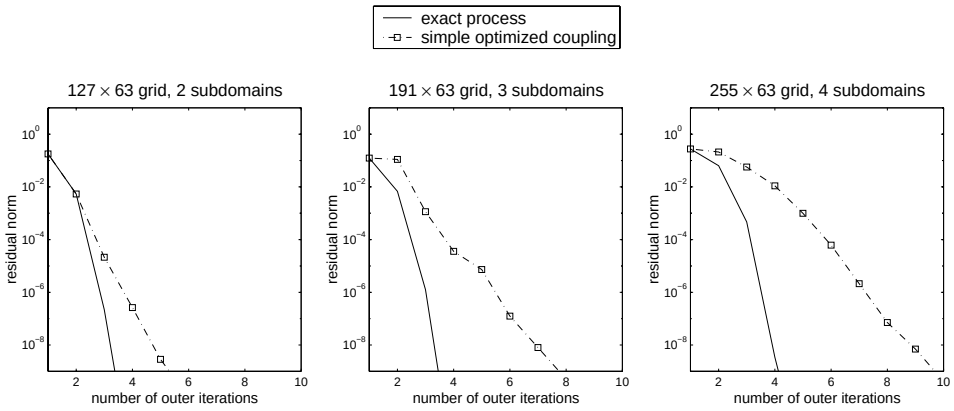


FIGURE 3.9. *Convergence history of Jacobi-Davidson applied to the discretized eigenvalue problem for the two-dimensional advection-diffusion operator (3.58) for accurate solutions to the correction equation and increasing values of $\omega_x$ and $n_x$ (solid lines) versus approximate solutions to the correction equation obtained from right preconditioned GMRES(3) with simple optimized (type 1) coupling (dash-dotted lines with '* $\square$*') on an increasing number of subdomains. For explanation see §3.5.4.*




So far we have only considered the eigenvalue problem for the Laplace operator. The analysis of §3.4 also accommodates problems with first order operators. To illustrate that this does not give essential differences, we consider

$$\frac{\partial^2}{\partial x^2} + \frac{2}{p}\frac{\partial}{\partial x} + \frac{\partial^2}{\partial y^2} + 5\frac{\partial}{\partial y} \tag{3.58}$$

on a domain with physical sizes $\omega_x = \frac{5}{4}p$ and $\omega_y = \frac{3}{4}$. Here $p \in \{2, 3, 4\}$ is the number of subdomains. With Jacobi-Davidson we compute the largest eigenvalue. In order to be in the convergence region of interest, Jacobi-Davidson is started with a vector equal to $(\mathbf{A} - 25\mathbf{I})^{-1}$ times the vector (3.57) (25 is close to the largest eigenvalue). All other settings are the same as in the previous experiment of this section.

Fig. 3.9 shows the convergence history of Jacobi-Davidson for accurate solutions and for approximate solutions of the correction equation. The approximate solutions are obtained from right preconditioned GMRES(3) with simple optimized (type 1) coupling. As in the previous experiment, the preconditioner is constructed only once at the first Jacobi-Davidson step. We see that the pictures in Fig. 3.9 are similar to those in Fig. 3.8.

## 3.6 Conclusions

In this chapter we have outlined and analyzed how a nonoverlapping domain decomposition technique can be incorporated in the Jacobi-Davidson method. For large eigenvalue problems the solution of correction equations may become too expensive in terms of CPU time or/and memory. Domain decomposition may be attractive in a parallel computing environment.

For a model eigenvalue problem with constant coefficients we have analyzed how the coupling equations should be tuned. By numerical experiments we have verified our analysis. Indeed, further experiments showed that tuning of the coupling results in faster convergence of the Jacobi-Davidson process.

In realistic problems, the coefficient functions will not be constant and the domain will have a complicated geometry. For the determination of suitable coupling matrices, we intend to locally apply the approach that we discussed here. This 'local' approach is one of the subjects of the next chapter.

# Chapter 4

# Domain decomposition for the Jacobi-Davidson method: practical strategies

**Abstract**

The Jacobi-Davidson method is an iterative method for the computation of solutions of large eigenvalue problems. In each iteration the (approximate) solution of a specific linear system is needed. In an chapter 3 we proposed a strategy based on a nonoverlapping domain decomposition technique for the computation of (approximate) solutions of such a linear system. That strategy was analysed for model problems with simple domains. In this chapter we discuss aspects that are relevant for eigenvalue problems with variable coefficients and more complicated domains.

**Keywords:** Eigenvalue problems, domain decomposition, Jacobi-Davidson, Schwarz method, nonoverlapping, iterative methods.

**2000 Mathematics Subject Classification:** 65F15, 65N25, 65N55.

## 4.1 Introduction

The Jacobi-Davidson method [46] is an iterative method for the computation of several selected eigenvalues and corresponding eigenvectors of large scale eigenvalue problems. The major part, in terms of computational work and memory requirements, of this method results from solving the so called correction equation. To reduce this work, we proposed in chapter 3 an approach for the computation of (approximate) solutions of the correction equation with a domain decomposition method for eigenvalue problems related to PDE's. The domain decomposition method, a nonoverlapping additive Schwarz method is based on work by Tang [57] and Tan & Borsboom [55, 56] for ordinary linear systems. Application of this domain decomposition method to the correction equation is not straightforward because the correction equation is not an ordinary linear system: the linear operator is shifted by an (approximate) eigenvalue and involves two projections. Also the tuning of the domain coupling parameters needs additional care. For the tuning of these parameters in §3.4 of chapter 3 we analysed a model eigenvalue problem: an advection-diffusion operator with constant coefficients.

Here we further refine the domain decomposition approach for Jacobi-Davidson. The point of view in this chapter is more practical than conceptual. It contains two major ingredients: the construction of a preconditioner in case of variable coefficients in the PDE (§4.3) and domain decomposition of more complicated geometries (§4.4). First, we give in §4.2 a recapitulation of the most relevant results in chapter 3.

## 4.2 Domain decomposition in Jacobi-Davidson

Here we recapitulate the incorporation of a domain decomposition method in the Jacobi-Davidson method, as proposed in chapter 3. The domain decomposition method is a nonoverlapping additive Schwarz method and based on work by Tang [57] and further generalized by Tan & Borsboom [55, 56]. Although in this section the two subdomain case is considered, it can easily be generalized to the case of more than two subdomains in one direction (see §3.5.4 of chapter 3), and it leads to obvious strategies for more general subdomain splittings as we will see in this chapter.

### 4.2.1 A nonoverlapping additive Schwarz method

Suppose we want to compute a solution of the linear system

$$\mathbf{B}\,\mathbf{y} = \mathbf{d}, \tag{4.1}$$

which originates from the discretization of some partial differential equation on a domain $\Omega$. Such a discretization typically results in a banded $\mathbf{B}$ and the unknowns $\mathbf{y}$ are coupled only locally. The domain $\Omega$ is decomposed into two nonoverlapping subdomains $\Omega_1$ and $\Omega_2$. $\Gamma$ is the interface between $\Omega_1$ and $\Omega_2$. By taking into account the band structure on the subgrids

that cover $\Omega_1$ and $\Omega_2$, we partition $\mathbf{B}$, $\mathbf{y}$, and $\mathbf{d}$, respectively, as

$$\begin{bmatrix} \mathbf{B}_{11} & \mathbf{B}_{1\ell} & \mathbf{B}_{1r} & \mathbf{0} \\ \mathbf{B}_{\ell 1} & B_{\ell\ell} & B_{\ell r} & \mathbf{0} \\ \mathbf{0} & B_{r\ell} & B_{rr} & \mathbf{B}_{r2} \\ \mathbf{0} & \mathbf{B}_{2\ell} & \mathbf{B}_{2r} & \mathbf{B}_{22} \end{bmatrix}, \quad \mathbf{y} \equiv \begin{bmatrix} \mathbf{y}_1 \\ y_\ell \\ y_r \\ \mathbf{y}_2 \end{bmatrix}, \quad \text{and} \quad \begin{bmatrix} \mathbf{d}_1 \\ d_\ell \\ d_r \\ \mathbf{d}_2 \end{bmatrix} \quad \text{respectively.}$$

The labels ($1$, $\ell$, $r$, and $2$ respectively) indicate the relation to the grid of the elements/operations ($\Omega_1$, left from $\Gamma$, right from $\Gamma$, and $\Omega_2$, respectively). For instance, $\mathbf{B}_{11}$ is the part of $\mathbf{B}$ restricted to the interior gridpoints of $\Omega_1$, $\mathbf{B}_{\ell\ell}$ is the part restricted to the gridpoints in $\Omega_1$ left from $\Gamma$, $\mathbf{B}_{\ell r}$ is the part that couples gridpoints in $\Omega_1$ left from $\Gamma$ to adjacent gridpoints in $\Omega_2$ right from $\Gamma$, etc. For these partitionings we define the corresponding *canonical enhancements* by

$$\mathbf{B}_C \equiv \begin{bmatrix} \mathbf{B}_{11} & \mathbf{B}_{1\ell} & \mathbf{B}_{1r} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{B}_{\ell 1} & B_{\ell\ell} & B_{\ell r} & 0 & 0 & \mathbf{0} \\ \mathbf{0} & C_{\ell\ell} & C_{\ell r} & -C_{\ell\ell} & -C_{\ell r} & \mathbf{0} \\ \mathbf{0} & -C_{r\ell} & -C_{rr} & C_{r\ell} & C_{rr} & \mathbf{0} \\ \mathbf{0} & 0 & 0 & B_{r\ell} & B_{rr} & \mathbf{B}_{r2} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{B}_{2\ell} & \mathbf{B}_{2r} & \mathbf{B}_{22} \end{bmatrix}, \quad \underset{\sim}{\mathbf{y}} \equiv \begin{bmatrix} \mathbf{y}_1 \\ y_\ell \\ \widetilde{y}_r \\ \widetilde{y}_\ell \\ y_r \\ \mathbf{y}_2 \end{bmatrix}, \text{ and } \quad \underline{\mathbf{d}} \equiv \begin{bmatrix} \mathbf{d}_1 \\ d_\ell \\ 0 \\ 0 \\ d_r \\ \mathbf{d}_2 \end{bmatrix}. \tag{4.2}$$

Here $C$ is a nonsingular interface coupling matrix:

$$C \equiv \begin{bmatrix} C_{\ell\ell} & C_{\ell r} \\ -C_{r\ell} & -C_{rr} \end{bmatrix}, \tag{4.3}$$

and $\widetilde{y}_\ell$ and $\widetilde{y}_r$ are extra unknowns.

With these enhancements the enhancement of linear system (4.1) is defined:

$$\mathbf{B}_C \, \underset{\sim}{\mathbf{y}} = \underline{\mathbf{d}}. \tag{4.4}$$

Note that the solution $\underset{\sim}{\mathbf{y}}$ of (4.4) yields the components of the solution $\mathbf{y}$ of (4.1). It will turn out that (4.4) lends itself more for parallel computing by tuning the $C$-matrices carefully.

The idea is to precondition the enhanced system (4.4) by performing accurate solves on the subdomains $\Omega_1$ and $\Omega_2$ (indicated by the framed parts in (4.2)) and to tune the coupling $C$ between the subdomains (outside the framed parts) to speed up the iteration process for the computation of solutions to (4.4). Hence, the matrix $\mathbf{B}_C$ is split as $\mathbf{B}_C = \mathbf{M}_C - \mathbf{N}$, where the preconditioner $\mathbf{M}_C$ is the framed part of $\mathbf{B}_C$ in (4.2). Approximate solutions to the preconditioned enhanced system

$$\mathbf{M}_C^{-1} \, \mathbf{B}_C \, \underset{\sim}{\mathbf{y}} = \mathbf{M}^{-1} \, \underline{\mathbf{d}} \tag{4.5}$$

will be computed with an iterative method. Observe that the iterates of such a method are linear combinations of powers of $\mathbf{M}_C^{-1} \, \mathbf{B}_C$. The tuning of the coupling $C$ is based on that

observation: performing subdomain solves with $\mathbf{M}_C$ of the matrix splitting $\mathbf{B}_C = \mathbf{M}_C - \mathbf{N}$ induces approximation errors, these errors are propagated in each iteration by the *error propagation matrix* $\mathbf{M}_C^{-1}\,\mathbf{N}$. The matrix $C$ is chosen so that these errors are damped out by higher powers of $\mathbf{M}_C^{-1}\,\mathbf{B}_C = \mathbf{I} - \mathbf{M}_C^{-1}\,\mathbf{N}$. For this tuning we use information from the equations that lead after discretization to $\mathbf{B}$. Tan and Borsboom [55, 56] constructed such a tuning for ordinary linear systems that originate from advection dominated problems.

### 4.2.2 Jacobi-Davidson and the correction equation

Now, we briefly summarize the Jacobi-Davidson method [46]. This method computes iteratively a solution for a (generalized) eigenvalue problem. We restrict ourselves to standard eigenvalue problems $\mathbf{A}\,\mathbf{x} = \lambda\,\mathbf{x}$. Each iteration step of Jacobi-Davidson consists of

1. the computation of an approximate solution $(\theta, \mathbf{u})$ to the wanted eigenpair $(\lambda, \mathbf{x})$ from a subspace via a Rayleigh-Ritz principle,

2. the computation of a vector $\mathbf{t}$ that corrects the approximate eigenvector $\mathbf{u}$ from a correction equation:

$$\mathbf{t} \perp \mathbf{u}, \qquad (\mathbf{I} - \mathbf{u}\mathbf{u}^*)\,(\mathbf{A} - \theta\,\mathbf{I})\,(\mathbf{I} - \mathbf{u}\mathbf{u}^*)\,\mathbf{t} = \mathbf{r} \quad \text{with} \quad \mathbf{r} \equiv \theta\,\mathbf{u} - \mathbf{A}\,\mathbf{u}, \tag{4.6}$$

3. the expansion of the subspace with $\mathbf{t}$.

The computation of (approximate) solutions of correction equation (4.6) involves most of the computational work of the Jacobi-Davidson method. This is the motivation for investigating the domain decomposition method, in an attempt to speed up parallel computation.

For that purpose, we showed in §3.3 of chapter 3 how to enhance the correction equation (4.6):

$$\underline{\mathbf{t}} \perp \underline{\mathbf{u}}, \quad (\mathbf{I} - \underline{\mathbf{u}}\,\underline{\mathbf{u}}^*)\,\mathbf{B}_C\,(\mathbf{I} - \underline{\mathbf{u}}\,\underline{\mathbf{u}}^*)\,\underline{\mathbf{t}} = \underline{\mathbf{r}}, \tag{4.7}$$

with $\mathbf{B} \equiv \mathbf{A} - \theta\,\mathbf{I}$, $\underline{\mathbf{u}} \equiv (\mathbf{u}_1^T, u_\ell^T, 0^T, 0^T, u_r^T, \mathbf{u}_2^T)^T$, and $\underline{\mathbf{r}} \equiv (\mathbf{r}_1^T, r_\ell^T, 0^T, 0^T, r_r^T, \mathbf{r}_2^T)^T$. Then a, for instance, left preconditioner

$$(\mathbf{I} - \underline{\mathbf{u}}\,\underline{\mathbf{u}}^*)\mathbf{M}_C(\mathbf{I} - \underline{\mathbf{u}}\,\underline{\mathbf{u}}^*)$$

can be incorporated as follows:

$$\mathbf{P}\,\mathbf{M}_C^{-1}\mathbf{B}_C\,\mathbf{P}\,\underline{\mathbf{t}} = \mathbf{P}\,\mathbf{M}_C^{-1}\underline{\mathbf{r}} \quad \text{with} \quad \mathbf{P} \equiv \mathbf{I} - \frac{\mathbf{M}_C^{-1}\underline{\mathbf{u}}\,\underline{\mathbf{u}}^*}{\underline{\mathbf{u}}^*\mathbf{M}_C^{-1}\underline{\mathbf{u}}}. \tag{4.8}$$

### 4.2.3 Tuning of the coupling for the correction equation

For a model eigenvalue problem of an advection-diffusion operator

$$\mathcal{L} = a\frac{\partial^2}{\partial x^2} + b\frac{\partial^2}{\partial y^2} + u\frac{\partial}{\partial x} + v\frac{\partial}{\partial y} + c, \tag{4.9}$$

with *constant* coefficients $a, b, u, v$, and $c$ we proposed efficient coupling parameters for the correction equation in §3.4 of chapter 3. We recollect the main results here. For specific details we refer to the original paper.

The operator (4.9) is assumed to be discretized over a rectangular grid over a rectangular domain by, say, standard 5-point discretization stencils. The discretized operator is expressed as a sum of tensor products of one-dimensional operators:

$$L = L_x \otimes \mathbf{I} + \mathbf{I} \otimes L_y.$$

We exploit exact information about the eigenvectors of the operator $L_y$ in the direction parallel to the interface. Perpendicular to the interface, essentially two different types of behavior can be distinguished: globally harmonic and locally dominating behavior. By approximating the dominating behavior with the dominants, specific knowledge on the number of gridpoints in the direction perpendicular to the interface is avoided. This leads to very useful coupling parameters.

The coupling parameters are expressed by the components of the interface coupling matrix $C$ (4.3):

$$C_{\ell\ell} = C_{rr} = 1 + \gamma L_y \quad \text{and} \quad C_{\ell r} = C_{r\ell} = \alpha + \beta L_y. \tag{4.10}$$

In §3.4.4 of chapter 3 different types of optimized coupling are distinguished, depending on which of the coupling parameters $\alpha$, $\beta$, and $\gamma$ are used for the optimization. Coupling with parameter $\alpha$ only is referred to as simple optimized coupling, and combinations of $\alpha$ with $\beta$ and/or $\gamma$ as stronger optimized couplings.

For all these types of optimized coupling, the procedure to determine the collection $\widehat{E}$ ((3.48) in §3.4.4 of chapter 3) that corresponds to the modes with dominated behavior is the same. This determination consists of the computation of the extremal values of $\widehat{E}$: the lower bound $\mu$ and the upper bound $M$. For the operator (4.9) with constant coefficients, the parameter $\mu$ depends amongst others on the shift $\theta$ and turns out to be a critical parameter (see §3.5.2.1 in chapter 3), whereas $M$ only depends on the mesh widths and the coefficients (see the expression for $M$ in §3.4.4 of chapter 3). The lower bound $\mu$ is computed by means of the integer value $l_e$. This $l_e$ is in the range of all possible wave numbers $l_y$ of the eigenvalue problem in the $y$-direction and marks the subdivision in harmonic and dominated behavior of the eigenvectors of the error propagation matrix in the $x$-direction.

## 4.3 Variable coefficients

In this section we outline a strategy for the eigenvalue problem associated with a convection-diffusion operator with variable coefficients. The main idea is based on the assumption that variable coefficients can be approximated locally by frozen coefficients. With this strategy we construct preconditioners for some relevant examples. For these cases the spectrum of the resulting error reduction matrix is also computed in order to show the effects of the strategy.

### 4.3.1 Frozen coefficients

In this section the two-dimensional advection-diffusion operator (4.9) will have *variable* coefficients $a = a(x, y)$, $b = b(x, y)$, $u = u(x, y)$, $v = v(x, y)$, and $c = c(x, y)$ for $(x, y) \in \Omega$. We will show how the preconditioner $\mathbf{M}_C$ can be constructed for this case.

First observe that, globally, two parts can be distinguished in the preconditioner $\mathbf{M}_C$:

- a subblock $\mathbf{M}_p$ that describes the local subproblem on subdomain $\Omega_p$,
- the coupling $C$ between the subdomains/subproblems that consists of the coupling parameters.

The coupling parameters are optimized with respect to those eigenmodes of the error propagation matrix associated with locally dominating behavior. These eigenmodes reduce exponentially when moving away from the interface. Therefore, effective coupling parameters can be interpreted (§3.2.5 of chapter 3) as those values for which the subproblems are decoupled "as much as possible". So, with respect to the modes with dominating behavior, a local phenomenon on subdomain $\Omega_p$ is expected to be captured well by subblock $\mathbf{M}_p$ with effective coupling parameters and these modes are relevant over a small area near the interface only.

This observation motivates the following strategy of *frozen* coefficients, here the $x$-direction ($y$-direction) refers to the direction perpendicular (parallel) to the interface:

- For each grid point in the $y$-direction we consider the values of the coefficients locally near the interface.
- With these values of the coefficients, we compute effective coupling parameters for a problem with appropriate constant coefficients.
- These computed coupling parameters are used as the local coupling parameters for the problem with variable coefficients.

Although, for simplicity, we consider the simple coupling with $\alpha$ (4.10) (§3.4.4 of chapter 3) only, the strategy can also be applied for stronger couplings. For variable coefficients we can estimate $l_e$ only roughly, therefore we allow a continuous $l_e$ here: we compute its values by $l_e \equiv \widetilde{l}_e + 1$ instead of $l_e \equiv \lfloor \widetilde{l}_e \rfloor + 1$ as in §3.4.4 of chapter 3.

We will now describe several illustrative numerical experiments. The experiments are performed in MATLAB 5.3.

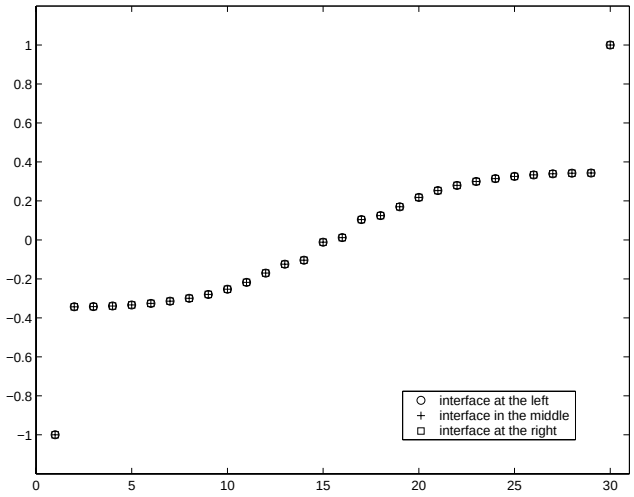
### 4.3.2 Numerical experiments

We will illustrate the construction of the preconditioner $\mathbf{M}$, with the strategy formulated in §4.3.1, by some numerical experiments. We do this for two subdomains, similar to the constant coefficients problem. Effective coupling parameters for more subdomains can be derived from the two subdomain case. Recall that, the $x$-direction ($y$-direction) refers to the direction perpendicular (parallel) to the interface.

The eigenvalue problem $\mathcal{L}\,\widehat{\varphi} = \widehat{\lambda}\,\widehat{\varphi}$, for the operator defined by (4.9), with variable coefficients and Dirichlet conditions on the external boundary of $\Omega$, gives rise to a matrix $\mathbf{A}$ with eigenvalue $\lambda$ after discretization. We consider the correction equation in the asymptotic case, i.e. we take for $\theta$ the exact eigenvalue $\lambda$ in the correction equation. So a preconditioner $\mathbf{M}$ will be constructed for the operator $\mathbf{B}_C$ where $\mathbf{B} \equiv \mathbf{A} - \lambda\mathbf{I}$. After construction of the preconditioner we will investigate its effectiveness by computing the nonzero eigenvalues of the error propagation matrix $\mathbf{I} - \mathbf{M}^{-1}\mathbf{B}_C$.

First we investigate the dependency of the preconditioner on the position of the interface (§4.3.2.1). This is motivated by the fact that for constant coefficients the preconditioner does not depend on the position of the interface $\Gamma$. Then we do some numerical experiments with coefficients that are typical for practical situations: coefficients that behave (locally) exponential (§4.3.2.2), harmonical (§4.3.2.3), and with (locally) a large jump (§4.3.2.4).

FIGURE 4.1. *Nonzero eigenvalues of the error propagation matrix for the eigenvalue problem of situation A, the constant coefficients case, in §4.3.2.1. Shown are the eigenvalues for three different configurations of the two subdomains: interface at the left ('$\circ$'), interface in the middle ('$+$') and interface at the right ('$\square$'). The eigenvalues are sorted ($x$-axis) by order of magnitude ($y$-axis).*




#### 4.3.2.1 Small jump coefficient perpendicular to interface

First we investigate whether the preconditioner depends much on the position of the interface for a modest variation in the coefficient in the $x$-direction. For that purpose we consider the following two situations on $\Omega \equiv (0,4) \times (0,1)$:

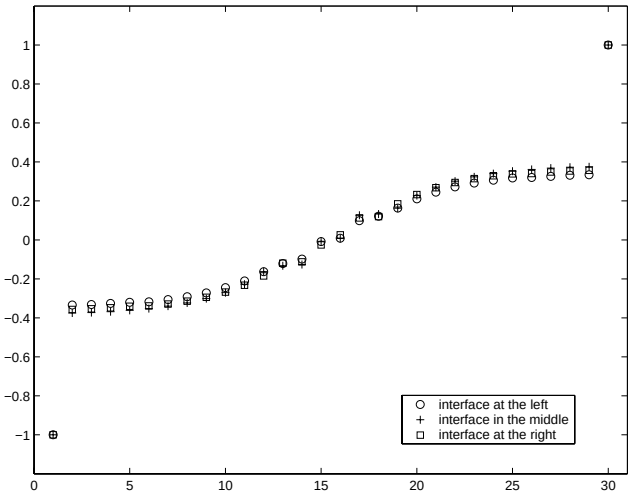
- situation A:
$$\mathcal{L} = \frac{3}{2}\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right),$$
- situation B:
$$\mathcal{L} = \begin{cases} \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} & \text{for} \quad 0 < x < 2 \\ 2\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right) & \text{for} \quad 2 \leq x < 4 \end{cases},$$

and we are interested in the largest eigenvalue of $\mathcal{L}$.

Domain $\Omega$ is covered by a $60 \times 15$ grid and is decomposed in $\Omega_1$ and $\Omega_2$ in three different ways:

- configuration with interface more leftwards:
$$\Omega_1 = (0, 1\frac{1}{3}) \times (0,1) \quad \text{and} \quad \Omega_2 = (1\frac{1}{3}, 4) \times (0,1),$$

FIGURE 4.2. *Nonzero eigenvalues of the error propagation matrix for the eigenvalue problem of situation B, the case with a modest jump in the coefficient in the $x$-direction, in §4.3.2.1. Shown are the eigenvalues for three different configurations of the two subdomains: interface at the left ('$\circ$'), interface in the middle ('+') and interface at the right ('$\square$'). The eigenvalues are sorted ($x$-axis) by order of magnitude ($y$-axis).*




- configuration with interface in the middle:

$$\Omega_1 = (0,2) \times (0,1) \quad \text{and} \quad \Omega_2 = (2,4) \times (0,1),$$

  here the interface is located precisely at the jump of the $x$-coefficient,

- configuration with interface more to the right:

$$\Omega_1 = (0,2\frac{2}{3}) \times (0,1) \quad \text{and} \quad \Omega_2 = (2\frac{2}{3},4) \times (0,1).$$

For the constant coefficients situation A it is known (§3.4.4 of chapter 3) that the eigenvalues of the error propagation matrix do not depend much on the location of the interface, i.e. for this two subdomain case they are virtually independend of $n_{x1}$. Therefore all three configurations will yield practically the same spectrum. This is confirmed by the results of the numerical experiment, as is shown in Fig. 4.1.

Situation B has a jump in the coefficient in the $x$-direction and we can not predict what will happen. Application of the strategy in §4.3.1 leads to the following coupling parameters: $\alpha_\ell = \alpha_r = -2.15$ (left configuration), $\alpha_\ell = -2.15$ and $\alpha_r = -2.07$ (middle configuration), and $\alpha_\ell = \alpha_r = -2.07$ (right configuration). Obviously, the values of the $\alpha$'s do not vary much. For these values the preconditioners are constructed. The corresponding spectra of $\mathbf{I} - \mathbf{M}^{-1}\,\mathbf{B}_C$ are shown in Fig. 4.2. From this we conclude that also these spectra do not differ much: they almost behave as for a constant coefficient problem. Apparently, the position of the interface does not play a role of importance for the construction of an optimal preconditioner for a modest variable coefficient in the $x$-direction.

#### 4.3.2.2 Exponential coefficient parallel to interface

Now, we focus on the $y$-direction. We start with

$$\mathcal{L} = \frac{\partial^2}{\partial x^2} + \frac{\partial}{\partial y}[e^y \frac{\partial}{\partial y}], \tag{4.11}$$

defined on $\Omega = (0,2)\times(0,1)$. We are again interested in the largest eigenvalue. The domain is covered by a $31\times31$ grid, and decomposed into two equal subdomains $\Omega_1 = (0,1)\times(0,1)$ and $\Omega_2 = (1,2)\times(0,1)$.

The coefficient $e^y$ in the $y$-direction varies between $1$ and $e$, with average $\int_0^1 e^y dy \approx 1.71$. If our strategy (§4.3.1) is applied to this problem, then the coupling parameters $\alpha$ vary between $-2.39$ and $-2.93$. We compare this local strategy with two other approaches:

- a preconditioner $\mathbf{M}$ with semi-local $\alpha$'s obtained by first averaging a cluster of 5 successive coefficients in the discretized $y$-direction, with $\alpha$ based on an average, and putting this $\alpha$ on the 5 corresponding positions in the preconditioner,
- a preconditioner $\mathbf{M}$ with fixed $\alpha = -2.66$. This is the coupling parameter that corresponds to the average coefficient $1.71$. If we take this $\alpha = -2.66$ for all $0 < y < 1$ then we apply some global optimization strategy: we approach the variable coefficient problem with a constant coefficient problem.

The nonzero eigenvalues of the error propagation matrices corresponding to the preconditioners of the three approaches are shown in Fig 4.3. We do not observe significant differences between the three approaches. Obviously, fluctuations in the $y$-coefficient are not large enough so that a more global strategy is also applicable.

FIGURE 4.3. *Nonzero eigenvalues of the error propagation matrix for the eigenvalue problem with modest exponential coefficient in the $y$-direction from §4.3.2.2. Shown are the eigenvalues for three different approaches in the construction of the preconditioner: local optimized coupling parameters ('$\circ$'), semi-local coupling parameters ('$\square$') and a fixed coupling parameter ('$+$'). The eigenvalues are sorted ($x$-axis) by order of magnitude ($y$-axis).*

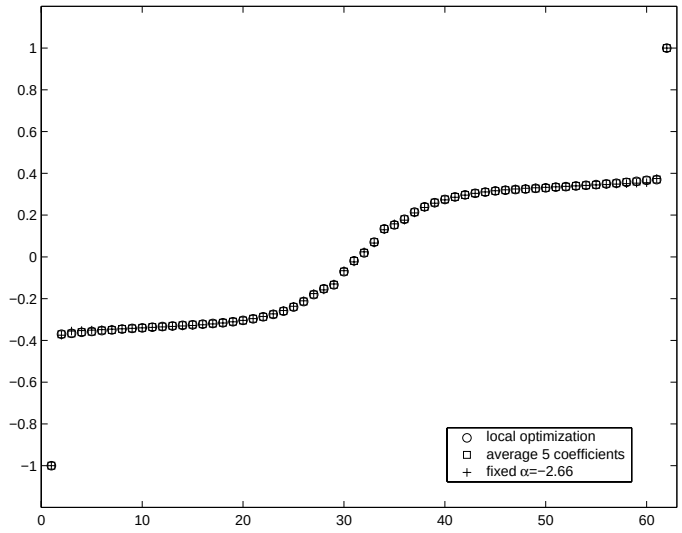
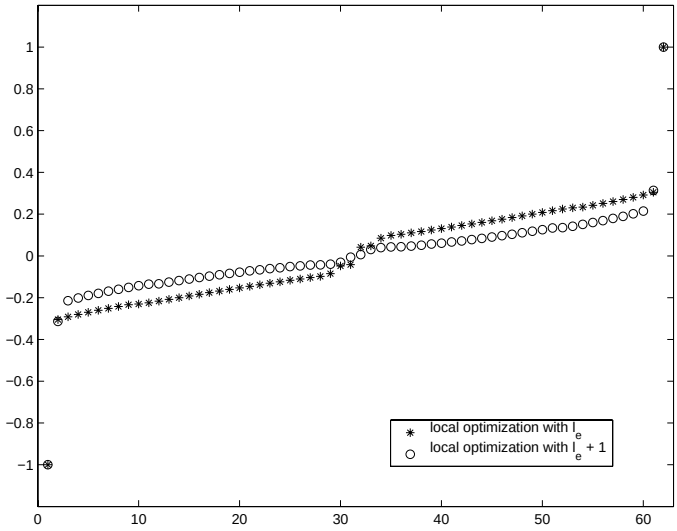


FIGURE 4.4. *Nonzero eigenvalues of the error propagation matrix for the eigenvalue problem with highly varying exponential coefficient in the $y$-direction from §4.3.2.2. Shown are the eigenvalues for the local optimization strategy with $l_e$ ('$*$') and the local optimization strategy with $l_e+1$ ('$\circ$'). The eigenvalues are sorted ($x$-axis) by order of magnitude ($y$-axis).*




The last observation motivates us to blow up the coefficient in the $y$-direction:

$$\mathcal{L} = \frac{\partial^2}{\partial x^2} + \frac{\partial}{\partial y}[e^{4y}\frac{\partial}{\partial y}]. \tag{4.12}$$

(The (sub)domain(s) and (sub)grid(s) remain unchanged, we still focus on the largest eigenvalue.)

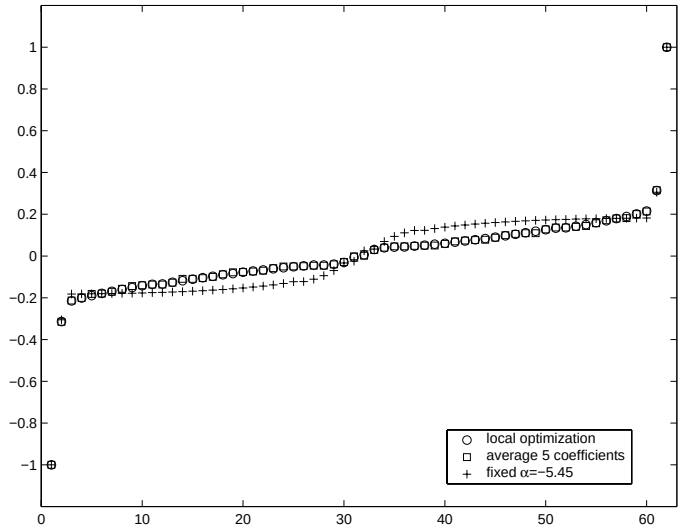
For $0 \leq y \leq 1$ the value of $e^{4y}$ varies between 1 and $e^4 \approx 54.6$ with average $\int_0^1 e^{4y}dy \approx 12.64$. If we apply our local optimization strategy then the value of

$$l_e = \frac{2}{\pi}(n_y+1)\arcsin\left(\frac{h_y}{2}\sqrt{-\theta e^{-4y}}\right) + 1$$

(cf. §3.4.4 of chapter 3) is less than 2 for 16 of the 31 grid points in the $y$-direction as $\theta \approx -67.20$. Since the mode $l_y = 1$ corresponds to the desired eigenvector, this mode should not be included for the optimization. Therefore, to avoid values lower than 2, we do the optimization for $l_e + 1$. Fig. 4.4 illustrates that indeed this results in a better spectrum for $\mathbf{I} - \mathbf{M}^{-1}\mathbf{B}_C$.

With the optimization for $l_e + 1$, the computed coupling parameter $\alpha$ varies between $-3.64$ and $-23.78$; the coupling parameter that corresponds to the average coefficient $12.64$ is equal to $-5.45$. We constructed also the preconditioners with semi-local $\alpha$ (again optimization for $l_e + 1$) and a constant $\alpha = -5.45$. The corresponding nonzero eigenvalues of the error propagation matrix are shown in Fig. 4.5.

FIGURE 4.5. *Nonzero eigenvalues of the error propagation matrix for the eigenvalue problem with highly varying exponential coefficient in the $y$-direction from §4.3.2.2. Shown are the eigenvalues for three different approaches in the construction of the preconditioner: local optimized coupling parameters ('$\circ$'), semi-local coupling parameters ('$\square$') and a fixed coupling parameter ('$+$'). The eigenvalues are sorted ($x$-axis) by order of magnitude ($y$-axis).*




If we compare the three plots in this figure then we see that for the constant $\alpha$ the eigenvalues at positions $6, 7, \ldots, 29$ and $34, 35, \ldots, 56$ are somewhat less clustered near the origin. We believe that this is mainly due to the local phenomena in the $y$-direction to which the (semi-) local coupling parameters adapt better. On the other hand, for the constant $\alpha$ the outliers at positions $2, 3, 58$, and $59$ seem to be a little bit better.

From this problem we learn that for the determination of coupling parameters one should monitor $l_e$. This value may become too close to a critical value. Increasing $l_e$ in such a case may improve the spectrum of the error propagation matrix.

#### 4.3.2.3 Harmonic first and second order coefficients

For an operator with harmonic coefficients we adapted a linear problem from [8]. Because we do not want interference of effects due to the discretization, we decreased the periods of both second and first order coefficients and the magnitudes of the first order coefficients of the original operator in [8]. In this way we obtain

$$\begin{aligned}\mathcal{L} \quad \equiv \quad & \frac{\partial}{\partial x}[(1+\frac{1}{2}\sin(3\pi x))\frac{\partial}{\partial x}] - (5\sin(2\pi x)\cos(2\pi y))\frac{\partial}{\partial x} + \\ & \frac{\partial}{\partial y}[(1+\frac{1}{2}\sin(3\pi x)\sin(3\pi y))\frac{\partial}{\partial y}] + (5\cos(2\pi x)\sin(2\pi y))\frac{\partial}{\partial y}. \quad (4.13)\end{aligned}$$

The four largest eigenvalues of the operator in (4.13), after discretization ($n_x = n_y = 31$), are

$$\lambda_1 \approx -23.109,\ \ \lambda_2 \approx -47.2073,\ \ \lambda_3 \approx -52.6884 \text{ and } \lambda_4 \approx -73.7715\ .$$

The domain is decomposed into two equal subdomains with physical sizes $[0, 0.5] \times [0, 1]$ and $[0.5, 1] \times [0, 1]$.

First we consider $\theta$ equal to the largest eigenvalue $\lambda_1$. If we apply the local optimization strategy then $l_e$ varies between 2 and 3. This is what we expect: the mode $l_y = 1$ corresponds to $\lambda_1$ and should not be taken into account. The nonzero eigenvalues of the error propagation matrix $\mathbf{I} - \mathbf{M}^{-1}\mathbf{B}_C$ for the resulting preconditioner $\mathbf{M}$ are shown in Fig. 4.6. Indeed, the eigenvectors of the error propagation matrix are damped for all values of $l_y$ larger than 1.

Next, we consider $\theta = \lambda_4$. For the operator with $c = -70$ in [8, §4.4] and [54, §5.5] it was reported that this is a more difficult problem than with $c = -20$. This is because the shift is more in the interior of the spectrum.

Inspection of the eigenvector corresponding to $\lambda_4$ indicates that it globally behaves like the eigenvector of $\lambda^{(2,2)}$ (i.e. $l_x = 2$ and $l_y = 2$) of the Laplace operator with constant coefficients. Therefore, we expect that for this case also $l_y = 2$ should not be taken into account for the local tuning. This is roughly confirmed by the computed values of $l_e$: $3 < l_e < 5$, for $0 < y < 1$. Fig. 4.7 shows the spectrum of the error propagation matrix. Note that, in contrast to the others, which are real, the values at positions 2, 3, 60, and 61 on the horizontal axis represent absolute values of two pairs of complex conjugate eigenvalues. They correspond to the modes $l_y = 2$ and $l_y = 3$ that were not taken into account for the optimization. But also with these modes included, the resulting spectrum is quite attractive for iteration purposes.

FIGURE 4.6. *Nonzero eigenvalues of the error propagation matrix with $\theta = \lambda_1$ for the eigenvalue problem with harmonic first and second order coefficients from §4.3.2.3. The eigenvalues are sorted ($x$-axis) by order of magnitude ($y$-axis).*

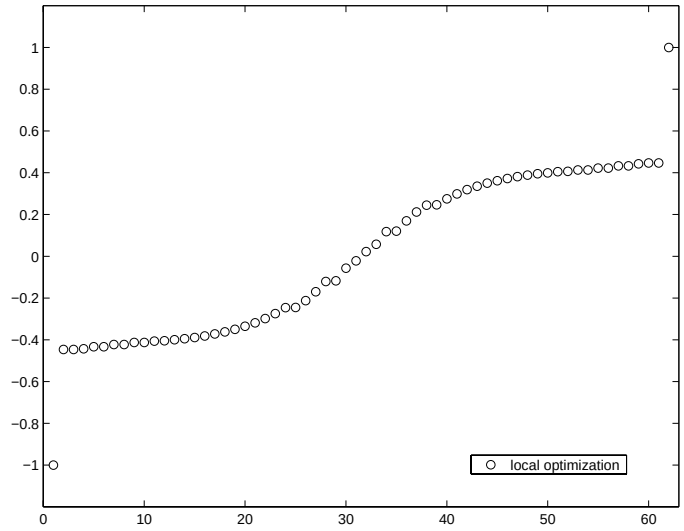



FIGURE 4.7. *Nonzero eigenvalues of the error propagation matrix with $\theta = \lambda_4$ for the eigenvalue problem with harmonic first and second order coefficients from §4.3.2.3. The eigenvalues are sorted ($x$-axis) by order of magnitude ($y$-axis).*

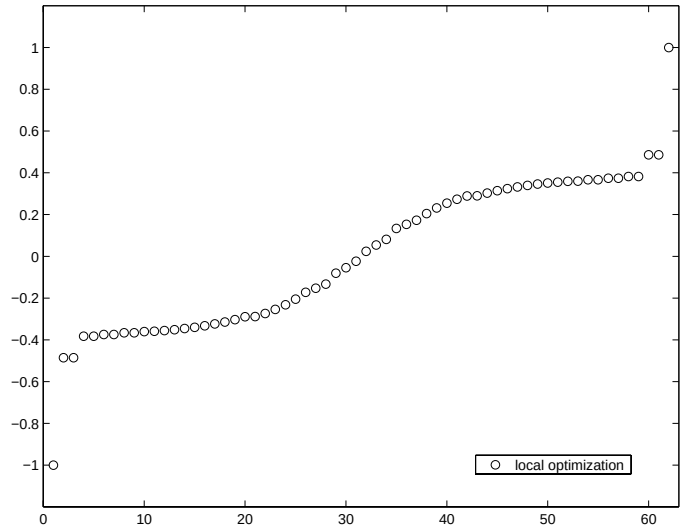
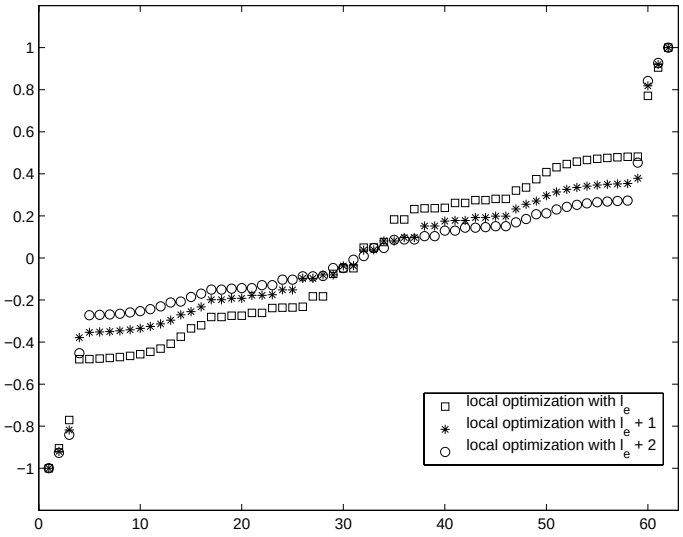


FIGURE 4.8. *Nonzero eigenvalues of the error propagation matrix for the eigenvalue problem with large jump in the coefficients from §4.3.2.4. Shown are the eigenvalues for the local optimization strategy with $l_e$ ('$\square$'), the local optimization strategy with $l_e + 1$ ('$*$') and the local optimization strategy with $l_e + 2$ ('$\circ$'). The eigenvalues are sorted ($x$-axis) by order of magnitude ($y$-axis).*




#### 4.3.2.4 A large jump in coefficient

In practical situations one has often to deal with an operator that has a large jump in its coefficients, as, for example, when modeling oil reservoirs (see, e.g. [60] where the coefficients jump from $1^{-7}$ to 1) with high-permeability in the physical domain adjacent to locations with low-permeability. We consider the following problem:

$$\mathcal{L} \equiv \frac{\partial}{\partial x}[c(y)\frac{\partial}{\partial x}] + \frac{\partial}{\partial y}[c(y)\frac{\partial}{\partial y}] \text{ with } c(y) = \begin{cases} 1 \text{ for } 0 \leq y < 0.25 & \text{(region 1)} \\ 1000 \text{ for } 0.25 \leq y < 0.75 & \text{(region 2)} \\ 1 \text{ for } 0.75 \leq y \leq 1 & \text{(region 3)} \end{cases} \tag{4.14}$$

defined on $[0, 2] \times [0, 1]$. We focus on the largest eigenvalue of this operator, the corresponding eigenvector is the most smooth one among all eigenvectors. The domain is decomposed into two equal subdomains with physical sizes $[0, 1] \times [0, 1]$ and $[1, 2] \times [0, 1]$.

Based on our strategy §4.3.1 we constructed a preconditioner. For the tuning also the parameter $l_e$ shows a sharp contrast on the different regions: $l_e = 5.0040$ on region 1 and 3 and $l_e = 1.1258$ on region 2. Although $l_e = 1.1258$ should not be included for the tuning because this value is close to the mode $l_y = 1$ that correponds to the desired eigenvector, we will not skip this case here, in order to see what will happen. We propose to optimize for $l_e + 1$ instead of $l_e$. In addition we also computed the coupling parameters locally for $l_e + 3$, just to see how critical this parameter is.

Fig. 4.8 shows the computed nonzero eigenvalues of the error propagation matrix for these three cases. From this we observe that 6 eigenvalues (at the positions $1, 2, 3, 60, 61, 62$ on the horizontal axis, they correspond to $l_y = 1, 2, 3$) can not be damped properly. The mode $l_y = 1$ corresponds to the desired eigenpair, hence the corresponding eigenvalues at position $1$ and $62$ will be removed by the projections in the correction equation (see §3.5.2.2 of chapter 3). We believe that the large values of the eigenvalues at position $2$, $3$, $60$, and $61$ are due to the discrepancy between the value of $l_e$ on region 2 and the other two regions. Apart from these 6 outliers, the remaining eigenvalues seem not to be so good when optimizing with $l_e$, as expected from the critical value of $l_e$ on region 2. From the eigenvalues at positions $4$ and $59$ on the horizontal axis we see that tuning with $l_e + 1$ should be preferred indeed.

## 4.4 Geometry

We now focus our attention to the geometry of the domain on which operator (4.9) is defined. The model analysis in chapter 3 was performed for a rectangular domain that was decomposed into two subdomain. In §3.5.4 of chapter 3 it was shown that the resulting coupling parameters can also be applied to decompositions with more than two subdomains in one direction.

In this section we will consider the effects for domain decomposition in two directions. We do this for an operator (4.9) with constant coefficients. First, we investigate in §4.4.1 whether an additional coupling mechanism is needed for subdomains that have only one cornerpoint in common (cross coupling). Then we discuss in §4.4.2 how domains that are a composition of rectangular subdomains can be treated. For nonrectangular compositions some options are left for the determination of coupling parameters, these options are briefly investigated in §4.4.2.1. We conclude this section with a more complicated domain, just to illustrate the capabilities of the approach (§4.4.2.2).

### 4.4.1 Cross coupling

With a numerical experiment we want to investigate whether an additional cross coupling is required between subdomains that have only one cornerpoint in common. For that purpose we consider a preconditioner $\mathbf{M}_C$ that is based on a decomposition of the domain $\Omega = [0, 2] \times [0, 2]$ in four subdomains in one direction (indicated by decomposition 1) and a preconditioner that is based on a decomposition of the same domain $\Omega$ in two directions (decomposition 2). For decomposition B the domain is split in the $x$-direction and in the $y$-direction, which results in $4$ subdomains separated by one interface in the $x$-direction and one in the $y$-direction. We compare the Jacobi-Davidson process for decomposition 1 with the Jacobi-Davidson process for decomposition 2 without cross coupling. Such a coupling could be worth further investigation if decomposition 1 would result in a significant faster overall process.

TABLE 4.1. *Convergence history of Jacobi-Davidson when applied to the discretized ($n_x = n_y = 63$ top table, $n_x = n_y = 127$ bottom table) eigenvalue problem of the two-dimensional ($b_x = b_y = 2$) Laplace operator for approximate solutions to the correction equation obtained with 4 right preconditioned GMRES iterations. For the construction of the preconditioner with simple optimized coupling the domain is decomposed into four subdomains in two different ways: decomposition 1 in the $x$-direction only ($p_x = 4$ and $p_y = 1$, on the left) and decomposition 2 in both directions ($p_x = 2$ and $p_y = 2$, on the right). For explanation see §4.4.1.*

| | decomposition 1 | | decomposition 2 | |
|---|---|---|---|---|
| step | $\theta - \lambda$ | $\|\mathbf{r}\|_2$ | $\theta - \lambda$ | $\|\mathbf{r}\|_2$ |
| $\mathbf{n_x = n_y = 63}$ | | | | |
| 1 | -6.50e-02 | 1.51e+00 | -6.50e-02 | 1.51e+00 |
| 2 | -2.43e-05 | 1.43e-01 | -5.45e-06 | 6.40e-02 |
| 3 | -2.64e-07 | 1.62e-02 | -3.21e-07 | 2.53e-02 |
| 4 | -2.84e-08 | 7.33e-03 | -3.78e-09 | 2.59e-03 |
| 5 | -6.82e-11 | 3.03e-04 | -1.46e-10 | 5.84e-04 |
| 6 | -2.20e-12 | 6.40e-05 | -3.91e-13 | 3.57e-05 |
| 7 | -3.38e-14 | 2.22e-06 | -4.44e-14 | 1.68e-06 |
| 8 | 1.95e-14 | 7.44e-08 | -1.14e-13 | 1.97e-07 |
| 9 | -1.85e-13 | 1.77e-09 | 9.50e-14 | 1.71e-08 |
| 10 | -9.95e-14 | 9.57e-11 | -1.03e-13 | 2.16e-09 |
| $\mathbf{n_x = n_y = 127}$ | | | | |
| 1 | -6.51e-02 | 1.54e+00 | -6.51e-02 | 1.54e+00 |
| 2 | -5.15e-05 | 5.06e-01 | -1.82e-05 | 1.68e-01 |
| 3 | -3.80e-06 | 1.48e-01 | -6.82e-07 | 6.64e-02 |
| 4 | -3.52e-08 | 1.30e-02 | -7.63e-08 | 2.61e-02 |
| 5 | -3.39e-10 | 1.58e-03 | -7.24e-09 | 3.08e-03 |
| 6 | -1.24e-11 | 1.91e-04 | -2.67e-10 | 1.37e-03 |
| 7 | -6.66e-12 | 2.03e-04 | -4.71e-11 | 6.52e-04 |
| 8 | -1.48e-12 | 7.23e-05 | -8.24e-13 | 3.91e-05 |
| 9 | -7.23e-13 | 5.69e-06 | -7.28e-13 | 2.86e-06 |
| 10 | -5.47e-13 | 5.32e-07 | -3.98e-13 | 7.66e-07 |

We aim at the largest eigenvalue $\lambda$ and corresponding eigenvector $\widehat{\varphi}$ of the Laplace operator on $\Omega$ with Dirichlet boundary condition $\widehat{\varphi} = 0$ on the external boundary $\partial\Omega$. The external boundary is located at grid positions $i = 0, i = n_x + 1, j = 0$, and $j = n_y + 1$ (left picture in Fig. 4.10). The boundary conditions on these points do not lead to a contribution to the discretized operator: the operator is defined on the "internal" points $i = 1, \ldots, n_x$ and $j = 1, \ldots, n_y$ only. We run the experiment for two grids: one with $n_x = n_y = 63$ and one with $n_x = n_y = 127$, the meshwidth is taken constant. The subgrids of decomposition 1 ($p_x = 4, p_y = 1$) have dimensions $n_{x1} = 15, n_{x2} = n_{x3} = n_{x4} = 16$, and $n_y = 63$ ($n_{x1} = 31, n_{x2} = n_{x3} = n_{x4} = 32$, and $n_y = 127$, respectively). For decomposition 2 ($p_x = 2, p_y = 2$), the numbers are $n_{x1} = n_{y1} = 31$ ($n_{x1} = n_{y1} = 63$, respectively), and $n_{x2} = n_{y2} = 32$ ($n_{x2} = n_{y2} = 64$, respectively).

The startvector of Jacobi-Davidson for both decompositions is the vector

$$\{(\frac{j_x}{n_x+1}(1-\frac{j_x}{n_x+1}), \frac{j_y}{n_y+1}(1-\frac{j_y}{n_y+1})) \mid 1 \le j_x \le n_x, 1 \le j_y \le n_y\}, \quad (4.15)$$

Approximate solutions to the correction equation of Jacobi-Davidson are computed by 4 steps of right preconditioned GMRES (see §3.3.3 of chapter 3), compared to left preconditioning this has the advantage of unknowns which are defined only on the grid points that correspond to the $\underset{\sim}{\cdot}_\ell$ and $\underset{\sim}{\cdot}_r$ parts. The preconditioner based on domain decomposition is constructed with simple optimized coupling (§3.4.4 of chapter 3). For the determination of coupling parameters on each interface we act as if there is no other interface. Both decompositions require about the same amount of computational work per Jacobi-Davidson step.

The experiment is performed in MATLAB 5.3.

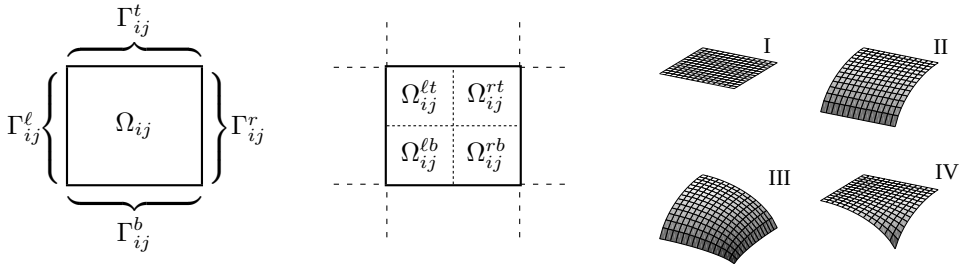
Table 4.1 shows the results of the experiment. For both grids ($n_x = n_y = 63$ and $n_x = n_y = 127$) we see that the Jacobi-Davidson processes for decomposition 1 and decomposition 2 vary only a little in convergence behavior and it can not be concluded that the process for decomposition 1 is significantly faster. For example, for $n_x = n_y = 63$ the error in the eigenvalue and the norm of the residual at step 6 is smaller for decomposition B, for $n_x = n_y = 127$ this happens at step 8. At other steps the situation is reversed. We believe these fluctuations are attributed to the different decompositions.

From this experiment we conclude that for the computation of a global solution (i.e. a solution that posseses global behavior all over the domain $\Omega$) to the eigenvalue problem, application of domain decomposition in two directions without cross coupling may lead to a satisfactory preconditioner.

### 4.4.2 Composition of rectangular subdomains

It seems that cross coupling is not necessary for a decomposition into two directions. Hence we proceed with domains which are composed of rectangular subdomains. Here we shall show how one can deal with these geometries.

FIGURE 4.9. *Subdomain $\Omega_{ij}$ and its internal/external boundaries (left picture), its splitting into four parts for the determination of a startvector (middle picture), and the four types of components of this startvector on such a part (right picture).*




Let $\Omega_{ij}$ be the interior of a rectangular area with number $(i, j)$. With $\Gamma^\ell_{ij}, \Gamma^r_{ij}, \Gamma^b_{ij}$, and $\Gamma^t_{ij}$, respectively, we denote the left, right, bottom, and top boundary of $\Omega_{ij}$, respectively (see left picture of Fig. 4.9). For such a boundary, say $\Gamma^\ell_{ij}$, two situations can occur:

1. $\Omega_{ij}$ has no adjacent subdomain on the left,

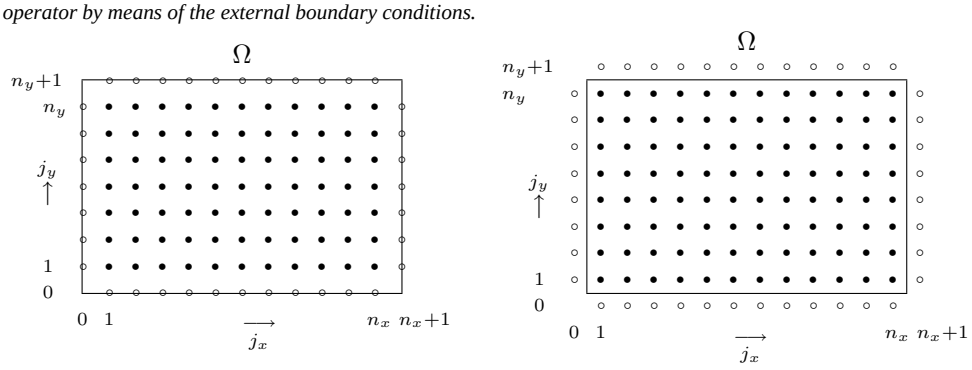
2. $\Omega_{ij}$ does have an adjacent subdomain on the left.

For situation 1, the values on $\Gamma^\ell_{ij}$ are external boundary values and these are imposed by the eigenvalue problem itself. For situation 2, the values on $\Gamma^\ell_{ij}$ are internal boundary values and we need to determine appropriate coupling parameters for the coupling between the subgrids that cover $\Omega_{ij}$ and $\Omega_{i-1j}$.

Define $\overline{\Omega}_{ij} \equiv \Omega_{ij} \cup \Gamma^\ell_{ij} \cup \Gamma^r_{ij} \cup \Gamma^b_{ij} \cup \Gamma^t_{ij}$. Let $I$ be the index set that contains the coordinates $(i, j)$ of the subdomains $\Omega_{ij}$ on which the operator is defined and $\Omega_{ij} \cap \Omega_{kl} = \emptyset$ for all $(i, j), (k, l) \in I$ with $i \neq k$ or/and $j \neq l$. The composition is then given by $\overline{\Omega} = \bigcup_{(i,j)\in I} \overline{\Omega}_{ij}$. With $\Omega$ we denote the interior of $\overline{\Omega}$.

In contrast to the preceding experiments, the discretization of domain $\Omega$ will be different. For a rectangular $\Omega$ this is illustrated in Fig. 4.10 (the left picture shows the old situation, the right one the new situation). The reason for this change is that we can treat all subgrids that cover the subdomains in a similar way now, the situation in the right picture in Fig. 4.10 can be seen as a special case (a composition of only one subdomain: $\Omega = \Omega_{11}$). For a subdomain $\Omega_{ij}$ the values at the circles (o) in the right picture of Fig. 4.10 correspond to the $\widetilde{\cdot}$ components of an enhanced vector in case of an internal boundary. Otherwise, for an external boundary, the values at these gridpoints are eliminated by means of the external boundary conditions.

FIGURE 4.10. *Two different discretizations for a domain $\Omega$. The left picture shows a discretization with extremal grid points (the circles* (o)*) located at the (external) boundary. In the right picture the discretization is such that the boundary is between the extremal grid points (the circles* (o)*) and the first/last row/column of the internal grid points (the bullets* (•)*). For both discretizations, the values at the circles* (o) *are eliminated in the discretized operator by means of the external boundary conditions.*




*Startvector*

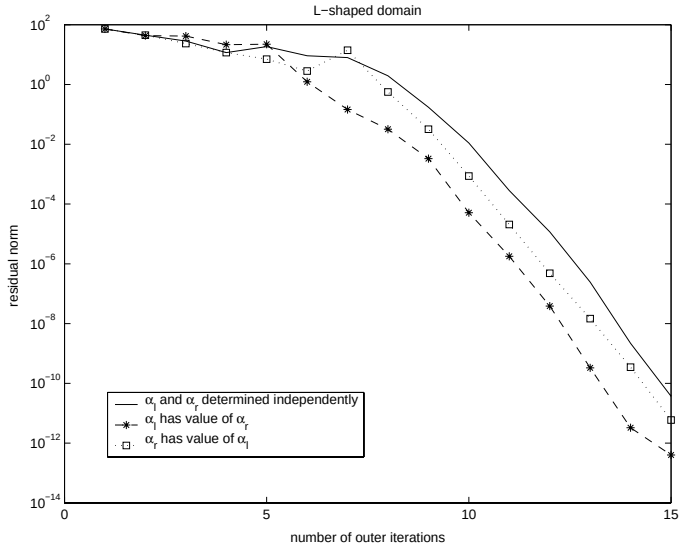
We also need a startvector for the Jacobi-Davidson method. This is described now for the case of homogeneous Dirichlet conditions at the external boundary. For the determination of the startvector we split each subdomain $\Omega_{ij}$ in four equal parts: $\Omega_{ij}^{\ell b}, \Omega_{ij}^{\ell t}, \Omega_{ij}^{rb}$, and $\Omega_{ij}^{rt}$ (see the middle picture of Fig. 4.9). We let the component of the startvector on such a part depend on whether the operator is defined on the adjacent subdomains. Essentially, four types of components exist, they are illustrated by plot I, II, III, and IV in the right picture of Fig. 4.9. For example, there are three subdomains next to $\Omega_{ij}^{\ell b}$: $\Omega_{i-1j-1}, \Omega_{i-1j}$, and $\Omega_{ij-1}$. Since on each adjacent subdomain the operator is defined (marked by "$\times$") or not (marked by "$\circ$"), eight configurations are possible:

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| $\Omega_{i-1j-1}$ | $\times$ | $\times$ | $\times$ | $\circ$ | $\circ$ | $\times$ | $\circ$ | $\circ$ |
| $\Omega_{i-1j}$ | $\times$ | $\times$ | $\circ$ | $\times$ | $\circ$ | $\circ$ | $\circ$ | $\times$ |
| $\Omega_{ij-1}$ | $\times$ | $\circ$ | $\times$ | $\circ$ | $\times$ | $\circ$ | $\circ$ | $\times$ |
| plot | I | II | II | II | II | III | III | IV |

*Coupling parameters*

Now, we formulate how the coupling parameters for an internal boundary of $\Omega_{ij}$ can be determined. Suppose $\Omega_{ij}$ has an internal boundary $\Gamma_{ij}^r$. In order to be able to optimize the coupling, information about the eigenvectors of the operator in the $y$-direction is needed (§3.4 of chapter 3). Let $i_1$ indicate the smallest, and $i_2$ the largest integer with $i_1 \leq i \leq i_2$ such that the operator is defined on $\Omega_{kj}$ for $k = i_1, \ldots, i_2$. We determine the required information from the operator in the $y$-direction on the subcomposition $\bigcup_{k=i_1,\ldots,i_2} \overline{\Omega}_{kj}$ by ignoring the interfaces between the subdomains $\Omega_{kj}$. This is the same approach as in §4.4.1 for a rectangular domain $\Omega$. For a nonrectangular domain $\Omega$ some options are left, this will be discussed next.

FIGURE 4.11. *Convergence history of Jacobi-Davidson when applied to the discretized eigenvalue problem of the two-dimensional Laplace operator for approximate solutions to the correction equation obtained with* $4$ *right preconditioned GMRES iterations. The operator is defined on an L-shaped domain* $\Omega$*, here the long side has size* $b_y = 4$ *and the short side* $b_x = 2$*. For the construction of the preconditioner with simple optimized coupling* $\Omega$ *is decomposed into five equal subdomains of size* $b_{xi} = b_{yj} = 1$*. Each subdomain is covered by a* $20 \times 20$ *subgrid. For explanation see §4.4.2.1.*




#### 4.4.2.1 Nonrectangular compositions

For nonrectangular compositions of subdomains some options are left for the determination of coupling parameters. We will illustrate and investigate this through some numerical experiments.

The experiments are performed in MATLAB 6.0.

We start with an L-shaped domain $\Omega$, and we want to compute the largest eigenvalue and corresponding eigenvector ("the MATLAB logo") of the Laplace operator on $\Omega$. The domain is a composition of 5 square subdomains of equal size ($b_{xi} = b_{yj} = 1$), two subdomains in the $x$-direction and four in the $y$-direction. Subdomain $\Omega_{21}$ has no neighbouring subdomains in the $y$-direction, the subdomains $\Omega_{12}, \Omega_{13}$, and $\Omega_{14}$ are on top of $\Omega_{11}$. Each subdomain is covered by a $20 \times 20$ subgrid.

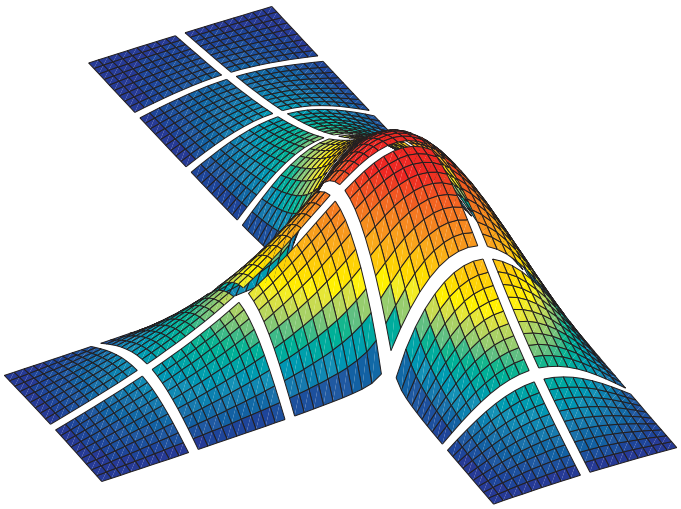
We determine the coupling parameters at the interfaces as in §4.4.2. We do this for the simple optimized coupling. Then $\alpha_\ell$ of $\Omega_{11}$ is determined by considering the Laplace operator in the $y$-direction on $\Omega_{11} \cup \Omega_{12} \cup \Omega_{13} \cup \Omega_{14}$. For $\alpha_r$ of $\Omega_{21}$ this is done on $\Omega_{21}$ only. Hence, the values of $\alpha_\ell$ and $\alpha_r$ differ. We wonder how important these differences are and how they affect the quality of the preconditioner. To investigate this we compare the Jacobi-Davidson processes for

- option 1: $\alpha_\ell$ and $\alpha_r$ are determined independently as described,
- option 2: $\alpha_\ell$ has the same value as $\alpha_r$, and
- option 3: $\alpha_r$ has the same value as $\alpha_\ell$.

The construction of the startvector for Jacobi-Davidson is shown in §4.4.2. The value of the coupling parameter shows a significant difference for the three options. Approximate solutions to the correction equation are obtained from 4 steps with right preconditioned GMRES.

Fig. 4.11 shows the convergence history of Jacobi-Davidson for these three options. The first impression is that option 2 should be preferred. But if we study the convergence plots after iteration 8, when convergence starts, then it can be observed that the plots are almost parallel. Because of this we expect that the pre-convergence phase of the process for option 2 is, accidentally, somewhat shorter. To confirm this we repeated the experiment for another composition.

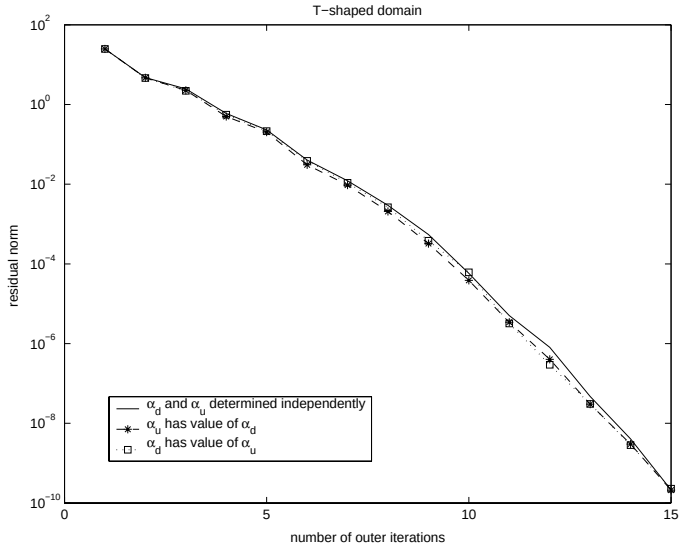
FIGURE 4.12. *Eigenvector of the discretized eigenvalue problem of the two-dimensional Laplace operator on a T-shaped domain. See for explanation §4.4.2.1.*



The domain $\Omega$ is now T-shaped, the long side of $\Omega$ has size $b_x = 8$ and the short side $b_y = 5$. It is a composition of 22 square subdomains of equal size ($b_{xi} = b_{yj} = 1$). Each subdomain is covered by a $10 \times 10$ subgrid. Again we aim at the largest eigenvalue and corresponding eigenvector (see Fig. 4.12) of the Laplace operator on $\Omega$. For the correction equation also 4 steps with right preconditioned GMRES are done. We consider the three options for the coupling parameters $\alpha_b$ and $\alpha_t$ at the interface that split the $\overline{|\ |}$ into $|\ |$ and $\overline{\ \ \ }$.

The convergence history of the Jacobi-Davidson processes for the three options on this domain are plotted in Fig. 4.13. Now, fast convergence starts immediately after step 1. The three options for $\alpha_\ell$ and $\alpha_r$ show almost the same overall process. We conclude that also for nonrectangular compositions the determination of coupling parameters as formulated in §4.4.2, may lead to a good preconditioner.

FIGURE 4.13. *Convergence history of Jacobi-Davidson when applied to the discretized eigenvalue problem of the two-dimensional Laplace operator for approximate solutions to the correction equation obtained with* 4 *right preconditioned GMRES iterations. The operator is defined on a T-shaped domain* $\Omega$*, here the long side has size* $b_x = 8$ *and the short side* $b_y = 5$*. For the construction of the preconditioner with simple optimized coupling* $\Omega$ *is decomposed into* 22 *equal subdomains of size* $b_{xi} = b_{yj}$*. Each subdomain is covered by a* $10 \times 10$ *subgrid. For explanation see §4.4.2.1.*




### 4.4.2.2 More complicated domains

Our final experiment is intended to illustrate that with a combination of Jacobi-Davidson, and the preconditioner based on domain decomposition, eigenvalues and corresponding eigenvectors can be computed of operators that are defined on more complicated geometries. For that purpose we consider the Laplace operator on a square domain ($b_x = b_y = 17$). The domain is a composition of square subdomains of equal size ($b_{xi} = b_{yj} = 1$). We write the initials and age in years of Utrecht University on the domain by excluding the subdomains that correspond to these parts from the domain. Each subdomain is covered by a $10 \times 10$ subgrid. With Jacobi-Davidson we want to compute the largest eigenvalue and corresponding eigenvector. The construction of the startvector is described in §4.4.2. Approximate solutions to the correction equation are computed with 10 steps of right preconditioned GMRES.

TABLE 4.2. *Convergence history of Jacobi-Davidson applied to the discretized eigenvalue problem of the two-dimensional Laplace operator on a more complicated domain for approximate solutions to the correction equation obtained with 10 right preconditioned GMRES iterations. The preconditioner is based on domain decomposition with simple optimized coupling. For explanation see §4.4.2.2.*

| step | $\theta$ | $\|\mathbf{r}\|_2$ | step | $\theta$ | $\|\mathbf{r}\|_2$ |
|---|---|---|---|---|---|
| 1 | -5.538666263730 | 2.17e+01 | 14 | -0.9918317516109 | 6.19e-04 |
| 2 | -1.681336714181 | 9.44e+00 | 15 | -0.9918311440008 | 1.13e-03 |
| 3 | -1.023404498995 | 9.27e-01 | 16 | -0.9918296679279 | 5.46e-04 |
| 4 | -0.9954697880199 | 3.30e-01 | 17 | -0.9918288842076 | 5.85e-04 |
| 5 | -0.9927244240975 | 1.41e-01 | 18 | -0.9918283703891 | 1.94e-04 |
| 6 | -0.9925038578219 | 2.19e-02 | 19 | -0.9918283246088 | 2.88e-05 |
| 7 | -0.9923720853492 | 7.60e-02 | 20 | -0.9918283045227 | 1.68e-05 |
| 8 | -0.9920896120028 | 4.21e-02 | 21 | -0.9918283015561 | 1.63e-06 |
| 9 | -0.9920104205556 | 2.10e-02 | 22 | -0.9918283013716 | 2.78e-07 |
| 10 | -0.9919586304587 | 2.41e-02 | 23 | -0.9918283013527 | 4.51e-08 |
| 11 | -0.9918967786952 | 1.96e-02 | 24 | -0.9918283013551 | 3.95e-09 |
| 12 | -0.9918336138825 | 1.47e-02 | 25 | -0.9918283013552 | 2.86e-10 |
| 13 | -0.9918323113539 | 1.25e-03 | | | |

The preconditioner is based on domain decomposition with simple optimized coupling, with the determination of the coupling parameters as in §4.4.2. The experiment is performed in MATLAB 6.0.

Table 4.2 contains the results of the experiment. Displayed are the first 13 digits of the approximate eigenvalue and the residual norm of the approximate eigenpair at each Jacobi-Davidson step. The approximate eigenvector at step 25 is shown in Fig. 4.14.

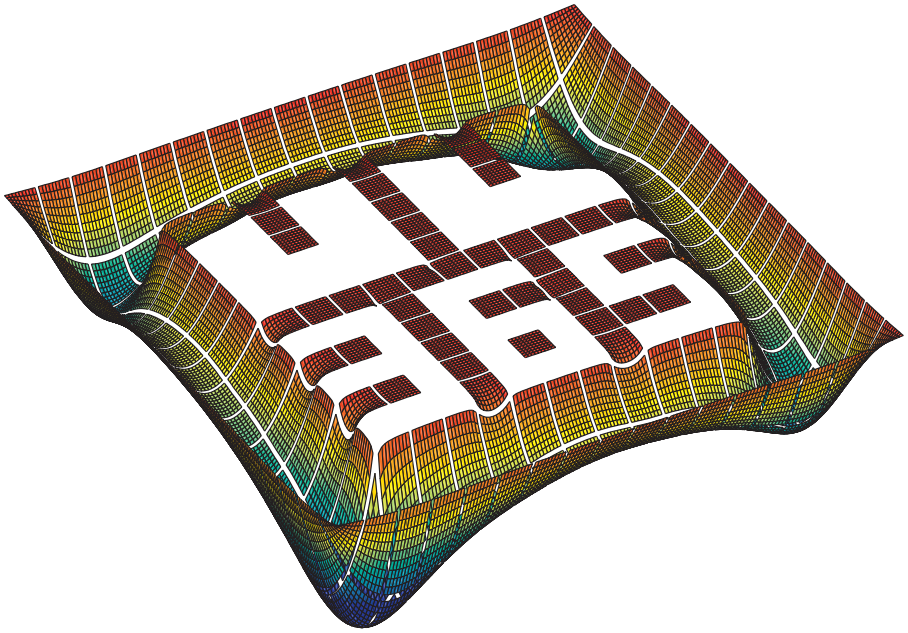
## 4.5 Conclusions

In this chapter we considered two major practical extensions of the domain decomposition approach for the Jacobi-Davidson method as proposed in chapter 3.

First, we outlined a strategy for the case of a PDE with variable coefficients. The strategy is based on the assumption that variable coefficients can be approximated locally by frozen coefficients. Numerical experiments showed that this strategy leads to useful preconditioners for a number of typical PDE's.

Secondly, we extended the domain decomposition approach to more complicated geometries. For that purpose, a sophisticated startvector for the Jacobi-Davidson method had to be constructed. Also, the strategy for the determination of coupling parameters was generalized by taking into account the geometry. We illustrated its effectiveness numerically for some different geometries.

FIGURE 4.14. *Eigenvector of the discretized eigenvalue problem of the two-dimensional Laplace operator on a more complicated domain. See for explanation §4.4.2.2.*



# Chapter 5

# Domain decomposition on different levels of the Jacobi-Davidson method

**Abstract**

Most computational work of Jacobi-Davidson [46], an iterative method suitable for computing solutions of large dimensional eigenvalue problems, is due to a so-called correction equation on the intermediate level. In chapter 3 an approach based on domain decomposition is proposed to reduce the wall clock time and local memory requirements for the computation of (approximate) solutions to the correction equation. This chapter discusses the aspect that the domain decomposition approach can also be applied on the highest level of the Jacobi-Davidson method. Numerical experiments show that for large scale eigenvalue problems this aspect is nontrivial.

**Keywords:** Eigenvalue problems, domain decomposition, Jacobi-Davidson, Schwarz method, nonoverlapping, iterative methods.

**2000 Mathematics Subject Classification:** 65F15, 65N25, 65N55.

## 5.1 Introduction

For the computation of solutions to a large scale linear eigenvalue problem several iterative methods exist. Amongst them, the Jacobi-Davidson method [46] has some interesting and useful properties.

One essential property is that it allows flexibility in the so-called correction equation. At each iteration Jacobi-Davidson generates an (approximate) solution of this equation: the correction vector. Special care is also needed: most computational work of the method arises from the correction equation. It involves a linear system with size equal to that of the original eigenvalue problem and therefore the incorporation of a good preconditioner is crucial, when solving the correction equation iteratively.

For the solution of large linear systems originating from the discretization of partial differential equations, domain decomposition methods proved to be a successful tool. However, the linear system that is described by the correction equation may be highly indefinite and is given in an unusual manner so that the application of a domain decomposition method needs special attention. In chapter 3 a preconditioner based on domain decomposition for the correction equation is constructed for advection-diffusion type of eigenvalue problems.

This chapter discusses the aspect that Jacobi-Davidson is a nested iterative method if the correction equation is solved approximately with an iterative method. Before the specific preconditioner can be applied to a linear system, the system needs first to be extended to a so-called enhanced system. For the use within Jacobi-Davidson an enhancement of the linear system defined by the correction equation is than obvious. But other choices are possible as well, for example enhancement at the level of the eigenvalue problem itself. These choices are discussed here. For approximate solutions of the correction equation the differences between them will turn out to be nontrivial.

This chapter is organized as follows. First §5.2 summarizes the domain decomposition technique: enhancements are introduced and the preconditioner is described. Then §5.3 highlights the different levels of enhancement in the Jacobi-Davidson method. In §5.4 it will be shown how for each level the preconditioner is incorporated and the differences are discussed. Section §5.5 concludes with some illustrative numerical examples, also to indicate the importance of the differences for the numerical treatment of large scale eigenvalue problems.

## 5.2 A nonoverlapping Schwarz method

This section briefly outlines the domain decomposition technique, a nonoverlapping additive Schwarz method which is based on previous work by Tang [57] and Tan & Borsboom [55, 56].

We will describe enhancements of matrices and vectors in an algebraic way. Then it is shown how to apply these enhancements to linear systems and (standard) eigenvalue problems. We conclude this section with a recapitulation of the construction of preconditioners for these enhanced systems. This allows also for a physical/geometric interpretation.

For simplicity, but without loss of generality, only the two subdomain case is considered.

### 5.2.1 Enhancement of matrices and vectors

Suppose that the matrix $\mathbf{B}$ has been partitioned as follows:

$$\begin{bmatrix} \mathbf{B}_{11} & \mathbf{B}_{1\ell} & \mathbf{B}_{1r} & \mathbf{0} \\ \mathbf{B}_{\ell 1} & B_{\ell\ell} & B_{\ell r} & \mathbf{0} \\ \mathbf{0} & B_{r\ell} & B_{rr} & \mathbf{B}_{r2} \\ \mathbf{0} & \mathbf{B}_{2\ell} & \mathbf{B}_{2r} & \mathbf{B}_{22} \end{bmatrix}.$$

Note that the parts $\mathbf{B}_{12}, \mathbf{B}_{\ell 2}, \mathbf{B}_{r1}$, and $\mathbf{B}_{21}$ are zero. This is a typical situation that will be motivated in §5.2.3. The *enhancement* $\mathbf{B}_C$ for $\mathbf{B}$ is defined by

$$\mathbf{B}_C \equiv \begin{bmatrix} \mathbf{B}_{11} & \mathbf{B}_{1\ell} & \mathbf{B}_{1r} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{B}_{\ell 1} & B_{\ell\ell} & B_{\ell r} & 0 & 0 & \mathbf{0} \\ \mathbf{0} & C_{\ell\ell} & C_{\ell r} & -C_{\ell\ell} & -C_{\ell r} & \mathbf{0} \\ \mathbf{0} & -C_{r\ell} & -C_{rr} & C_{r\ell} & C_{rr} & \mathbf{0} \\ \mathbf{0} & 0 & 0 & B_{r\ell} & B_{rr} & \mathbf{B}_{r2} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{B}_{2\ell} & \mathbf{B}_{2r} & \mathbf{B}_{22} \end{bmatrix}. \tag{5.1}$$

The submatrices $C_{\ell\ell}, C_{\ell r}, C_{r\ell}$ and $C_{rr}$ define the *coupling* matrix $C \equiv \begin{bmatrix} C_{\ell\ell} & -C_{\ell r} \\ -C_{r\ell} & C_{rr} \end{bmatrix}$.
Except for $\mathbf{I}_0$, the enhancement of the identity matrix $\mathbf{I}$ with zero coupling matrix, in the following all enhanced matrices will have a nonsingular coupling $C$.
For a vector $\mathbf{y}$, partitioned as $\left( \mathbf{y}_1^T,\, y_\ell^T,\, y_r^T,\, \mathbf{y}_2^T \right)^T$, three types of enhancement are defined:

- the *zero enhancement* $\mathbf{y}_0 \equiv (\, \mathbf{y}_1^T,\, y_\ell^T,\, 0^T,\, 0^T,\, y_r^T,\, \mathbf{y}_2^T \,)^T$,
- the *unbalanced enhancement* $\underset{\sim}{\mathbf{y}} \equiv (\, \mathbf{y}_1^T,\, y_\ell^T,\, \widetilde{y}_r^T,\, \widetilde{y}_\ell^T,\, y_r^T,\, \mathbf{y}_2^T \,)^T$, and
- the *balanced enhancement* $\underline{\mathbf{y}} \equiv (\, \mathbf{y}_1^T,\, y_\ell^T,\, y_r^T,\, y_\ell^T,\, y_r^T,\, \mathbf{y}_2^T \,)^T$.

The other way around, given one of these enhancements, say $\underset{\sim}{\mathbf{y}}$, the *restriction* of $\underset{\sim}{\mathbf{y}}$ is defined by $\mathbf{y} \equiv \left( \mathbf{y}_1^T,\, y_\ell^T,\, y_r^T,\, \mathbf{y}_2^T \right)^T$, that is by just skipping the $\widetilde{\cdot}_r$ and $\widetilde{\cdot}_\ell$ parts. The parts $y_\ell, y_r, \widetilde{y}_\ell$, and $\widetilde{y}_r$ have equal size. The values of these parts are generated by an iteration process that will be discussed in §5.2.3.

### 5.2.2 Enhancement of linear systems of equations

With the enhancements introduced in §5.2.1, a linear system

$$\mathbf{B}\,\mathbf{y} = \mathbf{d} \tag{5.2}$$

can be enhanced to

$$\mathbf{B}_C\,\underset{\sim}{\mathbf{y}} = \mathbf{d}_0. \tag{5.3}$$

It is easy to see that $\mathbf{y}$ is a solution of (5.2) if and only if $\underline{\mathbf{y}}$ is a solution of (5.3): $\mathbf{B}_C\,\underline{\mathbf{y}} = (\mathbf{B}\,\mathbf{y})_0$.

For a standard eigenvalue problem

$$\mathbf{A}\,\mathbf{x} = \lambda\,\mathbf{x}, \tag{5.4}$$

one should be aware of the fact that both sides contain the unknown vector $\mathbf{x}$. Therefore, (5.4) is rewritten as

$$(\mathbf{A} - \lambda\,\mathbf{I}\,)\;\mathbf{x} = \mathbf{0}. \tag{5.5}$$

If an eigenvalue $\lambda$ is already known then this is a linear equation that describes the corresponding eigenvector $\mathbf{x}$. Therefore, we call (5.5) the *eigenvector equation*. Similar to the enhancement of a linear system, equation (5.5) is enhanced to

$$(\mathbf{A} - \lambda\,\mathbf{I}\,)_C\;\underset{\sim}{\mathbf{x}} = (\mathbf{A}_C - \lambda\,\mathbf{I}_0\,)\;\underset{\sim}{\mathbf{x}} = \mathbf{0}. \tag{5.6}$$

It can easily be verified that, given some $\lambda$, $\mathbf{x}$ is a solution of (5.5) if and only if $\underline{\mathbf{x}}$ is a solution of (5.6).

Another option is an enhancement of the the eigenvalue problem (5.4) itself:

$$\mathbf{A}_C\,\underset{\sim}{\mathbf{x}} = \lambda\,\mathbf{I}_0\,\underset{\sim}{\mathbf{x}}. \tag{5.7}$$

Note that here, artificially, an extra eigenvalue $\infty$ of multiplicity dim $C$ is created. We can consider (5.7) as a generalized eigenvalue problem and use some numerical method for generalized eigenvalue problems. In case of the Jacobi-Davidson method, the numerical example from §5.5.1 will show that such a *black box approach* is not so successful.

### 5.2.3 Construction of the preconditioner

The motivation for the enhancements in §5.2.1 and §5.2.2 is the possibility to precondition an enhanced system by performing accurate solves of subsystems and to tune the coupling between those subsystems for improved speed of convergence of the iterative solver in which the preconditioner is incorporated.

For the enhanced linear system (5.3), the two subsystems are described by the boxed parts in (5.1). Let $\mathbf{M}_C$ denote this part of $\mathbf{B}_C$. The preconditioned enhanced system

$$\mathbf{M}_C^{-1}\,\mathbf{B}_C\,\underset{\sim}{\mathbf{y}} = \mathbf{M}_C^{-1}\,\mathbf{d}_0 \tag{5.8}$$

is solved by a convenient iterative method. The key observation is that the iterates of such a method are (linear combinations of) powers of $\mathbf{M}_C^{-1}\,\mathbf{B}_C$ times a vector. This motivates the construction of a coupling $C$, such that the error induced by the matrix splitting $\mathbf{B}_C = \mathbf{M}_C - \mathbf{N}$ is damped out by higher powers of $\mathbf{M}_C^{-1}\,\mathbf{B}_C = \mathbf{I} - \mathbf{M}_C^{-1}\,\mathbf{N}$ applied to $\mathbf{M}_C^{-1}\,\mathbf{d}_0$. Such a tuning of $C$ requires knowledge of the (physical) equations from which $\mathbf{B}$ arises via discretization. The subsystems described by $\mathbf{M}_C$ represent a discretization of the (physical) equations on the subdomains and $C$ can be interpreted as discretized coupling equations between the subdomains. For such a typical discretization only a small number of unknowns

couple with unknowns from different subdomains. In §5.2.1 these unknowns are indicated by $_\ell$ and $_r$ (the unknowns at the *left* and *right* respectively from the internal interface between the subdomains). The parts $\widetilde{y}_\ell$ and $\widetilde{y}_r$ respectively are virtual copies of the parts $y_\ell$ and $y_r$ respectively. For the iterates $\underset{\sim}{\mathbf{y}}^{(i)}$ generated by the iterative method the values of these parts will not be the same in general, but if convergence takes place then $\widetilde{y}_\ell^{(i)} \to y_\ell^{(i)}$ and $\widetilde{y}_r^{(i)} \to y_r^{(i)}$, and therefore $\underset{\sim}{\mathbf{y}}^{(i)} \to \underline{\mathbf{y}}^{(i)}$.

Most of the unknowns are coupled only to unknowns from the same subdomain (in §5.2.1, for instance, this internal coupling is described by $\mathbf{B}_{11}$). This also explains that $\mathbf{B}_{12}, \mathbf{B}_{\ell 2}, \mathbf{B}_{r1}$, and $\mathbf{B}_{21}$ are zero in typical situations of interest.

For enhanced linear systems of the form (5.3), tuning of the coupling has been proposed by Tan and Borsboom [55, 56]. The construction of suitable couplings for enhancements of linear correction equations occurring in the Jacobi-Davidson method, for a model eigenvalue problem, was described in chapter 3. In the next sections we will show that such a coupling can also be used on the highest level of the Jacobi-Davidson method, namely by enhancing the eigenvector equation.

## 5.3 Solution of the eigenvalue problem

### 5.3.1 The Jacobi-Davidson method

For an eigenvalue problem the Jacobi-Davidson method [46] computes iteratively a solution. Here, for simplicity, the standard eigenvalue problem (5.4) is considered. The ingredients of each iteration are:

**Extract** an approximate eigenpair $(\theta, \mathbf{u})$ from a search subspace $\mathbf{V}$ via a *Rayleigh-Ritz* principle.
The *Rayleigh* part projects $\mathbf{A}$ on $\mathbf{V}$ by constructing the *matrix Rayleigh quotient* of $\mathbf{V}$ or *interaction matrix*

$$H \equiv \mathbf{V}^* \mathbf{A} \mathbf{V}.$$

The *Ritz* part solves the projected eigenvalue problem

$$H\, s = \theta\, s, \tag{5.9}$$

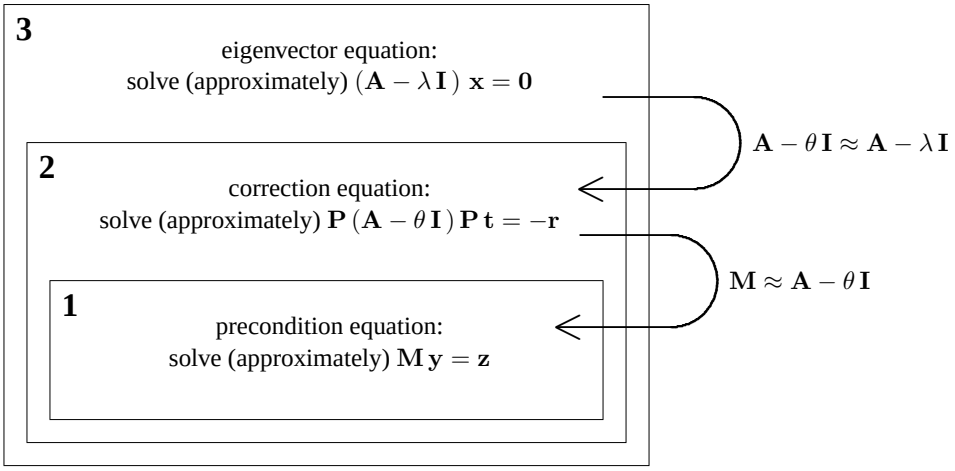
selects a *Ritz value* $\theta$ and computes the corresponding *Ritz vector* $\mathbf{u} \equiv \mathbf{V}\, s$ and residual $\mathbf{r} \equiv (\, \mathbf{A} - \theta \mathbf{I}\,)\ \mathbf{u}$.

**Correct** the approximate eigenvector $\mathbf{u}$.
The *correction vector* $\mathbf{t}$ is computed from the *correction equation*

$$\mathbf{t} \perp \mathbf{u}, \quad \mathbf{P}\ (\, \mathbf{A} - \theta \mathbf{I}\,)\ \mathbf{P}\, \mathbf{t} = -\mathbf{r} \quad \text{with} \quad \mathbf{P} \equiv \mathbf{I} - \frac{\mathbf{u}\mathbf{u}^*}{\mathbf{u}^*\mathbf{u}}. \tag{5.10}$$

**Expand** the search subspace $\mathbf{V}$ with the correction vector $\mathbf{t}$.

FIGURE 5.1. *Three levels in the Jacobi-Davidson method suitable for enhancement.*




For not too low dimensional problems, most computational work of Jacobi-Davidson is in the second ingredient. Convergence of the method depends strongly on the accuracy of (approximate) solutions of the correction equation. In many practical cases exact solution of the correction equation is not feasible because of time and/or memory constraints. Then one has to rely on approximate solutions obtained from some iterative method for linear systems. For the convergence of such a method a good preconditioner is highly desirable.

Three levels of the Jacobi-Davidson method are distinguished (indicated by numbers in Fig. 5.1): the eigenvector equation that describes the eigenvector corresponding to an eigenvalue $\lambda$ on the highest level, the correction equation that describes the correction vector on the intermediate level, and the *precondition equation* that describes preconditioning on the lowest level. The different levels are related by the involved linear operators (indicated by arrows in Fig. 5.1): as the exact eigenvalue $\lambda$ in the operator $\mathbf{A} - \lambda \mathbf{I}$ is not known beforehand, it is replaced by an approximation $\theta$, which leads to the operator $\mathbf{A} - \theta \mathbf{I}$. This operator is replaced by a preconditioner $\mathbf{M}$ with which it is cheaper to solve systems.

The relationships between the levels are the motivation for the different levels of enhancement that will be considered in §5.3.2. If solutions to the correction equation are computed with a preconditioned iterative solver then Jacobi-Davidson consists of two nested iterative solvers. In the innerloop a search subspace for the (approximate) solution of the correction equation is built up by powers of $\mathbf{M}^{-1}\ (\mathbf{A} - \theta \mathbf{I})$ for fixed $\theta$. In the outerloop a search subspace for the (approximate) solution of the eigenvalue problem is built up by powers of $\mathbf{M}^{-1}\ (\mathbf{A} - \theta \mathbf{I})$ for variable $\theta$. As $\theta$ varies slight in succeeding outer iterations, one may take advantage of the nesting for the special preconditioner of §5.2.3 as we will show in §5.4.

### 5.3.2 Different levels of enhancement

All three levels from §5.3.1 are suitable for the enhancements as introduced in §5.2. Therefore we consider the following enhancements:

- *enhanced precondition equation*
- *enhanced correction equation*
- *enhanced eigenvector equation*

First, we will show §5.3.2.1 that these three enhancements lead to different correction equations. Then, in §5.3.2.2, we will discuss how the two other ingredients may be adapted to fit the corresponding correction equation in the Jacobi-Davidson method.

#### 5.3.2.1 Correction equation

**Enhanced precondition equation**
A correction $\mathbf{t}$ is computed from the correction equation (5.10) for the standard eigenvalue problem (5.4).

**Enhanced correction equation**
A correction $\underset{\sim}{\mathbf{t}}$ is computed from the enhancement of correction equation (5.10) (§3.3.2 of chapter 3):

$$\underset{\sim}{\mathbf{t}} \perp \mathbf{u}_0, \quad \mathbf{P}\,(\,\mathbf{A}_C - \theta\,\mathbf{I}_0\,)\,\mathbf{P}\,\underset{\sim}{\mathbf{t}} = -\mathbf{r}_0 \quad \text{with} \quad \mathbf{P} \equiv \mathbf{I} - \frac{\mathbf{u}_0\mathbf{u}_0^*}{\mathbf{u}_0^*\mathbf{u}_0}, \tag{5.11}$$

$\mathbf{u}_0$ the zero enhancement of $\mathbf{u}$ and $\mathbf{r}_0$ the zero enhancement of the residual $\mathbf{r} \equiv (\,\mathbf{A} - \theta\,\mathbf{I}\,)\,\mathbf{u}$.

Note that as $\mathbf{r}_0 = (\,\mathbf{A}_C - \theta\,\mathbf{I}_0\,)\,\underline{\mathbf{u}}$, from equation (5.11) also a correction $\underset{\sim}{\mathbf{t}}$ for a balanced enhanced $\underline{\mathbf{u}}$ can be computed.

**Enhanced eigenvector equation**
A correction $\underset{\sim}{\mathbf{t}}$ is computed from the correction equation for the enhancement (5.6) of the eigenvector equation (5.5):

$$\underset{\sim}{\mathbf{t}} \perp \mathbf{u}_0, \quad \mathbf{P}\,(\mathbf{A}_C - \theta\,\mathbf{I}_0\,)\,\mathbf{P}\,\underset{\sim}{\mathbf{t}} = -\mathbf{P}\,\underset{\sim}{\mathbf{r}} \quad \text{with} \quad \mathbf{P} \equiv \mathbf{I} - \frac{\mathbf{u}_0\mathbf{u}_0^*}{\mathbf{u}_0^*\mathbf{u}_0}, \tag{5.12}$$

$\mathbf{u}_0$ the zero enhancement of the restriction of the unbalanced $\underset{\sim}{\mathbf{u}}$ and residual $\underset{\sim}{\mathbf{r}} \equiv (\mathbf{A}_C - \theta\,\mathbf{I}_0\,)\,\underset{\sim}{\mathbf{u}}$. Note that this residual also measures the errors $u_\ell - \widetilde{u}_\ell$ and $\widetilde{u}_r - \widetilde{u}_r$.

Equation (5.12) is derived now. It is a straightforward "generalization" of the derivation of the correction equation for (5.4) via some first order correction approach [46].[1]

[1] For this ingredient of Jacobi-Davidson, where the value of the approximate eigenvalue $\theta$ is known, the correction equation (5.12) can also be obtained by considering the enhanced eigenvector equation (5.6) as an instance of the "generalized eigenvalue problem" (5.7) [44, 26]. However, the first ingredient, the construction of an interaction matrix and determination of a new Ritz pair is with respect to the original eigenvalue problem (5.4).

Suppose we have computed a pair $(\theta, \mathbf{u})$ that approximates some eigenpair $(\lambda, \mathbf{x})$. Furthermore, also an unbalanced enhancement $\underset{\sim}{\mathbf{u}}$ of $\mathbf{u}$ is available from the information collected so far. We want to compute a correction $\underset{\sim}{\mathbf{t}}$ to $\underset{\sim}{\mathbf{u}}$, such that $\underline{\mathbf{x}} = \underset{\sim}{\mathbf{u}} + \underset{\sim}{\mathbf{t}}$. Here $\underline{\mathbf{x}}$ is the balanced enhancement of $\mathbf{x}$. The enhanced eigenvector equation (5.6) yields

$$(\mathbf{A}_C - \lambda \mathbf{I}_0)(\underset{\sim}{\mathbf{u}} + \underset{\sim}{\mathbf{t}}) = \mathbf{0}.$$

We rewrite this as follows:

$$(\mathbf{A}_C - \theta \mathbf{I}_0)\underset{\sim}{\mathbf{t}} = -\underset{\sim}{\mathbf{r}} + (\lambda - \theta)\mathbf{u}_0 + (\lambda - \theta)\mathbf{t}_0.$$

Here $\mathbf{u}_0 = \mathbf{I}_0 \underset{\sim}{\mathbf{u}}$ and $\mathbf{t}_0 = \mathbf{I}_0 \underset{\sim}{\mathbf{t}}$. For $\theta$ close to $\lambda$ and $\underset{\sim}{\mathbf{u}}$ close to $\underline{\mathbf{x}}$ the term $(\lambda - \theta)\mathbf{t}_0$ is of second order. This contribution is neglected:

$$(\mathbf{A}_C - \theta \mathbf{I}_0)\underset{\sim}{\mathbf{t}} = -\underset{\sim}{\mathbf{r}} + (\lambda - \theta)\mathbf{u}_0. \tag{5.13}$$

The difference $\lambda - \theta$ on the right hand side of (5.13) is not known. This contribution disappears by projecting on the space orthogonal to $\mathbf{u}_0$:

$$\mathbf{P}\,(\mathbf{A}_C - \theta \mathbf{I}_0)\underset{\sim}{\mathbf{t}} = -\mathbf{P}\,\underset{\sim}{\mathbf{r}} \quad \text{with} \quad \mathbf{P} \equiv \mathbf{I} - \frac{\mathbf{u}_0 \mathbf{u}_0^*}{\mathbf{u}_0^* \mathbf{u}_0}. \tag{5.14}$$

If convergence takes place, that is if $\theta$ converges to an eigenvalue $\lambda$, then the operator $\mathbf{A}_C - \theta \mathbf{I}_0$ in (5.14) becomes singular. Because of this we can not compute proper solutions for (5.14). We repair this by also restricting the domain of the operator $\mathbf{A}_C - \theta \mathbf{I}_0$ to the space orthogonal to $\mathbf{u}_0$. Then, one arrives at (5.12), the correction equation for the enhanced eigenvector equation (5.6). Observe that on the right-hand side there is also a projection where in the correction equations (5.10) and (5.11) there is not. This is because $\underset{\sim}{\mathbf{r}}$ is not perpendicular to $\mathbf{u}_0$ in general.

#### 5.3.2.2 Incorporation in the Jacobi-Davidson method

For the incorporation of the three different enhanced equations of §5.3.2 we have to specify at which stage in the Jacobi-Davidson method vectors need to be enhanced and restricted. We discuss this for each enhanced equation here.

**Enhanced precondition equation**

Here, (approximate) solutions to the correction equation of Jacobi-Davidson are computed with an iterative method in combination with a preconditioner. Preconditioning consists of the following steps: enhance a vector, multiply the enhanced vector with the preconditioner $\mathbf{M}_C^{-1}$ and restrict the result. All other ingredients remain the same as in §5.3.1.

**Enhanced correction equation**

In this case the correction equation (5.10) of Jacobi-Davidson is enhanced. For that purpose the operator $\mathbf{A} - \theta \mathbf{I}$ is enhanced to $\mathbf{A}_C - \theta \mathbf{I}_0$ and the vectors $\mathbf{u}$ and $\mathbf{r}$ to $\mathbf{u}_0$ and $\mathbf{r}_0$ respectively.

It is easy to see that if the enhanced correction equation (5.11) is solved exactly then the solution is balanced: $\underset{\sim}{\mathbf{t}}$ and the restriction $\mathbf{t}$ of this $\underset{\sim}{\mathbf{t}}$ is also the unique solution of the original correction equation (5.10). However, if the enhanced correction equation (5.11) is solved only approximately then the solution $\underset{\sim}{\mathbf{t}}$ is unbalanced in general.

For the next outeriteration we restrict $\underset{\sim}{\mathbf{t}}$ to $\mathbf{t}$ and expand $\mathbf{V}$ with this $\mathbf{t}$. Jacobi-Davidson continues with the Rayleigh-Ritz principle of §5.3.1.

**Enhanced eigenvector equation**

For this situation also the vectors on the highest level of the eigenvector equation are enhanced. During the Jacobi-Davidson process an enhanced subspace $\underset{\sim}{\mathbf{V}}$ is built up. A new approximate eigenpair $(\theta, \mathbf{u})$ for the original eigenvalue problem (5.4) is computed with respect to the restricted subspace $\mathbf{V}$ of $\underset{\sim}{\mathbf{V}}$. The enhanced vector $\underset{\sim}{\mathbf{u}} \equiv \underset{\sim}{\mathbf{V}}\, s$ corresponding to $\mathbf{u} \equiv \mathbf{V}\, s$ with residual $\underset{\sim}{\mathbf{r}} \equiv (\mathbf{A}_C - \theta\, \mathbf{I}_0)\, \underset{\sim}{\mathbf{u}}$ is formed. We take this unbalanced enhancement $\underset{\sim}{\mathbf{u}}$ as it contains more information than the balanced enhancement $\underline{\mathbf{u}}$. For approximate solutions of the correction equation (5.12) this will turn out to be more efficient for the overall process (see §5.4 and §5.5).

For $\underset{\sim}{\mathbf{u}} = \underline{\mathbf{u}}$ the residual $\underset{\sim}{\mathbf{r}}$ equals $\mathbf{r}_0$ and is perpendicular to $\mathbf{u}_0$. If so then it is easy to see that the correction equations (5.12) and (5.11) are identical. When solved exactly both correction equations yield the same balanced correction vector $\underline{\mathbf{t}}$.

In general the solution of (5.12) is unbalanced. As new approximate solutions to the original eigenvalue problem (5.4) are extracted from $\mathbf{V}$ we need an orthonormal $\mathbf{V}$. Therefore we orthogonalize $\underset{\sim}{\mathbf{t}}$ with respect to the semi inner product defined by

$$\underset{\sim}{\mathbf{y}}^*\, \mathbf{I}_0\, \underset{\sim}{\mathbf{z}} \quad \text{for} \quad \underset{\sim}{\mathbf{y}}, \underset{\sim}{\mathbf{z}} \in \underset{\sim}{\mathbf{V}}. \tag{5.15}$$

Then $\underset{\sim}{\mathbf{V}}$ is expanded.

Overall we conclude the following: if the three different Jacobi-Davidson processes are started with the same search subspace $\mathbf{V}$ where the enhancement of $\mathbf{V}$ for the process with enhanced eigenvector equation is balanced and if the correction equations are solved exactly, then the three processes are equivalent.

This conclusion is of theoretical interest only. As already remarked in §5.2.3, the enhancements are introduced in order to accomodate a preconditioner based on domain decomposition. In practice, approximate solutions to the correction equation are computed by means of such a preconditioner $\mathbf{M}_C \approx \mathbf{A}_C - \theta\, \mathbf{I}_0$. The next section discusses preconditioning for the different enhanced equations.

## 5.4 Preconditioning

Now, at this point, we are able to describe the application of the preconditioner based on domain decomposition for the two highest levels of enhancement in the Jacobi-Davidson method.

The Jacobi-Davidson process with the lowest level of enhancement, that is the process with enhanced precondition equation will not be considered furthermore. The reason why is that, if only the precondition equation is enhanced, then for an effective preconditioner $\mathbf{M}_C$ this requires knowledge of the value of the correction vector on the (internal) subdomain boundaries. As the correction vector is the unknown vector, that is not practical.
The two other processes, with enhancement on the intermediate and highest level, however, first compute an approximate solution for the correction equation in an enhanced subspace built by powers of $\mathbf{M}_C^{-1}\ (\ \mathbf{A}_C - \theta\,\mathbf{I}_0\ )$ times a vector. Then an effective preconditioner can be constructed (§5.2.3).

For the enhanced correction equation it was shown in §3.3.2 of chapter 3 how to incorporate a preconditioner in the correction equation (5.11). In order to accomodate also for an unbalanced $\underset{\sim}{\mathbf{u}}$, we will discuss here how a preconditioner can be incorporated in correction equation (5.12). Similar to [46, §4.1] and [47, §3.1.1] first a 1-step approximation is considered in §5.4.1. This makes it easier to emphasize the difference by means of an example in §5.4.2. It also facilitates the interpretation of the approximate solution of (5.12) with a preconditioned Krylov method later on in §5.4.3.

### 5.4.1 1-step approximation

For the Jacobi-Davidson process with enhanced correction equation the 1-step approximation is given by (cf. step 1 in §3.3.3 of chapter 3):

$$\underset{\sim}{\mathbf{t}}^{(0)} = -\mathbf{P}'\,\mathbf{M}_C^{-1}\,\underline{\mathbf{r}} \quad \text{with} \quad \mathbf{P}' \equiv \mathbf{I} - \frac{\mathbf{M}_C^{-1}\mathbf{u}_0\mathbf{u}_0^*}{\mathbf{u}_0^*\mathbf{M}_C^{-1}\mathbf{u}_0}. \tag{5.16}$$

By premultiplying from the left with a preconditioner $\mathbf{M}_C \approx \mathbf{A}_C - \theta\,\mathbf{I}_0$ and imposing that the approximate correction vector is orthogonal to $\mathbf{u}_0$, equation (5.13) yields a 1-step approximation for the Jacobi-Davidson process with enhanced eigenvector equation:

$$\underset{\sim}{\mathbf{t}}^{(0)} = -\mathbf{P}'\,\mathbf{M}_C^{-1}\,\underset{\sim}{\mathbf{r}} \quad \text{with} \quad \mathbf{P}' \equiv \mathbf{I} - \frac{\mathbf{M}_C^{-1}\mathbf{u}_0\mathbf{u}_0^*}{\mathbf{u}_0^*\mathbf{M}_C^{-1}\mathbf{u}_0}. \tag{5.17}$$

The only difference between (5.16) and (5.17) is the residual: for $\underset{\sim}{\mathbf{u}} \neq \underline{\mathbf{u}}$ the residuals $\underline{\mathbf{r}}$ and $\underset{\sim}{\mathbf{r}}$ are not equal. In that case the solutions of (5.16) and (5.17) may differ. The appearance of an unbalanced $\underset{\sim}{\mathbf{u}} \neq \underline{\mathbf{u}}$ in the process with enhanced eigenvector equation is very likely when the correction equation is not solved exactly. This is illustrated by the example in §5.4.2. It shows also why it may be attractive to allow for an unbalanced search subspace for Jacobi-Davidson as in the process with an enhanced eigenvector equation.

### 5.4.2 Example

Suppose that the process with the enhanced correction equation is started with $\mathbf{V}^{(0)} \equiv \mathbf{u}^{(0)}$ and the process with the enhanced eigenvector equation with $\underset{\sim}{\mathbf{V}}^{(0)} \equiv \underline{\mathbf{u}}^{(0)}$, where $\underline{\mathbf{u}}^{(0)}$ is the balanced enhancement $\underline{\mathbf{u}}^{(0)}$ of $\mathbf{u}^{(0)}$. Then we have that

$$\theta_0 = \frac{(\mathbf{u}^{(0)})^* \mathbf{A} \mathbf{u}^{(0)}}{(\mathbf{u}^{(0)})^* \mathbf{u}^{(0)}}, \tag{5.18}$$

for both processes. For simplicity of notation, let $\mathbf{B}_{C_0} \equiv \mathbf{A}_{C_0} - \theta_0 \mathbf{I}_0$ with coupling $C_0$. Given some preconditioner $\mathbf{M}_{C_0} \approx \mathbf{B}_{C_0}$, the 1-step approximations (5.16) and (5.17) yield the same approximate correction vector $\underset{\sim}{\mathbf{t}}^{(0)} = -\mathbf{P}^{(0)} \mathbf{M}_{C_0}^{-1} \mathbf{B}_{C_0} \underline{\mathbf{u}}^{(0)}$, with

$$\mathbf{P}^{(0)} \equiv \mathbf{I} - \frac{\mathbf{M}_{C_0}^{-1} \mathbf{u}_0^{(0)} (\mathbf{u}_0^{(0)})^*}{(\mathbf{u}_0^{(0)})^* \mathbf{M}_{C_0}^{-1} \mathbf{u}_0^{(0)}}.$$

Note that in general $\underset{\sim}{\mathbf{t}}^{(0)}$ is unbalanced. The process with the enhanced correction equation deals with this unbalanced vector $\underset{\sim}{\mathbf{t}}^{(0)}$ by restricting it to $\mathbf{t}^{(0)}$. With this vector the search subspace is expanded to $\mathbf{V}^{(1)} \equiv \text{span}\left(\mathbf{u}^{(0)}, \mathbf{t}^{(0)}\right)$. The new search subspace $\underset{\sim}{\mathbf{V}}^{(1)}$ of the process with the enhanced eigenvector equation is spanned by $\underline{\mathbf{u}}^{(0)}$ and $\underset{\sim}{\mathbf{t}}^{(0)}$.

The next outer iteration a new approximate eigenpair is determined. As the restriction of $\underset{\sim}{\mathbf{V}}^{(1)}$ is equal to $\mathbf{V}^{(1)}$, the interaction matrices of the two processes are identical. Because of this, both processes determine the same new approximate eigenvalue, the Ritz value $\theta_1$. Let the corresponding Ritz vector be $\mathbf{u}^{(1)}$. The process with the enhanced correction equation enhances this vector into a balanced vector $\underline{\mathbf{u}}^{(1)}$ that can be written as

$$\underline{\mathbf{u}}^{(1)} = \alpha \, \underline{\mathbf{u}}^{(0)} + \beta \, \underline{\mathbf{t}}^{(0)}$$

for some $\alpha$ and $\beta$. These coefficients $\alpha$ and $\beta$ also describe the unbalanced enhanced Ritz vector $\underset{\sim}{\mathbf{u}}^{(1)}$ of the process with enhanced eigenvector equation:

$$\underset{\sim}{\mathbf{u}}^{(1)} = \alpha \, \underline{\mathbf{u}}^{(0)} + \beta \, \underset{\sim}{\mathbf{t}}^{(0)},$$

so, for this case, an approximate solution of the correction equation leads to an unbalanced $\underset{\sim}{\mathbf{u}} \neq \underline{\mathbf{u}}$.

If one proceeds, the process with enhanced correction equation computes a new approximate correction vector equal to

$$-\mathbf{P}^{(1)} \mathbf{M}_{C_1}^{-1} \mathbf{B}_{C_1} \underline{\mathbf{u}}^{(1)} = -\alpha \, \mathbf{P}^{(1)} \mathbf{M}_{C_1}^{-1} \mathbf{B}_{C_1} \underline{\mathbf{u}}^{(0)} - \beta \, \mathbf{P}^{(1)} \mathbf{M}_{C_1}^{-1} \mathbf{B}_{C_1} \underline{\mathbf{t}}^{(0)}. \tag{5.19}$$

The new approximate correction vector for the other process can be written as an operator applied to the start vector $\underline{\mathbf{u}}^{(0)}$:

$$\begin{aligned} -\mathbf{P}^{(1)} \mathbf{M}_{C_1}^{-1} \mathbf{B}_{C_1} \underset{\sim}{\mathbf{u}}^{(1)} &= -\alpha \, \mathbf{P}^{(1)} \mathbf{M}_{C_1}^{-1} \mathbf{B}_{C_1} \underline{\mathbf{u}}^{(0)} - \beta \, \mathbf{P}^{(1)} \mathbf{M}_{C_1}^{-1} \mathbf{B}_{C_1} \underset{\sim}{\mathbf{t}}^{(0)} \\ &= \left( -\alpha \, \mathbf{P}^{(1)} \mathbf{M}_{C_1}^{-1} \mathbf{B}_{C_1} + \beta \, \mathbf{P}^{(1)} \mathbf{M}_{C)1}^{-1} \mathbf{B}_{C_1} \mathbf{P}^{(0)} \mathbf{M}_{C_0}^{-1} \mathbf{B}_{C_0} \right) \underline{\mathbf{u}}^{(0)}. \end{aligned} \tag{5.20}$$

We consider a coupling $C_i$ that is tuned as in chapter 3. As already remarked in §5.2.3, such a coupling damps out errors by increasing powers of $\mathbf{M}_{C_i}^{-1}\,\mathbf{B}_{C_i}$. Furthermore, if $\theta_1$ is close to $\theta_0$, which is the case when Jacobi-Davidson is in the region of convergence, then for the optimized coupling, $C_1$ is close to $C_0$ and, as a result, $\mathbf{B}_{C_1} \approx \mathbf{B}_{C_0}$ and $\mathbf{M}_{C_1} \approx \mathbf{M}_{C_0}$. Because of this, in equation (5.20) remaining error components from the previous outer iteration are damped in the next outer iteration. In equation (5.19), however, the damping of error components in the next outer iteration is disturbed.

From this example we learn that, if approximate solutions for the correction equation are obtained from a 1-step approximation, then we may expect that the process with the enhanced eigenvector equation converges faster than the process with the enhanced correction equation.

### 5.4.3 Higher order approximations

From §5.3.2.2 we know that for exact solutions of the correction equation the processes with enhanced correction equation and enhanced eigenvector equation are identical. The example of §5.4.2 resulted in our expectation that for 1-step approximations the process with enhanced eigenvector equation converges faster than the process with enhanced correction equation. The question remains how the two processes are related for higher order approximate solutions of the correction equation.

The process with enhanced correction equation computes such a solution with a preconditioned Krylov method. For that purpose it was shown in §3.3.2 of chapter 3 how to incorporate a preconditioner in the correction equation (5.11):

$$\mathbf{P}'\,\mathbf{M}_C^{-1}\,(\,\mathbf{A}_C - \theta\,\mathbf{I}_0\,)\,\mathbf{P}'\,\underset{\sim}{\mathbf{t}} = \mathbf{P}'\,\mathbf{M}^{-1}\,\underline{\mathbf{r}} \quad \text{with} \quad \mathbf{P}' \equiv \mathbf{I} - \frac{\mathbf{M}_C^{-1}\mathbf{u}_0\mathbf{u}_0^*}{\mathbf{u}_0^*\mathbf{M}_C^{-1}\mathbf{u}_0}. \tag{5.21}$$

The situation for the process with enhanced eigenvector equation is considered now.

A higher order approximation for a solution of (5.12) can be obtained by considering not only $\underset{\sim}{\mathbf{t}}^{(0)}$ from (5.17) but also terms defined by the sequence

$$\underset{\sim}{\mathbf{z}}^{(i)} = \mathbf{P}'\,\mathbf{M}_C^{-1}\,(\mathbf{A}_C - \theta\,\mathbf{I}_0\,)\,\underset{\sim}{\mathbf{z}}^{(i-1)} \quad \text{with} \quad \mathbf{P}' \equiv \mathbf{I} - \frac{\mathbf{M}_C^{-1}\mathbf{u}_0\mathbf{u}_0^*}{\mathbf{u}_0^*\mathbf{M}_C^{-1}\mathbf{u}_0}$$

and $\underset{\sim}{\mathbf{z}}^{(0)} = \underset{\sim}{\mathbf{t}}^{(0)}$ for $i = 1, 2, \ldots$ (cf. [47, §3.1.1]). These vectors span the Krylov subspace $\mathcal{K}_m(\mathbf{P}'\,\mathbf{M}_C^{-1}\,(\mathbf{A}_C - \theta\,\mathbf{I}_0\,), \mathbf{P}'\,\mathbf{M}_C^{-1}\,\underset{\sim}{\mathbf{r}})$. Note that the coupling $C$ in §5.2.3 is chosen such that (most) error components induced by the splitting $\mathbf{A}_C - \theta\,\mathbf{I}_0 = \mathbf{M}_C - \mathbf{N}_C$ are damped out by increasing powers of $\mathbf{M}_C^{-1}\,(\mathbf{A}_C - \theta\,\mathbf{I}_0\,)$. So for larger $m$ a better approximation $\underset{\sim}{\mathbf{t}}^{(m)}$ to the solution of (5.12) can be extracted from the Krylov subspace, for instance, with GMRES which computes the solution in $\mathcal{K}_m$ that has a minimal residual in $\ell_2$-norm.

In fact, in this way, with a Krylov method an approximate solution $\underset{\sim}{\mathbf{t}}^{(m)}$ is computed to the preconditioned correction equation

$$\mathbf{P}'\,\mathbf{M}_C^{-1}\,(\mathbf{A}_C - \theta\,\mathbf{I}_0\,)\,\mathbf{P}'\,\underset{\sim}{\mathbf{t}} = \mathbf{P}'\,\mathbf{M}_C^{-1}\,\underset{\sim}{\mathbf{r}} \quad \text{with} \quad \mathbf{P}' \equiv \mathbf{I} - \frac{\mathbf{M}_C^{-1}\mathbf{u}_0\mathbf{u}_0^*}{\mathbf{u}_0^*\mathbf{M}_C^{-1}\mathbf{u}_0}. \tag{5.22}$$

Again, it can be shown that for $\underset{\sim}{\mathbf{u}} = \underline{\mathbf{u}}$ the preconditioned correction equations (5.21) and (5.22) are identical.

As for higher order solutions of the correction equations (5.21) and (5.22) the error components are damped more, it is to be expected that the difference between the two processes, as illustrated in §5.4.2, becomes less significant: damping due to the outerloop in the process with enhanced eigenvector equation then has a smaller contribution to the overall process. This expectation is verified numerically in the next section.

## 5.5 Numerical experiments

The numerical experiments presented in this section are intended to illustrate how the process with enhanced correction equation and the process with enhanced eigenvector equation are related for approximate solutions of the correction equation. In addition, the first experiment of §5.5.1 also includes the black box approach, where we apply the Jacobi-Davidson QZ (JDQZ) method [44, 26] to the enhanced eigenvalue problem (5.7) (see §5.2.2). For approximate solutions of the correction equation, we expect (§5.4) that the process with enhanced eigenvector equation converges faster, we will verify this in §5.5.2. Then, §5.5.3 will show how one may take advantage of this knowledge when an eigenvalue problem is solved by massively parallel computations.

In all experiments, Jacobi-Davidson is applied to the discretized eigenvalue problem for the two dimensional Laplace operator on a domain $\Omega$. The domain is covered by a grid of uniform mesh size and the matrix $\mathbf{A}$ in (5.4) represents the discretization of the Laplace operator via central differences. Our goal is the eigenvalue of smallest absolute value and the corresponding eigenvector. The startvector of Jacobi-Davidson is the parabola shaped vector (3.57) from §3.5.1 in chapter 3.

For the construction of the preconditioner $\mathbf{M}_C$, $\Omega$ is decomposed into subdomains. Each subdomain is covered by a subgrid. The enhancements from §5.2 consists of adding an extra row of gridpoints at all four borders of each subgrid, the function values defined on these extra points correspond to the $\widetilde{\cdot}$-parts of the enhancement of the vector that represents the function on the original grid. The coupling $C$ is optimized simply as explained in §3.4.4 of chapter 3.

The experiments are done with MATLAB 5.3.0 on a Sun Sparc Ultra 5 workstation.

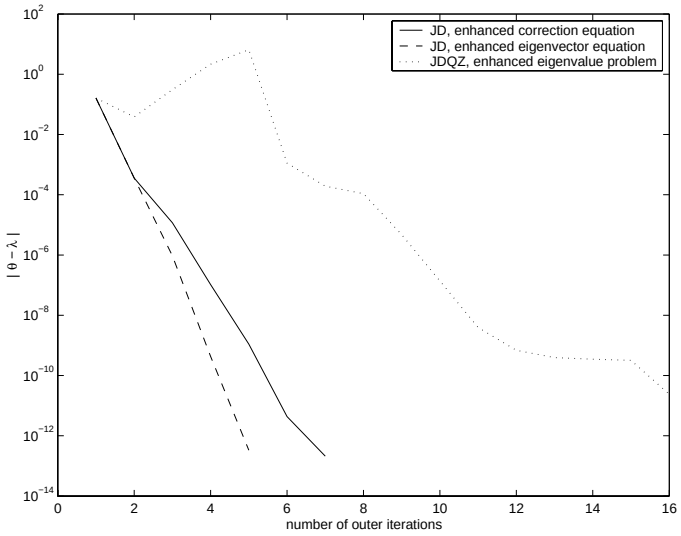
### 5.5.1 Black box approach

We start with a numerical example that compares a black box approach where we apply the Jacobi-Davidson QZ (JDQZ) method [44, 26] to the enhanced eigenvalue problem (5.7) (see §5.2.2) with Jacobi-Davidson with enhanced correction equation and Jacobi-Davidson with enhanced eigenvector equation for approximate solves of the correction equation.

We take $\Omega = [0, 2] \times [0, 1]$ and cover this domain by a $63 \times 31$ grid. The domain $\Omega$ is decomposed into two subdomains $\Omega_1$ and $\Omega_2$ such that $\Omega_1$ is covered by a $(26 + 1) \times 31$ subgrid and $\Omega_2$ by a $(37+1)\times 31$ subgrid. As for the enhanced eigenvalue problem (5.7) we can not change the matrix $C$ during the iteration process we construct this $C$ only once for the $\theta_0$ given by (5.18) for all three cases. We compute approximate solves of the correction equation with left preconditioned GMRES(3). For all three processes the target is set to $0$.

Fig. 5.2 presents the results. Shown are the errors $|\theta-\lambda|$ of the approximate eigenvalue $\theta$ as a function of the number of outer iterations for Jacobi-Davidson with enhanced correction equation, Jacobi-Davidson with enhanced eigenvector equation, and JDQZ with enhanced eigenvalue problem. It can be observed that the latter process needs far more iterations for convergence. The bump in this convergence plot may be due to the extra eigenvalue $\infty$ of (5.7). A target $-13$ instead of $0$ resulted in a modest improvement, without bump, but still the convergence is quite slow compared to the other two processes. Furthermore, in case of the enhanced eigenvalue problem we can not adjust the coupling matrix during the process, where for the other two processes we can. For the special preconditioner of §5.2.3 this is of interest as the coupling matrix depends on $\theta$: $C = C(\theta)$.

FIGURE 5.2. *Convergence history of Jacobi-Davidson applied to the discretized eigenvalue problem for the two dimensional Laplace operator on the domain* $[0, 1]^2$. *Three cases are considered: Jacobi-Davidson with enhancement of the correction equation, Jacobi-Davidson with enhancement of the eigenvector equation, and JDQZ applied to the enhanced eigenvalue problem. Approximate solutions of the correction equation are obtained with left preconditioned GMRES(3). See for explanation* §*5.5.1.*



### 5.5.2 Effect of the accuracy of the correction vector

For exact solution of the correction equation we know from §5.3.2.2 that the process with enhanced correction equation and the process with enhanced eigenvector equation are equivalent. The example of §5.4.2 showed that an approximate solution of the preconditioned correction equation affects the two processes differently. At the end of §5.4.3 it was argued that for approximate solutions of higher accuracy this effect becomes less important. This phenomenon is considered now by a numerical example.

Here $\Omega = [0,1]^2$ and $\Omega$ is covered by a $200 \times 200$ grid. This domain is decomposed into $8 \times 8$ square subdomains. So each subdomain is covered by a $25 \times 25$ subgrid. Approximate solutions to the correction equations (5.21) and (5.22) respectively of the two processes are solved by right preconditioned GMRES($m$). The number of GMRES-steps $m$ is kept fixed for each outer iteration. For a reduction of computational overhead this approach is not recommended, but some tolerance strategy is advisable [26, §4]. Here our objective is a comparison of two processes with nearly the same computational costs per outer iteration. As the difference of these processes is expected to depend on the accuracy of the approximate solution of the correction equation, three values of $m$ are considered: $m = 4, 8,$ and $16$.

Table 5.1 shows the convergence history of the two processes (the one with enhanced correction equation on the left, the other one with enhanced eigenvector equation on the right). For each process we have listed: the error in the eigenvalue (columns 2 and 5) and the $\ell_2$-norm of the residual (columns 3 and 6). We checked that for the same number of outer iterations both processes need nearly the same amount of flops.

For all three values of $m$ the error in the eigenvalue at step 2 shows no difference for both processes. This is explained by example §5.4.2: because of the startvector both processes compute the same correction vector $\widetilde{\mathbf{t}}$, this results in the same interaction matrix at step 2 from which the same approximate eigenvalue $\theta$ is determined. Note that for this argument the accuracy of $\widetilde{\mathbf{t}}$ does not matter, the point is that $\widetilde{\mathbf{t}}$ is not different here for the two processes. However, for $m = 4$ and $m = 8$ the values of the residual norm in column 3 and 5 differ at step 2. For $m = 4$ the value is about 50 times smaller for the process with enhanced eigenvector equation than the process with enhanced correction equation, for $m = 8$ about 40 times, and for $m = 16$ no difference can be observed. So for a more accurate correction vector the difference diminishes, as anticipated in §5.4.

After step 2 also the approximate eigenvalue is different for the two processes. From the results in the table one observes that if the correction equations are solved with GMRES($4$), that is solutions are of low accuracy, then the process with enhanced correction equation indeed needs significantly more outer iterations than the process with enhanced eigenvector equation for convergence.

TABLE 5.1. *Convergence history of Jacobi-Davidson applied to the discretized eigenvalue problem for the two dimensional Laplace operator on the domain* $[0,1]^2$*. Two cases are considered: Jacobi-Davidson with enhancement of the correction equation and Jacobi-Davidson with enhancement of the eigenvector equation, for solutions of the correction equation of different accuracy. See for explanation* §*5.5.2.*

| | process with enhanced correction equation | | process with enhanced eigenvector equation | |
|---|---|---|---|---|
| step | $\theta - \lambda$ | $\|\mathbf{r}\|_2$ | $\theta - \lambda$ | $\|\mathbf{r}\|_2$ |
| | | GMRES(4) | | |
| 1 | -2.61e-01 | 6.23e+00 | -2.61e-01 | 6.23e+00 |
| 2 | -4.87e-03 | 1.14e+01 | -4.87e-03 | 2.23e-01 |
| 3 | -2.24e-04 | 4.00e+00 | -1.94e-05 | 1.13e-01 |
| 4 | -1.49e-06 | 3.19e-01 | -1.96e-08 | 1.13e-04 |
| 5 | -3.85e-08 | 4.24e-02 | -3.75e-11 | 6.59e-05 |
| 6 | -1.34e-09 | 8.59e-03 | -9.38e-13 | 1.21e-06 |
| 7 | -3.35e-11 | 1.67e-03 | -7.07e-13 | 1.34e-07 |
| 8 | -1.22e-12 | 2.04e-04 | -1.95e-13 | 4.43e-09 |
| 9 | -2.03e-13 | 2.23e-05 | -9.91e-13 | 1.47e-10 |
| 10 | 2.72e-12 | 2.66e-06 | | |
| 11 | 1.49e-12 | 2.42e-07 | | |
| 12 | -6.44e-12 | 3.23e-08 | | |
| 13 | 1.84e-12 | 2.70e-09 | | |
| 14 | -6.48e-12 | 3.42e-10 | | |
| | | GMRES(8) | | |
| 1 | -2.61e-01 | 6.23e+00 | -2.61e-01 | 6.23e+00 |
| 2 | -5.16e-06 | 4.67e-01 | -5.16e-06 | 1.12e-02 |
| 3 | -6.00e-10 | 6.65e-03 | -8.31e-11 | 1.59e-04 |
| 4 | -2.70e-13 | 9.88e-05 | -6.11e-13 | 4.78e-07 |
| 5 | -2.59e-13 | 2.43e-06 | -4.83e-13 | 5.01e-10 |
| 6 | 1.99e-13 | 1.61e-08 | | |
| 7 | -3.87e-13 | 3.39e-10 | | |
| | | GMRES(16) | | |
| 1 | -2.61e-01 | 6.23e+00 | -2.61e-01 | 6.23e+00 |
| 2 | -2.22e-07 | 1.13e-02 | -2.22e-07 | 1.13e-02 |
| 3 | -5.33e-14 | 9.41e-07 | -6.18e-13 | 2.41e-09 |
| 4 | -5.33e-14 | 1.10e-10 | -5.83e-13 | 2.70e-11 |

### 5.5.3 The number of subdomains

For the application of Jacobi-Davidson to a realistic large scale eigenvalue problem most computational work is needed for the correction equation. In addition, also the storage of the matrix and vectors may be a problem. Then a parallel approach is advisable. Computation of approximate solutions to the correction equation with a preconditioner based on domain decomposition makes this possible. We illustrate that in such a situation the difference between the processes with enhanced correction equation and enhanced eigenvector equation as observed in §5.5.2 is of interest.

The experiment from §5.5.2 is considered for different decompositions of the domain $[0,1]^2$ with different numbers of gridpoints (hence different gridspacings). The number of gridpoints per subdomain is taken fixed: each subdomain is covered by a $25 \times 25$ subgrid of uniform mesh size. Such a typical situation may occur when the gridsize of a subdomain is limited because of memory and/or computational time in a parallel computing environment. The domain $[0,1]^2$ is decomposed in three different ways:

- $4 \times 4$ (= 16) subdomains (#gridpoints = order of $\mathbf{A}$ = $4^2 \cdot 25^2 = 10.000$)
- $8 \times 8$ (= 64) subdomains (#gridpoints = order of $\mathbf{A}$ = $8^2 \cdot 25^2 = 40.000$)
- $16 \times 16$ (= 256) subdomains (#gridpoints = order of $\mathbf{A}$ = $16^2 \cdot 25^2 = 160.000$)

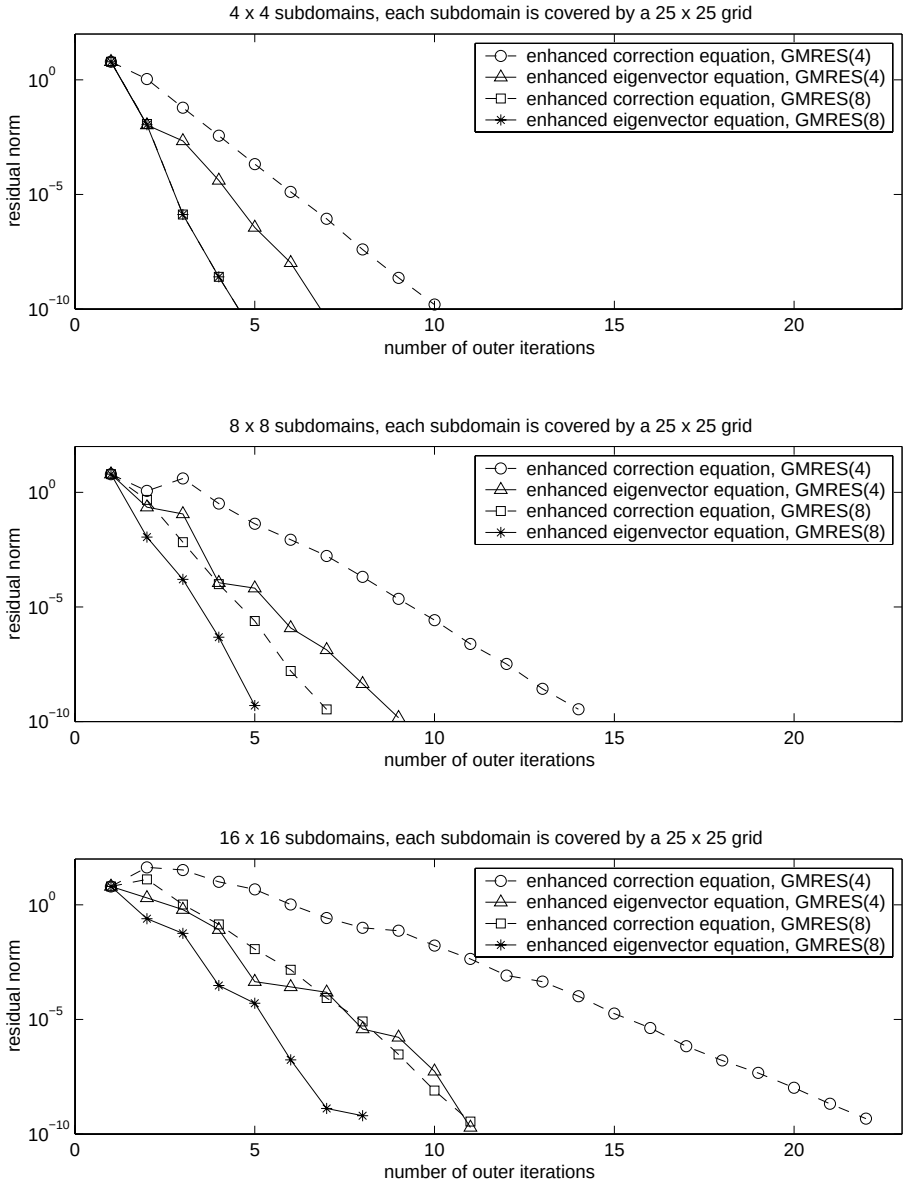
For the fixed number $m$ of right preconditioned GMRES steps we have taken $m = 4$ and $m = 8$.

Results are presented in Fig. 5.3 ($4 \times 4$ subdomains in the top picture, $8 \times 8$ subdomains in the middle picture, and $16 \times 16$ subdomains in the bottom picture). The pictures show the convergence of the residual norm for the process with enhanced correction equation (indicated by dashed lines) and the process with enhanced eigenvector equation (indicated by solid lines) as a function of the number of outer iterations.

From the pictures it can be observed that given a value of $m$ the difference between the processes increases when the number of subdomains increases. That is explained as follows: for a larger number of subdomains more error components are induced by the matrix splitting $\mathbf{M}_C - \mathbf{N}_C = \mathbf{A}_C - \theta \mathbf{I}_0$, then for the same value of $m$ the approximate correction vector contains relatively more less damped error components and these components are treated better by the process with the enhanced eigenvector equation in the outerloop.

In case of a realistic large scale eigenvalue problem when a parallel approach is advisable, the outcome of this experiment suggests to use the process with enhanced eigenvector equation.

FIGURE 5.3. *Convergence history of Jacobi-Davidson applied to the discretized eigenvalue problem for the two dimensional Laplace operator on the domain* $[0,1]^2$. *Two cases are considered: Jacobi-Davidson with enhancement of the correction equation and Jacobi-Davidson with enhancement of the eigenvector equation on different grids and for approximate solutions of the correction equation. See for explanation* §*5.5.3.*



## 5.6 Conclusions

In this chapter three different levels of enhancement in the Jacobi-Davidson method have been considered for the computation of eigenvalues and eigenvectors of a matrix. These enhancements serve to incorporate a preconditioner based on domain decomposition in the correction equation of Jacobi-Davidson.

For exact solutions of the correction equation the three approaches are equivalent. But for approximate solutions of the correction equation that is not the case. Because of the specific structure of the preconditioner, it is optimized for damping error components of powers of the preconditioned matrix, two levels are of importance. It is shown that for low accurate solutions of the correction equation one of these two approaches should be preferred when the number of subdomains is large. Such a situation may occur if a large scale eigenvalue problem needs a massively parallel treatment.

# Bibliography


[1] A. ALAVI, J. KOHANOFF, M. PARRINELLO, AND D. FRENKEL, *Ab initio molecular dynamics with excited electrons*, Phys. Rev. Lett., 73 (1994), pp. 2599–2602.

[2] W. E. ARNOLDI, *The principle of minimized iterations in the solution of the matrix eigenvalue problem*, Quart. Appl. Math., 9 (1951), pp. 17–29.

[3] A. J. C. BELIËN, *Wave dynamics and heating of coronal magnetic flux tubes*, Ph.D. thesis, Vrije Universiteit Amsterdam, The Netherlands, 1996.

[4] P. E. BJØRSTAD AND O. B. WIDLUND, *To overlap or not to overlap: a note on a domain decomposition method for elliptic problems*, SIAM J. Sci. Comput., 10 (1989), pp. 1053–1061.

[5] J. G. BLOM AND J. G. VERWER, *VLUGR3: a vectorizable adaptive grid solver for PDEs in 3D. I. algorithmic aspects and applications*, Appl. Numer. Math., 16 (1994), pp. 129–156.

[6] E. BRAKKEE, *Domain decomposition for the incompressible Navier-Stokes Equations*, Ph.D. thesis, Technische Universiteit Delft, Delft, The Netherlands, 1996.

[7] E. BRAKKEE AND P. WILDERS, *The influence of interface conditions on convergence of Krylov-Schwarz domain decomposition for the advection-diffusion equation*, J. Sci. Comput., 12 (1997), pp. 11–30.

[8] X.-C. CAI, W. D. GROPP, AND D. E. KEYES, *A comparison of some domain decomposition and ILU preconditioned iterative methods for nonsymmetric elliptic problems*, Num. Lin. Alg. Appl. 1 (1994), pp. 477–504.

[9] X.-C. CAI, W. D. GROPP, D. E. KEYES, R. G. MELVIN, AND D. P. YOUNG, *Parallel Newton-Krylov-Schwarz algorithms for the transonic full potential equation*, SIAM J. Sci. Comput., 19:246–265, 1998.

[10] X.-C. CAI AND D. E. KEYES, *Nonlinearly preconditioned inexact Newton algorithms*, submitted to SIAM J. Sci. Comput.

[11] T. F. CHAN AND D. GOOVAERTS, *On the relationship between overlapping and nonoverlapping domain decomposition methods*, SIAM J. Matrix Anal. Appl., 13 (1992), pp. 663–670.

[12] T. F. CHAN AND T. P. MATHEW, *Domain decomposition algorithms*, Acta Numerica (1994), pp. 61–143.

[13] R. CHEN AND H. GUO, *Benchmark calculations of bound states of $HO_2$ via basic Lanczos algorithm*, Chem. Phys. Lett., 277 (1997), pp. 191–198.

[14] M. CROUZEIX, B. PHILIPPE, AND M. SADKANE, *The Davidson method*, SIAM J. Sci. Comput., 15 (1994), pp .62–76.

[15] E. R. DAVIDSON, *The iterative calculation of a few of the lowest eigenvalues and corresponding eigenvectors of large real-symmetric matrices*, J. Comput. Phys., 17 (1975), pp. 87–94.

[16] Q. DENG, *An analysis for a nonoverlapping domain decomposition iterative procedure*, SIAM J. Sci. Comput., 18 (1997), pp. 1517–1525.

[17] J. DESCLOUX, J.-L. FATTEBERT, AND F. GYGI, *Rayleigh quotient iteration, an old recipe for solving modern large-scale eigenvalue problems*, Computers in Physics, 12 (1998), pp. 22–27.

[18] E. DE STURLER AND D. R. FOKKEMA, *Nested Krylov methods and preserving the orthogonality*, N. Duane Melson, T. A. Manteuffel, and S. F. McCormick, editors, *Sixth Copper Mountain Conference on Multigrid Methods*, NASA Conference Publication 3224, Part 1 (1993), pp. 111–125.

[19] E. DE STURLER, *Truncation strategies for optimal Krylov subspace methods*, SIAM J. Numer. Anal., 36 (1999), pp. 864–889.

[20] H. A. DIJKSTRA, M. J. SCHMEITS, AND C. A. KATSMAN, *Internal variability of the North Atlantic wind-driven ocean circulation*, Surveys in Geophysics, 20 (1999), pp. 463–503.

[21] I. S. DUFF, R. G. GRIMES, AND J. G. LEWIS, *Users' guide for the Harwell-Boeing sparse matrix collection*, Technical Report TR/PA/92/86", CERFACS, Toulouse, France, 1992. Also RAL Technical Report RAL 92-086.

[22] J.-L. FATTEBERT, *Une méthode numérique pour la résolution des problèmes aux valeurs propres liés au calcul de structure électronique moléculaire*, Ph.D. thesis, Ecole Polytechnique Fédérale de Lausanne, France, 1997.

[23] J. G. F. FRANCIS, *The QR transformation, a unitary analogue to the LR transformation–part 1*, Comp. J., 4 (1961), pp. 265–271.

[24] J. G. F. FRANCIS, *The QR transformation–part 2*, Comp. J., 4 (1961), pp. 332–345.

[25] D. R. FOKKEMA, G. L. G. SLEIJPEN, AND H. A. VAN DER VORST, *Accelerated inexact Newton schemes for large systems of nonlinear equations*, SIAM J. Sci. Comput., 19 (1998), pp. 657–674.

[26] D. R. FOKKEMA, G. L. G. SLEIJPEN, AND H. A. VAN DER VORST, *Jacobi-Davidson style QR and QZ algorithms for the reduction of matrix pencils*, SIAM J. Sci. Comput., 20 (1999), pp. 94–125.

[27] J. P. GOEDBLOED, *Plasma-vacuum interface problems in magnetohydrodynamics*, Physica 12D (1984), pp. 107–132.

[28] A. HADJIDIMOS, D. NOUTSOS, AND M. TZOUMAS, *Nonoverlapping domain decomposition: a linear algebra viewpoint*, Math. Comput. Simulation, 51 (2000), pp. 597–625.

[29] Z. JIA AND G. W. STEWART, *An analysis of the Rayleigh-Ritz method for approximating eigenspaces*, Technical Report TR-99-24/TR-4015, Department of Computer Science, University of Maryland, USA, 1999.

[30] W. KERNER, J. P. GOEDBLOED, G. T. A. HUYSMANS, S. POEDTS, AND E. SCHWARZ, *CASTOR: normal-mode analysis of resistive MHD plasmas*, J. Comp. Phys., 142 (1998), pp. 271–303.

[31] C. LANCZOS, *An iteration method for the solution of the eigenvalue problem of linear differential and integral operators*, Journal of Research, Nat. Bur. Stand., 45 (1950), pp. 255–282.

[32] G. LUBE, L. MÜLLER, AND F. C. OTTO, *A non-overlapping domain decomposition method for the advection-diffusion problem*, Computing, 64 (2000), pp. 49–68.

[33] R. B. MORGAN AND D. S. SCOTT, *Generalizations of Davidson's method for computing eigenvalues of sparse symmetric matrices*, SIAM J. Sci. Stat. Comput.,7 (1986), pp. 817–825.

[34] R. B. MORGAN, *On restarting the Arnoldi method for large nonsymmetric eigenvalue problems*, Math. Comp., 65 (1996), pp. 1213–1230.

[35] C. B. MOLER AND G. W. STEWART, *An algorithm for generalized matrix eigenvalue problems*, SIAM J. Num. Anal., 10 (1973), pp. 241–256.

[36] J. OLSEN, P. JØRGENSEN, AND J. SIMONS, *Passing the one-billion limit in full configuration-interaction (FCI) calculations*, Chem. Phys. Lett.,169 (1990), pp. 463–472.

[37] B. N. PARLETT, *The symmetric eigenvalue problem*, Prentice-Hall, Englewood Cliffs, N.J., 1980.

[38] TH. ROTTNER, I. LENHARDT, G. ALEFELD, AND K. SCHWEIZERHOF, *Nonlinear structural finite element analysis using the preconditioned Lanczos method on serial and parallel computers*, BIT, 37 (1997), pp. 759–769.

[39] A. RUHE, *Rational Krylov, a practical algorithm for large sparse nonsymmetric matrix pencils*, SIAM J. Sci. Comput., 19 (1998), pp. 1535–1551.

[40] Y. SAAD, *Numerical methods for large eigenvalue problems*, Manchester University Press, Manchester, UK, 1992.

[41] Y. SAAD AND M. H. SCHULTZ, *GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM J. Sci. Stat. Comput., 7 (1986), pp. 856–869.

[42] M. SADKANE, *Block-Arnoldi and Davidson methods for unsymmetric large eigenvalue problems*, Numer. Math., 64 (1993), pp. 195–211.

[43] H. A. SCHWARZ, *Gesammelte Mathematische Abhandlungen*, Vol. 2, pp. 133–143, Springer, Berlin, 1890. First published in Vierteljahrsschrift der Naturforschenden Gesellschaft in Zürich, 15 (1870), pp. 272–286.

[44] G. L. G. SLEIJPEN, A. G. L. BOOTEN, D. R. FOKKEMA, AND H. A. VAN DER VORST, *Jacobi-Davidson type methods for generalized eigenproblems and polynomial eigenproblems*, BIT, 36 (1996), pp. 595–633.

[45] G. L. G. SLEIJPEN AND H. A. VAN DER VORST, *The Jacobi-Davidson method for eigenvalue problems and its relation with accelerated inexact Newton scheme*. in Iterative Methods in Linear Algebra II, S. D. Margenov, and P. S. Vassilevski, eds., IMACS Ann. Comput. Appl. Math., 3:377–389, 1996.

[46] G. L. G. SLEIJPEN AND H. A. VAN DER VORST, *A Jacobi-Davidson iteration method for linear eigenvalue problems*, SIAM J. Matrix Anal. Appl., 17 (1996), pp. 401–425.

[47] G. L. G. SLEIJPEN, H. A. VAN DER VORST, AND E. MEIJERINK, *Efficient expansion of subspaces in the Jacobi-Davidson method for standard and generalized eigenproblems*, Electron. Trans. Numer. Anal. 7 (1998), pp. 75–89.

[48] G. L. G. SLEIJPEN AND F. W. WUBS, *Effective preconditioning techniques for eigenvalue problems*, preprint no. 1117, Dep. Math., University Utrecht, 1999.

[49] B. F. SMITH, P. E. BJØRSTAD, AND W. D. GROPP, *Domain decomposition: parallel multilevel methods for elliptic partial differential equations*, Cambridge University Press, 1996.

[50] D. C. SORENSEN, *Truncated QZ methods for large scale generalized eigenvalue problems*, ETNA, 7 (1998), pp. 141–162.

[51] D. C. SORENSEN AND C. YANG, *A truncated RQ-iteration for large scale eigenvalue calculations*, SIAM J. Matrix Anal. Appl., 19 (1998), pp. 1045–1073.

[52] A. STATHOPOULOS, Y. SAAD, AND C. F. FISCHER, *Robust preconditioning of large sparse symmetric eigenvalue problems*, J. Comput. Appl. Math., 64 (1995), pp. 197–215.

[53] A. STATHOPOULOS, Y. SAAD, AND K. WU, *Dynamic thick restarting of the Davidson, and the implicitly restarted Arnoldi methods*, SIAM J. Sci. Comput., 19 (1998), pp .227–245.

[54] H. SUN AND W. P. TANG, *An overdetermined Schwarz alternating method*, SIAM J. Sci. Stat. Comput. 17 (1996), pp. 884–905.

[55] K. H. TAN AND M. J. A. BORSBOOM, *On generalized Schwarz coupling applied to advection-dominated problems*, in Domain decomposition methods in scientific and engineering computing (University Park, PA, 1993), D. E. Keyes and J. C. Xu, eds., Amer. Math. Soc., Providence, RI, 1994, pp. 125–130.

[56] K. H. TAN, *Local coupling in domain decomposition*, Ph.D. thesis, Utrecht University, Utrecht, The Netherlands, 1995.

[57] W. P. TANG, *Generalized Schwarz splittings*, SIAM J. Sci. Stat. Comput., 13 (1992), pp. 573–595.

[58] R. S. VARGA, *Matrix iterative analysis*, Prentice-Hall Inc., Englewood Cliffs, N.J., 1962.

[59] H. A. VAN DER VORST AND C. VUIK, *GMRESR: a family of nested GMRES methods*, Num. Lin. Alg. Appl., 1 (1994), pp .369–386.

[60] C. VUIK, A. SEGAL, AND J. A. MEIJERINK, *An efficient preconditioned CG method for the solution of a class of layered problems with extreme contrasts in the coefficients*, J. Comput. Phys., 152 (1999), pp. 385–403.

[61] P. WILDERS AND E. BRAKKEE, *Schwarz and Schur: an algebraic note on equivalence properties*, SIAM J. Sci. Comput., 20 (1999), pp. 2297–2303.

[62] S. M. WILKINSON AND D. P. THAMBIRATNAM, *Mode coupling in the vibration response of asymmetric buildings*, Computers & Structures, 56 (1995), pp. 1039–1051.

[63] J. XU, *Iterative methods by space decomposition and subspace correction*, SIAM Review, 34 (1992), pp. 581–613.

[64] G. YAO AND R. E. WYATT, *A Krylov-subspace Chebyshev method and its application to pulsed laser-molecule interaction*, Chem. Phys. Lett., 239 (1995), pp. 207–216.

# Samenvatting

Oplossingen van eigenwaardeproblemen zijn nodig bij onderzoek naar een breed scala van fenomenen. Van aardbevingen tot zonnevlammen. Van kernfusie tot veranderingen in het klimaat.

Voor het beter kunnen begrijpen en voorspellen van zo'n fenomeen worden modellen ontwikkeld. Hiervoor worden vergelijkingen opgesteld waarvan oplossingen kenmerkend gedrag beschrijven. In een enkel geval zijn dergelijke vergelijkingen met het blote hoofd exact op te lossen. Realistische modellen zijn echter vaak zo complex dat een exacte oplossing moeilijk of überhaupt niet bepaald kan worden. Dan is men aangewezen op het gebruik van computers.

De meeste modellen bestaan uit *continue* vergelijkingen: binnen een bepaald gebied (denk aan een bak met water) beschrijft een oplossing het gedrag (de stroming van het water) overal (in elk punt van de bak). Voor het met een computer berekenen van een oplossing worden de continue vergelijkingen eerst *gediscretiseerd* (in de bak wordt een denkbeeldig rooster aangebracht, de continue vergelijkingen worden vervangen door *discrete* vergelijkingen waarvan oplossingen de stroming van het water op de roosterpunten beschrijven). De berekende oplossing benadert een oplossing van de continue vergelijkingen. Deze benadering zal, in het algemeen, beter zijn voor een fijnmaziger rooster. Een fijnmaziger rooster op hetzelfde gebied heeft echter een groter aantal roosterpunten, met als gevolg dat het aantal te berekenen onbekenden groter is.

Een typisch fenomeen waar een eigenwaardeprobleem in voorkomt is een systeem dat door een aandrijvende kracht kan gaan resoneren (denk aan een brug die mee gaat trillen als er in een bepaald tempo overheen wordt gelopen). Zo'n trilling wordt beschreven door een *eigenvector* van het eigenwaardeprobleem. De bijbehorende *eigenwaarde*, een getal, geeft aan of de trilling gedempt of versterkt wordt. Voor bijvoorbeeld het ontwerpen van bruggen is dit van belang om te weten: een trilling die versterkt wordt kan ervoor zorgen dat de brug instort.

Kenmerkend voor de eigenwaardeproblemen voor dit soort fenomenen is dat ze, mede door een fijnmazig rooster, grootschalig zijn: het aantal onbekenden in een eigenvector kan oplopen van duizend tot ettelijke miljoenen. Gelukkig zijn meestal niet alle eigenwaarden en/of eigenvectoren nodig. Vaak is een aantal tussen één tot enkele tientallen voldoende. Onder deze omstandigheden, een grootschalig eigenwaardeprobleem en een relatief klein

aantal benodigde oplossingen, is de Jacobi-Davidson methode een zeer geschikte manier om met de computer oplossingen te berekenen.

Voor het vinden van een oplossing projecteert de Jacobi-Davidson methode het grote eigenwaardeprobleem op een klein eigenwaardeprobleem. De methode is erop gericht dat steeds meer informatie over de gewenste eigenwaarden/eigenvectoren in het kleine eigenwaardeprobleem bevat is. Met behulp van dit kleine probleem kunnen benaderingen voor deze eigenwaarden/eigenvectoren dan vrij goedkoop berekend worden.

Voor een specifieke component van de Jacobi-Davidson methode kan de benodigde rekentijd echter de pan uit rijzen: de zogenaamde *correctie* vergelijking. Dit is een soort stelsel lineaire vergelijkingen. De correctie vergelijking geeft aan welke belangrijke informatie over de gezochte eigenwaarden/eigenvectoren nog niet in het kleine eigenwaardeprobleem aanwezig is. In hoofdstuk 2 van dit proefschrift zijn een aantal alternatieven voor deze correctievergelijking nader onderzocht.

Omdat het aantal onbekenden in de correctie vergelijking gelijk is aan het aantal onbekenden in een eigenvector van het grote eigenwaardeprobleem, moeten (benaderende) oplossingen voor deze correctie vergelijkingen op een slimme manier berekend worden om de rekentijd laag te houden. Naar één van die manieren heb ik in de rest van mijn proefschrift gekeken, deze manier is gebaseerd op *domeindecompositie*.

Domeindecompositie deelt het gebied waar het model beschreven wordt op in kleinere deelgebiedjes. Er worden nieuwe continue vergelijkingen opgesteld die het gedrag op elk deelgebiedje beschrijven. Op deze manier wordt het grote probleem, beschreven door de continue vergelijkingen op het gehele gebied, opgedeeld in kleinere deelprobleemjes. De som der delen is echter nog niet het geheel: er moet ook nog beschreven worden hoe de gedragingen op de deelgebiedjes op elkaar aansluiten (denk aan de bak met water die is opgedeeld in deelbakjes: als in een deelbakje een golf een bepaalde kant opstroomt dan zal op een gegeven moment een (denkbeeldige) rand van het deelbakje bereikt worden, er moet dan ook aangegeven worden hoe de golf in het aangrenzende deelbakje doorstroomt). Dit aansluiten wordt beschreven door de *interne randvoorwaarden*. Als de exacte interne randvoorwaarden bekend zijn dan beschrijven de continue vergelijkingen op de deelgebiedjes samen met deze interne randvoorwaarden precies het oorspronkelijke, grote probleem. Een oplossing voor het grote probleem kan dan berekend worden door met de computer oplossingen voor alle deelprobleempjes te berekenen en deze op een geschikte wijze aan elkaar te plakken. Zo'n aanpak is praktisch waardevol bij een groot aantal onbekenden: deelproblemen kunnen parallel, onafhankelijk en tegelijkertijd, door verschillende computers opgelost worden.

Bovenstaande is alleen het geval als de exacte interne randvoorwaarden bekend zijn. Dit vereist echter kennis van de oplossing voor het grote probleem. Omdat die oplossing juist berekend moet worden is die kennis niet exact voorhanden. Om hier aan tegemoet te komen wordt een oplossing iteratief berekend. Met behulp van de oplossing van de vorige iteratiestap op het ene deelgebiedje wordt op het aangrenzende deelgebiedje een schatting voor de interne randvoorwaarden bijgesteld en een nieuwe oplossing berekend. Hiervoor moet er elke iteratiestap gecommuniceerd worden tussen de computers die deze deelgebiedjes voor hun rekening nemen. Het iteratieproces kan vergeleken worden met een project (het grote

probleem oplossen) dat door een groep personen wordt uitgevoerd: elke persoon moet zijn eigen deeltaak (een deelprobleem oplossen) uitvoeren, om te zorgen dat de deeltaken op elkaar aansluiten moet er regelmatig tussen de personen overlegd/vergaderd (communicatie tussen computers) worden om deeltaken op elkaar af te stemmen (bijstellen van schatting interne randvoorwaarden). Nadeel van dit iteratieproces is dat er veel tijd kan gaan zitten in de communicatie, zoals ook bij vergaderingen. Om het aantal iteratiestappen en daarmee ook de hoeveelheid communicatie, terug te brengen is wat slims bedacht.

Door een uitbreiding van de onbekenden die pal naast een (denkbeeldige) rand liggen met virtuele tegenhangers (elk deelbakje met water wordt aan elke rand een klein beetje groter gemaakt) wordt er enige vrijheid geintroduceerd. Deze vrijheid vertaalt zich in koppelingsparameters die vrij gekozen kunnen worden. Door een analyse te maken van hoe een fout van een oplossing zich voortplant in het iteratieproces, kan bepaald worden voor welke waarden van de koppelingsparameters het aantal benodigde iteratiestappen afneemt.

Voor de correctievergelijking van Jacobi-Davidson wordt in hoofdstuk 3 van dit proefschrift een dergelijke analyse verricht voor een modelprobleem. Numerieke experimenten laten zien dat dit een scherpe analyse is. Hoofdstuk 4 legt uit hoe, met behulp van de gevonden waarden van de koppelingsparameters voor zo'n modelprobleem, schattingen kunnen worden gemaakt voor meer realistische eigenwaardeproblemen uit de praktijk en het laat zien dat deze schattingen ook effectief zijn. In hoofdstuk 5 wordt stilgestaan bij het feit dat Jacobi-Davidson in combinatie met de domeindecompositie methode voor de correctie vergelijking een geneste iteratieve methode is: een iteratieve methode voor het berekenen van oplossingen voor het eigenwaardeprobleem (de "buitenlus") met daar binnenin een iteratieve methode voor het berekenen van oplossingen voor de correctievergelijking (de "binnenlus"). Door ook gebruik te maken van informatie van de vorige iteratiestap van de buitenlus in volgende binnenlus kan het aantal iteratiestappen nog meer teruggebracht worden.

# Dankwoord

De inhoud van dit proefschrift vormt de weerslag van het onderzoek dat ik de afgelopen vier jaar bij het Mathematisch Instituut in Utrecht en het CWI in Amsterdam heb verricht. Zonder de hulp van anderen was dit boekwerkje niet in deze vorm tot stand gekomen.

Gerard en Henk wil ik bedanken voor de prettige manier waarop ze me begeleid hebben, voor de leerzame en inspirerende gedachtenwisselingen en voor het constructieve en minutieuze commentaar op stukken tekst van mijn hand.

Het was plezierig om in Utrecht een kamer te delen met Jos, Mike en weer Jos. Ook op het CWI in Amsterdam had ik een eigen werkplek, met name de laatste twee jaren heb ik daar in het gezelschap van Harald en Mervyn de nodige uren achter een computer doorgebracht. Met plezier denk ik terug aan de MAS 2.3 werkbesprekingen met Herman, Margreet, eerst Auke en later Jos en de boeiende voordrachten op de MPR (Massaal Parallel Rekenen) bijeenkomsten. De numerieke voordrachten in Utrecht waren ook altijd erg de moeite waard en op de CFD besprekingen heb ik het een en ander op kunnen steken van Navier-Stokes en multi-methoden.

Dank aan medepromovendi, numerici, MAS-sers, overige stafleden en ondersteunend personeel voor de aangename en behulpzame werkomgeving(en).

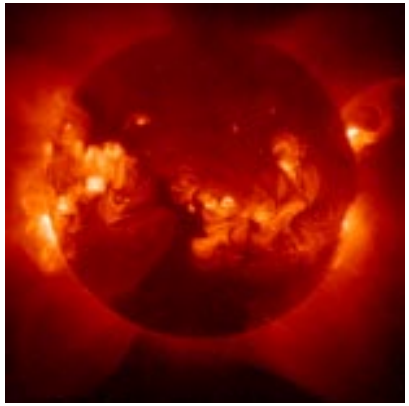
Gedurende mijn promotietijd heb ik de nodige afwisseling gevonden bij de Utrechtse studentenzeilvereniging Histos (onder andere in de RedakCie van het maandblad de Zeillat), de Belangenvereniging voor AIO's & OIO's in Utrecht (BAU) en het Landelijk AIO & OIO Overleg (LAIOO). In het zeilseizoen was ik ook regelmatig na het werk op het water te vinden: samen met Jeroen nam ik dan de bus naar Loosdrecht om daar nog een paar uurtjes te zeilen. Om naast het zeilen ook 's winters in beweging te blijven heb ik met plezier meegetraind met de Utrechtse studentenkorfbalvereniging Hebbes en geschaatst met collega's van het CWI op de Jaap Edenbaan.

Tenslotte wil ik mijn ouders en zus bedanken voor de plezierige jeugd en dat ik het zo ver heb kunnen schoppen.

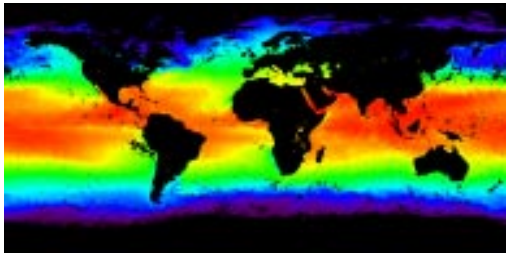
# Curriculum Vitae

Ik ben op 8 september 1972 geboren in Amsterdam. In juni 1990 behaalde ik het Atheneumdiploma aan het Pieter Nieuwland College te Amsterdam. Vervolgens ben ik aan de Universiteit van Amsterdam begonnen met studeren. Na de propedeuse Wiskunde in augustus 1991 volgde een jaar later de propedeuse Natuurkunde. Begin 1997 studeerde ik af in de Numerieke Wiskunde bij dr. W. Hoffmann, met de scriptie “Parallel numerical algorithms based on subspace expansions”.

Twee dagen later startte ik als onderzoeker in opleiding (OIO) binnen een samenwerkingsverband tussen het Mathematisch Instituut van de Universiteit Utrecht en de cluster Modelling, Analysis and Simulation (MAS) van het Centrum voor Wiskunde & Informatica (CWI) in Amsterdam. Het promotieonderzoek heeft geresulteerd in dit proefschrift. De onderwijstaak bij het Mathematisch Instituut, die bestond uit het verzorgen van diverse werkcolleges, vormde voor mij een verfrissend tegenwicht.

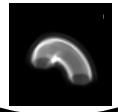


astrophysical processes



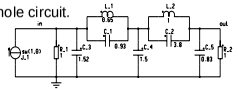

electric circuit design

One of the proposed mechanisms to explain the heating of the solar corona is resonant absorption of magnetic Alfven waves. Analysis of these magnetohydrodynamic waves in a coronal loop needs the solution of very large generalized eigensystems.



climate modelling

The behavior of each component (like coil, capacitor, ...) is described by one or more equations. Together with Kirchhoff's law this gives a coupled system of equations for the behavior of the whole circuit.



Stability of the circuit is related to the eigenfrequencies of this coupled system.

eigenvalue problem

$Ax=\lambda Bx$

in millions of unknowns