DELFT UNIVERSITY OF TECHNOLOGY

REPORT 04-09

A comparison of Deflation and the balancing Neumann-Neumann preconditioner

R. Nabben, C. Vuik

ISSN 1389-6520

Reports of the Department of Applied Mathematical Analysis

Delft 2004

A comparison of Deflation and the balancing Neumann-Neumann preconditioner

R. Nabben* C. Vuik†

December 22, 2004

Abstract

In this paper we compare various preconditioners for the numerical solution of partial differential equations. We compare the well-known balancing Neumann-Neumann preconditioner used in domain decomposition methods with a so-called deflation preconditioner. We prove that the effective condition number of the deflated preconditioned system is always, i.e. for all deflation vectors and all restrictions and prolongations, below the condition number of the system preconditioned by the balancing Neumann-Neumann preconditioner. Even more, we establish that both preconditioners lead to almost the same spectra. The zero eigenvalues of the deflation preconditioned system are replaced by eigenvalues which are one if the balancing Neumann-Neumann preconditioner is used. Moreover, we proved that the A-norm of the errors of the iterates build by the deflation preconditioner is always below the A-norm of the errors of the iterates build by the balancing Neumann-Neumann preconditioner. Additionally, the amount of work of one iteration of the deflation preconditioned system is less than the amount of work of one iteration of the balancing Neumann-Neumann preconditioned system. Finally, we establish that the deflation preconditioner and the balancing Neumann-Neumann preconditioner produces the same iterates if one uses certain starting vectors. Numerical results for porous media flows emphasize the theoretical results.

Keywords. deflation, coarse grid correction, balancing Neumann Neumann, preconditioners, Conjugate Gradients, porous media flow, scalable parallel preconditioner

AMS subject classifications. 65F10, 65F50, 65N22

1 Introduction

The Conjugate Gradient method is the most used method to solve large linear systems of equations

$$Ax = b$$

whose coefficient matrices A are sparse and symmetric positive definite. Such systems are encountered, for example, when a finite volume/difference/element method is used to discretize an elliptic partial differential equation.

The convergence rate of the Conjugate Gradient method (cg-method) is bounded as a function of the condition number of the system matrix to which it is applied. If the condition number of A is large it is advisable to solve, instead, a preconditioned system $M^{-1}Ax = M^{-1}b$, where the symmetric positive definite preconditioner M is chosen such that $M^{-1}A$ has a more clustered spectrum or a smaller condition number than that of A. Furthermore, M must be cheap to

*TU Berlin Institut für Mathematik, MA 3-3 Strasse des 17. Juni 136 D-10623 Berlin Germany, phone 49 30-314-29291 (nabben@math.tu-berlin.de)

†Delft University of Technology, Faculty of Electrical Engineering, Mathematics and Computer Science, Delft Institute of Applied Mathematic, Mekelweg 4, 2628 CD Delft, The Netherlands, phone: 31 15 2785530, fax: 31 15 2787209 (c.vuik@math.tudelft.nl)

solve relative to the improvement it provides in convergence rate. A final desirable property in a preconditioner is that it should parallelize well, especially on distributed memory computers.

In [16] two different preconditioner are compared, namely a deflation preconditioner and a coarse grid correction preconditioner. It is shown that the deflation preconditioner leads to a smaller condition number than a coarse grid correction preconditioner like the BPS preconditioner by Bramble, Paschiak and Schatz [1].

Here we compare the deflation preconditioner with the balancing Neumann-Neumann preconditioner by Mandel [11].

In the following we give a brief introduction into both preconditioning techniques.

To describe the deflation method we define the projection P_D by

$$P_D = I - AZ(Z^T AZ)^{-1} Z^T, \quad Z \in \mathbb{R}^{n \times r}, \tag{1}$$

where the column space of Z is the deflation subspace, i.e. the space to be projected out of the residual, and I is the identity matrix of appropriate size. We assume that $r \ll n$ and that Z has rank r. Under this assumption $E \equiv Z^T AZ$ may be easily computed and factored and is symmetric positive definite. Since $x = (I - P_D^T)x + P_D^T x$ and because

$$(I - P_D^T)x = Z(Z^T AZ)^{-1} Z^T Ax = ZE^{-1} Z^T b \tag{2}$$

can be immediately computed, we need only compute $P_D^T x$. In light of the identity $AP_D^T = P_D A$, we can solve the deflated system

$$P_D A\tilde{x} = P_D b \tag{3}$$

for $\tilde{x}$ using the Conjugate Gradient method, premultiply this by P_D^T and add it to (2).

Obviously (3) is singular. But a positive semidefinite system can be solved by the cg-method as long as the right-hand side is consistent (i.e. as long as $b = Ax$ for some x) [9]. This is certainly true for (3), where the same projection is applied to both sides of the nonsingular system. Since the null space never enters the iteration, the corresponding zero-eigenvalues do not influence the convergence [9, 21]. Motivated by this fact, we define the *effective condition number* of a positive semidefinite matrix $C \in \mathbb{R}^{n \times n}$ with r zero eigenvalues to be the ratio of its largest to smallest *positive* eigenvalues:

$$\kappa_{\text{eff}}(C) = \frac{\lambda_n}{\lambda_{r+1}}.$$

It is possible to combine both a standard preconditioning and preconditioning by deflation (for details see [7]). The convergence is then described by the effective condition number of $M^{-1} P_D A$. For more details about the deflation preconditioner see [17, 15, 4, 13, 14, 10, 23, 2, 24, 25, 7, 22, 16].

We compare the preconditioned deflation operator with the balancing Neumann-Neumann preconditioner proposed by Mandel [11, 12, 3, 18]. As the FETI algorithm [5, 6] the balancing Neumann-Neumann preconditioner is one of the domain decomposition methods that have been most carefully implemented and severely tested on the very largest existing parallel computer systems.

Applied to some symmetric positive definite problems the balancing Neumann-Neumann preconditioner leads to condition numbers which grow like $0(1 + log(\frac{H}{h}^2))$ in both two and three dimensions [20]. Moreover, the condition numbers are independent of jumps in the coefficients in the matrices [20].

In our notation the balancing Neumann-Neumann preconditioner is given by

$$P_B = (I - ZE^{-1} Z^T A) M^{-1} (I - AZE^{-1} Z^T) + ZE^{-1} Z^T, \tag{4}$$

here $Z \in \mathbb{R}^{n \times r}$, $E = Z^T AZ$ and M is a symmetric positive definite matrix - the Neumann-Neumann preconditioner. Note that P_B is symmetric and positive definite. For more details we refer to the the books [20] and [19]. Further references are [11].

As a first comparison of both preconditioners we easily observe that balancing Neumann-Neumann preconditioner needs per iteration 3 matrix vector products and the coarse grid operator

is used 2 times. This makes the balancing Neumann-Neumann preconditioner per iteration more expensive than the deflation approach.

In this article we give a detailed comparison of these two preconditioners. We prove that the effective condition number of the deflated preconditioned system $M^{-1}P_DA$ is always below the condition number of the system preconditioned by the balancing Neumann-Neumann preconditioner P_BA. Even more, we establish that the spectra of P_BA is the same as $M^{-1}P_DA$, except the r zero eigenvalues are replaced by eigenvalues which are one.

This implies that for all matrices $Z \in \mathbb{R}^{n\times r}$ and all positive definite preconditioners M^{-1} the effective condition number of the deflated preconditioned system is below or equal to the condition number of the system preconditioned by the balancing Neumann-Neumann preconditioner! However the condition number is not the only parameter which influence the convergence behavior of the Conjugate Gradient method. The convergence may be significantly faster if the eigenvalues of A are clustered [21]. But we obtain from the above mentioned result that the clustering of the eigenvalues of the two different preconditioned systems is the same. However, we have a cluster at zero in one case and at one in the other case. These results are stated in Section 2.

However, there are other properties which influence the convergence behavior of the Conjugate Gradient method, e.g. the starting vector, the right hand side and the location of the clusters of eigenvalues. Therefore, a more detailed comparison is given in Section 3. There we proof that the A-norm of the errors of the iterates build by the deflation preconditioner is always below the A-norm of the errors of the iterates build by the balancing Neumann-Neumann preconditioner. Moreover, we establish that the deflation preconditioner and the balancing Neumann-Neumann preconditioner produces the same iterates if one uses certain starting vectors. More precisely we show which terms in the preconditioned Conjugate Gradient method are the same for both methods and which terms are different. At the end of Section 3 we prove that the condition of the balancing Neumann-Neumann preconditioned system decrease if one take a finer grid as a coarse grid.

In Section 4 numerical results emphasize our theoretical results.

2 Spectral properties

In this section we compare the effective condition number for the deflation and balancing Neumann-Neumann preconditioned matrices. In Section 2.1 we give some definitions and preliminary results. Thereafter a comparison is made if the projection vectors are equal to eigenvectors in Section 2.2 and for general projection vectors in Section 2.3.

2.1 Notations and Preliminary Results

In the following we denote by $\lambda_i(M)$ the eigenvalues of a matrix M. If the eigenvalues are real the $\lambda_i(M)'s$ are ordered increasingly.

For two Hermitian $n \times n$ matrices A and B we write $A \succeq B$, if $A - B$ is positive semidefinite. Next we mention well-known properties of the eigenvalues of Hermitian matrices.

Lemma 2.1 *Let $A, B \in \mathbb{C}^{n\times n}$ be Hermitian. For each $k = 1, 2, \ldots, n$ we have*

$$\lambda_k(A) + \lambda_1(B) \leq \lambda_k(A + B) \leq \lambda_k(A) + \lambda_n(B)$$

From the above lemma we easily obtain the next lemma.

Lemma 2.2 *If $A, B \in \mathbb{C}^{n\times n}$ are positive semidefinite with $A \succeq B$, then $\lambda_i(A) \geq \lambda_i(B)$.*

Moreover, we will use

Lemma 2.3 *Let $A, B \in \mathbb{C}^{n\times n}$ be Hermitian and suppose that B has rank at most r. Then*

- $\lambda_k(A+B) \leq \lambda_{k+r}(A), \quad k = 1, 2, \cdots n-r,$
- $\lambda_k(A) \leq \lambda_{k+r}(A+B), \quad k = 1, 2, \cdots n-r.$

Lemma 2.1, Lemma 2.2 and Lemma 2.3 can be found e.g. as Theorem 4.3.1, Corollary 7.7.4. and Theorem 4.3.6, respectively, in [8].

2.2 Projection vectors chosen as eigenvectors

In this subsection we compare the effective condition number of $P_D A$ and $P_B A$ if the projection vectors are equal to the eigenvectors of A.

Definition 2.4 *Let λ_i be the eigenvalues of A. Choose the eigenvectors v_k of A such that $v_k^T v_j = \delta_{kj}$, and define $Z = [v_1 \dots v_r]$.*

Theorem 2.5 *Using Z as given in Definition 2.4 and preconditioner M equal to the identity, the spectrum of $P_B A$ is:*

$$spectrum(P_B A) = \{1, \dots, 1, \lambda_{r+1}, \dots, \lambda_n\}.$$

Proof: For this choice of Z it appears that

$$E = Z^T A Z = diag(\lambda_1, \dots, \lambda_r). \tag{5}$$

We consider $P_B A v_k$. For $k = 1, \dots, n$ we obtain

$$P_B A v_k = (I - Z diag(\frac{1}{\lambda_1}, \dots, \frac{1}{\lambda_r})\ Z^T A)(I - AZ diag(\frac{1}{\lambda_1}, \dots, \frac{1}{\lambda_r})\ Z^T)\lambda_k v_k +$$

$$Z diag(\frac{1}{\lambda_1}, \dots, \frac{1}{\lambda_r})\ Z^T \lambda_k v_k.$$

Using the orthogonality properties of the eigenvectors one obtains:

$$P_B A v_k = v_k, \text{ for } k = 1, \dots, r.$$

For $k = r+1, \dots, n$ the same orthogonality properties lead to:

$$P_B A v_k = \lambda_k v_k, \text{ for } k = r+1, \dots, n.$$

□

In order to compare both approaches we note that

$$\kappa_{eff}(P_D A) = \frac{\lambda_n}{\lambda_{r+1}}, \tag{6}$$

and

$$\kappa(P_B A) = \frac{\max\{1, \lambda_n\}}{\min\{1, \lambda_{r+1}\}}. \tag{7}$$

From (6) and (7) it follows that $\kappa(P_B A) \geq \kappa_{eff}(P_D A)$, so the convergence bound based on the effective condition number implies that Deflated CG converges faster than CG combined with the balancing Neumann-Neumann preconditioner if both methods use the eigenvectors corresponding to the r smallest eigenvalues as projection vectors.

2.3 Projection vectors chosen as general vectors

In the previous section we showed that the deflation technique leads to a smaller effective condition number than the balancing Neumann-Neumann preconditioner, if eigenvectors are used. However, computing the r smallest eigenvalues is mostly very expensive. Moreover, in multigrid methods and domain decomposition methods special interpolation and prolongation matrices are used to obtain grid independent convergence rates. So a comparison only for eigenvectors is not enough. But in this section we generalize the results of the last section. We prove that the effective condition number of the deflated preconditioned system is always, i.e. for all matrices $Z \in \mathbb{R}^{n\times r}$, and all preconditioners M^{-1}, below the condition number of the system preconditioned by the balancing Neumann-Neumann preconditioner. To do this we repeat some properties of the projection operator P_D used in the Deflation method (see [7]). The operator P_D is defined as:

$$P_D = I - AZE^{-1}Z^T, \text{ where } E = Z^TAZ. \tag{8}$$

Furthermore, the following identities hold:

$$P_D^2 = P_D, \;\; P_DAZ = 0, \;\; Z^TP_D = P_D^TZ = 0, \quad \text{and } AP_D^T = P_DA.$$

We start with a result for the deflation preconditioner which helps to compare the deflation and the balancing Neumann-Neumann preconditioner.

Theorem 2.6 *Let $A \in \mathbb{R}^{n\times n}$ be symmetric positive definite. Let $Z \in \mathbb{R}^{n\times r}$ with $rankZ = r$. Then*

$$\sigma(P_D^TM^{-1}P_DA) = \sigma(M^{-1}P_DA).$$

Proof:
Let λ be a nonzero eigenvalue of $M^{-1}P_DA$, i.e. there exists a nonzero vector x such that

$$M^{-1}P_DAx = \lambda x.$$

But then

$$M^{-1}P_DAx = M^{-1}P_D^2Ax = M^{-1}P_DAP_D^Tx = \lambda x \neq 0.$$

Thus P_D^Tx is a nonzero vector. Moreover,

$$P_D^TM^{-1}P_DAP_D^Tx = \lambda P_D^Tx.$$

Thus

$$\sigma(M^{-1}P_DA) \subseteq \sigma(P_D^TM^{-1}P_DA).$$

Now let λ be a nonzero eigenvalue of $P_D^TM^{-1}P_DA$, i.e. there exists a nonzero vector y such that

$$y^TP_D^TM^{-1}P_DA = \lambda y^T.$$

Thus $y^TP_D^T$ is nonzero and we obtain

$$y^TP_D^TM^{-1}P_DAP_D^T = \lambda y^TP_D^T.$$

But

$$y^TP_D^TM^{-1}P_DAP_D^T = y^TP_D^TM^{-1}P_D^2A = y^TP_D^TM^{-1}P_DA.$$

Hence

$$y^TP_D^TM^{-1}P_DA = \lambda y^TP_D^T.$$

Thus

$$\sigma(P_D^TM^{-1}P_DA) \subseteq \sigma(M^{-1}P_DA),$$

which completes the proof. □

Using Theorem 2.6 we obtain

Theorem 2.7 *Let $A \in \mathbb{R}^{n\times n}$ be symmetric positive definite. Let $Z \in \mathbb{R}^{n\times r}$ with $rankZ = r$. Then the preconditioner defined in* (1) *and* (4) *satisfies*

$$\lambda_n(M^{-1}P_DA) \leq \lambda_n(P_BA) \tag{9}$$
$$\lambda_{r+1}(M^{-1}P_DA) \geq \lambda_1(P_BA). \tag{10}$$

Proof: We can write P_B as

$$P_B = P_D^TM^{-1}P_D + ZE^{-1}Z^T.$$

Thus

$$A^{\frac{1}{2}}P_BA^{\frac{1}{2}} = A^{\frac{1}{2}}P_D^TM^{-1}P_DA^{\frac{1}{2}} + A^{\frac{1}{2}}ZE^{-1}Z^TA^{\frac{1}{2}}.$$

Since $A^{\frac{1}{2}}ZE^{-1}Z^TA^{\frac{1}{2}}$ is a symmetric positive semidefinite matrix of rank r, we obtain with Lemma 2.2

$$\lambda_i(P_BA) = \lambda_i(A^{\frac{1}{2}}P_BA^{\frac{1}{2}}) \geq \lambda_i(A^{\frac{1}{2}}P_D^TM^{-1}P_DA^{\frac{1}{2}}) = \lambda_i(P_D^TM^{-1}P_DA)$$

Using Lemma 2.3 we get

$$\lambda_{r+1}(P_D^TM^{-1}P_DA) \geq \lambda_1(P_BA).$$

Using Theorem 2.6 we get the desired result.

□

It follows from Theorem 2.7 that

$$\kappa(P_BA) \geq \kappa_{eff}(M^{-1}P_DA)$$

so the convergence bound based on the effective condition number implies that preconditioned Deflated CG converges faster than CG preconditioned by the balancing Neumann-Neumann preconditioner.

It appears that the results given in Theorem 2.5 can be generalized to general projection vectors.

Theorem 2.8 *Suppose that the spectrum of $M^{-1}P_DA$ is given by:*

$$spectrum(M^{-1}P_DA) = \{0, \ldots, 0, \mu_{r+1}, \ldots, \mu_n\},$$

then

$$spectrum(P_BA) = \{1, \ldots, 1, \mu_{r+1}, \ldots, \mu_n\}.$$

Proof: We know that $M^{-1}P_DAZ = 0$, so the eigenvectors corresponding to the zero eigenvalues of $M^{-1}P_DA$ are equal to $\{z_1, \ldots, z_r\}$. On the other hand it is easy to check that

$$P_BAZ = (P_D^TM^{-1}P_D + ZE^{-1}Z^T)AZ = P_D^TM^{-1}P_DAZ + ZE^{-1}Z^TAZ = Z.$$

This implies that $\{z_1, \ldots, z_r\}$ are the eigenvectors corresponding to the eigenvalues of P_BA, which are equal to 1.

Now we consider the eigenvalue μ_i, with $r+1 \leq i \leq n$. Suppose v_i is the corresponding eigenvector of $M^{-1}P_DA$, so $M^{-1}P_DAv_i = \mu_iv_i$. Since

$$M^{-1}P_DAv_i = M^{-1}P_D^2Av_i = M^{-1}P_DAP_D^Tv_i = \mu_iv_i \neq 0,$$

the vector $P_D^Tv_i$ is nonzero. Using this vector, it follows that:

$$P_BA(P_D^Tv_i) = (P_D^TM^{-1}P_D + ZE^{-1}Z^T)AP_D^Tv_i = \tag{11}$$
$$= P_D^TM^{-1}P_DAP_D^Tv_i + ZE^{-1}Z^TAP_D^Tv_i = \tag{12}$$
$$= P_D^TM^{-1}P_D^2Av_i = P_D^TM^{-1}P_DAv_i = \mu_iP_D^Tv_i. \tag{13}$$

So the vectors $P_D^Tv_i$ are eigenvectors of P_BA corresponding to the eigenvalues μ_i. □

Thus both preconditioners lead to almost the same spectra with the same clustering. The zero eigenvalues of the deflation preconditioned system are replaced by eigenvalues which are one if the balancing Neumann-Neumann preconditioner is used.

3 Comparing the norm of the residuals

In order to make a more detailed comparison of the Deflation operator and the balancing Neumann-Neumann preconditioner for general projection vectors we start to compare the vectorspaces which contain the approximations of both methods. Using CG with P_B as preconditioner and start vector $x_{0,B} = 0$ it is well known that

$$x_{k,B} \in K^k\{P_BA, P_Bb\},$$

where the Krylov subspace $K^k\{P_BA, P_Bb\} = span\{P_Bb, P_BAP_Bb, \ldots, (P_BA)^{k-1}P_Bb\}$.

Theorem 3.1 *The Krylov space used in the CG method with P_B as preconditioner and startvector $x_{0,B} = 0$ has the following property:*

$$K^k\{P_BA, P_Bb\} \subset span\{ZE^{-1}Z^Tb, P_D^TM^{-1}P_Db, \ldots, P_D^T(M^{-1}P_DA)^{k-1}M^{-1}P_Db\}. \tag{14}$$

Proof: To start the proof we first note that

$$P_Bb = P_D^TM^{-1}P_Db + ZE^{-1}Z^Tb.$$

So the property holds for $k = 1$. For $k = 2$ we note that

$$P_BAP_Bb = (P_D^TM^{-1}P_D + ZE^{-1}Z^T)A(P_D^TM^{-1}P_D + ZE^{-1}Z^T)b.$$

Writing out the various terms on the right-hand side one obtains:

$$ZE^{-1}Z^TAZE^{-1}Z^Tb = ZE^{-1}Z^Tb,$$

$$P_D^TM^{-1}P_DAP_D^TM^{-1}P_Db = P_D^TM^{-1}P_DP_DAM^{-1}P_Db = P_D^TM^{-1}P_DAM^{-1}P_Db,$$

where we have used that $AP_D^T = P_DA$ and $P_D^2 = P_D$. Finally the terms

$$P_D^TM^{-1}P_DAZE^{-1}Z^T \text{ and } ZE^{-1}Z^TAP_D^TM^{-1}P_D,$$

are both zero because they contain the combination $P_DAZ = 0$ or $Z^TAP_D^T = (P_DAZ)^T = 0$. Repeating this argument for $(P_BA)^iP_Bb$ for $i = 3, \ldots k-1$ proves the theorem.

□

With respect to the approximation using preconditioned CG combined with Deflation, we note that $x = (I - P_D^T)x + P_D^Tx = ZE^{-1}Z^Tb + P_D^Tx$. So after k iterations of preconditioned CG applied to $AP_D^Tx = P_DAx = P_Db$ we get the approximation $\tilde{x}_{k,D}$. The approximation $x_{k,D}$ of the solution vector x is then given by $x_{k,D} = ZE^{-1}Z^Tb + P_D^T\tilde{x}_{k,D}$. The vector $x_{k,D}$ is contained in the following space:

$$x_{k,D} \in ZE^{-1}Z^Tb + span\{P_D^TM^{-1}P_Db, \ldots, P_D^T(M^{-1}P_DA)^{k-1}M^{-1}P_Db\}.$$

This implies that both approximations are element of the same space. So the difference in quality of the approximation only depends on which norm is minimized.

Lemma 3.2 *For the Deflation iterates $x_{k,D}$ and $\tilde{x}_{k,D}$ with start vector $\tilde{x}_{0,D} = 0$ the following optimality property holds*

$$\|x - x_{k,D}\|_A = \|P_D^T(x - \tilde{x}_{k,D})\|_A = \min_{\xi \in K^k\{M^{-1}P_DA, M^{-1}P_Db\}} \|P_D^T(x - \xi)\|_A \tag{15}$$

Proof: The first equality follows from the fact that $x = (I - P_D^T)x + P_D^Tx$. If CG is applied to the preconditioned system

$$L^{-1}P_DAL^{-T}y = L^{-1}P_Db, \tag{16}$$

the following expression holds

$$\|\tilde{y} - y_k\|_{L^{-1}P_DAL^{-T}} = \min_{\eta \in K^k\{L^{-1}P_DAL^{-T}, L^{-1}P_Db\}} \|\tilde{y} - \eta\|_{L^{-1}P_DAL^{-T}}, \tag{17}$$

where $\tilde{y}$ is a solution of (16). Note that $\tilde{x} = L^{-T}\tilde{y}$ is a solution of $P_DAx = P_Db$. Rewriting (17) with $\xi = L^{-T}\eta$ leads to

$$\|L^T(\tilde{x} - \tilde{x}_{k,D})\|_{L^{-1}P_DAL^{-T}} = \min_{\xi \in K^k\{M^{-1}P_DA, M^{-1}P_Db\}} \|L^T(\tilde{x} - \xi)\|_{L^{-1}P_DAL^{-T}}.$$

Using the equalities:

$$\|L^T(\tilde{x} - \tilde{x}_{k,D})\|^2_{L^{-1}P_DAL^{-T}} = (\tilde{x} - \tilde{x}_{k,D})^T P_DA(\tilde{x} - \tilde{x}_{k,D}) =$$

$$= (\tilde{x} - \tilde{x}_{k,D})^T P_D^2 A(\tilde{x} - \tilde{x}_{k,D}) = \|P_D^T(\tilde{x} - \tilde{x}_{k,D})\|^2_A = \|P_D^T(x - \tilde{x}_{k,D})\|^2_A$$

leads to the proof of the lemma.

□

Theorem 3.3 *Let $x_{k,D}$ and $\tilde{x}_{k,D}$ be the Deflation iterates with start vector $\tilde{x}_{0,D} = 0$. For every $x_k \in span\{ZE^{-1}Z^Tb, P_D^TM^{-1}P_Db, \ldots, P_D^T(M^{-1}P_DA)^{k-1}M^{-1}P_Db\}$ the following inequality holds:*

$$\|x - x_{k,D}\|_A \leq \|x - x_k\|_A.$$

Proof:

We decompose x_k as follows $x_k = \alpha ZE^{-1}Z^Tb + P_D^T\xi$, where $\xi \in K^k\{M^{-1}P_DA, M^{-1}P_Db\}$. Substituting this into $\|x - x_k\|^2_A$ shows that

$$\|x - x_k\|^2_A = \|x - \alpha ZE^{-1}Z^Tb - P_D^T\xi\|^2_A.$$

Using the equation $x = (I - P_D^T)x + P_D^Tx = ZE^{-1}Z^Tb + P_D^Tx$ we obtain

$$\begin{aligned} \|x - x_k\|^2_A &= \|(1-\alpha)ZE^{-1}Z^Tb - P_D^T(x - \xi)\|^2_A \\ &= (1-\alpha)^2\|ZE^{-1}Z^Tb\|^2_A + \|P_D^T(x - \xi)\|^2_A + \\ &\quad (1-\alpha)b^TZE^{-1}Z^TAP_D^T(x - \xi) + \\ &\quad (1-\alpha)(x - \xi)^TP_DAZE^{-1}Z^Tb. \end{aligned}$$

The last two terms are equal to zero, because $Z^TAP_D^T = (P_DAZ)^T = 0$. For $x_{k,D}$ we know that $\alpha = 1$. This together with Lemma 3.2 implies

$$\|x - x_{k,D}\|^2_A \leq (1-\alpha)^2\|ZE^{-1}Z^Tb\|^2_A + \|P_D^T(x - \xi)\|^2_A = \|x - x_k\|^2_A,$$

where $\xi \in K^k\{M^{-1}P_DA, M^{-1}P_Db\}$.

□

Theorem 3.1 and Theorem 3.3 imply

Theorem 3.4 *The iterates $x_{k,D}$ and $x_{k,B}$ of the CG method with start vector zero and preconditioned by the deflation preconditioner and the balancing Neumann-Neumann preconditioner respectively satisfy*

$$\|x - x_{k,D}\|_A \leq \|x - x_{k,B}\|_A.$$

Next we are able to prove that using a certain start vector the iterates $x_{k,D}$ are equal to the $x_{k,B}$.

Theorem 3.5 *Using $x_{0,B} = ZE^{-1}Z^Tb$ and $\tilde{x}_{0,D} = 0$ it follows that $x_{k,D} = x_{k,B}$.*

Proof: Using the start vector $x_{0,B} = ZE^{-1}Z^T b$ it appears that

$$r_{0,B} = b - Ax_{0,B} = (I - AZE^{-1}Z^T)b = P_D b.$$

This implies that the Krylov subspace is given by $K^k\{P_B A, P_B P_D b\}$. For $k = 1$ it follows from $P_D^2 = P_D$ and $Z^T P_D = Z^T(I - AZE^{-1}Z^T) = 0$ that

$$(P_D^T M^{-1} P_D + ZE^{-1}Z^T)P_D b = P_D^T M^{-1} P_D b.$$

For $k = 2$ we know from the proof of Theorem 3.1 that

$$P_B A P_B P_D b = ZE^{-1}Z^T P_D b + P_D^T M^{-1} P_D A M^{-1} P_D^2 b.$$

Note that $Z^T P_D = Z^T(I - AZE^{-1}Z^T) = 0$ so

$$P_B A P_B P_D b = P_D^T M^{-1} P_D A M^{-1} P_D b.$$

Repeating this argument shows that

$$K^k\{P_B A, P_B P_D b\} = span\{P_D^T M^{-1} P_D b, \ldots, P_D^T (M^{-1} P_D A)^{k-1} M^{-1} P_D b\} =$$
$$= P_D^T K^k\{M^{-1} P_D A, M^{-1} P_D b\}.$$

We again use the fact that CG combined with the balancing Neumann-Neumann preconditioner minimizes

$$(x - x_{k,B})^T A (x - x_{k,B}),$$

where

$$x_{k,B} = ZE^{-1}Z^T b + P_D^T \xi, \text{ and } \xi \in K^k\{M^{-1} P_D A, M^{-1} P_D b\}$$

due to the choice of the start vector $x_{0,B} = ZE^{-1}Z^T b$. We have that

$$x - x_{k,B} = x - ZE^{-1}Z^T b - P_D^T \xi = P_D^T (x - \xi).$$

But by Lemma 3.2 the optimal ξ is nothing else than $\tilde{x}_{k,D}$. So we obtain

$$x - x_{k,B} = P_D^T (x - \tilde{x}_{k,D}).$$

Since $x = ZE^{-1}Z^T b + P_D^T x$ we get

$$x_{k,D} = ZE^{-1}Z^T b + P_D^T \tilde{x}_{k,D} = x_{k,B}.$$

□

Using the identity $x_{k,D} = x_{k,B}$ it is easy to see that Theorem 2.11 of [16] implies that the balancing Neumann Neumann preconditioner with $x_{0,B} = ZE^{-1}Z^T b$ converges faster than the additive coarse grid preconditioner.

In the following we give a more detailed analysis of the preconditioned cg method for both preconditioners if the above start vectors are used. We prove which quantities in the preconditioned CG-algorithm (PCG) are same for both preconditioners and which are different. To make this paper self-containing we repeat the PCG-algorithm.

PCG-algorithm for $Ax = b$ with preconditioner M^{-1}.
$r_0 := b - Ax_0, \quad z_0 = M^{-1} r_0, \quad p_0 := z_0$
For $j = 0, 1, \ldots$ until convergence, do
  $\alpha_j := (r_j, z_j)/(Ap_j, p_j)$
  $x_{j+1} := x_j + \alpha_j p_j$
  $r_{j+1} := r_j - \alpha_j A p_j$
  $z_{j+1} := M^{-1} r_{j+1}$
  $\beta_j := (r_{j+1}, z_{j+1})/(r_j, z_j)$
  $p_{j+1} := z_{j+1} + \beta_j p_j$
end

Moreover, we need the next proposition

Proposition 3.6 *Let P_D, P_B and M^{-1} be defined as above. Then*

$$P_D^T P_B P_D = P_D^T M^{-1} P_D = P_D^T P_B = P_B P_D. \tag{18}$$

Proof: Since $P_D = I - AZE^{-1}Z^T$ we have $P_D^T Z = Z - ZE^{-1}Z^T AZ = 0$. Hence,

$$P_D^T P_B P_D = P_D^{T^2} M^{-1} P_D^2 + P_D^T Z E^{-1} Z P_D = P_D^T M^{-1} P_D.$$

Similarly,

$$P_D^T P_B = P_D^{T^2} M^{-1} P_D + P_D Z E^{-1} Z P_D = P_D^T M^{-1} P_D.$$

Since $P_D^T M^{-1} P_D$ is symmetric we have also $P_D^T M^{-1} P_D = P_B P_D$. □

Now, we can prove the following Theorem:

Theorem 3.7 *Using the PCG algorithm with the balancing Neumann-Neumann preconditioner P_B and $x_{0,B} = ZE^{-1}Z^T b$ on one side and with the deflation preconditioner $M^{-1}P_D$ and $\tilde{x}_{0,D} = 0$ on the other side we have for all j*

$$\begin{aligned}
(r_{j,D}, z_{j,D}) &= (r_{j,B}, z_{j,B}),\\
(P_D A p_{j,D}, p_{j,D}) &= (A p_{j,B}, p_{j,B})\\
r_{j+1,D} &= r_{j+1,B}\\
z_{j+1,B} &= P_D^T z_{j+1,D}\\
p_{j+1,B} &= P_D^T p_{j+1,D}\\
\beta_{j,B} &= \beta_{j,D}\\
x_{j+1,B} &= x_{j+1,D} = ZE^{-1}Z^T + P_D^T \tilde{x}_{j+1,D}.
\end{aligned}$$

Proof: If we use PCG for

$$P_D A x = P_D b$$

with preconditioner M^{-1} and start vector $x_0 = 0$ we obtain

$$\begin{aligned}
x_{0,D} &= 0, \quad r_{0,D} = P_D b, \quad z_{0,D} = M^{-1} P_D b,\\
p_{0,D} &= z_{0,D} = M^{-1} P_D b, \quad \alpha_{0,D} = \frac{(r_{0,D}, z_{0,D})}{(P_D A p_{0,D}, p_{0,D})},\\
\tilde{x}_{1,D} &= 0 + \alpha_{0,D} M^{-1} P_D b,\\
x_{1,D} &= ZE^{-1}Z^T b + \alpha_{0,D} P_D^T M^{-1} P_D b.
\end{aligned}$$

If we use PCG for

$$Ax = b$$

with preconditioner P_B and start vector $x_0 = ZE^{-1}Z^T b$ we obtain

$$\begin{aligned}
x_{0,B} = ZE^{-1}Z^T b, \quad r_{0,B} = P_D b, \quad z_{0,B} = P_B P_D b,\\
p_{0,B} = z_{0,B} = P_B P_D b, \quad \alpha_{0,B} = \frac{(r_{0,B}, z_{0,B})}{(A p_{0,B}, p_{0,B})},\\
x_{1,B} = ZE^{-1}Z^T b + \alpha_{0,B} P_B P_D b.
\end{aligned}$$

Obviously, we have for all iterates

$$P_D r_{j+1,D} = P_D(P_D b - P_D A x_{j+1,D}) = r_{j+1,D}. \tag{19}$$

The identity

$$P_B P_D = P_D^T M^{-1} P_D^2 + ZE^{-1}Z^T P_D = P_D^T M^{-1} P_D$$

is frequently used in the following analysis.

Next, we prove the following identities by induction:

$$\begin{aligned}
(r_{j,D}, z_{j,D}) &= (r_{j,B}, z_{j,B}), \quad (r_{j+1,D}, z_{j+1,D}) = (r_{j+1,B}, z_{j+1,B}) \\
(P_D A p_{j,D}, p_{j,D}) &= (A p_{j,B}, p_{j,B}) \\
r_{j+1,D} &= r_{j+1,B} \\
z_{j+1,B} &= P_D^T z_{j+1,D} \\
p_{j+1,B} &= P_D^T p_{j+1,D} \\
\beta_{j,B} &= \beta_{j,D} \\
x_{j+1,B} &= x_{j+1,D} = Z E^{-1} Z^T + P_D^T \tilde{x}_{j+1,D}.
\end{aligned}$$

In the following we use the Proposition 3.6 and (19). For $j = 0$ we have

$$(r_{0,D}, z_{0,D}) = b^T P_D^T M^{-1} P_D b = b^T P_D^T P_B P_D b = (r_{0,B}, z_{0,B}).$$

$$\begin{aligned}
(P_D A p_{0,D}, p_{0,D}) &= b^T P_D^T M^{-1} P_D A M^{-1} P_D b \\
&= b^T P_D^T M^{-1} P_D A P_D^T M^{-1} P_D b \\
&= b^T P_D^T P_B A P_B P_D b \\
&= (A p_{0,B}, p_{0,B}).
\end{aligned}$$

Hence, $\alpha_{0,D} = \alpha_{0,B}$.

$$\begin{aligned}
r_{1,D} &= P_D b - \alpha_{0,D} P_D A M^{-1} P_D b = P_D b - \alpha_{0,D} A P_D^T M^{-1} P_D b \\
&= P_D b - \alpha_{0,B} A P_B P_D b \\
&= r_{1,B}.
\end{aligned}$$

$$\begin{aligned}
x_{1,D} &= Z E^{-1} Z^T b + \alpha_{0,D} P_D^T M^{-1} P_D b \\
&= Z E^{-1} Z^T b + \alpha_{0,B} P_B P_D b = x_{1,B}
\end{aligned}$$

$$\begin{aligned}
P_D^T z_{1,D} &= P_D^T M^{-1} r_{1,D} = P_D^T M^{-1} P_D r_{1,D} \\
&= P_B P_D r_{1,D} = P_B r_{1,B} = z_{1,B}.
\end{aligned}$$

Thus

$$\begin{aligned}
(r_{1,B}, z_{1,B}) &= (r_{1,D}, P_D^T z_{1,D}) \\
&= (P_D r_{1,D}, z_{1,D}) = (r_{1,D}, z_{1,D}).
\end{aligned}$$

Hence $\beta_{0,D} = \beta_{0,B}$. Next,

$$\begin{aligned}
p_{1,B} &= z_{1,B} + \beta_{0,B} p_{0,B} \\
&= P_D^T z_{1,D} + \beta_{0,B} P_B P_D b \\
&= P_D^T z_{1,D} + \beta_{0,B} P_D^T M^{-1} P_D b \\
&= P_D^T (z_{1,D} + \beta_{0,D} p_{0,D}) \\
&= P_D^T p_{1,D}.
\end{aligned}$$

Now assume that the above identities hold for $j-1$ and that $(r_{j,B}, z_{j,B}) = (r_{j,D}, z_{j,D})$ holds. We then have

$$\begin{aligned}
(A p_{j,B}, p_{j,B}) &= (A P_D^T p_{j,D}, P_D^T p_{j,D}) \\
&= p_{j,D}^T P_D A P_D^T p_{j,D} \\
&= (P_D A p_{j,D}, p_{j,D}).
\end{aligned}$$

Hence, $\alpha_{j,D} = \alpha_{j,B}$. Since

$$\begin{aligned} x_{j+1,B} &= x_{j,B} + \alpha_{j,B} p_{j,B}, \\ \tilde{x}_{j+1,D} &= \tilde{x}_{j,D} + \alpha_{j,D} p_{j,D}, \end{aligned}$$

we obtain

$$\begin{aligned} x_{j+1,D} &= ZE^{-1}Z^T + P_D \tilde{x}_{j,D} + \alpha_{j,D} P_D^T p_{j,D} \\ &= x_{j,B} + \alpha_{j,B} p_{j,B} \\ &= x_{j+1,B}. \end{aligned}$$

$$\begin{aligned} r_{j+1,B} &= r_{j,B} - \alpha_{j,B} A p_{j,B} \\ &= P_D r_{j,D} - \alpha_{j,D} A P_D^T p_{j,D} \\ &= P_D r_{j,D} - \alpha_{j,D} P_D A p_{j,D} \\ &= P_D r_{j+1,D} \\ &= r_{j+1,D}. \end{aligned}$$

Moreover,

$$\begin{aligned} z_{j+1,B} &= P_B r_{j+1,B} = P_B P_D r_{j+1,D} \\ &= P_D^T M^{-1} P_D r_{j+1,D} = P_D^T M^{-1} r_{j+1,D} \\ &= P_D^T z_{j+1,D}. \end{aligned}$$

$$\begin{aligned} (r_{j+1,B}, z_{j+1,B}) &= (P_D r_{j+1,D}, P_D^T z_{j+1,D}) \\ &= r_{j+1,D}^T P_D^T z_{j+1,D} \\ &= (r_{j+1,D}, z_{j+1,D}). \end{aligned}$$

Hence, $\beta_{j,B} = \beta_{j,D}$. Next we have

$$\begin{aligned} p_{j+1,B} &= z_{j+1,B} + \beta_{j,B} p_{j,B} \\ &= P_D^T z_{j+1,D} + \beta_{j,D} P_D^T p_{j,D} \\ &= P_D^T p_{j+1,D}, \end{aligned}$$

which completes the proof. □

In the following we show how the eigenvalues and the condition number of the system preconditioned by balancing Neumann-Neumann behave if we choose a coarser grid. Therefore let $Z_1 \in \mathbb{R}^{n \times r}$ and $Z_2 \in \mathbb{R}^{n \times s}$ with $rank Z_1 = r$ and $rank Z_2 = s$. Define

$$\begin{aligned} E_1 := Z_1^T A Z_1 \qquad &\text{and} \qquad E_2 := Z_2^T A Z_2, \\ P_{D_1} = I - A Z_1 E_1^{-1} Z_1^T \qquad &\text{and} \qquad P_{D_2} = I - A Z_2 E_2^{-1} Z_2^T. \end{aligned}$$

Moreover let

$$P_{B_1} = P_{D_1}^T M^{-1} P_{D_1} + Z_1 E_1^{-1} Z_1^T \text{ and } P_{B_2} = P_{D_2}^T M^{-1} P_{D_2} + Z_2 E_2^{-1} Z_2^T. \tag{20}$$

We then have

Theorem 3.8 *Let A and $M \in \mathbb{R}^{n \times n}$ be symmetric positive definite. Let P_{B_1} and P_{B_2} be defined as in* (20)*. If $Im Z_1 \subseteq Im Z_2$, then*

$$\lambda_n(P_{B_1}A) \geq \lambda_n(P_{B_2}A). \quad (21)$$
$$\lambda_{r+1}(P_{B_1}A) \leq \lambda_{s+1}(P_{B_2}A). \quad (22)$$

Moreover

$$cond(P_{B_1}A) \geq cond(P_{B_2}A).$$

Proof:
Theorem 2.12 in [16] states that

$$\begin{aligned} \lambda_n(M^{-1}P_{D_1}A) &\geq \lambda_n(M^{-1}P_{D_2}A), \\ \lambda_{r+1}(M^{-1}P_{D_1}A) &\leq \lambda_{s+1}(M^{-1}P_{D_2}A). \end{aligned}$$

Thus, with Theorem 2.8 we get

$$\begin{aligned} cond(P_{B_1}A) &= \frac{max(1,\lambda_n(M^{-1}P_{D_1}A))}{min(1,\lambda_{r+1}(M^{-1}P_{D_1}A))} \\ &\geq \frac{max(1,\lambda_n(M^{-1}P_{D_2}A))}{min(1,\lambda_{s+1}(M^{-1}P_{D_2}A))} \\ &= cond(P_{B_2}A). \end{aligned}$$

□

If a finer grid is used as a coarse grid in the balancing Neumann-Neumann preconditioner the amount of work to solve the coarse grid system is increasing. But then Theorem 3.8 states that the condition number of the system preconditioned by the balancing Neumann-Neumann preconditioner decreases. So the more work on the coarse grid system will lead to less iterations.

4 Numerical experiments

In all our numerical experiments, the multiplication $y = E^{-1}b$ is done by solving y from $Ey = b$, where E is decomposed in its Cholesky factor. In this section, balancing Neumann Neumann is abbreviated as BNN. The choice of the boundary conditions is such that all problems have as exact solution the vector with components equal to 1. In order to make the convergence behavior representative for general problems we chose a random vector as starting solution, in stead of the zero start vector.

4.1 Artificial test problems

We apply both methods (Deflation and BNN) to the Poisson equation. It appears that in the numerical experiments $\|x - x_{k,D}\|_A \leq \|x - x_{k,B}\|_A$ but the differences are very small. From Theorem 2.8 it follows that the spectrum of the balancing Neumann Neumann preconditioner consists of two parts: in one part the eigenvalues are equal to 1, and in the other part the eigenvalues are equal to the nonzero eigenvalues of the Deflated matrix. This suggests that if the eigenvalues equal to 1 are interior eigenvalues the convergence is close to the convergence of the preconditioned Deflation method, otherwise these eigenvalues may influence the convergence.

Scaling properties
Note that P_DA is scaling invariant whereas P_BA is not scaling invariant. This means that if deflation is applied to a system $\gamma Ax = \gamma b$ the effective condition number of $P_{D\gamma A}\gamma A = (I - \gamma AZ(Z^T\gamma AZ)^{-1}Z^T)\gamma A$ is independent of the scalar γ, i.e

$$\kappa_{eff}(P_{D\gamma A}\gamma A) = \frac{\gamma\lambda_n(P_{DA}A)}{\gamma\lambda_{r+1}(P_{DA}A)} = \kappa_{eff}(P_{DA}A).$$

Whereas the condition number of $P_B\gamma A$ depends on the choice of γ,

$$\kappa(P_{B\gamma A}\gamma A) \neq \kappa(P_{BA}A).$$

To check this in practice, we do experiments with BNN using various values of γ. From Figure 1 it appears that the convergence of the balancing Neumann Neumann preconditioner is worse if $\gamma \neq 1$. We note that the Deflation method (for all values of γ) has the same convergence as the BNN method with $\gamma = 1$.

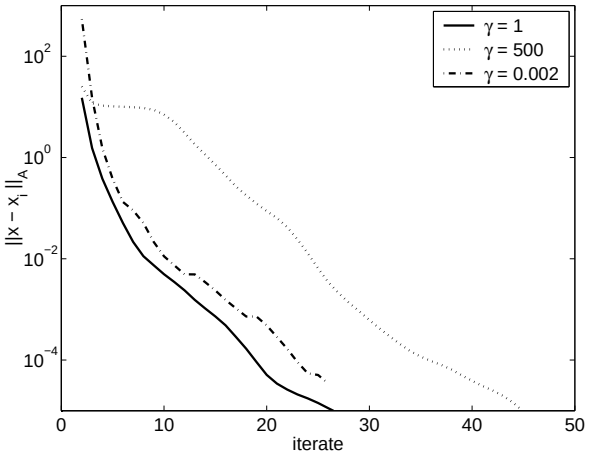

Figure 1: Comparison of the balancing Neumann Neumann preconditioner for various values of γ

Inaccurate solution

If the dimensions of matrix E becomes large (i.e. many projection vectors are used) it seems to be a good idea to compute E^{-1} approximately (by an iterative method/or to do the procedure recursively). It appears that the balancing Neumann-Neumann preconditioner is insensitive to the accuracy of the approximation of E^{-1}, while Deflation is sensitive to it.

To illustrate this we consider the same Poisson problem. In both examples 7 projection vectors are used. We replace E^{-1} by $\tilde{E}^{-1} = (I+\epsilon R)E^{-1}(I+\epsilon R)$, where R is a symmetric $r \times r$ matrix with

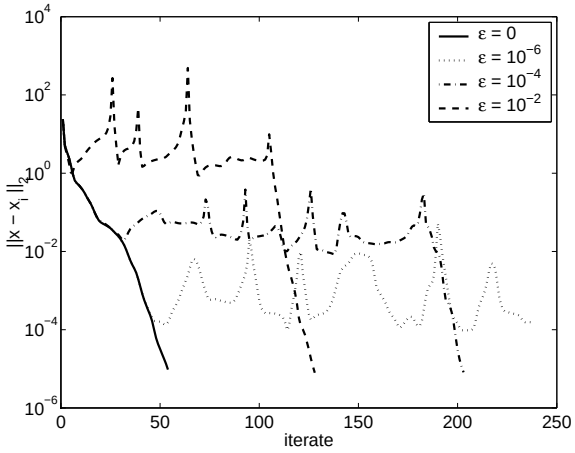

Figure 2: Convergence behavior of DICCG including perturbations

random elements chosen from the interval $[-\frac{1}{2}, \frac{1}{2}]$. From Figure 2 it follows that the convergence is good as long as $|\epsilon| < 10^{-6}$.

Starting solution for the BNN preconditioner

In Theorem 3.5 we have proven that $x_{k,B} = x_{k,D}$ if $x_{0,B} = ZE^{-1}Z^Tb$ and $\tilde{x}_{0,D} = 0$. In this paragraph we illustrate this by numerical examples. In Figure 3 we plot the convergence of BNN with start vector $x_{0,B} = ZE^{-1}Z^Tb$. It appears that the choice $\gamma = 500$ leads to the same results as $\gamma = 1$ (and Deflation). Furthermore, the convergence for the choice $\gamma = 0.002$ is initially also the same but later on the convergence becomes worse. This can be explained by rounding errors. Using the choice $\gamma = 0.002$ the eigenvalues equal to 1 are large with respect to the other eigenvalues. Initially, due to the start vector the components of the corresponding eigenvectors are zero or small. During iterations, the perturbations in large eigencomponents increase, which leads to the same convergence as if the method is started with $x_{0,B} = 0$. To enlarge this effect we have also done experiments where the matrix E^{-1} is replaced by $\tilde{E}^{-1}$ with $\epsilon = 10^{-2}$. The results are given in Figure 4. Note that the same effect now appears for both values of γ.

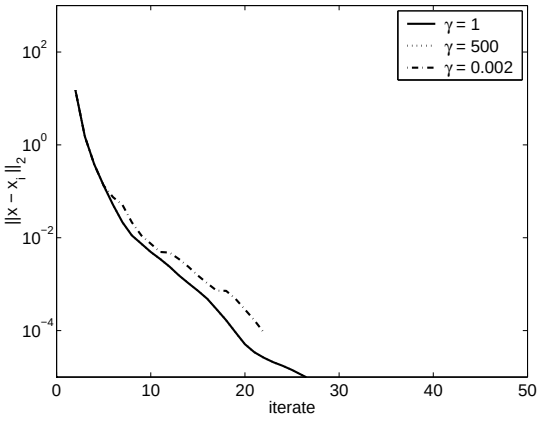

Figure 3: Convergence of BNN with $x_{0,B} = ZE^{-1}Z^Tb$ and $\epsilon = 0$

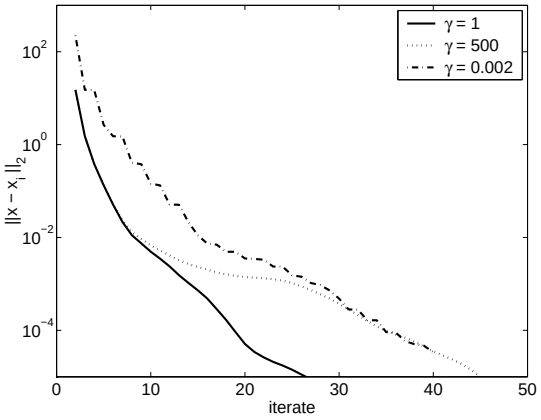

Figure 4: Convergence of BNN with $x_{0,B} = ZE^{-1}Z^Tb$ and $\epsilon = 10^{-2}$

4.2 Porous media flows

In this section we simulate a porous media oil flow in a 3 dimensional layered geometry, where the layers vary in thickness and orientation (see figures 5 and 6 for a 4 layer problem). Figure 5 shows a part of the earth's crust. The depth of this part varies between 3 and 6 kilometers, whereas horizontally its dimensions are 40 x 60 kilometers. The upper layer is a mixture of sandstone and shale and has a permeability of 10^{-4}. Below this layer, shale and sandstone layers are present with permeabilities of 10^{-7} and 10 respectively. We consider a problem with 9 layers. Five sandstone

layers are separated by four shale layers. At the top of the first sandstone/shale layer a Dirichlet boundary condition is posed, so the IC preconditioned matrix has 4 small eigenvalues. We use 4 *physical* projection vectors and stop if $\|r_k\|_2 \leq 10^{-5}$. Trilinear hexahedral elements are used and the total number of gridpoints is equal to 148185. The results are given in Table 1. It appears that the norm of the residuals for both preconditioners are the same. Due to extra work per iteration BNN costs more CPU time. In our implementation of BNN we used 2 matrix vector products 1 preconditioner vector product and the coarse grid operator is used 3 times. The computations are performed on an AMD Athlon, 1.4 GHz processor with 256 Mb of RAM. The code is compiled with FORTRAN g77 on LINUX.

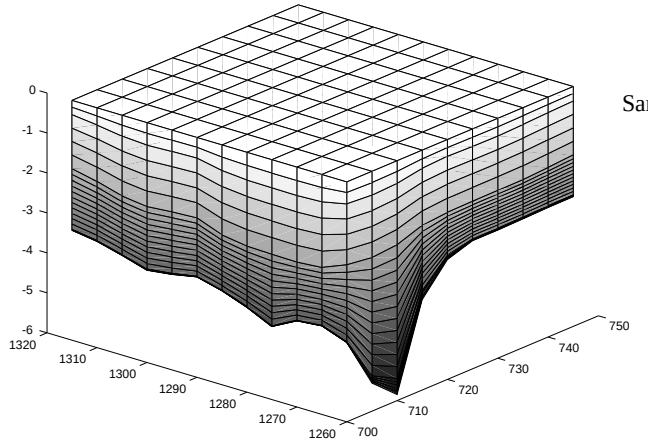

Figure 5: The geometry of an oil flow problem

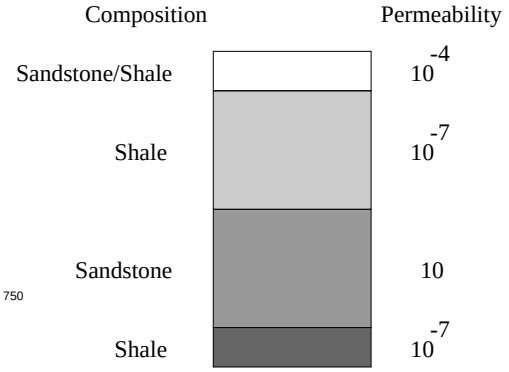

Figure 6: Permeabilities for each layer

method	deflation	BNN
iterations	36	36
CPU time in seconds	6.3	9.8

Table 1: The results for the oil flow problem

5 Conclusions

In this paper we compared various preconditioners for the numerical solution of partial differential equations.

We have given a detailed comparison of the well-known balancing Neumann-Neumann preconditioner used in domain decomposition methods and the deflation preconditioner.

We proved that both preconditioners lead to almost the same spectra. The zero eigenvalues of the deflation preconditioned system are replaced by eigenvalues which are one if the balancing Neumann-Neumann preconditioner is used. Thus the effective condition number of the deflated preconditioned system is always, i.e. for all deflation vectors and all restrictions and prolongations, below or equal to the condition number of the system preconditioned by the balancing Neumann-Neumann preconditioner. Moreover, we proved that the A-norm of the errors of the iterates build by the deflation preconditioner is always below the A-norm of the errors of the iterates build by the balancing Neumann-Neumann preconditioner. Hence, the Conjugate Gradient method applied to the deflated preconditioned system converges always faster than the Conjugate Gradient method applied to the system preconditioned by the balancing Neumann-Neumann preconditioner. Additionally, the amount of work of one iteration of the deflation preconditioned system is less than the amount of work of one iteration of the balancing Neumann-Neumann preconditioned

system. Hence the deflation preconditioner leads to a less number of iterations and each iteration is of less amount of work.

Moreover, we established that the deflation preconditioner and the balancing Neumann-Neumann preconditioner produces the same iterates if one uses certain starting vectors. More preciously we showed which terms in the preconditioned Conjugate Gradient method are the same for both methods and which terms are different. Numerical results for porous media flows emphasized the theoretical results.

References

[1] J.H. Bramble, J.E. Pasciak, and A.H. Schatz. The construction of preconditioners for elliptic problems by substructuring. I. *Math. Comput.*, 47:103–134, 1986.

[2] H. De Gersem and K. Hameyer. A deflated iterative solver for magnetostatic finite element models with large differences in permeability. *Eur. Phys. J. Appl. Phys.*, 13:45–49, 2000.

[3] M. Dryja and O.B. Widlund. Schwarz methods of Neumann-Neumann type for three-dimensional elliptic finite element problems. *Comm. Pure Appl. Math.*, 48:121–155, 1995.

[4] M. Eiermann, O.G. Ernst, and O. Schneider. Analysis of acceleration strategies for restarted minimal residual methods. *J. Comput. Appl. Math.*, 123:261–292, 2000.

[5] C. Farhat and F.-X. Roux. A method of finite element tearing and interconnecting and its parallel solution algorithm. *Int. J. Numer. Methods Eng.*, 32:1205–1227, 1991.

[6] C. Farhat and F.-X. Roux. Implicit parallel processing in structural mechanics. *Comput. Mech. Adv.*, 2:1–124, 1994.

[7] J. Frank and C. Vuik. On the construction of deflation-based preconditioners. *SIAM Journal on Scientific Computing*, 23:442–462, 2001.

[8] R.A. Horn and C.R. Johnson. *Matrix Analysis.* Cambridge University Press, Cambridge, 1985.

[9] E .F. Kaasschieter. Preconditioned Conjugate Gradients for solving singular systems. *J. Comput. Appl. Math.*, 24:265–275, 1988.

[10] L.Yu. Kolotilina. Preconditioning of systems of linear algebraic equations by means of twofold deflation. I. Theory. *J. Math. Sci.*, 89:1652–1689, 1998.

[11] J. Mandel. Balancing domain decomposition. *Commun. Appl. Numer. Meth.*, 9:233–241, 1993.

[12] J. Mandel and M. Brezina. Balancing domain decomposition for problems with large jumps in coefficients. *Mathematics of Computation*, 216:1387–1401, 1996.

[13] L. Mansfield. On the Conjugate Gradient solution of the Schur complement system obtained from domain decomposition. *SIAM J. Numer. Anal.*, 27(6):1612–1620, 1990.

[14] L. Mansfield. Damped Jacobi preconditioning and coarse grid deflation for Conjugate Gradient iteration on parallel computers. *SIAM J. Sci. Stat. Comput.*, 12(6):1314–1323, 1991.

[15] R. B. Morgan. A restarted GMRES method augmented with eigenvectors. *SIAM J. Matrix Anal. Appl.*, 16(4):1154–1171, 1995.

[16] R. Nabben and C. Vuik. A comparison of deflation and coarse grid correction applied to porous media flow. *to appear in SIAM J. Numer. Anal.*

[17] R. A. Nicolaides. Deflation of Conjugate Gradients with applications to boundary value problems. *SIAM J. Numer. Anal.*, 24(2):355–365, 1987.

[18] L.F. Pavarino and O.B. Widlund. Balancing Neumann-Neumann methods for incompressible Stokes equations. *Comm. Pure Appl. Math.*, 55:302–335, 2002.

[19] A. Quarteroni and A. Valli. *Domain Decomposition Methods for Partial Differential Equations*. Oxford Science Publications, Oxford, 1999.

[20] B. Smith, P. Bjørstad, and W. Gropp. *Domain Decomposition*. Cambridge University Press, Cambridge, 1996.

[21] A. van der Sluis and H. A. van der Vorst. The rate of convergence of Conjugate Gradients. *Numer. Math.*, 48:543–560, 1986.

[22] F.J. Vermolen, C. Vuik, and A. Segal. Deflation in preconditioned Conjugate Gradient methods for finite element problems. *Journal of Computational Methods in Sciences and Engineering*, 2004. to appear.

[23] 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.*, 152:385–403, 1999.

[24] C. Vuik, A. Segal, J.A. Meijerink, and G.T. Wijma. The construction of projection vectors for a Deflated ICCG method applied to problems with extreme contrasts in the coefficients. *Journal of Computational Physics*, 172:426–450, 2001.

[25] C. Vuik, A. Segal, L. El Yaakoubi, and E. Dufour. A comparison of various deflation vectors applied to elliptic problems with discontinuous coefficients. *Applied Numerical Mathematics*, 41:219–233, 2002.