Two sides tangential filtering decomposition

Laura Grigori, Frédéric Nataf, Qiang Niu

INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Two sides tangential filtering decomposition

Laura Grigori — Frederic Nataf — Qiang Niu

N° 6554

June 2008

Thème NUM

ISSN 0249-6399 ISRN INRIA/RR--6554--FR+ENG

INSTITUT NATIONAL
DE RECHERCHE
EN INFORMATIQUE
ET EN AUTOMATIQUE

INRIA

centre de recherche
SACLAY - ÎLE-DE-FRANCE

Two sides tangential filtering decomposition

Laura Grigori* , Frederic Nataf † , Qiang Niu‡

Thème NUM — Systèmes numériques
Équipes-Projets Grand-Large

Rapport de recherche n° 6554 — June 2008 — 25 pages

Abstract: In this paper, left and two sides tangential filtering decompositions are introduced. The filtering preconditioner constructed by the decomposition is combined with classical **ILU**(**0**) preconditioner in multiplicative ways to produce composite preconditioners. Analysis show that the composite preconditioners benefit from each of the preconditioners. On the filtering vector, we reveal that *ones* is a reasonable choice, which is effective and can avoid the preprosessing needed in other methods to build the filtering vector. Numerical tests show that the composite preconditioners are rather robust and efficient for block tridiagonal linear systems arising from the discretisation of partial differential equations with strongly varying coefficients.

Key-words: preconditioner; linear system; frequency filtering decomposition; GMRES

* INRIA Saclay - Ile de France, Laboratoire de Recherche en Informatique Universite Paris-Sud 11, France (Email:laura.grigori@inria.fr).

† Laboratoire J. L. Lions, CNRS UMR7598, Universite Paris 6, France; Email: nataf@ann.jussieu.fr

‡ School of Mathematical Sciences, Xiamen University, Xiamen, 361005, P.R. China; The work of this author was performed during his visit to INRIA, funded by China Scholarship Council; Email:kangniu@gmail.com

Centre de recherche INRIA Saclay – Île-de-France
Parc Orsay Université
4, rue Jacques Monod, 91893 ORSAY Cedex
Téléphone : +33 1 72 92 59 00

Two sides tangential filtering decomposition

Résumé : Dans ce papier nous introduisons la decomposition basée sur un filtrage tangentiel à gauche et de deux côtés. Le préconditionnement obtenu par ce filtrage est combiné avec le préconditionnement classique **ILU**(**0**) de manière multiplicative. Notre analyse montre que le préconditionnement composé hérite des avantages de chacun des prconditionnements. Pour le vecteur de filtrage, nous montrons que *ones* est un choix efficace et peut éviter la phase de pretraitement nécessaire par les autres méthodes pour établir ce vecteur. Les résultats numériques montrent que le préconditionnement composé est robuste et efficace pour les systmes linéaires avec une structure bloc tridiagonal rsultant de la discrétisation des équations différentielles partielles avec des coefficients fortement variables.

Mots-clés : préconditionnement, systèmes linéaires, décomposition basé sur filtrage tangentiel, GMRES

1 Introduction

Large sparse linear system

$$\mathbf{A}\mathbf{x} = \mathbf{b} \tag{1}$$

with

$$\mathbf{A} = \begin{bmatrix} D_1 & U_1 & & \\ L_1 & D_2 & \ddots & \\ & \ddots & \ddots & U_{n_x-1} \\ & & L_{n_x-1} & D_{n_x} \end{bmatrix} \in \mathcal{R}^{n\times n}, \quad \mathbf{b} \in \mathcal{R}^n$$

arise in many applications. For example, when solving non-linear partial differential equations in heterogeneous media, the simulation procedure usually involves a sequence of linear systems of form (1). The quality of the simulation depends to a large degree on the efficiency of the linear system solvers. Due to the discontinuous coefficients in the **PDE** problems and the large size of $\mathbf{A}$, solving (1) usually poses difficulties for Krylov subspace iterative methods preconditioned by conventional preconditioners, e.g. **ILU**(**0**). Therefore, there is a strong need for constructing efficient preconditioners.

As is well known, the block tridiagonal matrix $\mathbf{A}$ admits an incomplete decomposition [10, 11, 16] of the form $\mathbf{A} = (\mathbf{L}+\mathbf{T})\mathbf{T}^{-1}(\mathbf{T}+\mathbf{U})$, or explicitly

$$\mathbf{A} = \begin{bmatrix} T_1 & & & \\ L_1 & T_2 & & \\ & \ddots & \ddots & \\ & & L_{n_x-1} & T_{n_x} \end{bmatrix} \begin{bmatrix} T_1^{-1} & & & \\ & T_2^{-1} & & \\ & & \ddots & \\ & & & T_{n_x}^{-1} \end{bmatrix} \begin{bmatrix} T_1 & U_1 & & \\ & T_2 & \ddots & \\ & & \ddots & U_{n_x-1} \\ & & & T_{n_x} \end{bmatrix}, \tag{2}$$

where $T_i \in \mathcal{R}^{n_{yz}\times n_{yz}}$ are nonsingular and $n_{yz} = n/n_x$. The matrices T_i can be computed by the following recursive formula

$$T_i = \begin{cases} D_1, & i = 1, \\ D_i - L_{i-1}T_{i-1}^{-1}U_{i-1}, & 1 < i \le n_x. \end{cases} \tag{3}$$

If the size of matrix $\mathbf{A}$ is small, then the procedure (3) can be used to form the exact block **LU** decomposition for $\mathbf{A}$. Thus, by solving the block triangular linear systems involved in (2), the original problem (1) can be solved directly. However, for large sparse linear systems, the above procedure is too costly, since T_i generally becomes denser and denser during the recursion (3). For example, if D_1 is an irreducible tridiagonal matrix, then the inverse of D_1 is a dense matrix, and T_2 and all the subsequent $T_i s$. Therefore, using procedure (3) directly is not practical for large problems. But the process can be used to construct preconditioners by approximating the Schur complements appropriately. The construction of the Schur complement approximations should consider two factors: the degree of approximation, and the preservation of the sparsity.

Many research works have addressed approaches to construct the Schur complement approximations [6, 15, 17]. Particularly, several papers consider approximations that are able to produce the same effect with the corresponding exact Schur complements when operating on certain vectors. The preconditioners constructed in this way can preserve certain filtering property as will be described

later. Several ways to construct the filtering preconditioners have been proposed. Particularly, a class of Frequency Filtering Decomposition (**FFD**) and Tangential Frequency Filtering Decomposition (**TFFD**) preconditioners have been investigated in [1, 2, 8, 9, 20, 21, 22]. These methods take into account the above two factors and develop several elegant approaches to approximate the Schur complements. Especially for the vectors 'close to' the filtering vectors, the Schur complements can be well approximated. On the choice of the filtering vectors, the sine functions are considered in [24], eigenvectors associated with certain generalized eigenvalue problems are used in [20, 21], adaptive test vectors are suggested in [22], and Ritz vectors are considered in [2].

These methods rely on an approximate incomplete block ILU factorization preconditioner

$$\mathbf{M} = \begin{bmatrix} \tilde{T}_1 & & & \\ L_1 & \tilde{T}_2 & & \\ & \ddots & \ddots & \\ & & L_{n_x-1} & \tilde{T}_{n_x} \end{bmatrix} \begin{bmatrix} \tilde{T}_1^{-1} & & & \\ & \tilde{T}_2^{-1} & & \\ & & \ddots & \\ & & & \tilde{T}_{n_x}^{-1} \end{bmatrix} \begin{bmatrix} \tilde{T}_1 & U_1 & & \\ & \tilde{T}_2 & \ddots & \\ & & \ddots & U_{n_x-1} \\ & & & \tilde{T}_{n_x} \end{bmatrix}. \tag{4}$$

During the construction of $\tilde{T}_i$, different approaches are considered to enable $\mathbf{M}$ to have (or approximately have) the right filtering property

$$\mathbf{A}\mathbf{f} = \mathbf{M}\mathbf{f}, \tag{5}$$

for $\mathbf{f} = [f_1, \ldots, f_{n_x}]^T \in \mathcal{F}$, where $\mathcal{F}$ is a test subspace. In particular, the following induction formula is proposed in [2]

$$\tilde{T}_i = \begin{cases} D_1, & i = 1, \\ D_i - L_{i-1}(2\beta_{i-1} - \beta_{i-1}\tilde{T}_{i-1}\beta_{i-1})U_{i-1}, & 1 < i \leq n_x. \end{cases} \tag{6}$$

where β_{i-1} is an approximation to $\tilde{T}_{i-1}^{-1}$, $i = 2, \ldots, n_x$, and can be computed by certain filtering conditions.

In this paper, we first show that the approximations β_{i-1}, $i = 1, \ldots, n_x$, can be determined in a different way, that enables the preconditioner to satisfy certain left filtering property. Then we propose a generalization of the induction formula (6). For symmetric problems, if the same left and right filtering vectors are used, then the new formula is equivalent to (6). However, for nonsymmetric problems, the new formula has both the right and left filtering properties. The choice of the filtering vector is an important issue. Instead of using the Ritz vector (corresponding to the lowest Ritz value) as the filtering vector, we propose to use $ones = [1, 1, \ldots, 1]^T$ as the filtering vectors in this paper. This simple choice of the filtering vector is rather efficient for our tested problems and can save the preprocessing that is needed in other methods to construct the filtering vector.

By combining the newly built two sides tangential frequency filtering decomposition preconditioner with the **ILU(0)** preconditioner $\mathbf{M}_{ilu}$, the composite preconditioners implicitly defined by

$$\mathbf{M}^{-1} + \mathbf{M}_{ilu}^{-1} - \mathbf{M}^{-1}\mathbf{A}\mathbf{M}_{ilu}^{-1}$$

and

$$\mathbf{M}^{-1} + \mathbf{M}_{ilu}^{-1} - \mathbf{M}_{ilu}^{-1}\mathbf{A}\mathbf{M}^{-1}$$

are discussed. It is shown that both composite preconditioners can partially inherit the filtering properties. We illustrate several interesting properties of the composite preconditioner with left filtering property. Particularly, it is shown that the sum of the residual vector is zero throughout the iterations, if the starting approximation is chosen appropriately.

By assuming each of the preconditioners is derived from a splitting of $\mathbf{A}$, the explicit forms of the composite preconditioners are discussed. Based on the explicit form and certain assumptions, we show that the preconditioned matrix by using each of the composite preconditioner is symmetric with respect to certain non-standard inner product. This is potentially useful if the property can be exploited in the iterative methods, see references [13, 18, 19]. Spectrum analysis shows that the composite preconditioners benefit from each of the preconditioners, and tend to make the spectrum clustered at one. Several examples are given to illustrate the spectrum distribution of the preconditioned matrices by using different preconditioners. On the two combination approaches, we reveal that there is little difference between them. Particularly, For the preconditioned fixed point iteration, we proved that the two combination approaches produce the same convergence rate. For the Krylov subspace methods, e.g. **FGMRES** method employed in this paper, we find that there is at most one step difference between the two combination preconditioning approaches. Finally, some challenging linear system problems arising from discretization of boundary problems are tested. The results show that the composite preconditioners proposed in this paper are efficient, and converge much faster than the classical **ILU**(**0**) preconditioner, and similar type of composite preconditioners by using different filtering vectors.

The paper is organized as follows. In Section 2, after briefly reviewing the **TFFD** [2], we introduce a left filtering approach of constructing the decomposition. Then we propose the two sides tangential frequency filtering decomposition. The properties of two sides tangential frequency filtering preconditioner and the composite preconditioners are analyzed in Section 3. Numerical tests are described in Section 4. we conclude the paper in Section 5, and present the spectrum distribution plots as an Appendix in Section 6.

2 On the tangential filtering decomposition

2.1 Left tangential filtering decomposition and generalization

In this subsection, we first briefly recall the tangential filtering decomposition based preconditioners which have been proposed and analyzed in [2], then illustrate another approach to complete the decomposition by enabling the preconditioner to have the left filtering property.

The derivation of the tangential filtering decomposition (5) is based on the following observation: assume that β_{i-1} is a sparse approximation of $\tilde{T}_{i-1}^{-1}$ and it satisfies

$$||I - \tilde{T}_{i-1}\beta_{i-1}|| \leq \alpha < 1. \tag{7}$$

Then we have

$$\begin{aligned} (\tilde{T}_{i-1}\beta_{i-1})^{-1} &= (I - (I - \tilde{T}_{i-1}\beta_{i-1}))^{-1} \\ &= I + (I - \tilde{T}_{i-1}\beta_{i-1}) + (I - \tilde{T}_{i-1}\beta_{i-1})^2 + \cdots \end{aligned}$$

From the above series and assumption (7), we can see that the first two terms $I + (I - \tilde{T}_{i-1}\beta_{i-1})$ can be used to approximate the inverse of $\tilde{T}_{i-1}\beta_{i-1}$. Therefore, using $I + (I - \tilde{T}_{i-1}\beta_{i-1})$ as a preconditioner for the preconditioned matrix $\tilde{T}_{i-1}\beta_{i-1}$ again, a twice preconditioned matrix can be deduced as follows

$$\tilde{T}_{i-1}\beta_{i-1}(I + (I - \tilde{T}_{i-1}\beta_{i-1})) = \tilde{T}_{i-1}(2\beta_{i-1} - \beta_{i-1}\tilde{T}_{i-1}\beta_{i-1}).$$

Regarding

$$M_{2\beta} = 2\beta_{i-1} - \beta_{i-1}\tilde{T}_{i-1}\beta_{i-1}$$

as a single preconditioner for $\tilde{T}_{i-1}$, then

$$||I - \tilde{T}_{i-1}M_{2\beta}|| = ||(I - T_{i-1}\beta_{i-1})^2|| \leq \alpha^2.$$

Therefore, $M_{2\beta}$ is a better approximation of $\tilde{T}_{i-1}^{-1}$ than β_{i-1}. The matrix β_{i-1} can be determined by letting $\mathbf{A}$ satisfies the right filtering property (5), which is equivalent to enabling

$$(\tilde{T}_{i-1}\beta_{i-1} - I)U_{i-1}f_i = 0.$$

Thus, a diagonal matrix β_{i-1} can be computed as

$$\beta_{i-1} = Diag(\tilde{T}_{i-1}^{-1}U_{i-1}f_i./(U_{i-1}f_i)), \tag{8}$$

where $Diag(v)$ is the diagonal matrix constructed from the vector v, and $./$ denotes the pointwise vector division.

During the above construction of the filtering preconditioner, we notice that a left filtering property

$$\mathbf{g}^T\mathbf{A} = \mathbf{g}^T\mathbf{M} \tag{9}$$

can also be exploited, where $\mathbf{g} = [g_1, \ldots, g_{n_x}]^T \in \mathcal{G}$, and $\mathcal{G}$ is a test subspace. Based on the procedure of constructing the right filtering preconditioners, the left filtering property (9) can be satisfied by determining β_{i-1} appropriately. The same analysis shows that it is sufficient to enable

$$g_i^T L_{i-1}(\beta_{i-1}\tilde{T}_{i-1} - I) = 0$$

i.e.

$$\beta_{i-1} = Diag(\tilde{T}_{i-1}^{-T}L_{i-1}^T g_i./L_{i-1}^T g_i). \tag{10}$$

This way of constructing preconditioners with left filtering property is reminiscent of residual constraint type methods [4, 23].

2.2 Two sides tangential filtering decomposition

In this paper, we introduce the two sides tangential filtering decomposition. Suppose we have two approximations β_{i-1} and γ_{i-1} to $\tilde{T}_{i-1}^{-1}$. Assume the approximations satisfy

$$||I - \tilde{T}_{i-1}\beta_{i-1}|| \leq \alpha < 1 \quad and \quad ||I - \tilde{T}_{i-1}\gamma_{i-1}|| \leq \alpha < 1$$

respectively. Then we can combine the two approximations as

$$M_{\beta\gamma} = \beta_{i-1} + \gamma_{i-1} - \beta_{i-1}\tilde{T}_{i-1}\gamma_{i-1}. \tag{11}$$

By the assumptions, it holds that

$$||I - \tilde{T}_{i-1}M_{\beta\gamma}|| \leq ||I - \tilde{T}_{i-1}\beta_{i-1}||||I - \tilde{T}_{i-1}\gamma_{i-1}|| \leq \alpha^2.$$

Therefore, $M_{\beta\gamma}$ should also be a better approximation of $\tilde{T}_{i-1}^{-1}$ than just using β_{i-1} or γ_{i-1}. Generally, combining two different types of the preconditioners should be more efficient than combining a single type preconditioner by itself. Here, we should remind that the approximations discussed above are related to approximating the Schur complements appeared in (3).

Using the new combination approach (11) to approximate $\tilde{T}_{i-1}^{-1}$, the induction formula (6) can be replaced by

$$\tilde{T}_i = \begin{cases} D_1, & i = 1, \\ D_i - L_{i-1}(\beta_{i-1} + \gamma_{i-1} - \gamma_{i-1}\tilde{T}_{i-1}\beta_{i-1})U_{i-1}, & 1 < i \leq n_x, \end{cases} \tag{12}$$

where β_{i-1} and γ_{i-1} are approximations to $\tilde{T}_{i-1}^{-1}$, and the approaches to form the approximations will be described later. We should point out that the same notations $\tilde{T}_i$ are used in (12) as the ones used in formula (6).

By setting $\Theta_{i,i-1} = L_{i-1}\gamma_{i-1}$ and $\Theta_{i-1,i} = \beta_{i-1}U_{i-1}$, then it is not difficult to see that the induction formula (12) reduces to the following formula

$$\tilde{T}_i = \begin{cases} D_1, & i = 1, \\ D_i - \Theta_{i,i-1}U_{i-1} + L_{i-1}\Theta_{i-1,i} - \Theta_{i,i-1}\tilde{T}_{i-1}\Theta_{i-1,i}, & 1 < i \leq n_x, \end{cases}$$

proposed in [21] for nonsymmetric problems. However, the approach of constructing the approximations discussed in this paper is quite different from that of [21], where symmetrization is carried out before determining the transfer matrices $\Theta_{i,j}$. Thus the filtering properties don't exist any longer.

Based on the induction formula (12), the incomplete factorization preconditioner can be written in compact form

$$\mathbf{M} = (\mathbf{L} + \tilde{\mathbf{T}})\tilde{\mathbf{T}}^{-1}(\tilde{\mathbf{T}} + \mathbf{U}), \tag{13}$$

with

$$\tilde{\mathbf{T}} = \begin{bmatrix} \tilde{T}_1 & & & \\ & \tilde{T}_2 & & \\ & & \ddots & \\ & & & \tilde{T}_{n_x} \end{bmatrix}, \mathbf{L} = \begin{bmatrix} 0 & & & \\ L_1 & 0 & & \\ & \ddots & \ddots & \\ & & L_{n_x-1} & 0 \end{bmatrix}, \mathbf{U} = \begin{bmatrix} 0 & U_1 & & \\ & 0 & \ddots & \\ & & \ddots & U_{n_x-1} \\ & & & 0 \end{bmatrix},$$

where the diagonal blocks $\tilde{T}_i$, $i = 1, 2, \ldots, n_x$, are formed by (12). Expanding (13) explicitly, $\mathbf{M}$ can be written as

$$\mathbf{M} = \mathbf{L} + \mathbf{U} + BDiag(\tilde{T}_1, \tilde{T}_1 + L_1\tilde{T}_1^{-1}U_1, \ldots, \tilde{T}_{n_x} + L_{n_x-1}\tilde{T}_{n_x}^{-1}U_{n_x-1}), \tag{14}$$

where $BDiag$ denotes the block diagonal matrix. Thus, only from the second to the last block diagonal part of $\mathbf{M}$ differs from that of $\mathbf{A}$. We remark that the form of preconditioner $\mathbf{M}$ resembles the constraint preconditioner [12] for saddle point problems, whereas it is more general and is used as a preconditioner for block tridiagonal linear systems.

When applying the preconditioner $\mathbf{M}$, linear systems of the form $\mathbf{M}\varphi = \mathbf{z}$ need to be solved. Using (14), this is equivalent to solving

$$(\mathbf{L}\tilde{\mathbf{T}}^{-1} + \mathbf{I})\mathbf{y} = \mathbf{z} \;\; and \;\; (\tilde{\mathbf{T}} + \mathbf{U})\varphi = \mathbf{y}. \tag{15}$$

As both linear systems in (15) are block triangular, the forward and backward sweeps only involve solving linear systems of the type

$$\tilde{T}_i u_i = v_i, \quad 1 \leq i \leq n_x. \tag{16}$$

The following lemma provides some information on how to compute the diagonal matrices β_{i-1} and γ_{i-1} such that $\mathbf{M}$ has the two sides filtering properties.
Lemma 2.1
The difference between the preconditioner $\mathbf{M}$ and the coefficient matrix $\mathbf{A}$ has the following block diagonal form

$$\mathbf{M} - \mathbf{A} = BDiag(N_1, N_2, \ldots, N_{n_x}),$$

where

$$N_i = \begin{cases} 0, & i = 1, \\ L_{i-1}(\gamma_{i-1}\tilde{T}_{i-1} - I)\tilde{T}_{i-1}^{-1}(\tilde{T}_{i-1}\beta_{i-1} - I)U_{i-1}, & 1 < i \leq n_x. \end{cases} \tag{17}$$

Proof
From (2), (14) and the induction formula (12), it is easy to see that

$$N_1 = 0,$$

and

$$N_i = L_{i-1}(-\gamma_{i-1} - \beta_{i-1} + \gamma_{i-1}\tilde{T}_{i-1}\beta_{i-1} + \tilde{T}_{i-1}^{-1})U_{i-1}, \quad 2 \leq i \leq n_x,$$

or written in compact form

$$N_i = L_{i-1}(\gamma_{i-1}\tilde{T}_{i-1} - I)\tilde{T}_{i-1}^{-1}(\tilde{T}_{i-1}\beta_{i-1} - I).$$

Thus (17) holds. □

Now we consider how to form the approximations γ_{i-1} and β_{i-1}. Let

$$\mathbf{f} = [f_1, f_2, \ldots, f_{n_x}]^T \quad and \quad \mathbf{g} = [g_1, g_2, \ldots, g_{n_x}]^T$$

be two given vectors. If there are no zero entries in the vectors $U_{i-1}f_i$ and $L_{i-1}^T g_i$, then it is possible to find diagonal matrices β_{i-1} and γ_{i-1} such that $\mathbf{M}$ produces the same effect with $\mathbf{A}$ when operating on these vectors, i.e.

$$(\mathbf{M} - \mathbf{A})\mathbf{f} = 0 \tag{18}$$

and

$$\mathbf{g}^T(\mathbf{M} - \mathbf{A}) = 0. \tag{19}$$

From (17), we can see that it is sufficient to make

$$(\tilde{T}_{i-1}\beta_{i-1} - I)U_{i-1}f_i = 0$$

and

$$g_i^T L_{i-1}(\gamma_{i-1}\tilde{T}_{i-1} - I) = 0.$$

These requirements can be satisfied by setting β_{i-1} and γ_{i-1} as follows,

$$\beta_{i-1} = Diag(\tilde{T}_{i-1}^{-1} U_{i-1} f_i ./ U_{i-1} f_i), \tag{20}$$

and

$$\gamma_{i-1} = Diag(\tilde{T}_{i-1}^{-T} L_{i-1}^T g_i ./ L_{i-1}^T g_i), \tag{21}$$

We can see that the above approach of constructing the preconditioners actually merged the right filtering property (18) and the left filtering property (19) together. The preconditioner constructed by this approach is called two sides filtering preconditioner in this paper.

2.3 On the choice of the filtering vectors

The choice of the filtering vector is an important issue, and is widely studied in [2, 8, 9, 20, 21, 22]. Generally, the filtering vector should enable the preconditioner to effectively damp the error components in different frequencies. It has been suggested in [20, 21, 24] that several preconditioners should be constructed by using different types of filtering vectors. Particularly, the sine function

$$(f^j)_k = \sin(\pi \omega_j h k)$$

is considered in [24], where h is the grid size, ω_j is a frequency. The filtering vectors are generalized to eigenvectors associated with certain generalized eigenvalue problem in [20, 21]. The number of filtering vectors is suggested to be proportional of $\log_2(n)$. Then the final preconditioning process is equivalent to implementing a single preconditioner that is formed by combining these different preconditioners in a multiplicative way. For a special class of model problems, the convergence rate is proven to be independent of the number of unknowns. However, there are some difficult cases on which the preconditioned iterative solver is not efficient. As an improvement, an adaptive filtering approach is considered in [22]. The method uses a sequence of filtering vectors (error approximations) that can be computed adaptively. Other filtering methods, for example the tangential decomposition [8] and two-frequency decomposition [9], just consider the average filtering condition, not the exact one. These methods of using a sequence of filtering preconditioners are appealing, but considerable setup time and memory are needed.

In [2], the authors propose a low frequency tangential filtering decomposition, which forms preconditioners with right filtering property. By combining the filtering preconditioner with the classical **ILU**(**0**) preconditioner in a multiplicative way, a composite preconditioner is analyzed. The filtering vector is chosen as the Ritz vector corresponding to the lowest eigenvalue of the preconditioned matrix (by **ILU**(**0**)). The approach has the merit of efficiently smoothing both the high and the low frequency error components, and can effectively mitigate the setup time and memory requirement [2]. However, a preprocessing is still needed to generate the filtering vector, which causes extra computation time.

In this paper, we recommend to use $ones = [1, 1, \ldots, 1]^T$ as both the left and the right filtering vectors. As it will be illustrated by the numerical examples, using *ones* as the filtering vector is robust and generally better than other vectors in terms of iterations. Moreover, this choice can save the preprocessing that is needed in other methods to form the filtering vectors. Therefore, the choice for the filtering vector can be much more efficient in terms of total computational cost and solution time.

For the left filtering vector, we believe that using *ones* as the filtering vector is especially important. According to the analysis in the papers [3, 4, 23], the left filtering is equivalent to imposing a zero sum constraint on the residual vectors computed by the preconditioned iterative solver. By setting appropriate initial approximate solution, this constraint ensures the mass conservation property, and hence the iterative methods based on this preconditioning matrix have zero material balance errors in all phases.

For the choice of the right filtering vectors, we also tried other vectors, e.g. Ritz vectors. However, it is not as efficient as the results of using *ones*. To further exploit the potential power of the tangential frequency filtering preconditioner, we also test and compare different combination approaches of the left and right filtering vectors, like using *ones* as the left filtering vector, and Ritz vector as the right filtering vector, and so on (see the numerical examples in Section 4). It might be possible to explore better choices of the right filtering vector. However, we believe that the preconditioner using *ones* as the filtering vector is efficient in smoothing the error components globally.

3 Analysis of the two sides filtering preconditioner and combination preconditioning

3.1 Properties of the filtering preconditioners

In this subsection, we restrict $\mathbf{A}$ to be symmetric positive definite, and use $\mathbf{A} \succ \mathbf{B}$ ($\mathbf{A} \succeq \mathbf{B}$) to denote that $\mathbf{A} - \mathbf{B}$ is symmetric positive definite (semidefinite). Consider the preconditioner $\mathbf{M}$ formed by (13), which ensure the left filtering property (18) and the right filtering property (19). Assume $\mathbf{g} = \mathbf{f}$ is used in the symmetric case, then it is obvious that the approximations $\beta_i = \gamma_i$. Furthermore, the following Lemma holds and it has been established in [2].

Lemma 3.1
If $\mathbf{A} \succ \mathbf{0}$, then matrices $\tilde{T}_i \succeq T_i$, $1 \leq i \leq n_x - 1$. Moreover, $\mathbf{M} \succ \mathbf{0}$ and $\mathbf{M} - \mathbf{A} \succeq \mathbf{0}$ hold.

As $\mathbf{M}^{-1}\mathbf{A} = \mathbf{M}^{-\frac{1}{2}}(\mathbf{M}^{-\frac{1}{2}}\mathbf{A}\mathbf{M}^{-\frac{1}{2}})\mathbf{M}^{\frac{1}{2}}$, and $\mathbf{M}^{-\frac{1}{2}}\mathbf{A}\mathbf{M}^{-\frac{1}{2}}$ is symmetric, so the eigenvalues of $\mathbf{M}^{-1}\mathbf{A}$ are all real. Then we have the following lemma, it is a tailored version of the generalized Bendisxon theorem established in [7].

Lemma 3.2
For matrices $\mathbf{A} \succ 0$ and $\mathbf{M} \succ 0$, define function $h(v)$ as follows

$$h(v) = \frac{v^T \mathbf{A} v}{v^T \mathbf{M} v}.$$

Assume that there exist positive scalars α_1 and α_2 such that

$$\alpha_1 \leq h(v) \leq \alpha_2, \quad \forall \ v \in \mathcal{R}^n \backslash \{0\},$$

then we have $\alpha_1 \leq \lambda(\mathbf{M}^{-1}\mathbf{A}) \leq \alpha_2$, where $\lambda(\cdot)$ represents the eigenvalue of the corresponding matrix.

With Lemma 3.1 and Lemma 3.2, we have the following theorem

Theorem 3.1
Let

$$\mathbf{A} = \mathbf{M} - \mathbf{N}, \tag{22}$$

be the splitting of coefficient matrix $\mathbf{A}$ induced by the filtering preconditioner $\mathbf{M}$, then the fixed point iteration corresponding to the splitting (22) is convergent, i.e. $\rho(\mathbf{M}^{-1}\mathbf{N}) < 1$.

Proof
As $h(v) = \frac{v^T\mathbf{A}v}{v^T\mathbf{M}v} = \frac{v^T\mathbf{A}v}{v^T\mathbf{A}v+v^T\mathbf{N}v}$, and $\mathbf{N}$ is symmetric positive semidefinite as shown by Lemma 3.1, we have

$$0 < h(v) = \frac{v^T\mathbf{A}v}{v^T\mathbf{A}v + v^T\mathbf{N}v} \leq \frac{v^T\mathbf{A}v}{v^T\mathbf{A}v} = 1.$$

From Lemma 3.2 and nonsingularity of matrices $\mathbf{A}$ and $\mathbf{M}$, we have

$$0 < \lambda(\mathbf{M}^{-1}\mathbf{A}) = \lambda(\mathbf{I} - \mathbf{M}^{-1}\mathbf{N}) \leq 1,$$

i.e. $0 \leq \lambda(\mathbf{M}^{-1}\mathbf{N}) < 1$. The equality holds at the filtering vector $\mathbf{f}$, which is also an eigenvector of matrix $\mathbf{M}^{-1}\mathbf{A}$ corresponding eigenvalue 1. □

For $\mathbf{A} \succ 0$, Theorem 3.1 reveals that the splitting (22) induced by the filtering preconditioner $\mathbf{M}$, is a convergent splitting.

From the spectrum distribution of the preconditioned matrix $\mathbf{M}^{-1}\mathbf{A}$, we can also observe that all the eigenvalues are in the interval $(0, 1]$, this will be shown on some examples in the Appendix of the paper.

3.2 Properties of multiplicative combination preconditioning.

Let $\mathbf{M}_{ilu}$ be the $\mathbf{ILU}(\mathbf{0})$ preconditioner. The associated splitting is

$$\mathbf{A} = \mathbf{M}_{ilu} - \mathbf{N}_{ilu}.$$

There are two multiplicative approaches to combine the preconditioners $\mathbf{M}$ and $\mathbf{M}_{ilu}$,

$$\mathbf{M}_{c_r}^{-1} = \mathbf{M}^{-1} + \mathbf{M}_{ilu}^{-1} - \mathbf{M}_{ilu}^{-1}\mathbf{A}\mathbf{M}^{-1} \tag{23}$$

and

$$\mathbf{M}_{c_l}^{-1} = \mathbf{M}^{-1} + \mathbf{M}_{ilu}^{-1} - \mathbf{M}^{-1}\mathbf{A}\mathbf{M}_{ilu}^{-1}. \tag{24}$$

Here the subscript c_r (c_l) refers to the composite preconditioner, where the subscript $_r$ ($_l$) in c_r (c_l) implies that the corresponding preconditioner has the right (left) filtering property, as will be illustrated later.

Experimentally, we have observed that there is no important difference between using $\mathbf{M}_{c_r}$ or $\mathbf{M}_{c_l}$ as the preconditioner. In the next paragraph, we investigate the properties of the composite preconditioners and throw light on the reason of tiny difference between the two combination approaches. The following theorems reveal that the composite preconditioners $\mathbf{M}_{c_r}$ inherits the right filtering property (18), while $\mathbf{M}_{c_l}$ inherits the left filtering property (19) .

Theorem 3.2
The composite preconditioner $\mathbf{M}_{c_r}$ inherits the right filtering property (18) of $\mathbf{M}$, that is, if $(\mathbf{M}-\mathbf{A})\mathbf{f}=0$, then

$$(\mathbf{M}_{c_r}-\mathbf{A})\mathbf{f}=0. \tag{25}$$

Proof
From (18) and (23) we have

$$\begin{aligned}\mathbf{M}_{c_r}^{-1}\mathbf{A}\mathbf{f} &= \mathbf{M}_{ilu}^{-1}\mathbf{A}\mathbf{f}+\mathbf{M}^{-1}\mathbf{A}\mathbf{f}-\mathbf{M}_{ilu}^{-1}\mathbf{A}\mathbf{M}^{-1}\mathbf{A}\mathbf{f}\\ &= \mathbf{M}_{ilu}^{-1}\mathbf{A}\mathbf{f}+\mathbf{f}-\mathbf{M}_{ilu}^{-1}\mathbf{A}\mathbf{f}\\ &= \mathbf{f},\end{aligned}$$

which is equivalent to $(\mathbf{M}_{c_r}-\mathbf{A})\mathbf{f}=0$. □
Theorem 3.3
The composite preconditioner $\mathbf{M}_{c_l}$ inherits the left filtering property (19) of $\mathbf{M}$, that is, if $\mathbf{g}^T(\mathbf{M}-\mathbf{A})=0$, then

$$\mathbf{g}^T(\mathbf{M}_{c_l}-\mathbf{A})=0. \tag{26}$$

Proof
From (19) and (24) we have

$$\begin{aligned}\mathbf{g}^T\mathbf{A}\mathbf{M}_{c_l}^{-1} &= \mathbf{g}^T\mathbf{A}\mathbf{M}^{-1}+\mathbf{g}^T\mathbf{A}\mathbf{M}_{ilu}^{-1}-\mathbf{g}^T\mathbf{A}\mathbf{M}^{-1}\mathbf{A}\mathbf{M}_{ilu}^{-1}\\ &= \mathbf{g}^T+\mathbf{g}^T\mathbf{A}\mathbf{M}_{ilu}^{-1}-\mathbf{g}^T\mathbf{A}\mathbf{M}_{ilu}^{-1}\\ &= \mathbf{g}^T,\end{aligned}$$

which is equivalent to $\mathbf{g}^T(\mathbf{M}_{c_l}-\mathbf{A})=0$. □

Remarks: When using the filtering preconditioner $\mathbf{M}$ proposed in [2] to combine with $\mathbf{ILU}(\mathbf{0})$, the composite preconditioner has the right filtering property if combination approach (23) is used. However, there is no filtering property if we use the combination approach (24).

For preconditioner $\mathbf{M}_{c_l}$ with left filtering property, if the starting vector $\mathbf{x}_0$ is chosen as $\mathbf{x}_0=\mathbf{M}_{c_l}^{-1}\mathbf{b}$, then the sum of the residual vector $\mathbf{r}_0$ is equal to zero, i.e.

$$\mathbf{g}^T\mathbf{r}_0=\mathbf{g}^T(\mathbf{b}-\mathbf{A}\mathbf{x_0})=\mathbf{g}^T(\mathbf{M}_{c_l}-\mathbf{A})\mathbf{x}_0=\mathbf{0}.$$

Actually, this interesting property can be preserved throughout the iterations, and it has been mentioned in the fixed point iteration setting [4, 23] without proof. For the preconditioned Krylov subspace iteration methods, we give the following theorem formally.
Theorem 3.4
For preconditioned Krylov subspace iterative methods, if the preconditioner $\mathbf{M}_{c_l}$ with left filtering property (26) is used, and the starting vector $\mathbf{x}_0$ is set to be $\mathbf{M}_{c_l}^{-1}\mathbf{b}$, then we have

$$\mathbf{g}^T\mathbf{r}_k=0, \tag{27}$$

where $\mathbf{r}_k=\mathbf{b}-\mathbf{A}\mathbf{x}_k$, and $\mathbf{x}_k$ is a computed approximate solution.
Proof

Suppose the left preconditioning is used, then the kth approximate solution $\mathbf{x}_k$ is derived from the combined subspace

$$\mathbf{x}_k \in \mathbf{x}_0 + \mathcal{K}(\mathbf{r}_0, \mathbf{M}_{c_l}^{-1}\mathbf{A}\mathbf{r}_0, \ldots, (\mathbf{M}_{c_l}^{-1}\mathbf{A})^{k-1}\mathbf{r}_0).$$

Thus, $\mathbf{x}_k$ takes the form of

$$\mathbf{x}_k = \mathbf{x}_0 + \mathcal{P}_{k-1}(\mathbf{M}_{c_l}^{-1}\mathbf{A})\mathbf{M}_{c_l}^{-1}\mathbf{r}_0,$$

where $\mathcal{P}_{k-1}(\lambda)$ is a polynomial of degree no more than $k-1$. Theorefore we have

$$\begin{aligned} \mathbf{r}_k &= \mathbf{r}_0 - \mathbf{A}\mathcal{P}_{k-1}(\mathbf{M}_{c_l}^{-1}\mathbf{A})\mathbf{M}_{c_l}^{-1}\mathbf{r}_0 \\ &= \mathbf{r}_0 - \mathcal{P}_k(\mathbf{A}\mathbf{M}_{c_l}^{-1})\mathbf{r}_0. \end{aligned}$$

Suppose $\mathcal{P}_k(\lambda) = \sum_{i=1}^{k} \alpha_i \lambda^i$, then

$$\begin{aligned} \mathbf{g}^T\mathbf{r}_k &= \mathbf{g}^T\mathbf{r}_0 - \mathbf{g}^T\mathcal{P}_k(\mathbf{A}\mathbf{M}_{c_l}^{-1})\mathbf{r}_0 \\ &= \textstyle\sum_{i=1}^{k} \alpha_i \mathbf{g}^T(I - (\mathbf{A}\mathbf{M}_{c_l}^{-1})^i)\mathbf{r}_0 \\ &= \textstyle\sum_{i=1}^{k} \alpha_i \mathbf{g}^T(I - \mathbf{A}\mathbf{M}_{c_l}^{-1})\mathcal{Q}_{i-1}(\mathbf{A}\mathbf{M}_{c_l}^{-1})\mathbf{r}_0 \\ &= \textstyle\sum_{i=1}^{k} \alpha_i \mathbf{g}^T(\mathbf{M}_{c_l} - \mathbf{A})\mathbf{M}_{c_l}^{-1}\mathcal{Q}_{i-1}(\mathbf{A}\mathbf{M}_{c_l}^{-1})\mathbf{r}_0 \\ &= 0, \end{aligned}$$

where $\mathcal{Q}_{i-1}(\lambda) = \frac{1-\lambda^i}{1-\lambda}$ is a polynomial of degree $i-1$, for each $i = 1, \ldots, k$. □

Now we regard the composite preconditioners $\mathbf{M}_{c_r}$ and $\mathbf{M}_{c_l}$ are derived from the following splittings of $\mathbf{A}$, respectively.

$$\mathbf{A} = \mathbf{M}_{c_r} - \mathbf{N}_{c_r}, \qquad \mathbf{A} = \mathbf{M}_{c_l} - \mathbf{N}_{c_l}. \tag{28}$$

For the corresponding fixed point iteration

$$\mathbf{x}_{k+1} = \mathbf{M}_c^{-1}\mathbf{N}_c\mathbf{x}_k + \mathbf{M}_c^{-1}\mathbf{b}, \tag{29}$$

with $c = c_r, c_l$, we conclude with the following theorem.

Theorem 3.5

For the fixed point iteration (29), the usage of $\mathbf{M}_{c_r}$ and $\mathbf{M}_{c_l}$ as preconditioner leads to the same convergence rate.

Proof

As

$$\begin{cases} \mathbf{I} - \mathbf{M}_{c_r}^{-1}\mathbf{A} = (\mathbf{I} - \mathbf{M}_{ilu}^{-1}\mathbf{A})(\mathbf{I} - \mathbf{M}^{-1}\mathbf{A}) \\ \mathbf{I} - \mathbf{M}_{c_l}^{-1}\mathbf{A} = (\mathbf{I} - \mathbf{M}^{-1}\mathbf{A})(\mathbf{I} - \mathbf{M}_{ilu}^{-1}\mathbf{A}) \end{cases}, \tag{30}$$

and $(\mathbf{I}-\mathbf{M}_{ilu}^{-1}\mathbf{A})(\mathbf{I}-\mathbf{M}^{-1}\mathbf{A})$ has the same nonzero eigenvalues as $(\mathbf{I}-\mathbf{M}^{-1}\mathbf{A})(\mathbf{I}-\mathbf{M}_{ilu}^{-1}\mathbf{A})$, the eigenvalues of $\mathbf{I}-\mathbf{M}_{c_r}^{-1}\mathbf{A}$ and $\mathbf{I}-\mathbf{M}_{c_l}^{-1}\mathbf{A}$ are the same. This implies

$$\rho(\mathbf{I} - \mathbf{M}_{c_r}^{-1}\mathbf{A}) = \rho(\mathbf{I} - \mathbf{M}_{c_l}^{-1}\mathbf{A}),$$

i.e.

$$\rho(\mathbf{M}_{c_r}^{-1}\mathbf{N}_{c_r}) = \rho(\mathbf{M}_{c_l}^{-1}\mathbf{N}_{c_l}).$$

Thus, for the fixed point iteration, the same convergence rate will be obtained by using either $\mathbf{M}_{c_r}$ or $\mathbf{M}_{c_l}$ as preconditioner. □

For the fixed point iteration, from Theorem 3.5 we can see that there is no difference in convergence rate between using the preconditioner $\mathbf{M}_{c_r}$ or $\mathbf{M}_{c_l}$. For preconditioned Krylov subspace method, we can also expect that the two combination approaches will be nearly the same. This is exactly what we have observed in the numerical tests.

For a special class of matrix which often arise from discretization of elliptic and parabolic differential equations, the following theorem reveals that the fixed point iteration (29) associated with the composite preconditioners are convergent, and the converges faster than just using $\mathbf{ILU}(\mathbf{0})$ preconditioner or filtering preconditioner $\mathbf{M}$. We first recall a useful result which will be used in our proof. It has been established in [5] in more general operator setting,

Lemma 3.3 [Ashby, Holst, Manteuffel and Saylor [5]]

If $\mathbf{A}$ is symmetric positive definite and $\mathbf{G}$ is $\mathbf{A}$-self-adjoint in the sense that $(\mathbf{Gu}, \mathbf{v})_{\mathbf{A}} = (\mathbf{u}, \mathbf{Gv})_{\mathbf{A}}$, then

$$||\mathbf{G}||_{\mathbf{A}} = \rho(\mathbf{G}).$$

Theorem 3.6

Assume $\mathbf{A}$ is symmetric M-matrix, then the fixed point iteration (29) associated with the composite preconditioners are convergent, i.e.

$$\rho(\mathbf{M}_c^{-1}\mathbf{N}_c) \leq \rho(\mathbf{M}_{ilu}^{-1}\mathbf{N}_{ilu}) \cdot \rho(\mathbf{M}^{-1}\mathbf{N}) < 1,$$

where $c = c_r, c_l$.

Proof

Firstly, for symmetric M-matrix $\mathbf{A}$, the splitting associated with $\mathbf{M}_{ilu}$ preconditioner is regular splitting and thus convergent [14], i.e. $\rho(\mathbf{M}_{ilu}^{-1}\mathbf{N}_{ilu}) < 1$. Secondly, from the definition of M-matrix, we have $\mathbf{A}$ is symmetric positive definite, i.e. $\mathbf{A} \succ 0$. Therefore, from Theorem 3.1 we also have $\rho(\mathbf{M}^{-1}\mathbf{N}) < 1$. Thirdly, as

$$((\mathbf{I} - \mathbf{M}^{-1}\mathbf{A})\mathbf{u}, \mathbf{v})_{\mathbf{A}} = (\mathbf{u}, (\mathbf{I} - \mathbf{M}^{-1}\mathbf{A})\mathbf{v})_{\mathbf{A}}$$

and

$$((\mathbf{I} - \mathbf{M}_{ilu}^{-1}\mathbf{A})\mathbf{u}, \mathbf{v})_{\mathbf{A}} = (\mathbf{u}, (\mathbf{I} - \mathbf{M}_{ilu}^{-1}\mathbf{A})\mathbf{v})_{\mathbf{A}},$$

where $\mathbf{u}, \mathbf{v} \in \mathcal{R}^n$ and $(\mathbf{u}, \mathbf{v})_{\mathbf{A}}$ is the inner product induced by SPD matrix $\mathbf{A}$. So both $\mathbf{I} - \mathbf{M}^{-1}\mathbf{A}$ and $\mathbf{I} - \mathbf{M}_{ilu}^{-1}\mathbf{A}$ are self-adjoint (or symmetric) with respect to the inner product induced by matrix $\mathbf{A}$. Then based on Lemma 3.3, we have

$$||\mathbf{I} - \mathbf{M}^{-1}\mathbf{A}||_{\mathbf{A}} = \rho(\mathbf{I} - \mathbf{M}^{-1}\mathbf{A})$$

and

$$||\mathbf{I} - \mathbf{M}^{-1}\mathbf{A}||_{\mathbf{A}} = \rho(\mathbf{I} - \mathbf{M}_{ilu}^{-1}A).$$

Therefore

$$\begin{aligned}
\rho(\mathbf{M}_c^{-1}\mathbf{N}_c) &= \rho(\mathbf{I} - \mathbf{M}_c^{-1}\mathbf{A}) \\
&\leq ||\mathbf{I} - \mathbf{M}_c^{-1}\mathbf{A}||_{\mathbf{A}} \\
&\leq ||\mathbf{I} - \mathbf{M}^{-1}\mathbf{A}||_{\mathbf{A}} \cdot ||\mathbf{I} - \mathbf{M}_{ilu}^{-1}\mathbf{A}||_{\mathbf{A}} \\
&= \rho(\mathbf{I} - \mathbf{M}^{-1}\mathbf{A}) \cdot \rho(\mathbf{I} - \mathbf{M}_{ilu}^{-1}\mathbf{A}) \\
&= \rho(\mathbf{M}_{ilu}^{-1}\mathbf{N}_{ilu}) \cdot \rho(\mathbf{M}^{-1}\mathbf{N}) \\
&< 1.
\end{aligned}$$

The proof is complete. □

The following corollary is an immediately consequence of Theorem 3.6, the proof is omitted.

Corollary 4.1

Assume $\mathbf{A}$ is symmetric M-matrix, then

$$\rho(\mathbf{M}_c^{-1}\mathbf{N}_c) \leq |1 - \lambda_{min}(\mathbf{M}^{-1}\mathbf{A})| \cdot |1 - \lambda_{min}(\mathbf{M}_{ilu}^{-1}\mathbf{A})|,$$

where $c = c_r, c_l$.

From the spectrum distribution plots (in Appendix), it is easy to see that even though sometimes $\lambda_{min}(\mathbf{M}^{-1}\mathbf{A})$ and $\lambda_{min}(\mathbf{M}_{ilu}^{-1}\mathbf{A})$ are close to zero, whereas $\lambda_{min}(\mathbf{M}_c^{-1}\mathbf{A})$ can be well separated from zero. This implies that the fixed point iteration associated with the composite preconditioner should be much faster than that of $\mathbf{M}_{ilu}$ or $\mathbf{M}$.

Subsequently, we give explicit forms of the composite preconditioners, and discuss some interesting properties. From the definition of $\mathbf{M}_{c_l}^{-1} = \mathbf{M}^{-1} + \mathbf{M}_{ilu}^{-1} - \mathbf{M}^{-1}\mathbf{A}\mathbf{M}_{ilu}^{-1}$, we have

$$\begin{aligned}
\mathbf{M}_{c_l}^{-1} &= \mathbf{M}^{-1}(\mathbf{I} - \mathbf{A}\mathbf{M}_{ilu}^{-1}) + \mathbf{M}_{ilu}^{-1} \\
&= \mathbf{M}^{-1}(\mathbf{N}_{ilu}\mathbf{M}_{ilu}^{-1}) + \mathbf{M}_{ilu}^{-1} \\
&= (\mathbf{M}^{-1}\mathbf{N}_{ilu} + \mathbf{I})\mathbf{M}_{ilu}^{-1} \\
&= \mathbf{M}^{-1}(\mathbf{N}_{ilu} + \mathbf{M})\mathbf{M}_{ilu}^{-1}.
\end{aligned}$$

Thus, the composite preconditioner $\mathbf{M}_{c_l}$ has an explicit form of

$$\mathbf{M}_{c_l} = \mathbf{M}_{ilu}(\mathbf{N}_{ilu} + \mathbf{M})^{-1}\mathbf{M}. \tag{31}$$

By similar procedure, the composite preconditioner $\mathbf{M}_{c_r}$ has the form of

$$\mathbf{M}_{c_r} = \mathbf{M}(\mathbf{N} + \mathbf{M}_{ilu})^{-1}\mathbf{M}_{ilu}.$$

For $\mathbf{A} \succeq 0$, it is easy to see that the composite preconditioners are generally nonsymmetric. The following theorems show that under certain assumptions, $\mathbf{A}\mathbf{M}_{c_l}^{-1}$ is symmetric under the meaning of certain nonstandard inner product.

Theorem 3.7

For $\mathbf{A} \succ 0$, and filtering preconditioner $\mathbf{M}$ constructed by using the same left and right filtering vectors, assume $\mathbf{N}_{ilu} \succ 0$, then we have $(\mathbf{M}_{c_l} - \mathbf{A})\mathbf{N}_{ilu}^{-1}\mathbf{M}_{ilu} \succeq 0$, and $\mathbf{I} - \mathbf{A}\mathbf{M}_{c_l}^{-1}$ is symmetric positive semidefinite with respect to the inner product defined by the symmetric positive definite matrix $\mathbf{M}_{ilu}^{-1}(\mathbf{N}_{ilu} + \mathbf{N}_{ilu}\mathbf{M}^{-1}\mathbf{N}_{ilu})\mathbf{M}_{ilu}^{-1}$.

Proof.

By the assumption and Lemma 3.1, we have

$$\begin{aligned}
\mathbf{A}^{-1} \succeq \mathbf{M}^{-1} &\iff \mathbf{N}_{ilu}\mathbf{A}^{-1}\mathbf{N}_{ilu} + \mathbf{N}_{ilu} \succeq \mathbf{N}_{ilu}\mathbf{M}^{-1}\mathbf{N}_{ilu} + \mathbf{N}_{ilu} \\
&\iff \mathbf{N}_{ilu}(\mathbf{A}^{-1}\mathbf{N}_{ilu} + \mathbf{I}) \succeq \mathbf{N}_{ilu}(\mathbf{M}^{-1}\mathbf{N}_{ilu} + \mathbf{I}) \\
&\iff [\mathbf{N}_{ilu}(\mathbf{M}^{-1}\mathbf{N}_{ilu} + \mathbf{I})]^{-1} \succeq [\mathbf{N}_{ilu}(\mathbf{A}^{-1}\mathbf{N}_{ilu} + \mathbf{I})]^{-1} \\
&\iff (\mathbf{I} + \mathbf{M}^{-1}\mathbf{N}_{ilu})^{-1}\mathbf{N}_{ilu}^{-1} \succeq (\mathbf{A}^{-1}\mathbf{N}_{ilu} + \mathbf{I})^{-1}\mathbf{N}_{ilu}^{-1} \\
&\iff \mathbf{M}_{ilu}^{-1}\mathbf{M}_{ilu}(\mathbf{M}^{-1}\mathbf{N}_{ilu} + \mathbf{I})^{-1}\mathbf{N}_{ilu}^{-1} \succeq \mathbf{M}_{ilu}^{-1}\mathbf{M}_{ilu}(\mathbf{A}^{-1}\mathbf{N}_{ilu} + \mathbf{I})^{-1}\mathbf{N}_{ilu}^{-1} \\
&\iff \mathbf{M}_{ilu}^{-1}(\mathbf{M}_{c_l} - \mathbf{A})\mathbf{N}_{ilu}^{-1} \succeq 0 \\
&\iff (\mathbf{M}_{c_l} - \mathbf{A})\mathbf{N}_{ilu}^{-1}\mathbf{M}_{ilu} \succeq 0.
\end{aligned}$$

Based on the above analysis and (31), we have $(\mathbf{I} - \mathbf{A}\mathbf{M}_{c_l}^{-1})\mathbf{M}_{c_l}\mathbf{N}_{ilu}^{-1}\mathbf{M}_{ilu} \succeq 0$, and thus

$$\begin{aligned}(\mathbf{I} - \mathbf{A}\mathbf{M}_{c_l}^{-1})\mathbf{M}_{c_l}\mathbf{N}_{ilu}^{-1}\mathbf{M}_{ilu} &= (\mathbf{I} - \mathbf{A}\mathbf{M}_{c_l}^{-1})\mathbf{M}_{ilu}(\mathbf{N}_{ilu} + \mathbf{M})^{-1}\mathbf{M}\mathbf{N}_{ilu}^{-1}\mathbf{M}_{ilu} \\ &= (\mathbf{I} - \mathbf{A}\mathbf{M}_{c_l}^{-1})\mathbf{M}_{ilu}(\mathbf{M}^{-1}\mathbf{N}_{ilu} + \mathbf{I})^{-1}\mathbf{N}_{ilu}^{-1}\mathbf{M}_{ilu} \\ &= (\mathbf{I} - \mathbf{A}\mathbf{M}_{c_l}^{-1})\mathbf{M}_{ilu}(\mathbf{N}_{ilu} + \mathbf{N}_{ilu}\mathbf{M}^{-1}\mathbf{N}_{ilu})^{-1}\mathbf{M}_{ilu} \\ &\succeq 0. \end{aligned} \tag{32}$$

Noting from Lemma 3.1 that $\mathbf{M} \succ 0$, and with the assumption $\mathbf{N}_{ilu} \succ 0$, we have $\mathbf{M}_{ilu}(\mathbf{N}_{ilu} + \mathbf{N}_{ilu}\mathbf{M}^{-1}\mathbf{N}_{ilu})^{-1}\mathbf{M}_{ilu} \succ 0$. Thus

$$\mathbf{M}_{ilu}^{-1}(\mathbf{N}_{ilu} + \mathbf{N}_{ilu}\mathbf{M}^{-1}\mathbf{N}_{ilu})\mathbf{M}_{ilu}^{-1}(\mathbf{I} - \mathbf{A}\mathbf{M}_{c_l}^{-1}) \succeq 0.$$

Therefore, $\mathbf{I}-\mathbf{A}\mathbf{M}_{c_l}^{-1}$ is symmetric positive semidefinite with respect to the inner product defined by symmetric positive definite matrix $\mathbf{M}_{ilu}^{-1}(\mathbf{N}_{ilu}+\mathbf{N}_{ilu}\mathbf{M}^{-1}\mathbf{N}_{ilu})\mathbf{M}_{ilu}^{-1}$. □

For composite preconditioner $\mathbf{M}_{c_r}$, the following theorem can be proved similarly.

Theorem 3.8
For $\mathbf{A} \succ 0$, and filtering preconditioner $\mathbf{M}$ constructed by using the same left and right filtering vectors, assume $\mathbf{N}_{ilu} \succ 0$, then we have $(\mathbf{M}_{c_r} - \mathbf{A})\mathbf{N}^{-1}\mathbf{M} \succeq 0$, and $\mathbf{I}-\mathbf{A}\mathbf{M}_{c_r}^{-1}$ is symmetric positive semidefinite with respect to the inner product defined by the symmetric positive definite matrix $\mathbf{M}^{-1}(\mathbf{N}+\mathbf{N}\mathbf{M}_{ilu}^{-1}\mathbf{N})\mathbf{M}^{-1}$.

In the following section, we will show the eigenvalue distribution obtained by different preconditioners. The spectrum distributions of several representative matrices in our test sets are displayed from Figures 1 to 5 in the Appendix of this paper. From these figures we can see that the composite preconditioners tend to make the spectrum clustered at 1. In the symmetric case, even if complex eigenvalues appear due to the nonsymmetric composite preconditioner, their imaginary parts are usually very small.

4 Numerical results

In this section, we present some numerical results to illustrate the performance of the preconditioners discussed in this paper. The preconditioned **FGMRES** method is employed as the linear system solver. The performance of composite preconditioners is compared with $\mathbf{M}_{ilu}$. Several different approaches of constructing the filtering preconditioner $\mathbf{M}$ are considered, the meaning of the notations are described below. The combination approach (24) is used for all the composite preconditioners, hence they all have the left filtering property.

$\mathbf{M}_c$: Combine $\mathbf{M}_{ilu}$ with filtering preconditioner $\mathbf{M}$, where $\mathbf{M}$ is constructed by using two sides filtering approach (20) and (21), both filtering vectors are chosen as *ones*.

$\mathbf{M}_{c1r}$: Combine $\mathbf{M}_{ilu}$ with filtering preconditioner $\mathbf{M}$, where $\mathbf{M}$ is constructed by using two sides filtering approach (20) and (21), the left filtering vector is chosen as *ones*, and the right filtering vector is chosen as the Ritz vector (corresponding to the smallest Ritz value) computed at the kth step of GMRES preconditioned by $\mathbf{M}_{ilu}$.

$\mathbf{M}_{cr1}$: Combine $\mathbf{M}_{ilu}$ with filtering preconditioner $\mathbf{M}$, where $\mathbf{M}$ is constructed by just using the right filtering approach (8), and the filtering vector is chosen as *ones*, i.e. the same with the preconditioner constructed in [2], except using *ones* instead of the Ritz vector as the filtering vector.
$\mathbf{M}_{cl1}$: Combine $\mathbf{M}_{ilu}$ with filtering preconditioner $\mathbf{M}$, where $\mathbf{M}$ is constructed by just using the left filtering approach (10), and the filtering vector is chosen as *ones*.
$\mathbf{M}_{crr}$: Combine $\mathbf{M}_{ilu}$ with filtering preconditioner $\mathbf{M}$, where $\mathbf{M}$ is constructed by just using the right filtering approach (8), the filtering vector is chosen as the Ritz vector computed at the kth step of GMRES preconditioned by $\mathbf{M}_{ilu}$ [2].

For symmetric problems, the preconditioners $\mathbf{M}_{cr1}$, $\mathbf{M}_{cl1}$ and $\mathbf{M}_c$ are equivalent when the same filtering vector is used. Therefore, just $\mathbf{M}_c$ is displayed in the tables for symmetric problems. For comparisons, the step k used to construct the Ritz vector is set to be 25 and 50, respectively.

In the tests, we stop the algorithm when the relative norm $\frac{||\mathbf{b}-\mathbf{A}\mathbf{x}_k||}{||\mathbf{b}||}$ is less than 10^{-12}. The exact solution is generated randomly. Unless special explanations, the initial approximate solution is always chosen to ensure that the sum of the residual vectors are zero all throughout the iterations, see Theorem 3.4. In the following tables, *iter* denotes the number of iterations, *error* denotes the infinite norm of the difference between the final approximate solution and the exact solution, and *t.c* denotes the total cost (the number of the preconditioner solves or the number of matrix vector products). We use "-" to denote that the method fails to converge within 200 iteration steps. As the different subspace dimension to compute the Ritz vectors has no influence on $\mathbf{M}_i$, $\mathbf{M}_c$, $\mathbf{M}_{cl1}$ and $\mathbf{M}_{cr1}$, thus the same result only appear one time in each table. For the $\mathbf{M}_{ilu}$ preconditioner, every iteration requires only one preconditioner solve, so the total preconditioner solve is equal to the iteration number. Therefore, just *iter* is presented in the tables for $\mathbf{M}_{ilu}$ preconditioner. For the composite preconditioners, the actual cost is presented, which is equal to the total number of precoditioner solves, assuming that the **ILU**(**0**) preconditioner has the same cost with the filtering preconditioner.

4.1 Description of the tested problems

We consider the boundary value problem as in [2]

$$\begin{aligned} \eta(x)u + div(\mathbf{a}(x)u) - div(\kappa(x)\nabla u) &= f \ \ in \ \ \Omega \\ u &= 0 \ \ on \ \ \partial\Omega_D \\ \tfrac{\partial u}{\partial n} &= 0 \ \ on \ \ \partial\Omega_N \end{aligned} \tag{33}$$

where $\Omega = [0,1]^n$ ($n = 2$, or 3), $\partial\Omega_N = \partial\Omega \setminus \partial\Omega_D$. The function η, the vector field $\mathbf{a}$, and the tensor κ are the given coefficients of the partial differential operator. In 2D case, we have $\partial\Omega_D = [0,1] \times \{0,1\}$, and in 3D case, we have $\partial\Omega_D = [0,1] \times \{0,1\} \times [0,1]$.

The following five cases are considered:

`Case 4.1:` *The advection-diffusion problem with a rotating velocity in two dimensions:*

The tensor κ is the identity, and the velocity is $\mathbf{a} = (2\pi(x_2 - 0.5), 2\pi(x_1 - 0.5))^T$. The function η is zero. The uniform grid with $n \times n$ nodes, $n = 100, 200, 300, 400$ nodes are tested respectively. Table 4 displays the results obtained by using different preconditioners.

Table 1: Results for `Case 4.2`, non-Homogeneous problems in two dimensions; symmetric

	$\mathbf{M}_{ilu}$		$\mathbf{M}_c$			$\mathbf{M}_{c1r}$			$\mathbf{M}_{crr}$		
$1/h$	iter	*error*	*iter*	*error*	*t.c*	*iter*	*error*	*t.c*	*iter*	*error*	*t.c*
100	107	1.4e-9	26	1.1e-10	**52**	**25**	1.1e-10	50+50	**25**	6.8e-11	50+50
100						**25**	8.3e-10	50+25	**25**	9.5e-11	50+25
200	187	2.7e-9	37	6.3e-10	**74**	36	2.3e-10	72+50	**35**	2.9e-11	70+50
200						36	6.3e-10	72+25	**35**	5.4e-11	70+25
300	-	-	45	8.0e-10	**90**	44	5.9e-10	88+50	**42**	3.7e-10	84+50
300						45	5.7e-10	90+25	**44**	7.0e-10	88+25
400	-	-	52	1.2e-9	**104**	51	9.8e-10	102+50	**50**	5.7e-9	100+50
400						52	9.3e-10	104+25	**51**	1.2e-9	102+25

`Case 4.2:` *Non-Homogenous problems with large jumps in the coefficients in two dimensions:*

The coefficient η and $\mathbf{a}$ are both zero. The tensor κ is isotropic and discontinuous. It jumps from the constant value 10^3 in the ring $\frac{1}{2\sqrt{2}} \leq |x - c| \leq \frac{1}{2}$, $c = (\frac{1}{2}, \frac{1}{2})^T$, to 1 outside. We tested uniform grids with $n \times n$ nodes, $n = 100, 200, 300, 400$. Table 1 displays the results obtained by using different preconditioners.

`Case 4.3:` *Skyscraper problems:*
The tensor κ is isotropic and discontinuous. The domain contains many zones of high permeability which are isolated from each other. Let $[x]$ denote the integer value of x. In 2D, we have

$$\kappa(x) = \begin{cases} 10^3 * ([10 * x_2] + 1), & if \ \ [10 * x_i] = 0 \ mod(2) \ , \ i = 1, 2, \\ 1, & otherwise. \end{cases}$$

and in 3D

$$\kappa(x) = \begin{cases} 10^3 * ([10 * x_2] + 1), & if \ \ [10 * x_i] = 0 \ mod(2) \ , \ i = 1, 2, 3, \\ 1, & otherwise. \end{cases}$$

Table 2 displays the results obtained by using different preconditioners for 2D and 3D problems.

`Case 4.4:` *Convective skyscraper problems:*
The same with the Skyscraper problems except that the velocity feild is changed to be $\mathbf{a} = (1000, 1000, 1000)^T$. The tested results are displayed in Table 3.

Table 2: Results for `Case 4.3`, skyscrapers problems in two (top) and three (bottom) dimensions; symmetric

	$\mathbf{M}_{ilu}$		$\mathbf{M}_c$			$\mathbf{M}_{c1r}$			$\mathbf{M}_{crr}$		
$1/h$	iter	*error*	*iter*	*error*	*t.c*	*iter*	*error*	*t.c*	*iter*	*error*	*t.c*
100	-	-	**26**	6.8e-7	**52**	**26**	4.7e-6	52+50	36	3.3e-6	72+50
100						44	5.1e-7	88+25	29	3.1e-7	58+25
200	-	-	**39**	1.5e-6	**78**	45	1.6e-6	90+50	40	1.3e-6	80+50
200						87	8.9e-8	174+25	143	5.7e-7	286+25
300	-	-	**46**	1.5e-6	**92**	85	3.4e-6	170+50	53	8.2e-7	106+50
300						130	1.5e-7	260+25	179	8.1e-8	358+25
400	-	-	**60**	3.7e-6	**120**	193	3.0e-6	386+50	156	6.1e-6	312+50
400						161	1.3e-6	322+25	200	4.0e-3	400+25
20	125	2.3e-8	**11**	1.0e-8	**22**	**11**	3.6e-9	22+50	13	8.2e-9	26+50
20						**10**	1.2e-8	20+25	14	2.2e-9	28+25
30	198	1.3e-7	**14**	5.6e-8	**28**	22	9.8e-8	44+50	27	1.1e-9	54+50
30						15	1.0e-8	30+25	27	4.0e-10	54+25
40	-	-	**15**	5.3e-7	**30**	16	1.3e-7	32+50	26	6.8e-9	52+50
40						**15**	6.7e-8	30+25	25	8.1e-8	50+25

`Case 4.5:` *Anisotropic layers:*
The domain is made of 10 anisotropic layers with jumps of up to four orders of magnitude and an anisotopy ratio of up to 10^3 in each layer. For 3D problem, the cube is divided in to 10 layers parallel to $z = 0$, of size 0.1, in which the coefficients are constant. The coefficient κ_x in the ith layer is given by $v(i)$, the latter being the ith component of the vector $v = [\alpha, \beta, \alpha, \beta, \alpha, \beta, \gamma, \alpha, \alpha]$, where $\alpha = 1$, $\beta = 10^2$ and $\gamma = 10^4$. We have $\kappa_y = 10\kappa_x$ and $\kappa_z = 1000\kappa_x$. The velocity field is zero. Numerical results are shown in Table 5.

From Table 1 - 5 we can see that $\mathbf{M}_c$ and $\mathbf{M}_{cl1}$ produce the best results for most of the nonsymmetric problems. Particularly, for 2D problems, both methods need nearly the same iteration numbers, whereas $\mathbf{M}_c$ needs two sides computation of the local approximations, and $\mathbf{M}_{cl1}$ needs one side computation. Hence $\mathbf{M}_{cl1}$ is more appealing than $\mathbf{M}_c$. For 3D problems, $\mathbf{M}_c$ is faster than $\mathbf{M}_{cl1}$ in terms of iteration numbers. For the symmetric problems $\mathbf{M}_c$, $\mathbf{M}_{cr1}$ and $\mathbf{M}_{cl1}$ are equivalent, and their numerical results are always better than using Ritz vector as the filtering vector. From the tested problems, we conclude that,
• if just one side filtering is adopted, left filtering is generally better than right filtering.
• using *ones* as the filtering vectors is very effective, especially as the left filtering vector.
• two sides filtering approach is best for 3-D nonsymmetric problems.

5 Concluding remarks

In this paper, we have discussed the left and two sides tangential filtering decompositions. The filtering preconditioner constructed by the introduced decompo-

Table 3: Results for `Case 4.4`, convective skyscrapers in two (top) and three (bottom) dimensions; nonsymmetric

	$\mathbf{M}_{ilu}$		$\mathbf{M}_{c}$			$\mathbf{M}_{c1r}$			$\mathbf{M}_{crr}$			$\mathbf{M}_{cl1}$			$\mathbf{M}_{cr1}$		
$1/h$	*iter*	*error*	*iter*	*error*	*t.c*	*iter*	*error*	*t.c*	*iter*	*error*	*t.c*	*iter*	*error*	*t.c*	*iter*	*error*	*t.c*
100	181	1.1e-8	**19**	1.6e-10	**38**	**19**	1.8e-10	38+50	22	2.0e-8	44+50	**19**	1.0e-10	**38**	22	2.0e-9	44
100						**19**	1.9e-10	38+25	22	5.8e-9	44+25						
200	-	-	**26**	1.8e-8	**52**	26	2.3e-8	52+50	32	1.4e-7	64+50	**26**	1.4e-8	**52**	30	8.0e-8	60
200						27	1.5e-8	54+25	34	3.1e-8	68+25						
300	-	-	**28**	6.5e-8	**56**	33	7.3e-8	66+50	39	9.3e-8	78+50	**28**	1.5e-8	**56**	36	9.7e-8	72
300						38	5.1e-8	76+25	98	4.8e-8	196+25						
400	-	-	40	1.3e-8	80	42	1.0e-7	84+50	121	1.4e-7	242+50	**38**	6.0e-8	**76**	52	4.7e-7	104
400						97	6.6e-8	194+25	126	2.1e-7	252+25						
20	64	6.7e-10	**6**	1.3e-10	**12**	**6**	5.4e-10	12+50	11	5.5e-11	22+50	9	1.2e-11	18	10	4.7e-11	20
						7	9.4e-11	14+25	14	1.6e-10	28+25						
30	105	2.9e-9	**12**	6.0e-11	**24**	**15**	1.2e-9	30+50	19	6.5e-10	38+50	32	6.5e-10	64	15	3.4e-10	30
30						**13**	1.0e-10	26+25	17	4.1e-10	34+25						
40	114	3.5e-9	**10**	7.7e-11	**20**	11	2.7e-10	22+50	13	3.4e-9	26+50	13	2.7e-12	26	13	4.8e-10	26
40						14	6.1e-10	28+25	22	1.7e-9	44+25						

Table 4: Results for `Case 4.1`, advection-diffusion problem in two dimensions; nonsymmetric

	$\mathbf{M}_{ilu}$		$\mathbf{M}_{c}$			$\mathbf{M}_{c1r}$			$\mathbf{M}_{crr}$			$\mathbf{M}_{cl1}$			$\mathbf{M}_{cr1}$		
$1/h$	iter	*error*	*iter*	*error*	*t.c*	*iter*	*error*	*t.c*	*iter*	*error*	*t.c*	*iter*	*error*	*t.c*	*iter*	*error*	*t.c*
100	107	1.1e-9	27	8.5e-11	**54**	26	9.4e-11	52+50	**25**	1.5e-10	50+50	26	1.3e-10	52	26	1.2e-10	52
100						26	9.0e-11	52+25	**25**	8.3e-11	50+25						
200	197	5.6e-9	38	4.1e-10	**76**	35	6.3e-10	70+50	**34**	3.5e-10	68+50	37	4.3e-10	74	37	3.1e-10	74
200						37	6.4e-10	74+25	**35**	2.8e-10	70+25						
300	-	-	46	6.1e-10	**92**	44	4.5e-10	88+50	**42**	5.1e-10	84+50	45	6.0e-10	90	45	6.6e-10	90
300						45	1.2e-9	90+25	**44**	6.3e-10	88+25						
400	-	-	52	1.5e-9	**104**	51	1.2e-9	102+50	**50**	8.5e-10	100+50	52	1.4e-9	**104**	52	1.2e-9	**104**
400						52	1.2e-9	104+25	**51**	1.1e-9	102+25						

Table 5: Results for `Case 4.5`, anisotropic layers in two (top) and three (bottom) dimensions; symmetric

	$\mathbf{M}_{ilu}$		$\mathbf{M}_c$			$\mathbf{M}_{c1r}$			$\mathbf{M}_{crr}$		
$1/h$	iter	*error*	*iter*	*error*	*t.c*	*iter*	*error*	*t.c*	*iter*	*error*	*t.c*
100	188	5.2e-7	**18**	1.2e-7	**36**	20	3.4e-7	40+50	21	4.3e-7	42+50
100						52	6.9e-7	104+25	24	5.0e-7	48+25
200	-	-	**29**	2.3e-8	**58**	34	2.2e-6	68+50	38	1.8e-6	76+50
200						33	1.8e-7	66+25	34	1.4e-7	68+25
300	-	-	**40**	1.8e-7	**80**	72	2.6e-6	144+50	51	1.4e-5	102+50
300						63	1.8e-6	126+25	66	5.2e-6	132+25
400	-	-	**51**	3.0e-8	**102**	108	9.9e-6	216+50	52	1.4e-6	104+50
400						65	1.1e-7	130+25	93	1.6e-7	186+25
20	25	8.7e-8	**10**	1.5e-8	**20**	14	1.4e-8	28+50	**10**	4.4e-7	20+50
20						13	1.1e-8	26+25	**10**	2.3e-8	20+25
30	33	3.9e-7	**11**	7.8e-8	**22**	17	1.3e-7	34+50	**11**	4.7e-8	22+50
30						12	8.6e-8	24+25	**11**	1.0e-7	60+25
40	40	9.3e-7	**11**	1.6e-7	**22**	43	1.6e-7	86+50	14	1.3e-8	28+50
40						35	9.26e-8	70+25	13	3.0e-8	26+25

sition is combined with the classical **ILU**(**0**) preconditioner in multiplicative ways. The composite preconditioners are very efficient in damping the high and low frequency modes, and thus perform very well for the block tridiagonal linear systems arising from the discretization of **PDE** problems on Cartesian grids. On the filtering vector, we adopt *ones* as the filtering vector in this paper. There are several advantages of this choice. First, it is as efficient as other vector choices, and the preprocessing that is needed to construct the filtering preconditioner can be saved, second, using *ones* as the left filtering vector is able to enable the zero material balance error all throughout the iterations, which is important to improve the convergence. The framework of constructing the preconditioner discussed in this paper is interesting. As further work, we are interested in extending these preconditioning techniques to the problems arising from the discretization of **PDE**s on unstructured grids, in conjunction with the matrix reordering techniques.

6 Appendix

The eigenvalue distribution of the preconditioned matrix is plotted in the following figures. The notations used in the figures are as follows:
A: the coefficient matrix
$\mathbf{M}_{ilu}^{-1}\mathbf{A}$: the preconditioned matrix by **ILU**(0) preconditioner.
$\mathbf{M}_{f}^{-1}\mathbf{A}$: the preconditioned matrix by two sides filtering preconditioner proposed in this paper.
$\mathbf{M}_{c}^{-1}\mathbf{A}$: the preconditioned matrix by combination preconditioner (24) (the same as using (23)).

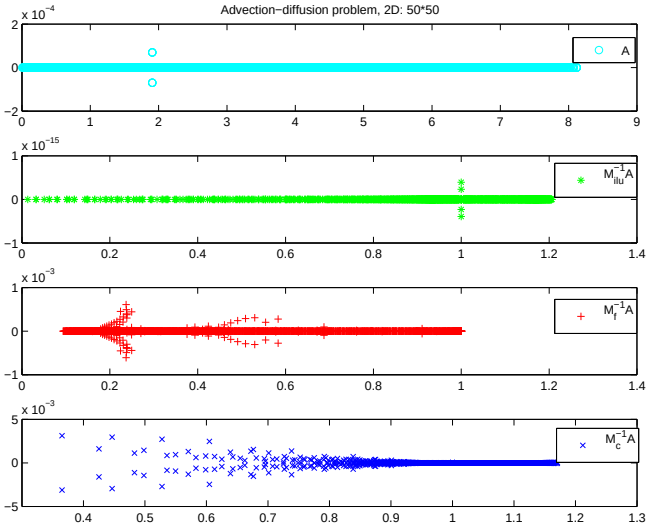

Figure 1: Spectrum distribution of the preconditioned matrix, `Case 4.1`.

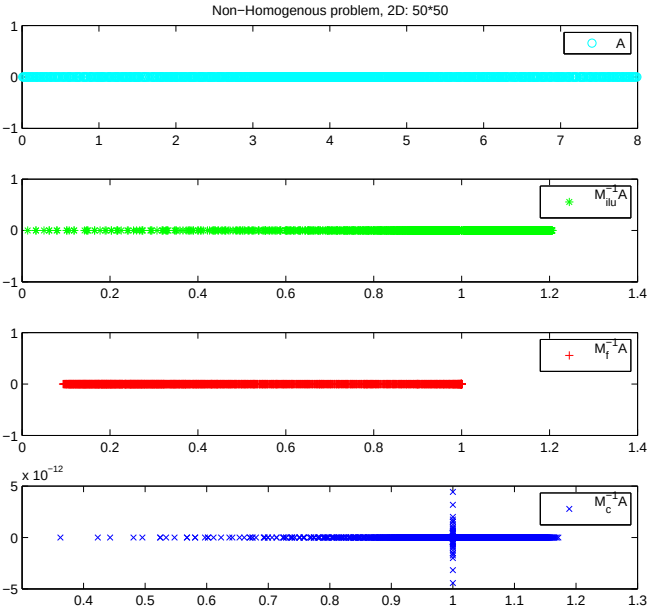

Figure 2: Spectrum distribution of the preconditioned matrix, `Case 4.2`.

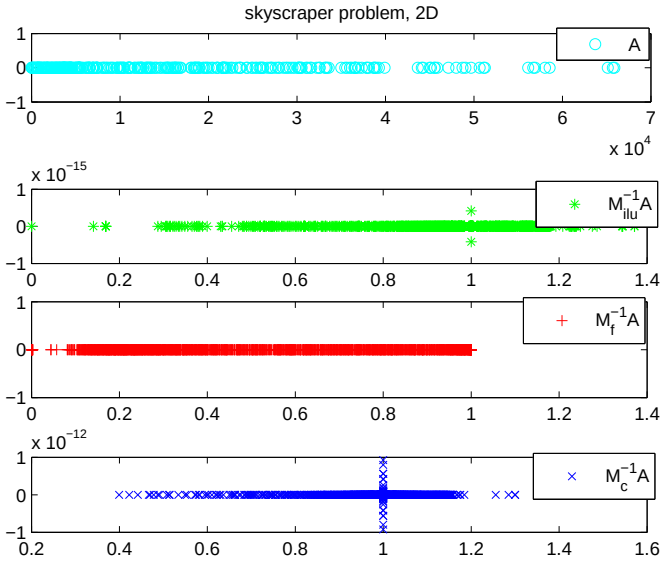

Figure 3: Spectrum distribution of the preconditioned matrix, `Case 4.3`.

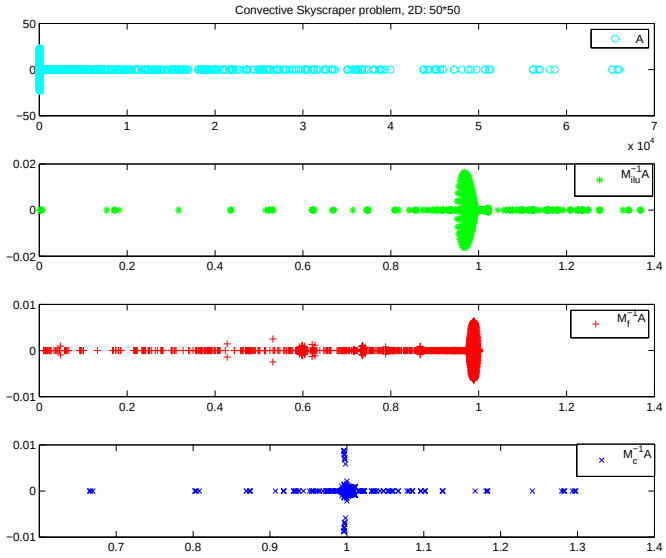

Figure 4: Spectrum distribution of the preconditioned matrix, `Case 4.4`.

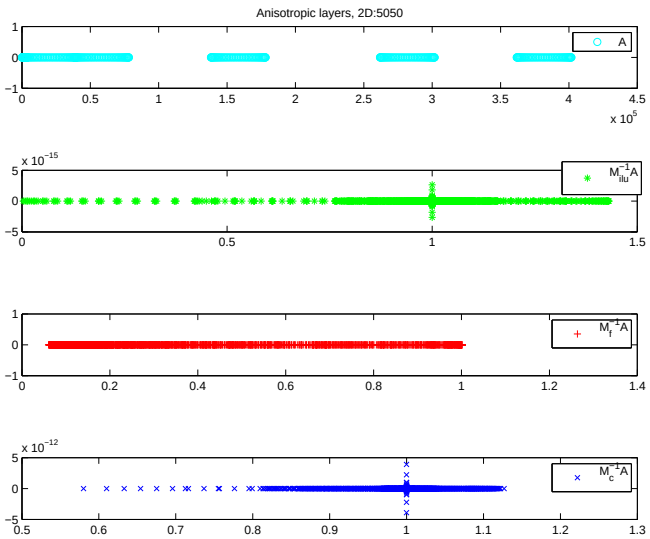

Figure 5: Spectrum distribution of the preconditioned matrix, `Case 4.5`.

References

[1] Y. Achdou and F. Nataf, *An iterated tangential filtering decomposition*, Numer. Linear Algebra Appl., 10, (2003), pp.511-539.

[2] Y. Achdou and F. Nataf, *Low frequency tangential filtering decomposition*, Numer. Linear Algebra Appl., 14, (2007), pp.129-147

[3] J. R. Appleyard, *Proof that simple Colsum modified ILU factorization of matrices arising in Fluid folw problems are not singular*, by John Appleyard B.A., Ph.D, Polyhedron Software Ltd, May 2004.

[4] J. R. Appleyard and I. M. Cheshire, *Nested Factorization*, SPE 12264, presented at the Seventh SPE Symposium on Reservoir Simulation, San Francisco, 1983.

[5] S. F. Ashby, M. J. Holst, T. A. Manteuffel and P. E. Saylor *The role of inner product in stopping criteria for conjugate gradient iterations*, BIT., 41, (2001), pp.26-52.

[6] O. Axelsson, *Iterative solution methods*, Cambridge University Press, New York, 1994.

[7] Z. Bai and M. Ng, *Preconditioners for Nonsymmetric Block-Toeplitz-Like-Plus-Diagonal Linear System*, Numer. Math., 96, (2003), pp. 197-220

[8] A. Buzdin, *Tangential decomposition*, Computing., 61, (1998) pp.257-276.

[9] A. Buzdin and G. Wittum, *Two-frequency decomposition,* Numer. Math., 97, (2004), pp.269-295.

[10] P. Concus, G. H. Golub and G. Meurant, *Block Preconditioning for the Conjugate Gradient method*, SIAM J. Sci. Statist. Comput., 6, (1985), pp.220-252

[11] M. J. Gander and F. Nataf, *AILU: a preconditioner based on the analytic factorization of the elliptic operator*, Numer. Linear Algebra Appl., 7, (2000), pp.505-526.

[12] C. Keller, N. I. M. Gould and A. J. Wathen, *Constraint preconditioning for indefinite linear systems,* SIAM J. Matrix Anal. Appl., 21, (2000), pp. 1300-1317.

[13] J. Liesen and B. N. Parlett, *On nonsymmetric saddle point matrices that allow conjugate gradient iterations* Numer. Math., 108, (2008), pp.605-624.

[14] J. A. Meijerink and H. A. van der Vorst, *An iterative solution method for linear systems of which the coefficient matrix is symmetric M-matrix*, Math. Comput., 137, (1977), pp.148-162.

[15] G. Meurant, *Computer Solution of Large Linear Systems*, North-Holland Publishing Co., Amsterdam, 1999.

[16] G. Meurant, *A review on the inverse of symmetric tridiagonal and block tridiagoal matrices*, SIAM J. Matrix Anal. Appl. 13, (1992), pp.707-728.

[17] Y. Saad, *Iterative Methods for Sparse Linear Systems*, PWS Publishing Company: Boston, MA, 1996.

[18] M. Stoll and A. Wathen, *Combination preconditioning and self-adjointness in non-standard inner products with application to saddle point problems*, Oxford preprint, NA-07/11, 2007.

[19] M. Stoll and A. Wathen, *The Bramble-Pasciak$^+$ preconditioner for saddle point problems*, Oxford preprint, NA-07/13, 2007.

[20] C. Wagner, *Tangential frequency filtering decompositions for symmetric matrices*, Numer. Math., 78, (1997), pp.119-142.

[21] C. Wagner, *Tangential frequency filtering decompositions for unsymmetric matrices* Numer. Math., 78, (1997), pp.143-163.

[22] C. Wagner and G. Wittum, *Adaptive filtering,* Numer. Math., 78, (1997), pp.305-382.

[23] J. R. Wallis, R. P. Kendall and T. E. Little, *Constraint Residual Acceleration of Conjugate Residual Methods*, SPE 13536, presented at the SPE 1985 Reservoir Simulation Symposium held in Dallas, Texas, 1985.

[24] G. Wittum, *Filternde Zerlegungen,* Schnelle Löser fuer grosse Gleichungssysteme. Teubner Skripten zur Numerik, Band 1, Teubner-Verlag, Stuttgart, 1992.

Centre de recherche INRIA Saclay – Île-de-France
Parc Orsay Université - ZAC des Vignes
4, rue Jacques Monod - 91893 Orsay Cedex (France)

Centre de recherche INRIA Bordeaux – Sud Ouest : Domaine Universitaire - 351, cours de la Libération - 33405 Talence Cedex
Centre de recherche INRIA Grenoble – Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier
Centre de recherche INRIA Lille – Nord Europe : Parc Scientifique de la Haute Borne - 40, avenue Halley - 59650 Villeneuve d'Ascq
Centre de recherche INRIA Nancy – Grand Est : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex
Centre de recherche INRIA Paris – Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex
Centre de recherche INRIA Rennes – Bretagne Atlantique : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex
Centre de recherche INRIA Sophia Antipolis – Méditerranée : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399