Preconditioners for two-by-two block matrices with square blocks

Owe Axelsson[1,2], Maya Neytcheva[2]

[1]Institute of Geonics, Czech Academy of Sciences, Ostrava, Czech Republic

[2]Department of Information Technology, Uppsala University, Uppsala, Sweden

owe.axelsson@it.uu.se, maya.neytcheva@it.uu.se

May 15, 2018

Abstract

Two-by-two block matrices with square blocks arise in the numerical treatment of numerous applications of practical significance, such as optimal control problems, constrained by a state equation in the form of partial differential equations, multiphase models, solving complex linear systems in real arithmetics, to name a few. Such problems lead to algebraic systems of equations with matrices of a certain two-by-two block form. For such matrices, a number of preconditioners has been proposed, some of them with tight eigenvalue bounds. In this paper it is shown that in particular one of them, referred to as *PRESB*, is very efficient, not only giving robust, favourable properties of the spectrum but also enabling an efficient implementation with low computational complexity. Various applications and generalizations of this preconditioning technique, such as in time-harmonic parabolic and Stokes equations, eddy current electromagnetic problems and problems with additional box-constraints, i.e. upper and/or lower bounds of the solution, are also discussed.

The method is based on the use of coupled inner-outer iterations, where the inner iteration can be performed to various relative accuracies. This leads to variable preconditioners, thus, a flexible version of a Krylov subspace iteration method must be used. Alternatively, some version of a defect-correction iterative method can be applied.

1 Introduction

Two-by-two block matrices with square blocks can be seen as generic two-by-two block matrices. Matrices with such a structure arise in a vast variety of applications, including discrete Navier-Stokes problems, linear elasticity, poro-elasticity, flow in porous media and numerous other coupled multiphysics problems. Matrices of this class arise also in domain decomposition, hierarchical finite element discretizations and two- and multilevel

frameworks, where the structure is imposed in order to gain efficiency in the underlying numerical solution method. Two-by-two block matrices *with square blocks* arise also in many applications such as when solving complex-valued matrix systems in real arithmetics, in some multiphase models and in solving various types of optimal control problems with partial differential equation (PDE) state equations. For time-harmonic problems, a complex-valued system arises which can be written in such a form to avoid use of complex arithmetics. For optimal control time-harmonic PDE-constrained problems a two-by-two or four-by-four block system of such a form arises, see for instance [1, 2] and the references therein. As shown in [1, 3, 2] such systems arise also in electromagnetic eddy current problems. A modification of this method for problems with local control and observation subdomains has appeared in [2]. There, the two-by-two block structure appears on two levels, the outermost, consisting of two two-by-two block matrices, and the inner level, consisting of the basic two-by-two block matrix structure.

The arising block matrices in the latter applications correspond to discretization of normally 3D space finite elements, which implies that very large systems must be solved. Therefore, iterative solution methods must be used. The choice of proper preconditioners is then crucial. For block-structured matrices, in general, the best performing preconditioners utilize the underlying block matrix structures. Clearly, all known preconditioning techniques for general two-by-two block matrices are applicable when the blocks are square. However, in the search of highly efficient and robust preconditioned iterative solvers, the utilization of the fact that the blocks are square brings additional advantages. For the target matrices, several efficient preconditioning methods have been presented in recent years. As the amount of published research is vast, we refer to a selection of articles only. The efficient preconditioning techniques include (i) the block-diagonal and block-triangular preconditioner , e.g., [1, 2, 4], (ii) the preconditioned modified Hermitian and skew Hermitian (PMHSS) iteration method, e.g. [5, 6, 7] and [3]. (iii) the preconditioned square block matrix (PRESB) in [8]-[11] and [3]. Preconditioners with other structures have been analysed in, for example, [12]. In nonlinear problems (not considered here) using Newton type methods, a sequence of such linear problems has to be solved. Here, three levels of iterations emerge. So, the total number of iterations on the inner level multiply up, which makes it even more necessary to use the most efficient methods on each level.

The purpose of the present paper is to make an analytical comparison of some of these methods and to present shorter and extended proofs of results that have appeared earlier. For various implementations and applications of these preconditioners, tested on complex symmetric algebraic systems and on systems arising from discrete optimal control problems, with partial differential equations as constraints, we refer to some recent publications, (i) [1, 13], (ii) [14, 6] and (iii) [8]-[15], [2] and [15], where extensive numerical tests can also be found.

It has been shown that, with one exception, namely the block-diagonal preconditioner for small values of the reluctivity problem parameter (see [13]), all of the methods are robust with respect to mesh, regularization method and problem parameters. Hence, the spectral condition number and, therefore, also the convergence factor is bounded uniformly with respect to all these parameters. For preconditioning that leads to tight and clustered

eigenvalue bounds, the methods have the tendency to achieve a superlinear rate of convergence. This is similar to the case when one uses operator preconditioning and has a compact perturbation type of preconditioner to the given operator, see e.g. [16]. Namely, the number of iterations, needed for each further reduction of the residual error with a fixed factor, decreases. In such cases, computational efficiency and ease of implementation become dominant factors rendering the efficiency of the method.

Iterative methods can be formulated as preconditioned Krylov subspace methods or as some form of defect-correction or iterative splitting method. For convenience of the reader we recall that the latter can be formulated as follows.

To solve a system $\mathcal{A}\boldsymbol{x} = \boldsymbol{f}$ with a generic preconditioner $\mathcal{P}$ such a method takes the form

$$\mathcal{P}(\boldsymbol{x}^{k+1} - \boldsymbol{x}^k) = \boldsymbol{f} - \mathcal{A}\boldsymbol{x}^k, \quad \text{that is,} \quad \mathcal{P}\boldsymbol{x}^{k+1} = \boldsymbol{f} + (\mathcal{P} - \mathcal{A})\boldsymbol{x}^k,\ k = 0, 1, \cdots .$$

Hence, $\mathcal{P}(\boldsymbol{x}^{k+1} - \boldsymbol{x}) = (\mathcal{P} - \mathcal{A})(\boldsymbol{x}^k - \boldsymbol{x})$, which shows that the convergence factor equals $\varrho(I - \mathcal{P}^{-1}\mathcal{A})$, where ρ is the spectral radius. In the considered methods the eigenvalues of $\mathcal{P}^{-1}\mathcal{A}$ are mostly real and contained in an interval $[\alpha, 1]$, where $1 \geq \alpha > 0$. Hence, the condition number equals $1/\alpha$ and the convergence factor equals $1 - \alpha$. As we see in the sequel, under certain conditions, for some of the considered methods $\alpha \geq 1/2$. Similar results hold when the eigenvalues are complex.

Besides block-matrix preconditioned Krylov subspace methods we present also a special version of an alternating direction iteration (ADI) matrix splitting method. For classical alternating direction iteration forms of matrix splitting methods, see [17], also [18]. On the innermost level of iterations all methods require the solution of some form of elliptic problem. For this it is assumed that some standard software toolbox or library is used. Since most of the mentioned problems to be solved in practice are three space dimensional and, hence, of very large scale, some iterative solution method must be used even as an inner solver. This requires the use of a variable [19] or flexible iteration method, such as FGMRES [20], as outer iteration method.

The paper is composed as follows. The general type of algebraic problems that are considered are given in Section 2, with assumptions of the properties of the block matrices involved. The methods are presented and analysed in Section 3. A survey of some problems, where such matrices occur, mostly from PDE-constrained optimization and a brief summary of some previously done numerical tests, is found in Section 4. The paper ends with some concluding remarks.

2 Algebraic problem types in focus

We consider the following special type of linear systems of algebraic equations, $\mathcal{A}\boldsymbol{w} = \boldsymbol{q}$, where

$$\mathcal{A} = \begin{bmatrix} A & B_2 \\ B_1 & -A \end{bmatrix}, \quad \boldsymbol{w} = \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix}, \quad \boldsymbol{q} = \begin{bmatrix} \boldsymbol{f} \\ \boldsymbol{g} \end{bmatrix}. \tag{1}$$

Here and throughout the paper $\mathcal{A}$ is used as a generic notation of such a two-by-two block matrix with square blocks. In (1), $A, \boldsymbol{x}, \boldsymbol{y}, \boldsymbol{f}, \boldsymbol{g}$ are real-valued and A, B or B_i, $i = 1, 2$ are square matrices. Unless specifically mentioned, the matrices B are considered real-valued as well. We also use the abbreviations 'spd' and 'spsd' to denote 'symmetric positive definite' and 'symmetric positive semidefinite', correspondingly. The nullspace of a matrix A is denoted by $\mathcal{N}(A)$.

One specific form of $\mathcal{A}$, namely, when B is symmetric,

$$\begin{bmatrix} A & B \\ B & -A \end{bmatrix} \qquad \text{or, in an alternative form,} \qquad \begin{bmatrix} A & -B \\ B & A \end{bmatrix} \tag{2}$$

is paid particular attention to, as it turns out that such matrices occur in numerous applications, some of them exemplified in Section 4.

For the more general form (1) we suppose that one of the following assumptions hold:
Assumption Ia. A is spd, $A + B_i$, $i = 1, 2$ are nonsingular $B_1 = B$, $B_2 = B^T$.

Assumption Ib. A is spsd, $B_1 = B$, $B_2 = B^T$ and $\mathcal{N}(A) \cap \mathcal{N}(B) = \{0\}$.

For the particular case (2), the assumptions $B_1 = B_2 = B$ and one of the assumptions IIa,b hold.

Assumption IIa. A and B are spsd (which, since $\mathcal{N}(A) \cap \mathcal{N}(B) = \{0\}$, implies that $A + B$ is spd). Here A and B are related via the eigenvalue problem $\mu(A + B)\boldsymbol{z} = B\boldsymbol{z}$, $\|\boldsymbol{z}\| \neq 0$.

Assumption IIb. A is spd and B is related to A via the parameter α in the generalized eigenvalue problems, $\mu A\boldsymbol{z} = B\boldsymbol{z}$, $\|\boldsymbol{z}\| \neq \boldsymbol{0}$, where $\alpha = \max Re(\mu)/|\mu|$, and $Re(\mu) \geq 0$.

Remark 2.1. *We note that, due to the fact that the blocks are square, there are several equivalent forms of the system (2), for instance, any of the forms*

$$\begin{bmatrix} B & -A \\ A & B \end{bmatrix} \begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{x} \end{bmatrix} = \begin{bmatrix} -\boldsymbol{f} \\ \boldsymbol{g} \end{bmatrix}, \qquad \begin{bmatrix} A & B \\ -B & A \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ -\boldsymbol{y} \end{bmatrix} = \begin{bmatrix} \boldsymbol{f} \\ -\boldsymbol{g} \end{bmatrix}.$$

This gives us the freedom to exchange the roles of A and B if this would make the solution of the system easier.

Remark 2.2. *In various problems the matrix in focus occurs in the form* $\begin{bmatrix} A & \beta B \\ B & -A \end{bmatrix}$ *or* $\begin{bmatrix} A & B \\ B & -\beta^{-1}A \end{bmatrix}$, *where $\beta > 0$. In such cases, in order to get a better relation between A and B and a better scaling of the whole system, and to bring it on a more conventional form for the application of some of the methods, it is advisable to scale it as*

$$\begin{bmatrix} A & \sqrt{\beta}B \\ \sqrt{\beta}B & -A \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \widetilde{\boldsymbol{y}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{f} \\ \widetilde{\boldsymbol{g}} \end{bmatrix},$$

where $\widetilde{\boldsymbol{y}} = \sqrt{\beta}\boldsymbol{y}$, $\widetilde{\boldsymbol{g}} = \sqrt{\beta}\boldsymbol{g}$, respectively, $\widetilde{\boldsymbol{y}} = \frac{1}{\sqrt{\beta}}\boldsymbol{y}$, $\widetilde{\boldsymbol{g}} = \sqrt{\beta}\boldsymbol{g}$.

In some cases the matrix B is complex. A typical form of it can be

$$\begin{bmatrix} A & E-iF \\ E+iF & -A \end{bmatrix}, \tag{3}$$

where A is spd and E, F - spsd.

3 Preconditioning methods

Without the claim that we exhaust all techniques that have been developed to precondition matrices of the type (1), (2), (3), and utilizing their structure, we present here four preconditioning techniques that can be considered as basic. We derive the resulting condition number of the preconditioned matrix and, therefore, also of the rate of convergence of the corresponding defect-correction iteration matrix.

3.1 Block-diagonal preconditioners

3.1.1 Real-valued off-diagonal blocks

Consider $\mathcal{P}_D = \begin{bmatrix} A+B & 0 \\ 0 & A+B \end{bmatrix}$ as a preconditioner to $\mathcal{A}_0 = \begin{bmatrix} A & B \\ B & -A \end{bmatrix}$ and $\mathcal{A}_1 = \begin{bmatrix} A & -B \\ B & A \end{bmatrix}$.

Proposition 3.1. *Let Assumption IIa hold true, A and B be spsd and $\mathcal{N}(A)\cap\mathcal{N}(B) = \{0\}$. Let μ be the eigenvalues of the generalized eigenvalue problem $\mu(A+B)\boldsymbol{x} = B\boldsymbol{x}$, $\boldsymbol{x} \neq \boldsymbol{0}$, i..e., $0 < \mu \leq 1$. Then the eigenvalues of the preconditioned matrices satisfy*

(i) $\lambda(\mathcal{P}_D^{-1}\mathcal{A}_0) \in [-1, -\frac{\sqrt{2}}{2}) \cup (\frac{\sqrt{2}}{2}, 1]$

(ii) $\lambda(\mathcal{P}_D^{-1}\mathcal{A}_1) = 1 - \mu \pm i\mu$, *i.e., the eigenvalues are contained in a triangle in the right half of the complex plane, see Figure 1.*

Proof.

(i) We write $\lambda\mathcal{P}_D \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix} = \begin{bmatrix} A & B \\ B & -A \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix}$ in the form $\lambda\mathcal{P}_D \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix} = \begin{bmatrix} A+B & 0 \\ 0 & -(A+B) \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix} + \begin{bmatrix} -B & B \\ B & B \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix}$. It follows that the eigenvalues are real. Since $A+B$ is positive definite, its square root exists. The matrix $\widehat{B} = (A+B)^{-\frac{1}{2}}B(A+B)^{-\frac{1}{2}}$ has eigenvalues μ and λ satisfies

$$det\left(\begin{bmatrix} \lambda - 1 + \mu & -\mu \\ -\mu & \lambda + 1 - \mu \end{bmatrix}\right) = 0,$$

that is, $\lambda^2 = (1-\mu)^2 + \mu^2$, or $\frac{1}{2} \leq \lambda^2 \leq 1$.

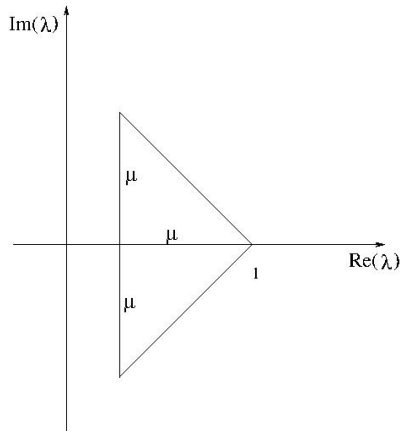

Figure 1: Triangular domain for the eigenvalues of $\mathcal{P}_D^{-1}\mathcal{A}_1$

(ii) In a similar way, $(1-\lambda)\mathcal{P}_D \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix} = \begin{bmatrix} B & B \\ -B & B \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix}$, or

$$(1-\lambda) \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix} = \begin{bmatrix} \mu & \mu \\ -\mu & \mu \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix},$$

that is, $(1-\lambda-\mu)^2+\mu^2=0$, or $\lambda = 1-\mu \pm i\mu$, which completes the proof.

□

It follows that the second method does not have a uniform convergence behaviour with respect to eigenvalues $\mu \to 1$.

3.1.2 Complex off-diagonal blocks

Consider now $\mathcal{A}$ as in (3) and the block-diagonal preconditioner used in [1, 13]. As shown in [1], matrices in this form arise in some time-harmonic optimal control problems. It is not directly applicable for the case where the control and state functions are prescribed only on a subset of the whole domain of definition. The following holds.

Proposition 3.2. *Let* $\mathcal{A} = \begin{bmatrix} A & E-iF \\ E+iF & -A \end{bmatrix}$, *where* A *is spd,* E, F *are spsd. Let* $\mathcal{P}_D = \begin{bmatrix} D & 0 \\ 0 & D \end{bmatrix}$, $D = A+E+F$, *and assume that* $ED^{-1}F = FD^{-1}E$. *This holds if* $F = \omega(A+\delta E)$, $\omega > 0$, $\delta \leq 1$. *Then* $(\mathcal{P}_D^{-1}\mathcal{A})^2$ *is block-diagonal and its eigenvalues are real and contained in the interval* $\frac{1}{4} \leq \lambda\left((\mathcal{D}^{-1}\mathcal{A})^2\right) \leq 1$. *This holds uniformly with respect to both the discretization and the problem parameters. If* $F = \omega A$, $\omega > 0$, *then* $\frac{1}{3} \leq \frac{1}{2(1+\omega/(1+\omega^2))} \leq \lambda((\mathcal{P}_D^{-1}\mathcal{A})^2) \leq 1$.

Proof. See [2]. The proof is based on the fact that

$$
\begin{aligned}
(\mathcal{P}_D^{-1}\mathcal{A})^2 &= \begin{bmatrix} \tilde{A} & \widetilde{E}-i\widetilde{F} \\ \widetilde{E}+i\widetilde{F} & -\tilde{A} \end{bmatrix}^2 = \begin{bmatrix} \tilde{A}^2+\widetilde{E}^2+\widetilde{F}^2 & \tilde{A}(\widetilde{E}-i\widetilde{F})-(\widetilde{E}-i\widetilde{F})\tilde{A} \\ (\widetilde{E}+i\widetilde{F})\tilde{A}-\tilde{A}(\widetilde{E}+i\widetilde{F}) & \tilde{A}^2+\widetilde{E}^2+\widetilde{F}^2 \end{bmatrix} \\
&= \begin{bmatrix} \tilde{A}+\widetilde{E}^2+\widetilde{F}^2 & 0 \\ 0 & \tilde{A}+\widetilde{E}^2+\widetilde{F}^2 \end{bmatrix},
\end{aligned}
$$

i.e., is block-diagonal, where $\tilde{A} = D^{-1}A$ etc. □

As an example, in time-harmonic optimal control problems, discretized using finite elements, $A = M$, $E = \sqrt{\beta}$, $F = i\sqrt{\beta}\omega M$, where M is a mass matrix, K is the discretization of the negative Laplacian and $\omega = k\frac{2\pi}{T}$, $k = 0, 1, \cdots$ is the angular frequency of the harmonic control function $u(x,t) = e^{-i\omega t}u_0(x)$. As has been shown earlier in [1, 13], this proposition shows that the eigenvalues of the block-diagonally preconditioned matrix $\left\{\begin{bmatrix} D^{-1} & 0 \\ 0 & D^{-1} \end{bmatrix}\begin{bmatrix} M & \sqrt{\beta}(K-i\omega M) \\ \sqrt{\beta}(K+i\omega M) & -M \end{bmatrix}\right\}^2$ are bounded below by $\frac{1}{2(1+\omega/(1+\omega^2))}$. Hence, for small or large values of ω the lower bound is close to $1/2$, the value taken for $\omega = 0$.

Note that this result holds for the square of the preconditioned matrix, i.e. corresponds to a double amount of iterations as compared to a preconditioning method for which the condition number is bounded by 2. Alternatively, it can be said to correspond to a convergence factor $\frac{1}{\sqrt{2}}$.

3.2 The HSS/PMHSS method

The technique to construct solution methods and preconditioners for linear systems, based on the Hermitian – Skew-Hermitian splitting (HSS) of the system matrix is of an alternating directions type. It has been used in various contexts and applied to scalar and vector equations. At present (May 2018), a bibliography search on 'Hermitian skew-Hermitian' via Google Scholar encounters over 10000 published items and a comprehensive survey of those goes beyond the aim of this paper. As the method is used to solve linear systems with two-by-two square blocks, we describe briefly its origin and developments.

As the name suggests, it involves the Hermitian part $H = \frac{1}{2}(\boldsymbol{A} + \boldsymbol{A}^*)$ and the skew-symmetric part $S = \frac{1}{2}(\boldsymbol{A} - \boldsymbol{A}^*)$ of a generic matrix $\boldsymbol{A} \in C^{n\times n}$ and we aim at solving $\boldsymbol{A}\widehat{\boldsymbol{x}} = \widehat{\boldsymbol{b}}$, $\widehat{\boldsymbol{x}} = \boldsymbol{x} + i\boldsymbol{y}, \widehat{\boldsymbol{b}} = \boldsymbol{a} + i\boldsymbol{b} \in C^n$. Here A^* is the complex conjugate of A. The method originates in [21] in the form

$$
\begin{aligned}
(\alpha I + H)\widehat{\boldsymbol{x}}^{k+\frac{1}{2}} &= (\alpha I - S)\widehat{\boldsymbol{x}}^k + \widehat{\boldsymbol{b}} \\
(\alpha I + S)\widehat{\boldsymbol{x}}^k &= (\alpha I - H)\widehat{\boldsymbol{x}}^{k+\frac{1}{2}} + \widehat{\boldsymbol{b}}
\end{aligned}
$$

Clearly, the largest computational cost in HSS is to solve systems with $\alpha I + H$ and $\alpha I + S$ per iteration. The convergence is shown by proving that the spectral radius of the iteration

matrix $M(\alpha) = (\alpha I + S)^{-1}(\alpha I - H)(\alpha I + H)^{-1}(\alpha I - S)$ is bounded by 1 for all $\alpha > 0$ and an optimal value of α is derived, as a function of the minimum and the maximum eigenvalues of H. As the method in this form is not robust with respect to the problem size, numerous follow-up papers have dealt with its improvement, such as [5, 6]. The HSS method is applicable also for matrices in block form, saddle point form etc. We consider its performance for matrices of the considered form only.

Remark 3.1. *What concerns the solution of systems with complex symmetric matrices, the HSS approach can be applied in two ways: directly to the complex matrix* $\boldsymbol{A} = A + iB$ *or after the system* $\boldsymbol{A}\widehat{\boldsymbol{x}} = \widehat{\boldsymbol{b}}$ *has been written as a twice larger real system*

$$\begin{bmatrix} A & -B \\ B & A \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix} = \begin{bmatrix} \boldsymbol{a} \\ \boldsymbol{b} \end{bmatrix}. \tag{4}$$

The Preconditioned Modified HSS method, applied directly to the complex system, cf. [5], leads to

$$\begin{aligned} (\alpha V + A)\widehat{\boldsymbol{x}}^{k+\frac{1}{2}} &= (\alpha V - iB)\widehat{\boldsymbol{x}}^{k} + \widehat{\boldsymbol{b}}, \\ (\alpha V + B)\widehat{\boldsymbol{x}}^{k+1} &= (\alpha V + iA)\widehat{\boldsymbol{x}}^{k+\frac{1}{2}} - i\widehat{\boldsymbol{b}}. \end{aligned}$$

Here V *is an spd matrix to be chosen. In the case when* A *is spd, the choice* $V = A$, $\alpha = 1$ *leads to*

$$(A + B)\widehat{\boldsymbol{x}}^{k+1} = \frac{1}{2}\left((i+1)(A - iB)\widehat{\boldsymbol{x}}^{k} + (1-i)\widehat{\boldsymbol{b}}\right), \tag{5}$$

thus, we need to solve only one system with real matrix and a complex right hand size. Due to that, given that A *is spd, this version of the PMHSS method has been performing fastest on some of the benchmark test problems in [15]. Thus, if we possess an iterative solution method that handles real matrices and complex right hand sides, the method in (5) is to be recommended. Further, we note that any real system of the form*

$$\begin{bmatrix} A & -\alpha B \\ \beta B & \gamma A \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix} = \begin{bmatrix} \boldsymbol{a} \\ \boldsymbol{b} \end{bmatrix},$$

where $\alpha > 0, \beta > o, \gamma > 0$, *can be written as a complex system as*

$$\left(\sqrt{\frac{\alpha\gamma}{\beta}}A + i\alpha B\right)\left(\sqrt{\frac{\beta}{\alpha\gamma}}\boldsymbol{x} + i\boldsymbol{y}\right) = \boldsymbol{a} + i\sqrt{\frac{\alpha}{\beta\gamma}}\boldsymbol{b}.$$

The tests in [15] have been performed in Matlab, where such a solver is implemented. However, up to the knowledge of the authors, in the available numerical linear algebra packages, used for large scale problems, such a solver is not included and complex systems are solved after formulating them in real form. Therefore, we consider further the PMHSS method applied to systems as in (4).

We consider the derivations and the results in [6, 8, 9] as most closely related to the topics discussed here. We introduce directly the Preconditioned Modified Hermitian – skew-Hermitian (PMHSS) method, as in [9].

3.2.1 Matrix $\mathcal{A} = \begin{bmatrix} A & -B \\ B & A \end{bmatrix}$

Consider again the solution of a linear system

$$\mathcal{A} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix} = \begin{bmatrix} \boldsymbol{f} \\ \boldsymbol{g} \end{bmatrix}, \tag{6}$$

where A and B are square, spsd and $\mathcal{N}(A) \cap \mathcal{N}(B) = \{0\}$.

The original form of PMHSS can be derived, based on the following two matrix splittings. The first is of $\mathcal{A}$ itself,

$$\mathcal{A} = \begin{bmatrix} \alpha V + A & 0 \\ 0 & \alpha V + A \end{bmatrix} - \begin{bmatrix} \alpha V & B \\ -B & \alpha V \end{bmatrix}$$

and the other associated alternating direction splitting reads as,

$$\widehat{\mathcal{A}} = \begin{bmatrix} B & A \\ -A & B \end{bmatrix} = \begin{bmatrix} \alpha V + B & 0 \\ 0 & \alpha V + B \end{bmatrix} - \begin{bmatrix} \alpha V & -A \\ A & \alpha V \end{bmatrix}.$$

Here V is an spd matrix to be chosen and $\alpha > 0$ is a real method parameter to be determined. Following [14, 6], to solve (6) we use the defect-correction or matrix splitting iteration method. Namely, given an initial vector $\begin{bmatrix} \boldsymbol{x}^{(0)} \\ \boldsymbol{y}^{(0)} \end{bmatrix}$, for $k = 0, 1, \cdots$ until convergence, solve

$$\begin{bmatrix} \alpha V + A & 0 \\ 0 & \alpha V + A \end{bmatrix} \begin{bmatrix} \boldsymbol{x}^{(k+1/2)} \\ \boldsymbol{y}^{(k+1/2)} \end{bmatrix} = \begin{bmatrix} \alpha V & B \\ -B & \alpha V \end{bmatrix} \begin{bmatrix} \boldsymbol{x}^{(k)} \\ \boldsymbol{y}^{(k)} \end{bmatrix} + \begin{bmatrix} \boldsymbol{f} \\ \boldsymbol{g} \end{bmatrix}, \tag{7}$$

$$\begin{bmatrix} \alpha V + B & 0 \\ 0 & \alpha V + B \end{bmatrix} \begin{bmatrix} \boldsymbol{x}^{(k+1)} \\ \boldsymbol{y}^{(k+1)} \end{bmatrix} = \begin{bmatrix} \alpha V & -A \\ A & \alpha V \end{bmatrix} \begin{bmatrix} \boldsymbol{x}^{(k+1/2)} \\ \boldsymbol{y}^{(k+1/2)} \end{bmatrix} + \begin{bmatrix} \boldsymbol{g} \\ -\boldsymbol{f} \end{bmatrix}. \tag{8}$$

Note the reordering of the equations and change of sign in the right-hand side of the second equation. This is analogous to an alternating direction splitting type method.

Since α is a factor of V and both α and V are to be chosen, we can include α in V or, equivalently, put $\alpha = 1$.

We make the following observations. First, whether applied as a solution procedure or as a preconditioner, PMHSS in the form (7)-(8) requires the solution of *four* linear systems and this makes the method computationally heavy. To reduce the arithmetic cost, we can choose V to be equal to either A or B, whichever is nonsingular. Then it makes sense to let α to be separately chosen. Numerous HSS-related papers, such as [6], include efforts to determine an optimal value of α, that would minimize the convergence factor of the method. However, the provided numerical examples show that $\alpha = 1$ leads to nearly the same convergence results as for the optimally computed value of the parameter. Therefore, in the analysis below we set $\alpha = 1$. In this case the method becomes parameter-independent, the convergence factor reduces to $\frac{\sqrt{2}}{2}$ (a factor $\sqrt{2}$ larger than that for the

PRESB method, Section 3.3) and the framework requires the solution of two systems of equations with the matrix $A+B$.

Convergence of the method and its variants has been shown in the related publications. Clearly, it is also important to estimate the rate of convergence. In order to estimate the convergence factor of the above iterative procedure, in a somewhat new way, we compute bounds for the eigenvalues of the corresponding iteration matrix,

$$G = \begin{bmatrix} (V+B)^{-1} & 0 \\ 0 & (V+B)^{-1} \end{bmatrix} \begin{bmatrix} V & -A \\ A & V \end{bmatrix} \begin{bmatrix} (V+A)^{-1} & 0 \\ 0 & (V+A)^{-1} \end{bmatrix} \begin{bmatrix} V & B \\ -B & V \end{bmatrix}.$$

Note, that if $\begin{bmatrix} \widehat{\boldsymbol{x}} \\ \widehat{\boldsymbol{y}} \end{bmatrix}$ is the exact solution, then

$$\begin{bmatrix} \boldsymbol{x}^{(k+1)} - \widehat{\boldsymbol{x}} \\ \boldsymbol{y}^{(k+1)} - \widehat{\boldsymbol{y}} \end{bmatrix} = G \begin{bmatrix} \boldsymbol{x}^{(k)} - \widehat{\boldsymbol{x}} \\ \boldsymbol{y}^{(k)} - \widehat{\boldsymbol{y}} \end{bmatrix}, k = 0, 1, \cdots$$

Therefore, we must estimate the spectral radius $\rho(G)$ of G.

Proposition 3.3. *Assume that A and B are spsd, $\mathcal{N}(A) \cap \mathcal{N}(B) = \emptyset$ and let V be spd. Let $\widetilde{A} = V^{-\frac{1}{2}} A V^{-\frac{1}{2}}$, $\widetilde{B} = V^{-\frac{1}{2}} B V^{-\frac{1}{2}}$, $\widehat{A} = (I+\widetilde{A})^{-1}\widetilde{A}$ and $\widehat{B} = (I+\widetilde{B})^{-1}\widetilde{B}$.*

(i) The spectral radius of G is bounded from above as

$$\rho(G) \leq \min_{\lambda \in Sp(\widehat{A},\widehat{B})} \sqrt{(1-\lambda)^2 + \lambda^2},$$

and $\rho(G) \leq \frac{\sqrt{2}}{2}$ if $V = A$ or $V = B$ if either A or B is spd. Here $Sp(\widehat{A}, \widehat{B})$ denotes the set of eigenvalues of the generalized eigenvalue problem $\widehat{A}\boldsymbol{v} = \lambda \widehat{B}\boldsymbol{v}$.

(ii) If $\widehat{A}$ and $\widehat{B}$ commute, which holds in particular if $V = A$ (or $V = B$), then

$$\rho(G) \leq 1 - 2(\mu + \nu + 2\mu\nu),$$

where μ is an eigenvalue of $\widehat{A}(I-\widehat{A})$ and ν - of $\widehat{B}(I-\widehat{B})$. If $\nu = 0$ and $\mu = \frac{1}{4}$, then $\max \rho(G) = \frac{\sqrt{2}}{2}$ is attained. If $\widehat{B}$ is such that $\nu = \min \lambda(\widehat{B}(I-\widehat{B})) > 0$, the spectral radius is smaller than $\frac{\sqrt{2}}{2}$.

Proof. Make first a similarity transformation of G with $\begin{bmatrix} V^{\frac{1}{2}} & 0 \\ 0 & V^{\frac{1}{2}} \end{bmatrix}$, which results in

$$\begin{bmatrix} V^{\frac{1}{2}} & 0 \\ 0 & V^{\frac{1}{2}} \end{bmatrix} G \begin{bmatrix} V^{-\frac{1}{2}} & 0 \\ 0 & V^{-\frac{1}{2}} \end{bmatrix} = \\ \begin{bmatrix} (I+\widetilde{B})^{-1} & 0 \\ 0 & (I+\widetilde{B})^{-1} \end{bmatrix} \begin{bmatrix} I & -\widetilde{A} \\ \widetilde{A} & I \end{bmatrix} \begin{bmatrix} (I+\widetilde{A})^{-1} & 0 \\ 0 & (I+\widetilde{A})^{-1} \end{bmatrix} \begin{bmatrix} I & \widetilde{B} \\ -\widetilde{B} & I \end{bmatrix}.$$

This is followed by a second similarity transformation, now with the matrix $\begin{bmatrix} I & \widetilde{B} \\ -\widetilde{B} & I \end{bmatrix}$, giving

$$\begin{aligned} \widetilde{G} &= \begin{bmatrix} I & \widetilde{B} \\ -\widetilde{B} & I \end{bmatrix} \begin{bmatrix} (I+\widetilde{B})^{-1} & 0 \\ 0 & (I+\widetilde{B})^{-1} \end{bmatrix} \begin{bmatrix} I & -\widetilde{A} \\ \widetilde{A} & I \end{bmatrix} \begin{bmatrix} (I+\widetilde{A})^{-1} & 0 \\ 0 & (I+\widetilde{A})^{-1} \end{bmatrix} \\ &= \left(\begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix} - \begin{bmatrix} \widehat{B} & -\widehat{B} \\ \widehat{B} & \widehat{B} \end{bmatrix} \right) \left(\begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix} - \begin{bmatrix} \widehat{A} & \widehat{A} \\ -\widehat{A} & \widehat{A} \end{bmatrix} \right), \end{aligned} \tag{9}$$

where $\widehat{A} = (I+\widetilde{A})^{-1}\widetilde{A}$ and $\widehat{B} = (I+\widetilde{B})^{-1}\widetilde{B}$. Note, that the eigenvalues of $\widehat{A}$ and $\widehat{B}$ are contained in the interval $[0,1)$.

Let λ be an eigenvalue of $\widehat{A}$. Then the eigenvalues of the second factor in $\widetilde{G}$ satisfy

$$(1-\gamma_2) \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix} = \lambda \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix}, |\boldsymbol{x}| + |\boldsymbol{y}| \neq 0,$$

that is, $\gamma_2 = 1 - \lambda(1 \pm i) = 1 - \lambda \pm i\lambda$. Hence, $|\gamma_2|^2 = (1-\lambda)^2 - \lambda^2$, in other words,

$$\frac{1}{2} \leq |\gamma_2|^2 \leq 1.$$

The lower bound is taken for $\lambda = \frac{1}{2}$. Similarly, the spectral radius of the first factor in $\widetilde{G}$, γ_2, satisfies $\frac{1}{2} \leq |\gamma_1|^2 \leq 1$.

If we choose V to minimize the spectral radius $|\gamma_2|$, i.e., let $V = A$, then $\lambda = 1, \gamma_1 \leq 1$ and $\rho(\widetilde{G}) \leq \frac{1}{\sqrt{2}} = \frac{\sqrt{2}}{2}$, which proves part (i). This is the result in [6], derived in a different way.

We prove next part (ii). Writing out $\widetilde{G}$ in (9) shows that

$$\widetilde{G} = \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix} - \begin{bmatrix} \widehat{A} + \widehat{B} - 2\widehat{A}\widehat{B} & \widehat{B} - \widehat{A} \\ \widehat{A} - \widehat{B} & \widehat{A} + \widehat{B} - 2\widehat{A}\widehat{B} \end{bmatrix}$$

or

$$\widetilde{G} = \begin{bmatrix} I - (\widehat{A} + \widehat{B}) + 2\widehat{A}\widehat{B} & \widehat{A} - \widehat{B} \\ \widehat{B} - \widehat{A} & I - (\widehat{A} + \widehat{B}) + 2\widehat{A}\widehat{B} \end{bmatrix}.$$

Form now its normal matrix, $\widetilde{G}^T\widetilde{G}$. Computation shows that

$$\widetilde{G}^T\widetilde{G} = \begin{bmatrix} D & 0 \\ 0 & D \end{bmatrix},$$

where

$$\begin{aligned} D &= I - 2(\widehat{A} + \widehat{B}) + 2(\widehat{A}^2 + \widehat{B}^2) + 4\widehat{A}\widehat{B}(I - \widehat{A})(I - \widehat{B}) \\ &= I - 2X - 2Y + 4XY \end{aligned}$$

with $X = \widehat{A}(I - \widehat{A})$ and $Y = \widehat{B}(I - \widehat{B})$. Note that $0 < X < \frac{1}{4}I$ and $0 < Y < \frac{1}{4}I$. It holds also that

$$D = I - 2(X + Y(1 - 2X) \text{ and } D = I - 2(Y + X(1 - 2Y).$$

It follows that the upper bound $D \leq \frac{1}{4}I$ is taken for $Y = 0$, $X = \frac{1}{4}I$, respectively, for $X = 0$, $Y = \frac{1}{4}I$. Hence, $\rho(G) \leq \frac{\sqrt{2}}{2}$. If $X = \frac{1}{4}I$ but Y is spd, then $\rho(G) \leq \sqrt{\frac{1}{2} - \min(\lambda(Y))}$. □

Remark 3.2. *Note, that in part (ii) the eigenvalues are real, while in part (i) and [21, 6, 22], the eigenvalues are complex. This could make the method in general less suitable as a preconditioning technique for either a Krylov subspace method or a Chebyshev iteration method. The idea to use the Chebyshev method to solve the arising systems in PMHSS is analysed in [23] and successfuly applied for the solution of a discrete PDE-constrained distributed control problem.*

3.2.2 General two-by-two block matrices

As mentioned, HSS and its modifications have been also applied for more general cases for matrices with rectangular off-diagonal blocks ([24, 25]), Sylvester equations ([26]), etc, For example, for the Stokes problem, in [25], a new version of HSS, named the regularized HSS (RHSS) method, is presented. Given,

$$\mathcal{A}\begin{bmatrix} x \\ y \end{bmatrix} := \begin{bmatrix} A & B^* \\ -B & 0 \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} f \\ g \end{bmatrix},$$

where A is spd and B has full rank, it is based on the two splittings

$$\mathcal{A} = \begin{bmatrix} A & 0 \\ 0 & Q \end{bmatrix} + \begin{bmatrix} 0 & B^* \\ -B & -Q \end{bmatrix}$$

and

$$\mathcal{A} = \begin{bmatrix} 0 & B^* \\ -B & Q \end{bmatrix} + \begin{bmatrix} A & 0 \\ 0 & -Q \end{bmatrix},$$

where Q is an Hermitian positive semidefinite matrix to be chosen. The corresponding alternating direction type of iteration method takes the following form.

Given $\begin{bmatrix} \boldsymbol{x}^{(0)} \\ \boldsymbol{y}^{(0)} \end{bmatrix}$, for $k = 0, 1, \cdots$ until convergence let

$$\begin{bmatrix} \alpha I + A & 0 \\ 0 & \alpha I + Q \end{bmatrix}\begin{bmatrix} \boldsymbol{x}^{(k+1/2)} \\ \boldsymbol{y}^{(k+1/2)} \end{bmatrix} = \begin{bmatrix} \alpha I & -B^* \\ B & \alpha I + Q \end{bmatrix}\begin{bmatrix} \boldsymbol{x}^{(k)} \\ \boldsymbol{y}^{(k)} \end{bmatrix} + \begin{bmatrix} \boldsymbol{f} \\ \boldsymbol{g} \end{bmatrix} \tag{10}$$

and

$$\begin{bmatrix} \alpha I & B^* \\ -B & \alpha I + Q \end{bmatrix}\begin{bmatrix} \boldsymbol{x}^{(k+1)} \\ \boldsymbol{y}^{(k+1)} \end{bmatrix} = \begin{bmatrix} \alpha I - A & 0 \\ 0 & \alpha I + Q \end{bmatrix}\begin{bmatrix} \boldsymbol{x}^{(k+1/2)} \\ \boldsymbol{y}^{(k+1/2)} \end{bmatrix} + \begin{bmatrix} \boldsymbol{f} \\ \boldsymbol{g} \end{bmatrix}. \tag{11}$$

This can be implemented as

(i) Solve $(\alpha I + A)\boldsymbol{x}^{(k+1/2)} = \alpha \boldsymbol{x}^{(k)} - B^* \boldsymbol{y}^{(k)} + \boldsymbol{f}$

(ii) Solve

$$\begin{bmatrix} \alpha I & B^* \\ -B & \alpha I + Q \end{bmatrix} \begin{bmatrix} \boldsymbol{x}^{(k+1)} \\ \boldsymbol{y}^{(k+1)} \end{bmatrix} = \begin{bmatrix} \boldsymbol{f}^{(k+1/2)} \\ \boldsymbol{g}^{(k+1/2)} \end{bmatrix}, \tag{12}$$

where

$$\begin{bmatrix} \boldsymbol{f}^{(k+1/2)} \\ \boldsymbol{g}^{(k+1/2)} \end{bmatrix} = \begin{bmatrix} (\alpha I - A)\boldsymbol{x}^{(k+1/2)} + \boldsymbol{f} \\ B\boldsymbol{x}^{(k)} + (\alpha I + Q)\boldsymbol{y}^{(k)} + 2\boldsymbol{g} \end{bmatrix}.$$

Here the second remainder term is obtained by adding the two second equations remainder terms in (10) and (11). The system (12) can be solved via the reduced Schur complement system,

$$\left(\alpha I + Q + \frac{1}{\alpha} BB^*\right) \boldsymbol{y}^{(k+1)} = \frac{1}{\alpha} B\boldsymbol{f}^{(k+1/2)} + \boldsymbol{g}^{(k+1/2)}.$$

After computation of $\boldsymbol{y}^{(k+1)}$, then $\boldsymbol{x}^{(k+1)}$ is found as

$$\boldsymbol{x}^{(k+1)} = \frac{1}{\alpha} \left(\boldsymbol{f}^{(k+1/2)} - B^* \boldsymbol{y}^{(k+1)}\right).$$

It is seen that during each iteration, the method requires a solution with the matrix $\alpha I + A$ and one with $\alpha I + \frac{1}{\alpha} BB^* + Q$. For small values of α the latter is ill-conditioned and its solution can be costly. Both direct solvers and inner PCG iterative solvers have been used and it has been found that the inner iterations with use of a flexible GMRES method as an outer iteration method could save computational labor and computer time.

3.3 The PRESB method

We present now a preconditioning technique, that, although successfully used in various contexts since 2012, has not been given any short name, cf. [27, 28, 29, 15, 8, 9, 10, 30, 11]. We refer to it as the *Preconditioning for REal matrices with Square Blocks (PRESB)*. It has been also shown that it is applicable for complex-valued blocks, where we can interpret the name as *PREconditioning for Square Blocks*.

3.3.1 General properties

Let $\mathcal{P}_{PRESB} = \begin{bmatrix} A + B_1 + B_2 & B_2 \\ B_1 & -A \end{bmatrix}$ be a preconditioner to $\mathcal{A}$ in (1), to be used in a Krylov type of iteration method or in a matrix splitting type of iteration method.

Following [10, 2], we show first that the solution of systems

$$\mathcal{P}_{PRESB} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix} = \begin{bmatrix} \boldsymbol{f} \\ \boldsymbol{g} \end{bmatrix}, \tag{13}$$

can be computed efficiently.

Changing the sign of the second equation and adding the first, the system can be written in the equivalent form,

$$\begin{cases} (A + B_1 + B_2)\boldsymbol{x} + B_2\boldsymbol{y} = \boldsymbol{f} \\ (A + B_2)\boldsymbol{x} + (A + B_2)\boldsymbol{y} = \boldsymbol{f} - \boldsymbol{g} \end{cases}$$

i.e.,

$$\begin{bmatrix} A + B_1 & B_2 \\ 0 & A + B_2 \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{z} \end{bmatrix} = \begin{bmatrix} \boldsymbol{f} \\ \boldsymbol{f} - \boldsymbol{g} \end{bmatrix},$$

where $\boldsymbol{z} = \boldsymbol{x} + \boldsymbol{y}$. Hence, $\boldsymbol{z} = (A + B_2)^{-1}(\boldsymbol{f} - \boldsymbol{g})$ and $(A + B_1)\boldsymbol{x} = \boldsymbol{f} - B_2\boldsymbol{z}$. Therefore the algorithm to compute the solution of (13) can be written as

- Solve $(A + B_2)\boldsymbol{z} = \boldsymbol{f} - \boldsymbol{g}$, compute $\widetilde{\boldsymbol{f}} = \boldsymbol{f} - B_2\boldsymbol{z}$
- Solve $(A + B_1)\boldsymbol{x} = \widetilde{\boldsymbol{f}}$, compute $\boldsymbol{y} = \boldsymbol{z} - \boldsymbol{x}$

Hence, besides some vector additions, the algorithm involves a solution of a linear systems with $A + B_2$, followed by a matrix vector multiplication with B_2 and a solution with the matrix $A + B_1$. In practice, the solution of the *two* linear systems contributes to the major cost of computing an action of $\mathcal{P}^{-1}_{PRESB}$. In our applications they correspond to elliptic operators.

It is seen that the above procedure is equivalent to the use of the following form of the inverse of $\mathcal{P}_{PRESB}$,

$$\mathcal{P}^{-1}_{PRESB} = \begin{bmatrix} I & 0 \\ -I & I \end{bmatrix} \begin{bmatrix} (A + B_1)^{-1} & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} I & -B_2 \\ 0 & I \end{bmatrix} \begin{bmatrix} I & 0 \\ 0 & -(A + B_2)^{-1} \end{bmatrix} \begin{bmatrix} I & 0 \\ -I & I \end{bmatrix}. \tag{14}$$

This form follows also directly from the factorization

$$\mathcal{P}_{PRESB} = \begin{bmatrix} I & 0 \\ I & -(A + B_2) \end{bmatrix} \begin{bmatrix} I & B_2 \\ 0 & I \end{bmatrix} \begin{bmatrix} A + B_1 & 0 \\ I & I \end{bmatrix}$$

and has been already used in [10, 2].

3.3.2 Spectral analysis

For the analysis of the rate of convergence of the preconditioned iteration method we need information about the eigenvalue distribution of the preconditioned matrix $\mathcal{P}^{-1}_{PRESB}\mathcal{A}$. This is next derived, considering various assumptions of the matrices involved. We first reduce the corresponding generalized eigenvalue problem to a more convenient form.

For the generalized eigenvalue problem,

$$\lambda\, \mathcal{P}_{PRESB} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix} = \mathcal{A} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix}, \quad \|\boldsymbol{x}\| + \|\boldsymbol{y}\| \neq 0,$$

it holds

$$(1-\lambda)\begin{bmatrix} A+B_1+B_2 & B_2 \\ -B_1 & A \end{bmatrix}\begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix} = \begin{bmatrix} (B_1+B_2)\boldsymbol{x} \\ 0 \end{bmatrix}. \tag{15}$$

It follows that $\lambda = 1$ for eigenvectors $(\boldsymbol{x}, \boldsymbol{y})$ such that $\{\boldsymbol{x} \in \mathcal{N}(B_1+B_2), \boldsymbol{y} \in \mathbb{C}^n \text{ arbitrary}\}$. Hence, the dimension of the eigenvector space corresponding to $\lambda = 1$ is $n + n_0$, where n_0 is the dimension of the nontrivial nullspace of $B_1 + B_2$.

An addition of the equations in (15) shows that

$$(1-\lambda)(A+B_2)(\boldsymbol{x}+\boldsymbol{y}) = (B_1+B_2)\boldsymbol{x}$$

and, hence, from the first equation in (15), it follows

$$(1-\lambda)(A+B_1)\boldsymbol{x} = (I - B_2(A+B_2)^{-1})(B_1+B_2)\boldsymbol{x}. \tag{16}$$

Further, it can be rewritten as

$$(1-\lambda)(A+B_1)\boldsymbol{x} = A(A+B_2)^{-1}(B_1+B_2)\boldsymbol{x}. \tag{17}$$

3.3.3 Spectrum for a symmetric and nonsingular matrix B

Proposition 3.4. *Assume that $B_1 = B_2 = B$ and that Assumption IIa holds. Then the eigenvalues λ of $\mathcal{P}^{-1}_{PRESB}\mathcal{A}$ are bounded by*

$$1 \geq \lambda \geq \frac{1}{2}\left(1 + \min_{\mu} |1-2\mu|^2\right),$$

where μ is an eigenvalue of the generalized eigenvalue problem $\mu(A+B)\boldsymbol{z} = B\boldsymbol{z}$, $\|\boldsymbol{z}\| \neq 0$, i.e. $0 \leq \mu \leq 1$.

Proof. With $B_1 = B_2 = B$, it follows from (16) that

$$(1-\lambda)\boldsymbol{x} = 2\left(I - (A+B)^{-1}B\right)(A+B)^{-1}B\boldsymbol{x}.$$

Hence,

$$\begin{aligned} 1-\lambda &= 2(1-\mu)\mu = 2\left(\frac{1}{2} + \left(\frac{1}{2}-\mu\right)\right)\left(\frac{1}{2} - \left(\frac{1}{2}-\mu\right)\right) = \\ &= \frac{1}{2}\left(1-(1-2\mu)^2\right) \leq \frac{1}{2}\left(1 - \min_{\mu}|1-2\mu|^2\right), \end{aligned} \tag{18}$$

where $0 \leq \mu \leq 1$, so

$$1 \geq \lambda \geq \frac{1}{2}\left(1 + \min_{\mu}(1-2\mu)^2\right).$$

□

We extend now this proposition to the case of complex eigenvalues μ but still under the condition that $B_1 = B_2 = B$.

Proposition 3.5. *Let A be spsd and $B_1 = B_2 = B$ and let the eigenvalues of $\mu(A+B)\boldsymbol{z} = B\boldsymbol{z}$, $\|\boldsymbol{z}\| \neq 0$ satisfy $1 - 2\mu = \xi + i\eta$ where $0 < \xi < 1$ and $|\eta| < (2/(\sqrt{2}+1))^{1/2}$. Then*

$$|1-\lambda| = \frac{1}{2}\sqrt{(1-\xi^2)^2 + \eta^4 + 2\eta^2 + 2\xi^2\eta^2} < 1.$$

Proof. It follows from (18) that

$$1 - \lambda = \frac{1}{2}(1 + (1-2\mu))(1 - (1-2\mu)) = \frac{1}{2}(1+\xi+i\eta)(1-\xi-i\eta) = \frac{1}{2}(1 - \xi^2 + \eta^2 - 2i\xi\eta)$$

so

$$\begin{aligned} |1-\lambda|^2 &= \frac{1}{4}\left[(1-\xi^2+\eta^2)^2 + 4\xi^2\eta^2\right] = \frac{1}{4}\left((1-\xi^2)^2 + \eta^4 + 2\eta^2 + 2\xi^2\eta^2\right) \\ &= \frac{1}{4}(1 + \xi^4 - 2(\xi^2 + \eta^2 - \xi^2\eta^2) + \eta^4 + 4\eta^2) \\ &= \frac{1}{4}(1 + \xi^4 - 2(\xi^2(1 - \frac{1}{2}\eta^2) + \eta^2(1 - \frac{1}{2}\xi^2)) + \eta^4 + 4\eta^2) < 1, \end{aligned}$$

since $0 < \xi < 1$ and $\eta^2 < 2(\sqrt{2}-1)$, i.e., $\eta^4 + 4\eta^2 < 4$. □

3.3.4 Spectrum for complex matrices B_1, B_2

Consider now the matrix in (1) where $B_2 = B^*$, $B_1 = B$, i.e. it can be complex valued.. This statement has already been shown in [2] but with a slightly different proof.

Proposition 3.6. *Let Assumption IIb hold and assume that $Re(\mu) \geq 0$ where $\mu A\boldsymbol{z} = B\boldsymbol{z}$, $\|\boldsymbol{z}\| \neq 0$. Then the eigenvalues of $\mathcal{P}_{PRESB}^{-1}\mathcal{A}$ satisfy*

$$1 \geq \lambda \geq \frac{1}{1+\alpha}, \quad \textit{where } \alpha = \max_{\mu}\{Re(\mu)/|\mu|\}.$$

Proof. It follows from (16) that

$$(1-\lambda)(A+B_1)\boldsymbol{x} = A(A+B_2)^{-1}(B_1+B_2)\boldsymbol{x}.$$

Let $\widetilde{B}_i = A^{-1/2}B_iA^{-1/2}$, $i = 1, 2$ and $\widetilde{\boldsymbol{x}} = A^{1/2}\boldsymbol{x}$. Then

$$(1-\lambda)(I+\widetilde{B}_2)(I+\widetilde{B}_1)\widetilde{\boldsymbol{x}} = (\widetilde{B}_1+\widetilde{B}_2)\widetilde{\boldsymbol{x}}$$

so

$$(1-\lambda)\widetilde{\boldsymbol{x}}^*(I + \widetilde{B}_2\widetilde{B}_1 + \widetilde{B}_2 + \widetilde{B}_1)\widetilde{\boldsymbol{x}} = \widetilde{\boldsymbol{x}}^*(\widetilde{B}_1+\widetilde{B}_2)\widetilde{\boldsymbol{x}}, \tag{19}$$

where $\widetilde{x}^*$ denotes the complex conjugate vector.

It suffices to consider $\lambda \neq 1$, i.e. $(\widetilde{B}_1+\widetilde{B}_2)\boldsymbol{x} \neq \boldsymbol{0}$. From (19) follows

$$(1-\lambda)\widetilde{\boldsymbol{x}}^*\left((I-\widetilde{B}_2)(I-\widetilde{B}_1) + 2(\widetilde{B}_1+\widetilde{B}_2)\right)\widetilde{\boldsymbol{x}} = \widetilde{\boldsymbol{x}}^*(\widetilde{B}_1+\widetilde{B}_2)\widetilde{\boldsymbol{x}},$$

where $\widetilde{B}_1 = \widetilde{B}$, $\widetilde{B}_2 = \widetilde{B}^*$ and $\widetilde{B} = A^{-1/2}BA^{-1/2}$. Further, since $\widetilde{B}\widetilde{\boldsymbol{z}} = \mu\widetilde{\boldsymbol{z}}$, $\widetilde{\boldsymbol{z}} = A^{1/2}\boldsymbol{z}$, where $|\mu| \neq 0$, it follows that

$$(1-\lambda)\left((1-\overline{\mu})(1-\mu) + 4Re(\mu)\right) = 2Re(\mu)$$

or

$$(1-\lambda)\left(1+|\mu|^2+2Re(\mu)\right) = 2Re(\mu),$$

i.e.

$$1-\lambda = \frac{2Re(\mu)}{1+|\mu|^2+2Re(\mu)} \leq \frac{2\alpha|\mu|}{1+|\mu|^2+2\alpha|\mu|} = \frac{\alpha}{\frac{1}{2}\left(\frac{1}{|\mu|}+|\mu|\right)+\alpha} \leq \frac{\alpha}{1+\alpha},$$

that is, $\lambda \geq \frac{1}{1+\alpha}$. Further, since by assumption, $\widetilde{B}+\widetilde{B}^*$ is spsd, it follows from (19) that $\lambda \leq 1$. □

The above shows that the relative size, $Re(\mu)/|\mu|$ of the real part of the spectrum of $\widetilde{B} = A^{-1/2}BA^{-1/2}$ determines the lower eigenvalue bound of $\mathcal{P}_{PRESB}^{-1}\mathcal{A}$ and, hence, the rate of convergence of the preconditioned iterative solution method. For a small such relative part the convergence of the iterative solution method will be exceptionally rapid. As we show below, such small parts can occur for time-harmonic problems with a large value of the frequency.

A proof of rate of convergence under Assumption IIa follows below.

Proposition 3.7. *Let Assumption IIa hold. Then* $1 \geq \lambda(\mathcal{P}_{PRESB}^{-1}\mathcal{A}) \geq \frac{1}{2}$.

Proof. The generalized eigenvalue problem takes here the form

$$\lambda \begin{bmatrix} A+B+B^* & B^* \\ -B & A \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix} = \begin{bmatrix} A & B^* \\ -B & A \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix}, \quad \|\boldsymbol{x}\| + \|\boldsymbol{y}\| \neq 0.$$

Hence

$$(1-\lambda) \begin{bmatrix} A+B+B^* & B^* \\ -B & A \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix} = \begin{bmatrix} (B+B^*)\boldsymbol{x} \\ 0 \end{bmatrix},$$

and it follows from (17) that

$$(1-\lambda)\boldsymbol{x} = (A+B)^{-1}A(A+B^*)^{-1}(B+B^*)\boldsymbol{x}.$$

Clearly, any vector $\boldsymbol{x} \in \mathcal{N}(B+B^*)$ corresponds to an eigenvalue $\lambda = 1$. Note that $(1-\lambda)(\boldsymbol{x}+\boldsymbol{y}) = (A+B^*)^{-1}(B+B^*)\boldsymbol{x}$. Hence, if $A(A+B^*)^{-1}(B+B^*)\boldsymbol{x} = \boldsymbol{0}$ for some $\boldsymbol{x} \neq \boldsymbol{0}$ and $\lambda \neq 1$, then $\boldsymbol{0} = A(\boldsymbol{x}+\boldsymbol{y}) = (A+B)\boldsymbol{x}$, which implies $\boldsymbol{x} = \boldsymbol{0}$. Hence, $\lambda = 1$ also. To estimate the other eigenvalues, $\lambda \neq 1$, we can consider subspaces orthogonal to this space, for which $\lambda = 1$. We denote the corresponding inverse of A as a generalized inverse, $A^\dagger$. It holds then

$$(1-\lambda)\boldsymbol{x} = [(A+B^*)A^\dagger(A+B)]^{-1}(B+B^*)\boldsymbol{x}$$

or

$$(1-\lambda)\boldsymbol{x} = [A + B^* A^\dagger B + B^* + B]^{-1}(B + B^*)\boldsymbol{x}$$

that is,

$$\begin{aligned}(1-\lambda)\widetilde{\boldsymbol{x}} &= (I + \widetilde{B}^*\widetilde{B} + \widetilde{B}^* + \widetilde{B})^{-1}(\widetilde{B} + \widetilde{B}^*)\widetilde{\boldsymbol{x}} = \\ &= \left((I-\widetilde{B}^*)(I-\widetilde{B}) + 2(\widetilde{B}^* + \widetilde{B})\right)^{-1}(\widetilde{B}^* + \widetilde{B})\widetilde{\boldsymbol{x}},\end{aligned}$$

where $\widetilde{B} = A^{\dagger 1/2} B A^{\dagger 1/2}$ and $\widetilde{\boldsymbol{x}} = \left(A^\dagger\right)^{1/2}\boldsymbol{x}$. It follows that $0 \le 1-\lambda \le \frac{1}{2}$, i.e. $\lambda \ge \frac{1}{2}$. Hence, $1 \ge \lambda \ge \frac{1}{2}$. □

3.4 An alternating direction type of splitting (ADISP)

We consider now a special type of alternating direction iterative solution method for two-by-two block matrices. Thereby we use $A + B$ as a correction matrix in both iteration steps, i.e., with the method parameter $\alpha = 1$.

Hence, given $\begin{bmatrix}\boldsymbol{x}^{(0)}\\ \boldsymbol{y}^{(0)}\end{bmatrix}$, for $k = 0, 1, \cdots$ until convergence, solve

$$\begin{aligned}\begin{bmatrix}A+B & 0\\ 0 & A+B\end{bmatrix}\begin{bmatrix}\boldsymbol{x}^{(k+1/2)}\\ \boldsymbol{y}^{(k+1/2)}\end{bmatrix} &= \begin{bmatrix}B & B\\ -B & B\end{bmatrix}\begin{bmatrix}\boldsymbol{x}^{(k)}\\ \boldsymbol{y}^{(k)}\end{bmatrix} + \begin{bmatrix}\boldsymbol{f}\\ \boldsymbol{g}\end{bmatrix},\\ \begin{bmatrix}A+B & 0\\ 0 & A+B\end{bmatrix}\begin{bmatrix}\boldsymbol{x}^{(k+1)}\\ \boldsymbol{y}^{(k+1)}\end{bmatrix} &= \begin{bmatrix}A & -A\\ A & A\end{bmatrix}\begin{bmatrix}\boldsymbol{x}^{(k+1/2)}\\ \boldsymbol{y}^{(k+1/2)}\end{bmatrix} + \begin{bmatrix}\boldsymbol{g}\\ -\boldsymbol{f}\end{bmatrix}.\end{aligned} \tag{20}$$

Proposition 3.8. *Let A and B be spsd and $A+B$ be spd. Then the iteration method (20) converges with a convergence factor bounded by*

$$\frac{1}{2}\left(1 - \max_{0<\mu<1}(1-2\mu)^2\right) \le \frac{1}{2},$$

where μ is an eigenvalue of $\mu(A+B)\boldsymbol{z} = B\boldsymbol{z}$, $\|\boldsymbol{z}\| \neq 0$.

Proof. The iteration matrix, corresponding to (20), is

$$\begin{bmatrix}(A+B)^{-1}A & -(A+B)^{-1}A\\ (A+B)^{-1}A & (A+B)^{-1}A\end{bmatrix}\begin{bmatrix}(A+B)^{-1}B & (A+B)^{-1}B\\ -(A+B)^{-1}B & (A+B)^{-1}B\end{bmatrix} =$$

$$= \begin{bmatrix}2(A+B)^{-1}A(A+B)^{-1}B & 0\\ 0 & 2(A+B)^{-1}A(A+B)^{-1}B\end{bmatrix}.$$

Hence, the rate of convergence is determined by the spectral radius

$$2\varrho\left((A+B)^{-1}A(A+B)^{-1}B\right).$$

It holds

$$(A+B)^{-1}A(A+B)^{-1}B = \left(I - (A+B)^{-1}B\right)(A+B)^{-1}B.$$

The result follows now as in Proposition 3.4. □

Note that each step of the iteration method (20) requires four solutions with the matrix $A+B$. However, one can perform each pair of solutions in parallel or use a version of the method for multiple right-hand sides, which enables a further extent of parallelism.

As shown in Section 3.1, the use of a block-diagonal preconditioner, where only two systems with $A+B$ have to be solved during each iteration step, leads to either an indefinite spectrum or to a complex-valued spectrum with no uniformly valued bound that can guarantee a fast rate of convergence.

3.5 A combination of PRESB and ADISP

We show now that the combination of PRESB with the alternating direction type iterative splitting method can give a very efficient iterative solution method.

Consider the linear system

$$\begin{bmatrix} A & -B \\ B & A \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix} = \begin{bmatrix} \boldsymbol{f} \\ \boldsymbol{g} \end{bmatrix},$$

and its alternative form

$$\begin{bmatrix} B & A \\ -A & B \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix} = \begin{bmatrix} \boldsymbol{f} \\ -\boldsymbol{g} \end{bmatrix},$$

We assume that A and B are spsd and $\mathcal{N}(A) \cap \mathcal{N}(B) = \emptyset$ but for the derivation of the rate of convergence we assume that A and B are nonsingular. However, as already remarked, we can perturb the matrices with a small fraction of the identity matrix and at the end of the proof simply let these perturbations go to zero. We use the matrix splittings

$$\begin{bmatrix} A+2B & -B \\ B & A \end{bmatrix} - \begin{bmatrix} 2B & 0 \\ 0 & 0 \end{bmatrix} \text{ and } \begin{bmatrix} B+2A & A \\ -A & B \end{bmatrix} - \begin{bmatrix} 2A & 0 \\ 0 & 0 \end{bmatrix}.$$

Depending on which is dominating, $\lambda(A^{-1}B)$ or $\lambda(B^{-1}A)$, we have two versions to choose among. If $\lambda(B^{-1}A)$ is dominating, we choose the following method, but if $\lambda(A^{-1}B)$ we choose the formulation where A and B are interchanged.

The iterative method takes then the form

$$\begin{aligned} \begin{bmatrix} A+2B & -B \\ B & A \end{bmatrix} \begin{bmatrix} \boldsymbol{x}^{(k+1/2)} \\ \boldsymbol{y}^{(k+1/2)} \end{bmatrix} &= \begin{bmatrix} 2B & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{x}^{(k)} \\ \boldsymbol{y}^{(k)} \end{bmatrix} + \begin{bmatrix} \boldsymbol{f} \\ \boldsymbol{g} \end{bmatrix}, \\ \begin{bmatrix} B & A \\ -A & B+2A \end{bmatrix} \begin{bmatrix} \boldsymbol{x}^{(k+1)} \\ \boldsymbol{y}^{(k+1)} \end{bmatrix} &= \begin{bmatrix} 0 & 0 \\ 0 & 2A \end{bmatrix} \begin{bmatrix} \boldsymbol{x}^{(k+1/2)} \\ \boldsymbol{y}^{(k+1/2)} \end{bmatrix} + \begin{bmatrix} \boldsymbol{f} \\ -\boldsymbol{g} \end{bmatrix}, \ k = 1, 2, \cdots, \end{aligned} \tag{21}$$

where $\begin{bmatrix} \boldsymbol{x}^{(0)} \\ \boldsymbol{y}^{(0)} \end{bmatrix}$ is an initial approximation.

Note that, except for the sign of the off-diagonal blocks, the matrices have a fully symmetric expression with respect to A and B. Therefore, it suffices to do the derivation for just one of the matrix pairs.

The matrix $\begin{bmatrix} A+2B & -B \\ B & A \end{bmatrix}$ can be factorized as

$$\begin{bmatrix} A+2B & -B \\ B & A \end{bmatrix} = \begin{bmatrix} I & -BA^{-1} \\ 0 & I \end{bmatrix} \begin{bmatrix} S_B & 0 \\ 0 & A \end{bmatrix} \begin{bmatrix} I & 0 \\ A^{-1}B & I \end{bmatrix},$$

where $S_B = A + 2B + BA^{-1}B$ is the Schur complement matrix. Hence,

$$\begin{bmatrix} A+2B & -B \\ B & A \end{bmatrix}^{-1} \begin{bmatrix} 2B & 0 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} I & 0 \\ -A^{-1}B & I \end{bmatrix} \begin{bmatrix} S_B^{-1} & 0 \\ 0 & A^{-1} \end{bmatrix} \begin{bmatrix} I & BA^{-1} \\ 0 & I \end{bmatrix} \begin{bmatrix} 2B & 0 \\ 0 & 0 \end{bmatrix} =$$
$$\begin{bmatrix} I & 0 \\ -A^{-1}B & I \end{bmatrix} \begin{bmatrix} 2S_B^{-1}B & 0 \\ 0 & 0 \end{bmatrix} = 2 \begin{bmatrix} S_B^{-1}B & 0 \\ -A^{-1}BS_B^{-1}B & 0 \end{bmatrix}.$$

Similarly,

$$\begin{bmatrix} B & A \\ -A & B+2A \end{bmatrix}^{-1} \begin{bmatrix} 0 & 0 \\ 0 & 2A \end{bmatrix} = \begin{bmatrix} I & -B^{-1}A \\ 0 & I \end{bmatrix} \begin{bmatrix} B^{-1} & 0 \\ 0 & S_A^{-1} \end{bmatrix} \begin{bmatrix} I & 0 \\ AB^{-1} & I \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & 2A \end{bmatrix} =$$
$$\begin{bmatrix} I & -B^{-1}A \\ 0 & I \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & 2S_A^{-1}A \end{bmatrix} = 2 \begin{bmatrix} 0 & -B^{-1}AS_A^{-1}A \\ 0 & S_A^{-1}A \end{bmatrix},$$

with $S_A = B + 2A + AB^{-1}A$. Therefore, in the iterative procedure

$$\begin{bmatrix} \boldsymbol{x}^{(k+1)} \\ \boldsymbol{y}^{(k+1)} \end{bmatrix} = G \begin{bmatrix} \boldsymbol{x}^{(k)} \\ \boldsymbol{y}^{(k)} \end{bmatrix} + \text{r.h.s terms},$$

the iteration matrix is the product

$$G = \begin{bmatrix} B & A \\ -A & B+2A \end{bmatrix}^{-1} \begin{bmatrix} 0 & 0 \\ 0 & 2A \end{bmatrix} \begin{bmatrix} A+2B & -B \\ B & A \end{bmatrix}^{-1} \begin{bmatrix} 2B & 0 \\ 0 & 0 \end{bmatrix}$$

which equals

$$G = 4 \begin{bmatrix} 0 & -B^{-1}AS_A^{-1}A \\ 0 & S_A^{-1}A \end{bmatrix} \begin{bmatrix} S_B^{-1}B & 0 \\ -A^{-1}BS_B^{-1}B & 0 \end{bmatrix} = 4 \begin{bmatrix} B^{-1}AS_A^{-1}BS_B^{-1}B & 0 \\ -S_A^{-1}BS_B^{-1}B & 0 \end{bmatrix}.$$

Denote $X = A^{-1}B$ and note that

$$S_B^{-1}B = A + 2B + BA^{-1}B = (I + 2X + X^2)^{-1}X = (I + X)^{-2}X \quad \text{and}$$
$$S_A^{-1}A = B + 2A + AB^{-1}A = (I + 2X^{-1} + X^{-2})^{-1}X^{-1} = (I + X)^{-2}X.$$

Hence, $S_A^{-1}A = S_B^{-1}B = ((I+X)^{-2}X)^2$. Further, $B^{-1}AS_A^{-1}BS_B^{-1}B = B^{-1}AS_A^{-1}AA^{-1}BS_B^{-1}B = S_A^{-1}AS_B^{-1}B$, so

$$G = \begin{bmatrix} Y^2 & 0 \\ -Y^2 & 0 \end{bmatrix} \text{ with } Y = 2(I + X)^{-2}X.$$

Hence,

$$G^TG = \begin{bmatrix} Y^2 & -Y^2 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} Y^2 & 0 \\ -Y^2 & 0 \end{bmatrix} = \begin{bmatrix} 2Y^4 & 0 \\ 0 & 0 \end{bmatrix}.$$

Clearly,

$$0 \leq \rho(Y) = \lambda_{max}(Y) \leq \frac{2\lambda(X)}{1 + 2\lambda_{max}(X) + \lambda_{max}(X)^2} \leq \frac{1}{2},$$

so

$$\rho(G) \leq \sqrt{2}\lambda_{max}(Y)^2 \leq \begin{cases} \sqrt{2}\left(\dfrac{2\lambda(X)}{1 + 2\lambda_{max}(X) + \lambda_{max}(X)^2}\right)^2 & \text{if } \lambda_{max}(X) < 1, \\ \dfrac{\sqrt{2}}{4} & \text{if } \lambda_{max}(X) \geq 1. \end{cases}$$

The derivations show that the combined PRESB ∪ADISP method converges very rapidly and even extremely rapidly when $\lambda_{max}(X) \ll 1$. Since, as follows from Proposition 3.7, the condition number of the standard PRESB method is bounded by 2, when used as a preconditioner in a Krylov subspace iteration method, the convergence factor is $\frac{\sqrt{2}-1}{\sqrt{2}+1} = \frac{1}{(\sqrt{2}+1)^2}$, thus about a factor 1.8 smaller. However, Krylov subspace iterations require global scalar products that, for very large problems on the current high performance computer platforms, can deteriorate the performance of the method. The above PRESB ∪ADISP method does not require any global inner products, at least not for the outer iteration. But note that during each iteration step of the combined method *four* solutions of systems with the matrix $A + B$ need to be solved.

For completeness we mention that PRESB ∪ADISP can itself be used as a preconditioner for a Krylov subspace method. Nevertheless, as seen from (21), this needs some initial computation of the first residual, and the eigenvalues of the preconditioned matrix $I - G$ are complex-valued, which slows down the iterations.

Remark 3.3. *Other well-known preconditioning strategies for general two-by-two block matrices, such as block-triangular preconditioners, are also applicable, cf., e.g. [3, 37, 39, 40, 42, 44]. We do not discuss those here any further. Although robust with respect to the involved parameters, some of these have been shown to be computationally less efficient than PRESB on a benchmark suite in [8, 9, 10].*

4 A survey of some problems, leading to two-by-two block matrices with square blocks

We survey now various applications where two-by-two block matrices of special form with square blocks appear.

4.1 Complex valued systems

Let

$$(A + iB)(\boldsymbol{x} + i\boldsymbol{y}) = \boldsymbol{a} + i\boldsymbol{b}, \tag{22}$$

where $A, B, \boldsymbol{x}, \boldsymbol{y}, \boldsymbol{a}, \boldsymbol{b}$ are real-valued and A, B are square. We assume that $\mathcal{N}(A) \cap \mathcal{N}(B) = \{0\}$, so (22) has a unique solution. To avoid complex-valued arithmetic computations and to form a matrix structure for which efficient preconditioners can be constructed, we rewrite (22) in a real valued form,

$$\begin{bmatrix} A & -B \\ B & A \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix} = \begin{bmatrix} \boldsymbol{a} \\ \boldsymbol{b} \end{bmatrix},$$

where a two-by-two block system of the form (2) arises.

4.2 Optimal control problems with a PDE-constrained state equation

In the recent decades numerical solution techniques for these problems have been actively developed. As it is not feasible to refer to all, we list some works, ordered in the year of publishing (with examples from 2000 to 2018), that are related to the numerical solution of the arising algebraic systems in general or to the problems, considered here, [31, 32, 33, 34, 35, 36, 37, 14, 38, 1, 3, 4, 13, 39, 40, 6, 41, 42, 43, 8, 9, 10, 26, 44, 2, 11, 12, 30].

As shown, for instance, in [1, 6, 11, 13] and references therein, there are several types of optimal control problems for PDEs that, after a proper discretization, lead to a linear system with a matrix of the form (2) or (3).

There are many applications of optimal control problems with different types of stationary and time-dependent partial differential equations and distributed or only locally defined optimal control functions and observation regions. Further, there can be additional box-constraints of the solution and control functions. Most relevant problems for this study are those with distributed control. Due to their importance, in this section we survey some of these problems.

4.2.1 Poisson and convection-diffusion equation optimal control problem

As an introductory problem, consider minimizing the cost functional with a Tikhonov type of regularization term

$$\min_{y,u} J(y, u) = \frac{1}{2}\|y - y_d\|^2_{L_2(\Omega)} + \frac{1}{2}\beta\|u\|^2_{L_2(\Omega)},$$

where the control function u is distributed on the whole, given bounded Lipschitz domain, $\Omega \subset \mathbb{R}^d$, $d = 1, 2$ or 3, y_d is the target solution and y is a solution of a differential equation,

$$\mathcal{L}(y) = u \text{ in } \Omega$$

with some given boundary conditions. Further, $\beta > 0$ is a small regularization parameter, chosen to obtain a solution close to y_d but not too small as this leads to an ill-conditioned system. As an example, let

$$\begin{aligned} \mathcal{L}(y) = -\varepsilon\Delta y + (\boldsymbol{w}\cdot\nabla)y = u \quad &\text{in} \quad \Omega \\ u = g \quad &\text{on} \quad \partial\Omega, \end{aligned} \tag{23}$$

where the vectorial function $\mathbf{w}$ is divergence free or $\nabla \cdot \boldsymbol{w} \leq 0$ and $y \in H^1(\Omega)$. The scalar ε (e.g. viscosity) is positive but often $\varepsilon \ll 1$, which adds to the ill-conditioning of the problem.

The problem can be formulated with a Lagrangian multiplier λ to form the constrained Lagrangian, saddle point optimization problem,

$$\inf_{y,u}\sup_{\lambda}\mathcal{L}(y,u,\lambda) = J(y,u) + \int_\Omega \lambda(\mathcal{L}y - u)d\Omega.$$

Normally the problem is discretized using a standard Galerkin finite element method, i.e. the minimization takes place in some discrete FEM subspace of $H^1(\Omega)$.

The first order necessary Karush-Kuhn-Tucker (KKT) conditions, which are also sufficient for the existence of a solution, leads then to a block matrix system,

$$\begin{bmatrix} M & 0 & K^T \\ 0 & \beta M & -M \\ K & -M & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{u} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \boldsymbol{d} \\ \boldsymbol{0} \\ \boldsymbol{b} \end{bmatrix}. \tag{24}$$

Here M is the mass matrix corresponding to the chosen FE basis functions and the L_2-inner product, K is the stiffness matrix, $\boldsymbol{d}$ contains the discretized version of the target state and and $\boldsymbol{b}$ contains the boundary terms. Note that $\boldsymbol{\lambda}$ acts as an adjoint variable to $\boldsymbol{y}$, i.e. satisfies the equation, adjoint to $\mathcal{L}$.

For simplicity, we discretize the state, the control and the adjoint state variables using the same finite element basis functions. Using the relation $\boldsymbol{u} = \frac{1}{\beta}\boldsymbol{\lambda}$, the system (24) can be reduced, to get

$$\begin{bmatrix} M & K^T \\ K & -\frac{1}{\beta}M \end{bmatrix} \begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \boldsymbol{d} \\ \boldsymbol{b} \end{bmatrix}.$$

After scaling $\boldsymbol{\lambda}$, $\tilde{\boldsymbol{\lambda}} = \frac{1}{\sqrt{\beta}}\boldsymbol{\lambda}$ and multiplying the second equation with $\sqrt{\beta}$, and letting $\tilde{\boldsymbol{b}} = \sqrt{\beta}\boldsymbol{b}$ and $\widetilde{K} = \sqrt{\beta}K$, we get

$$\begin{bmatrix} M & \widetilde{K}^T \\ \widetilde{K} & -M \end{bmatrix} \begin{bmatrix} \boldsymbol{y} \\ \tilde{\boldsymbol{\lambda}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{d} \\ \tilde{\boldsymbol{b}} \end{bmatrix}, \tag{25}$$

which is the prototype of the two-by-two block matrix with square matrix blocks, we deal with.

The value of the regularization parameter (β) for the control cost influences how close the solution $\boldsymbol{y}$ becomes to the desired solution $\boldsymbol{y}_d$. The solution method used is a coupled

outer-inner iteration method. In [8] 10^{-6} is chosen as relative outer stopping criterion and the block $M + \sqrt{\beta}K$ is approximated by one V-cycle AMG iteration. It is found (see [8, 9]) that the number of outer iterations is nearly constant and never exceeds 6, for β varied between 10^{-2} to 10^{-10}. The relative precision $\|\boldsymbol{y} - \boldsymbol{y}_d\|_2/\|\boldsymbol{y}\|_2$ varies then from about $4 \cdot 10^{-1}$ to $3.6 \cdot 10^{-4}$. Hence, in the average there are not more than one iteration per decimal of relative accuracy. It is also found that the norm of the control function increased somewhat with decreasing β, from about 4.7 to $2.3 \cdot 10^2$. This is because the function to be minimized compensates somewhat for small values of β by increasing $\|\boldsymbol{u}\|$, as otherwise it would be too insensitive to the control.

As the propositions show, the number of iterations for all test problems are bounded for all values of the discretization parameter h and are in fact nearly constant. Other preconditioners, such as the ones presented in [6], [39], and [41] require more iterations and in general do not behave fully robust with respect to all parameters.

In [9] the convection-diffusion equation as a state constraint is also tested with $w = [\cos\theta, \sin\theta]$ for $\theta = \pi/4$. The block matrix system takes then the form

$$\begin{bmatrix} M & 0 & F^T \\ 0 & \beta M & -M \\ F & -M & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{u} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \boldsymbol{b} \\ \boldsymbol{0} \\ \boldsymbol{d} \end{bmatrix},$$

where M is the mass matrix and $F = \varepsilon K + N$, where K is the stiffness (Laplace operator) matrix and N comes from the convection term. The control function (or Lagrange multiplier) can be eliminated as before. The state $\boldsymbol{y}$, control $\boldsymbol{u}$ and adjoint variable $\boldsymbol{\lambda}$ are discretized using Q_1 basis functions. The value of the viscosity in the test problem is chosen as $\varepsilon = 1/500$ and $\varepsilon = 1/1500$. It is found ([9]) that the number of iterations for the larger values of β increase somewhat for the smaller values of ε and h, but for the practically useful values of β there is hardly any increase and there are very few iterations, between 3 and 9 for $\beta = 10^{-10}$ and $\beta = 10^{-6}$. This means that the PRESB preconditioner performs exceptionally well.

Optimal control problems with a nonsymmetric operator leads in general to somewhat different block matrix systems if one uses discretize-then-optimize or optimize-then discretize methods. As has been seen, the discretize-then-optimize method leads to an inconsistent but symmetric block matrix system. The inconsistency can be handled a local projection stabilization scheme, see [45].

4.2.2 Stokes type of optimal control problem

Consider now an optimal control problem with a Stokes state equation,

$$\begin{cases} -\Delta y + \nabla p &= u \quad \text{in } \Omega \\ \nabla \cdot y &= 0 \quad \text{in } \Omega \\ y &= g \quad \text{on } \partial\Omega. \end{cases}$$

Here we need two Lagrange multipliers, $\tilde{\lambda}^{(y)}$ and $\tilde{\lambda}^{(p)}$, corresponding to both the velocity y and the pressure p. The system is self-adjoint, so the discretize-then-optimize and optimize-

then-discretize yield the same block matrix system. Using the relation $\mathbf{u} = \frac{1}{\beta}\tilde{\boldsymbol{\lambda}}^{(y)}$, the system is reduced to a four-by-four block matrix which, after the scaling $\boldsymbol{\lambda}^{(y)} = -\tilde{\boldsymbol{\lambda}}^{(y)}/\sqrt{\beta}$, $\boldsymbol{\lambda}^{(p)} = -\tilde{\boldsymbol{\lambda}}^{(p)}/\sqrt{\beta}$ and correspondingly for the right hand side vector, takes the form

$$\mathcal{A}^R \begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{p} \\ \boldsymbol{\lambda}^{(y)} \\ \boldsymbol{\lambda}^{(p)} \end{bmatrix} = \begin{bmatrix} M & 0 & -F^T & -B^T \\ 0 & 0 & -B & 0 \\ F & B^T & M & 0 \\ B & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{p} \\ \boldsymbol{\lambda}^{(y)} \\ \boldsymbol{\lambda}^{(p)} \end{bmatrix} = \begin{bmatrix} \boldsymbol{b} \\ \boldsymbol{0} \\ \boldsymbol{f} \\ \boldsymbol{g} \end{bmatrix}$$

with a skew-symmetric block matrix. By permuting rows 2 and 4 and columns 2 and 4, it is seen that the block diagonal and the Schur complement matrix are nonsingular, which implies that $\mathcal{A}^R$ is also nonsingular. As shown in Theorem 1 in [9], in this case our block matrix preconditioner leads to eigenvalue bounds $1/3 \leq \lambda \leq 1$, but if F is symmetric, the lower bound equals $1/2$, as also follows from Proposition 3.6.

For comparison, in this test problem an inexact Uzawa method is used for the inner iterations. The number of Uzawa iterations is fixed to 6. This enables use of the MINRES method, (see [46]). Again, the small number of outer iterations is very robust and varies between 6 (for $\beta = 10^{-2}$) to 4, for $\beta \leq 10^{-6}$. As before, the method is fully parameter independent and performs best among methods compared. It performed equally well for a velocity tracking problem as for a lid-driven cavity problem. However, in general it is more efficient to use an inner iteration method with stopping criteria, i.e. a variable preconditioner as this avoids the problem of choosing a proper number of Uzawa iterations.

4.2.3 Time-harmonic parabolic optimal control problems

In many problems, the control function and also the target function, are time-harmonic, leading to a special complex valued form of the system matrix in (25). This can be rewritten in real valued matrix form, leading to a four-by-four block matrix, with two-by-two sub-block matrices. Such time-harmonic problems occur for instance in electromagnetic problems with alternating currents, see Section 4.2.5.

In some problems the control and the discrete state functions are time-harmonic. As an example, consider first the problem,

$$\text{minimize } J(y,u), \qquad y, u \in H^1(\Omega)$$

subject to the parabolic problem

$$\begin{aligned} \tfrac{\partial y(x,t)}{\partial t} - \Delta y(x,t) &= u(x,t) && \text{in } \Omega \times (0,T) \\ y(x,t) &= 0 && x \in \Gamma \times (0,T), \Gamma = \partial\Omega \\ y(x,0) &= y(x,T) && \text{in } \Omega \\ u(x,0) &= u(x,T) && \text{in } \Omega, \end{aligned}$$

where

$$J(y,u) = \frac{1}{2}\int_0^T \int_\Omega |y(x,t) - y_d(x,t)|^2 dxdt + \frac{1}{2}\beta \int_0^T \int_\Omega |u(x,t)|^2 dxdt.$$

The target function is assumed to be time-harmonic, $y_d(x,t) = y_d(x)e^{i\omega t}$, with period, i.e. angular velocity, $\omega = k2\pi/T$, for some positive integer k. Then since the problem is linear, the solution and the control are also time-harmonic, $y(x,t) = y(x)e^{i\omega t}$ and $u(x,t) = u(x)e^{i\omega t}$. Therefore $y(x)$, $u(x)$ are time-independent solutions of the following optimal control problem,

$$\text{minimize } \frac{1}{2}\int_\Omega |y(x) - y_d(x)|^2 dx + \frac{1}{2}\beta \int_\Omega |u(x)|^2 dx,$$

subject to

$$\begin{cases} i\omega y(x) - \Delta y(x) &= u(x), \quad \text{in } \Omega \\ y(x) &= 0, \quad x \in \Gamma. \end{cases}$$

We assume that $y(x)$ and $y_d(x)$ are real-valued but the control $u(x)$ must be complex-valued, $u(x) = u_0(x) + iu_1(x)$. (If the solution function and the target state are also complex-valued, as has been mentioned above one can separate the real and imaginary parts, which leads to two systems of with block matrix of the following form.) If the target function and the state and control are approximated with a truncated Fourier series, $y_d(x,t) = \sum_{k=1}^N y_d^{(k)}(x)e^{i\omega_k t}$, then, since we deal with linear problems, we obtain N separate uncoupled time-harmonic systems to solve, i.e. they can be solved fully in parallel.

For nonlinear problems we can use a two- or multi-level solution method, leading to a linear, Newton type or quasi Newton (see e.g. [11] and the references therein) problem to be solved on the finest mesh. For each frequency, after the same reduction $\lambda = \beta u$ as done before, we obtain a reduced system,

$$\begin{bmatrix} M & \beta(K - i\omega M) \\ K + i\omega M & -M \end{bmatrix} \begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{u} \end{bmatrix} = \begin{bmatrix} M\boldsymbol{y}_d \\ \boldsymbol{0} \end{bmatrix}. \tag{26}$$

Here y is real valued, whereas u contains an imaginary part, $u = u_0 + iu_1$.

As shown in [10], we can reduce the system in (26) to a real valued form,

$$\mathcal{A}\begin{bmatrix} \boldsymbol{y} \\ -\boldsymbol{u}_0 \end{bmatrix} = \begin{bmatrix} M & -\tilde{\beta}K \\ K & M \end{bmatrix} \begin{bmatrix} \boldsymbol{y} \\ -\boldsymbol{u}_0 \end{bmatrix} = \frac{1}{1+\beta\omega^2} \begin{bmatrix} M\boldsymbol{y}_d \\ \boldsymbol{0} \end{bmatrix},$$

where $\tilde{\beta} = \beta/(1+\beta\omega^2)$.

Here matrix $\mathcal{A}$ can be preconditioned with the same type of method as used previously, which again leads to eigenvalue bounds $\frac{1}{2} \le \lambda \le 1$ of the preconditioned matrix. Furthermore, since $\tilde{\beta} \to 0$ when $\omega \to \infty$, it follows that the eigenvalues cluster strongly near $\lambda = 1$ for large frequencies.

Two systems with $M + \sqrt{\tilde{\beta}}K$ must be solved during each outer iteration step. These inner systems can be solved as before with variable preconditioners such as in the Generalized Conjugate Gradient method ([19]) or FGMRES. Alternatively, as already mentioned, one can use a fixed number of Uzawa steps in which case no flexible outer solver is needed. However, then one must come up with a good choice of the number of Uzawa steps, normally found by a trial and error approach. The numerical tests in [30] show that in general fewer iterations are needed in the flexible version of the outer solver.

4.2.4 Time-harmonic Stokes optimal control problem

Consider now the corresponding time-harmonic optimal control problem with a Stokes equation as state equation, i.e. with a divergence free solution,

$$\begin{aligned}
\mathcal{L}(y,p) = \tfrac{\partial}{\partial t}y(x,t) - \Delta y(x,t) + \nabla p(x,t) &= u(x,t) && \text{in } \Omega \times (0,T),\\
\nabla \cdot y(x,t) &= 0 && \text{in } \Omega \times (0,T)\\
y(x,t) &= 0 && \text{on } \Gamma \times [0,T]\\
y(x,0) &= y(x,T) && \text{in } \Omega\\
p(x,0) &= p(x,T) && \text{in } \Omega\\
u(x,0) &= u(x,T) && \text{in } \Omega.
\end{aligned}$$

Let $\gamma^{(y)} = \gamma^{(y)}(x,t)$, $\lambda^{(p)} = \lambda^{(p)}(x,t)$ be Lagrange multipliers corresponding to the velocity and pressure components. The Lagrangian functional becomes then

$$\begin{aligned}
\mathcal{L}(y,p,u,\lambda^{(y)},\lambda^{(p)}) &= \frac{1}{2}\int_0^T \int_\Omega |y(x,t) - y_d(x,t)|^2 dxdt +\\
&+ \int_0^T \int_\Omega \lambda^{(y)}((\mathcal{L}y,p) - u)dxdt +\\
&+ \int_0^T \int_\Omega \nabla \cdot y(x,t)\lambda^{(p)} dxdt + \frac{1}{2}\beta \int_0^T \int_\Omega |u(x,t)|^2 dxdt.
\end{aligned}$$

Here y_d and y are assumed to be time-harmonic, i.e.

$$y(x,t) = y(x)e^{i\omega t}, \quad y_d(x,t) = y_d(x)e^{i\omega t}, \quad \omega = k2\pi/T.$$

After a suitable discretization, the gradient conditions $\nabla\mathcal{L}(y,p,u,\lambda^{(y)},\lambda^{(p)}) = 0$, leads to five necessary conditions, where we eliminate $M\lambda^{(y)} = \beta M u$ as before and reduce the problem to four equations. After a change of signs, they take the form

$$\begin{bmatrix} M & 0 & \beta(K - i\omega M) & -\beta D^T \\ 0 & 0 & -\beta D & 0 \\ K + i\omega M & -D^T & -M & 0 \\ D & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} y \\ p \\ u \\ \tilde{\lambda}^{(p)} \end{bmatrix} = \begin{bmatrix} My_d \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

where $\tilde{\lambda}^{(p)} = \lambda^{(p)}/\beta$.

Here y, p and $\lambda^{(p)}$ are real-valued but u is complex-valued, $u = u_0 + iu_1$. As for the parabolic optimal control problem, we can formulate the corresponding real valued matrix system,

$$\begin{bmatrix} M & 0 & \tilde{\beta}K & -\beta D^T \\ 0 & 0 & -\tilde{\beta}D & 0 \\ K & -D^T & -M & 0 \\ -D & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} y \\ p \\ u_p \\ \tilde{\lambda}^{(p)} \end{bmatrix} = \begin{bmatrix} My_d \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

where $\tilde{\beta} = \beta/(1+\beta\omega^2)$ and D is the discrete divergence matrix. Here the same type of preconditioner as used before can be applied. After a scaling of matrices the preconditioner takes the form (see [10])

$$\mathcal{P}_{FS} = \begin{bmatrix} M+F+F^T & 2D^T & -F^T & -D^T \\ 2D & 0 & -D & 0 \\ F & D^T & M & 0 \\ D & 0 & 0 & 0 \end{bmatrix}.$$

Besides some matrix vector multiplications, an action of its inverse involves now the solution of one system with matrix $\begin{bmatrix} M+F & D^T \\ D & 0 \end{bmatrix}$ and one with the matrix $\begin{bmatrix} M+F^T & D^T \\ D & 0 \end{bmatrix}$, both of which are of Stokes saddle point type. As shown in [10], if M and $F+F^T$ are symmetric and positive definite, and D has full rank, then the eigenvalues of the preconditioned matrix are real and bounded by $1/3 \le \lambda \le 1$.

Further, as shown in [10], one can apply various preconditioned methods to solve the two Stokes systems. As described before, the inner system can be solved with an Uzawa method or with a flexible GMRES method, for instance using a block lower-triangular preconditioner. The numerical tests in [10] show that the latter outperforms the Uzawa method and leads to very few outer iterations, typically between 4 and 8, independently on h. For the larger values of ω, $\omega = 10^4$, the number of iteration are 5 or fewer.

Note that in this problem there are three levels of iterations, the outermost, the inner Stokes solver and the innermost preconditioned method for the basic elliptic problem. Since the iterations multiply up, it is important to have efficient preconditioners on all levels.

4.2.5 An eddy current electromagnetic optimal control problem

Electromagnetic problems are governed by Maxwell's equations. Here the Laplacian operator in the parabolic problem is replaced by a curl-curl operator and the regularized optimal control problem takes the form

$$\text{minimize}_{(y,u)} \frac{1}{2}\int_{\Omega\times(0,T)} |y-y_d|^2 dx\,dt + \frac{\beta}{2}\int_{\Omega\times(0,T)} |u|^2 dx\,dt,$$

subject to the state equation,

$$\begin{cases} \sigma\frac{\partial y}{\partial t} + \operatorname{curl}(\nu \operatorname{curl} y) + \varepsilon y = u \quad \text{in } \Omega\times(0,T) \\ y\times\mathbf{n} = 0 \quad \text{on } \Gamma\times(0,T), \qquad y = y_0 \quad \text{on } \Gamma\times\{0\}. \end{cases}$$

For details of the derivation, see [1] and [2].

For a time-harmonic problem, the initial condition is replaced by the periodicity equation, $y(0) = y(T)$, in Ω. Introducing a Lagrange multiplier λ to impose the state equation, the Lagrangian functional becomes

$$\mathcal{L}(y,u,\lambda) = J(y,u) + \int_{\Omega\times(0,T)} \left(\sigma\frac{\partial y}{\partial t} + \operatorname{curl}(\nu\operatorname{curl} y) + \varepsilon y - u\right)\lambda\, dx\,dt.$$

The first order necessary conditions $\nabla_\lambda \mathcal{L}(y,u,\lambda)=0$ gives the relation $\beta u=\lambda$ in $\Omega\times(0,T)$, which enables elimination of the control variable. As before, since the problem is linear we can use a truncated Fourier series expansion for y and u, which decouples the equations so that it suffices to consider the analysis for only one frequency.

For the finite element discretization we use the lowest order tetrahedral edge elements, originally introduced in Nèdèlec, [47]. After a reordering of the equations, this yields the following system of linear equations,

$$\begin{bmatrix} M & 0 & K & -M_\omega \\ 0 & M & M_\omega & K \\ K & M_\omega & -\beta^{-1}M & 0 \\ -M_\omega & K & 0 & -\beta^{-1}M \end{bmatrix} \begin{bmatrix} \boldsymbol{y}^c \\ \boldsymbol{y}^s \\ \boldsymbol{\lambda}^c \\ \boldsymbol{\lambda}^s \end{bmatrix} = \begin{bmatrix} \boldsymbol{y}_d^c \\ \boldsymbol{y}_d^s \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix},$$

where $\boldsymbol{y}(t)=\boldsymbol{y}^c\cos(\omega t)+\boldsymbol{y}^s\sin(\omega t)$, etc and

$$M=[M_{ij}],\quad M_{ij}=\int_\Omega u_j v_i dx,\quad (M_\omega)_{ij}=\int_\Omega \sigma\omega u_j v_i dx,\quad i,j=1,2,\cdots,n\,,$$

$$K=[K_{ij}],\quad K_{ij}=\int_\Omega \nu\ \mathrm{curl}\, u_j\ \mathrm{curl}\, v_i+\varepsilon\int_\Omega u_j v_i dx.$$

Further u_i, v_i refer to finite element basis functions in

$$H_0(\mathrm{curl})=\{v\in L^2(\Omega):\ \mathrm{curl}\ v\in L^2(\Omega), v\times\mathbf{n}=0 \text{ on } \Gamma\}$$

on edges i,j. The values on the edges become

$$(y_d^c)_i=\int_\Omega y_d^c v_i dx,\quad (y_d^s)_i=\int_\Omega y_d^s v_i dx.$$

In a similar way as has been done before, we modify the system by multiplying the last two equations with $\sqrt{\beta}$ and scale the multiplier variables to $\begin{bmatrix}\widetilde{\boldsymbol{\lambda}}^c\\ \widetilde{\boldsymbol{\lambda}}^s\end{bmatrix}=\frac{1}{\sqrt{\beta}}\begin{bmatrix}\boldsymbol{\lambda}^c\\ \boldsymbol{\lambda}^s\end{bmatrix}$. Using the same type of preconditioning as in Section 3.3, obtained by adding the off-diagonal blocks to the primary diagonal block, we get

$$\mathcal{P}_{PRESB}=\begin{bmatrix} M+2\widetilde{K} & 0 & \widetilde{K} & -\widetilde{M_\omega} \\ 0 & M+2\widetilde{K} & \widetilde{M_\omega} & \widetilde{K} \\ \widetilde{K} & \widetilde{M} & -M & 0 \\ -\widetilde{M} & \widetilde{K} & 0 & -M \end{bmatrix},$$

where $\widetilde{K}=\sqrt{\beta}K$ and $\widetilde{M_\omega}=\sqrt{\beta}M_\omega$.

To get the same form of the matrix as in Section 3.3, let

$$A=\begin{bmatrix} M & 0 \\ 0 & M \end{bmatrix},\qquad B=\begin{bmatrix} \widetilde{K} & \widetilde{M_\omega} \\ -\widetilde{M_\omega} & \widetilde{K} \end{bmatrix}.$$

Then $\mathcal{A} = \begin{bmatrix} A & B^* \\ B & -A \end{bmatrix}$ and $C = \begin{bmatrix} A + 2\widetilde{K}' & B^* \\ B & -A \end{bmatrix}$, where $\widetilde{K}' = \begin{bmatrix} \widetilde{K} & 0 \\ 0 & \widetilde{K} \end{bmatrix}$. Then $B + B^* = 2\widetilde{K}'$, which is spd and it follows from Proposition 3.6 that the eigenvalues λ of $\mathcal{P}_{PRESB}^{-1}\mathcal{A}$ satisfy $\frac{1}{1+\alpha} \leq \lambda \leq 1$, where α is the ratio, $\alpha = Re(\mu)/|\mu|$ and μ is an eigenvalue of the generalized eigenvalue problem,

$$\begin{bmatrix} \widetilde{K} & \widetilde{M}_\omega \\ -\widetilde{M}_\omega & \widetilde{K} \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix} = \mu \begin{bmatrix} M & 0 \\ 0 & M \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix} \text{ or } \begin{bmatrix} \widehat{K} & \widehat{M}_\omega \\ -\widehat{M}_\omega & \widehat{K} \end{bmatrix} \begin{bmatrix} M^{1/2}\,\boldsymbol{x} \\ M^{1/2}\,\boldsymbol{y} \end{bmatrix} = \mu \begin{bmatrix} M^{1/2}\,\boldsymbol{x} \\ M^{1/2}\,\boldsymbol{y} \end{bmatrix},$$

where $\widehat{K} = M^{-\frac{1}{2}}\widetilde{K}M^{-\frac{1}{2}}$ and $\widehat{M}_\omega = M^{-\frac{1}{2}}\widetilde{M}_\omega M^{-\frac{1}{2}}$. Hence $\alpha = \frac{\|\widehat{K}^{-1/2}\widehat{M}_\omega\widehat{K}^{-1/2}\|}{1+\|\widehat{K}^{-1/2}\widehat{M}_0\widehat{K}^{-1/2}\|}$.

The arising inner systems with the block matrix $\begin{bmatrix} M + \widetilde{K} & \widetilde{M}_\omega \\ -\widetilde{M}_\omega & M + \widetilde{K} \end{bmatrix}$ can also be solved by iteration using the same type of preconditioner as for the outer system, i.e. with $\begin{bmatrix} M + \widetilde{K} + 2\widetilde{M}_\omega & \widetilde{M}_\omega \\ -\widetilde{M}_\omega & M + \widetilde{K} \end{bmatrix}$.

Here, the corresponding eigenvalues $\tilde{\lambda}$ satisfy

$$(\tilde{\lambda} - 1) \begin{bmatrix} M + \widetilde{K} + 2\widetilde{M}_\omega & \widetilde{M}_\omega \\ -\widetilde{M}_\omega & M + \widetilde{K} \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix} = - \begin{bmatrix} 2\widetilde{M}_\omega & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix},$$

from which it follows that $\tilde{\lambda} \leq 1$ and

$$(\tilde{\lambda} - 1)\boldsymbol{x}^T(M + \widetilde{K} + 2\widetilde{M}_\omega + \widetilde{M}_\omega(M + \widetilde{K})^{-1}\widetilde{M}_\omega)\boldsymbol{x} = -2\boldsymbol{x}^T\widetilde{M}_\omega\boldsymbol{x},$$

i.e.

$$(\tilde{\lambda} - 1)\widehat{\boldsymbol{x}}^T(I + 2\widehat{\widetilde{M}}_\omega + \widehat{\widetilde{M}}_\omega^2)\widehat{\boldsymbol{x}} = -2\widehat{\boldsymbol{x}}^T\widehat{M}_\omega\widehat{\boldsymbol{x}},$$

where $\widehat{\widetilde{M}}_\omega = (M+\widetilde{K})^{-1/2}\widetilde{M}_\omega(M+\widetilde{K})^{-1/2}$ and $\widehat{\boldsymbol{x}} = (M+\widetilde{K})^{1/2}\boldsymbol{x}$. It follows that $\tilde{\lambda}-1 \geq -\frac{1}{2}$, i.e. $\tilde{\lambda} \geq \frac{1}{2}$, so $\frac{1}{2} \leq \tilde{\lambda} \leq 1$.

In practice mostly the control and observation are restricted to subdomains of Ω. Since this would need a separate section we do not consider this here, but it has been shown in [2], that our preconditioner performs as well for these problems also.

As before, numerical tests show that a lower inner relative accuracy can save in total number of inner iterations without hardly any increase of outer iterations. The numerical tests also confirm that the method is fully robust with respect to all parameters, mesh size, control function parameter and problem parameters, reluctivity and conductivity.

4.2.6 Box constrained optimal control problems

Consider now an optimal control problem where a box constraint is imposed on the solution of the state equation. For simplicity consider just the Poisson equation as state equation,

i.e.

$$\begin{cases} -\Delta y &= u \quad \text{in } \Omega, \\ y &= g \quad \text{on } \partial\Omega \end{cases}.$$

The solution y and g must be a projection onto the box defined by the constraints, $\underline{y} \leq y \leq \overline{y}$, where the lower and upper bounds are given.

To handle such constraints one must use some special regularization method, such as a Lavrentiev (mixed variable) method [33] or a Moreau-Yosida [42, 35] stabilization method. For the latter the Lagrangian functional to minimize becomes

$$\begin{aligned} J(y,u) &= \frac{1}{2}\|y-y_d\|^2_{L^2(\Omega)} + \frac{1}{2}\beta\|u\|^2_{L^2(\Omega)} + \\ &+ \frac{1}{2\varepsilon}\|\max\{0, y-\overline{y}\}\|^2_{L^2(\Omega)} + \frac{1}{2\varepsilon}\|\min\{0, y-\underline{y}\}\|^2_{L^2(\Omega)} \end{aligned}$$

subject to the given differential equation, where $\varepsilon > 0$ is a small regularization parameter. Following the 'first discretize-then optimize' approach the discrete version of the minimization problem takes the form,

$$\begin{aligned} &\text{minimize}\, \frac{1}{2}(\boldsymbol{y}-\boldsymbol{y}_d)^T M(\boldsymbol{y}-\boldsymbol{y}_d) + \frac{1}{2}\beta \boldsymbol{u}^T M \boldsymbol{u} + \\ + \quad &\frac{1}{2\varepsilon}\max\{\boldsymbol{0}, \boldsymbol{y}-\overline{\boldsymbol{y}}\}^T M \max\{\boldsymbol{0}, \boldsymbol{y}-\overline{\boldsymbol{y}}\} + \frac{1}{2\varepsilon}\min\{\boldsymbol{0}, \boldsymbol{y}-\underline{\boldsymbol{y}}\}^T M \min\{\boldsymbol{0}, \boldsymbol{y}-\underline{\boldsymbol{y}}\}, \end{aligned}$$

subject to $K\boldsymbol{y} = M\boldsymbol{u} + \boldsymbol{b}$. Here $\boldsymbol{y}, \boldsymbol{y}_d, \overline{\boldsymbol{y}}, \underline{\boldsymbol{y}}$ and $\boldsymbol{u}$ now represent vectors and $\boldsymbol{b}$ represents the boundary condition. To simplify the problem, as in [11] we assume that M is a lumped, i.e. diagonal mass matrix.

Using a simplified (semi-smooth) Newton method (see e.g. [33, 43, 36]) at each iteration step we must solve a linear system

$$\begin{bmatrix} M+\varepsilon^{-1}D_0 & 0 & -K^T \\ 0 & \beta M & M \\ -K & M & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{y}^{(k)} \\ \boldsymbol{u}^{(k)} \\ \boldsymbol{\lambda}^{(k)} \end{bmatrix} = \begin{bmatrix} \boldsymbol{c}_{\mathcal{A}} \\ \boldsymbol{0} \\ \boldsymbol{d} \end{bmatrix}, \quad k = 1, 2, \cdots \tag{27}$$

where D_0 is a diagonal matrix with entries equal to zero for indices I corresponding to inactive points, i.e. where the box constraints are not taken, or equal to the elements of M for active sets $\mathcal{A}$. Further,

$$\boldsymbol{c}_{\mathcal{A}} = M\boldsymbol{y}_d + \varepsilon^{-1}(D_+\overline{\boldsymbol{y}} + D_-\underline{\boldsymbol{y}})$$

where D_+, D_- are the parts of M corresponding to the currently active sets. Hence $D_{\mathcal{A}} = D_+ \cup D_-$. Further $\lambda^{(k)}$ represents the current value of the Lagrange multiplier to handle the state equation. As shown in [11], the system (27) can be reduced and scaled to

$$\mathcal{P}_{PRESB} \begin{bmatrix} \boldsymbol{y}^{(k)} \\ \tilde{u}^{(k)} \end{bmatrix} = \begin{bmatrix} M+\frac{1}{\varepsilon}D_0 & -\widetilde{K}^T \\ \widetilde{K} & M \end{bmatrix} \begin{bmatrix} \boldsymbol{y}^{(k)} \\ \tilde{\boldsymbol{u}}^{(k)} \end{bmatrix} = \begin{bmatrix} \boldsymbol{c}_{\mathcal{A}} \\ \tilde{\boldsymbol{d}} \end{bmatrix},$$

where $D_0 = \begin{bmatrix} 0 & 0 \\ 0 & M_A \end{bmatrix} \begin{matrix} (n_I) \\ (n_{\mathcal{A}}) \end{matrix}$, $\widetilde{K} = \sqrt{\beta}K$, $\tilde{\boldsymbol{d}} = \sqrt{\beta}\boldsymbol{f}$. Here $n_I, n_{\mathcal{A}}$ denote the current number of inactive and active points.

Letting $A = M + \alpha D$, where $\alpha > 0$ is a preconditioning parameter, as preconditioner we choose

$$\mathcal{B} = \begin{bmatrix} A + \widetilde{K} + \widetilde{K}^T & -\widetilde{K}^T \\ \widetilde{K} & A \end{bmatrix}$$

for which, as we have seen in (14), it holds

$$\mathcal{B}^{-1} = \begin{bmatrix} I & (A+\widetilde{K})^{-1} \\ 0 & (A+\widetilde{K})^{-1} \end{bmatrix} \begin{bmatrix} I & 0 \\ -\widetilde{K} & I \end{bmatrix} \begin{bmatrix} (A+\widetilde{K})^{-1} & -(A+\widetilde{K})^{-1} \\ 0 & I \end{bmatrix},$$

i.e. an action of $\mathcal{B}^{-1}$ involves a solution with the matrices

$$\begin{aligned} A + \widetilde{K}^T &= M + \alpha D + \widetilde{K}^T \\ A + \widetilde{K} &= M + \alpha D + \widetilde{K}. \end{aligned}$$

They are both of elliptic type and arising systems with them can be solved using various existing software packages.

An eigenvalue analysis of $\mathcal{B}^{-1}\mathcal{P}_{PRESB}$ (see [11]) reveals that for $\alpha \geq 1$ there are in general three clusters of eigenvalues, one near $\varepsilon/(\alpha+\varepsilon)$, one near the upper bound $(1+\varepsilon)/(\alpha+\varepsilon)$ for eigenvalues $\lambda \neq 1$ and one near $\lambda = 1$.

If we choose β sufficiently small relative to h^2, so that $\sqrt{\beta}h^{-2} \leq (1+\varepsilon)/\varepsilon$, then $\varepsilon/(\alpha+\varepsilon)$ becomes a lower eigenvalue bound.

The nonlinearity of the problem is of a special nature. The location of active points is unknown and can vary during the iterations. However, it turns out that a hybrid nonlinear-linear method, similar to the nonlinear two- or multilevel mesh method, functions well, see further [11].

Numerical tests show that there are very few outer iterations needed for convergence. The number of inner iterations is stabilized using a suitable dynamic stopping criterion. The regularization parameter ε must be small but should not be too small, compared to h. For further details, see [11].

4.3 PRESB for the numerical solution of the Schrödinger equation

In [15], the PRESB preconditioner is applied to solve the algebraic systems arising in the discrete time-dependent Schrödinger equation,

$$i\hbar\frac{\partial}{\partial t}\Psi(x,t) = \widehat{H}\Psi(x,t). \tag{28}$$

Here $\hbar$ is Planck's constant, divided by 2π and $\widehat{H}$ is the (quantum) Hamiltonian, expressing the kinetic and the potential energy operators of the system under consideration. The

arising linear system to be solved is of the form

$$(M + i\frac{\tau}{2}K)\boldsymbol{u}^{(n+1)} = (M - i\frac{\tau}{2}K)\boldsymbol{u}^n,$$

where τ is the time step and $\boldsymbol{u}^n$, the solution of the previous time level, is assumed to be known.

The test matrices M and K for this problem are both spd and originate from the spatial discretization of (28) using Radial Basis Functions. These matrices are dense and can be extremely ill-conditioned. In this case PRESB outperforms strongly the PMHSS method, cf. [15].

4.4 PRESB applied to multiphase problems

As already mentioned, PRESB has been successfully used for the sequential and parallel computer simulations of two- and multiphase flow problems, modelled using the Cahn-Hilliard equation, cf. [27, 28, 29].

5 Concluding remarks

In this paper we compare four types of methods, used to precondition two-by-two block matrices with square blocks. In particular, we summarize the theoretical properties of the PRESB preconditioner and the existing numerical evidence that PRESB outperforms the other methods, having a condition number bound 2 or smaller, i.e. a corresponding bound for the rate of convergence factor 1/2 or smaller. This holds uniformly with respect to both problem and method parameters, including the discretization step. Nevertheless, for some tested two-dimensional problems it is shown in [15] that PMHSS is quite competitive and gives smaller computing times than PRESB, referred to in [15] as 'C-to-R'. A method, also referred to as 'C-to-R' in [48] is a forerunner to PRESB. An observation made is that for problems with a complex-valued solution, the imaginary part of the residuals can be sometimes significantly larger than their real part. This indicates that one should use an alternative iteration approach also for the PRESB method, where applicable, that is, alternate with the method between the matrices $\begin{bmatrix} A & B \\ B & -A \end{bmatrix}$ and $\begin{bmatrix} B & A \\ A & -B \end{bmatrix}$. In [15], indeed, it is shown that such a version of PRESB ('C-to-R') can give smaller error norms.

The strongest competitor is a special version of the PMHSS method in the form of a particular alternating direction method, for which the convergence factor bound is also 1/2. However, if applied in its generic form, during each iteration step, this method requires about a double amount of computations as required by the PRESB method, i.e. it can be said to correspond to a convergence factor $1/\sqrt{2}$. Furthermore, in general the spectrum of the iteration matrix has complex eigenvalues. With a special choice of the involved auxiliary matrix, the computational cost of PRESB and PMHSS becomes comparable.

In various publications the considered methods have been shown to have a robust performance with respect to various parameters involved. In some cases, as shown in [13],

the convergence of the block-diagonal preconditioning method deteriorates for small values of the cost regularization and the reluctivity parameters, when applied to an eddy current electromagnetic problem.

In an extensive set of test problems of different types of problems, published in earlier works, it has been shown that the PRESB method converges uniformly well with respect to all mesh, regularization and problem parameters. As listed in Section 4 this includes complex symmetric linear systems, Cahn-Hilliard multiphase models, Schrödinger equation, (distributed) optimal control problems with constraint in the form of an elliptic equation, convection-diffusion, Stokes, time-harmonic parabolic equation, eddy current electromagnetic problems, also when box-type of state constraints are imposed.

Another advantage of PRESB is that in many applications it gives tight and sharp eigenvalue bounds, which implies that a Chebyshev acceleration method can be applied instead of a Krylov subspace iteration method (see e.g. [49]). In this way one can avoid computations of global vector inner products and save time, both computational and global communication, in particular when the method is implemented on high performance parallel computer platforms.

Acknowledgement

The first author has been supported by The Ministry of Education, Youth and Sports from the National Programme of Sustainability (NPU II) project "IT4Innovations of excellence in science - LQ1602".

References

[1] M. Kolmbauer, U. Langer, A robust preconditioned MINRES solver for distributed time-periodic eddy current optimal control problems, *SIAM J. Sci. Comput.* 34 (2012), B785-B809.

[2] O. Axelsson, D. Lukáš, Preconditioning methods for eddy current optimally controlled time-harmonic electromagnetic problems. *J. Numer. Math.* 2018. To appear.

[3] M. Stoll, A. Wathen, Preconditioning for partial differential equation constrained optimization with control constraints. *Numer. Linear Algebra Appl.* 19 (2012), 53-71.

[4] J.-W. Pearson, M. Stoll, A.-J. Wathen, Regularization-Robust Preconditioners for Time-Dependent PDE-Constrained Optimization Problems, *SIAM J. Matrix Anal. & Appl.*, 33 (2012), 1126-1152.

[5] Z.-Z. Bai, M Benzi, F Chen, On preconditioned MHSS iteration methods for complex symmetric linear systems, *Numer. Algorithms* 56 (2011), 297-317.

[6] Z.-Z. Bai, M. Benzi, F. Chen, Z.-Q. Wang, Preconditioned MHSS iteration methods for a class of block two-by-two linear systems with applications to distributed control problems. *IMA J. Numer. Anal.* 33 (2013), 343-369.

[7] Z.-Z. Bai, Rotated block triangular preconditioning based on PMHSS, *Sci. China Math.* 56 (2013), 2523-2538.

[8] O. Axelsson, S. Farouq, M. Neytcheva, Comparison of preconditioned Krylov subspace iteration methods for PDE-constrained optimization problems. Poisson and convection-diffusion control. *Numer. Algorithms*, 73 (2016), 631-663.

[9] O. Axelsson, S. Farouq, M. Neytcheva. Comparison of preconditioned Krylov subspace iteration methods for PDE-constrained optimization problems. Stokes control. *Numer. Algorithms*, 74 (2017), 19-37.

[10] O. Axelsson, S. Farouq, M. Neytcheva. A preconditioner for optimal control problems constrained by Stokes equation with a time-harmonic control. *J. Comp. Appl. Math.* 310 (2017): 5-18.

[11] O. Axelsson, M. Neytcheva, A. Ström, An efficient preconditioning method for the state box-constrained optimal control problem. TR 2018-008, May 2018, `http://www.it.uu.se/research/publications/reports/2018-008/`. Submitted.

[12] Yi-Fen Ke, Chang-Feng Ma, Some preconditioners for elliptic PDE-constrained optimization problems, *Comput. Math. Appl.* 75 (2018), 2795-2813.

[13] M. Kolmbauer, The Multiharmonic Finite Element and Boundary Element Method for Simulation and Control of Eddy Current Problems. Ph.D. thesis, *Johannes Kepler Universität*, Linz, Austria (2012).

[14] Z.-Z. Bai, Block preconditioners for elliptic PDE-constrained optimization problems. *Computing* 91 (2011), 379-395.

[15] O. Axelsson, M. Neytcheva, B. Ahmad. A comparison of iterative methods to solve complex valued linear algebraic systems. *Numer. Algorithms* 66 (2014), 811-841.

[16] O. Axelsson, J. Karatson, F. Magoules, Superlinear convergence using block preconditionerd for the real system formulaton of complex Helmholtz equations, *J of Computational and Applied Mathematics*, 2018, DOI 10.1016/J.cam.2018.01.029.

[17] R.S. Varga, Matrix Iterative Analysis. *Prentice-Hall, Inc., Englewood Cliffs*, N. J., 1962.

[18] O. Axelsson, *Iterative Solution Methods*. Cambridge University Press, Cambridge 1994.

[19] O. Axelsson, P.S. Vassilevski, A black box generalized conjugate gradient solver with inner iterations and variable-step preconditioning. *SIAM. J. Matrix Anal. Appl.* 12 (1991), 625-644.

[20] Y. Saad, A flexible inner-outer preconditioned GMRES algorithm. *SIAM J. Sci. Comput.* 14 (1993), 461-469.

[21] Z.-Z. Bai, G. Golub, M. Ng, Hermitian and skew-Hermitian splitting methods for non-Hermitian positive definite linear systems, *SIAM J. Matrix Anal. Appl.* 24 (2003), 603-626.

[22] Jing Wang, Xue-Ping Guo, Hong-Xiu Zhong Accelerated GPMHSS Method for Solving Complex Systems of Linear Equations, *East Asian J. Appl. Math.* 7 (2107), 143–155.

[23] Z.-Q. Wang, On a Chebyshev accelerated splitting iteration method with application to two2̆010by2̆010two block linear systems, *Numer. Linear Algebra Appl.* 2018;e2172. https://doi.org/10.1002/nla.2172

[24] Z.-Z. Bai, G. Golub, Accelerated Hermitian and skew-Hermitian splitting iteration methods for saddle-point problems. *IMA J. Numer. Anal.* 27 (2007), 1-23.

[25] Z.-Z. Bai, M. Benzi, Regularized HSS iteration methods for saddle-point linear systems. *BIT Numer. Math.* 57 (2017), 287-311.

[26] Y. Dong, C. Gu, On PMHSS iteration methods for continuous Sylvester equations, *J. Computational Math.* 35 (2017), 600-619.

[27] P. Boyanova, M. Do-Quang, M. Neytcheva, Efficient preconditioners for large scale binary Cahn-Hilliard models, *Computational Methods in Applied Mathematics*, 12 (2012), 1–22.

[28] O. Axelsson, P. Boyanova, M. Kronbichler, M. Neytcheva, X. Wu. Numerical and computational efficiency of solvers for two-phase problems, *Computers and Math. Appl.*, 65 (2013), 301–314.

[29] P. Boyanova, M. Neytcheva, Efficient numerical solution of discrete multi-component Cahn-Hilliard systems *Computers and Math. Appl.*, 67 (2014), 106–121.

[30] O. Axelsson, M. Neytcheva, Z.-Z. Liang, Parallel solution methods and preconditioners for evolution equations. *Math. Modelling and Analysis*, 23 (2018), 287–308.

[31] A. Battermann, E. Sachs, Block preconditioners for KKT systems in PDE-governed optimal control problems, Schulz V. (eds) Fast Solution of Discretized Optimization Problems. ISNM International Series of Numerical Mathematics, vol 138. Birkhuser, Basel, 2000, 1-18.

[32] E Haber, U M Ascher, Preconditioned all-at-once methods for large, sparse parameter estimation problems, *Inverse Problems* 17 (2001), 1847-1864.

[33] K. Ito, K. Kunisch, Semi-smooth Newton methods for state-constrained optimal control problems. *Systems & Control Letters* 50 (2003), 221-228.

[34] J. Schöberl, W. Zulehner, Symmetric indefinite preconditioners for saddle point problems with applications to PDE-constrained optimization problems. *SIAM J. Matrix Anal. Appl.* 29 (2007), 752-773.

[35] M. Hintermüller, M. Hinze, Moreau-Yosida regularization in state constrained elliptic control problems: error estimates and parameter adjustment. *SIAM J. Numer. Anal.* 47 (2009), 1666-1683.

[36] R. Herzog, E. Sachs, Preconditioned conjugate gradient method for optimal control problems with control and state constraints *SIAM J. Matrix Anal. Appl.* 31 (2010), 2291-2317.

[37] T Rees, M. Stoll, Block-triangular preconditioners for PDE-constrained optimization, *Numer. Linear Algebra Appl.* 17 (2010), 977-996.

[38] M. Kollmann, W. Zulehner, A robust preconditioner for distributed optimal control for Stokes flow with control constraints. *Numerical Math. Advanced Appl.* 2011, 771-779, Springer, Heidelberg, 2013.

[39] J. Pearson, A. Wathen, A new approximation of the Schur complement in preconditioners for PDE-constrained optimization. *Numer. Linear Algebra Appl.* 19 (2012), 816-829.

[40] V. Simoncini, Reduced order solution of structured linear systems arising in certain PDE-constrained optimization problems, *Comput. Optim. Appl.* 53 (2012), 591-617.

[41] W. Zulehner, Efficient solvers for saddle point problems with applications to PDE-constrained optimization. Advanced finite element methods and applications, 197-216, Lect. Notes Appl. Comput. Mech., 66, Springer, Heidelberg, 2013.

[42] J.-W. Pearson, M. Stoll, A.-J. Wathen, Preconditioners for state-constrained optimal control problems with Moreau-Yosida penalty function. *Numer. Linear Alg. Appl.* 21 (2014), 81-97.

[43] M. Porcelli, V. Simoncini, M. Tani, Preconditioning of active-set Newton methods for PDE-constrained optimal control problems. *SIAM J. Sci. Comput.* 37 (2015), S472-S502.

[44] B. Morini, V. Simoncini, M. Tani, A comparison of reduced and unreduced KKT systems arising from interior point methods, *Comput. Optim. Appl.* 68 (2017), 1-27.

[45] R. Becker, B. Vexler, Optimal control of the convection-diffusion equation using stabilized finite element methods, *Numer. Math.*, 106, 349-367 (2007).

[46] C.C. Paige, M.A. Saunders, Solution of sparse indefinite systems of linear equations. *SIAM J. Numer. Anal.* 12 (1975), 617-629.

[47] J.C. Nèdèlec. Mixed finite elements in $\mathbb{R}^3$. *Numer. Math.* 35 (1980), 315-341.

[48] O. Axelsson, A. Kucherov, Real valued iterative methods for solving complex symmetric linear systems. *Numer. Linear Algebra Appl.* 7 (2000), 197-218.

[49] O. Axelsson, Z.-Z. Liang, Inner product-free iterative solution and elimination methods for linear systems of three-by-three block matrix form. 2018. Submitted.